Combining FDI and AI Approaches within Causal-Model-based Diagnosis

Sylviane Gentil, Jacky Montmain, Christophe Combastel

Combining FDI and AI Approaches Within Causal-Model-Based Diagnosis

Sylviane Gentil, Jacky Montmain, and Christophe Combastel

***Abstract*—This paper presents a model-based diagnostic method designed in the context of process supervision. It has been inspired by both artificial intelligence and control theory. AI contributes tools for qualitative modeling, including causal modeling, whose aim is to split a complex process into elementary submodels. Control theory, within the framework of fault detection and isolation (FDI), provides numerical models for generating and testing residuals, and for taking into account inaccuracies in the model, unknown disturbances and noise. Consistency-based reasoning provides a logical foundation for diagnostic reasoning and clarifies fundamental assumptions, such as single fault and exoneration. The diagnostic method presented in the paper benefits from the advantages of all these approaches. Causal modeling enables the method to focus on sufficient relations for fault isolation, which avoids combinatorial explosion. Moreover, it allows the model to be modified easily without changing any aspect of the diagnostic algorithm. The numerical submodels that are used to detect inconsistency benefit from the precise quantitative analysis of the FDI approach. The FDI models are studied in order to link this method with DX component-oriented reasoning. The recursive on-line use of this algorithm is explained and the concept of local exoneration is introduced.**

***Index Terms*—Causal graph, causal reasoning, diagnosis, fault detection, fault filtering, fault isolation, supervision.**

I. Introduction

FAULT analysis is an important activity in almost all industries. The need for dependability in industrial plants or availability of complex devices is becoming a major issue in the fulfillment of increasingly stringent requirements with respect to productivity or safety. Fault analysis is a prerequisite for safety studies, process supervision or establishing a maintenance policy. It can be carried out *a priori,* generally following on from failure mode effect analysis. In this case, all possible failures that could occur are hypothesized, their effects are predicted and the counteractions for eliminating or minimizing these effects are designed. Fault diagnosis generally addresses the study of faults during the routine use of the installation. Diagnostic results can be used to decide about on-line recovery actions or plant shut-down, and the maintenance schedule. The work presented in this paper is intended for on-line diagnosis of industrial processes such as nuclear plants, power plants or petrochemical plants.

The diagnosis of complex systems is complicated. It has been an area of very active investigation for many years. Two research communities have been particularly involved in studying fault diagnosis: the artificial intelligence community, known as the diagnostic (DX) community, and the control theory community, known as the fault detection and isolation (FDI) community. It is nevertheless worth noting that very few industrial applications have been reported [30], [37]. The general objective of AI is to reproduce human reasoning and more generally any human cognitive mode of comprehension, perception, representation, and decision making, as faithfully as possible. This is a major divergence with control theory, which processes numerical data and algorithms in order to stabilize systems or optimize production.

Within the DX community, diagnosis is considered as a reasoning process. Poole [45] distinguishes two kinds of reasoning for solving diagnostic tasks: normal-operation-oriented reasoning and abnormal-operation-oriented or abductive reasoning. Normal-operation-oriented diagnosis uses knowledge about how normal components work to detect deviations from normality in observed behavior, from which a minimal set of faults is hypothesized. Abnormal-operation-oriented diagnosis uses knowledge about how the components are affected by some specific faults in order to trace those faults. AI diagnostic techniques are varied. Empirical information and experience may be encoded as associative knowledge in rule bases. When many experimental data describing faults are available, case-based reasoning is a powerful approach. Model-based diagnosis uses deep knowledge of the device for diagnosis. Reiter [48] proposed a logical theory of model-based diagnosis, also referred to as consistency-based diagnosis. The analysis is aimed at obtaining consistency between the observations and the model by removing assumptions about the behavior of some components. This theory was extended and formalized in [18]. Many refinements have since been proposed. Struss [51] emphasizes many advantages of AI model-based diagnosis, such as the possibilities of explicit conceptual modeling, automated model composition, and structural model revision. The AI community has also proposed an important concept: qualitative modeling, an aspect of which is causal modeling, as used in this paper. Causal modeling enables a complicated process to be decomposed into elementary submodels and is thus very suitable for complex system analysis. Causal models provide explanations of the behavior of the modeled system that are close to human reasoning, which is completely excluded by the purely numerical calculus that constitutes the control basis supporting the FDI methods.

Manuscript received July 18, 2002; revised September 30, 2003. This paper was recommended by Guest Editor G. Biswas.

S. Gentil is with the Laboratoire d'Automatique de Grenoble, CNRS-INPG-UJF, 38402 Saint-Martin d'Hères Cedex, France (e-mail: sylviane.gentil@inpg.fr).

J. Montmain is with Ecole des Mines d'Alès, Site EERIE, 30035 Nîmes Cedex 1, France (e-mail: jacky.montmain@ema.fr).

C. Combastel is with Ecole Nationale Supérieure de l'Electronique et de ses Applications (ENSEA), 95014 Cergy-Pontoise Cedex, France (e-mail: combastel@ensea.fr).

FDI focuses on engineering systems, such as production facilities, machines, vehicles, electrical drives, etc., whether faults occur in the plant (the technical equipment itself) or in its measurement and control instruments (sensor and actuator faults) [23]. This approach makes a distinction between fault detection (deciding that faults are present) and fault isolation (deciding which particular fault is present) [19], [29]. As with most control theory, diagnosis is based on a dynamic quantitative model of the system under study, generally represented as a set of differential algebraic equations. In contrast to the AI approach, the model is numerical and as precise as required by the diagnostic objective. Generally, the model represents the normal behavior of the system, in the absence of any fault and characterizes deterministic phenomena, taken into consideration using basic "laws" of physics, biology, etc. But it can also include detailed knowledge about how faults or unknown disturbances affect the variables of the system. It can take into account noise, which is a stochastic process affecting the measurements and/or the system's behavior. Until recently, the management of noise, disturbances, and model errors was not seen as a major issue for the AI DX community. With the FDI approach, computations result in numerical quantities, the residuals, whose properties enable diagnosis with very accurate quantitative information. Nevertheless, several logical assumptions are implicit in the FDI formulation, whereas they are clearly formulated within the DX consistency-based diagnostic framework.

Reasoning and computing may thus be considered in opposition. The *combined method* presented in this paper brings them together. It relies on both a qualitative causal representation of the process and on quantitative local models. It has been inspired by artificial intelligence for the causal modeling of physical systems and for studying logical soundness. But it takes advantage of control theory at the level of each elementary submodel to check local consistency. Process dynamics are taken into account using relations between variables that manage time explicitly. Section II presents the interest of causal modeling for representing the normal behavior of complex systems and the basic principle of diagnostic causal reasoning. Nevertheless, knowing a causal structure is insufficient for making a successful diagnosis. In particular, variables and their relations may be represented in a numerical (FDI) or a symbolic (DX) manner. Section III briefly recaps and compares the advantages and disadvantages of these two approaches, to justify the choices that have been made. Section IV details the generation of causal model based residuals. Section V presents the proposed recursive isolation algorithm, which avoids combinatorial explosion, and places it in the contexts of both FDI and DX. The conclusion discusses the advantages and limits of this combined method.

II. Causal Models of Complex Systems

The causal-model-based diagnostic method presented in this paper concerns the design of a supervisory system. Nowadays, supervision is no longer designed to eliminate the operators from process control, but instead to support them in their decision-making tasks. The diagnostic procedure to be integrated into the supervisory system must therefore be provided with explanatory features. Techniques based on causal graphs are a pertinent approach for this purpose [21].

In this section, the notion of causality and how it can be applied to diagnosis is discussed, in order to justify the choices that have been made for the proposed method.

A. Causality

Causality occupies a central position in human cognition. Informal descriptions of real-world phenomena in the form A *causes* B, are exceedingly common. Causal descriptions are the source of various reasoning modes. B can be predicted or explained using A. Causality plays an essential role in human decision-making by providing a basis for choosing that action which is likely to lead to a desired result. [11] claims that diagnosis is typically a causal process, because it consists in designating the faulty components that have caused, and can explain, the observed malfunctions.

It is difficult to give a sound definition of causality, which should satisfy the following criteria. It is general rather than restricted to a narrow class of phenomena. It is precise and unambiguous and can be used as a basis for logical reasoning and/or computation. It can be employed to answer the questions. Did or does or will A cause B or vice-versa? If there is a causal connection between A and B, what is its strength? It does not lead to counterintuitive conclusions.

The AI community has been working for a long time on representations of causality. In particular, causal modeling, whether applied in the context of economical systems [31] or qualitative physics [16], has been the subject of a famous debate. Iwasaki and Simon are interested in causality when the system is modeled by a set of mathematical relations. Calculability imposes causal ordering. Causality is clear in differential relations. For instance, $\dot{y} = f(u, y)$ imposes an orientation of the resolution because u cannot be deduced from y; thus u must be computed from another equation. Difficulties appear when relations model simultaneous evolutions. For instance, Ohm's law $U = RI$ does not impose a direction, as it can be used to compute either U from I or the opposite. De Kleer and Brown define causality from an engineering point of view. The experienced engineer analyzes the functioning of a system only by propagating important conceptual entities through the system topology. The behavior of the system arises out of interactions between its constitutive components. Propagation of constraints is nondeterministic, discovering multiple orderings including, but not restricted to, Simon's causal ordering [17]. The differences between the two previous approaches result from one single fact. De Kleer and Brown link causality to the structure of the system (local analysis), while Iwasaki and Simon link causality to the form of the equations describing the system (global analysis).

Bayesian networks are another example of causal modeling in a situation where understanding of what is actually going on is incomplete, so entities have to be defined probabilistically [44]. Bayesian networks are directed acyclic graphs, where the nodes are random variables, and dependence assumptions that must hold between them are represented by the arcs. These relations are different from logical relations since they

allow conditional rather than implicative reasoning. The name "Bayesian networks" is completely neutral about the status of the networks [3]. Bayesian approaches have been proposed for learning Bayesian networks from a combination of prior knowledge and statistical data [26].

In this paper, the basis for process representation is causal representations of physical deterministic system behavior. A causal structure is a qualitative description of the effect or influence that system entities (variables, faults, etc.) have on other entities. It may be represented by a directed graph (digraph). A causal graph, which represents a process at a high level of abstraction, is appropriate for supervising the process.

In the following subsection, the use of causality and more specifically of causal graphs is illustrated in the field of diagnosis. Its main objective, in this respect, is to deal with the combinatorial explosion that arises with model-based approaches.

B. Diagnosing With Causal Graphs

[7] states that one of the main limitations in logical model-based diagnosis is its computational complexity, and proposes a specific knowledge compilation approach to focus reasoning on abductive diagnosis [10]. In the logical theory of *abductive diagnosis* [4]–[6], [35], [46] diagnosis is formalized as reasoning from effects to causes. Causal knowledge is represented as logical implications of the form $causes \rightarrow effects$ where causes are usually abnormalities or faults, but may also include normal situations. The pieces of causal knowledge can be organized in a directed graph. This abductive type of reasoning contrasts with deductive reasoning from causes to effects.

Emphasis on structure has been the central theme in probabilistic reasoning and several attempts have imported this theme into model-based diagnosis. [15], [24] and [14] propose a comprehensive approach for model-based diagnosis that includes characterizing and computing preferred consequences—one consequence is a Boolean expression that characterizes consistency-based diagnoses—assuming that the system description is augmented with a system structure represented as a directed graph, explaining the interconnections between system components. With a formulation based on logic, Darwiche shows that there is a connection between the complexity of computing consequences and the topology of the underlying system structure. Diagnosis becomes easier because the causal structure of a system explains independences that can be used to decompose the global consequences into local ones that can be evaluated locally. Minimal diagnoses are those which are considered to be most plausible. An algorithm that enumerates the minimal diagnoses, characterized by a consequence, is proposed. What is most important about Darwiche's approach is that it ties the computational complexity of diagnostic reasoning to the topology of a system structure [13].

Causality, assimilated to calculability, has also been used in FDI approaches. The structural model of a system represents its normal behavior and is made up of a set of formal equations. A Boolean matrix, known as the *incidence matrix*, represents the *system structure* [50]. The columns represent variables, the rows represent equations, and 1 in the matrix element i, j indicates that the variable j is used in equation i. A matching operation directs the links between relations and variables. This matching provides a breakdown that can be considered as a bipartite graph in which the nodes are alternately a variable, a relation, etc. The objective is to find relations that contain only known variables, which can be used for the purposes of diagnosis to check the coherence of the observations with the model. This can be shown to be equivalent to finding over-determined subsystems in the incidence matrix. Based on this bipartite graph, [25] proposes a causal-graph approach for studying system reconfigur-ability, which appears as a consequence of multiple controllability paths in the system's causal graph.

Influence graphs are another type of causal approach to diagnosis, which is used in this paper and detailed in the following subsection. They avoid fault modeling, which could be unfeasible in the case of a complex system. It provides a tool for reasoning about the way in which normal or abnormal changes propagate. It is suitable for physical explanations of the dynamical evolution of variables, whether normal or abnormal.

C. Diagnosis and Influence Graphs

When the graph nodes represent the system variables, the directed arcs symbolize the normal relations among them and these relations are deterministic, the graph is frequently referred to as an influence graph. No a priori assumption is made about the type of relations labeling each arc. They could be qualitative or quantitative. The digraph is above all a reasoning structure that can be enriched as knowledge becomes available. It can include loops. The simplest influence graph structure is the signed digraph (SDG). The branches are labeled by signs: "$+$" (or "$-$") when the variables at each end of the arc have the same (or opposite) trends. In this paper, arcs are labeled with dynamic quantitative relations, justified from the diagnostic needs of industrial plants.

All influence-graph-based diagnostic methods implement the same basic principle. The objective is to account for deviations detected in the evolution of the variables with respect to the normal behavior, using a minimum of malfunctions at the source. Malfunctions can be related to physical components, so as to obtain a minimal diagnosis. If significant deviations are detected, primary faults, directly attributable to a failure or an unmeasured disturbance, are hypothesized. The propagation paths in the directed graph are analyzed to determine whether this fault hypothesis is sufficient to account for secondary faults, resulting from its propagation in the process over time. The algorithm is a backward/forward procedure starting from an inconsistent variable. The backward search bounds the fault space by eliminating the normal measurements causally upstream. Then each possible primary deviation generates a hypothesis, which is forward tested using the states of the variables and the functions of the arcs.

A diagnostic method using an SDG as the basic data structure was initially presented in [28]. The state of a variable is expressed in the quantity-space $\{+, 0, -\}$, according to whether the value is normal (0), higher than normal $(+)$, or lower than normal $(-)$. The graph resulting from diagnosis is exclusively made up of signed nodes ($+$ or $-$) and consistent branches—branches for which the product of the signs of the initial and final nodes is the same as the sign of the branch. It is a representation of the propagation of the fault in the system.

The consistency test of the initial and final nodes is carried out recursively and constitutes the basic isolation procedure. The roots of such a subgraph are candidates for the origins of the failure.

This approach has since been considerably enhanced, essentially introducing more information on the variable states. This is justified by the ambiguity arising from a rough qualitative variable representation. A five-range pattern of the variable states $\{-, -?, 0, +?, +\}$ was proposed in [49] to avoid the pitfalls of a wrong diagnosis. The association of ambiguity symbols ($-?$ or $+?$) to the nodes avoids incorrect threshold choice and thus provides robustness. [42] uses numerical information to represent the deviations in variables. Representing variables with fuzzy sets is used in [53] to achieve progressive quantification.

None of these studies takes variables dynamics into account. This point is nevertheless important because the signatures of the observed faults can change over time. Temporal fault filtering is a required diagnostic functionality. [36] introduces quantitative temporal information within the arcs. The same representation is used in this paper. Arcs support differential or difference equations that are parameterized with quantitative parameters such as gains, delays, and time constants. How can such a digraph be obtained?

Following de Kleer and Brown's approach to causality, focused on the engineer's causal knowledge, a first method for obtaining such a causal model for a complex system is based on engineering knowledge. It relies on a functional top-down analysis of the process [32]. Nodes are selected as variables that are meaningful to the supervision operator, generally measured variables. Arcs can focus on various physical phenomena (balance, transportation, storage...). Temporal parameters in the dynamic relations supported by the arcs can be estimated using standard identification procedures [34].

Following Iwasaki and Simon's approach to causality, the causal graph may represent calculability. It can be obtained from a numerical simulator representing a system of differential equations. The causal relations between the variables are in this case implicit, related to the sampling of differential equations by the simulation algorithms. The digraph can also be deduced from the set of formal equations S_E describing the set of process variables S_V, arranged with a causal ordering mechanism [31]. The causal ordering can be performed within the theoretical framework of a bipartite graph [47]. The graph $G = (S_V \cup S_E, A)$ is defined, where A is the set of arcs such that an arc exists between $V_i \in V$ and $e_j \in E$ if and only if V_i is involved in e_j. The causal ordering therefore arises from determining a perfect matching in G [27], [52]. As in the engineer's approach, several causal orders can be found for the same system.

Bond graph formalism is another way to derive temporal influence graphs. Bond graphs have been proposed for a long time for the modeling of dynamic physical systems because they provide a systematic framework for building consistent and well constrained models suitable for multiple domains (electricity, mechanics...) [43]. They are deduced from a deep knowledge of the physical mechanisms occurring in the process and describe material and energy exchanges, accumulation or transportation. A bond graph can include causality constraints, even if, once again, directing some arcs in the graph may have several solutions [1]. [40] proposes using a bond-graph for diagnosis, labeling its arcs with the names of the components whose behavior is described by each particular arc. The description of the evolution of the variable is qualitative. In [39] and [40], the idea is to predict the future behavior of the system for each abnormal deviation in terms of their qualitative time-derivative changes. When a discrepancy between the measurement and the nominal value is detected, a backward propagation algorithm operates on the temporal causal graph to implicate component parameters. Next, a forward-propagation algorithm predicts dynamic qualitative deviations in magnitude and derivatives of the observations under the fault conditions. This is called the signature. Then, the monitoring module compares reported signatures and actual observations as they change dynamically after faults have occurred. Transients generated by failures are dynamic, so the signatures of the observed variables change over time.

The causal fault filtering method presented in this paper deals with the quantitative dynamic case in influence-graph-based diagnosis and can be compared to the progressive monitoring method. The problems mentioned in the progressive monitoring approach are solved by means of a quantitative approach. A measured variable is no longer described by its qualitative value and other higher order derivatives but by numerical values obtained by solving the dynamic equation associated to it. As long as qualitative parameters subsist in the model, diagnostic reasoning is subject to ambiguous decisions, whereas numerical values allow normal dynamic effects to be distinguished naturally from fault propagation.

The combined method presented in this paper takes advantage of the precision of FDI fault indicators because it uses a quantitative model. At the same time, it benefits from the results of the logical soundness of DX through the use of a causal structure that supports the diagnostic reasoning. The digraph is a reasoning structure and it enables explicit management of the logical assumptions made in the diagnostic reasoning.

III. Axioms of Diagnosis

The basic tools of the DX and FDI diagnostic approaches are briefly summarized in this section, to provide the definitions and concepts useful for understanding the proposed algorithm.

A. DX Approach

The DX community has been concerned with the modeling of the diagnostic reasoning itself: the foundations of logical reasoning have always been considered as major research points. In the consistency based approach [48], the description of the behavior of the system is component-oriented and rests on first-order logic. The {SD (system description), COMP (components)} pair constitutes the model. The system description takes the form of logical operations [12]. The extension of the predicate $ab(.)$ represents the set of abnormal components.

Let OBS be the set of observations. Diagnostic reasoning has been summarized in the following way [33]. A diagnosis is a minimal set $\Delta \subset$ COMP of abnormal components such that $\{ab(c) : c \in \Delta\} \cup \{\neg ab(c) : c \in \text{COMP} \setminus \Delta\} \cup \text{SD} \cup \text{OBS}$ is consistent. Δ is minimal if no subset $\Delta' \subset \Delta$ is a diagnosis.

The diagnosis relies on the conflict notion: a conflict is a set of components $C \subset \text{COMP}$ such that $\text{SD} \cup \text{OBS} \cup \{\neg ab(c) : c \in C\}$ is inconsistent; the observations indicate that at least one of its components must behave abnormally. A diagnosis is thus a set Δ of components such that $\text{COMP} \setminus \Delta$ is not a conflict. The diagnosis proceeds in two steps: the first step determines the set of conflict sets $\mathbf{C}$; the second step computes diagnoses from the conflict sets, using hitting sets: $\mathbf{H} \subseteq \bigcup_{C \in \mathbf{C}} C$ such that $H \cap C \neq 0$ for any C in $\mathbf{C}$. Reiter [48] has shown that $\Delta \subset \text{COMP}$ is a minimal diagnosis for $\{\text{SD}, \text{COMP}, \text{OBS}\}$ if and only if Δ is a minimal hitting set for the collection of conflict sets $\mathbf{C}$.

Diagnosis in this framework is logically sound but a major drawback is the issue of combinatorial explosion for systems involving many components [14], as in the case of industrial processes, with whose diagnosis this paper is concerned. Another difficult point is checking consistency in a qualitative framework, when numerical continuous valued OBS obtained, for instance, from an industrial plant data acquisition system, are used. An example can be found in [40].

B. FDI Approach

The FDI community is especially concerned with industrial process modeling and control. Models are quantitative and dynamic [22]. Two basic representations can be used: state space models and input–output relations. (1) is an example of an input–output relation, which takes into consideration the way faults f and unknown disturbances d affect the measurable output y of the system, excited by an input u

$$y = h(u, f, d, t). \tag{1}$$

y and u represent observations (OBS). Disturbances are uncontrolled input signals whose presence is undesired but normal (such as the wind for a plane or a resistive torque for a motor) and must be distinguished from faults. Noise is a special kind of disturbance related to random uncertainty. Faults are deviations from normal behavior in the plant or its instrumentation. Additive process faults are unknown inputs acting on the plant, which are normally zero. Multiplicative process faults lead to changes in model parameters. Sensor and actuator faults are other significant types of faults, represented as additive signals. A model (1) can take into account both additive faults (extra signals) and modifications to the model parameters (change in h).

The model is used to compute numerical fault indicators, known as residuals, r_j, which are null when there is no fault affecting the system. Residual generation refers to the elaboration of relevant fault indicators and has received much attention within the FDI community. It is worth noting that a residual, by using appropriate filters, can represent a much more elaborate quantity than a simple comparison of a process measurement with its model prediction [23].

A residual r_j must have a computational form (2), known as an analytical redundancy relation, deduced from the model, depending only on OBS, possibly at different times

$$r_j = hc_j(u, y, t) \tag{2}$$

TABLE I
EXAMPLE OF INCIDENCE TABLE TO DESIGNATE STRUCTURED RESIDUALS

	f_1	f_2	f_3
r_1	1	1	0
r_2	1	0	1
r_3	0	1	1

The residual evaluation form is expressed by (3)

$$r_j = he_j(u, y, f, d, t) \tag{3}$$

which shows how it is influenced by the faults and the unknown disturbances. Ideally, a residual should be decoupled from the unknown disturbances and dependent only on a single fault f_j

$$r_j = he_j(u, y, f_j, t). \tag{4}$$

In (4), when f_j is null, r_j should be zero. In the DX view, if r_j is not zero, this results from an inconsistency between the model and the observations. When new data come from the acquisition system, residuals are computed using (2) and are interpreted to obtain Boolean symptoms. This step is known as fault detection. Using numerical models and numerical data considerably facilitates this computation. The decision can be made simply by comparing the residuals to a fixed threshold, obtained empirically. Knowing the way the model is affected by disturbances, noise, or parameter imprecision allows this decision to be given a mathematical foundation, using fuzzy set theory [21], [38], or statistical decision theory [2], well suited to detection in systems with model uncertainty or disturbed by random perturbations.

In realistic situations, it is not easy to obtain structured residuals such as (4): a residual is generally sensitive to a subset of faults. Consequently, the Boolean symptoms are organized according to an *incidence table* (also called signature table). An incidence table is a binary matrix where each line is associated to a residual r_j and each column is associated to a fault f_j. In this matrix, "1" means that the residual is sensitive to the fault. "0" means that the residual is perfectly decoupled from the fault (Table I). In the FDI context, only single faults are hypothesized, otherwise the incidence matrix would become much larger. A multiple-fault signature is generally roughly associated to the logic OR operation of the elementary signatures.

The diagnosis is obtained by means of on-line pattern matching: the residual vector (showing observed inconsistencies) is compared at each time to the columns of the incidence matrix (fault signatures). Thus diagnosis is reduced to finding a theoretical fault signature similar to the practical one. This step is known as fault isolation.

Analyzing this procedure from a logical point of view shows that it is not logically sound, which is its major drawback. In fact, it relies implicitly on the exoneration assumption, as has been fully highlighted in a collective work published in [8], [9] and is briefly explained here.

The exoneration assumption means that a faulty component necessarily shows its faulty behavior, i.e. causes any analytical redundancy relation (ARR) in which its model is involved not to be satisfied by any given set of observations. Equivalently, given the set of observations, any set of components whose model is involved in a satisfied ARR is exonerated, i.e. each component

of the ARR support is considered to behave correctly. In this general exoneration assumption, there is a single-fault exoneration assumption—each individual component shows its faulty behavior—and a noncompensation assumption—the individual effects of faulty components never compensate each other.

Let $\{ARR_1\} \subset \{ARR\}$ be the subset of potentially affected by a set of faults $F_1 \subset F_1$, and let f_p be the present fault. The exoneration assumption can be expressed as follows:

$$\{ARR_1(\text{OBS})\} = 0 \leftrightarrow f_p \in \overline{F_1} \tag{5}$$

where $\overline{F_1}$ is the complement of the set F_1 in F. (5) is equivalent to (6)

$$\{\text{ARR}_1(\text{OBS})\} \neq 0 \leftrightarrow f_p \in F_1. \tag{6}$$

When the exoneration assumption is not made

$$\{\text{ARR}_1(\text{OBS})\} = 0 \leftarrow f_p \in \overline{F_1} \tag{7}$$

whose contrapositive is

$$\{\text{ARR}_1(\text{OBS})\} \neq 0 \rightarrow f_p \in F_1. \tag{8}$$

C. Discussion

In conclusion, the FDI community has paid attention to numerical system modeling: taking into account model uncertainty, nonmeasurable disturbances, variable dynamics, and the possibility of noise. The generation and use of theoretical fault signatures reduces the diagnostic reasoning to a simple pattern-matching activity. Nevertheless, this procedure is not logically sound. Moreover, it is worth noting that even a small modification in the model structure leads to the necessity of restarting the generation of residuals from scratch, together with fault detection and fault isolation, which is a tremendous drawback of this approach. The DX community has been more concerned with modeling diagnostic reasoning. In particular, in the consistency-based approach, logical foundations have been considered as major research points. No particular assumption about the fault manifestations is needed. However, checking consistency can be a difficult point when using quantitative signal description.

The diagnostic method presented in the following paragraphs uses advantages from several approaches. Causal modeling, presented in Section II, enables the method to focus on relations that will be shown to be sufficient to allow fault isolation. This avoids the combinatorial explosion that could be feared when dealing with industrial plants. Moreover, it allows the model to be modified easily without changing anything in the diagnostic algorithm. It also enables the logical assumptions of the diagnosis reasoning to be managed clearly. The numerical submodels that are used to detect inconsistency benefit from the precise quantitative analysis of the FDI decision. The FDI models are studied in order to link this method with DX component-oriented reasoning. A weak restriction to the no-exoneration diagnostic framework is proposed that does not require the introduction of fault mode modeling as in [41]. For this reason, the concept of local exoneration is introduced.

IV. Causal-Graph-Based Residual Generation

A. The Causal Graph

The basic knowledge for dynamic causal models is knowledge of the causal dependence between some variables and of the equations relating these variables and modeling the system components. It is assumed below that only measured variables are represented in the causal graph. The nodes of causal graphs based on expert knowledge are often related to observations. Such variables are also the most meaningful in the context of human-centered process supervision. How to transform a causal graph with many intermediary variables into a graph with only measured variables is explained and illustrated on an industrial example in [27]. Among the variables involved in the equations modeling the components' normal behavior, the following ones can be distinguished:

$$\text{OBS} = E \cup Y \tag{9}$$

$$\text{OBS}^* = E^* \cup Y^*. \tag{10}$$

OBS refers to observations that are either sensor outputs Y or known exogenous inputs, E. E and Y are disjoint sets. OBS* refers to variables symbolizing the genuine value of the corresponding variable in OBS. There is a one-to-one relationship between variables in OBS (respectively, E, Y) and variables in OBS* (respectively, E^*, Y^*).

Let COMP denote the set of system components. The behavior of each component $C \in$ COMP is modeled by a set of equations, EQ(C). Therefore, EQ = EQ(COMP) denotes the set of equations modeling the whole system. For each equation eq $\in$ EQ, the support Supp(eq) of equation eq is the component C such that eq $\in$ EQ(C). Let Eq denote a set of equations. Supp(Eq) is the support of the composition of equations in Eq. It satisfies

$$\text{Supp}(\text{Eq}) \subset \bigcup_{\text{eq} \in \text{Eq}} \text{Supp}(\text{eq}). \tag{11}$$

The inclusion in (11) may not be strict when the composition of equations in Eq leads to some algebraic simplifications. Sensors $S \subset$ COMP and actuators $A \subset$ COMP are components with specific properties: in accordance with the definition of an equation support, the sensor $S(i)$ is the support of the equation linking $Y^*(i) \in Y^*$ (whose value $y^*(i)$ is unknown) and $Y(i) \in Y$ (whose value $y(i)$ is known because it is the sensor $S(i)$ output). By convention, lower case notation refers to variable values whereas uppercase notation refers to variable symbols. By analogy with sensors, the actuator $A(i)$ is a component whose model links the unknown value $e^*(i)$ of $E^*(i) \in E^*$ and the known value $e(i)$ of $E(i) \in E$.

The causal directed graph that is used to extract relevant tests is $\boldsymbol{\Gamma}(\boldsymbol{N}, \boldsymbol{M})$ where $\boldsymbol{N} =$ OBS* refers to the nodes and $\boldsymbol{M} \subset (N, N)$ stands for the set of arcs. Each arc $M = (N(i), N(j)) \in \boldsymbol{M}$ is directed from $N(i)$ to $N(j)$. The existence of such an arc M means that a value of the variable $N(i)$ is necessary to compute a value of the variable $N(j)$ with the set of equations denoted Eq(M). Eq($\underline{M}$) is extended to a set of arcs $\underline{M} \subset \boldsymbol{M}$ as the union of Eq(M), for each arc M in the set $\underline{M}$.

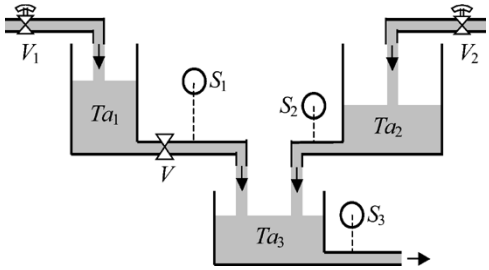


Fig. 1. Three-tanks example.

B. An Example Application

The example in Fig. 1 illustrates the notation used in this section. The process consists of three tanks, pipes, feeding valves, and flow sensors. Tank Ta_1 (respectively, Ta_2) is fed with the input flow $q_{\text{in},1}$ (respectively, $q_{\text{in},2}$) regulated by the valve opening actuator V_1 (respectively, V_2). The output flow $q_{\text{out},1}$ (respectively, $q_{\text{out},2}$) of tank Ta_1 (respectively, Ta_2), feeds tank Ta_3. The output flow of Ta_3 is $q_{\text{out},3}$. Valve V, with a fixed opening, regulates $q_{\text{out},1}$. The output flows of tanks Ta_1, Ta_2, Ta_3 are measured by sensors S_1, S_2, S_3 respectively. The set of components of the three tanks system is COMP

$$\text{COMP} = \{V_1, V_2, V, Ta_1, Ta_2, Ta_3, S_1, S_2, S_3\} \quad (12)$$

$$A = \{V_1, V_2\}, \quad S = \{S_1, S_2, S_3\}. \quad (13)$$

$A \subset \text{COMP}$ and $S \subset \text{COMP}$ are the set of actuators and sensors respectively. An example of a dynamic equation constituting the model of a component is the mass balance for tank Ta_1, expressed in (14) as a difference equation. Equations (14) and (15) constitute EQ($Ta1$)

$$z_1(k+1) = \eta_1 \cdot z_1(k) + \beta_1 \cdot q_{\text{in},1}(k) \quad (14)$$

$$q^*_{\text{out},1}(k) = \kappa_1 \cdot \sqrt{z_1(k)}, \quad q_{\text{in},1}(k) = \phi_1 \cdot p^*_1(k) \quad (15)$$

where, k represents the (discrete) time instant $z_1(k)$, the liquid level at time k, and ϕ_1, the tank intake flow. The model of the valve opening actuator V_1 links the genuine and the known (controlled) opening ratio, $p_1(k)$, as indicated by (16). Similarly, sensor S_1 links the genuine and the known (measured) output flows (17)

$$p^*_1(k) = p_1(k) \quad (16)$$

$$q^*_{\text{out},1}(k) = q_{\text{out},1}(k). \quad (17)$$

The interpretation of (17) is that the measurement of the output flow is equal to its genuine value as long as the sensor S_1 is not faulty. A similar interpretation holds for each sensor in S and each actuator in A.

Additional notations with index 1 replaced by 2 or 3 can be used to model all the components of the three tanks system. From the notational conventions, sets E and Y are particularized as

$$E = \{P_1; P_2\},$$
$$Y = \{Q_{\text{out},1}; Q_{\text{out},2}; Q_{\text{out},3}\},$$
$$\text{OBS} = E \cup Y \quad (18)$$

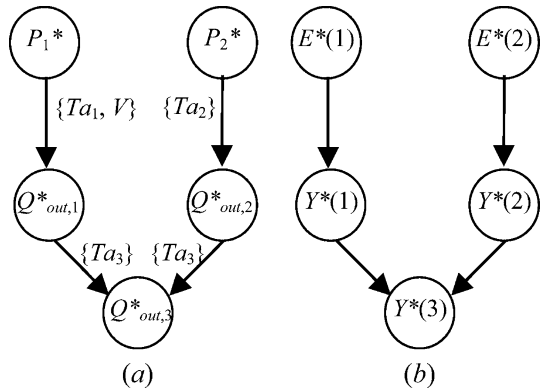


Fig. 2. Causal graph of the three tanks system. (a) Nodes: specific notation. (b) Nodes: generic notation.

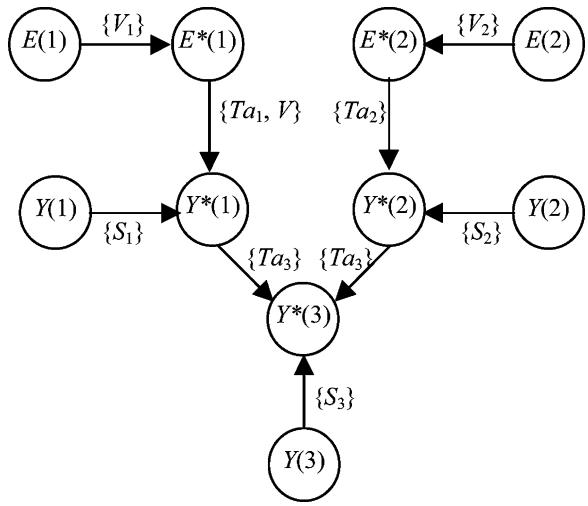


Fig. 3. Augmented causal graph of the three-tanks system.

$$E^* = \{P^*_1; P^*_2\}$$
$$Y^* = \left\{Q^*_{\text{out},1}; Q^*_{\text{out},2}; Q^*_{\text{out},3}\right\}$$
$$\text{OBS}^* = E^* \cup Y^*. \quad (19)$$

Fig. 2 shows the causal graph $\mathbf{\Gamma}$ of the three tanks system. The nodes represent variables describing the causal physical phenomena. Their value can be known using the process instrumentation. The other variables, such as levels for instance, are intermediary variables which are used for computations but not represented in the graph. Fig. 2 also shows the support of the equations related to each arc of the causal graph.

The known variables constituting the set OBS are related to the system instrumentation. Taking these observations into account results in an augmented graph now including sensors and actuators (Fig. 3).

In Fig. 3, the arcs related to the instrumentation are oriented from nodes in OBS toward their corresponding node in OBS* because observations in OBS are the inputs of the fault diagnosis system. The aim of the next paragraphs is to show how the causal graph can be helpful in order to exploit recursively the redundancy between possible estimations of nodes in OBS*.

C. Introduction to the Global and Local Subgraphs

For each node in Y^*, two kinds of subgraphs in $\mathbf{\Gamma}$ will be investigated: *global* and *local*. The aim is to obtain *global* fault diagnosis resulting from recursive *local* reasoning. Each subgraph refers to some equations that can be used to compute estimations of nodes in Y^*. Moreover, sensors and actuators also provide an estimation of nodes in Y^* and E^*. Therefore, several residuals based on the redundancy resulting from two estimations of a single variable in Y^* can be defined. The supports of the equations involved in the computation of those residuals will be studied in order to propose a recursive isolation procedure.

The next two paragraphs introduce some notation in order to define the global and local subgraphs of $\mathbf{\Gamma}$. Let N be a node in $\boldsymbol{N}(\boldsymbol{N} = \text{OBS}^* \text{ for } \mathbf{\Gamma})$. $Par(N) \subset \boldsymbol{N}$ is the set of parents nodes of N. $Anc(N) \subset \boldsymbol{N}$ is the set of ancestors of N, which is recursively defined as $Anc(N) = Par(N) \cup Anc(Par(N))$. The extension of $Par(.)$ (respectively, $Anc(.)$) to a set of nodes $\underline{N} \subset \boldsymbol{N}$ is defined as the union of parents (respectively, ancestors) of each node in $\underline{N}$. $Arcs(\underline{N})$ is the set of arcs of the causal graph $\mathbf{\Gamma}$ linking two nodes in $\underline{N}$.

To understand the notation used in the next paragraphs, it is important to note that $X(i)$ denotes the i^{th} element of the set X, whereas the index notation X_i refers to a set related to the i^{th} global model or to the i^{th} local model. Paragraph G will illustrate the generic notations introduced in paragraphs D, E, and F, by particularizing them to the three tanks example.

D. The Global Subgraphs and Global Residuals

The arcs of the *global* subgraph of $\mathbf{\Gamma}$ related to the node $Y^*(i) \in Y^*$, $i = 1 \dots card(Y^*)$, are those in $\underline{G_i}$ (20). $card(.)$ denotes the cardinal of a set. The name *global* is justified by the fact that all the ancestors of $Y^*(i)$ in $\mathbf{\Gamma}$ are involved

$$\underline{G_i} = Arcs\left(Y^*(i) \cup Anc\left(Y^*(i)\right)\right) \tag{20}$$
$$g_i = \text{Eq}(\underline{G_i}) \tag{21}$$
$$G_i = \text{Supp}(g_i). \tag{22}$$

g_i (respectively, G_i) therefore stands for the equations (respectively, the support) related to $\underline{G_i}$. The structure of the global subsystem related to $Y^*(i)$ can be represented by statement (23)

$$Y_i^* = G_i\left(Eg_i^*\right), \quad Eg_i^* = Anc\left(Y^*(i)\right) \cap E^*. \tag{23}$$

Equation (23) means that an estimation of the variable represented by $Y_i^* = Y^*(i)$ can be calculated from an estimation of the exogenous variables represented by Eg_i^* and the equations g_i whose support is G_i. Let y_i^0 denote the estimation of the genuine value y_i^* of the variable Y_i^* at time k, based on the exogenous variables

$$y_i^0 = g_i(eg_i). \tag{24}$$

y_i^0 is derived from the structure of the global subsystem (23) and the use of eg_i as an estimation of the genuine values eg_i^* of the variables in Eg_i^*. eg_i is a vector representing the value of known exogenous variables in Eg_i. Therefore, the support of (24) is the union of G_i and the actuators linking variables in Eg_i and Eg_i^*

$$\text{Supp}\left(y_i^0\right) = G_i \cup \left(\bigcup_{E^*(j) \in Eg_i^*} A(j)\right), \quad i = 1 \dots card(Y^*). \tag{25}$$

The global residual r_i^0 is based on the redundancy resulting from the estimation of $Y_i^* = Y^*(i)$ by both y_i^0 and y_i (26). y_i is the value provided by the sensor $S(i)$. The support of (26) is thus the union of $S(i)$ and $\text{Supp}(y_i^0)$ (27).

$$r_i^0 = y_i - y_i^0, \quad i = 1 \dots card(Y^*) \tag{26}$$
$$\text{Supp}\left(r_i^0\right) = S(i) \cup \text{Supp}\left(y_i^0\right). \tag{27}$$

E. The Local Subgraphs and Local Residuals

The arcs of the *local* subgraph of $\mathbf{\Gamma}$ related to the node $Y^*(i) \in Y^*$, $i = 1 \dots card(Y^*)$, are those in $\underline{L_i}$ (28). The name *local* is justified by the fact that only the parents of $Y^*(i)$ in $\mathbf{\Gamma}$ are involved (whereas all the ancestors are involved in the global subgraph)

$$\underline{L_i} = Arcs(Y^*(i) \cup Par\left(Y^*(i)\right)) \tag{28}$$
$$l_i = \text{Eq}(\underline{L_i}) \tag{29}$$
$$L_i = \text{Supp}(l_i). \tag{30}$$

l_i (respectively, L_i) thus stands for the equations (respectively, the support) related to $\underline{L_i}$. The structure of the local subsystem related to $Y^*(i)$ can be represented by statement (31)

$$Y_i^* = L_i\left(U_i^*, El_i^*\right), \quad U_i^* = Par\left(Y^*(i)\right) \cap Y^*, \quad El_i^* = Par\left(Y^*(i)\right) \cap E^*. \tag{31}$$

According to (31), computing an estimation of $Y_i^* = Y^*(i)$ from the equations l_i relies on the estimation of U_i^* and El_i^*. el_i is a possible estimation of El_i^* involving the actuators whose model links the variables in El_i and El_i^*. The estimation of the j^{th} variable in U_i^*, denoted $U_i^*(j) \in Y^*$, can be calculated by two different ways. A Boolean configuration vector m defines whether $U_i^*(j)$ is estimated from the global model estimation $u_i^0(j)$ or from the related measurement value $u_i(j)$

$$y_i^m = l_i\left(m \otimes u_i + (\neg m) \otimes u_i^0, el_i\right). \tag{32}$$

The configuration $m = [\dots m(j) \dots]$ is a binary vector having $card(U_i^*)$ elements. Each element is in $\{0; 1\}$. The operator $\neg$ represents the logical negation. The operator $\otimes$ stands for the element by element product of two vectors having the same dimension. In (32), $m(j)$ enables selecting the value that will be used to estimate the j^{th} local model input $U_i^*(j)$ as indicated in Table II.

TABLE II
LOCAL MODEL INPUT CONFIGURATION

$m(j)$	*Estimation of* $U_i^*(j)$:	*Support related to the estimation*
0	$u_i^0(j)$	$Supp(y_k^0)$, k such that $Y^*(k) = U_i^*(j)$ (25)
1	$u_i(j)$	$S(k)$ k such that $Y^*(k) = U_i^*(j)$

Numerical evaluation of y_i^m is thus possible because (32) depends only on known or calculable values. Table II shows that using a measurement to estimate a local model input enables local reasoning (i.e. cutting the influence of propagated faults) whereas using the global model estimation focuses on the faults propagated specifically by that input. Moreover, it should be noted that the values of y_i^0 given by (24) and by (32) when $m = 0\,(\forall j, m(j) = 0)$ are the same.

A first set of local residuals is based on the redundancy resulting from the estimation of $Y_i^* = Y^*(i)$ by both y_i^m and y_i (33). y_i is the known value provided by the sensor $S(i)$. The support of (33), depending on the local input configuration m is given by (34) to (36), where $U_{i,m}^* = \{U_i^*(j) \in U_i^*/m(j) = 1\}$ and $U_{i,\neg m}^* = \{U_i^*(j) \in U_i^*/m(j) = 0\}$

$$r_i^m = y_i - y_i^m, \quad i = 1 \ldots card(Y^*) \tag{33}$$

$$\begin{aligned}\text{Supp}\,(r_i^m) = &\,\text{Local}\,(Y_i^*, m)\\ &\cup \text{Upstream}\,(Y_i^*, m)\end{aligned} \tag{34}$$

$$\begin{aligned}\text{Local}\,(Y_i^*, m) = &\, S(i) \cup L_i \cup \left(\bigcup_{E^*(j) \in El_i^*} A(j) \right)\\ &\cup \left(\bigcup_{Y^*(j) \in U_{i,m}^*} S(j) \right)\end{aligned} \tag{35}$$

$$\text{Upstream}\,(Y_i^*, m) = \bigcup_{Y^*(j) \in U_{i,\neg m}^*} \text{Supp}\left(y_j^0\right). \tag{36}$$

Two particular cases can be used to illustrate the local input configuration and its consequence in terms of support (34).

Case 1: $m = 0\ (\forall j, m(j) = 0)$.

$$\begin{aligned}\text{Supp}\left(r_i^0\right) = &\,\text{Local}\,(Y_i^*, 0)\\ &\cup \text{Upstream}\,(Y_i^*, 0)\end{aligned} \tag{37}$$

$$\text{Local}\,(Y_i^*, 0) = S(i) \cup L_i \cup \left(\bigcup_{E^*(j) \in El_i^*} A(j) \right) \tag{38}$$

$$\text{Upstream}\,(Y_i^*, 0) = \bigcup_{Y^*(j) \in U_i^*} \text{Supp}\left(y_j^0\right). \tag{39}$$

It is worth noting that the last term in (37) refers to components whose faulty behavior is subject to influence r_i^0 through an influence on the local input variables in U_i^*. In addition to faults propagated through U_i^*, r_i^0 is also sensitive to faults whose origin is local.

Case 2: $m = 1\ (\forall j, m(j) = 1)$

$$\text{Supp}\left(r_i^1\right) = \text{Local}\,(Y_i^*, 1) \tag{40}$$

$$\text{Local}\,(Y_i^*, 1) = \text{Local}\,(Y_i^*, 0) \cup \left(\bigcup_{Y^*(j) \in U_i^*} S(j) \right) \tag{41}$$

$$\text{Upstream}\,(Y_i^*, 1) = \varnothing. \tag{42}$$

The local model input configuration $m = 1$ makes r_i^1 insensitive to faults propagated through U_i^*. r_i^1 is only sensitive to faults whose origin is local and to faults occurring in sensors related to the local model inputs in U_i^*. Therefore, the Boolean configuration vector m enables selection of the fault propagation paths in the causal graph to which residual r_i^m is sensitive.

A second set of local residuals is based on the redundancy resulting from the estimation of $Y_i^* = Y^*(i)$ by both y_i^1 and y_i^m

$$\underline{r}_i^m = y_i^1 - y_i^m, \quad i = 1 \ldots card(Y^*) \tag{43}$$

$$\begin{aligned}\underline{r}_i^m = &\, l_i\,(m \otimes u_i + (\neg m) \otimes u_i, el_i)\\ &- l_i\left(m \otimes u_i + (\neg m) \otimes u_i^0, el_i\right).\end{aligned} \tag{44}$$

The support (46) of $\underline{r}_i^m$ is then deduced from (45)

$$(\neg m) \otimes u_i = (\neg m) \otimes u_i^0 \Rightarrow \underline{r}_i^m = 0 \tag{45}$$

$$\text{Supp}\,(\underline{r}_i^m) = \left(\bigcup_{Y^*(j) \in U_{i,\neg m}^*} \text{Supp}\left(r_j^0\right) \right) \tag{46}$$

$$\begin{aligned}\text{Supp}\,(\underline{r}_i^m) = &\left(\bigcup_{Y^*(j) \in U_{i,\neg m}^*} S(j) \right)\\ &\cup \text{Upstream}\,(Y_i^*, m).\end{aligned} \tag{47}$$

Unlike r_i^m, $\underline{r}_i^m$ cannot be influenced by the local faults. Moreover, when $m = 0$, $\underline{r}_i^0$ is influenced by all the upstream faults propagated through all the inputs in U_i^* of the local model related to Y_i^*. $\underline{r}_i^0$ is also sensitive to faults occurring in the sensors related to inputs in U_i^*.

It is also worth noting that, once the global estimates y_i^0 are computed for each node in Y^*, the computation of any residual like r_i^m or $\underline{r}_i^m$ only involves local equations l_i (32), (33), (43). Moreover, all the global estimates can be calculated by a single simulation involving all the equations in EQ(COMP $\setminus S$). Consequently, based on the global estimates, each residual like r_i^m or $\underline{r}_i^m$ can be calculated very quickly. This makes the proposed approach attractive in order to satisfy the requirements of a real-time implementation.

F. Residual and Test Isolation Properties

Let r denote a residual such as r_i^m or $\underline{r}_i^m$, for instance. Supp(r) is the support of the equations necessary for computing r. r is never exactly null, even in the case of fault-free behavior: it may be influenced by deviations due to noise, modeling errors or disturbances. A threshold λ is thus chosen such that the test $T \Leftrightarrow (r > \lambda)$ implies that the system's behavior is faulty. As r is only sensitive to faults occurring in the components of its support, it follows that when test T is true, at least one component in Supp(r) can be suspected. The conflict set related to the test T is thus the support of r. This can be particularized to the residuals r_i^m and $\underline{r}_i^m$

$$\text{Conflict}\,(T_i^m) = \text{Supp}\,(r_i^m), \quad \text{Conflict}\,(\underline{T}_i^m) = \text{Supp}\,(\underline{r}_i^m). \tag{48}$$

G. Application to the Example

In the case of the three-tanks system, the supports used to define conflict sets are given in Table III. The lines L_1 to G_3 directly result from the application of (22) and (30) to the causal graph in Fig. 2. The lines describing the *local* and *upstream* components of a given node thus result from equations (38), (39) and (35), (36).

TABLE III
SUPPORTS RELATED TO THE CAUSAL GRAPH

Components	V_1	Ta_1	V	S_1	V_2	Ta_2	S_2	Ta_3	S_3
L_1		X	X						
G_1		X	X						
L_2						X			
G_2						X			
L_3								X	
G_3		X	X			X		X	
Local(Y_1^*,0)	X	X	X	X					
Upstream(Y_1^*,0)									
Local(Y_2^*,0)					X	X	X		
Upstream(Y_2^*,0)									
Local(Y_3^*,0)								X	X
Upstream(Y_3^*,0)	X	X	X		X	X			

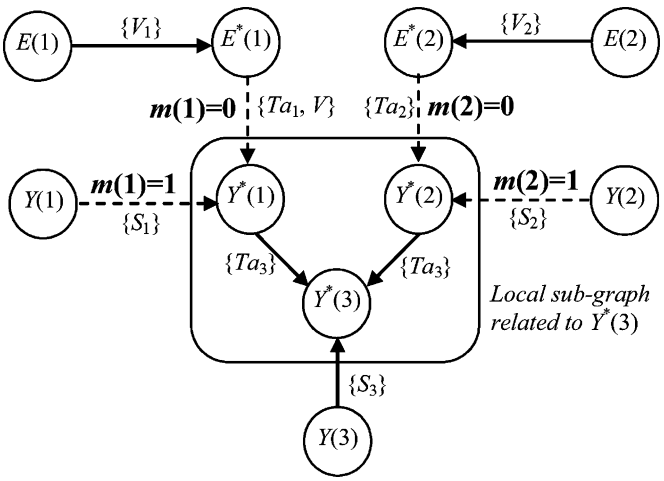


Fig. 4. Local model input configuration related to node Y^* (3).

TABLE IV
CONFLICT SETS OF THE TESTS RELATED TO Y^* (3)

Components	V_1	Ta_1	V	S_1	V_2	Ta_2	S_2	Ta_3	S_3
T_1^0	X	X	X	X					
T_1^1	X	X	X	X					
T_2^0					X	X	X		
T_2^1					X	X	X		
$T_3^{[0\,0]} = T_3^0$	X	X	X		X	X		X	X
$T_3^{[0\,1]}$	X	X	X				X	X	X
$T_3^{[1\,0]}$				X	X	X		X	X
$T_3^{[1\,1]} = T_3^1$				X			X	X	X
$\underline{T}_3^{[0\,0]} = \underline{T}_3^0$	X	X	X	X	X	X	X		
$\underline{T}_3^{[0\,1]}$	X	X	X	X					
$\underline{T}_3^{[1\,0]}$					X	X	X		

In order to illustrate the impact of the local model input configuration m on the conflict sets related to the tests, we shall examine node Y^* (3) in Fig. 2. The causal graph (Fig. 2), as well as the augmented graph (Fig. 3), are shown in Fig. 4. The rounded rectangle in Fig. 4 highlights the local subgraph related to Y^* (3). Fig. 4 illustrates local input configurations. For $j = 1, 2$ if $m(j) = 0$ then an arc links $Par(Y^*(j)) = E^*(j)$ to $Y^*(j)$, otherwise an arc links $Y(j)$ to $Y^*(j)$. When $m(j) = 0$, the estimation of Y^* (3) depends on the faults that have an influence on $Y^*(j)$ whereas when $m(j) = 1$, the estimation of Y^* [see (3)] depends on the faults affecting S_j (and does not depend on the faults propagated through $Y^*(j)$).

The ability of local model input configurations to select fault propagation paths is detailed in Table IV. Table IV is obtained from the conflict sets related to tests and from the supports in Table III.

In order to emphasize the links between Fig. 4 and Table IV, the calculation paths and the conflict sets related to the tests $T_3^{[10]}$ and T_3^1 are illustrated in Fig. 5.

V. RECURSIVE ISOLATION

A. Isolation Algorithm

Using the previous residuals and their isolation properties, we shall now describe a recursive isolation algorithm. It is applied each time new data are acquired on the system to be diagnosed. The interest of recursive isolation is to prevent combinatorial explosion. Thus, systematic calculation of all the possible residuals that are mentioned in Section IV is excluded, as it could take a too long when applied to a complex industrial process. The objective of causal diagnosis is to search for the source node(s) whose state(s) explain all the observed deviations in the graph. Once a source node is found, it has to be interpreted in terms of possible faulty components. The set of components Up_i, IS_i and Loc_i are thus defined in (49), (50), and (51), in accordance with the notations of Section IV.

$$\mathrm{Up}_i = \mathrm{Upstream}\,(Y_i^*, 0) \tag{49}$$

$$IS_i = \bigcup_{Y^*(j) \in U_i^*} S(j) \tag{50}$$

$$\mathrm{Loc}_i = \mathrm{Local}\,(Y_i^*, 0)\,. \tag{51}$$

Up_i and Loc_i, respectivel, refer to upstream and local components with respect to node $Y_i^* = Y^*(i)$. IS_i is the set of sensors related to U_i^*, the measured input nodes of the local subgraph related to Y_i^*. Introducing such partitions within the components in COMP will be shown to be relevant in order to implement recursive isolation.

The proposed algorithm can be divided into two steps at each time sample. The execution of the second step is conditioned by the result of the first one, which is outlined in Table V and explained below.

1) Step 1: Detection and First Isolation Level: At each sample time, for each node $Y^*(i) \in Y^*$, the tests T_i^1 and $\underline{T}_i^0$ are computed. According to (40) and (41) (respectively, (47)), the conflict sets of T_i^1 (respectively, $\underline{T}_i^0$) can be expressed as

$$\mathrm{Conflict}\left(T_i^1\right) = IS_i \cup Loc_i \tag{52}$$

$$\mathrm{Conflict}\left(\underline{T}_i^0\right) = IS_i \cup Up_i. \tag{53}$$

At each sample time, for each node $Y^*(i) \in Y^*$, a local diagnosis reasoning based on the tests T_i^1 and $\underline{T}_i^0$ is done. Table VI shows the (local) minimal diagnosis resulting from the test values.

The tests T_i^1 and $\underline{T}_i^0$ enable fault detection and a first isolation level determining whether the source node is local and/or upstream with respect to $Y^*(i)$. Moreover, the link with suspect components can be made explicit according to the definitions of Up_i, IS_i, and Loc_i. However, unlike IS_i and Loc_i, the cardinal of Up_i may be very large. That is the reason why Up_i is

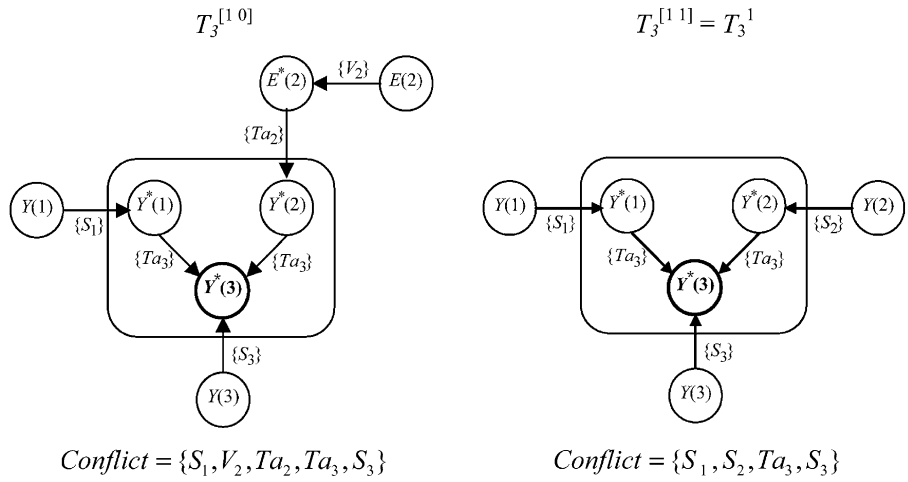


Fig. 5. Conflict sets of the tests $T_3^{[1\,0]}$ and T_3^1.

TABLE V
ISOLATION PROPERTIES OF THE RESIDUAL TESTS USED IN THE RECURSIVE ISOLATION ALGORITHM

	Up_i	IS_i	Loc_i	
T_i^1		X	X	Step 1 : Detection and First isolation level
$\underline{T}_i^0$	X	X		
…	X			Step 2 : Second isolation level (Selection of upstream paths + recursive calls)
$\underline{T}_i^m$	X			
…		X		

TABLE VI
FIRST ISOLATION LEVEL: LOCAL DIAGNOSIS REASONING

Tests value	(Local) minimal diagnosis
$(\underline{T}_i^0, T_i^1) = (0,0)$	{ } *(no fault detected)*
$(\underline{T}_i^0, T_i^1) = (0,1)$	{ $\{IS_i\}$, $\{Loc_i\}$ }
$(\underline{T}_i^0, T_i^1) = (1,0)$	{ $\{Up_i\}$, $\{IS_i\}$ }
$(\underline{T}_i^0, T_i^1) = (1,1)$	{ $\{IS_i\}$, $\{Up_i, Loc_i\}$ }

never made explicit (i.e. its components are never described in extension), but the diagnosis is refined through a systematic and recursive search for the propagation paths. This is the aim of the second isolation level.

2) Step 2: Second Isolation Level: Each time Up_i belongs to one of the diagnosis computed at step 1, the algorithm corresponding to the second isolation level (Table VII) is run. This algorithm determines the local input(s) that have propagated the deviations observed from $Y^*(i)$ in order to go backward in the causal graph toward the source node(s). The residuals computed from the local input configurations are used to this purpose. Subsequently, the search for the fault sources is pursued in the so-determined upstream directions through recursive calls to step 1 (Table VII).

B. Algorithm Properties and Logical Assumptions

All the variable nodes such that a fault has been locally detected at step 1, reflect an abnormal behavior with respect to their reference value (value that is estimated from the actuator values). This abnormal behavior is not necessarily due to a fault that is local to this node. It may well be due to the propagation of another fault. Nevertheless, informing the operator about this

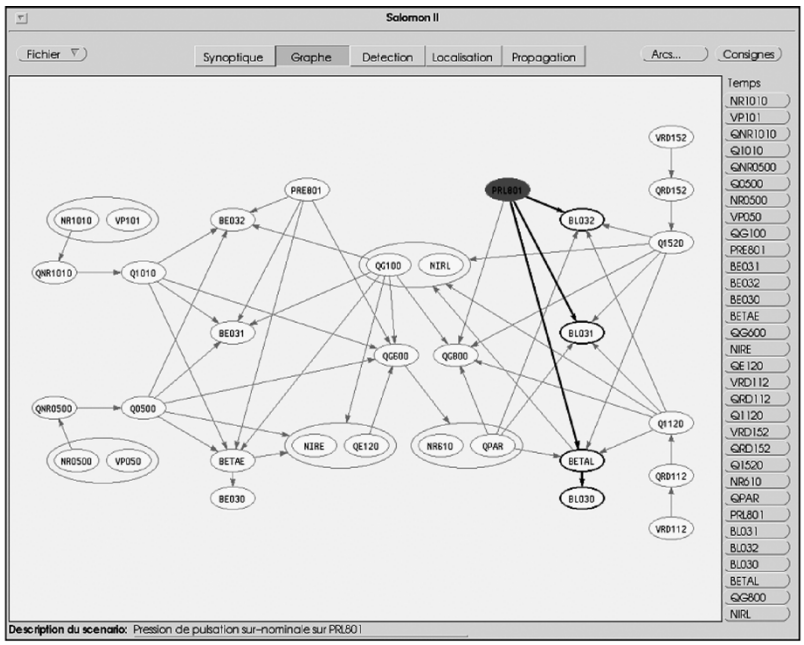
TABLE VII
ALGORITHM FOR PATH SELECTION AND RECURSIVE CALLS

```
For j = 1...card(U_i^*)
    m = [1...1]
    m(j) = 0      /* Select the j^th propagation path */
    If T_i^m = 1 Then
        /* The node U_i^*(j) is suspected; it belongs to the
           propagation path of the deviations: recursive call */
        Go to Step 1 with Y_i^* = U_i^*(j)
    Else
        /* Search for the source not pursued in the upstream
           direction with respect to U_i(j): local exoneration */
    End If
End For
```

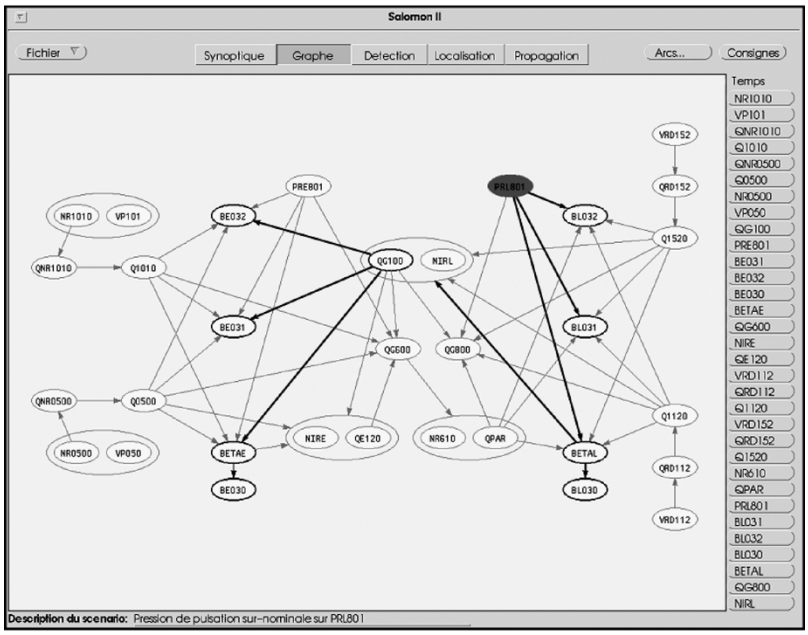
abnormal behavior allows him/her to prepare corrective actions. Enlightening the corresponding node in the causal graph provides a simple means to make this information available through an interface (Fig. 6).

The recursive fault isolation proposed in this paper, which is based on the process causal knowledge, enables on-line determination and evaluation only of the residuals required for fault isolation reasoning. That is an important difference with the FDI approach, where the size of the signature vector is fixed *a priori*: all the ARRs have to be evaluated at each measurement acquisition step. Another important difference is that the causal diagnostic algorithm does not require any assumption about the way single fault effects may be combined to tackle the multiple fault case.

The logical assumptions on which the algorithm is based are now summarized. On the one hand, the test values at time k, $T_i^1(k)$ and $\underline{T}_i^0(k)$ are only used when equal to 1. However, when both $T_i^1(k)$ and $\underline{T}_i^0(k)$ are null, the diagnosis related to $Y^*(i)$ stops. This corresponds to a local exoneration. Notice that components related specifically to this node could be suspected thanks to other tests related to other nodes. Thus, exoneration is not definitive, but local. On the other hand, the test value $\underline{T}_i^m(k) = 1 (m \neq \vec{0})$ of the residual $\underline{r}_i^m(k)$ leads to the parent on the related path being suspected. The backward diagnostic analysis is only carried out on the set of suspected parents. All the other parents of $Y^*(i)$ are exonerated at this step of the



Time: 00:03:00



Time: 00:31:00

Fig. 6. Fault propagation subgraph evolution with time.

algorithm but not definitively. The exoneration associated to $\underline{T}_i^m(k) = 0$ is merely local. Each element $U_i^*(j)$ of the nonsuspected set can be detected as abnormal when evaluated through its own tests (execution of step 1 for node $U_i^*(j)$) or suspected in another backward diagnostic analysis from any other downstream variable.

Local exoneration is a weak assumption that allows complex installations to be diagnosed using algorithms compatible with real time requirements.

C. Time Management

On-line fault diagnosis of an industrial process requires time management. Fault propagation in the process takes time, due to the dynamics of the process. Thus, the consequences on other variables of a single source fault can appear at different time instants. Dynamic monitoring of the effects of the primary fault is necessary to ensure continuous assessment of the disturbed functions; revising the initial decision if necessary or relating them if appropriate to the problem identified earlier. This is known as fault filtering or progressive monitoring. This point is generally disregarded in standard FDI or DX systems, where the objectives are instead early detection and isolation. This problem is addressed qualitatively in [40] and quantitatively by the algorithm proposed in this section.

Once a propagation subgraph has been identified, any subsequent abnormal deviations will be tested, using the same

consistency tests as those described in Section IV and V-A. This is done in order to determine whether they correspond to the occurrence of a new fault, or whether they are only the consequences of faults previously detected and accounted for. Fault effect propagation will lead to new simulation errors (global residuals). Related variables will become the new terminal nodes of the fault propagation subgraph. It is worth noting that the isolation algorithm proposed in this paper does not consist simply in linking variables that have been detected to be faulty and that are related by an arc in the causal graph. It relies on consistency tests taking advantage of the dynamic properties of the arcs. It tests recursively every parent of a detected variable, until a local fault has been proven. In this way, a source fault that would have been corrected by the operator but whose effects would be still propagating due to long delays would still be considered as the explanation of the faulty behavior.

The propagation subgraph evolves with time as the effects of the fault propagate dynamically. Dynamic monitoring allows the propagation of fault consequences in the process to be explained continuously by the same source. Dynamic fault signature recognition can thus be included in the operating tools in control rooms. Displaying the fault propagation subgraph in the control interface provides an additional explanation that is much appreciated by operators [20], [21]. Fig. 6 shows two subsequent views of such an interface, in which the time evolution of the fault propagation subgraph can be seen.

VI. Conclusion

This paper presents a method for model-based diagnosis devoted to on-line supervision of complex processes involving human operators. In this context, the intelligibility and pertinence to the operator of the results provided by a diagnostic system become legitimate issues, because the results of the diagnostic system become part of the operator's reasoning.

The proposed diagnostic method is based on the interaction between artificial intelligence and control techniques. Standard FDI approaches are augmented with a causal-graph representation of the physical process. The causal graph enables decomposition of the complex system to be diagnosed into subsystems that are represented by elementary relations between variables. Inference is performed locally on each subsystem. The causal structure enables reducing the diagnostic computational complexity.

At the local level, FDI techniques based on numerical residual generation and analysis can be exploited. The method gains from the precision of control methods for representing the local submodels. Various numerical simulation possibilities of these submodels are studied and shown to generate residuals appropriate for fault isolation. It is simple to compare these simulation results with numerical data acquired from the process, and their precision is easily quantified. Sensitivity with respect to uncontrollable phenomena such as noise or nonmeasurable disturbances can be evaluated as well as the influence of model parameter uncertainty. Process dynamics are taken into account using relations between variables that manage time explicitly. Fault filtering and fault propagation monitoring thus occur naturally. The residual supports were studied in order to link this method with DX component-oriented reasoning. A drawback of this numerical approach is that the model parameters have to be rather precise. It could be necessary for them to be re-identified, from time to time, in order to update the model.

On the global level, diagnostic reasoning is supported by the causal structure. This structure, which was proposed in the framework of qualitative modeling, is very general and enables diagnostic inference to be clearly separated from process representation. Hence, the model can be easily modified if some parts of the process evolve, which is quite common in industry. Another important consequence is that the causal decomposition of the model supports diagnostic explanation, which is relevant when diagnosis is intended for on-line process supervision. Finally, a result of this modeling approach is that several assumptions, which are implicit in the FDI formulation and which are clearly explained within the DX consistency-based diagnostic approach, can be overcome. No logical assumption about multiple fault occurrence and component global exoneration is necessary a priori. From a theoretical point of view the approach cannot be considered as logically sound, because local exoneration is necessary in order to simplify the diagnostic computational aspects and avoid combinatorial explosion, which generally results from a purely logical approach. However, this limit is much less restrictive than the classical exoneration assumption. First, a precise quantitative analysis of the variable is used and not a mere Boolean analysis of the parents' influences. Then, when a faulty variable is wrongly exonerated in the backward diagnostic analysis carried out on the set of suspected parents, this means that: 1) no fault has been detected on this variable in the detection step (the fault consequences are thus not significant on this variable) and 2) this variable can be reanalyzed in another backward diagnostic analysis, through another path in the graph, for the same original fault to be explained, depending on the graph structure.

This work could be extended using FDI knowledge when it is possible to model faults and disturbances. The experience of the FDI community in input decoupling could be used to generate residuals locally ensuring a good compromise between sensitivity to faults and robustness with respect to disturbances.

Another possible extension consists in envisaging changes in the process structure. Causal process representation seems to be well suited to automatic model construction, including structural changes. The diagnostic method proposed in this paper could thus be extended to the diagnosis of hybrid systems.

To summarize, the FDI and DX communities use very different models: numerical models within the FDI environment as opposed to abstract multilevel models within the DX environment. The diagnostic method described in this paper presents a unified framework inspired by both approaches and combines tools from both communities. Consequently, it can evolve, gaining from the progress made in both fields.

References

[1] H. Ahriz and S. Xia, "Automatic modeling for diagnosis," in *Proc. 11th Int. Workshop on Qualitative Reasoning*, 1997, pp. 3–12.

[2] M. Basseville and I. Nikiforov, *Detection of Abrupt Changes*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[3] E. Charmiak, "Bayesian networks without tears," in *Proc. American Association for Artificial Intelligence*, Anaheim, CA, 1991.

[4] L. Console, D. Theseider Dupré, and P. Torasso, "A theory of diagnosis for incomplete causal models," in *Proc. 11th Int. Joint Conf. Artificial Intelligence*, 1989, pp. 1311–1317.

[5] ——, "On the relationship between abduction and deduction," *J. Logic Comput.*, vol. 1, no. 5, pp. 661–690, 1991.

[6] L. Console and P. Torasso, "An approach to the compilation of operational knowledge from causal models," *IEEE Trans. Syst., Man, Cybern.*, vol. 22, pp. 772–789, Nov./Dec. 1992.

[7] L. Console, L. Portinale, and D. Theseider Dupré, "Using compiled knowledge to guide and focus abductive diagnosis," *IEEE Trans. Knowl. Data Eng.*, vol. 8, pp. 690–706, May 1996.

[8] M. O. Cordier, P. Dague, M. Dumas, F. Levy, J. Montmain, M. Staroswiecki, and L. Trave-Massuyes, "AI and automatic control approaches of model based diagnosis: links and underlying hypotheses," in *Proc. 4th IFAC Safeprocess on Fault Detection, Supervision and Safety for Technical Processes*, 2000, pp. 274–279.

[9] M. O. Cordier, P. Dague, F. Levy, M. Dumas, J. Montmain, M. Staroswiecki, and L. Trave-Massuyes, "A comparative analysis of AI and control theory approaches to model-based diagnosis," in *Proc. ECAI'2000*, 2000, pp. 136–140.

[10] P. T. Cox and T. Pietrzykowski, "General diagnosis by abductive inference," in *Proc. IEEE Symp. Logic Programming*, 1987, pp. 183–189.

[11] R. Davis, "Diagnosis via causal reasoning: paths of interaction and the locality principle," in *Proc. Amer. Ass. Artificial Intelligence*, 1983, pp. 88–94.

[12] P. Dague and B. Dubuisson, *Diagnostic par Intelligence Artificielle et Reconnaissance des Formes*. Paris, France: Hermès, 2001.

[13] A. Darwiche, "Model-based diagnosis using causal networks," in *Proc. Int. Joint Conf. Artificial Intelligence*, 1995, pp. 211–217.

[14] ——, "Model-based diagnosis using structured system descriptions," *J. AI Res.*, vol. 8, pp. 165–222, 1998.

[15] R. Dechter and A. Dechter, "Structure-driven algorithms for truth maintenance," *Artif. Intell.*, vol. 82, pp. 1–20, 1996.

[16] J. de Kleer and J. S. Brown, "A qualitative physics based on confluences," *Artific. Intell.*, vol. 24, pp. 7–83, 1984.

[17] ——, "Theories of causal ordering," *Artif. Intell.*, vol. 29, no. 1, pp. 33–62, 1986.

[18] J. de Kleer, A. Mackworth, and R. Reiter, "Characterizing diagnoses and systems," *Artif. Intell.*, vol. 56, no. 2–3, pp. 197–222, 1992.

[19] S. Ding, E. Ding, and T. Jeinsch, "An approach to analysis and design of observer and parity relation based FDI system," in *Proc. IFAC World Congr.*, Beijing, China, 1999.

[20] A. Evsukoff, J. Montmain, and S. Gentil, "Causal model based supervising and training," in *Proc. IFAC Workshop on Line Fault Detection and Supervision in the Chemical Industries*, Lyon, France, 1998.

[21] A. Evsukoff, S. Gentil, and J. Montmain, "Fuzzy reasoning in co-operative supervision systems," *Contr. Eng. Pract.*, vol. 8, pp. 389–407, 2000.

[22] P. Frank, "Analytical and qualitative model-based fault diagnosis, a survey and some new results," *Eur. J. Control*, vol. 1, no. 2, pp. 6–28, 1996.

[23] P. Frank and S. Ding, "Current development in the theory of FDI," in *Proc. IFAC Safeprocess on Fault Detection, Supervision and Safety for Technical Processes*, 2000, pp. 16–27.

[24] H. Geffner and J. Pearl, "An improved constraint-propagation algorithm for diagnosis," in *Proc. Int. Joint Conf. Artificial Intelligence*, 1987, pp. 1105–1111.

[25] A.-L. Gehin, M. Assas, and M. Staroswiecki, "Structural analysis of system reconfigurability," in *Proc. IFAC Safeprocess on Fault Detection, Supervision and Safety for Technical Processes*, 2000, pp. 292–297.

[26] D. Heckerman, "A tutorial on learning with Bayesian networks," in *Proc. NATO Advanced Study Institute on Learning in Graphical Models*, M. I. Jordan, Ed. Norwell, MA: Kluwer, 2000, pp. 301–354.

[27] B. Heim, S. Gentil, S. Cauvin, L. Travé-Massuyès, and B. Braunschweig, "Fault diagnosis of a chemical process using causal uncertain model," in *Proc. Eur. Conf. Artificial Intelligence ECAI 2002*, Lyon, France, 2002.

[28] M. Iri, K. Aoki, E. O'Shima, and H. Matsuyama, "Graphical approach to the problem of locating the origin of the system failure," *J. Oper. Res. Soc. Jpn.*, vol. 23, no. 4, pp. 295–312, 1980.

[29] R. Isermann, "Supervision, fault-detection and fault-diagnosis methods—An introduction," *Contr. Eng. Pract.*, vol. 5, no. 5, pp. 639–652, 1997.

[30] R. Isermann and P. Ballé, "Trends in the application of model-based fault detection and diagnosis of technical processes," *Contr. Eng. Pract.*, vol. 5, no. 5, pp. 709–719, 1997.

[31] Y. Iwasaki and H. A. Simon, "Causality in device behavior," *Artif. Intell.*, vol. 29, no. 1, pp. 3–33, 1986.

[32] L. Leyval, S. Gentil, and S. Feray-Beaumont, "Model based causal reasoning for process supervision," *Automatica*, vol. 30, no. 8, pp. 1295–1306, 1994.

[33] A. Ligeza and B. Górny, "Systematic conflict generation in model based diagnosis," in *Proc. IFAC Safeprocess on Fault Detection, Supervision and Safety for Technical Processes*, 2000, pp. 1103–1108.

[34] L. Ljung, *System Identification: Theory for the User*, ser. Information and System Science Series. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[35] P. J. F. Lucas, "Analysis of notions of diagnosis," *Artif. Intell.*, vol. 105, pp. 295–343, 1998.

[36] J. Montmain, L. Leyval, and S. Gentil, "Qualitative analysis for decision making in supervision of industrial continuous processes," *Math. Comput. Simul.*, vol. 36, pp. 149–163, 1994.

[37] J. Montmain, "Supervision applied to nuclear fuel reprocessing," *AI Commun.*, vol. 13, pp. 61–81, 2000.

[38] J. Montmain and S. Gentil, "Dynamic causal model diagnostic reasoning for on-line technical process supervision," *Automatica*, vol. 36, pp. 1137–1152, 2000.

[39] P. J. Mosterman, G. Biswas, and S. Narasimham, "Measurement selection and diagnosability of complex physical systems," in *Proc. 8th Int. Workshop on Principles of Diagnosis, DX'97*, 1997, pp. 79–86.

[40] P. J. Mosterman and G. Biswas, "Diagnosis of continuous valued systems in transient operating regions," *IEEE Trans. Syst., Man, Cybern. A*, vol. 29, pp. 554–565, Nov. 1999.

[41] M. Nyberg, "A general framework for fault diagnosis based on hypothesis testing," in *Proc. DX'00*, Morelia, Michoacan, Mexico, 2000.

[42] B. Palowitch and M. Kramer, "The application of a knowledge based expert system to chemical plant fault diagnosis," in *Proc. Amer. Control Conf.*, 1986, pp. 646–651.

[43] M. Paynter, *Analysis and Design of Engineering Systems*. Cambridge, MA: MIT Press, 1961.

[44] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. San Mateo, CA: Morgan Kaufmann, 1988.

[45] D. Poole, "Normality and faults in logic-based diagnosis," in *Proc. Int. Joint Conf. Artificial Intelligence*, 1989, pp. 1304–1310.

[46] ——, "Representing diagnosis knowledge," *Ann. Math. Artif. Intell.*, vol. 11, pp. 33–50, 1994.

[47] N. Porté, S. Boucheron, S. Sallantin, and F. Arlabosse, "An algorithmic view at causal ordering," in *Proc. 2nd International Workshop on Qualitative Physics QR'88*, Paris, France, 1988.

[48] R. Reiter, "A theory of diagnosis from first principles," *Artif. Intell.*, vol. 32, pp. 57–95, 1987.

[49] J. Shiozaki, H. Matsuyama, K. Tano, and E. O'Shima, "Fault diagnosis of chemical processes by the use of signed, directed graphs: Extension to five-range patterns of abnormality," *Int. Chem. Eng.*, vol. 25, no. 4, pp. 651–659, 1985.

[50] M. Staroswiecki and M. Deckerck, "Analytical redundancy in nonlinear interconnected systems by means of structural analysis," in *Proc. IFAC Symp. Advanced Information Processing in Automatic Control*, 1989, pp. 23–27.

[51] P. Struss, "AI methods for model-based diagnosis," in *12th International Workshop Principles Diagnosis DX01-Bridge Workshop*, Via Lattea, Italy, 2001.

[52] L. Travé-Massuyès and R. Milne, "Diagnosis of dynamic systems based on explicit and implicit behavioral models: An application to gas turbines in esprit project tiger," *Appl. Artif. Intell. J.*, 1996.

[53] C. Yu and C. Lee, "Fault diagnosis based on qualitative/quantitative process knowledge," *AIChE J.*, vol. 37, no. 4, pp. 617–627, 1991.