# Distributed Diagnosis in Formations of Mobile Robots

Matthew J. Daigle, Xenofon D. Koutsoukos, *Member, IEEE,* and Gautam Biswas, *Senior Member, IEEE*

***Abstract*— Multi-robot systems are being increasingly used for a variety of tasks in manufacturing, surveillance, and space exploration. These systems can degrade or develop faults during operation, and, therefore, require online diagnosis algorithms to ensure safe operation. Centralized approaches to online diagnosis of robot formations do not scale well for two primary reasons: (*i*) the computational complexity of the algorithm grows significantly with the number of robots, and (*ii*) the individual robots must communicate a large number of measurements to a central diagnoser. To overcome these problems, we present a distributed, model-based, qualitative fault diagnosis approach for formations of mobile robots. The approach is based on a bond graph modeling framework that can deal with multiple sensor types and isolate process, sensor, and actuator faults. The diagnosis scheme employs relative measurement orderings to discriminate among faults by exploiting the temporal order of measurement deviations. This increases the discriminatory power of the measurement set and produces a more efficient fault isolation algorithm. We describe a distributed diagnoser design algorithm applied to robot formations. Experimental results demonstrate the improvement in both the discriminatory power of the measurements produced by the relative measurement orderings, and the computational efficiency achieved by the distributed diagnosis approach.**

***Index Terms*— Model-based diagnosis, mobile robots, multi-robot systems**

## I. Introduction

AUTONOMOUS multi-robot teams can perform a wide range of collaborative tasks in manufacturing, surveillance, and space exploration. In many cases, the execution of the task requires formation control [1]–[4], and the success of the overall operation depends on each robot operating in an error-free manner. Faults in one robot can propagate to other robots over communication links, and this can cause problems in maintaining the formation required to execute the desired tasks (e.g., collaboratively moving a load). Degradations and faults must be detected and isolated early to allow for reconfiguration and continued operation [5], and this can be achieved only if diagnostic mechanisms are incorporated into multi-robot systems. However, the diagnosis of robot formations is a difficult problem. A global, centralized model is usually needed to capture the interactions that govern the propagation of fault effects between robots, but centralized approaches have many weaknesses. Specifically, these approaches (*i*) create a single point of failure, (*ii*) do not scale well as formation size increases, (*iii*) do not exploit the computational resources available on each robot, and (*iv*) incur large communication overhead.

In this paper, we present a distributed, qualitative fault diagnosis approach for mobile robot formations. Our approach is based on an extended bond graph model [6], which provides a comprehensive framework for modeling the physical components, sensors, and actuators, as well as the communication processes among the robots. The methodology, a generalization of our initial work on two coupled robots presented in [7], is based on the TRANSCEND framework [8], [9] that employs a qualitative approach to fault isolation in dynamic systems. In TRANSCEND, analysis of fault transient behavior is based on fault signatures, which are predicted time-derivative effects of faults on measurements derived from the system model. Faults are isolated by tracking dynamic system behavior and comparing the symbolic magnitude and slope of measurements against predicted fault signatures when faults are detected. We use a qualitative methodology since traditional methods, such as parity relations [10], typically do not apply to multiplicative faults, do not easily extend to nonlinear systems, and are suitable for centralized diagnosis schemes. Discrete-event approaches [11] are hard to apply because they model event-based and not continuous dynamics. Parametric fault effects are difficult to represent as a fixed sequence of discrete changes in measurements. Further, the inability to analyze fault transients may result in a loss of diagnosability, especially for capacity- and inertia-related faults.

Interactions between the robots cause fault effects to propagate across robot boundaries, and, therefore, require additional discriminatory power to isolate all the faults of interest. We solve this problem by introducing the concept of *relative measurement orderings* [12], which is based on the intuition that faults cause deviations in some measurements before others. Relative measurement orderings use the predicted temporal order of measurement deviations to increase the discriminatory power of a set of measurements. A formal diagnosability analysis for single, persistent faults in robot formations shows that a combination of fault signatures and relative measurement orderings increases the discriminatory power of the measurements and facilitates more efficient diagnoses.

Our approach is applicable to rigid formations of mobile robots. In rigid formations there is a strong coupling between the dynamics of the robot behaviors, which is exploited by our algorithm to improve the discriminatory power of the measure-

Manuscript received May 22, 2006; revised October 30, 2006. This work was supported in part by grants NSF CNS-0452067, NSF CNS-0347440, and NSF CNS-0615214.

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: matthew.j.daigle@vanderbilt.edu; xenofon.koutsoukos@vanderbilt.edu; gautam.biswas@vanderbilt.edu).

ments and the efficiency of the diagnosers. The approach may not offer advantages for collaborative multi-robot applications that do not exhibit coupling between the dynamic behavior of individual robots. Specifically, for applications where faults do not propagate between the mobile robots, our method can be still applied but will naturally result in independent diagnosers for each robot. In addition, our approach assumes persistent single faults with an abrupt profile. The single fault assumption is reasonable for many safety-critical engineering systems since the probability of multiple faults occurring simultaneously is very small [10], [13]. Although persistent and abrupt faults may be restrictive, the abrupt fault profile is a good approximation for many practical faults in sensors and actuators.

Other commonly occurring fault profiles include incipient faults that describe slow degradations in actuator performance and sensor drifts. Fault diagnosis of incipient faults has been studied in [14]. Preliminary work on extending the TRANSCEND approach to incipient fault diagnosis is presented in [15]. Further, diagnosis of multiple faults in TRANSCEND is presented in [16]. Incorporating such approaches for diagnosis of formations of mobile robots is an interesting research problem but it is beyond the scope of this paper. Faults can also be intermittent as opposed to persistent, e.g. when wheel slippage occurs. It should be noted that even if such diagnosis methods are available, they should be complemented by diagnosis of single abrupt faults that are more likely, and may have catastrophic consequences if not detected and isolated quickly.

Distributed diagnosis algorithms that extend the basic TRANSCEND scheme are presented in [17]. This paper extends the distributed algorithms to incorporate relative measurement orderings, and this enables each robot to individually determine a globally correct local diagnosis with a small set of measurements. Based on this extended approach, a distributed diagnosis scheme is designed and applied to formations of robots with a local diagnoser on each robot. In contrast to a centralized diagnosis approach, our solution scales well to large formations, minimizes the communication costs associated with fault isolation, takes advantage of the computational resources available on each robot, and avoids the need for a centralized coordinator for the local diagnoses. Experimental results for a system consisting of four robots in formation demonstrate the effectiveness of this approach. The results illustrate the advantages of the method, namely (*i*) scalability, (*ii*) increasing the discriminatory power of the measurements, and (*iii*) improving the efficiency of the distributed diagnosis approach.

Fault diagnosis in continuous systems is a very active area of research, and has been investigated by many researchers (see [10], [18]–[24]). Most work in diagnosis of mobile robots has concentrated on the single-robot case. A survey of such methods can be found in [25]. TRANSCEND has been applied in the single-robot case for diagnosis of actuator faults using fault signatures derived from a simplified bond graph model [26]. The parity relation approach has also been applied to nonlinear single-robot systems in [27]. Particle filtering techniques can be used for nonlinear and hybrid systems, and have been employed in single-robot diagnosis in [28] and [29]. Fault detection in mobile robots has been addressed in [30] by developing a technique which accounts for both kinematic and dynamic behaviors in order to generate better residuals in spite of parametric uncertainty. Work in [31] addressed sensor fault detection and identification using multiple model adaptive estimation based on a bank of Kalman filters. This work was extended in [32] by using a neural network to detect and identify both sensor and mechanical failures based on the output of the filter bank.

In contrast to previous work in mobile robot diagnosis, our approach can be applied efficiently to diagnosis of process, sensor, and actuator faults in robot formations in a distributed fashion by employing relative measurement orderings. In addition to easily handling multiplicative faults, our approach qualifies residuals with a richer feature set than parity relations approaches and incorporates temporal information, resulting in increased discriminatory power of the measurements. Quantitative techniques, like particle filtering, do not scale well with the number of possible faults and are difficult to distribute among multiple robots with limited computational resources. We use qualitative fault isolation instead which is very efficient but currently is limited to abrupt faults. To deal with parametric uncertainty, we incorporate model uncertainty as a parameter in our fault detection scheme and apply a statistical test on the residuals to robustly detect faults. We use a single distributed Kalman filter as opposed to using a bank of Kalman filters, which requires a Kalman filter for each fault and is not efficient for distributed systems with a large number of faults. None of the previous approaches explicitly use any temporal measurement deviation information to resolve ambiguities in the diagnosis results. Relative measurement orderings distinguish among faults based on event orderings, where the events are measurement deviations. The technique, therefore, has some similarities to discrete-event diagnosis approaches in [11], [33], and decentralized approaches in [34]. To our knowledge, this is the first time a distributed diagnosis approach is developed and demonstrated for process faults in formations of mobile robots.

Multi-agent and distributed diagnosis have been explored previously as well. In [35], distributed systems are diagnosed using an agent framework where some failures are diagnosed locally, and others require coordination between the agents. In [36], local diagnosers construct local diagnoses such that they are consistent with global diagnoses, sacrificing diagnostic precision for gains in computational complexity. In contrast to these approaches, we formulate this as a design problem, creating local diagnosers which are guaranteed to have enough information such that no coordination needs to occur, thus local diagnosis results correspond to global diagnosis results. The problem of addressing coordination failures in multi-robot teams is addressed in [37], [38]. Our approach deals with process faults, whereas coordination failures are better described as logical faults, which are at a higher level. Such approaches, however, can be considered as complementary to our work.

The paper is organized as follows. Section II presents our modeling methodology. Section III describes the multi-robot

diagnosis problem, and presents the computational architecture of the diagnosis scheme. Section IV presents our solution to the distributed fault detection problem. Section V discusses the fault isolation approach. Section VI demonstrates the effectiveness of the approach using experimental results for a four-robot formation. Section VII concludes the paper.

## II. Modeling

In this paper, we develop diagnostic solutions for formations of robots that employ leader-based control schemes of the type described in [1], [2]. The approach assumes that there exists a single global leader, for the purposes of control, who follows a known, planned trajectory. The remaining robots in the formation, i.e., the followers, must maintain their positions with respect to the global leader and/or other followers (local leaders). Correct behavior is defined with respect to a global objective. The global objective of the system is to maintain the overall formation while pursuing the planned trajectory and executing predefined tasks, e.g., collecting information or pushing a load.

Each follower robot implements two control laws, governing its translational and rotational velocities. These laws are functions of a robot's local information and information generated by its leaders. Therefore, the approach is scalable to a large number of robots without a corresponding increase in algorithm complexity. The formation model of [1], [2] is also general enough to model any type of rigid formation, since each robot's position is defined with respect to other robots in the formation. We adopt this formation modeling and control approach in our work, and apply our diagnosis framework to these robot configurations.

The control algorithms in [1], [2] assume a kinematic model of the robot, given the translational and rotational velocities as inputs. We develop a bond graph model of the system that captures both the kinematic and dynamic behavior of the robot under nominal and faulty system operation. The formation control mechanisms are explicitly built into the bond graph model for diagnosis, and, therefore, our approach can be used with any control scheme that ensures the robot team is in a rigid formation.

Each robot includes a local controller that regulates the velocities of its two wheels. The sensor suite includes motor encoders to measure wheel velocity and a gyroscope to measure heading. A distributed controller coordinates the formation by determining the desired velocities for each robot based on local and remote sensor measurements, communicated via a wireless network. In the remainder of the section, we present the model of the multi-robot system used for diagnosis.

### A. Modeling of a Single Mobile Robot

Each robot is modeled using a bond graph. Bond graphs define an energy-based, lumped-parameter, topological modeling scheme for models of dynamic systems [6]. They are particularly suitable for diagnosis because they incorporate causal and temporal information required for deriving and analyzing fault transients. Furthermore, components can be parameterized as bond graph element parameters, which makes

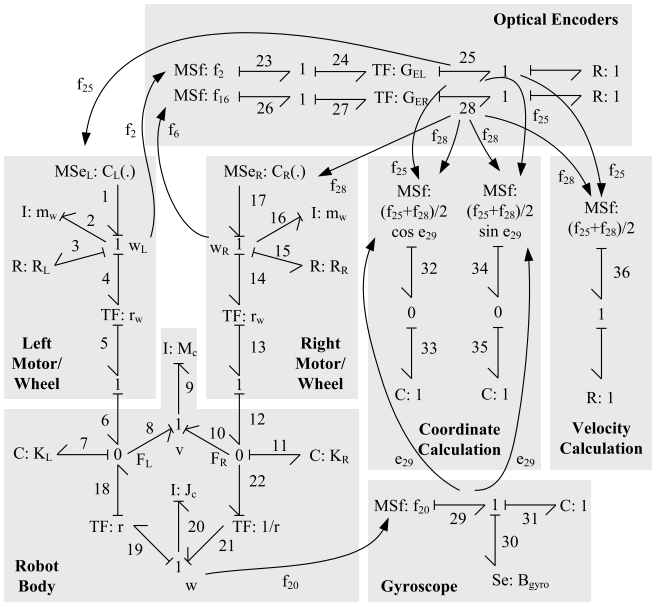


Fig. 1. Bond graph model of a single robot.

it easier to link observed fault transients to parameter value changes in the system components [8], [9].

A single robot consists of left and right wheel drive subsystems, a chassis, a gyroscope, and two motor encoders. The bond graph for a single robot is shown in Fig. 1. Bonds (energy transfer pathways) are represented as half arrows. Associated with each bond are two variables: *effort* and *flow*, denoted by $e_i$ and $f_i$, respectively, where $i$ is the bond number, and the product $e_i \times f_i$ defines the rate of energy transfer through the bond. Signals (information transfer pathways) are represented as arrows, and each link is associated with a single variable, as shown in Fig. 1. 1-junctions represent the common velocity points, e.g., the rotational velocity of the left wheel, $\omega_L$, the rotational velocity of the right wheel, $\omega_R$, the forward velocity of the robot, $v$, and the rotational velocity of the robot, $\omega$. 0-junctions represent common force points, e.g., the forces on the left and right sides of the robot, $F_L$ and $F_R$. The wheel subsystems include modulated sources of effort ($MSe_L$ and $MSe_R$) that model the actuator torque outputs that directly feed the wheels, and inertial (I) elements that model wheel mass and inertia, $m_w$. Resistive (R) elements (with parameters $R_L$ and $R_R$) model energy dissipation (i.e., friction) in the wheels. Transformers (TF) model the transformations between linear and rotational velocities. The robot body subsystem includes inertia components that model robot mass, $M_c$, and rotational inertia, $J_c$. The capacitive (C) components (with parameters $K_L$ and $K_R$) model the mechanical stiffness of the robot system.

Sensor models in the bond graph are derived from the kinematic relationships between the robot velocities and the measurements. Each robot includes a gyroscope and two motor encoders. The gyroscope computes the heading, $\theta(t)$, using a kinematic equation based on the rotational velocity, $\omega(t)$, of the robot body, i.e., $\dot{\theta}(t) = \omega$. The equations for the optical encoder measurements involve a gain transforming the wheels'

rotational velocities to their linear velocities, i.e., $v_L(t) = G_{EL}\ \omega_L(t)$ and $v_R(t) = G_{ER}\ \omega_R(t)$, where $G_{EL}$ and $G_{ER}$ are the encoder gains for the left and right wheels, respectively. These are used to calculate the measured translational velocity, $v$, of the robot. In the bond graph of Fig. 1, $v$ is represented by the flow variable $f_{36}$, associated with bond 36.

The sensors are modeled in the bond graph as modulated sources of flow that encapsulate the measurement equations for $v_L$, $v_R$, and $\theta$. For the gyroscope, the flow source is the rotational velocity of the robot, $\omega$, represented in the model as $f_{20}$. The measured variable, the heading, is $e_{29}$ (the effort variable associated with bond 29), which is the integral of $\omega$ plus the sensor bias (if any). For the case of the optical encoders, the flow is the rotational velocity of a wheel ($\omega_L$ and $\omega_R$) passed through a gain, so the measured variables are $f_{25}$ and $f_{28}$.

Position information is calculated in the bond graph using velocity and heading information. These are also modeled using modulated sources of flow. The $x$ and $y$ coordinates are described by:

$$\begin{aligned}\dot{x}(t) &= \frac{v_L(t) + v_R(t)}{2}\cos\theta(t),\\ \dot{y}(t) &= \frac{v_L(t) + v_R(t)}{2}\sin\theta(t).\end{aligned}$$

The modulated sources of flow provide these quantities, which are integrated to obtain the coordinate positions of the robot, $e_{33}$ for the $x$ coordinate, and $e_{35}$ for the $y$ coordinate.

Local controllers are also modeled in the bond graph. The input to the robots are the motor torques modeled as modulated sources of effort, which encapsulate the wheel control equations. In our model, each wheel has an accompanying PID controller. For example, the equation of the controller for the left wheel is given by:

$$\tau = K_p(v_L - v_{Ld}) + K_i\int_0^t (v_L - v_{Ld})dt + K_d\frac{d(v_L - v_{Ld})}{dt},$$

where $\tau$ is the torque applied by the motor, $K_p$, $K_i$, and $K_d$ are the controller gains, and $v_{Ld}$ is the reference velocity provided by the formation controller described in the next subsection. The torque for the left (right) wheel is represented in the bond graph by the modulated source of effort $MSe_L$ ($MSe_R$). The PID controller is represented in the bond graph by the function $C_L(\cdot)$ ($C_R(\cdot)$) that modulates the torque. The edges from the observed velocities to the wheel sources represent the control links for the PID controllers. Other controller types can be modeled similarly.

### *B. Modeling Formations of Mobile Robots*

Following the approach in [1], we model a formation as a tuple $\mathcal{F} = (\mathcal{S}, \mathcal{C})$, where $\mathcal{S}$ is a set of shape variables defining the formation structure, and $\mathcal{C}$ is a control graph showing the control strategies for each robot and dependencies on their neighbors. The shape variables $\mathcal{S}$ consist of relative bearings and separations between robots. Control laws maintain either the relative heading and separation of a follower to its leader (separation-bearing control, or SBC), or the separations of a follower from two leaders (separation-separation control, or SSC). In this way, formations can be constructed by defining for each robot its control strategy, shape variables, and leaders.

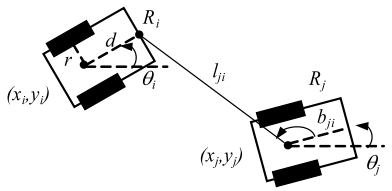


Fig. 2. Control variables in robot formations.

*Definition 1 (Control Graph):* A control graph $\mathcal{C}$ is a directed, acyclic graph, where each robot, $R_i$, defines a vertex. A directed edge $(R_i, R_j)$ implies that $R_i$ is a local leader to $R_j$, i.e., $R_j$ maintains its position with respect to $R_i$.

As in [1], we restrict a control graph with the following constraints: (*i*) The formation leader, $R_1$, has no incoming edges and at least one outgoing edge, and (*ii*) all other robots have at least one and no more than two incoming edges. If a robot has exactly one incoming edge, then it employs the SBC strategy, otherwise it has two incoming edges and it employs the SSC strategy. In general, a robot with three or more incoming edges is over-constrained for planar formations, so this is disallowed.

We denote by $l_{ji}$ the separation between $R_j$ and $R_i$ and by $b_{ji}$ the relative bearing from $R_j$ to $R_i$ as measured from $R_i$'s axis of symmetry to the line connecting the center of $R_j$'s wheel axis with the point $d$ units from the center of $R_i$'s wheel axis, as shown in Fig. 2. A subscript $d$ denotes a desired value, and a subscript $i$ indicates a variable associated with $R_i$.

With local leader $R_j$, the SBC control equations [2] for the follower, $R_i$, describe the desired rotational velocity $\omega_{di}$ and linear velocity $v_{di}$, and are given as follows.

$$\begin{aligned}\omega_{di} &= \frac{\cos\gamma_j}{d}\{\alpha_b l_{ji}(b_{dji} - b_{ji}) - v_j\sin b_{ji} + l_{ji}\omega_j + \\ &\qquad \rho_{ji}\sin\gamma_j\}\\ v_{di} &= \rho_{ji} - d\omega_{di}\tan\gamma_j,\end{aligned}$$

where

$$\begin{aligned}\rho_{ji} &= \frac{\alpha_l(l_{dji} - l_{ji}) + v_j\cos b_{ji}}{\cos\gamma_j},\\ \gamma_j &= \theta_j + b_{ji} - \theta_i,\end{aligned}$$

and $\alpha_l$ and $\alpha_b$ are control gains for the separation and bearing, respectively.

With local leaders $R_j$ and $R_k$, the SSC control equations [2] for the follower, $R_i$, are given as follows.

$$\begin{aligned}\omega_{di} &= \frac{1}{d\sin(\gamma_j - \gamma_k)}\{\alpha_{lj}(l_{dji} - l_{ji})\cos\gamma_k + \\ &\qquad v_j\cos b_{ji}\cos\gamma_k - \\ &\qquad \alpha_{lk}(l_{dki} - l_{ki})\cos\gamma_j - \\ &\qquad v_k\cos b_{ki}\cos\gamma_j\}\\ v_{di} &= \frac{\alpha_{lj}(l_{dji} - l_{ji}) + v_j\cos b_{ji} - d\omega_{di}\sin\gamma_j}{\cos\gamma_j},\end{aligned}$$

where $\alpha_{lj}$ and $\alpha_{lk}$ are control gains for separations from $R_j$ and $R_k$, respectively.

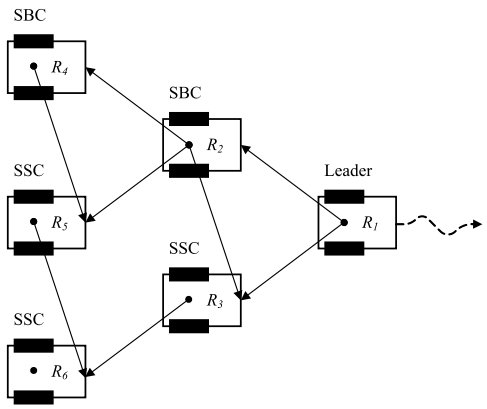


Fig. 3. Example formation of six robots.

The formation control outputs $v_{di}$ and $\omega_{di}$ can be decoupled into individual left and right wheel velocities to serve as inputs to the wheels' PID controllers as $v_{Ldi} = v_{di} - r\omega_{di}$ and $v_{Rdi} = v_{di} + r\omega_{di}$, where $r$ is half the wheel base (see Fig. 2).

Fig. 3 shows an example formation structure. $R_1$ is the leader, and moves in a pre-defined trajectory. $R_2$ employs SBC, so it only maintains its separation and bearing with respect to $R_1$. $R_3$ uses SSC, maintaining its separation from $R_1$ and $R_2$. $R_5$ and $R_6$ also utilize SSC, and $R_4$ uses SBC.

The distributed control algorithm is modeled in the bond graph in the same manner as the local PID control. Because heading, position, and velocity measurements are required for the control, signals are introduced from each robot's $x$, $y$, $\theta$, and $v$ measurements to its own wheel sources, and also to the wheel sources of each follower to represent the communication between the robots. For example, following the formation in Fig. 3, signal edges are constructed from $v_2$ ($f_{36,2}$) of $R_2$ to $MSe_{L,3}$ ($e_{1,3}$) and $MSe_{R,3}$ ($e_{17,3}$) of $R_3$ because $C_{L,3}(\cdot)$ and $C_{R,3}(\cdot)$ take $v_2$ as an argument. The multi-robot bond graph is derived from the composition of single robot bond graphs with these signal edges included.

### C. Modeling Faults

Faults are represented as abrupt parameter value changes in the bond graph model. Table I shows the possible actuator and sensor faults that can occur in the robot, and the corresponding parameters in the bond graph model (a superscript of + or − indicates the direction of change of the parameter value). Actuator (motor) faults are modeled as changes in the effort sources. A saturation fault in an actuator limits the maximum wheel velocity. Sensor bias is modeled as an additive fault, and is represented by a change in the effort source at the measured value (nominally the effort is 0). For example, a bias in the gyroscope manifests as an abrupt, constant value added to the true measurement value. Sensor failures are modeled as multiplicative faults and are parameterized by a change in the sensors' transformer gains. For the optical encoders, the nominal value of $G_{EL}$ (or $G_{ER}$) is $r_w$, the wheel radius. A fault in the encoder is modeled as a reduction in gain, i.e., its value reduces to a number in the interval $[0, r_w)$. This corresponds to a percentage of the encoder counts that are missed (at least 10%).

TABLE I
FAULT PARAMETERS IN THE BOND GRAPH MODEL

| Fault | Parameter |
|---|---|
| Left actuator saturation/failure | $MSe_L^-$ |
| Right actuator saturation/failure | $MSe_R^-$ |
| Left encoder (partial) failure | $G_{EL}^-$ |
| Right encoder (partial) failure | $G_{ER}^-$ |
| Gyroscope bias | $B_{gyro}^+, B_{gyro}^-$ |

Our diagnosis approach makes the following assumptions: (*i*) faults are persistent, (*ii*) only single faults occur, and (*iii*) the fault profile is an abrupt change. Although restrictive, many practical faults can be handled under this assumption. For example, if a robot gets stuck or is occluded by an obstacle, then this will manifest qualitatively as an actuator fault, i.e., the robot will slow down abruptly.

## III. DIAGNOSIS APPROACH

Our distributed diagnosis algorithm considers a finite set of abrupt, persistent faults, and makes the single fault assumption. We denote the complete set of system faults as $F$, and the complete set of measurements as $M$. For a system of $n$ robots, associated with each robot $R_i$ is a set of local faults $F_i$ and a set of local measurements $M_i$, such that

$$F = \bigcup_{1 \le i \le n} F_i \text{ and } M = \bigcup_{1 \le i \le n} M_i.$$

In distributed diagnosis, our objective is to design $n$ diagnosers $D_i$, one for each robot $R_i$, so that $D_i$ can diagnose all faults in $F_i$ using $M_i^+$, where $M_i^+ \triangleq M_i \cup M_{ci}$, and $M_{ci}$ are additional measurements from other robots. The design goal is to find the minimum set $M_{ci} \subseteq M$ such that each fault $f \in F_i$ can be uniquely isolated within the fault set $F$ using $M_i^+$. If $M_{ci} = \varnothing$, then $F_i$ is said to be *strongly independent* from the other fault sets [17], i.e., faults in $F_i$ can be globally isolated using only measurements in $M_i$, and we say that the diagnoser $D_i$ is independent of the other diagnosers. Otherwise, $F_i$ is said to be *weakly independent*, i.e., to obtain globally correct diagnoses for the faults in $F_i$, $D_i$ must use additional measurements. We will show in Section V-E that some robots' fault sets will be strongly independent, and others will be weakly independent.

The measurement set $M_i^+$ allows $D_i$ to distinguish uniquely every fault $f \in F_i$ from the fault set $F_i$ (local faults), and from $F - F_i$ (remote faults). Our design ensures that the effects observed on $M_i^+$ can only be explained by a single fault $f_{local} \in F_i$ or some (unknown) fault $f_{remote} \in F - F_i$. Each robot $R_i$ knows only the effects of faults in $F_i$ on measurements in $M_i^+$. Therefore, if there is no $f \in F_i$ which matches the observations, the fault is guaranteed to be remote. Under the single fault assumption, agreement between individual diagnosers is reached implicitly. If $f \in F$ occurs, it will belong to exactly one $F_i$ thus exactly one robot, $R_i$, will achieve the diagnosis $\{f\}$ and all other robots will eventually achieve the (empty) diagnosis $\varnothing$. Therefore, the global diagnosis is simply $\{f\}$. Practically, we do not have to wait for all other robots to complete their diagnostic tasks.

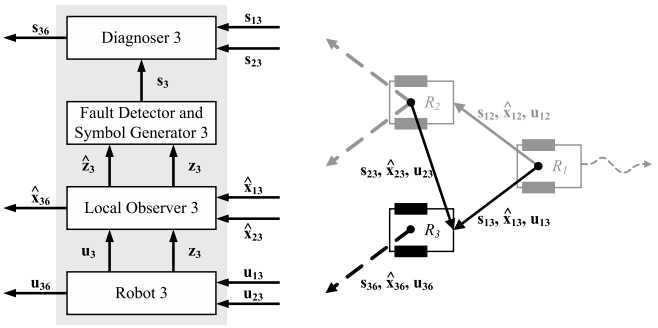


Fig. 4. Diagnosis architecture for $R_3$

A robot $R_i$ may conclude that its diagnosis $\{f\}$ is the global diagnosis if one of three conditions holds: (*i*) all measurements in $M_i^+$ have deviated, so by design no other fault could have occurred, (*ii*) all other robots have reached the diagnosis $\varnothing$, thus leaving only $\{f\}$, or (*iii*) measurement deviation information allows the robot to conclude a remote fault could not have occurred.

The diagnosis architecture for the multi-robot system consists of four core components. Fig. 4 illustrates the architecture for $R_3$ in the six-robot formation example. A follower robot $R_i$ (e.g., $R_3$) receives communicated inputs $\mathbf{u_{ji}}$ ($u_{13}$ and $u_{23}$) from each local leader $R_j$ ($R_1$ and $R_2$). The local observers, implemented as Kalman filters, are based on a state space model of each robot derived from the bond graph. An observer computes the output estimates, $\hat{\mathbf{z}}_\mathbf{i}$, ($\hat{\mathbf{z}}_\mathbf{3}$) given the input $\mathbf{u_i}$ ($\mathbf{u_3}$), the local measurements $\mathbf{z_i}$ ($\mathbf{z_3}$), and communicated state information, $\hat{\mathbf{x}}_\mathbf{ji}$ ($\hat{\mathbf{x}}_\mathbf{13}$ and $\hat{\mathbf{x}}_\mathbf{23}$), from each leader $R_j$ ($R_1$ and $R_2$) as necessary. It also outputs relevant state information, $\hat{\mathbf{x}}_\mathbf{ik}$ ($\hat{\mathbf{x}}_\mathbf{36}$) to each follower $R_k$ ($R_6$). The fault detectors compute the residuals of the measurements as differences between actual and predicted values. If a fault is detected, the symbol generator computes qualitative values, $\mathbf{s_i}$ ($\mathbf{s_3}$), i.e., fault signatures, for the changes in measurement values. Each local diagnoser uses these signatures and communicated signatures $\mathbf{s_{ji}}$ ($\mathbf{s_{13}}$ and $\mathbf{s_{23}}$) from each required robot $R_j$ ($R_1$ and $R_2$) to isolate the fault. The required communicated signatures are determined by the distributed diagnoser design discussed in Section V-E. The local diagnoser also outputs some of its own signatures, $\mathbf{s_{ik}}$ ($\mathbf{s_{36}}$), required by the diagnoser design for other diagnosers $D_k$ ($D_6$) to use.

## IV. Distributed Fault Detection

Fault detection operates on the residuals, defined as the difference between model-predicted and actual measurement outputs. For an ideal system with noiseless measurements and a perfect model, a nonzero residual vector indicates a fault occurrence. Noise and model imperfections make the residual generation and fault detection tasks more complex. We address this issue using a Kalman filter to track the system trajectory, and defining the fault detection task as a statistical test of significance. In both the Kalman filter and the fault detection test, all noise is assumed to be Gaussian with zero-mean.

The fault detection strategy is extended for multi-robot systems by using a distributed, decentralized, extended Kalman filter (DDEKF) scheme [39]. This method creates local filters for each robot, which share relevant observations and estimates. Each DDEKF produces estimates of the local state vector using local observations, local estimates, and required shared observations and estimates. For the formation system, each follower must know the estimates of $x, y, \theta$, and $v$ for each local leader. Each robot observes its own wheel velocities and heading, i.e., the local measurements are $\mathbf{z}_i$ = $[v_{Li}\ v_{Ri}\ \theta_i]^T$ for $R_i$. State space equations required by the Kalman filters are directly derived from the bond graph model [6]. Unknown bond graph parameters were estimated using system identification techniques. We use a discrete-time, reduced order form of the derived state space model, assuming the dynamics of the wheels are decoupled. For the reduced model, the local state vector for $R_i$ is $\mathbf{x}_i$ = $[x_i\ y_i\ \theta_i\ v_{Li}\ a_{Li}\ b_{Li}\ v_{Ri}\ a_{Ri}\ b_{Ri}\ x_j\ y_j\ \theta_j\ v_j\ x_k\ y_k\ \theta_k\ v_k]^T$, where $j$ and $k$ are the local leader robots. The variables $a_{Li}$, $b_{Li}$ and $a_{Ri}$, $b_{Ri}$ correspond to dynamic states of the left and right wheels, respectively, and are based on a 3-dimensional model of each wheel derived using system identification.

The DDEKF estimate update equations are given by:

$$\begin{aligned}
\hat{\mathbf{z}}_i(k) &= \mathbf{C}_i(k)\hat{\mathbf{x}}_i(k|k) \\
\hat{\mathbf{x}}_i(k|k) &= \mathbf{P}_i(k|k)\{\mathbf{P}_i^+(k|k-1)\hat{\mathbf{x}}_i(k|k-1) + \\
&\quad \sum_{j=1}^{n} \mathbf{P}_i^+(k|\mathbf{z}_j(k))\hat{\mathbf{x}}_i(k|\mathbf{z}_j(k))]\} \\
\mathbf{P}_i(k|k) &= [\mathbf{P}_i^+(k|k-1) + \sum_{j=1}^{n} \mathbf{P}_i^+(k|\mathbf{z}_j(k))]^+,
\end{aligned}$$

where $\mathbf{C}_i$ is the local output matrix, $\mathbf{P}_i$ the local covariance matrix, $n$ the number of subsystems, and $^+$ indicates the generalized matrix inverse.

The difference between the observed outputs, $\mathbf{z}_i(k)$, and the estimated outputs, $\hat{\mathbf{z}}_i(k)$, define the residual vector. Each robot computes its own measurement residuals. The fault detection scheme is based on a Z-test [40] that uses the estimated variance of the residuals and a pre-specified confidence level. A small sliding window (e.g., 5 samples) is used to estimate the current mean of the residuals, and this is preceded by a much larger sliding window (e.g., 100 samples) to estimate the variance [41]. When the current mean of one of the residual signals shows a statistically significant deviation from zero (accounting for modeling error), a fault is detected. By adjusting window sizes and the modeling error parameter, the detectors are tuned to keep the false alarm (false positive) rate below pre-specified thresholds for the fault magnitudes under consideration and the deployment environment. Since faults are persistent, missed detections (false negatives) will not occur unless the fault magnitude is very small. This tuning sets the sensitivity of the fault detector. At high sensitivity, detection is fast and the chance of false negatives is low, but the chance of false positives is high. At low sensitivity, detection is slow and the chance of false positives is low, but the chance of false negatives is high. Other fault detection strategies are discussed in [42]–[44] and the references therein, and could be applied instead.

The change in the residual is analyzed to determine if

an abrupt change (discontinuity) has occurred in the measurement. A discontinuity produces a smooth change in the opposite direction of the initial abrupt change [8]. A nondiscontinuous change, however, does not produce an immediate direction change. Again using the Z-test, the slope of the residual is determined over a small window (e.g., 6 samples) after the point of fault detection. If the slope is determined within the window and is opposite in direction from the initial change, the observed change is classified as a discontinuity. Otherwise, we assume no discontinuity has occurred.

## V. Fault Isolation

### A. Background

The TRANSCEND scheme [8], [9] is employed for diagnosis of abrupt faults in the multi-robot system. Fault isolation in TRANSCEND is based on a qualitative analysis of the transient dynamics caused by abrupt faults. Deviations in measurement values after a fault occurrence can be represented as a fault signature, where predicted deviations in magnitude and higher order derivative values are mapped to symbols of the set $\{+, 0, -\}$. Magnitude changes correspond to a deviation above normal, no deviation, and a deviation below normal, respectively, and derivative values imply increasing, steady, and decreasing values for the signals, respectively.

Fault isolation in TRANSCEND utilizes a Temporal Causal Graph (TCG) representation, which can be derived directly from the bond graph model of the system (see [8] for details). The TCG captures the causal and temporal relations between system variables based on the bond graph element constituent equations. It specifies the signal flow graph of the system in a form where edges are labeled with single component parameter values (e.g. $R_L$, $G_{EL}$), or direct $(+1)$ and inverse $(-1)$ proportionality relations between the source and destination vertices. Temporal relations (e.g. $dt/m_W$, $dt/M_c$) on the edges indicate that the source vertex affects the derivative of its destination vertex. Fig. 5 depicts the TCG model for a single robot, with state variables circled and measured variables boxed. The remaining variables are system variables algebraically related to the state variables.

The TCG of the entire system is derived systematically from the global system bond graph model. It consists of a TCG for each robot, with additional edges between the robot TCG models that convey the measurements required by the formation control. These additional edges start at the local or remote measurement vertex and end at the effort source vertices representing actuator torque. For example, an edge is required from $f_{36,2}$ of $R_2$'s TCG ($v_2$) to $e_{1,3}$ of $R_3$'s TCG ($\tau_{L3}$) because $R_3$'s control requires $v_2$ as an input. This represents the fact that the torque $\tau_{L3}$ is causally influenced by $v_2$. The labels on these edges include a $dt$ specifier to indicate a time delay due to the system dynamics. The sign of the label depends on whether a change in the measurement will cause a direct or inverse change on the control output. Because the control is nonlinear, the direction of change will depend on the robot's position. The effects of $x$, $y$, and $\theta$ depend on position, but the control output change due to a fault transient will always initially change in the same direction as the communicated velocity measurements. These edges capture the qualitative effects of the measurements in the transient dynamics of the robot's motion. Therefore, the global system TCG not only captures fault propagation within a single robot but also from one robot to another through the leader-follower interactions.

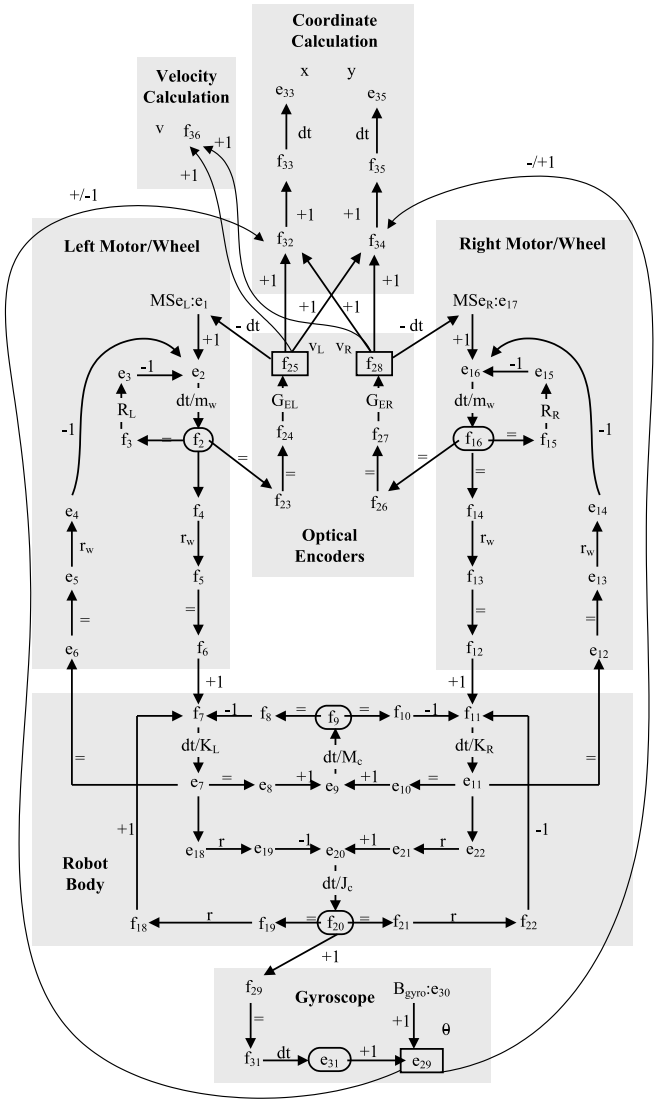


Fig. 5. TCG for a single robot of the multi-robot system.

The qualitative fault isolation scheme in TRANSCEND involves three steps: (*i*) generating initial fault hypotheses given the initial set of deviated measurements, (*ii*) generating fault signatures for all the hypotheses, and (*iii*) tracking the fault transients using the fault signatures and a progressive monitoring scheme for all initial fault hypotheses. Fault signatures [8] are generated by running a forward propagation algorithm on the TCG to predict qualitative effects of faults on measurements. The qualitative effect of a fault, + or −, is propagated to all measurement vertices in the TCG to determine fault signatures for each measurement. It can be shown that these provide a temporal progression of the predicted qualitative changes in the measured signal. By expressing the fault signature as derivative effects, measurement analysis can be

formulated as a progressive monitoring scheme, where lower order changes manifest before higher order changes. This is justified by a Taylor series expansion of a measured signal deviation [9].

In the robot TCG, for example, the fault $MSe_L^-$ starts at the vertex $e_1$. The − change propagates to the heading measurement (vertex $e_{29}$) by passing through four temporal edges ($e_2 \xrightarrow{dt/m_w} f_2$, $f_7 \xrightarrow{dt/K_L} e_7$, $e_{20} \xrightarrow{dt/J_c} f_{20}$, and $f_{31} \xrightarrow{dt} e_{31}$) with the sign getting reversed once, thus the first change is a $4^{th}$ order change of +. This change will eventually manifest as a change in magnitude and slope, which can be reliably measured.

Symbol generation on measurement residuals produces two qualitative symbols: a magnitude symbol (indicating a discontinuity), and a slope symbol (indicating the direction of change). Fault isolation in TRANSCEND compares residual magnitude and slopes to predicted fault signatures. Fault hypotheses whose signatures are consistent with the measured residual symbols are retained, and others are dropped.

### B. Fault Propagation Graph

The effects of a fault in a single robot may propagate to other robots in the system through the control links. This is modeled in the global TCG. For example, in Fig. 3, if an actuator fault occurs in $R_2$, it cannot maintain its separation and bearing with respect to $R_1$. Because $R_1$ will continue on its predefined trajectory, $R_3$ will lose its ability to maintain its pre-specified separation to both $R_1$ and $R_2$. Therefore, the fault in $R_2$ has now propagated to $R_3$. Since $R_3$ can no longer act as a leader to its followers, the formation will not be maintained, and the fault will further propagate to $R_6$. On the other hand, $R_4$ can still maintain its separation and bearing with respect to $R_2$, and thus $R_5$ can also maintain its separations to $R_2$ and $R_4$. Depending on the situation and control strategy employed, faults will, therefore, propagate to some parts of the system and not others. To make the overall diagnosis process more efficient, we can remove causal links between the robots where faults do not propagate, and still generate correct diagnosis results. The reduced interactions between the robots are captured in the form of a *fault propagation graph* that is derived from the control graph.

*Definition 2 (Fault Propagation Graph):* A fault propagation graph $\mathcal{G}$ is a directed, acyclic graph, where each robot, $R_i$, defines a vertex of the graph. A directed edge $(R_i, R_j)$ implies that faults may propagate from $R_i$ to $R_j$. We denote the parents of a robot $R_i$ in $\mathcal{G}$ as $Par(R_i)$ and the ancestors as $Anc(R_i)$.

Fig. 6 shows the fault propagation graph for the six-robot formation. To improve the efficiency of diagnosis, the fault propagation graph is constructed as a subset of the formation control graph. An edge $(R_i, R_j)$ in a formation control graph $\mathcal{C}$ is not included in the corresponding graph $\mathcal{G}$, if $R_j$ has only one incoming edge. Faults do not propagate to robots with single incoming edges, i.e., robots that have a single leader, because that robot can maintain its position relative to its leader for any arbitrary trajectory if it is not faulty itself. In the example formation illustrated in Fig. 3, edges $(R_1, R_2)$ and

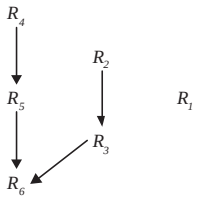


Fig. 6. Fault propagation graph for the formation of six robots.

$(R_2, R_4)$ are removed because $R_2$ and $R_4$ have single leaders. An edge $(R_i, R_j)$ from the control graph $\mathcal{C}$ is also removed if $R_j$ has another edge $(R_k, R_j)$ and $R_k$ has a single incoming edge $(R_i, R_k)$. Faults also do not propagate along these edges, because a fault in $R_i$ would not propagate to $R_k$, and since $R_j$ depends on both $R_i$ and $R_k$, it will not exhibit faulty behavior if $R_i$ does. In the example, edges $(R_1, R_3)$ and $(R_2, R_5)$ are removed. A simple algorithm can be constructed to find and remove all such edges. Note that only robots employing SBC can become source vertices, i.e., have no incoming edges, through this procedure.

The fault propagation graph describes whether to treat control information as inputs (through which faults do not propagate) or remote measurements (through which faults do propagate). The fault detection model can, therefore, be simplified with respect to $\mathcal{G}$. An absent link from $R_i$ and $R_j$ in $\mathcal{G}$ indicates that $R_j$ does not require estimates from $R_i$ to produce its local estimates.

### C. Diagnosability Analysis

An important prerequisite for diagnoser design is to determine whether the system is diagnosable, i.e., all faults of interest can be uniquely isolated with the given measurement set. A fault $f_1$ will be distinguished from another fault $f_2$ if, during the isolation process, a measurement deviation occurs that matches the fault signature for $f_1$ but not $f_2$.

Table II shows the fault signatures for faults and measurements of two robots, $R_2$ and $R_3$, in the six-robot formation. The signatures are generated from the system TCG, with only the magnitude change (discontinuity) symbol and the first non-zero direction of change symbol shown. A $\star$ symbol indicates an indeterminate effect, i.e., there are at least two paths of the same order that propagate + and − effects, and, therefore, the sign of the change cannot be computed using qualitative propagation. Some of the effects of these faults are determined by the robot's position, since the controller inputs are functions of separations and bearings, which are functions of position. Such effects are denoted by a $0\pm$. A 00 indicates that a fault has no effect on the corresponding measurement because there is no path to it.

From the signatures, it is clear that not all faults can be distinguished. If some fault $f$ occurs in $R_2$, one of $R_2$'s measurements will deviate. Because, faults in $R_3$ do not propagate to $R_2$'s measurements, the fault $f$ cannot belong to $R_3$. This is indicated by the absence of a causal path from $R_3$ to $R_2$ and qualified by the 00 symbols in the lower left segment of Table II. However, if an actuator fault occurs in $R_3$, deviations could match those of one of $R_2$'s

TABLE II
FAULT SIGNATURES FOR $R_2$ AND $R_3$ IN THE SIX-ROBOT FORMATION

| Fault | $v_{L2}$ | $v_{R2}$ | $v_2$ | $\theta_2$ | $v_{L3}$ | $v_{R3}$ | $v_3$ | $\theta_3$ |
|---|---|---|---|---|---|---|---|---|
| $MSe^-_{L,2}$ | 0− | 0⋆ | 0− | 0+ | 0± | 0± | 0− | 0± |
| $MSe^-_{R,2}$ | 0⋆ | 0− | 0− | 0− | 0± | 0± | 0− | 0± |
| $G^-_{EL,2}$ | −+ | 0⋆ | −+ | 0− | 0± | 0± | 0− | 0± |
| $G^-_{ER,2}$ | 0⋆ | −+ | −+ | 0+ | 0± | 0± | 0− | 0± |
| $B^+_{gyro,2}$ | 0+ | 0− | 0⋆ | +− | 0± | 0± | 0± | 0± |
| $B^-_{gyro,2}$ | 0− | 0+ | 0⋆ | −+ | 0± | 0± | 0± | 0± |
| $MSe^-_{L,3}$ | 00 | 00 | 00 | 00 | 0− | 0⋆ | 0− | 0+ |
| $MSe^-_{R,3}$ | 00 | 00 | 00 | 00 | 0⋆ | 0− | 0− | 0− |
| $G^-_{EL,3}$ | 00 | 00 | 00 | 00 | −+ | 0⋆ | −+ | 0− |
| $G^-_{ER,3}$ | 00 | 00 | 00 | 00 | 0⋆ | −+ | −+ | 0+ |
| $B^+_{gyro,3}$ | 00 | 00 | 00 | 00 | 0+ | 0− | 0⋆ | +− |
| $B^-_{gyro,3}$ | 00 | 00 | 00 | 00 | 0− | 0+ | 0⋆ | −+ |

faults. Since $R_2$'s measurements will never deviate if one of these faults occurs, we would have to wait infinitely long before we can be certain the fault does not belong to $R_2$. For example, if $MSe^-_{L3}$ occurs, its effects on $R_3$'s measurements could manifest as 0− on $v_{L3}$, 0− on $v_{R3}$, 0− on $v_3$, and 0+ on $\theta_3$, which can be explained by any of the faults for $R_2$. In general, we cannot distinguish between actuator faults occurring on different robots in the formation. Therefore, using the given measurement set and the fault signature approach, the system is not globally diagnosable. This motivates the need for employing additional discriminatory information to achieve global diagnosability.

### D. Relative Measurement Orderings

The traditional TRANSCEND scheme uses fault signatures to distinguish between faults. The order in which the measurements deviate is not taken into account when refining fault hypotheses. Relative measurement orderings capture the intuition that fault effects will manifest in some parts of the system before others. For example, a fault occurring in one robot will likely manifest first in that robot and then in the remote robot, if there are energy storage elements in the path between the local and remote sensors in the bond graph model. If there are no energy storage elements, the relation is algebraic and no delay will be observed.

*Definition 3 (Relative Measurement Ordering):* Consider a fault $f$ and measurements $m_1$ and $m_2$. If the effects of the fault $f$ manifest in $m_1$ before $m_2$ then we can define a relative measurement ordering between $m_1$ and $m_2$ for fault $f$, denoted as $m_1 \prec_f m_2$.

Relative measurement orderings can be derived from the TCG based on the notion of a *fault path*.

*Definition 4 (Fault Path):* A fault path for a fault $f$ and measurement $m$ is a path in the TCG which begins at the fault $f$ and ends at the measurement $m$.

The set of all fault paths from $f$ to $m$ is denoted by $FP_{f,m}$. The order of a fault path is defined as the number of temporal edges in the path. A minimum order fault path is a path in $FP_{f,m}$ that contains the minimum number of temporal edges needed to reach $m$ from $f$. More than one fault path of a specific order may exist for $f$ and $m$, since there are often multiple paths from one vertex to another in the TCG.

*Definition 5 (Minimum Order Fault Path Set):* The minimum order fault path set for $f$ and $m$ is the set of all minimum order fault paths, and is denoted as $FP^*_{f,m}$.

A fault path represents the temporal propagation of a fault to a specific measurement variable in the system. For a certain fault, there may be multiple fault paths leading to a measurement. Since the lowest order effect of a fault will manifest first [9], only the minimum order fault path sets are useful in determining relative measurement orderings. For this purpose, we define a method for comparing fault paths.

*Definition 6 (Temporal Subpath):* For $p \in FP_{f,m_1}$ and $p' \in FP_{f,m_2}$, $p$ is a temporal subpath of $p'$ ($p \sqsubset p'$) if all temporal edges in $p$ exist in $p'$ in the same ordering, and the order of $p$ is less than the order of $p'$.

The following theorem shows how relative measurement orderings are derived from the TCG.

*Theorem 1:* If for every $p' \in FP^*_{f,m_2}$ there exists $p \in FP^*_{f,m_1}$ such that $p \sqsubset p'$, then we have $m_1 \prec_f m_2$.

*Proof:* In the signal flow graph for the TCG, let $r_1$ be the measurement vertex corresponding to $m_1$, $r_2$ the vertex for $m_2$, and $r_f$ the successor vertex of the edge with fault parameter $f$. The transfer functions from $r_f$ to $r_1$, $R_1(s)$ and from $r_f$ to $r_2$, $R_2(s)$, can be derived. Assume for every $p' \in FP^*_{f,m_2}$ there exists $p \in FP^*_{f,m_1}$ such that $p \sqsubset p'$. Then each minimum order path from $r_f$ to $r_2$ must go through $r_1$ or a vertex which can be expressed as $r_1 \cdot G$, where $G$ is some constant gain. $R_2(s)$ is a sum of terms which each correspond to different forward paths from $r_f$ to $r_2$. Because lower order effects manifest first, terms that correspond to forward paths of non-minimum order can be removed to produce $R'_2(s)$. Similarly, $R'_1(s)$ can be produced. Because every minimum order path from $r_f$ to $r_2$ goes through a vertex $r_1 \cdot G$, $R'_1(s)$ must appear as a factor in each term of $R'_2(s)$, therefore $R'_2(s) = H(s)R'_1(s)$, where $H(s)$ is a proper transfer function. The order of $m_1$ is less than the order of $m_2$ by the definition of the $\sqsubset$ relationship, so the number of poles for $R'_1(s)$ must be less than the number for $R'_2(s)$. $H(s)$ must introduce more poles than zeros to $R'_2(s)$, and, therefore, $H(s)$ is strictly proper. From $H(s)$, we can discretize using the given sampling rate of the system to get $H(z)$. Since $H(s)$ is strictly proper, $H(z)$ is, therefore $r'_2(k) = f(r'_1(k-1))$. Since $r'_2(k)$ depends only on past values of $r'_1(k)$, with appropriately selected detection thresholds[1], a deviation resulting from fault $f$ will appear first in $m_1$ and then in $m_2$, thus $m_1 \prec_f m_2$. ■

Therefore, for a given fault $f$, we can say that it manifests in measurement $m_1$ before measurement $m_2$ if for all minimum order fault paths of $m_2$, there is a minimum order fault path for $m_1$ the fault will traverse before completely traversing the given fault path of $m_2$. The transient due to the fault is slower for $m_2$ than it is for $m_1$, thus, the fault will cause a deviation first in $m_1$ and then in $m_2$. If this ordering is violated, we can eliminate that particular fault hypothesis $f$.

[1]This guarantees that for some time $|r_1(k)|$ will be greater than $|r_2(k)|$, after that time $|r_2(k)|$ may overtake $|r_1(k)|$ depending on the gain of $H(z)$. Therefore thresholds must be small enough such that deviations will cross them before that time.

For example, consider an actuator fault of the left wheel of $R_1$, $MSe^-_{L,1}$. The minimum order fault path set for the velocity measurement of $R_1$, $v_{L1}$, consists of the path $\{e_1 \xrightarrow{+1} e_2 \xrightarrow{dt/m_w} f_2 \xrightarrow{=} f_{23} \xrightarrow{=} f_{24} \xrightarrow{G_{EL}} f_{25}\}$, which contains only one temporal edge with label $dt/m_w$, implying an integration effect. Minimum order fault path sets for all other measurements must pass through that same edge, thus the temporal subpath relation holds. Therefore, we can define the ordering $v_{L1} \prec_{MSe^-_{L,1}} m$ for all other measurements $m$.

Generation of the minimum order fault path sets can be performed through a simple graph search on the TCG. Starting at the fault parameter in the graph, a forward search is performed to find all minimum order fault paths to each measurement. Using these minimum order fault paths for a specific fault, the temporal subpath relation can be checked between minimum order fault paths to determine the measurement orderings as described in Theorem 1.

Informally, two faults can be distinguished using orderings if there exists two measurements which deviate in some order for one fault, and in the opposite order for the other fault. Discrimination between faults using relative measurement orderings is based on the notion of temporal conflicts in the ordering relationships.

*Definition 7 (Ordering Set):* An ordering set for a fault $f$, $\Omega_f$, is the set of all relative measurement orderings for fault $f$.

*Definition 8 (Temporal Conflict):* A temporal conflict between ordering sets $\Omega_{f_1}$ and $\Omega_{f_2}$ for measurement set $M$ exists if there are two measurements $m_i, m_j \in M$ such that $(m_i \prec_{f_1} m_j) \in \Omega_{f_1}$ and $(m_j \prec_{f_2} m_i) \in \Omega_{f_2}$.

For a given measurement set and for each fault, we can derive a fault signature set and also an ordering set from the TCG. Given the fault signatures, the ordering sets can be used as additional distinguishing information for fault isolation. Therefore, the discriminatory power of a set of measurements is enhanced by using both fault signatures and relative measurement orderings. For a given set of measurements, two faults can be discriminated if they have different fault signatures or if they have conflicts in their ordering sets. Further, these two notions provide independent information and can be combined to provide more discriminatory information to distinguish among fault hypotheses.

Using this information, actuator faults can now be globally distinguished. From the global TCG model, it follows that an actuator fault will appear first in the velocity measurement of that wheel and then in other measurements. Taking the example from Table II, if $MSe^-_{L,3}$ occurs, it will manifest first in $v_{L3}$. Since none of $R_2$'s measurements have yet deviated, we know that the fault is local to $R_3$. Using relative measurement orderings, actuator faults occurring on different robots can be distinguished.

Table III shows the relative measurement orderings for the fault parameters of $R_2$, for measurements associated with $R_2$, $R_3$, and $R_6$ in the six-robot formation. Orderings implied by transitivity are omitted from the table. As evidenced by Table III, faults manifest first in their associated measurement before other measurements in the system (e.g., actuator and encoder faults manifest first in velocity measurements of that wheel).

TABLE III
RELATIVE MEASUREMENT ORDERINGS FOR $R_2$ IN THE SIX-ROBOT FORMATION FOR MEASUREMENTS OF $R_2$, $R_3$, AND $R_6$

| Faults | Relative Measurement Orderings |
|---|---|
| $MSe_{L,2}, G_{EL,2}$ | $v_{L2} \prec v_{R2}, v_{L2} \prec \theta_2,$<br>$v_2 \prec v_{R2}, v_2 \prec \theta_2,$<br>$v_{L2} \prec v_{L3}, v_{L2} \prec v_{R3}, v_{L2} \prec v_3,$<br>$v_2 \prec v_{L3}, v_2 \prec v_{R3}, v_2 \prec v_3,$<br>$v_3 \prec \theta_3, v_3 \prec v_6,$<br>$v_{L2} \prec v_{L6}, v_{L2} \prec v_{R6}, v_{L2} \prec v_6,$<br>$v_2 \prec v_{L6}, v_2 \prec v_{R6}, v_2 \prec v_6,$<br>$v_6 \prec \theta_6$ |
| $MSe_{R,2}, G_{ER,2}$ | $v_{R2} \prec v_{L2}, v_{R2} \prec \theta_2,$<br>$v_2 \prec v_{R2}, v_2 \prec \theta_2,$<br>$v_{R2} \prec v_{L3}, v_{R2} \prec v_{R3}, v_{R2} \prec v_3,$<br>$v_2 \prec v_{L3}, v_2 \prec v_{R3}, v_2 \prec v_3,$<br>$v_3 \prec \theta_3, v_3 \prec v_6,$<br>$v_{R2} \prec v_{L6}, v_{R2} \prec v_{R6}, v_{R2} \prec v_6,$<br>$v_2 \prec v_{L6}, v_2 \prec v_{R6}, v_2 \prec v_6,$<br>$v_6 \prec \theta_6$ |
| $B^+_{gyro,2}, B^-_{gyro,2}$ | $\theta_2 \prec v_{L2}, \theta_2 \prec v_{R2},$<br>$\theta_2 \prec v_{L3}, \theta_2 \prec v_{R3}, \theta_2 \prec v_3,$<br>$v_3 \prec \theta_3, v_3 \prec v_6,$<br>$\theta_2 \prec v_{L6}, \theta_2 \prec v_{R6}, \theta_2 \prec v_6,$<br>$v_6 \prec \theta_6$ |

Because faults cannot propagate in the opposite direction, faults in $R_3$ and $R_6$ will both have orderings in the format of $m_i \prec_f m_j$, where $m_i$ is a local measurement, $f$ is a local fault, and $m_j$ is a measurement of $R_2$. If a fault occurs in $R_2$, either $v_2$ or $\theta_2$ will deviate before any measurement in $R_3$. Therefore, to distinguish between $R_2$'s faults and $R_3$'s faults, one of these measurements will be useful. Which one is useful depends on whether the fault is an actuator or encoder fault (where $v_2$ is useful) or a gyroscope fault (where $\theta_2$ is useful). This results in the following lemma.

*Lemma 1:* Faults appearing in a parent $R_p \in Par(R_i)$ in $\mathcal{G}$ can be distinguished from faults appearing in $R_i$ using orderings for both $v_p$ and $\theta_p$ and local measurements of $R_i$.

*Proof:* Given $R_p \in Par(R_i)$, all $f_i$ of $R_i$ do not manifest in $R_p$, because there is no causal path from $R_i$ to $R_p$ since $\mathcal{G}$ is acyclic. So, we have the orderings $m_i \prec_{f_i} v_p$ and $m_i \prec_{f_i} \theta_p$ for each $f_i$ of $R_i$. From the TCG analysis, each fault $f_p$ of $R_p$ passes through either $v_p$ or $\theta_p$ before any measurement of $R_i$, thus resulting in either $v_p \prec_{f_p} m_i$ or $\theta_p \prec_{f_p} m_i$ for all $m_i$ of $R_i$ (Theorem 1). Therefore, the ordering sets will always conflict and we can use this information to distinguish among the faults. ■

It is important to note that we cannot derive orderings comparing a left or right velocity of $R_3$ to a left or right velocity of $R_6$. This is because the path to a left or right velocity measurement of $R_6$ could go through either the left or right velocity of $R_3$, and we don't know which path is faster. Thus, if an actuator fault occurs in $R_2$, $R_3$'s left and right velocity measurements are useless to distinguish between local and remote actuator faults. However, we do have the ordering $v_3 \prec v_6$ for all of $R_2$'s faults. Thus we can use $v_3$ to distinguish between actuator faults in $R_1$ and $R_6$. The ordering $v_3 \prec v_6$ essentially says that either $v_{L3}$ or $v_{R3}$ will deviate before any measurements of $R_6$, since remote faults

affect the velocity measurement first ($v_6 \prec \theta_6$ for all remote faults). It does not matter which of $v_{L3}$ or $v_{R3}$ deviates first, only knowing that one will deviate is helpful. So although $v_2$ provides no extra discriminatory information in terms of fault signatures, it is helpful in terms of measurement orderings. This results in the following lemma.

*Lemma 2:* Faults appearing in an ancestor $R_a \in Anc(R_i)$ in $\mathcal{G}$ such that $R_a \in Anc(R_p)$ and $R_p \in Par(R_i)$, can be distinguished from faults appearing in $R_i$ using orderings for $v_p$ and local measurements of $R_i$.

*Proof:* Given $R_p \in Par(R_i)$, and $R_a \in Anc(R_p)$, all $f_i$ of $R_i$ do not manifest in $R_p$, because there is no causal path from $R_i$ to $R_p$ since $\mathcal{G}$ is acyclic. So, we have the ordering $m_i \prec_{f_i} v_p$ for each $f_i$ of $R_i$. From the TCG analysis, each fault $f_a$ of $R_a$ passes through $v_p$ before any measurement of $R_i$, thus resulting in $v_p \prec_{f_a} m_i$ for all $m_i$ of $R_i$ (Theorem 1). Therefore the ordering sets will always conflict, and we can distinguish the faults. ■

The following theorem shows how local and remote faults can be discriminated.

*Theorem 2:* A fault is local if and only if a local measurement deviates before a remote measurement.

*Proof:* If a fault is local, a local measurement will deviate before any remote measurement because for every local fault there is some set of local measurements that deviate before every other measurement. If a local measurement deviates before a remote measurement, the fault must be local because for all non-local faults, the fault will manifest in a parent (if the fault originated in an ancestor) before the local robot (Lemmas 1 and 2), or in a child before the local robot (because faults in a child never manifest in their parents). ■

### *E. Distributed Diagnosis*

If the system is globally diagnosable, then a centralized diagnoser can be constructed that can uniquely isolate all faults. Such an approach, however, results in a very large diagnoser that becomes a single point of failure. The single point of failure can be avoided by replicating the centralized diagnoser on each robot, however, this will be inefficient for large formations. In addition, the diagnosers on each robot will perform unnecessary computations involving fault hypotheses that are not relevant to the particular robot. We instead take a distributed approach, where each local diagnoser isolates faults in its subsystem using local measurements and some remote measurements, if required. Since accessing remote measurements is expensive, our design goal is to find the minimum number of remote measurements that makes each subsystem globally diagnosable. The design approach ensures that a local diagnosis will be globally correct, because exactly one robot isolates the true fault and knows no remote faults could have occurred. Since the local diagnosers achieve a global diagnosis, this avoids the need for a centralized coordinator [17].

The algorithm generates the distributed diagnoser by minimizing the number of shared measurements between subsystems. For each subsystem, if a fault is not globally diagnosable using local measurements, it searches neighboring subsystems for a minimal set of additional measurements to make the fault globally diagnosable. The pseudocode is given as Algorithm 1. In the worst case, all combinations of measurements are considered, so the algorithm is exponential. From a practical viewpoint, since the diagnosers are built offline, their design time complexity is not of much concern.

**Algorithm 1** Distributed Diagnoser Design

```
Input: local fault sets F_i, local measurement sets M_i, fault signatures, ordering sets, k subsystems
for subsystem i ∈ 1, ..., k do
  identify set F'_i ⊆ F_i such that f ∈ F'_i cannot be completely distinguished using M_i
  for f ∈ F'_i do
    identify minimum set of communicated measurements to globally diagnose f
    add this set to the local measurement set
  end for
end for
```

For the formation system, the subsystems are the individual robots. The diagnoser for $R_i$ is responsible for diagnosing faults in the set $F_i = \{MSe^-_{L,i}, MSe^-_{R,i}, G^-_{EL,i}, G^-_{ER,i}, B^+_{gyro,i}, B^-_{gyro,i}\}$ using measurements $M_i = \{v_{L,i}, v_{R,i}, v_i, \theta_i\}$, i.e., each robot is responsible for diagnosing faults in its components using its local measurements.

Running the algorithm shows that each robot must be communicated the velocity and heading measurements of its parents in the fault propagation graph. This ensures that each robot has enough information to produce an independent, globally correct diagnosis. From Lemma 1, each robot will need both $v$ and $\theta$ measurements of each parent in the fault propagation graph, in order to distinguish between local faults and those appearing in the parents. From Lemma 2, these measurements are enough to distinguish between local faults and those appearing in the ancestors in the fault propagation graph, therefore, these are the minimal communicated measurements.

Additionally, the local $v$ measurement is not necessary to distinguish local and remote faults because $v_L$ and $v_R$ provide the same information in this respect[2]. Essentially, the discriminatory information that the remote $v$ and $\theta$ measurements provide is that if a remote fault occurs, it will manifest in one of the remote $v$ or $\theta$ measurements before any local measurement, thus allowing the diagnosers to distinguish between local and remote faults.

Table IV illustrates the individual fault and measurement sets for the diagnosers in the six-robot formation. It is important to note that not all robots require remote measurements to determine a globally correct local diagnosis. Some of the robots ($R_1$, $R_2$, and $R_4$) require only local measurements, i.e., these diagnosers are independent. This is the case when the robot is a source vertex in the fault propagation graph, and occurs because the fault effects of other robots cannot propagate to it. Therefore, any fault effects it observes are known to be caused by a local fault.

[2]Alternatively, $v$ could be kept and $v_L$ and $v_R$ dropped, because the system would still be diagnosable. However, we opt to keep $v_L$ and $v_R$ instead so that we do not have to wait for $\theta$ to deviate in order to distinguish between faults of the left and right wheels.

TABLE IV
DISTRIBUTED DIAGNOSER DESIGN FOR THE SIX-ROBOT EXAMPLE

| Robot | Fault Set | Measurement Set |
|---|---|---|
| $R_1$ | $\{MSe^-_{L,1}, MSe^-_{R,1}, G^-_{EL,1}, G^-_{ER,1}, B^+_{gyro,1}, B^-_{gyro,1}\}$ | $\{v_{L,1}, v_{R,1}, \theta_1\}$ |
| $R_2$ | $\{MSe^-_{L,2}, MSe^-_{R,2}, G^-_{EL,2}, G^-_{ER,2}, B^+_{gyro,2}, B^-_{gyro,2}\}$ | $\{v_{L,2}, v_{R,2}, \theta_2\}$ |
| $R_3$ | $\{MSe^-_{L,3}, MSe^-_{R,3}, G^-_{EL,3}, G^-_{ER,3}, B^+_{gyro,3}, B^-_{gyro,3}\}$ | $\{v_{L,3}, v_{R,3}, \theta_3, v_2, \theta_2\}$ |
| $R_4$ | $\{MSe^-_{L,4}, MSe^-_{R,4}, G^-_{EL,4}, G^-_{ER,4}, B^+_{gyro,4}, B^-_{gyro,4}\}$ | $\{v_{L,4}, v_{R,4}, \theta_4\}$ |
| $R_5$ | $\{MSe^-_{L,5}, MSe^-_{R,5}, G^-_{EL,5}, G^-_{ER,5}, B^+_{gyro,5}, B^-_{gyro,5}\}$ | $\{v_{L,5}, v_{R,5}, \theta_5, v_4, \theta_4\}$ |
| $R_6$ | $\{MSe^-_{L,6}, MSe^-_{R,6}, G^-_{EL,6}, G^-_{ER,6}, B^+_{gyro,6}, B^-_{gyro,6}\}$ | $\{v_{L,6}, v_{R,6}, \theta_6, v_3, \theta_3, v_5, \theta_5\}$ |

These results easily extend to arbitrary formations that satisfy the constraints of [1], [2]. The fault propagation graph $\mathcal{G}$ can be derived from the control graph $\mathcal{C}$. The diagnoser design is direct from $\mathcal{G}$ since each robot needs the velocity and heading measurements of each parent in $\mathcal{G}$.

Like a centralized diagnoser, each local diagnoser runs an online fault isolation algorithm [8]. The algorithm starts with the set of local fault candidates and their associated fault signatures after an initial deviation has been detected. It matches the candidates' predicted fault signatures to observed measurement deviations as they appear, dropping candidates whose signatures are inconsistent with observed transients. Candidates are dropped if there exists an inconsistency between predicted and observed fault signatures or predicted and observed measurement orderings. Using relative measurement orderings makes fault isolation more efficient, because less measurements are required to uniquely isolate a fault, and the knowledge that a certain measurement has not yet deviated provides useful information.

### F. Scalability

The scalability of the approach can be characterized using two metrics, the size of the diagnoser and the number of communicated measurements, quantifying the computational and communication requirements, respectively. Let $F$ represent the complete fault set, $M$ the complete measurement set, and $n$ the number of robots. Also, let $F_i$ be the fault set of $R_i$, and $M_i^+$ be the measurement set for $R_i$ determined by the distributed diagnoser design algorithm. In a centralized approach, the central diagnoser must diagnose all faults in $F$ using measurements in $M$. The size of the diagnoser is then $S_C = |F||M| + |F||M|^2$ so that it can store both signatures and orderings for each fault. In the replicated centralized diagnoser approach, each diagnoser is of size $S_C$, resulting in $nS_C$ space. In the proposed distributed approach, however, $R_i$ must only diagnose faults in $F_i$ using measurements in $M_i^+$, resulting in space complexity $S_i = |F_i||M_i| + |F_i||M_i^+|^2$. The total space required for all individual distributed diagnosers will always be less than that of a centralized diagnoser, i.e., $\sum_i S_i < S_C$ if not all measurements are communicated. The reason is that some of the information is discarded because it is not useful in the local diagnosers. For example, we don't need to store anywhere the effects of $R_2$'s faults on $R_1$'s measurements, because none of $R_1$'s measurements are needed to diagnose $R_2$'s faults. Diagnoser size directly relates to diagnostic efficiency. The smaller the diagnoser size, the smaller the number of faults and measurements to consider, and thus the greater its computational efficiency.

The number of communicated measurements characterize the communication overhead incurred by the distributed algorithm. In a centralized approach, each robot must communicate its measurements to the centralized diagnoser, resulting in a total of $|M|$ communicated measurements. In a replicated centralized diagnoser approach, each robot would have to communicate its measurements to all other robots, resulting in a total of $\sum_i (n-1)|M_i|$ communicated measurements. In our distributed approach, however, communication is minimized by the diagnoser design algorithm. From Lemmas 1 and 2, only the velocity ($v$) and heading ($\theta$) measurements are required from each parent in the fault propagation graph. Therefore, at most two measurements must be communicated to each robot (except the formation leader) from each local leader (at most two), resulting in at most $4(n-1)$ communicated measurements for the worst case. Hence, the number of communicated measurements required per robot is independent of formation size. The total number of communicated measurements for all robots is linear in the formation size, so, like the centralized case, the approach scales linearly with large formations. In the six-robot example used throughout the paper, there are 4 edges in $\mathcal{G}$, resulting in a total of 8 communicated measurements for the distributed approach. For a centralized approach, since each robot has 3 measurements in its measurement set, 18 measurements must be communicated.

## VI. EXPERIMENTAL RESULTS

The effectiveness of the distributed detection and isolation algorithms is demonstrated in a laboratory setting with four ActivMedia Pioneer 3-DX mobile robots moving in the formation illustrated in Fig. 7. The robots communicate over an 802.11b wireless ad-hoc network. Fig. 8 shows the nominal trajectories of the robots moving at a pre-specified speed of 0.1 m/s. The experiment is run for 40 s. The top plot shows the robot trajectories, with their starting and ending locations drawn. The lower left plot shows the robot velocities, and the lower right plot shows their headings. All the robots maintain the shape variables, so the formation is maintained. All faults listed in Table I were introduced through software. The sampling period of the distributed controllers and diagnosers was 0.1 s. At the selected sampling rate, the packet loss was negligible (measured less than 0.1%). Since communication is expected at the selected sampling rate, persistent errors in the network links can be easily diagnosed by software (viewed as additional diagnosers) and are not considered here.

For this four-robot formation, the fault propagation graph includes only the edges $(R_2, R_3)$, $(R_2, R_4)$, and $(R_3, R_4)$. Therefore, $R_3$ requires measurements from $R_2$ ($v_2$ and $\theta_2$),

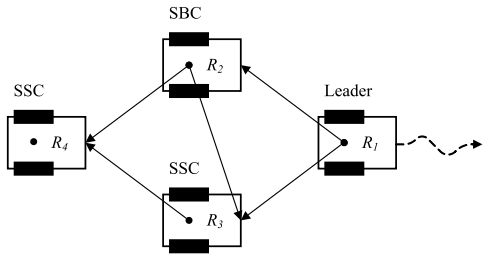


Fig. 7. Experimental setup.

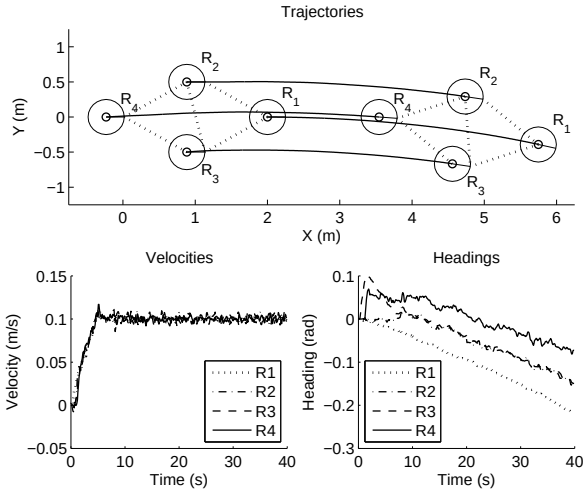


Fig. 8. Nominal system behavior.

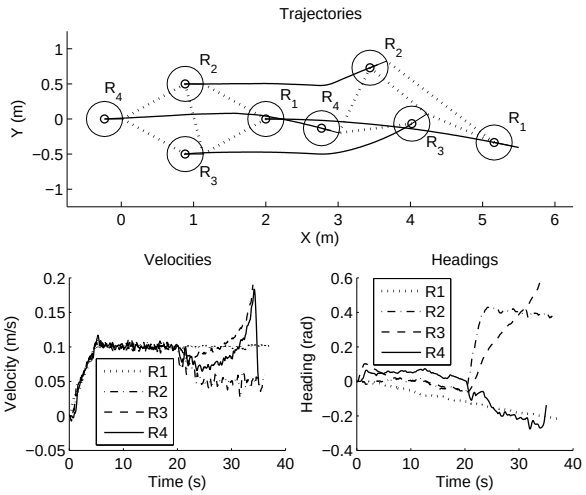


Fig. 9. System behavior with $MSe^-_{R,2}$ occurring with 0.05 m/s magnitude.

TABLE V
DIAGNOSIS TRACE FOR LEFT ACTUATOR FAULT OF $R_2$

| Time | Event | $R_1$ | $R_2$ | $R_3$ | $R_4$ |
|---|---|---|---|---|---|
| 20.0 | Fault injected ($R_2$) | $\emptyset$ | $\emptyset$ | $\emptyset$ | $\emptyset$ |
| 20.4 | Fault detected ($R_2$) | $\emptyset$ | $F_2$ | $F_3$ | $F_4$ |
| | $v_{L2}$ deviates | $\emptyset$ | $\{MSe^-_{L,2}, G^-_{EL,2}\}$ | $\emptyset$ | $\emptyset$ |
| 20.7 | $\theta_2$ deviates | $\emptyset$ | $\{MSe^-_{L,2}, G^-_{EL,2}\}$ | $\emptyset$ | $\emptyset$ |
| 21.0 | $v_{L2}$ 0− determined | $\emptyset$ | $\{MSe^-_{L,2}\}$ | $\emptyset$ | $\emptyset$ |
| | Diagnosis ($R_2$) | $\emptyset$ | $MSe^-_{L,2}$ | $\emptyset$ | $\emptyset$ |

and $R_4$ requires measurements from both $R_2$ and $R_3$ ($v_2$, $\theta_2$, $v_3$, and $\theta_3$). $R_1$ and $R_2$ are source vertices so they require only local measurements.

In the following, we illustrate our approach for an actuator fault of the left wheel of $R_2$ ($MSe^-_{R,2}$) at a magnitude of 0.05 m/s, i.e., the wheel velocity saturates at half the desired speed. Fig. 9 shows the faulty trajectories for the robots, and Table V traces the diagnosis steps. Initially, the diagnosers assume empty fault sets. The fault is injected at $t$ = 20.0 s. It causes the left wheel to slow down, therefore, the heading deviates, and the right wheel begins to speed up to keep its separation with $R_1$. $R_3$ and $R_4$ begin to slow down to maintain their positions with respect to $R_2$. A deviation in $v_{L,2}$ at 20.4 s triggers $R_2$'s fault isolation procedure. Six steps later the deviation is determined not to be discontinuous, i.e., the change in the measurement is smooth, not abrupt.

$R_2$ starts with its entire fault set, $F_2$, as the set of possible candidates. As predicted, $v_{L,2}$ is the first deviation, so based on orderings, the fault set is reduced to the faults of the left wheel. The change of $v_{L,2}$ matches the fault signature of $0-$, thus isolating the fault to be $MSe^-_{L,2}$. By design, this is guaranteed to be the globally unique fault, so recovery actions may commence and the other robots notified. Only one measurement deviation was needed to obtain a global diagnosis, so this demonstrates the efficiency of using relative measurement orderings in fault isolation. Because a communicated remote measurement ($v_2$) has deviated before any local measurements, $R_3$ and $R_4$ can eliminate all their local faults and determine a remote fault has occurred. $R_1$ does not observe a deviation in any of its local measurements so it does not produce a diagnosis.

We illustrate our approach now for an encoder fault of the right wheel of $R_2$ ($G^-_{ER,2}$) at a magnitude of 30%, i.e., the encoder misses 30% of its counts. Fig. 10 shows the faulty trajectories for the robots, and Table VI traces the diagnosis steps. Initially, the diagnosers assume empty fault sets. The fault is injected at $t$ = 20.0 s. It causes an abrupt decrease in the right velocity measurement, causing the right wheel to speed up. Therefore, the heading deviates, and the left wheel begins to slow down to keep its separation with $R_1$. $R_3$ and $R_4$ begin to speed up to maintain their positions with respect to $R_2$. A deviation in $v_{R,2}$ at 20.1 s triggers $R_2$'s fault isolation procedure. Six steps later the deviation is labelled as discontinuous.

$R_2$ starts with its entire fault set, $F_2$, as its set of possible candidates. As predicted, $v_{R,2}$ is the first deviation, so based on orderings, the fault set is reduced to only faults of the right wheel. The change of $v_{R,2}$ matches the fault signature of $-+$, thus isolating the fault to be $G^-_{ER,2}$. Again, by design, this is guaranteed to be the globally unique fault. For this example too, only one measurement deviation was needed to obtain a global diagnosis, so this demonstrates the efficiency of using relative measurement orderings in fault isolation. Because a communicated remote measurement ($v_2$) has deviated before

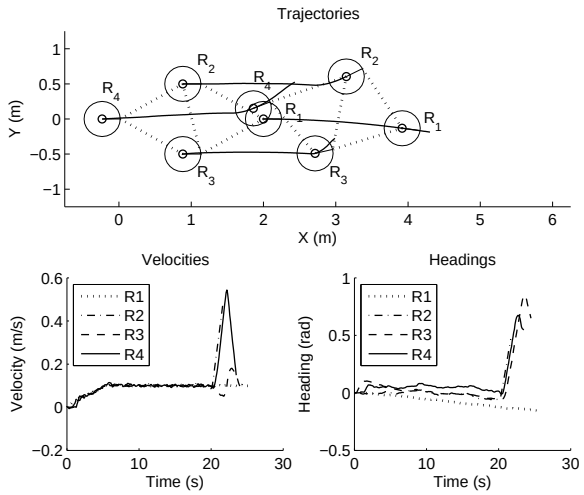


Fig. 10. System behavior with $G^-_{ER,2}$ occurring with 30% magnitude.

TABLE VI
DIAGNOSIS TRACE FOR RIGHT ENCODER FAULT OF $R_2$

| Time | Event | $R_1$ | $R_2$ | $R_3$ | $R_4$ |
|---|---|---|---|---|---|
| 20.0 | Fault injected ($R_2$) | $\emptyset$ | $\emptyset$ | $\emptyset$ | $\emptyset$ |
| 20.1 | Fault detected ($R_2$) | $\emptyset$ | $F_2$ | $F_3$ | $F_4$ |
| | $v_{R2}$ deviates | $\emptyset$ | $\{MSe^-_{R,2}, G^-_{ER,2}\}$ | $\emptyset$ | $\emptyset$ |
| 20.4 | $v_{R4}$ deviates | $\emptyset$ | $\{MSe^-_{R,2}, G^-_{ER,2}\}$ | $\emptyset$ | $\emptyset$ |
| 20.5 | $v_{L2}$ deviates | $\emptyset$ | $\{MSe^-_{R,2}, G^-_{ER,2}\}$ | $\emptyset$ | $\emptyset$ |
| 20.6 | $\theta_2$ deviates | $\emptyset$ | $\{MSe^-_{R,2}, G^-_{ER,2}\}$ | $\emptyset$ | $\emptyset$ |
| 20.7 | $v_{L2}$ $-+$ determined | $\emptyset$ | $\{G^-_{ER,2}\}$ | $\emptyset$ | $\emptyset$ |
| | Diagnosis ($R_2$) | $\emptyset$ | $G^-_{ER,2}$ | $\emptyset$ | $\emptyset$ |

any local measurements, $R_3$ and $R_4$ can eliminate all their local faults and determine a remote fault has occurred. $R_1$ does not observe a deviation in any of its local measurements so it does not produce a diagnosis.

All faults of interest, listed in Table I, were successfully isolated using the distributed diagnoser. The summary of the diagnosis results is shown in Table VII. Due to the high discriminatory power the combination of fault signatures and relative measurement orderings provide, all faults could be isolated with only a single measurement deviating. The magnitude of the fault and its time of injection are shown, along with all measurement deviations observed until a global diagnosis is known. All faults were injected at 20 s. Beside each measurement deviation is the time of detection followed by the time at which it was determined whether or not a discontinuity occurred. The approach is applicable for smaller fault magnitudes, as long as the fault detector is appropriately tuned. If the fault detector and symbol generation work correctly, then by construction the fault isolation will always execute correctly. The fault detector was tuned for the laboratory setting and the fault magnitudes under consideration. Multiple experiments were performed to achieve reliable fault detection, and, therefore, no false positives occurred. Because the fault magnitudes were sufficiently large compared to the system noise, false negatives did not occur either.

TABLE VII
DIAGNOSIS RESULTS

| Fault | Magnitude | Diagnosis Trace |
|---|---|---|
| $MSe^-_{L,2}$ | 90 mm/s | $v_{L2}$ $0-$ 20.4-21.0 s |
| $MSe^-_{L,2}$ | 70 mm/s | $v_{L2}$ $0-$ 20.3-20.9 s |
| $MSe^-_{L,2}$ | 50 mm/s | $v_{L2}$ $0-$ 20.4-21.0 s |
| $G^-_{ER,2}$ | 10% | $v_{R2}$ $-+$ 20.1-20.6 s |
| $G^-_{ER,2}$ | 30% | $v_{R2}$ $-+$ 20.1-20.7 s |
| $G^-_{ER,2}$ | 50% | $v_{R2}$ $-+$ 20.0-20.0 s |
| $B^+_{gyro,2}$ | 0.08 rad | $\theta_2$ $+-$ 20.1-20.1 s |
| $B^+_{gyro,2}$ | 0.1 rad | $\theta_2$ $+-$ 20.1-20.1 s |
| $B^-_{gyro,2}$ | -0.08 rad | $\theta_2$ $-+$ 20.1-20.1 s |
| $B^-_{gyro,2}$ | -0.1 rad | $\theta_2$ $-+$ 20.0-20.0 s |
| $MSe^-_{L,3}$ | 50 mm/s | $v_{L3}$ $0-$ 20.4-21.0 s |
| $MSe^-_{R,3}$ | 50 mm/s | $v_{R3}$ $0-$ 20.3-20.9 s |

## VII. CONCLUSIONS

In this paper we described an approach for distributed diagnosis in formations of mobile robots. We derived the system model encompassing the plant, sensors, actuators, communication, and control. The DDEKF scheme was applied for distributed estimation and tracking of nominal system behavior, and the Z-test was used for robust fault detection. The qualitative fault isolation scheme combined the use of fault signatures and relative measurement orderings, increasing the discriminatory power of the measurement sets. Measurement orderings were shown to be necessary to ensure diagnosability in the formation systems studied in this paper. Using both signatures and orderings, diagnosers can require fewer measurements, and diagnosis results are achieved faster. Distributed diagnosers were designed from a global system model, and the diagnosis scheme was shown to scale well with formation size. The design was such that each local diagnosis was globally correct, thus circumventing the need for a centralized coordinator. Experimental results demonstrated the validity and usefulness of the approach.

Future work will address the current limitations of the approach. Inclusion of incipient fault profiles and diagnosis of multiple faults are important, as well as addressing discrete and coordination failures. With these addressed, the diagnosis approach can be integrated into a fault-adaptive control architecture. Formations where the relations between the robots change over time may be addressed by adding machinery to reconfigure the local diagnosers appropriately.

## 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] J. P. Desai, J. P. Ostrowski, and V. Kumar, "Modeling and control of formations of nonholonomic mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 905–908, Dec 2001.

[2] ——, "Controlling formations of multiple mobile robots," in *Proceedings 1998 IEEE International Conference on Robotics and Automation*, 1998, pp. 2864–2869.

[3] T. Balch and R. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, Dec. 1998.

[4] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[5] L. E. Parker, "ALLIANCE: An architecture for fault tolerant multirobot cooperation," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 2, pp. 220–240, Apr. 1998.

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

[7] M. Daigle, X. Koutsoukos, and G. Biswas, "Distributed diagnosis of coupled mobile robots," in *Proceedings 2006 IEEE International Conference on Robotics and Automation*, May 2006, pp. 3787–3794.

[8] P. 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.

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

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

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

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

[13] J. de Kleer and B. C. Williams, "Diagnosis with behavioral modes," in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*. Detroit, MI, USA: Morgan Kaufmann, Aug. 1989, pp. 1324–1330.

[14] M. A. Demetriou and M. M. Polycarpou, "Incipient fault diagnosis of dynamical systems using on-line approximators," *IEEE Transactions on Automatic Control*, vol. 43, no. 11, pp. 1692–1617, Nov. 1998.

[15] I. Roychoudhury, G. Biswas, and X. Koutsoukos, "A Bayesian approach to efficient diagnosis of incipient faults," in *Proceedings of the 17th International Workshop on Principles of Diagnosis (DX 06)*, June 2006, pp. 243–250.

[16] M. Daigle, X. Koutsoukos, and G. Biswas, "Multiple fault diagnosis in complex physical systems," in *Proceedings of the 17th International Workshop on Principles of Diagnosis (DX 06)*, June 2006, pp. 69–76.

[17] I. Roychoudhury, G. Biswas, X. Koutsoukos, and S. Abdelwahed, "Designing distributed diagnosers for complex physical systems," in *Proceedings of the 16th International Workshop on Principles of Diagnosis (DX 05)*, Monterey, California, June 2005, pp. 31–36.

[18] R. J. Patton and J. Chen, "Observer-based fault detection and isolation: robustness and applications," *Control Engineering Practice*, vol. 5, no. 5, pp. 671–682, 1997.

[19] R. Isermann, "Supervision, fault-detection, and fault-diagnosis methods — and introduction," *Control Engineering Practice*, vol. 5, no. 5, pp. 639–652, 1997.

[20] 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.

[21] R. Rajamani, A. S. Howell, C. Chen, J. K. Hedrick, and M. M. Tomizuka, "A complete fault diagnostic system for automated vehicles operating in a platoon," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 4, pp. 553–564, July 2001.

[22] 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.

[23] V. Venkatasubramanian, R. Rengaswamy, and S. N. Kavuri, "A review of process fault detection and diagnosis Part II: Qualitative models and search strategies," *Computers and Chemical Engineering*, vol. 27, pp. 313–326, 2003.

[24] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin, "A review of process fault detection and diagnosis Part III: Process history based methods," *Computers and Chemical Engineering*, vol. 27, pp. 327–346, 2003.

[25] Z. Duan, Zi-xing Cai, and J. Yu, "Fault diagnosis and fault tolerant control for wheeled mobile robots under unknown environments: A survey," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, April 2005, pp. 3428–3433.

[26] M. Ji, Z. Zhang, G. Biswas, and N. Sarkar, "Hybrid fault adaptive control of a wheeled mobile robot," *IEEE/ASME Transactions on Mechatronics*, vol. 8, no. 2, pp. 226–233, June 2003.

[27] B. Halder and N. Sarkar, "Robust fault detection based on nonlinear analytic redundancy techniques with application to robotics," in *Proceedings of IMECE*, Nov. 2005, pp. 2864–2869.

[28] V. Verma, G. Gordon, R. Simmons, and S. Thrun, "Real-time fault diagnosis," *IEEE Robotics & Automation Magazine*, vol. 11, no. 2, pp. 56–66, June 2004.

[29] R. Dearden and D. Clancy, "Particle filters for real-time fault detection in planetary rovers," in *Proceedings of the International Workshop on Principles of Diagnosis (DX'02)*, Semmering, Austria, 2002, pp. 1–6.

[30] W. Dixon, I. Walker, and D. Dawson, "Fault detection for wheeled mobile robots with parametric uncertainty," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, vol. 2, July 2001, pp. 1245–1250.

[31] S. Roumeliotis, G. Sukhatme, and G. Bekey, "Sensor fault detection and identification in a mobile robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, Victoria, BC, Canada, Oct 1998, pp. 1383–1388.

[32] P. Goel, G. Dedeoglu, S. Roumeliotis, and G. Sukhatme, "Fault detection and identification in a mobile robot using multiple model estimation and neural network," in *IEEE International Conference on Robotics and Automation*, vol. 3, 2000, pp. 2302–2309.

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

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

[35] P. Fröhlich, I. de Almeida Móra, W. Nejdl, and M. Schroeder, "Diagnostic agents for distributed systems," in *ModelAge Workshop*, J.-J. C. Meyer and P.-Y. Schobbens, Eds., vol. 1760. Springer, 1997, pp. 173–186.

[36] N. Roos, A. ten Teije, and C. Witteveen, "A protocol for multi-agent diagnosis with spatially distributed knowledge," in *Proceedings of the AAMAS*. ACM, 2003, pp. 655–661.

[37] M. Kalech and G. A. Kaminka, "Towards model-based diagnosis of coordination failures," in *American Association for Artificial Intelligence*, M. M. Veloso and S. Kambhampati, Eds. AAAI Press; AAAI Press / The MIT Press, 2005, pp. 102–107.

[38] M. Kalech, G. A. Kaminka, A. Meisels, and Y. Elmaliach, "Diagnosis of multi-robot coordination failures using distributed CSP algorithms," in *American Association for Artificial Intelligence*. AAAI Press, 2006.

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

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

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

[42] M. Basseville and I. Nikiforov, *Detection of Abrupt Changes - Theory and Application*. Prentice-Hall Inc., 1993.

[43] E.-J. Manders, P. Mosterman, and G. Biswas, "Signal to symbol transformation techniques for robust diagnosis in TRANSCEND," in *Proceedings of the 10th International Workshop on Principles of Diagnosis*, 1999, pp. 155–165.

[44] E.-J. Manders and G. Biswas, "FDI of abrupt faults with combined statistical detection and estimation and qualitative fault isolation," in *Proceedings of the 5th Symposium on Fault Detection, Supervision and Safety for Technical Processes*, June 2003, pp. 347–352.

**Matthew J. Daigle** received the B.S. degree in computer science and computer and systems engineering from Rensselaer Polytechnic Institute, Troy, NY, in 2004, and the M.S. degree in computer science from Vanderbilt University, Nashville, TN, in 2006.

Since September 2004, he has been a Graduate Research Assistant with the Institute for Software Integrated Systems and Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, working towards the Ph.D. degree in computer science. During the summer of 2006, he was an intern with Mission Critical Technologies, Inc., at NASA Ames Research Center. His current research interests include model-based diagnosis, multi-robot systems, and hybrid systems.

Mr. Daigle is a recipient of the 4.0 Award and Ricketts Prize from Rensselaer Polytechnic Institute, and a University Graduate Fellowship from Vanderbilt University.

**Xenofon D. Koutsoukos** (S'95–M'00) received his Diploma in Electrical and Computer Engineering from the National Technical University of Athens (NTUA), Greece in 1993, M.S. degrees in Electrical Engineering and Applied Mathematics in 1998, and his Ph.D. in Electrical Engineering in 2000 from the University of Notre Dame.

From 2000 to 2002, he was a Member of Research Staff in the Xerox Palo Alto Research Center (PARC) working in the Embedded Collaborative Computing Area. Since 2002, he has been with the Department of Electrical Engineering and Computer Science at Vanderbilt University where he is currently an Assistant Professor and a Senior Research Scientist in the Institute for Software Integrated Systems (ISIS). His research work is in the area of hybrid systems, real-time embedded systems, and sensor networks. He has authored or co-authored more than 60 technical publications and is co-inventor of three US patents.

Dr. Koutsoukos is a recipient of the National Science Foundation CAREER award in 2004, he currently serves as Associate Editor of the ACM Transactions on Sensor Networks, and he is a member of IEEE and ACM.

**Gautam Biswas** (S'78–M'82–SM'91) is a Professor of Computer Science and Computer Engineering in the EECS Department and a Senior Research Scientist at the Institute for Software Integrated Systems (ISIS) at Vanderbilt University. He has a Ph.D. degree in Computer Science from Michigan State University in E. Lansing, MI.

Prof. Biswas 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. His research has been supported by funding from NASA, NSF, DARPA, and ONR.

Dr. Biswas is an associate editor of the IEEE Transactions on Systems, Man, and Cybernetics. He has served on the Program Committee of a number of conferences. He is a senior member of the IEEE Computer Society, ACM, AAAI, and the Sigma Xi Research Society.