Progress in Flight Dynamics, GNC, and Avionics **6** (2013) 281-298
DOI: 10.1051/eucass/201306281

A DECENTRALIZED FAULT DETECTION AND ISOLATION SCHEME FOR SPACECRAFT: BRIDGING THE GAP BETWEEN MODEL-BASED FAULT DETECTION AND ISOLATION RESEARCH AND PRACTICE

S. Indra[1,4], L. Travé-Massuyès[1,2], and E. Chanthery[1,3]

[1]CNRS; LAAS
7 Av. du Colonel Roche, Toulouse 31400, France
[2]University of Toulouse, LAAS
Toulouse 31400, France
[3]University of Toulouse, INSA
Toulouse 31400, France
[4]Center National d'Etudes Spatiales
Toulouse, France

This paper introduces a decentralized fault diagnosis and isolation (FDI) architecture for spacecraft and applies it to the attitude determination and control system (ADCS) of a satellite. A system is decomposed into functional subsystems. The architecture is composed of local diagnosers for subsystems which work with local models. Fault ambiguities due to interactions between subsystems are resolved at a higher level by a supervisor, which combines the partial view of the local diagnosers and performs isolation on request. The architecture is hierarchically scalable. The structure of the ADCS is modeled as constraints and variables and used to demonstrate the decentralized architecture.

1 INTRODUCTION

Modern spacecrafts are complex systems, with extremely high requirements on reliability. In ground-based systems, reliability can often be achieved through hardware redundancy. However, designing aerospace systems involves trading off between tough competing requirements, with hardware redundancy very costly in terms of size, weight, and complexity. Therefore, only a few of the most critical

components can usually be made physically redundant. Analytical redundancy can be a powerful alternative means of ensuring functional reliability. Analytical redundancy involves comparing the behavior of a system with a model of its expected behavior.

Fault detection and isolation based on a model of the system, known as model-based diagnosis (MBD) is one approach to using analytical redundancy to increase the reliability of a system. Reconfiguration actions can be initiated after the FDI phase. There is a wide gap between the theory of MBD and its adoption for real space missions due to lack of mission pull [1]. The costs associated with MBD stem largely from the high complexity of the algorithms and the modeling effort involved in diagnoser design. Efforts towards bridging the gap between theory and practice should focus, first, on considering realistic fault scenarios in the design phase. Second, it might be worthwhile to work towards realistic goals in the short term and use the experience gained to guide the development of more ambitious MBD applications. In this way, the cost-value trade-off for adopting MBD for space vehicles might be made more favorable. Automatic monitoring of housekeeping data and constructing decision support systems for operators and astronauts are some such applications. These systems should be integrated into existing operational procedures. Most complex systems can be decomposed functionally into subsystems. In the aerospace industry as in many others, the system integrator is responsible for defining the systems and the interfaces of subsystems which are then constructed and provided by subsuppliers. The diagnosis modules associated with the subsystems would also usually be designed by the subsuppliers.

This paper suggests a scheme for the decentralized diagnosis of space systems. The architecture is composed of local diagnosers working with local models of their subsystems, with their knowledge of the environment around them limited to information about which variables interface with other subsystems. The local diagnosers attempt to explain anomalies detected in their subsystems. The quantities exchanged with other subsystems are ignored and this might lead to ambiguities. These ambiguities are resolved at a higher level by a supervisory diagnoser. The architecture is hierarchically scalable, which means that local diagnosers of a level can act as supervisors for lower level diagnosers working on constituent components. The diagnoser takes into account a model of the system and also the anticipated faults in the design phase. The proposed diagnosis architecture is shown in Fig. 1 as applied to the subsystems of a satellite. In this paper, the FDI scheme is applied to the ADCS of a satellite, with the attitude determination (ADS) and attitude control (ACS) considered as subsystems with local diagnosers, and a supervisory diagnoser at the global ADCS level.

The model utilized by FDI algorithms can vary in framework and granularity. The present work deals with the decentralized diagnosis of systems modeled as continuous time systems. In particular, the Analytical Redundancy Relation (ARR) approach was utilized to FDI within a structural framework. Such an ap-

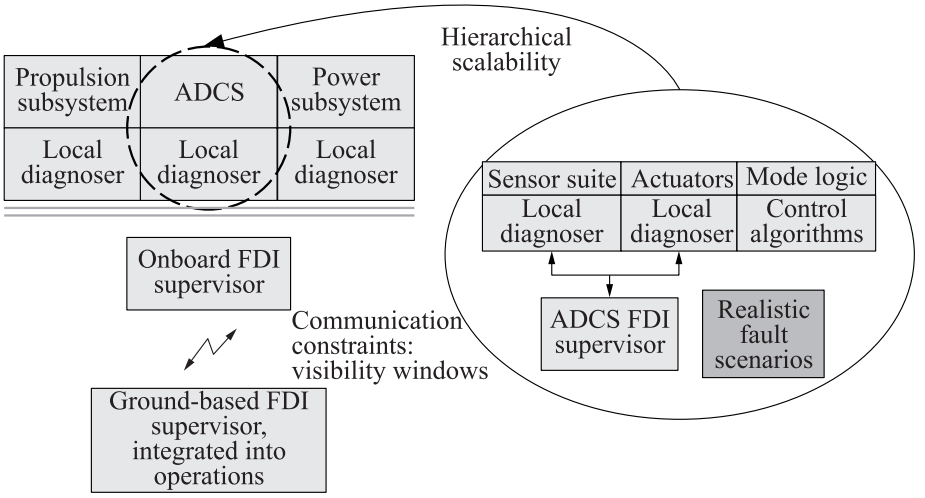

Figure 1 Decentralized diagnosis architecture applied to a satellite bus

proach is developed in [2] which describes an algorithm to analyze the structure of a system detecting redundant portions for use in ARR-based diagnosis methods. This approach is further developed in [3,4]. While [3] includes information about interesting faults to increase the efficiency of the algorithm, [4] provides a transition from structural analysis to analytical computation of residual generators.

There has been considerable recent work aiming to apply model-based FDI of continuous systems to aerospace systems and operations. Most of these works utilize Kalman filter or observer banks to model the nominal and faulty behaviors of the system. The works discussed below make an attempt to include real world constraints and considerations into the design phase of the FDI module.

The design of a decision support system for automated monitoring of reaction wheel telemetry is illustrated in [5]. A Kalman filter bank is used to detect and isolate faults with an Interacting Multiple Model (IMM) algorithm. A high fidelity reaction wheel model from [6] is utilized to demonstrate the effectiveness of the designed diagnoser.

An FDI module for the aerodynamic control surfaces of an atmospheric reentry vehicle is developed and demonstrated in [7]. An H_∞/H_- robust approach is used to design residual generators. Faults for the flap actuators of the HL-20 reentry vehicle are diagnosed in the autolanding phase of the mission. Performance indices critical for technology adoption such as detection delay, complexity, and computation requirements are used to demonstrate the viability of the proposed method.

A robust FDI approach for the thruster faults in the Mars Express (MEX) orbiter is developed in [8]. A description of the MEX orbiter structure and the uncertainty and disturbances sources is provided. Also, the fault detection, isolation, and recovery (FDIR) mechanism currently implemented on the spacecraft is introduced. The developed diagnoser is based on observer banks for FDI which are structurally decoupled from disturbances and estimated uncertainties.

The contribution of the present work is the design of a decentralized diagnosis architecture for continuous systems. Algorithms for the diagnosis of continuous systems are adapted for the described decentralized architecture.

A similar approach for the diagnosis of systems modeled in a qualitative framework was introduced in [9]. All the work discussed above deals with FDI for either ADS or ACS components along a centralized approach. In the present work, both the ADS and the ACS were considered as subsystems of an ADCS and the decentralized diagnosis framework was applied to resolve possible ambiguities between faults on the components which constitute the subsystems. The fault scenarios to be considered are taken into account in the design phase of the diagnoser.

The paper is structured as follows. Section 2 discusses the background to the diagnosis method used and the developed diagnosis architecture. The application of the decentralized diagnoser to the ADCS of a satellite is the subject of section 3. Section 4 concludes with a summary of the contribution and some perspective.

2 A DECENTRALIZED DIAGNOSIS ARCHITECTURE

This section begins with a summary of the theoretical background of the diagnosis approach used. Then, some notions required for extension of the ARR design approach to decentralized diagnosers are introduced. The diagnoser architecture seen in Fig. 2 is explained next, together with how the diagnoser is designed and implemented.

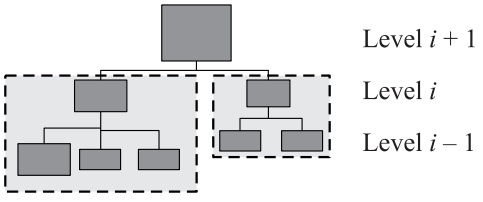

Figure 2 The decentralized diagnoser architecture

2.1 Background of Diagnosis Algorithms

The present approach to diagnosis is based on designing *residual generators* based on *structural redundancies* in the system. Residual generators are derived based on analytical redundancy relations which involve only observed quantities of the system. A residual generator takes as input the values of the observed variables and, in an ideal case, gives a nonzero output only in case the system behavior is inconsistent with the model. Most of this development follows that in [2, 3, 10].

Let the system description consist of a set of n equations involving a set of variables. The set of variables is partitioned into a set Z of n_Z known (or observed) variables and a set X of n_X unknown (or unobserved) variables. Let refer to the vector of known variables as z and the vector of unknown variables as x.

Let consider a *model*, denoted $M(z, x)$ or M for short, to be any set of equations relating the known variables z and the unknown variables x. The equations $m_i(z, x) \subseteq M(z, x)$, $i = 1, \ldots, n$, are assumed to be differential or algebraic equations in z and x.

Let say that a model $M(z, x)$ is consistent with a given trajectory of z, or concisely, consistent with z, if there is a trajectory of x such that the equations $M(z, x)$ are fulfilled.

Definition 1 (ARR for $M(z, x)$ [11]). *Let $M(z, x)$ be a model, then an equation $r(z, \dot{z}, \ddot{z}, \ldots) = 0$ is an* ARR for $M(z, x)$ *if for each z consistent with $M(z, x)$, the equation is fulfilled.*

An ARR can be used to check if the observed variables z are consistent with the model and can be used as the basis of residual generators as defined below.

Definition 2 (Residual Generator for $M(z, x)$ [11]). *A system taking a subset of the variables z as input and generating a scalar signal r as output is a* residual generator for the model $M(z, x)$, *if for all z consistent with $M(z, x)$, it holds that* $\lim_{t\to\infty} r(t) = 0$.

The *structure* of the system can be abstracted as a representation of which variables are involved in the different equations that make up the model of the system. This abstraction allows to study the diagnosability properties independently of the linear or nonlinear nature of the systems. However, it must be kept in mind that results obtained with such a structural representation are a best case scenario. Causality considerations and the presence of algebraic and differential loops determine which structural redundancies can be exploited for the design of residual generators.

Obtaining ARRs for a model $M(z, x)$ involves the elimination of unobserved variables, which can be infered from the bipartite graph. The bipartite graph indeed represents which unobserved variables are involved in the equations modeling the system. It can be shown [12] that ARRs correspond to so-called complete matchings between X and M on the bipartite graph $G(M \cup X \cup Z, \mathcal{A})$,

or equivalently on $G(M \cup X, A)$, where $A \subseteq \mathcal{A}$ and is the set of arcs such that $a(i,j) \in A$ if variable x_i is involved in relation m_j. A complete matching between X and M denoted by $\mathcal{M}(X,M)$, or $\mathcal{M}$ when there is no ambiguity, can be seen as a way to calculate the unobserved variables using the observed quantities. Equivalently, ARRs correspond to minimal structurally overdetermined (MSO) sets, which are the sets of equations of the system with one more equation than unknowns [2]. Unobserved variables can be solved for using the set of equations, and then the one redundant equation can be used to check for consistency. Let adopt an MSO set based ARR design method for the decentralized diagnoser architecture. However, for proving the equivalence of centralized and global diagnosers, the complete matching is used on a bipartite graph view on ARRs.

The structural properties of a system modeled as a set of equations can be analyzed by using the canonical Dulmage–Mendelson (DM). This decomposition of a system model M results in the division of the model into three parts, the structurally overdetermined part represented by M^+, which has more equations than unknowns, the structurally just determined part represented by M^o, and the structurally underdetermined part represented by M^-. The sets defined below formalize the notion of a structurally overdetermined set.

Definition 3 (Structurally Overdetermined (SO) equation set [2]). *A set M of equations is structurally overdetermined if M has more equations than unknowns.*

Definition 4 (Proper Structurally Overdetermined (PSO) equation sets [2]). *An SO set M is a proper structurally overdetermined (PSO) set if $M = M^+$.*

A PSO set is generically a testable subsystem, but it may contain smaller PSO subsets that are also testable subsystems. The minimal PSO sets, namely, the MSO sets, are of special interest since they are at the core of the isolability properties.

Definition 5 (MSO equation sets [2]). *An SO set is an MSO set if no proper subset is an SO set.*

An efficient algorithm to compute all possible MSO sets for a system is developed in [2]. However, the number of possible MSO sets increases exponentially with the redundancy present in the system as measured by the difference between the number of equations and the number of unknown variables. The redundant equation sets which need to be exploited to construct residual generators can be limited to those which correspond to realistic or interesting faults. Krysander *et al.* [3] introduce the concept of test equation supports (TES) which are the sets of equations that express redundancy specific to a set of considered faults. Each TES corresponds to a set of faults which influence the residual generator constructed from the TES. This set of faults is known as the *test support* (TS). The corresponding quantities expressing minimal redundancies are denoted minimal TES (MTES) and minimal TS (MTS). The set of MTES can be seen as a subset of the set of MSOs for the system corresponding to the set of interesting faults.

An algorithm for finding MTES and MTS for a given system structural description and set of interesting faults is developed by modifying the MSO algorithm of [2]. Here, $F(M)$ is used to denote the set of faults that influence any of the equations in M.

Definition 6 (TS [3]). *Given a model M and a set of faults F, a subset of faults $\zeta \subseteq F$ is a test support if there exists a PSO set $M' \subseteq M$ such that $F(M') = \zeta$.*

Definition 7 (MTS [3]). *Given a model, a test support is an MTS if no proper subset is a test support.*

Definition 8 (TES [3]). *An equation set M is a test equation support if M is a PSO set, $F(M) \neq \emptyset$, and for any $M' \supsetneq M$ where M' is a PSO set, it holds that $F(M') \supsetneq F(M)$.*

Definition 9 (MTES [3]). *A TES M is an MTES if there exists no subset of M that is a TES.*

An MSO set or MTES signifies the theoretical presence of a structural redundancy which could be used to develop a consistency check for a part of the system. The corresponding MTS represents the faults which can be detected with this consistency check. It means that the MTS sets characterize the maximum possible fault isolability. Whether a residual generator can be analytically derived depends upon the causality restrictions on the equations in the set and the presence of algebraic and differential loops. We use in our work the residual generator derivation method proposed in [4]. This method relies on developing a computational sequence to successively solve for the unknown variables involved in an equation set. One redundant equation together with the developed com-

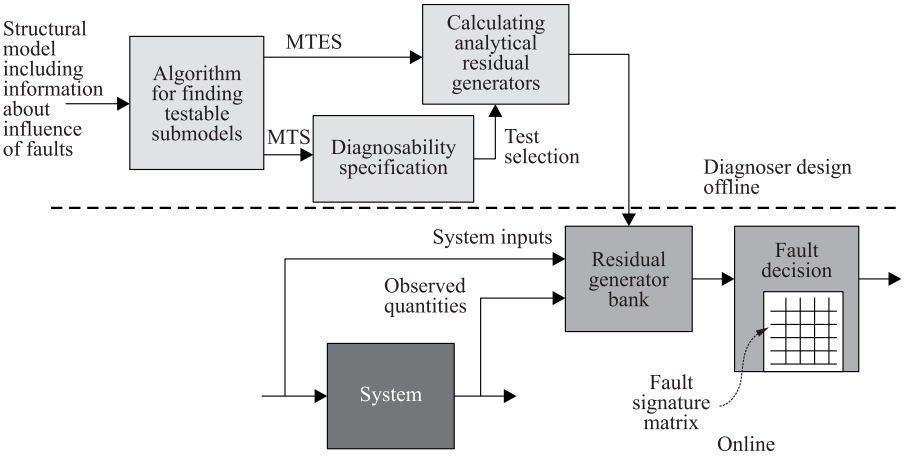

Figure 3 The design and implementation scheme for a centralized global diagnoser

putational sequence constitute a sequential residual generator. An algorithm to develop the computational sequence is provided.

The FDI scheme for a centralized case can be seen in Fig. 3. After offline design, the diagnoser is implemented as a residual generator bank. The fault identification is carried out after fault detection using fault signatures which are vectors composed of the binary residual bank output. Let now introduce the notions needed to decentralize the design and implementation of a diagnoser such as that of Fig. 3.

2.2 Notions for Decentralized Diagnosis

This subsection introduces the notions necessary to devise the proposed decentralized architecture. First, let introduce the decomposition of a global system into a set of subsystems. Then, let define a matching at the local level, at the global level, and at the supervisory level.

Hypothesis 1. *A decomposition of a system M, with associated bipartite graph $G(M \cup X \cup Z, A)$, into several subsystems M_i corresponds to a partition of its equations.*

Formally, let $M = \{M_1, M_2, \ldots, M_n\}$ with $M_i \subseteq M$:

- $M_i \neq \varnothing$;
- $\bigcup M_i = M$; and
- $M_i \cap M_j = \varnothing$ if $i \neq j$.

Definition 10 (variables of a subsystem i). *Considering $G(M \cup X \cup Z, A)$, X_i (Z_i) are defined as the subset of vertices of X (Z) that are adjacent to some vertices in M_i, i. e.,*

$$X_i = \{u \in X : \exists v \in M_i, (u, v) \in A\};$$
$$Z_i = \{u \in Z : \exists v \in M_i, (u, v) \in A\}.$$

The decomposition of the global system into several subsystems leads to n subsystems denoted $M_i(x_i^{\text{local}}, z_i)$, with associated subgraphs $G(M_i \cup X_i^{\text{local}} \cup Z_i, A_i)$, $i = 1, \ldots, n$, where X_i^{local} is defined below.

Definition 11 (local variables). *X_i^{local} is defined as the subset of vertices of X_i that are adjacent only to some vertices in M_i and not to some vertices of M_j, $j \neq i$, i. e.,*

$$X_i^{\text{local}} = \{u \in X_i : \nexists j (j \neq i) v \in M_j, (u, v) \in A\}.$$

Lemma 1. $X_i^{\text{local}} = X_i \Big\backslash \Big(\bigcup_{j=1, j \neq i}^{n} (X_i \cap X_j) \Big)$.

Definition 12 (shared variables). *X^{shared} is defined as the subset of vertices of X that cannot be considered as local variables for any subsystem, i. e.,*

$$X^{\text{shared}} = X \Big\backslash \Big(\bigcup_{i=1}^{n} X_i^{\text{local}} \Big) .$$

Lemma 2. *By definition, $\forall i (1, \ldots, n), X_i^{\text{local}} \cap X^{\text{shared}} = \varnothing$.*

Definition 13 (local complete matching). *A local complete matching $\mathcal{M}_i$ is a complete matching between X_i^{local} and M_i on the graph $G(M_i \cup X_i^{\text{local}}, A_i)$.*

Definition 14 (global complete matching). *A global complete matching $\mathcal{M}$ is a a complete matching between X and M on the graph $G(M \cup X, A)$.*

Definition 15 (hierarchical relation). *Let consider the local subsystem graphs $G(M_i \cup X_i^{\text{local}}, A_i)$, $i = 1, \ldots, n$, and assume that a local complete matching $\mathcal{M}_i$ exists for each of them. Also consider the set of relations that are not matched in any local complete matching $\mathcal{M}_i$. Let r be one of these relations. By construction, r relates to a set of variables, whose unknown variables belong to only one of the X_i^{local} and possibly to X^{shared}. With $\mathcal{M}_i$, it is possible to substitute each variable included in X_i^{local} in r^*, so as to get a new relation r' involving only unknown variables in X^{shared}. The new relation r' is to be transferred to the upper level and is called a hierarchical relation. Relation r is called the source relation of r'. The set of such relations is denoted R'.*

Definition 16 (hierarchical complete matching). *A hierarchical complete matching $\mathcal{M}_h$ is a complete matching between X^{shared} and R' on the graph $G_h(R' \cup X^{\text{shared}}, A')$.*

2.3 The Equivalence of Centralized and Decentralized Diagnosis

When designing decentralized diagnosers for a system, it is interesting to investigate any change in the diagnosability properties due to the decentralization. In particular, it is desirable that the properties such as detectability and isolability of faults are not altered by decentralization. This can be ensured if the set of ARRs derived in the global and decentralized scenarios are identical. This section formalizes this equivalence and provides the basis of the proof.

Proposition 1. *Let M be a system and $\{M_1, M_2, \ldots, M_n\}$ be a decomposition of M, then the set of ARRs that can be derived (in a centralized way) for M is identical to the set of ARRs that can be derived in a decentralized way, i. e., deriving the ARRs for each subsystem M_i and for the hierarchical system composed of the hierarchical relations.*

[*] *Substitute* refers to replacing the variable along the calculation chain defined by the complete matching up to known variables.

This proposition is proved by showing that there exists a global complete matching if and only if there exist local complete matchings and a hierarchical complete matching and that these matchings lead to identical ARRs.

2.3.1 From global to local complete matchings

Proposition 2. *Let $\mathcal{M}$ be a global complete matching on $G(M \cup X, A)$ that leads to a set of ARRs that is nonvoid, then for any decomposition into subsystems $\{M_1, M_2, \ldots, M_n\}$, it is possible to find a set of local complete matchings $\{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_n\}$ and a hierarchical complete matching $\mathcal{M}_h$ that leads to the same nonvoid set of ARRs.*

Proof concept: Suppose that $\mathcal{M}$ is a global complete matching of the system. When the system is decomposed into subsystems, each relation that is matched with a shared variable in $\mathcal{M}$ is now available for being a hierarchical relation. This means that at the hierarchical level, each shared variable can be matched to the hierarchical relation whose source relation is the one it was matched to in $\mathcal{M}$. Consequently, the matchings $\{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_n\}$ lead to the same ARRs.

Figure 4 shows the decomposition of a system into 2 subsystems, and the resulting matchings. The decomposition into two subsystems enables a simple

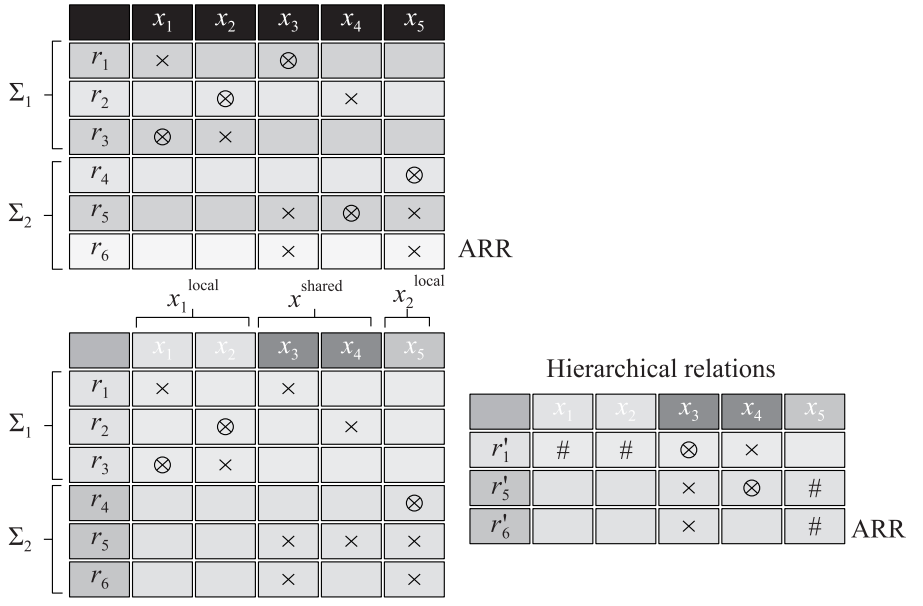

Figure 4 From a global system to a distributed system

illustration of the concept. However, the case of three or more subsystems can be seen in Fig. 5 where the same complete matching could involve variables shared between different subsystems. The global system represented by the adjacency matrix of $G(M \cup X, A)$ has 6 relations $r_1, \ldots, r_6$ and 5 variables $x_1, \ldots, x_5$. The system is decomposed into two subsystems Σ_1 and Σ_2, with $R_1 = \{r_1, r_2, r_3\}$ and $R_2 = \{r_4, r_5, r_6\}$. Thus, let define $X_1^{\text{local}} = \{x_1, x_2\}$, $X_2^{\text{local}} = \{x_5\}$, and $X^{\text{shared}} = \{x_3, x_4\}$. At the top of Fig. 4, there is the global complete matching marked by the relations with circles. At the bottom, the local complete matchings are shown for subsystems Σ_1 and Σ_2 on the left table and the resulting hierarchical relations r'_1, r'_5, and r'_6 on the right side. The # indicate the substituted variables in the hierarchical relations. The hierarchical complete matching is marked by the circles. One can notice that shared variables x_3 and x_4 are matched to r'_1 and r'_5, respectively, by $\mathcal{M}_h$ as they were to the source relation r_1 and r_5 by $\mathcal{M}$.

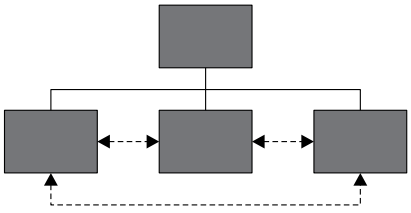

Figure 5 Possible interactions between subsystems at a level

2.3.2 From local to global complete matchings

Proposition 3. *Let $\{M_1, M_2, \ldots, M_n\}$ be the decomposition of a system into a set of n subsystems. Suppose that $(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_n)$ is the set of local complete matchings for each subsystem represented by $G(M_i \cup X_i^{\text{local}}, A_i)$ and $\mathcal{M}_h$ is the hierarchical complete matching on $G_h(R' \cup X^{\text{shared}}, A')$, then it is possible to find a global complete matching $\mathcal{M}$ on $G(M \cup X, A)$ that leads to the same set of ARRs.*

Proof concept: A hierarchical complete matching implies the existence of either a complete matching at the global level, i. e., on $G(M \cup X, A)$, or of a set of substitution paths in either of subsystems which allows the matching of the shared variables by substitution. The set of relations involved in the local and hierarchical matchings can be shown to be exactly the same as that involved in the global complete matching.

2.4 Diagnoser Architecture

The present decentralized diagnosis architecture is composed of a supervisory diagnoser for a system with local diagnosers for the subsystems composing the systems. The aim is to keep the structure hierarchically scalable as shown in Fig. 1. There is no communication between diagnosers at a level. Diagnosers

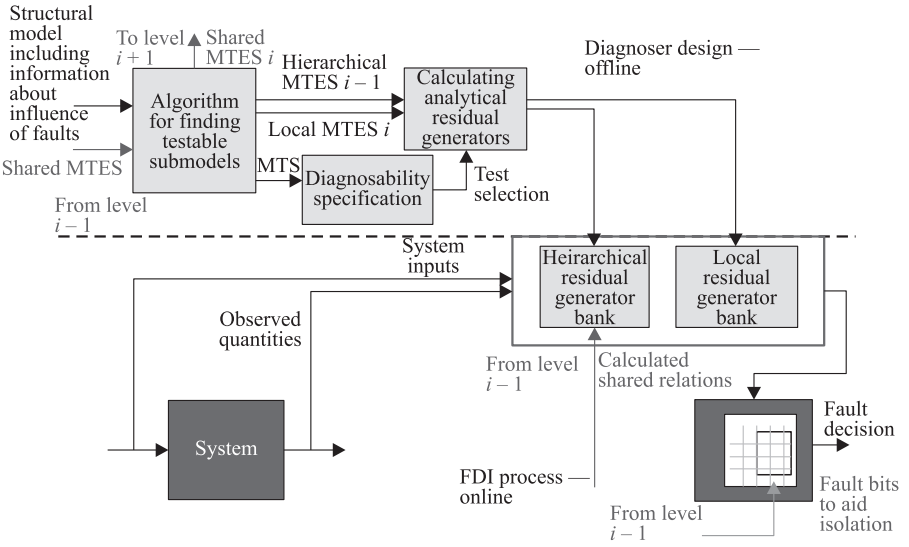

Figure 6 The design and implementation scheme of a decentralized diagnoser for a subsystem at level i

communicate between levels, with their supervisory diagnoser and the local diagnosers below them in the hierarchy. The aim is to expose as little information as possible about the subsystems. This is in keeping with the aim of achieving a decentralized architecture and also fits well into an integrator-subsystem supplier relationship.

The diagnosis process is explained below. The diagnoser design and implementation steps of the diagnosis process are explained with the help of Fig. 6 which can be considered as a decentralized counterpart of Fig. 3. The diagnoser for a subsystem at level i of the diagnoser hierarchy shown in Fig. 2 is considered. The communication required between diagnoser levels is highlighted in the explanation.

Decentralized diagnoser design

The diagnoser design is done *offline* and consists of the steps below. These steps are performed for each subsystem $M_{i,j}$, $j = 1, \ldots, n_i$, at each level $i = 1, \ldots, n_l$, with a nested loop. Here, i signifies the level in the hierarchy and j the enumeration of subsystem at that level.

1. Use the MTES algorithm with the structural model of the subsystem $M_{i,j}$ as input.

Output:

(a) local MTES for the subsystem $M_{i,j}$;
(b) MTS for the subsystem $M_{i,j}$; and
(c) shared MTES for the subsystem $M_{i,j}$.

2. Store shared MTES for supervisory diagnoser design at level $i+1$.
3. Use the MTES algorithm with the shared MTES of subordinate local diagnosers at level $i-1$.
   Output:
   (a) hierarchical MTES for subsystems at level $i-1$.
4. Use MTS and diagnosability specification to decide which residual generators to implement.
5. Derive residual generators for local MTES.
   Output:
   (a) local residual generators for subsystem $M_{i,j}$.
6. Derive residual generators for hierarchical MTES.
   Output:
   (a) hierarchical residual generators for subordinate local diagnosers at level $i-1$.

The practical issues related to implementing such a decentralized diagnoser must be mentioned here. On a satellite, subsystems would be connected to a system bus. For calculation of hierarchical residual generators, values communicated by different subsystems are used. The delay suffered by these communicated values on the system bus would need to be taken into account. However, these issues are out of the scope of this paper.

3 DIAGNOSER DESIGN FOR AN ATTITUDE DETERMINATION AND CONTROL SYSTEM

The task of controlling the attitude of a typical satellite can be decomposed into the attitude determination and attitude control functions. The ADS is composed of sensors which sense the rate and angular position of the satellite. An attitude estimate is achieved using sensor fusion, which is provided as input to the ACS. The ACS is composed of the control algorithm and the actuators which provide the stabilizing and/or control torque to the satellite. The satellite under study is assumed to be a three-axis stabilized satellite in orbit around the Earth. The reaction wheels and magnetorquers are considered as actuators.

Structural Modeling of the Attitude Determination and Control System

The structure of the ADCS is abstracted as a set of constraints relating a set of variables. A discussion of such modeling, only for the ADS, can be found in [13]. The constraints are denoted by C in the following discussion.

The structural model of the system is enriched with information about interesting faults. Following the development in [3], the faults are introduced as signals in the system model equations. The faults are considered on the rate and vector sensors of the ADS and the reaction wheels of the ACS. Such a fault class includes hard, soft, and intermittent faults.

Most of the constraints C are composed of three behavioral relations corresponding to the three axes. The decomposition of the ADCS structure into the ADS and ACS is illustrated in Fig. 7. These structures form the input for the present decentralized architecture and algorithms.

From the set of variables of the system, the sensed quantities form the set of observed variables, with all the rest assumed to be unobserved. Some of the unobserved variables are the internal states of the system whose values are available only through sensors. However, others like state estimations are calculated quantities and can be available for diagnosis. The general procedure for diagnoser design starts with assuming a small set of observed quantities, which is expanded to fulfill diagnosability and isolability specifications if required.

In the present work, a centralized diagnoser is designed globally for the ADCS by deriving MTES and MTS sets. As can be seen in Fig. 8, complete fault isola-

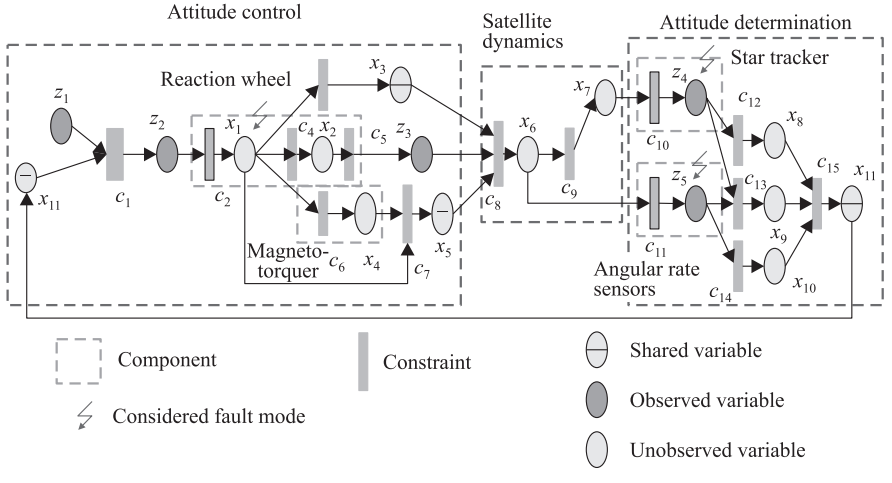

Figure 7 Structural modeling of the ADCS

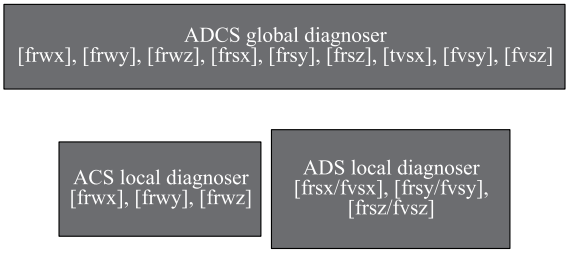

Figure 8 While the centralized global diagnoser ensures complete diagnosability, the local ADS diagnoser is ambiguous between faults in the rate and vector sensors

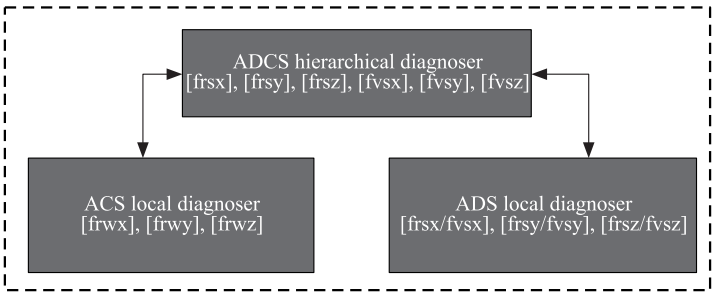

Figure 9 The decentralized architecture ensures diagnosability equivalent to the global diagnoser

bility can be achieved. When the diagnosers working with local models of the ADS and the ACS seperately are designed, the ADS diagnoser cannot disambiguate between faults in the rate and vector sensors. The high number of MSO sets which exist for the ADCS (2448) compared to the number of MTES (9) illustrates the massive computational advantage of only deriving MTES sets which correspond to the set of interesting faults, rather than all possible MSO sets. The proposed decentralized architecture will now be applied to the ADCS by designing the local and supervisory diagnosers (Fig. 9). It will be demonstrated that the isolability capability of such a decentralized diagnoser is equivalent to the global diagnoser above. From the point of view of the local diagnosers, the shared variables X^{shared} are now assumed to be observed. Shared MTES can be derived using the algorithm with this assumption.

It is seen that complete fault isolability is achieved with the presented assumption. Let note, therefore, that it might be possible to isolate (some of) the faults in the ambiguity sets [frsx & fvsx], [frsy & fvsy], [frsz & fvsz] if either some/all of the shared variables are sensed in the ACS, or shared relations exist

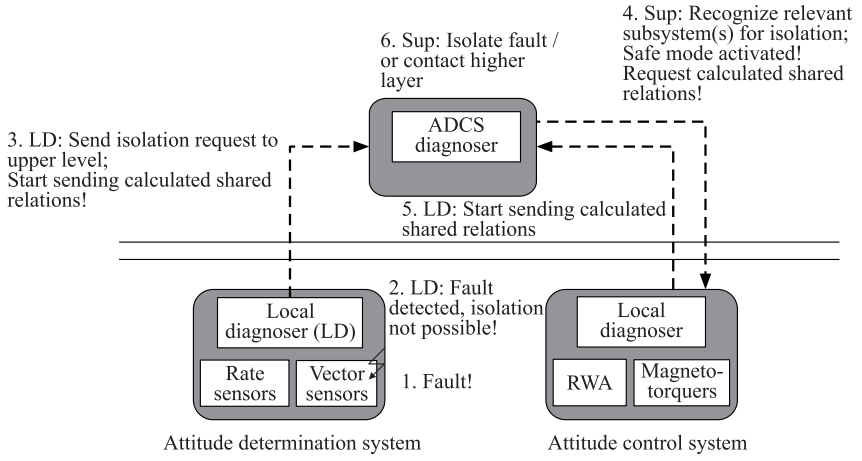

Figure 10 The decentralized diagnosis process applied to an ADCS

in the ACS which allow these variables to be expressed in terms of observable variables.

The functioning of the decentralized ADCS diagnoser is illustrated in Fig. 10. The local diagnosers run their local residual generator banks. It lets say that a fault appears in the vector sensor suite of the ADS. The local diagnoser detects a fault, but cannot isolate it. So, a fault isolation request is sent to the supervisory level diagnoser, and the local diagnoser starts sending the relevant calculated shared relations from the ADS. The fault code could be [frsx/fvsy], for example, indicating the source of the ambiguity. The supervisory layer will put the satellite into safe mode and then request the ACS local diagnoser to start providing the relevant calculated shared relations. The hierarchical residual generators are then evaluated at the supervisory level. The fault is isolated or if higher intervention is required, the ADCS diagnoser contacts the central diagnoser of the satellite.

This is just one possible diagnoser functioning process possible with the architecture. Importantly, this process ensures, first, that only the smallest possible set of residual generators is evaluated during nominal operation and, second, that communication bandwidth is not used under nominal operation for interaction between the local and supervisory diagnosers.

4 CONCLUDING REMARKS

In this paper, an architecture for decentralized FDI has been developed. Local diagnosers working with local models of their subsystems are coordinated by a

supervisor at a higher level to resolve ambiguities arising out of quantities shared between subsystems. Isolation is performed by the supervisor on request. The equivalence of this architecture to a centralized one in terms of diagnosability is shown. This framework is applied to the diagnosis of continuous systems using ARR-based residual generators. Algorithms for the diagnosis of continuous systems are adapted and integrated into the architecture. The structural model of the ADCS of a satellite is used to demonstrate the presented architecture. One of the directions of future work will be investigating diagnosis with different subsystems modeled in different frameworks.

ACKNOWLEDGMENTS

The work described in this paper was carried out as a part of a doctoral project jointly funded by the Center National d'Études Spatiales (CNES), Toulouse, and Thales Alenia Space. The authors wish to thank Raymond Soumagne of CNES and Xavier Olive at TAS for their support and valuable comments.

REFERENCES

1. Kurien, J., and M. D. R-Moreno. 2008. Costs and benefits of model-based diagnosis. *IEEE Aerospace Conference*.
2. Krysander, M., J. Åslund, and M. Nyberg. 2008. An efficient algorithm for finding minimal over-constrained sub-systems for model-based diagnosis. *IEEE Trans. Syst. Man Cyber. Part A: Syst. Humans* 38(1).
3. Krysander, M., J. Åslund, and E. Frisk. 2010. A structural algorithm for finding testable sub-models and multiple fault isolability analysis. *21st Workshop (International) on Principles of Diagnosis (DX-10)*.
4. Svard, C., and M. Nyberg. 2010. Residual generators for fault diagnosis using computation sequences with mixed causality applied to automotive systems. *Trans. Syst. Man Cyber. Part A* 40(6):1310–28.
5. Tudoroiu, N., and K. Khorasani. 2007. Satellite fault diagnosis using a bank of interacting Kalman filters. *IEEE Trans. Aerospace Electronic Systems* 43(4):1334–50.
6. Bialke, B. 1998. High fidelity mathematical modeling of reaction wheel performance. *Adv. Astronautical Sci. Guidance Control* 98:483–96.
7. Falcoz, A., D. Henry, and A. Zolghadri. 2010. Robust fault diagnosis for atmospheric reentry vehicles: A case study. *Trans. Syst. Man Cyber. Part A* 40:886–99.
8. Patton, R. J., F. J. Uppal, S. Simani, and B. Polle. 2010. Robust FDI applied to thruster faults of a satellite system. *Control Eng. Practice* 18(9):1093–109.

9. Console, L., C. Picardi, and D. T. Dupre. 2007. A framework for decentralized qualitative model based diagnosis. *20th Joint Conference (International) on Artificial Intelligence*.
10. Travé-Massuyès, L., T. Escobet, and X. Olive. 2006. Diagnosability analysis based on component-supported analytical redundancy relations. *Trans. Sys. Man Cyber. Part A* 36:1146–60.
11. Armengol, J., A. Bregon, T. Escobet, E. Gelso, M. Krysander, M. Nyberg, X. Olive, B. Pulido, and L. Travé-Massuyès. 2009. Minimal structurally overdetermined sets for residual generation: A comparison of alternative approaches. *7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes Proceedings*. 1480–85.
12. Blanke, M., M. Kinnaert, and J. Lunze. 2006. *Diagnosis and fault-tolerant control*. Springer.
13. Lorentzen, T., M. Blanke, and H. Niemann. 2003. Structural analysis — a case study of the Rømer satellite. *IFAC Safeprocess 2003 Proceedings*. Washington, D.C.