Near-Optimal Decentralized Diagnosis via Structural Analysis

Gustavo Perez-Zuniga, Elodie Chanthery, Louise Travé-Massuyès, Javier Sotomayor

Near Optimal Decentralized Diagnosis via Structural Analysis

G. Pérez-Zuñiga, E. Chanthery, L. Travé-Massuyès, J. Sotomayor-Moriano

***Abstract*—Health monitoring of current complex systems significantly impacts the total cost of the system. Centralized fault diagnosis architectures are sometimes prohibitive for large-scale interconnected systems such as distribution systems, telecommunication networks, water distribution networks or fluid power systems. Confidentiality constraints are also an issue. This paper presents a decentralized fault diagnosis method that only requires the knowledge of local models and limited knowledge of their neighboring subsystems. The method, implemented in the Decentralized Diagnoser Design (D^3) algorithm, is based on structural analysis and can advantageously be applied to high dimensional systems, linear or non-linear. Using the concept of isolation on request, a hierarchy is built according to diagnostic objectives. The resulting diagnoser is based on analytical redundancy relations generated along the hierarchy. Their number is optimized via binary integer linear programming while still guaranteeing maximal diagnosability at each level. D^3 proves of lower time complexity than its centralized equivalent. It is successfully applied to a non-linear combined cycle gas turbine power plant.**

***Index Terms*—Fault Detection and Isolation, Test selection, Decentralized diagnosis, Structural analysis.**

I. Introduction

CURRENT systems such as manufacturing processes, aircraft, power plants, telecommunication networks, and other distribution systems are often so complex that the only way to manage their complexity is to proceed with a decomposition into subsystems. Additionally, the industry is usually not open to sharing complete information about its systems, taking into account that usually multiple companies converge, that is, there are confidentiality constraints between subsystems. Monitoring the health status of such systems, in particular detecting and isolating faults, requires specific methods. This task, which is the focus of this paper, drives repair and maintenance activities that are all together critical in the system's life cycle. Their impact on the total cost of the system is quite significant.

Health monitoring of current complex systems significantly impacts the total cost of the system. Centralized fault diagnosis architectures are sometimes prohibitive for large-scale interconnected systems. To manage complexity, but also confidentiality constraints between subsystems, one solution is to use decentralized approaches.

G. Pérez-Zuñiga and J. Sotomayor-Moriano are with Engineering Department, Pontifical Catholic University of Peru, PUCP (e-mail: gustavo.perez, jsotom@pucp.edu.pe).

E. Chanthery and L. Travé-Massuyès are with LAAS-CNRS and ANITI, Université de Toulouse, CNRS, INSA, Toulouse, France (e-mail: elodie.chanthery, louise@laas.fr).

This project is supported by ANITI through the French "Investing for the Future – PIA3" program under the Grant agreement noANR-19-PI3A-0004.

Fault Detection and Isolation (FDI) can be achieved leveraging the analytical redundancy provided by a system model. The idea is to generate tests whose outputs, termed residuals, reflect inconsistencies between a reference model prediction and the actual system behavior. Such tests are designed using analytical approaches, such as observers [1], parameter estimation [2] or parity equations [3].

We adopt the parity equations approach and work with analytical redundancy relations (ARRs). The system is modeled by a set of ordinary differential equations. ARRs are generated from the model by eliminating unmeasured variables, which can be carried out in a pure analytical framework using elimination theory [4], [5]. However, another way to proceed is to first use structural analysis techniques to select sets of equations that can lead to ARR generation.

Structural analysis makes use of a structural abstraction of the model. Only the links between variables and equations are modeled. Despite the apparent simplicity of the approach, it provides a set of powerful tools, based on graph theory, to analyze the system and deduce relevant information. Moreover, it applies to both linear and non-linear systems [6]–[8].

The originality of this paper is to use structural analysis in a decentralized framework, for decentralized FDI of continuous systems. In this framework, one of the major assumptions is that the system is decomposed into a given set of subsystems. The decomposition results from taking into account functional, geographical or confidentiality constraints. This assumption arises from industrial requests aiming at protecting the data about the subsystems they develop. This is true in many application domains, like space, aeronautics, or power plants, where several suppliers are involved in the design phase.

This paper brings the following contributions:

- An original Decentralized Diagnoser Design method implemented in the D^3 algorithm : using the concept of isolation on request, this method consists in using first the local models of subsystems. If isolability of all the faults cannot be achieved, a higher level is investigated to improve isolability. A hierarchy of levels is thus built, driven by diagnostic goals. The advantages of this design are twofold: on one hand, the architecture naturally matches the system functional decomposition commonly adopted in engineering processes. On the other hand, it results in an anytime FDI algorithm. Indeed, if real time constraints are severe, the algorithm may be stopped at any level while still returning a consistent, although not necessarily precise, diagnosis.
- An advantageous inter linkage of test generation and test selection thanks to a binary integer linear programming

(BILP) model [9] that achieves to minimize the number of tests while maximizing fault isolability at each considered level.
- A thorough theoretical and experimental analysis of the properties of the D^3 algorithm and its application to the non-linear real case study of a gas turbine combined cycle power plant.

The paper is organized as follows: section II presents related work on decentralized diagnosis and structural analysis. In section III, the main concepts of the structural approach are summarized and Fault-Driven Minimal Structurally Overdetermined (FMSO) sets are introduced. The link between ARRs and FMSO sets is outlined and residual generation is explained. Section IV extends the concepts presented in the previous section to the decentralized fault diagnosis framework. Section V presents the algorithm D^3 for the design of the decentralized diagnoser that interlinks test generation and test selection and proves its properties. The computational efficiency of D^3 is proved in practice, and a Gas Turbine Combined-Cycle (GTCC) power plant is then used as a case study in section VI. Finally, section VII concludes the paper.

II. Related Work

Structural analysis is a general framework that can be used to analyze complex, dynamic systems. It ignores the details of parameter values to base the analysis on the structure of the system by means of efficient graph-based tools [10].

It has been used for about two decades by the FDI community to identify redundancy in the system model, independently of their nonlinear nature [7], [11], [12]. If we refer to diagnosis test selection, FlexDx [13], can be mentioned. However, all these works frame the problem in a centralized framework.

It is a long time since decentralized control has been approached in the literature [14]. Decentralized diagnosis has also been the subject of many works for discrete event systems [15]–[17] while decentralized fault diagnosis for continuous systems has only being dealt recently. The authors of [18]–[20] or more recently [21], [22] use decentralized state estimation and observers, and are therefore far from the work presented in this paper.

Like this paper, the authors of [23] address FDI with ARRs and make use of a graph to obtain local diagnosers for each subsystem and a global diagnoser for coordination. Although some structural properties are considered with the graph, they do not make use of structural diagnosis strictly speaking.

Among the papers that make use of structural analysis, [24] proposes two distributed diagnosis approaches. The first one computes tests in a centralized way, then uses a BILP problem to optimize the choice of tests and the decomposition into subsystems to minimize communication between the subsystems. Other works like [25] present methods to decompose process variables into different blocks automatically to minimize communication between sub-systems but without selecting tests accordingly. In [26], like in this paper, the decomposition structure is assumed to be dictated by functional or geographical constraints. Tests are generated locally and unknown variables shared by several subsystems are eliminated when necessary while minimizing communication between subsystems. In [27], the approach of [24] is compared to the one of [26]. The comparison metric is the size of the input matrices for the BILP problems in each approach. As [26] proceeds directly with a distributed approach, the size of the input matrices for the BILP problems is bigger.

The second approach is a distributed algorithm that is designed to minimize communication between the subsystems. It assumes that the global model of the system is not available and proceeds incrementally by augmenting the model subsystem by subsystem until full isolability is achieved. The approach uses Minimal Structurally Overdetermined (MSO) sets that are all computed for each subsystem. In the worse case, all the MSO sets are computed and the augmented system is the whole system.

Our paper proceeds similarly but, as motivated in [26], it does not use MSO sets but FMSO sets instead. These later necessarily involve some faults in their support, and they are much less numerous than MSO sets, which reduces significantly computation time. Although Minimal Test Equation Support (MTES) can be a solution to develop more powerful tests, FMSO sets are better suited to minimize communications between subsystems [26].

The work presented in this paper can be seen as a continuation of [28], driven by the idea of *isolation on request*. The algorithm proposed in this paper builds a diagnostic architecture that allows the diagnoser to adapt to the fault impacting the system at run time. It relies on the property that was demonstrated in [28], i.e. the ARRs generated along a hierarchical decentralized diagnosis architecture are just the same as the ones generated in a centralized architecture when the hierarchy is pushed up. Nevertheless, the proposed algorithm brings a new perspective with respect to [28] by taking into account inter-level restrictions and by taking benefit of BILP optimization to select only the necessary and sufficient tests (ARRs) at each level. On the other hand, although this paper resumes some ideas of [29], it goes much further in the analysis of the properties of the proposed algorithm and in its application to a case study representative of a real process.

III. Concepts For Diagnosis

A. Analytical Redundancy Relations

The model $\Sigma(z, x, \mathtt{f})$, or Σ for short, of a continuous system takes the form of a set of differential or algebraic equations $e_k(z, x), k = 1, \ldots, n_e$, that involve known variables corresponding to sensored inputs and outputs represented by vector z, unknown variables represented by vector x, and variables that represent the possible faults that may act on the system gathered in vector $\mathtt{f}$. Such models can be obtained from physical principles or derived from data using model identification techniques. Identification can be carried out by classical methods [30], but it can also be achieved by methods from artificial intelligence. For a system seen as a black box, methods based on neural networks or regression can be used. Recent graph structure learning methods [31] seem particularly promising for that purpose, as they allow to learn both the structure and the parameters of the model.

Residual generation relies on the analytical redundancy embedded in the model. It results from the number and location of sensors, hence from known variables. In the case of analytical redundancy, variable elimination can be applied to generate relations that involve measured variables only. These relations are used as residual generators.

Definition 1 (Residual Generator for $\Sigma(z, x, \mathtt{f})$): A relation of the form $arr(z', \dot{z}', \ddot{z}', ...) = r$, with input z' a subvector of z and output r, a scalar signal named *residual*, is a residual generator for the model $\Sigma(z, x, \mathtt{f})$ if, for all z consistent with $\Sigma(z, x, \mathtt{f})$, it holds that $\lim_{t \to \infty} r(t) = 0$.

The relation $arr(z', \dot{z}', \ddot{z}', ...) = r$ is qualified as an Analytical Redundancy Relation (ARR). It allows one to check whether the measured variables z are consistent with the system model. ARRs can be designed so that each is sensitive to a different subset of faults qualified as the *fault support* of the ARR. When the residual is non-zero, it means that at least one of the faults of the fault support has occurred.

There are many solutions to design residual generators based on a model or a sub-model with redundancy. Linear algebra based approaches can be used when the model is linear [32]–[34]. [35] proposes residual generation with mixed causality (integral and derivative). However, the complexity of these solutions increases rapidly with problem size and quickly becomes intractable.

The authors of [7] propose the sequential residual generation that is very efficient and simple when the model is of low differential index. The idea is to use graph theory and search for a matching, i.e. a substitution path for eliminating unknown variables in equations of the model. A relation including only known variables with a non-empty fault support can be used as residual generator.

In [11], SARR-Algorithm proposes the successive elimination of unknown variables in the equations that form the structural model with the goal of obtaining the set of minimal Structural ARRs (SARRs). Contrary to other solutions using structural analysis, SARRs account for causality and point at how the equations must be chained to generate an actual ARR.

Some considerations have to be taken into account when trying to generate residuals from redundancy in the model. If the model is perfect, all residue generators are of equivalent quality to detect defects. However, uncertainty and model noise deteriorate performance. It is indeed difficult to estimate the derivatives of known signals in a noisy environment. In [36], the author shows how to reduce the need to estimate derivatives of known signals by putting the residual generator in a state-space form. More recently, [37] proposes to use the Random Forest technique to select residuals properly.

B. Structural Analysis for Diagnosis

Structural analysis can be applied to large-scale systems described by numerous variables, both linear and non-linear, and even for systems under uncertainty for which the analytical model is not precisely known [10], [38].

1) Structural Model: A structural model is composed of a set of constraints, each one associated to a component of the system. The model generally represents nominal behavior, hence the violation of one constraint indicates that the system is faulty and points at the responsible component. The *structural model* of the system $\Sigma(z, x, \mathtt{f})$ can be obtained by abstracting the functional relations of $\Sigma(z, x, \mathtt{f})$. Let us define the sets Z, X, and F as the set of known (or measured) variables, unknown (or unmeasured) variables, and faults that may impact the system, respectively of cardinal n_z, n_x, and n_f.

The structural model can be represented by a matrix qualified as the *incidence matrix,* whose rows are associated to equations and columns to variables. Its elements take the value "1" when the variable is involved in the equation and "0" otherwise. Equivalently, the structural model can be represented by a bipartite graph $G(\Sigma \cup X \cup Z, A)$, where A is a set of edges linking equations of Σ and variables of X and Z. For the purpose of diagnosis and as explained later, this graph can be reduced to $G(\Sigma \cup X, \mathcal{A})$, where $\mathcal{A} \subseteq A$ and $\mathcal{A}$ is a set of edges such that $a(i, j) \in \mathcal{A}$ iff variable x_i is involved in equation e_j.

As an illustrative example, consider a system composed of 6 equations e_1 to e_6 relative to the set of known variables $Z = \{u, y_1, y_2\}$, the set of unknown variables $X = \{x_1, x_2, \dot{x}_1, \dot{x}_2\}$ and the set of faults $F = \{f_1, f_2, f_3\}$.

(e_1) $\dot{x}_1 = 0.7x_1 + 0.2x_2$ (e_2) $\dot{x}_2 = 0.5x_2 + u + f_3$
(e_3) $y_1 = x_1 + f_1$ (e_4) $y_2 = x_2 + f_2$
(e_5) $\dot{x}_1 = \frac{dx_1}{dt}$ (e_6) $\dot{x}_2 = \frac{dx_2}{dt}$

Fig. 1 shows the bipartite graph relating to the illustrative example.

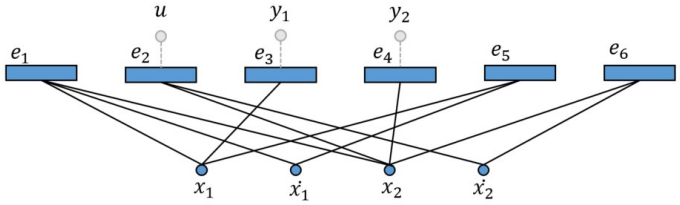

Fig. 1: The bipartite graphs $G(\Sigma \cup X \cup Z, A)$ and $G(\Sigma \cup X, A)$ obtained by removing gray edges and variables for the illustrative example.

2) Diagnosis via Structural Redundancy: When used for FDI purposes, the outputs of structural analysis are in the form of subsets of equations endowed with redundancy. These can be turned into diagnosis tests, i.e. ARRs or parity relations, which are designed off-line [7]. Diagnosis tests are then checked against observations on-line.

Redundancy in a system of the form $\Sigma(z, x, \mathtt{f})$ can be brought to light by the well-known Dulmage-Mendelsohn (DM) canonical decomposition [7], [39], [40]. It partitions the system into three subsystems:

- Σ^+ has more equations than unknown variables and is named the *structurally overdetermined* (SO) subsystem,
- Σ^0 is the *structurally just determined* subsystem,
- Σ^- has more unknown variables than equations and is named the *structurally underdetermined* subsystem.

In the following, the systems that are considered are well-defined systems, which means that $\Sigma = \Sigma^+$. This is the case

of the illustrative example.

Definition 2 (Structural redundancy): The structural redundancy $\rho_{\Sigma'}$ of a set of equations $\Sigma' \subseteq \Sigma$ is defined as the difference between the number of equations and the number of unknown variables.

If a set of equations is structurally redundant ($\rho_{\Sigma'} > 0$), it means that ARRs, i.e. residual generators, can be built using the equations in this set. Thus, structural analysis provides the means to find subsets of equations with structural redundancy. Once the subsets found, one of the solutions proposed in the previous section can be used to obtain residual generators.

If a set of equation Σ is such that $\Sigma = \Sigma^+$ and no proper subset of Σ is overdetermined, the concept of *minimally structurally overdetermined* (MSO) [12] is used. In the illustrative system, the set $\Sigma_1 = \{e_1, e_2, e_3, e_5, e_6\}$ is an MSO set and its structural redundancy is $\rho_{\Sigma_1} = 1$. An MSO set gives naturally rise to one residual generator because it has structural redundancy 1.

However, only MSO sets impacted by faults are interesting for diagnosis. This is why we refer to the concept of *fault support*.

Definition 3 (Fault support): The set $F_{\Sigma'}$ of faults that are involved in the equations of a subset $\Sigma' \subseteq \Sigma$ is defined as the fault support of Σ'.

In the following, we distinguish MSO sets of $\Sigma(z, x, \mathtt{f})$ whose fault support is empty, qualified as *Clear Minimal Structurally Overdetermined* (CMSO) sets, and MSO sets whose fault support is not empty, qualified as *Fault-Driven Minimal Structurally Overdetermined* (FMSO) sets [29].

In the illustrative example, the MSO set $\Sigma_1 = \{e_1, e_2, e_3, e_5, e_6\}$ is an FMSO set and its fault support is $F_{\Sigma_1} = \{f_1, f_3\}$. There are no CMSO sets.

An FMSO set φ identifies a just overdeterminated subset of $|\varphi|$ equations of the model, among which one is redundant. This means that all the unknown variables can be determined using $|\varphi| - 1$ equations, and that an ARR can be generated by substituting in the $|\varphi|^{th}$ equation. We illustrate the process of generating an ARR from an FMSO set below [6].

For the illustrative example, let us consider the FMSO set $\Sigma_1 = \{e_1, e_2, e_3, e_5, e_6\}$. Every unknown variable can be associated to one equation that is interpreted as a mechanism to determine it. For instance:

- e_3 determines $x_1 : x_1 = y_1$
- e_5 determines $\dot{x}_1 : \dot{x}_1 = \frac{dx_1}{dt} \Rightarrow \dot{x}_1 = \dot{y}_1$
- e_1 determines $x_2 : x_2 = \frac{1}{0.2}\dot{x}_1 - \frac{0.7}{0.2}x_1 \Rightarrow x_2 = \frac{1}{0.2}\dot{y}_1 - \frac{0.7}{0.2}y_1$
- e_6 determines $\dot{x}_2 : \dot{x}_2 = \frac{dx_2}{dt} \Rightarrow \dot{x}_2 = \frac{1}{0.2}\ddot{y}_1 - \frac{0.7}{0.2}\dot{y}_1$

Finally, the ARR is obtained with the redundant equation e_2 as follows:

$$\dot{x}_2 = 0.5x_2 + u \Rightarrow \frac{1}{0.2}\ddot{y}_1 - \frac{0.7}{0.2}\dot{y}_1 = \frac{0.5}{0.2}\dot{y}_1 - \frac{0.5\times 0.7}{0.2}y_1 + u$$
$$\Rightarrow 5\ddot{y}_1 - 6\dot{y}_1 + 1.75y_1 - u = 0$$

FMSO sets are also useful in defining the concepts of *detectable fault* and *isolable fault*.

Definition 4 (Detectable fault): A fault $f \in F$ is detectable in the system $\Sigma(z, x, \mathtt{f})$ if there exists an FMSO set $\varphi \in \Phi$ such that $f \in F_\varphi$.

Definition 5 (Isolable fault): Given two detectable faults f and f' of F, $f \neq f'$, f is isolable from f' if there exists an FMSO set $\varphi \in \Phi$ such that $f \in F_\varphi$ and $f' \notin F_\varphi$.

In the illustrative example, the faults f_1, f_2, and f_3 are detectable: $f_1 \in F_{\Sigma_1}$ and $f_3 \in F_{\Sigma_1}$, where $\Sigma_1 = \{e_1, e_2, e_3, e_5, e_6\}$ has been shown to be an FMSO set, $f_2 \in F_{\Sigma_2}$, where $\Sigma_2 = \{e_2, e_4, e_6\}$ is also an FMSO set. f_1 and f_3 are isolable from f_2 because $f_2 \notin F_{\Sigma_1}$ while $f_1 \in F_{\Sigma_1}$ and $f_3 \in F_{\Sigma_1}$.

Definition 6 (Ambiguity set and Isolability degree): The ambiguity set $\mathcal{A}(\Sigma)$ of a system Σ is the set of sets of not isolable faults. The system isolability degree $\mathcal{I}(\Sigma)$ is defined as the cardinal of its ambiguity set, i.e. $\mathcal{I}(\Sigma) = |\mathcal{A}(\Sigma)|$.

According to Definition 6, the system isolability degree $\mathcal{I}(\Sigma)$ is at most equal to n_f, which is met when all the faults are isolable. The system isolability degree $\mathcal{I}(\Sigma)$ defines an upper bound for the isolability degree that can be obtained with a diagnosis algorithm.

IV. Decentralized Diagnosis

A. Fault Diagnosis in a Decentralized Framework

The aim of this work is to propose a decentralized diagnoser design method. One of the assumptions is that the system is decomposable into sub-systems determined by functional, geographical or confidential constraints guided by engineering requirements. Each subsystem may represent the implementation of a primary function. The decomposition itself is not in the scope of this work but this can be a problem in its own right [41], [42].

Fig. 2 illustrates an example of a decentralized architecture with 3 sub-systems at the lowest level and 2 levels in the supervisory hierarchy. In the decomposition, each subsystem has its own local diagnoser. If isolability of all faults cannot be achieved at some level, higher level sub-systems can be considered with their own diagnosers. This can be applied recursively, so that this type of architecture is fully scalable.

More generally, the decentralized structure of $\Sigma(z, x, \mathtt{f})$ is defined as a hierarchical organization over several levels as illustrated in Fig. 3. $\Sigma_{1,i}$, $i = 1, \ldots, n_1$ represent sub-systems of the lowest level. $\Pi_{j,i}$ sets are equation sets that are only available at level j, for confidentiality reasons or because they are too far away or costly to access. $\Sigma_{j,i}$ represents the ith subsystem at level j, with $i = 1, ..., n_j$ and $j = 1, ..., m$, obtained by lumping some of the equations of some subsystems of the level just beneath and the equations of $\Pi_{j,i}$. Note that, by construction, $n_j < n_{j-1}, \forall j = 2, ..., m$.

In extension, $\Sigma_{j,i}$, $j = 1, ..., m$ and $i = 1, ..., n_j$, is denoted $\Sigma_{j,i}(z_{j,i}, x_{j,i}, \mathtt{f}_{j,i})$, where $z_{j,i}$, $x_{j,i}$ and $\mathtt{f}_{j,i}$ are the vectors of known variables, unknown variables and faults, respectively. The sets of unknown variables, known variables, and faults are denoted $X_{j,i}$, $Z_{j,i}$, and $F_{j,i}$, respectively. These are the subsets of variables of X, Z, and F involved in the subsystem $\Sigma_{j,i}$.

Parameter n_m of the highest level m is required to be equal to 1, i.e. level m is restricted to one single subsystem. Parameter n_1 is given by the imposed decomposition. The hierarchy, i.e. the number of levels m and the number of subsystems $n_j, j = 1, \ldots, m$ in each level $j > 1$, are constrained by

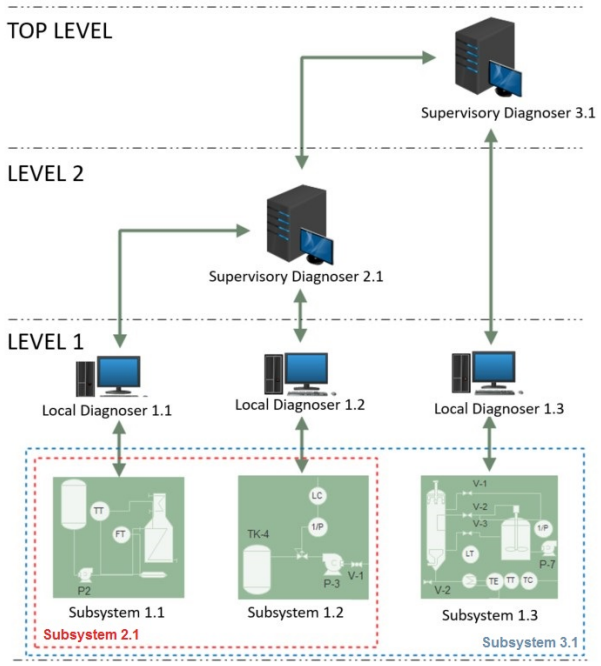

Fig. 2: A decentralized diagnosis architecture with 3 subsystems.

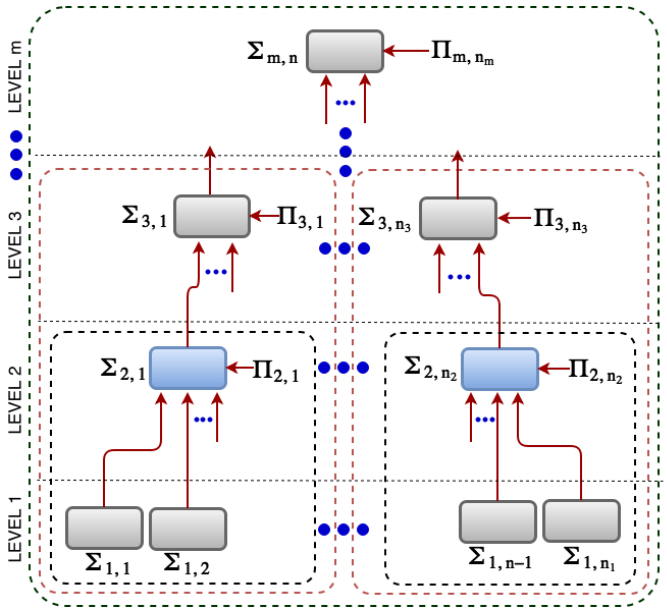

Fig. 3: General decentralized architecture.

allowed and forbidden inter-level communication as formally defined below.

Definition 7 (Inter-level communication): The inter-level communication is given by a set of $m-1$ bipartite graphs $S^j_{j-1}(\mathbb{N}^j_{j-1}, \mathbb{L}^j_{j-1})$, $j=2,\ldots,m$, such that $\mathbb{N}^j_{j-1} = \mathbb{N}_{j-1} \cup \mathbb{N}_j$, where:

- $\mathbb{N}_j = \{n_{j,i}, i=1,\ldots,n_j\}$ is a set of nodes corresponding to the subsystems $\Sigma_{j,i}, i=1,\ldots,n_j$, of level j, and $\mathbb{N}_{j-1}$ is defined similarly.
- $\mathbb{L}^j_{j-1} = \{l_{\nu,\xi}, \nu=1,\ldots,n_{j-1}, \xi=1,\ldots,n_j\}$ is a set of edges such that the edge $l_{\nu,\xi}$ between node $n_\nu \in \mathbb{N}_{j-1}$ and $n_\xi \in \mathbb{N}_j$ exists if communication is possible between subsystem $\Sigma_{j-1,\nu}$ at level $j-1$ and subsystem $\Sigma_{j,\xi}$ at level j.

Definition 8 (Local variable): A variable of a subsystem $\Sigma_{j,i}$ is said to be local if it is only involved in the equations of $\Sigma_{j,i}$. The set of local variables of $\Sigma_{j,i}$ is denoted by $X^l_{j,i}$.

$$X^l_{j,i} = X_{j,i} \backslash \Big(\bigcup_{\substack{k=1,\ldots,m \\ l=1,\ldots,n_k}} (X_{j,i} \cap X_{k,l}) \Big) \text{ with } (k,l) \neq (j,i) \quad (1)$$

Let us decompose the illustrative system in two subsystems $\Sigma_{1,1} = \{e_1, e_3, e_5\}$ and $\Sigma_{1,2} = \{e_2, e_4, e_6\}$, then local variables are given by $X^l_{1,1} = \{x_1, \dot{x}_1\}$ and $X^l_{1,2} = \{\dot{x}_2\}$.

Definition 9 (Shared Variable): A variable of $\Sigma_{j,i}$ is said to be shared if it is involved in the equations of $\Sigma_{j,i}$ and at least in the equations of another subsystem $\Sigma_{j',i'}$ with $(j,i) \neq (j',i')$. The set of shared variables of subsystem $\Sigma_{j,i}$ is denoted $X^s_{j,i}$ and

$$X^s_{j,i} = X_{j,i} \backslash X^l_{j,i} \quad (2)$$

X^s represents the set of shared variables of Σ.

In the illustrative system, there is one shared variable only so that $X^s_{1,1} = X^s_{1,2} = X^s = \{x_2\}$.

B. FMSO Sets for Decentralized Diagnosis

The concepts of FMSO sets and CMSO are extended to the decentralized context.

Definition 10 (Global FMSO set): An FMSO set φ of $\Sigma(z,x,\mathtt{f})$ is called a global FMSO set. The set of global FMSO sets is denoted by Φ.

Considering the illustrative system, $\Phi = \{\varphi_1, \varphi_2, \varphi_3\}$, where $\varphi_1 = \{e_2, e_4, e_6\}$, $\varphi_2 = \{e_1, e_3, e_4, e_5\}$ and $\varphi_3 = \{e_1, e_2, e_3, e_5, e_6\}$.

Definition 11 (Local FMSO set): φ is a local FMSO set of $\Sigma_{j,i}(z_{j,i}, x_{j,i}, \mathtt{f}_{j,i})$ if φ is an FMSO set of $\Sigma(z,x,\mathtt{f})$ and if its unknown variables can all be eliminated locally, i.e. with equations of $\Sigma_{j,i}$. The set of local FMSO sets of $\Sigma_{j,i}$ is denoted by $\Phi^l_{j,i}$. The set of all local FMSO sets is denoted by $\Phi^l = \bigcup_{\substack{j=1,\ldots,m \\ i=1,\ldots,n_j}} \Phi^l_{j,i}$.

Two important things have to be noticed: 1) a local FMSO set may involve shared variables if they can be eliminated locally; 2) a local FMSO set for any subsystem $\Sigma_{j,i}$ is also an FMSO set of Σ, hence, a global FMSO set.

In the illustrative system, $\Phi^l_{1,1} = \{\emptyset\}$, $\Phi^l_{1,2} = \{\varphi_1 = \{e_2, e_4, e_6\}\}$. As mentioned above, note that the local FMSO set φ_1 is also a global FMSO set.

If shared variables cannot be eliminated locally, they need equations of other subsystems to be substituted. This leads to the definition of *shared FMSO sets* for a subsystem $\Sigma_{j,i}$.

Definition 12 (Shared FMSO set): If φ is an FMSO set of $\tilde{\Sigma}_{j,i}(\tilde{z}_{j,i}, \tilde{x}_{j,i}, \tilde{\mathtt{f}}_{j,i})$, where $\tilde{z}_{j,i}$ is the vector of variables in $\tilde{Z}_{j,i} = Z_{j,i} \cup X^s_{j,i}$, $\tilde{x}_{j,i}$ is the vector of variables in $\tilde{X}_{j,i} = X^l_{j,i}$, and $\tilde{\mathtt{f}}_{j,i} = \mathtt{f}_{j,i}$, then φ is a shared FMSO set of subsystem $\Sigma_{j,i}(z_{j,i}, x_{j,i}, \mathtt{f}_{j,i})$. The set of shared FMSO sets for $\Sigma_{j,i}$ is

denoted by $\Phi^s_{j,i}$. The set of all shared FMSO sets is denoted by $\Phi^s = \bigcup_{\substack{j=1,\dots,m \\ i=1,\dots,n_j}} \Phi^s_{j,i}$.

From the above definition, a shared FMSO set φ for subsystem $\Sigma_{j,i}(z_{j,i}, x_{j,i}, \mathtt{f}_{j,i})$ is such that $\varphi \subseteq \Sigma_{j,i}$, $X_\varphi \subseteq X^l_{j,i}$, $Z_\varphi \cap X^s_{j,i} \neq \emptyset$, and $Z_\varphi \subseteq (Z_{j,i} \cup X^s_{j,i})$.

For subsystem $\Sigma_{1,1}$ of system (III-B1), $\tilde{Z}_{1,1} = Z_{1,1} \cup X^s_{1,1} = \{y_1, x_2\}$ and $\Phi^s_{1,1} = \{\varphi_4 = \{e_1, e_3, e_5\}\}$. For subsystem $\Sigma_{1,2}$, $\tilde{Z}_{1,2} = Z_{1,2} \cup X^s_{1,2} = \{y_2, u, x_2\}$ and $\Phi^s_{1,2} = \{\varphi_5 = \{e_4\}, \varphi_6 = \{e_2, e_6\}\}$.

CMSO sets can also be extended in a decentralized framework as *local CMSO sets* and *shared CMSO sets,* as done for FMSO sets. The set of local (shared) CMSO sets of $\Sigma_{j,i}$ is denoted by $\Psi^l_{j,i}$ ($\Psi^s_{j,i}$) and the set of all local (shared) CMSO sets by Ψ^l (Ψ^s) [29].

V. Decentralized Diagnosis Design Algorithm

This section presents the Decentralized Diagnoser Design algorithm D^3 for designing a decentralized diagnoser optimizing the number of FMSO sets while providing fault isolability guarantees.

A. Algorithm for Decentralized Diagnoser Design

The D^3 algorithm runs offline. It delivers a near-optimal distribution of ARRs on a hierarchy of subsystems. The user chooses a target isolability degree as a percentage of the number of faults, i.e. $\%n_f$. Given the system isolability degree $\mathcal{I}(\Sigma)$, which may be known by the user or not, the maximal isolability degree $\mathcal{I}^*(\Sigma)$ is defined as:

$$\mathcal{I}^*(\Sigma) = \min(\mathcal{I}(\Sigma), \mathcal{I}^t(\Sigma)) \tag{3}$$

The D^3 Algorithm.

1: $E_{0,1} = \emptyset$, $\mathcal{I}_0(\Sigma) = 0$, $j = 1$, $i = 1$ ▷ Init part
2: **while** $(j \leq m)$ && $(\mathcal{I}_{j-1}(\Sigma) < \mathcal{I}^*(\Sigma))$ **do** ▷ Main loop
3: **while** $i < n_j$ **do**
4: $\Upsilon_{j,i} \leftarrow$ `ComputeUpsilon`(j, i)
5: $\Sigma_{j,i} \leftarrow \Pi_{j,i} \cup \Upsilon_{j,i}$
6: $\Phi^l_{j,i} \leftarrow$ `ComputeLocFMSOOf`$(\Sigma_{j,i})$
7: $\Phi^{l\,*}_{j,i} \leftarrow$ `SelectOptLocFMSO`$(\Phi^l_{j,i})$
8: $ARR_{j,i} \leftarrow$ `ComputeARR`$(\Phi^{l\,*}_{j,i})$
9: $i = i + 1$
10: **end while**
11: $\mathcal{I}_j(\Sigma) \leftarrow$ `ComputeID`(j)
12: $j = j + 1, i = 1$
13: **end while**

In the initialization part of the D^3 algorithm (line 1), $E_{0,1}$ and $\mathcal{I}_0(\Sigma)$ are initial conditions for an artificial level 0 that allows us to apply the recursive operations for the following levels. Indexes j and i are initialized to 1. j represents the level in the decentralized architecture, while i numbers the subsystems in a given level. Two loops consider subsystems from $i = 1$ to $i = n_j$ at each level j from 1 up to possibly m.

The inter-level communication is taken into account at each level $j = 2, \dots, m$ by $\Upsilon_{j,i}$, where:

$$\Upsilon_{j,i} = \bigcup_{\nu / l_{\nu,i} \in \mathbb{L}^j_{j-1}} E_{j-1,\nu} \tag{4}$$

$$E_{j-1,\nu} = \{e \in \Phi^s_{j-1,\nu} \cup \Psi^s_{j-1,\nu}\} \tag{5}$$

In other words, for levels $j = 2, \dots, m$, $\Upsilon_{j,i}$ contains the equations involved in shared FMSO and CMSO of all subsystems of level $j-1$ whose connection is allowed with subsystem $\Sigma_{j,i}$ and that share variables. This is performed by function `ComputeUpsilon(j,i)` on line 4. Note that $\Upsilon_{1,i} = \emptyset$. The subsystem $\Sigma_{j,i}$ is then formed (line 5) by unioning $\Upsilon_{j,i}$ and $\Pi_{j,i}$ that brings additional information only available at level j.

Local[1] FMSO sets for $\Sigma_{j,i}$ are then computed with function `LocalFMSOOf()` (line 6). Function `SelectOptLocFMSO()` selects the best local FMSO sets among the local FMSO sets by solving a BILP problem as described in Subsection V-B (line 7). The set $ARR_{j,i}$ (line 8) represents the set of ARRs derived from the selected FMSO sets $\Phi^{l\,*}_{i,j}$ and is computed with function `ComputeARR()`. Finally, function `ComputeID()` computes the isolability degree $\mathcal{I}_j(\Sigma)$ at the current level (line 11). The condition of the loop checks whether if the isolability degree is still lower than the targeted isolability degree $\mathcal{I}^*(\Sigma)$ (line 2). If this is the case, a higher level of the hierarchy is explored. D^3 stops either when the targeted isolability degree $\mathcal{I}^*(\Sigma)$ is achieved or when the maximal number of levels m has been reached.

B. Optimal Selection of Local FSMO sets

For each subsystem $\Sigma_{j,i}$, D^3 solves a BILP problem [43] during the `ComputeLocFMSOOf()` function. This function selects the best local FMSO sets among all the computed local FMSO sets $\Phi^l_{j,i}$ of $\Sigma_{j,i}$. The selection criterion aims at maximizing the isolability degree while minimizing the number of chosen local FMSO sets.

Let $\mathbb{z}_{\varphi_i}$ be a boolean variable that is equal to 1 if the local FSMO set φ_i is selected, 0 otherwise. Let $\mathbb{f}_{kl\varphi_i}$ be a boolean variable equal to 1 if the fault f_k is isolable from the fault f_l by using the local FMSO set φ_i. Let e_{kl} be a variable that indicates that the fault f_k is isolable from the fault f_l when it is equal to 1 and the inverse when it is equal to 0. The e_{kl} variables are introduced in the objective function and the problem becomes a bi-objective optimization problem weighted by a parameter α. α represents a weighting between the maximization of the isolability degree and the minimization of the number of chosen local FMSO sets. The BILP model is as follows:

$$max(\ \alpha \sum_{(f_k,f_l) \in F_{j,i}} e_{kl} - (\alpha - 1) \sum_{\varphi_i \in \Phi^l_{j,i}} \mathbb{z}_{\varphi_i}) \tag{6}$$

$$\text{Subject to: } \sum_{\varphi_i \in \Phi^l_{j,i}} \mathbb{f}_{kl\varphi_i} \mathbb{z}_{\varphi_i} \geq e_{kl} \tag{7}$$

$$\mathbb{z}_{\varphi_i} \in \{0,1\} \text{ for } \varphi_i \in \Phi^l_{j,i}, \quad (k,l) \in F_{j,i}, \quad k \leq l \tag{8}$$

$$\alpha \in [0,1], \quad 0 \leq e_{kl} \leq 1 \tag{9}$$

[1]Note that "local" refers to a subsystem according to Definition 11, however this subsystem can be created at any level of the hierarchy.

Equation (6) is the objective function of the BILP problem. The constraint (7) is used to guarantee the maximum possible isolation for subsystem $\Sigma_{j,i}$.

Let us notice that efficiency dictates to limit the number of binary variables [2], hence the e_{kl}'s are coded as continuous variables in the program. In practice, the fact that the left-hand side of inequality (7) is binary forces the value of e_{kl} to be binary. Indeed, when equal to 0, e_{kl} is forced to 0 and when equal to 1, e_{kl} is forced to 1 by the maximization objective (6).

C. Algorithm Properties

This section analyzes the properties and complexity of the D^3 algorithm.

1) Prerequisites:

- Considered systems are well-defined systems, i.e., $\Sigma = \Sigma^+$.
- Proposition 3 of [29]: in a distributed architecture for which the system Σ is decomposed into a set of subsystems that can exchange information, the set of global FMSO sets is given by the union of local FMSO sets and compound FMSO sets, where compound FMSO sets are FMSO sets that include shared FMSO sets. Note that a distributed architecture can be viewed as a two levels hierarchy, in which the subsystems define level 1 and the information exchange defines level 2.

2) Properties:

Theorem 1: The D^3 algorithm guarantees maximal isolability degree if: (P) the inter-level communication is such that there exists a link $l_{\nu,\xi}$ from any subsystem $\Sigma_{j-1,\nu}, \nu \in \{1,\ldots,n_{j-1}\}$, with shared variables of level $j-1$ to some subsystem $\Sigma_{j,\xi}, \xi \in \{1,\ldots,n_j\}$, of level j in $\mathbb{L}^j_{j-1}$, $j = 1,\ldots,m$.

Proof 1: If property (P) is satisfied, any shared FMSO set φ of a given subsystem $\Sigma_{j-1,\nu}, \nu \in \{1,\ldots,n_{j-1}\}$, of level $j-1$ is part of some subsystem $\Sigma_{j,\xi}, \xi \in \{1,\ldots,n_j\}$, of level j. There are two cases:

- Case 1: $\Sigma_{j,\xi}$ has inherited from shared FMSO sets of other sub-systems $\Sigma_{j,\beta\neq\xi}$ so that φ is involved in a local FMSO set for $\Sigma_{j,\xi}$.
- Case 2: the condition of the previous item is not true and φ remains as a shared FMSO set for $\Sigma_{j,\xi}$ and by property (P), it is part of some sub-system of level $j+1$. Now, property (P) implies that there exists a level $\ell \leq m$ in the hierarchy for which the subsystems $\Sigma_{m,\nu}, \nu \in \{1,\ldots,n_\ell\}$ do not share variables and do not hence involve shared FMSO sets. This implies that a local FMSO set that includes φ is generated at some level $k \in \{j+1,\ldots,\ell\}$ of the hierarchy (this is guaranteed by $\Sigma = \Sigma^+$).

Case 1 and Case 2 imply that all shared FMSO sets are retrieved to generate local FMSO sets at some level $j \in \{2,\ldots,\ell\}$. In the decentralized hierarchy used by D^3, local FMSO sets generated at a given level j can be viewed as compound FMSO sets (as defined in Proposition 3 of [29]) with respect to shared FMSO sets of some lower level. Recursive application of Proposition 3 of [29] to every two subsequent levels of the D^3 hierarchy hence implies that the set of global FMSO sets is given by the set of local FMSO sets generated over the ℓ first levels of the D^3 hierarchy. Consequently, all global FMSO sets can be generated by D^3, which guarantees maximal isolability $\mathcal{I}^*(\Sigma)$ when $\mathcal{I}^*(\Sigma) = \mathcal{I}(\Sigma) \leq \mathcal{I}^t(\Sigma)$. If $\mathcal{I}^*(\Sigma) = \mathcal{I}^t(\Sigma) < \mathcal{I}(\Sigma)$, D^3 may stop before developing levels up to ℓ, guarantying however maximal isolability.■

Lemma 1: If the inter-level communication satisfies property (P) and $\mathcal{I}^t(\Sigma) \geq \mathcal{I}(\Sigma)$, the set of local FMSO sets generated by D^3 over all levels of the hierarchy is the set of global FMSO sets of system Σ, i.e., $\Phi^l = \Phi$.

Proof 2: This lemma is a direct consequence of Theorem 1.■

Theorem 2: The D^3 algorithm computes a near-optimal set of analytical residual generators that guarantee maximal fault isolability if the inter-level communication satisfies property (P).

Proof 3: By Theorem 1 and Lemma 1, the set of all local FMSO sets generated by D^3 over the levels 1 to ℓ of the hierarchy guarantee maximal fault isolability. Function `SelectOptimalLocalFMSO()` of D^3 makes a selection among local FMSO sets while preserving achievable isolability at the level of each subsystem. Although the solution of the BILP problem is proved optimal and so is the selected set of local FMSO sets for each subsystem, the union of these sets may overestimate the optimal solution for the whole system Σ, hence the D^3 algorithm is *near-optimal*.■

Theorem 3: The D^3 algorithm has lower time complexity than its centralized equivalent if and only if the highest level ℓ of the developed hierarchy has more than one subsystem, i.e., $n_\ell > 1$.

Proof 4: Given a system Σ with n_e equations and fixed order of structural redundancy ρ, the computation of FMSO sets has been proven to have order of $n_e^{\rho+1.5}$ time complexity and the number of FMSO sets to be bounded by $\rho.n_e^{\rho-1}$ [44]. BILP is known to be polynomial when the constraint matrix is totally unimodular. This is the case in the BILP formulation of FMSO set selection. The time complexity of an FMSO set selection step based on BILP is thus negligible compared to that of FMSO sets computation.

Let ρ^* denote the highest structural redundancy over all subsystems of the developed hierarchy:

$$\rho^* = \max_{\substack{j=1,\ldots,\ell \\ i=1,\ldots,n_j}} \rho_{\Sigma_{j,i}} \tag{10}$$

The computation of local FMSO sets as thus order of $n_e^{\rho^*+1.5}$ time complexity and the number of local FMSO sets is bounded by $\rho^*.n_e^{\rho^*-1}$. Moreover, as shown above, the BILP algorithm time complexity is negligible compared to that of local FMSO sets computation. Thus, each while loop of the D^3 algorithm has order of $n_e^{\rho^*+1.5}$ time complexity. The loop is calculated $m.n_1$ times at worst, hence, D^3 has order of $n_e^{\rho^*+1.5}$ time complexity.

[2]Note that in recent solvers, it may not be more efficient.

Let us now prove by contradiction that if $n_\ell > 1$, then any subsystem of the hierarchy has lower structural redundancy than the system Σ as a whole. Let Σ_1 be a subsystem of structural redundancy ρ_{Σ_1}. Let us assume that recombining Σ_1 with a subsystem Σ_2 results in $\Sigma = \Sigma_1 \cup \Sigma_2$ with structural redundancy $\rho_\Sigma \leq \rho_{\Sigma_1}$. Σ_2 must involve at least one equation e and the constraint $\rho_\Sigma \leq \rho_{\Sigma_1}$ implies that e must involve at least one variable that is not shared with Σ_1. This is impossible because Σ would then include a just determined part, which is in contradiction with the fact that the global system is overdetermined (such that $\Sigma = \Sigma^+$), so $\rho_{\Sigma_1} < \rho_\Sigma$. This is easily generalized for any proper subsystem of Σ.

The condition $n_\ell > 1$, i.e. there are more than one subsystem in the highest level of the hierarchy, guarantees that the whole system is never recomposed and that D^3 only handles proper subsystems of Σ. Consequently, the time complexity of the D^3 algorithm is lower than the time complexity of its centralized equivalent.■

D. Implementation of D^3 and online Decentralized Diagnoser

The implementation of the diagnoser designed by the D^3 algorithm is based on the notion of local fault signature matrix of the subsystems, defined as follows.

Definition 13 (Local Fault Signature Matrix of a subsystem): Let a set $\mathcal{R}_{j,i}$ composed of $n^r_{j,i}$ ARRs and $F_{j,i}$ the set of considered $n^f_{j,i}$ faults for the subsystem $\Sigma_{j,i}$. The signature of a fault $f \in F_{j,i}$ is a binary vector $FS_{j,i}(f) = [\tau_1, \tau_2, ...\tau_{n^r_{j,i}}]^T$ where τ_k, $k = 1...n^r_{j,i}$, is computed from $\mathcal{R}_{j,i} \times F_{j,i} \longrightarrow \{0, 1\}$ so that $\tau_k = 1$, if $arr_k \in \mathcal{R}_{j,i}$ is affected by f, and $\tau_k = 0$ otherwise. The signatures of all the faults in $F_{j,i}$ put together constitute the Local Fault Signature Matrix, denoted $\mathcal{S}^l_{j,i}$ of subsystem $\Sigma_{j,i}$, i.e. $\mathcal{S}^l_{j,i} = [FS_{j,i}(f_1), \ldots, FS_{j,i}(f_{n^f_{j,i}})]^T$.

The diagnoser is computed offline as a hierarchical residual generator bank based on the local FMSO sets computed for each subsystem at each level. In accordance with the *isolation on request* concept [28], the computations are only carried out at a higher level if there exist faults that remain not isolable at the current level and the maximal isolability degree $\mathcal{I}^*(\Sigma)$ is not achieved. This check is done with the local fault signature matrix of the subsystems. Measurements of known variables are used online to evaluate residuals up along the levels until a fault is actually detected.

VI. Experiments

A. Computational Efficiency and Scalability in Practice

In order to experimentally show the computational efficiency of D^3 and its scalability, the algorithm is applied to serially connected water tanks. Setups of 4, 6, 8 and 10 tanks are tested.

The 4 tanks system model, initially presented in [24], is composed of 20 equations with 8 known variables, 14 unknown variables and 6 faults (see [29] for details). The construction of setups with more tanks is done by adding two tank patterns. Fig. 4 illustrates the 10 tanks setup, where T_1 to T_{10} represent the 10 tanks, u_1 to u_5 and y_1 to y_{18} are the known variables. Setups with 4, 6 or 8 tanks are a subpart of this setup.

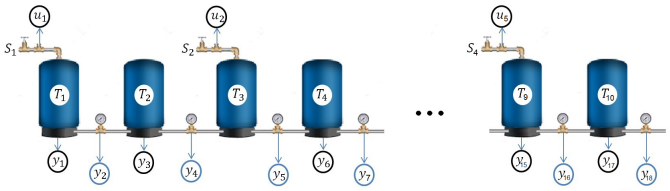

Fig. 4: Serially connected water tanks

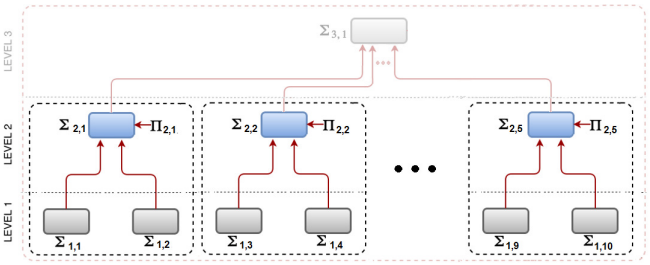

Fig. 5: Decentralized architecture diagnoser proposed for the tank system.

The hierarchical organization of the diagnoser proposed by D^3 for all setups is shown in Fig. 5. The D^3 algorithm and its centralized equivalent are applied to each setup. The execution time results reported in Table I leave no doubt about the gain recorded by D^3 as well as its scaling up.

TABLE I: Execution times for the centralized algorithm and D^3 algorithm

Setup	4 tanks	6 tanks	8 tanks	10 tanks
Number of tests	165	5977	33040	1296249
Centralized case	44.00 ms	637.03 ms	4965.28 ms	3984 s
Decentralized case	16.00 ms	28.05 ms	47.41 ms	59.81 ms
Local diagnosers:				
$\Sigma_{1,1}$ to $\Sigma_{1,10}$	$<$ 1ms	$<$ 1ms	$<$ 1ms	$<$ 1ms
$\Sigma_{2,1}$	9.00 ms	11.02 ms	13.01 ms	12.35 ms
$\Sigma_{2,2}$	7.06 ms	8.03 ms	10.01 ms	11.01 ms
$\Sigma_{2,3}$		9.03 ms	11.26 ms	12.24 ms
$\Sigma_{2,4}$			13.14 ms	10.43 ms
$\Sigma_{2,5}$				13.78 ms

B. Case Study: Gas Turbine Combined Cycle power plant

The method is applied to a gas turbine combined cycle power plant. A combined cycle power plant includes a gas turbine and a steam turbine. The remaining heat from the gas turbine is sent by a steam turbine that generates additional energy, producing up to 50% more electricity with the same amount of fuel than a conventional single-cycle plant. The nominal operation of a combined cycle power plant is more complex than than that of a conventional single cycle and therefore is more exposed to faults. First, natural gas is burned in a combustion chamber and passed through a gas turbine connected to an alternator. Hot gases are used to heat water and convert it into steam in a heat recovery unit. This steam is passed through a second turbine connected to another alternator, so that both generate electricity. One of the main

components of a gas turbine combined cycle power plant is the gas turbine generator system, which in turn is composed of the following components: compressor, electric generator, combustion chamber, gas turbine and heat recovery as shown in Fig. 7.

Gas turbine installations are ubiquitous in industry so their monitoring and diagnosis meets industrial standards (cf. ISO 19860:2005 [45]). Most installations are equipped with a trend monitoring system [46]. This equipment acquires operating data describing the condition of the gas-turbine installation that are used for the computation of short-term and long-term trends for selected parameters in mainly four areas: thermodynamics, vibrations, bearing lubrication, and control [47]. Without calling into question the usefulness of trend monitoring systems, model-based approaches applied to gas turbines have shown their interest in many works by taking advantage of the analytical redundancy provided by the model [48], [49]. Beyond monitoring the trends of isolated parameters, they make it possible to take into account the underlying relationships and existing interactions, which is highly beneficial for automatic diagnostics. This is why it is interesting to illustrate the performance of D^3 on a gas turbine generator.

The non-linear mathematical model of the gas turbine generator system used in this paper is based on thermodynamics, mass and energy conservation, fluid flow and heat transfer. The assumptions of this model are: ideal gas behavior, isentropic compression and linear flow characteristics for all control valves. The whole model $\Sigma(z, x, \mathtt{f})$ is composed of 37 equations e_1 to e_{37} that relate 20 known variables $Z = \{z_1, \ldots, z_{20}\}$ and 29 unknown variables $X = \{x_1, \ldots, x_{29}\}$. The set of target faults is $F = \{f_1, \ldots, f_5\}$ of cardinal n_f=5, where f_1 is a fault in the air pressure sensor at compressor discharge, f_2 is a fault in the position of the compressor inlet guide valves, f_3 is a fault in the position of the combustion chamber gas fuel valve, f_4 is a fault in the heat recovery pressure sensor and f_5 is a fault in the position of the combustion chamber gas valve. The full description of the model is given in [50]. A natural decomposition of the system provides the following five physically separated subsystems:

Compressor subsystem: $\Sigma_{1,1}$
$\Sigma_{1,1} = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$, $F_{1,1} = \{f_1, f_2\}$
$X_{1,1} = \{x_1, x_3, x_5, x_6, x_8, x_9, x_{10}, x_{28}\}$
$Z_{1,1} = \{z_1, z_2, z_3, z_4, z_5, z_6, z_7, z_{17}\}$

Combustion Chamber subsystem: $\Sigma_{1,2}$
$\Sigma_{1,2} = \{e_9, e_{11}, e_{12}, e_{13}, e_{14}, e_{15}, e_{16}, e_{17}\}$, $F_{1,2} = \{f_3\}$
$X_{1,2} = \{x_1, x_2, x_6, x_7, x_{10}, x_{12}, ..., x_{15}, x_{28}\}$
$Z_{1,2} = \{z_1, z_6, z_8, z_{18}\}$

Gas Turbine subsystem: $\Sigma_{1,3}$
$\Sigma_{1,3} = \{e_{18}, e_{19}, e_{20}, e_{21}, e_{22}, e_{23}, e_{24}, e_{25}\}$, $F_{1,3} = \{\}$
$X_{1,3} = \{x_1, x_4, x_6, x_8, x_{10}, x_{11}, x_{12}, x_{15}, ..., x_{19}, x_{29}\}$
$Z_{1,3} = \{z_1, z_2, z_6, z_{10}, z_{11}, z_{12}, z_{13}\}$

Electric Generator subsystem: $\Sigma_{1,4}$
$\Sigma_{1,4} = \{e_{26}, e_{27}, e_{28}\}$, $F_{1,4} = \{\}$
$X_{1,4} = \{x_{20}, x_{21}\}$
$Z_{1,4} = \{z_2, z_{13}\}$

Heat Recovery subsystem: $\Sigma_{1,5}$
$\Sigma_{1,5} = \{e_{29}, e_{30}, e_{31}, e_{32}, e_{33}, e_{34}, e_{35}\}$, $F_{1,5} = \{f_4, f_5\}$
$X_{1,5} = \{x_{15}, x_{23}, x_{24}, x_{25}, x_{26}, x_{27}\}$
$Z_{1,5} = \{z_{10}, z_{11}, z_{14}, z_{15}, z_{16}, z_{19}\}$

Besides, the functional decomposition considers the existence of restricted information not directly available for the subsystems in the first level and therefore only available in the upper hierarchical level 2. Restricted information has different sources, e.g. owner confidentiality, large distance between subsystems or difficult access. For level 2, it is denoted by $\Pi_{2,1}$ and $\Pi_{2,2}$. The corresponding additional sets of equations for forming subsystems $\Sigma_{2,1}$ and $\Sigma_{2,2}$ are $\Pi_{2,1} = \{e_8, e_{10}\}$ and $\Pi_{2,2} = \{e_{36}, e_{37}\}$, respectively.

Based on the system model and restricted information, the hierarchical organization of the diagnoser proposed for the gas turbine generator system is shown in Fig. 6. It includes 8 subsystems distributed in 3 levels, the 3rd level being the last possible level, i.e. m=3.

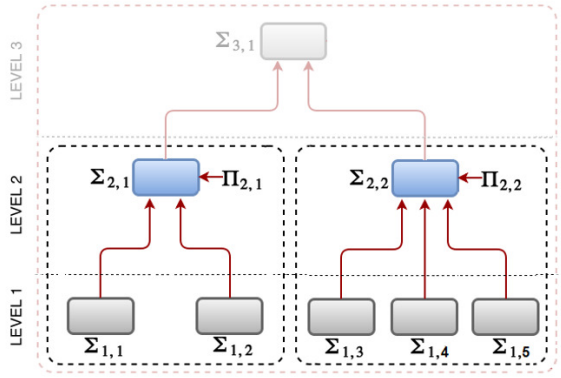

Fig. 6: Decentralized architecture diagnoser proposed for the gas turbine generator system.

C. Global FMSO Sets Calculation

As a way to validate the method, we determine the system fault isolability $\mathcal{I}(\Sigma)$ by calculating the set of global FMSO sets Φ assuming that the entire model is available.

TABLE II: Gas turbine generator system: Global FMSO sets

$\mathcal{A}(\Sigma)$	$[f_1]$,$[f_2]$,$[f_3]$,$[f_4]$,$[f_5]$	$\mathcal{I}(\Sigma)$	5	$\lvert\Phi\rvert$	59

As shown in Table II, all faults can be detected and isolated. Hence, the system isolability degree, which bounds isolability, is $\mathcal{I}(\Sigma) = n_f = 5$.

D. Decentralized Diagnoser Design

In this section, the operational algorithm for decentralized diagnosis design D^3 is applied with a targeted isolability degree equal to the system isolability degree, i.e. $\mathcal{I}^*(\Sigma) = \mathcal{I}(\Sigma) = n_f = 5$.

1) Shared variables: As a previous step to apply the proposed algorithm, the set of shared variables (cf. definition 9) is determined.

$$X^s = \{x_1, x_6, x_8, x_{10}, x_{12}, x_{15}, x_{28}\} \tag{11}$$

X^s is composed of 7 variables as shown in Fig. 7. Then, X^s is used to calculate *shared FMSO / CMSO sets* for each subsystem.

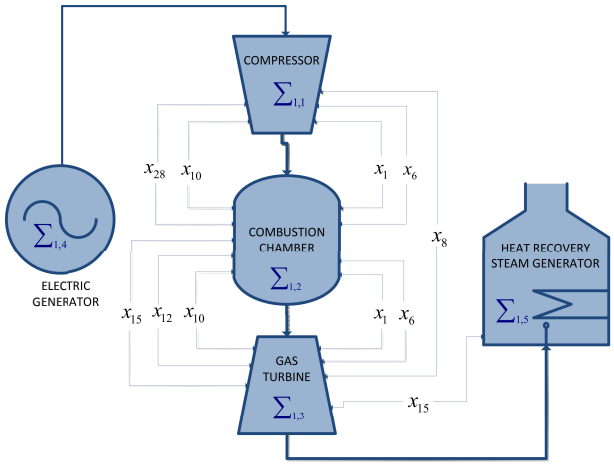

Fig. 7: Decentralized architecture for the gas turbine generator system.

2) Local FMSO sets at level 1: These are computed for $\Sigma_{1,1}$, $\Sigma_{1,2}$, $\Sigma_{1,3}$, $\Sigma_{1,4}$ and $\Sigma_{1,5}$ of level 1. None FMSO set is found for $\Sigma_{1,1}$, $\Sigma_{1,3}$ and $\Sigma_{1,4}$. Therefore, no fault can be detected locally for these subsystems.

For both subsystems $\Sigma_{1,2}$ and $\Sigma_{1,5}$ of level 1, only one FMSO set is found. Hence, the found FMSO sets $\Phi^l_{1,2}$ and $\Phi^l_{1,5}$ are selected without need to solve the BILP problems, allowing detectability for faults f_3 and f_4 locally in $\Sigma_{1,2}$ and $\Sigma_{1,5}$ respectively.

TABLE III: Selected Local FMSO Sets and isolability degree at level 1

$\Sigma_{1,2}$					
$F_{1,2}$	$\{f_3\}$	$\mathcal{A}(\Sigma_{1,2})$	$\{\{f_3\}\}$	$\mathcal{I}(\Sigma_{1,2})$	1
Local FMSO sets selected	$\Phi^{l\,*}_{1,2} = \{\varphi_1\}$, $\varphi_1 = \{e_{16}, e_{17}\}$				
$\Sigma_{1,5}$					
$F_{1,5}$	$\{f_4, f_5\}$	$\mathcal{A}(\Sigma_{1,5})$	$\{\{f_4\},\{f_5\}\}$	$\mathcal{I}(\Sigma_{1,5})$	2
Local FMSO sets selected	$\Phi^{l\,*}_{1,5} = \{\varphi_2\}$, $\varphi_2 = \{e_{29}, e_{30}, e_{31}, e_{32}, e_{33}, e_{34}\}$				
Isolability degree at level 1					
$\mathcal{A}_1(\Sigma)$	$\{\{f_1 f_2, f_5\},\{f_3\},\{f_4\}\}$	$\mathcal{I}_1(\Sigma)$	3		

At this stage, if f_3 or f_4 occur, they are diagnosed at level 1, without need of further analysis. However, not all system faults can be diagnosed at level 1.

The D^3 algorithm checks if the targeted isolability degree is achieved at level 1, which is not the case, since we have $\mathcal{I}_1(\Sigma) = 3 < \mathcal{I}^*(\Sigma) = 5$ (cf. Table III). The higher level 2 is thus explored as shown in Fig. 6.

TABLE IV: Shared FMSO and CMSO sets at level 1

Subsystem $\Sigma_{1,1}$	
Shared FMSO sets	$\Phi^s_{1,1} = \{\varphi_3, \varphi_4, \varphi_5\}$, $\varphi_3 = \{e_2, e_5, e_6\}$, $\varphi_4 = \{e_2, e_3\}$, $\varphi_5 = \{e_7\}$
Shared CMSO sets	$\Psi^s_{1,1} = \{\psi_1, \psi_2, \psi_3\}$ $\psi_1 = \{e_4\}$, $\psi_2 = \{e_3, e_5, e_6\}$, $\psi_3 = \{e_1\}$
Subsystem $\Sigma_{1,2}$	
Shared FMSO sets	$\Phi^s_{1,2} = \{\varphi_6\}$, $\varphi_6 = \{e_{16}, e_{17}\}$
Shared CMSO sets	$\Psi^s_{1,2} = \{\psi_4, \psi_5, \psi_6\}$, $\psi_4 = \{e_{12}, e_{13}, e_{14}, e_{15}\}$, $\psi_5 = \{e_9\}$, $\psi_6 = \{e_{11}\}$
Subsystem $\Sigma_{1,3}$	
Shared CMSO sets	$\Psi^s_{1,3} = \{\psi_7, \psi_8, \psi_9, \psi_{10}, \psi_{11}, \psi_{12}, \psi_{13}, \psi_{14}\}$ $\psi_7 = \{e_{18}\}$, $\psi_8 = \{e_{19}\}$, $\psi_9 = \{e_{20}\}$, $\psi_{10} = \{e_{21}\}$, $\psi_{11} = \{e_{22}\}$, $\psi_{12} = \{e_{23}\}$, $\psi_{13} = \{e_{24}\}$, $\psi_{14} = \{e_{25}\}$
Subsystem $\Sigma_{1,4}$	
Shared CMSO sets	$\Psi^s_{1,4} = \{\psi_{15}, \psi_{16}, \psi_{17}\}$ $\psi_{15} = \{e_{26}\}$, $\psi_{16} = \{e_{27}\}$, $\psi_{17} = \{e_{28}\}$
Subsystem $\Sigma_{1,5}$	
Shared FMSO sets	$\Phi^s_{1,5} = \{\varphi_7, \varphi_8, \varphi_9, \varphi_{10}\}$,$\varphi_7 = \{e_{30}, e_{31}, e_{32}, e_{33}, e_{34}\}$, $\varphi_8 = \{e_{29}, e_{31}, e_{34}\}$, $\varphi_9 = \{e_{29}, e_{30}, e_{31}, e_{32}, e_{33}\}$, $\varphi_{10} = \{e_{35}\}$
Shared CMSO sets	$\Psi^s_{1,5} = \{\psi_8\}$, $\psi_{18} = \{e_{29}, e_{30}, e_{32}, e_{33}, e_{34}\}$

3) Building level 2: Because faults f_1 and f_2 are not yet isolable, D^3 builds subsystem $\Sigma_{2,1}$ at level 2. Similarly, because fault f_5 is not yet isolable, D^3 builds subsystem $\Sigma_{2,2}$. These 2 new subsystems are built on demand, following the *isolation on request* concept.

To form $\Sigma_{2,1}$ and $\Sigma_{2,2}$, function `ComputeUpsilon()` is first used to retrieve the shared FMSO / CMSO sets $\Upsilon_{2,1}$ and $\Upsilon_{2,2}$ of the children systems of level 1 $\Sigma_{1,1}$, $\Sigma_{1,2}$, $\Sigma_{1,3}$, $\Sigma_{1,4}$ and $\Sigma_{1,5}$. Note that subsystems $\Sigma_{1,3}$ and $\Sigma_{1,4}$ are not impacted by any fault, leading to CMSO sets only (cf. Table IV). $\Upsilon_{2,1}$ and $\Upsilon_{2,2}$ are then lumped together with the additional information $\Pi_{2,1}$ and $\Pi_{2,2}$ (see architecture of Fig. 6) to obtain:

$$\Sigma_{2,1} = \{e_1, e_2, ..., e_7, e_9, e_{11}, ..., e_{17}\} \cup \{e_8, e_{10}\} \tag{12}$$

$$\Sigma_{2,2} = \{e_{18}, e_{19}, ..., e_{25}, ..., e_{29}, ..., e_{35}\} \cup \{e_{36}, e_{37}\} \tag{13}$$

4) Local FMSO sets at level 2: Three local FMSO sets are found for subsystem $\Sigma_{2,1}$. Solving a BILP program returns an optimal selection of two FMSO sets for this subsystem, guaranteeing isolation of faults f_1 and f_2. On the other hand, one local FMSO set is found for subsystem $\Sigma_{2,2}$, guaranteeing isolation of faults f_4 and f_5 (cf. Table V).

At this stage, the maximal isolability degree is achieved since $\mathcal{I}_2(\Sigma) = \mathcal{I}^*(\Sigma) = 5$. D^3 hence stops, having developed $\ell = 2$ levels. For this system, $\ell = 2 < m = 3$ and it has not been necessary to build up level 3, which is shown in transparency in Fig. 6.

E. Online Decentralized Diagnoser

The D^3 algorithm delivers a near-optimal distribution of ARRs on a hierarchy of subsystems that guarantees maximal

TABLE V: Subsystems $\Sigma_{2,1}$ and $\Sigma_{2,2}$: Selected Local FMSO Sets

$\Sigma_{2,1}$	
$\mathcal{A}(\Sigma_{2,1})$ $\{\{f_1\}, \{f_2\}, \{f_3\}\}$	$\mathcal{I}(\Sigma_{2,1})$ 3
Local FMSO sets selected	$\Phi^{l^*}_{2,1} = \{\varphi_{211}, \varphi_{212}\}$, $\varphi_{211} = \{e_7, e_8\}$, $\varphi_{212} = \{e_1, e_2, e_5, e_6, e_9, e_{10}, e_{11}, e_{12}, e_{13}, e_{14}, e_{15}\}$
$\Sigma_{2,2}$	
$\mathcal{A}(\Sigma_{2,2})$ $\{\{f_4\}, \{f_5\}\}$	$\mathcal{I}(\Sigma_{2,2})$ 2
Local FMSO sets selected	$\Phi^{l^*}_{2,2} = \{\varphi_{221}\}$, $\varphi_{221} = \{e_{35}, e_{37}\}$
Isolability degree at level 2	
$\mathcal{A}_2(\Sigma)$ $\{\{f_1\}, \{f_2\}, \{f_3\}, \{f_4\}, \{f_5\}\}$	$\mathcal{I}_2(\Sigma)5$

fault isolability. The analytical redundancy relations are computed from the selected FMSO sets according to Tables III and V. The decentralized diagnoser has two levels formed by residuals arr_1 for subsystem $\Sigma_{1,2}$, and arr_2 for subsystem $\Sigma_{1,5}$ at level 1, and by residuals arr_3 and arr_4 for subsystem $\Sigma_{2,1}$, and arr_5 for subsystem $\Sigma_{2,2}$ at level 2. Table VI reports the obtained signature matrix that indicates maximal isolability.

TABLE VI: Decentralized signature matrix for the gas turbine generator System

		f_1	f_2	f_3	f_4	f_5
Level 1	$arr_1 \in ARR_{1,2}$			X		
	$arr_2 \in ARR_{1,5}$				X	
Level 2	$arr_3 \in ARR_{2,1}$		X			
	$arr_4 \in ARR_{2,1}$	X				
	$arr_5 \in ARR_{2,2}$					X

A faulty case has been chosen to exemplify the method. Using a simulator of the GTCC, f_3 has been injected at 600s during 25s. Fig. 8 shows the set of residuals when fault f_3 occurs. As can be seen, this fault is detected at level 1 by residue arr_1 while all other residuals remain constant [3].

The algorithms for optimal selection of FMSO sets were implemented in Python and executed on a PC with a 1.6GHz processor. The execution times were measured in milliseconds, comparing the global diagnoser architecture that requires 4 out of 59 FMSO sets to achieve maximal isolability (cf. Section VI-C Table II) and the decentralized diagnoser architecture of D^3 that requires 5 FMSO sets in two levels (cf. Section VI-D, Tables III and V), which confirms the near-optimality result of Theorem 2. Results are presented in Table VII and show that the decentralized diagnoser architecture is faster than the global diagnoser architecture, which confirms the theoretical time complexity result of Theorem 3.

VII. Conclusion

Systems submitted to confidentiality constraints, geographically distributed or with limited access to some information require specific diagnosis architectures. This paper proposes a design method for decentralized diagnosis architectures named D^3 that meet this requirement.

D^3 computes recursively, by developing just the necessary levels, the set of residual generators guaranteeing maximal isolability. It has been theoretically proved of lower time complexity than its centralized equivalent.

The industrial case study of a Gas Turbine Combined Cycle Power Plant has been used to demonstrate the validity and performances of the decentralized diagnoser design approach.

The main advantages of the proposed decentralized architecture are that local diagnosers are designed according to the traditional function-by-function design organization, that subsystem models do not need to be exposed and that fault isolation is guided by the concept of isolation on request.

In future work, there is interest in investigating the double problem of choosing the system decomposition and diagnosis tests while minimizing inter-subsystem communication and maximizing diagnosability. A preliminary proposal along these lines is presented in [51]. Additionally, the impact of different inter-level communication constraints on the diagnosis results should be analyzed.

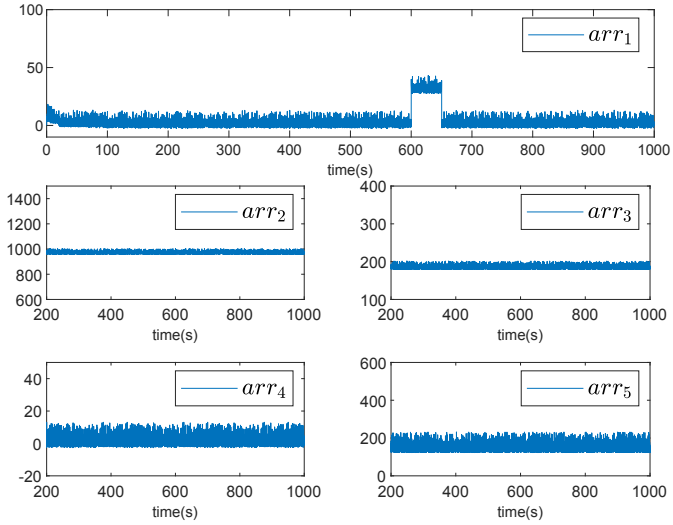

Fig. 8: Residuals for fault f_3 showing the fault at 600s by residual arr_1, while all other residuals remain constant[3]

TABLE VII: Execution times for global and decentralized architectures

Architecture	Execution Time	number of FMSO Selected
Global diagnoser	121.088 ms	4
Decentralized diagnoser	44.096 ms	5
Local diagnosers:		
$\Sigma_{1,1}$ to $\Sigma_{1,5}$	1.021 ms	2
$\Sigma_{2,1}$ to $\Sigma_{2,2}$	43.075 ms	3

References

[1] C. Edwards, S. K. Spurgeon, and R. J. Patton, "Sliding mode observers for fault detection and isolation." *Automatica, vol. 36(4), pp. 541-553*, 2000.

[2] R. Isermann, *Fault-Diagnosis applications: model-based condition monitoring: actuators, drives, machinery, plants, sensors, and fault-tolerant systems*. Springer Science & Business Media, 2011.

[3] If there were any, constants have been ignored in the expression of the ARRs.

[3] J. Gertler, *Structured Parity Equations in Fault Detection and Isolation*. Springer London, 2000.

[4] S. Diop, "Elimination in control theory," *Mathematics of control, signals and systems*, vol. 4, no. 1, pp. 17–32, 1991.

[5] Q. Zhang, M. Basseville, and A. Benveniste, "Fault detection and isolation in nonlinear dynamic systems: A combined input–output and local approach," *Automatica*, vol. 34, no. 11, pp. 1359–1373, 1998.

[6] J. Armengol, A. Bregón, T. Escobet, E. Gelso, M. Krysander, M. Nyberg, X. Olive, B. Pulido, and L. Travé-Massuyès, "Minimal structurally overdetermined sets for residual generation: A comparison of alternative approaches," *IFAC Proceedings Volumes*, vol. 42, no. 8, pp. 1480–1485, 2009.

[7] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*. Springer-Verlag Berlin Heidelberg, 2006.

[8] A. Bregon, M. Daigle, I. Roychoudhury, G. Biswas, X. Koutsoukos, and B. Pulido, "An event-based distributed diagnosis framework using structural model decomposition," *Journal of Artificial Intelligence Research, vol. 210, pp. 1-35*, 2014.

[9] R. J. Vanderbei, *Linear Programming: Foundations and Extensions*. Springer US, 2001.

[10] J. Cassar and M. Staroswiecki, "A structural approach for the design of failure detection and identification systems," in *IFAC Conference on Control of Industrial Systems, vol. 30(6), pp. 841-846*, 1997.

[11] L. Travé-Massuyès, T. Escobet, and X. Olive, "Diagnosability analysis based on component-supported analytical redundancy relations," *IEEE Trans. Syst., Man, Cybern. Part A: Sys and Humans, vol. 36(6), pp. 1146-1160*, 2006.

[12] M. Krysander, J. Aslund, and E. Frisk, "A structural algorithm for finding testable sub-models and multiple fault isolability analysis," in *International Workshop on Principles of Diagnosis*, 2010.

[13] M. Krysander, F. Heintz, J. Roll, and E. Frisk, "FlexDx: A reconfigurable diagnosis framework," *Engineering applications of artificial intelligence*, vol. 23, no. 8, pp. 1303–1313, 2010.

[14] L. Travé, A. Titli, and A. Tarras, *Large scale systems: decentralization, structure constraints and fixed modes*. Springer, 1989, vol. 120.

[15] Y. Wang, T.-S. Yoo, and S. Lafortune, "Diagnosis of discrete event systems using decentralized architectures," *Discrete Event Dynamic Systems*, vol. 17, no. 2, pp. 233–263, 2007.

[16] X. Cong, M. P. Fanti, A. M. Mangini, and Z. Li, "Decentralized diagnosis by Petri nets and integer linear programming," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, no. 99, pp. 1–12, 2017.

[17] A. Wada and S. Takai, "Decentralized diagnosis of discrete event systems subject to permanent sensor failures," *Discrete Event Dynamic Systems*, pp. 1–35, 2021.

[18] F. Boem, L. Sabattini, and C. Secchi, "Decentralized fault diagnosis for heterogeneous multi-agent systems," in *Conference on Control and Fault-Tolerant Systems (SysTol)*, 2016, pp. 771–776.

[19] X. Zhang, "Decentralized fault detection for a class of large-scale nonlinear uncertain systems," in *American Control Conference, pp. 5650-5655*, 2010.

[20] H. Ferdowsi, D. L. Raja, and S. Jagannathan, "A decentralized fault detection and prediction scheme for nonlinear interconnected continuous-time systems," in *IJCNN*. IEEE, 2012, pp. 1–7.

[21] S. Meynen, S. Hohmann, and D. Feßler, "Robust fault detection and isolation for distributed and decentralized systems," in *IEEE International Conference on Systems, Man, and Cybernetics*, 2020, pp. 401–407.

[22] S. Zhou, J. Bai, and F. Wu, "Decentralized fault detection and fault-tolerant control for nonlinear interconnected systems," *Processes*, vol. 9, no. 4, p. 591, 2021.

[23] V. Gupta and V. Puig, "Decentralized fault diagnosis using analytical redundancy relations: Application to a water distribution network," in *European Control Conference (ECC)*, 2016, pp. 1752–1757.

[24] H. Khorasgani, G. Biswas, and D. E. Jung, "Minimal structurally overdetermined sets selection for distributed fault detection." in *International Workshop on Principles of Diagnosis*, 2015, pp. 75–82.

[25] C. Zhao, H. Sun, and F. Tian, "Total variable decomposition based on sparse cointegration analysis for distributed monitoring of nonstationary industrial processes," *IEEE Transactions on Control Systems Technology, vol. 28(4)*, pp. 1–8, 2019.

[26] C. G. Pérez, E. Chanthery, L. Travé-Massuyès, and J. Sotomayor, "Fault driven minimal structurally overdetermined set in a distributed context," *International Workshop on Principles of Diagnosis*, 2016.

[27] E. Taskent and V.Puig, "Comparison of two distributed fault diagnosis approaches based on Binary Integer Linear Programming (BILP) optimization," in *International Workshop on Principles of Diagnosis*, 2018.

[28] E. Chanthery, L. Travé-Massuyès, and S. Indra, "Fault isolation on request based on decentralized residual generation," *IEEE Trans. on Sys., Man, and Cybernetics: Systems, vol. 46(5)*, 2016.

[29] C. G. Pérez, E. Chanthery, L. Travé-Massuyès, and J. Sotomayor, "Fault-driven structural diagnosis approach in a distributed context," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 14 254–14 259, 2017.

[30] H. Akaike, "A new look at the statistical model identification," *IEEE transactions on automatic control*, vol. 19, no. 6, pp. 716–723, 1974.

[31] W. Jin, Y. Ma, X. Liu, X. Tang, S. Wang, and J. Tang, "Graph structure learning for robust graph neural networks," in *26th ACM SIGKDD Int. Conf. on Knowledge Discovery & Data Mining*, 2020, pp. 66–74.

[32] E. Chow and A. Willsky, "Analytical redundancy and the design of robust failure detection systems," *IEEE Transactions on automatic control*, vol. 29, no. 7, pp. 603–614, 1984.

[33] E. Frisk and M. Nyberg, "A minimal polynomial basis solution to residual generation for fault diagnosis in linear systems," *Automatica*, vol. 37, no. 9, pp. 1417–1424, 2001.

[34] M. Nyberg and E. Frisk, "Residual generation for fault diagnosis of systems described by linear differential-algebraic equations," *IEEE Trans. on Automatic Control*, vol. 51, no. 12, pp. 1995–2000, 2006.

[35] C. Svärd and M. Nyberg, "Residual generators for fault diagnosis using computation sequences with mixed causality applied to automotive systems," *IEEE Trans. Syst. Man. Cybern. Part A*, vol. 40, no. 6, pp. 1310–1328, 2010.

[36] E. Frisk and J. Åslund, "Lowering orders of derivatives in non-linear residual generation using realization theory," *Automatica*, vol. 41, no. 10, pp. 1799–1807, 2005.

[37] E. Frisk and M. Krysander, "Residual selection for consistency based diagnosis using machine learning models," *IFAC-PapersOnLine*, vol. 51, no. 24, pp. 139–146, 2018.

[38] D. Düstegör, E. Frisk, V. Cocquempot, M. Krysander, and M. Staroswiecki, "Structural analysis of fault isolability in the damadics benchmark," *Control Engineering Practice, vol. 14(6), pp. 597-608*, 2006.

[39] K. Murota, *Matrices and Matroids for System Analysis*. Springer, 2000.

[40] A. L. Dulmage and N. S. Mendelsohn, "Coverings of bipartite graphs." *Canadian Journal of Mathematics, vol. 10, pp. 517-534*, 1958.

[41] C. Gei, F. Boem, and T. Parisini, "Optimal system decomposition for distributed fault detection: Insights and numerical results," *IFAC-PapersOnLine*, vol. 51, no. 24, pp. 578–585, 2018.

[42] J. Kościelny and A. Sztyber, "Decomposition of complex diagnostic systems," *IFAC-PapersOnLine*, vol. 51, no. 24, pp. 755–762, 2018.

[43] C. Artigues, O. C. Bassène, E. Chanthery, A. Gasmi, and L. Travé-Massuyès, "Optimal test/sensor selection problems formalized as integer programs," in *International Workshop on Principles of Diagnosis*, 2018.

[44] M. Krysander, J. Aslund, and M. Nyberg., "An efficient algorithm for finding minimal over-constrained sub-systems for model-based diagnosis," *IEEE Trans. Syst. Man Cy. A., vol. 38(1), pp. 197-206*, 2008.

[45] "Gas turbines — Data acquisition and trend monitoring system requirements for gas turbine installations," International Organization for Standardization, Standard ISO 19860:2005, Mar. 2005.

[46] C. De Michelis, C. Rinaldi, C. Sampietri, and R. Vario, "Condition monitoring and assessment of power plant components," in *Power Plant Life Management and Performance Improvement*. Elsevier, 2011, pp. 38–109.

[47] T. Giampaolo, *Gas turbine handbook: principles and practice*. CRC press, 2020.

[48] L. Travé-Massuyès and R. Milne, "Gas-turbine condition monitoring using qualitative model-based diagnosis," *IEEE Expert*, vol. 12, no. 3, pp. 22–31, 1997.

[49] C. Verde and M. Sanchez-Parra, *Application of Structural Analysis to Improve Fault Diagnosis in a Gas Turbine, Gas Turbines*. IntechOpen, 2010.

[50] M. Sanchez-Parra, "PID fault tolerant control for a gas turbine," in *Doctoral thesis, UNAM*, 2010.

[51] E. Chanthery, A. Sztyber, L. Travé-Massuyès, and C. G. Pérez, "Process decomposition and test selection for distributed fault diagnosis," in *Int. Conf. on Industrial, Engineering & Other Applications of Applied Intelligent Systems (IEA/AIE)*, 2020.

Gustavo Pérez-Zuñiga received the Ph.D. degree in Control Engineering from the Institut National des Sciences Appliquées (INSA), Toulouse, France, in 2017. He received the master degree in Control and Automation Engineering from Pontifical Catholic University of Peru, Lima, Peru in 2008. Currently, he is Featured Researcher and Associate Professor in the Engineering Department of the the Pontifical Catholic University of Peru. His current research interests include fault diagnosis and advanced control for large-scale complex and robotics systems with special focus on data-based and model-based methods. Dr. Pérez-Zuñiga is a member of the Safeprocess Technical Committee (https://tc.ifac-control.org/6/4) of the International Federation of Automatic Control.

Elodie Chanthery received the Engineering and master's degrees in control engineering from the Ecole Nationale Supérieure d'Electrotechnique, d'Electronique, d'Informatique, d'Hydraulique et des Telecommunications, Toulouse, France, in 2002, and the Ph.D. degree in mission planning for autonomous vehicles from the Ecole Nationale Supérieure de l'Aéronautique et de l'Espace—SupAero, Toulouse, France, in 2005. She received the Habilitation à diriger des recherches from the Institut National Polytechnique de Toulouse, Toulouse, France, in 2018. She is currently an Assistant Professor of control with the Institut National des Sciences Appliquees, Toulouse, France and is a co-chair of the chair "Synergistic transformations in model-based and data-based diagnosis" in the Artificial and Natural Intelligence Toulouse Institute (ANITI), France (https://aniti.univ-toulouse.fr). Her research activities are conducted in LAAS-CNRS in the field of diagnosis of complex hybrid systems, focusing on the links between diagnosis and other tasks, such as planning or prognosis with model-based and data-based techniques.

Louise Travé-Massuyès holds a position of Directrice de Recherche at Laboratoire d'Analyse et d'Architecture des Systèmes, Centre National de la Recherche Scientifique (LAAS-CNRS, https://www.laas.fr), Toulouse, France. She graduated in control engineering from the Institut National des Sciences Appliquées (INSA), Toulouse, France, in 1982 and received the Ph.D. degree from INSA in 1984. Her main research interests are in dynamic systems diagnosis and supervision with special focus on model-based and qualitative reasoning methods and data mining. She has been particularly active in establishing bridges between the diagnosis communities of Artificial Intelligence and Automatic Control. She is member of the International Federation of Automatic Control IFAC (https://www.ifac-control.org/) Safeprocess Technical Committee, Treasurer of the Society of Automatic Control, Industrial and Production Engineering (https://www.sagip.org). She holds the chair "Synergistic transformations in model-based and data-based diagnosis" in the Artificial and Natural Intelligence Toulouse Institute (ANITI), France (https://aniti.univ-toulouse.fr) and serves as Associate Editor for the well-known Artificial Intelligence Journal (https://www.journals.elsevier.com/artificial-intelligence).

Javier Sotomayor received his M.Sc. in Automatic Control Engineering from Leningrad Polytechnic Institute, Russia in 1985, and a Ph.D. from Saint Petersburg Polytechnic University in 1993. He has been professor in the Engineering Department of the Pontificia Universidad Católica del Perú (PUCP) since 1994. Currently he is leading the Control and Automation Research Group, and is also director of the Master´s Program in Control and Automation Engineering at PUCP. His research interests include system identification for control, control of complex industrial processes and model based fault diagnosis.