# A model integration of DX an FDI techniques for automatic determination of minimal diagnosis

**R. Ceballos, M.T. Gómez-López, R. M. Gasca and C. Del Valle**

Computer Engineering Superior Technical School of Seville, Spain

## Abstract

Two communities work in parallel in model-based diagnosis: FDI and DX. In this work an integration of the FDI and the DX communities is proposed. Only relevant information for the identification of the minimal diagnosis is used. In the first step, the system is divided into clusters of components, and each cluster is separated into nodes. The minimal and necessary set of contexts is then obtained for each cluster. These two steps automatically reduce the computational complexity since only the essential contexts are generated. In the last step, a signature matrix and a set of rules are used in order to obtain the minimal diagnosis. The evaluation of the signature matrix is on-line, the rest of the process is totally off-line.

## 1 Introduction

Diagnosis allows us to determine why a correctly designed system does not work as expected. Diagnosis is based on a set of integrated sensors which obtain a set of observations. The aim of diagnosis is to detect and identify the reason for any unexpected behaviour, and to isolate the parts which fail in a system. The behaviour of components is stored by using constraints. Inputs and outputs of components are represented as variables of the component constraints. These variables can be observable and non-observable depending on the sensors allocation of the sensors.

Two communities work in parallel, although separately, in model-based diagnosis: FDI (from Automatic Control) and DX (from Artificial Intelligence). Nevertheless, the integration of FDI with DX theories has been shown in recent work [Cordier *et al.*, 2000],[Ceballos *et al.*, 2004]. In the DX community, [Reiter, 1987], [de Kleer *et al.*, 1992] presented the diagnosis formalization. A general theory was proposed to explain the discrepancies between the observed and the correct behaviour by using a logical-based diagnosis process. In the FDI community, [Staroswiecki and Declerk, 1989] and [Cassar and Staroswiecki, 1997] presented the formalization of structural analysis, the process to obtain the ARRs (Analytical Redundancy Relation) of the system.

In this work an integration of FDI theories with the DX community is proposed, in order to improve the minimal di-agnosis determination. This integration has three phases. The structural pre-treatment in the first phase and the reduction of the model in the second phase enables the improvement of the computational complexity. The minimal diagnosis is obtained by applying an observational model to a signature matrix together with a set of precompiled rules. The evaluation of the signature matrix is on-line, however the rest of the process is totally off-line.

Our paper has been organized as follows. First, definitions and notations are established in order to clarify concepts. Section 3 shows two examples of the validation of this approach. Section 4 describes the advantages of the structural pretreatment. After that, in section 5, the process for the definition of the context network is explained. Section 6 describes the determination of the minimal diagnosis. Finally, conclusions are drawn and future work is outlined.

## 2 Definitions and notation

In order to clarify the diagnosis process some definitions must be established.

**Definition 2.1** *System Model*: A finite set of polynomial equality constraints ($P$) which determine the system behaviour. This is done by means of the relations between non-observable ($V_i$) and observable variables (sensors) of the system ($O_j$).

**Definition 2.2** *Observational Model*: A tuple of values for the observable variables.

**Definition 2.3** *Context*: A collection of components of the system, and their associated constraints. The number of possible contexts is $2^{nComp}$ - 1, where *nComp* is the number of components of the system.

**Definition 2.4** *Context Network*: A graph formed by all the contexts of the system in accordance with to the way proposed by ATMS[de Kleer, 1986]. The context network has a natural structure of a directed graph for set inclusion.

**Definition 2.5** *Diagnosis Problem*: A tuple formed by a system model and an observational model. The solution of this problem is a set of possible failed components.

## 3 Examples

The following examples are used in order to explain our methodology.
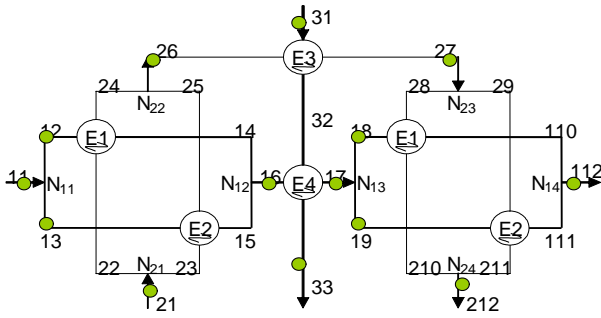
Figure 1: The example of heat exchangers

- A system of heat exchangers: As proposed in [Guernez, 1997], this system consists of six heat exchangers which distribute three flows $f_i$ that come at different temperatures $t_i$, in accordance with the sensor allocations. There are different subsystems, each one formed by two exchangers: {E1, E2}, {E3, E4} and {E5, E6}. Each one of the six exchangers and each one of the eight nodes of the system are considered as components. The observable variables (flows and temperatures) are represented by shaded circles in Figure 1. The normal behaviour of the system can be described by means of polynomial constraints based on three different kinds of balances:

$$\sum_i f_i = 0 \text{: mass balance at each node}$$
$$\sum_i f_i \cdot t_i = 0 \text{: thermal balance at each node}$$
$$\sum_{in} f_i \cdot t_i - \sum_{out} f_j \cdot t_j = 0 \text{: enthalpic balance for each heat exchanger}$$

- A polybox system: This polybox system is derived from the standard problem used in the diagnosis community [de Kleer *et al.*, 1992]. The system consists of fifteen components: nine multipliers, and six adders. The observable variables are represented by shaded circles in Figure 2.

## 4   Structural pretreatment

The first part of this section shows the way to divide the diagnosis problem into independent diagnosis subproblems. The second part of this section explains the way of grouping the components into nodes in order to reduce the number of non-observable variables to be considered in the system.

### 4.1   Identification of the clusters

The objective of this section is the partition of the system into independent subsets of components. This partition reduces the computational complexity of the diagnosis process since it enables the generation of the diagnosis of the whole system based on the diagnosis of the subsystems.

**Definition 4.1** *Cluster of components*: A set of components $T$ is a cluster, if it does not exist a common non-observable variable of any component of the cluster with any component outside the cluster, and if for all $T' \subset T$, $T'$ is not a cluster of components.

In a cluster, all common non-observable variables among the components belong to the same cluster, therefore all the
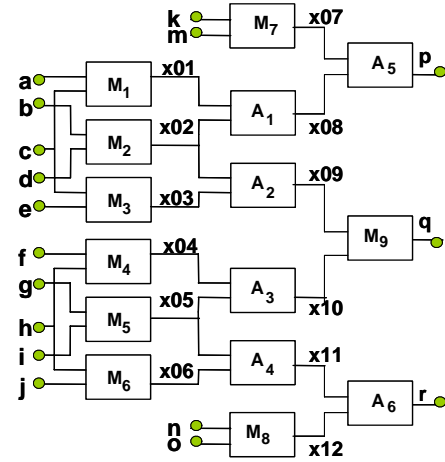


Figure 2: The polybox example

connections with components which are outside the cluster are monitored. A cluster of components is totally monitored, and for this reason the detection of faults inside the cluster is possible without information from other components which do not belong to the cluster. A more detailed explanation and the cluster detection algorithm appears in previous work [Ceballos *et al.*, 2004].

The diagnosis space for a system initially consists of $2^{nComp}$ diagnoses [de Kleer *et al.*, 1992], where *nComp* is the number of components of the system. Therefore the computational complexity for the diagnosis process is always smaller for an equivalent system divided into clusters, due to the reduced number of possible diagnoses.

**Example**: In the heat exchanger example, five clusters of components can be obtained [Ceballos *et al.*, 2004]: {$N_{11}$}, {$N_{13}$}, {$N_{12}$, $N_{21}$, $N_{22}$, $E_1$, $E_2$}, {$N_{14}$, $N_{23}$, $N_{24}$, $E_5$, $E_6$}, and {$E_3$, $E_4$}. These clusters allow the generation of an independent diagnosis process for each cluster, therefore the number of possible diagnoses is reduced to: $(2^1-1)+(2^1-1)+(2^5-1)+(2^5-1)+(2^2-1) = 67$. If each constraint of the model is considered as a component, then it is possible to obtain more clusters to a total of 9 clusters, as shown in Section 4.3.

### 4.2   Obtaining relations without non-observable variables

In the diagnosis process it is necessary to produce new relations without non-observable variables, in order to monitor the system behaviour by using only the observational model. Our approach uses a function named *NewRelations* (NR) which takes a set of constraints and obtains a set of new constraints without a set of non-observable variables. Example: NR({x-a·c, y-b·d, f-x-y}, {x ,y}) = {a·c + b·d - f = 0}.

This function can be implemented using different techniques. The Gröbner Basis algorithm [Buchberger, 1985] is used here. Gröbner basis theory is the origin of many symbolic algorithms used to manipulate equality polynomials. It is a combination of Gaussian elimination (for linear systems) and the Euclidean algorithm (for univariate polynomials over a field). The Gröbner basis can be used to produce an equiv-

```
NodesIdentification(T) return N , S
  E_v = List of components associated to the variable v
  N_i = List of components of the node i
  S_i = List of dispensable variables of the node i
  // Generate all the possible nodes of components
  foreach x ∈ T
    N_i = {x}
  endforeach
  // Detect all the components associated to a variable
  foreach N_i ∈ N
    foreach var_k ⊂ nonObsVar(N_i)
      E_k = E_k ∪ {N_i}
    endforeach
  endforeach
  // Merging the nodes of components
  while ∃ E_k ∈ E where |E_k| = 2 ∧ E_k = {N_i,N_j} ∧ i<j
    N_i = N_i ∪ N_j
    N = N \ N_j
    S_i = S_i ∪ S_j ∪ {k}
    foreach E_q ∈ E where N_j ∈ E_q
      E_q = E_q \ N_j
      if ¬ ( N_i ∈ E_q )
        E_q = E_q ∪ N_i
      endif
    endforeach
  endwhile
```

Figure 3: The algorithm to select the nodes of components

alent system which has the same solution as the original, and without having non-observable variables.

### 4.3 Obtaining the nodes of each cluster

Main assumption in this paper is to suppose that only one constraint is associated to each component. If it is necessary to apply this methodology to components with $n$ constraints (where $n > 1$), it is then possible decoupling the component $x$ into $n$ virtual components $x_i$ with one constraint each.

Our approach provide the minimal set of constraints to detect all the possible diagnoses of a system. The introduction of new definitions is necessary in order to efficiently generate these set of constraints:

**Definition 4.2** *Dispensable variable*: A non-observable variable $v_i$ is dispensable if there exist only two components $x_i$ and $x_j$ which include this variable in their related constraints. In the polybox example the variable *x04* and the variable *x08* are dispensable variables.

**Definition 4.3** *Node of components*: A single component could be a node of components if none of its non-observable variables is a dispensable variable. Two components, or, a component and a node of components, belong to the same node of components if they have a common dispensable variable.

**Lemma 1**. Let $N$ be a node of components. Let $C$ be a context. If $C \cap N \neq \emptyset \land \neg(N \subset C)$, then, by using only the components of the set $C$ - $\{C \cap N\}$ is possible to generate constraints without non-observable variables.

**Proof**. In order to generate constraints without non-observable variables by using the set of components $S = N \cap C$, it is necessary to eliminate the dispensable variables of

Table 1: Improvements obtained using structural pretreatment in the proposed examples

| Examples: | Clusters: | Nodes: | Vars.: | Ctxs.: |
|---|---|---|---|---|
| Heat E. | 9 | 14 | $26 \to 0$ | $2^{14}\text{-}1 \to 14$ |
| Polybox | 1 | 5 | $12 \to 2$ | $2^{15}\text{-}1 \to 31$ |

the set $S$. If the set $S$ does not have all the components of $N$, then there exists $v$, a dispensable variable, which cannot be eliminated, since it only appears in one component of the set $S$. Therefore, by using the components of the set $N \cap C$, it is not possible to generate constraints without non-observable variables.

**Corollary 1**. The contexts are built by using nodes of components instead of components, since it is impossible to generate constraints without non-observable variables by using a subset of a node of components.

The following algorithm obtains the set of nodes of a cluster $T$.

**Algorithm**: Figure 3 shows the algorithm *NodesIdentification(T)* which takes $T$, the set of components of a cluster, and returns $N$, the list of nodes, and $S$ the list of the dispensable variables of each node. The algorithm begins by creating $n$ nodes, where $n$ is the number of components of the cluster. All these nodes have one component. Each list $E_k$ stores all the nodes which have a constraint which includes the non-observable variable $k$. The auxiliary function *nonObsVar* returns the set of non-observable variables of a set of components.

The dispensable variables are detected when a list $E_k$ (associated with the variable $k$) includes only two nodes. In this case, the two nodes, $N_i$ and $N_j$, are merged into the node $N_i$. The lists $S_i$ and $S_j$ are merged (they are the set of dispensable variables of the new node), and obviously, variable $k$ must be included in set $S_i$. After this step it is necessary to update all the elements of the list $E_q$, in order to guarantee that node $N_i$ and not $N_j$ appears in all the elements of list $E$.

Our approach obtains the nodes of each cluster of the system. A new set of constraints without the dispensable variables is obtained by applying the *NewRelations* function to the set of constraints of each node. If the node of components have no dispensable variables it is not necessary to apply the *NewRelations* function.

**Example**: Table 1 shows the results obtained in the two proposed examples. The column *Nodes* shows the addition of all the nodes included in the clusters of the system. The column *Vars* shows the initial number of non-observable variables, and the final number of non-observable variables that are not dispensable variables. The column *Ctxs* shows the total number of possible contexts of the system, and the final number of possible contexts by using the nodes of components. In the heat exchanger example 9 clusters are obtained (if each constraint of the model is considered as a component). The non-observable variables are reduced from 26 to 0, as it appears in Table 1, because all the variables are dispensable. Table 2 shows the list of nodes of the polybox example, and the constraint obtained in each node by eliminating the dispensable variables.

## 5 Determination of the context network

In DX community the diagnosis is obtained by detecting conflicts. Many methodologies tries to use structural description of the system, those methods are known as compilation methods. In [Pulido and González, 2004] the Possible Conflicts (PCs) concept is proposed as a compilation technique. Each PC represent a subsystem within system description containing minimal analytical redundancy and being capable to become a conflict. Computing Analytical Redundancy Relations (ARRs)[Staroswiecki and Declerk, 1989] is the compilation technique of FDI methodology.

Our approach provide the minimal set of contexts which include an over-determined system of constraints that can detect a conflict in a cluster. The minimality issue was not guaranteed in the original ARR approach, but its guaranteed in our approach. In [Pulido and González, 2004] approach the PCs are obtained directly by using components, but our approach use nodes instead of components, therefore the size of the problem is reduced from $2^c$, where $c$ is the number of components, to $2^n$, where $n$ is the number of nodes.

A context network, in accordance with the way proposed by ATMS[de Kleer, 1986], is generated in order to obtain all the relevant contexts for the diagnosis process. In order to establish the smallest set of contexts it is necessary to introduce the following definitions.

**Definition 5.1** *Structural context*: This is a context where all the nodes are connected, that is, they compose a connected graph, and all the non-observable variables appear in at least two constraints. The function to determine which are structural contexts is named *isAStructural* and takes a context C and returns a true value if it is a structural context.

**Definition 5.2** *Minimal completed context*: A structural context C is completed context if the set of constraints of the nodes of the context is a over-determined system of constraints, and, if it is possible to generate new constraints without non-observable variables by using the set of constraints of the context. A completed context is minimal if no context C' $\subset$ C exists such that C' is a completed context.

**Lemma 2**. If C is a minimal completed context then no context C' exists, where C $\subset$ C', which can generate relevant constraints in the diagnosis process.

**Proof**. Let C be a completed context with $n$ nodes. Let C' be a context with $m$ nodes such that $C \cap C' = \emptyset$, and let be D a context with $n + m$ nodes such that $C \cup C' = D$. Therefore:

a) If C' is not a completed context, it is impossible to eliminate one of the non-observable variables of the context C', and hence it is not possible to generate a new constraint with all the nodes of the context D. It is possible to obtain constraints with fewer nodes than $n+m$ nodes of context D, however these constraints can be obtained in a context with less nodes, and hence D is not minimal.

b) If C' is a completed context, then context *D* can generate a new constraint with all the nodes of the context. However, with this new constraint, which uses all the nodes of context D, it is impossible to distinguish between faults of nodes included in *C* and those included in *C'*. Hence it is not relevant in the diagnosis process.

Therefore, in these two possible cases, it is not possible to

Table 2: Nodes for the polybox example

| Nodes | Components | Constraints |
|---|---|---|
| $N_1$ | $M_6 M_8 A_4 A_6$ | h·j + n·o - r + x05 |
| $N_2$ | $M_5$ | g·i - x05 |
| $N_3$ | $M_1 M_7 A_1 A_5$ | a·c + k·m - p + x02 |
| $N_4$ | $M_2$ | b·d - x02 |
| $N_5$ | $M_3 M_4 M_9 A_2 A_3$ | q - (f·h + x05)·(x02 + c·e) |

generate new relevant constraints for the diagnosis process if new nodes are added.

**Corollary 2**. If a context C is a minimal completed context it is not necessary to process contexts C' such that C $\subset$ C', since it is not possible to generate new relevant constraints for the diagnosis process.

The algorithm which generate the contexts of each cluster has $n$ - 1 stages, first the context with 2 nodes are obtained, then the context with 3 nodes, until it reaches the context with $n$ nodes, where $n$ is the number of nodes. The function *NewRelations* is only applied to the contexts which are structural contexts. When a minimal completed context C is found, the new constraints without non-observable variables are stored, and no contexts C', such that C $\subset$ C', are generated. These new constraints are named Context Analytical Redundancy Constraint.

**Definition 5.3** *Context Analytical Redundancy Constraint (CARC)*: A constraint obtained from a minimal completed context in such a way that only the observed variables are related.

**Example**: In order to clarify this section, Tables 2 and 3 shows the results obtained for the polybox example. This system includes only one cluster with 15 components. The number of possible contexts is reduced from $2^{15}-1$ to $2^5-1$. By applying the rules and the algorithm proposed in this section, 10 contexts of the possible 31 ($2^5-1$) are generated, however only 6 are minimal completed contexts. These 6 contexts generate 6 CARCs. Figure 4 shows the context network of the polybox example. Only the treated contexts are circled. The minimal completed contexts are circled in bold.

## 6 Determination of the minimal diagnosis

The last step is the determination of the minimal diagnosis using the set of CARCs. In order to clarify the methodology, we suppose that the sensor observations are correct. We propose using a signature matrix as in FDI, but in order to obtain the same minimal diagnosis as in DX approach, it is necessary to apply a set of rules which guarantee the no-exoneration case in the solution.

**Definition 6.1** *Fault signature*: Given a set of $n$ CARCs, denoted CARC= { $CARC_1$, $CARC_2$, ..., $CARC_n$}, and a set of $m$ faults denoted F = {$F_1$,...,$F_m$}, the signature of a fault $F_j$ is given by $FS_j = [s_{1j},..., s_{nj}]^T$ in which $s_{ij} = 1$ if the context which generated the $CARC_i$ involves the nodes included in the fault $F_j$, and $s_{ij} = 0$ otherwise.

**Definition 6.2** *Signature matrix*: All the signatures for the set of possible faults constituted the signature matrix.

**Definition 6.3** *Signature of an observation*: This is given by OS=[$OS_1$,...,$OS_n$] where $OS_i$=0 if the $CARC_i$ is satisfied,

Table 3: CARCs obtained in the polybox example

| Index | Context | CARC |
|-------|---------|------|
| 1 | $N_1N_2$ | h·j + n·o - r + g·i |
| 2 | $N_3N_4$ | a·c + k·m - p + b·d |
| 3 | $N_1N_3N_5$ | q - (f·h - h·j - n·o + r)·(-a·c - k·m + p + c·e) |
| 4 | $N_1N_4N_5$ | q - (f·h - h·j - n·o + r)·(b·d + c·e) |
| 5 | $N_2N_3N_5$ | q - (f·h + g·i)·(-a·c - k·m + p + c·e) |
| 6 | $N_2N_4N_5$ | q - (f·h + g·i)·(b·d + c·e) |



Figure 4: Context network of the polybox example

and $OS_i$=1 otherwise.

**Definition 6.4** *Diagnosis set*: The set of faults whose signatures are consistent with the signature of the observational model. Our approach supposed that an observation signature OS is consistent with another signature $FS_j$ if $OS_i = s_{ij} \ \forall$ i.

**Definition 6.5** *Minimal diagnosis*: A fault $F_j$ is a minimal diagnosis if $F_k$ is not a diagnosis $\forall$ faults $F_k \subset F_j$.

Table 4 shows the signature matrix for the polybox example in order to clarify these definitions and the process to obtain the minimal diagnosis. The signature OK = [0, ..., 0]$^T$ represents the no-fault case. The signature matrix is very similar to the corresponding matrix in the FDI methodology. However in our approach, the faults involve nodes instead of components.

In this example it is necessary to expand the number of columns of the signature matrix in order to obtain the multiple faults. Each fault $F_j$, which involves *n* nodes, is obtained using a fault $F_k$, which involves $n-1$ nodes, and a simple fault $F_s$ which is not involved in $F_k$. The multiple fault signature $F_j$ is given by $FS_j = [s_{1j},..., s_{nj}]^T$ in which $s_{ij} = 0$ if $s_{ik}=s_{is}$, and $s_{ij} = 1$ otherwise. The multiple fault signature $F_j$ is not added to the signature matrix if $\forall \ s_{ij} : s_{ij} = 1 \rightarrow s_{ij} = s_{ik}$, due to the implication that the new multiple fault is a superset of a previously obtained fault which involves fewer nodes, and therefore cannot be part of a minimal diagnosis. The generation of the signature matrix stops when it is impossible to generate new signatures of faults which involve *n* nodes, with the faults which involve $n-1$ nodes.

In FDI, the exoneration assumption [Cordier *et al.*, 2000] is implied, that is, given an observational model, each component of the support of a satisfied CARC is considered as functioning correctly, that is, it is exonerated. In the DX approach, the exoneration is not considered by default.

In order to obtain the same results as in the DX approach by using a signature matrix, it is necessary to apply a new definition of consistency. In the no-exoneration case an observation signature OS is consistent with another signature $FS_j$ if $\forall \ OS_i = 1$ then $s_{ij} = 1$. That is, only the non-satisfied CARCs are used, and $F_j$ must have the value 1 in each non-satisfied CARC. When the diagnosis set is obtained by using the new definition of consistency, we propose the application of a set of rules in order to detect which of the faults are minimal diagnoses, since many faults will be consistent with the observational model although they are not a minimal diagnosis. The following algorithm generates the rules to obtain the minimal diagnosis.

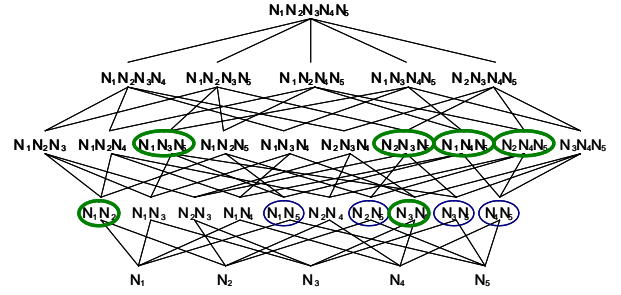**Algorithm**: Let CS(OS,FS) be a function which evaluates

whether the signature OS is consistent with signature FS. For each possible fault $F_j$ in the signature matrix, let $MD_{Fj}$ be a Boolean variable which holds information on whether a fault $F_j$ is a minimal diagnosis, and let $VC_{Fj}$ be a Boolean variable which holds information on whether a fault $F_j$ is a valid candidate for the generation of new faults that could be a minimal diagnosis. For each possible fault $F_j$ it is initially supposed that $VC_{Fj}$ = true.

The first step is to validate if the OK (no fault case) is a minimal diagnosis: $MD_{OK} = CS(OS,OKS)$, and, for any simple fault $F_j$, the equality $VC_{Fj} = \neg \ MD_{OK}$ must be satisfied.

If OK is not a minimal diagnosis, the following rules must be evaluated for all the possible faults (except OK) in the same sequential order as they appear in the signature matrix. These rules guarantee the correct detection of the minimal diagnosis for an observational model:

- For each fault $F_j$ with the signature $FS_j$, the equality $MD_{Fj} = VC_{Fj} \wedge CS(OS,FS_j)$ must be satisfied.

- For each fault $F_k$ which involves n + 1 nodes, where n $\geq$ 0, and which can be obtained using the fault $F_j$(that involves n nodes) and a simple fault $F_s$(which is not involved in $F_j$) then $MD_{Fj} \Rightarrow VC_{Fk} =$ false.

Table 5: A subset of the rules for the polybox example

| | |
|---|---|
| $MD_{OK} = CS(OS,OKS)$ | $MD_{F2} = VC_{F2} \wedge CS(OS,FS_2)$ |
| $VC_{F1} = \neg \ MD_{OK}$ | ... |
| ... | $MD_{F13} = VC_{F13} \wedge CS(OS,FS_{13})$ |
| $VC_{F5} = \neg \ MD_{OK}$ | $MD_{F13} \Rightarrow VC_{F123} =$ false |
| $MD_{F1} = VC_{F1} \wedge CS(OS,FS_1)$ | $MD_{F13} \Rightarrow VC_{F134} =$ false |
| $MD_{F1} \Rightarrow VC_{F12} =$ false | $MD_{F13} \Rightarrow VC_{F135} =$ false |
| ... | $MD_{F14} = VC_{F14} \wedge CS(OS,FS_{14})$ |
| $MD_{F1} \Rightarrow VC_{F14} =$ false | ... |
| $MD_{F1} \Rightarrow VC_{F15} =$ false | $MD_{F245} = VC_{F245} \wedge CS(OS,FS_{245})$ |

**Example**: Table 5 shows a subset of the rules for the polybox example. The generation of the rules for the verification of whether a fault is a minimal diagnosis can be done off-line, because these rules are the same for all the observational models. The bottom of Table 4 shows the VC and MD evaluation results for the observation signature OS = [0, 0, 1, 1, 1, 1]$^T$. Only the evaluation of the rules must be done on-line. This part of the process is a simple propagation of Boolean values.

Table 4: The signature matrix of the polybox example

| CARC | OK | $F_1$ | $F_2$ | $F_3$ | $F_4$ | $F_5$ | $F_{12}$ | $F_{13}$ | $F_{14}$ | $F_{15}$ | $F_{23}$ | $F_{24}$ | $F_{25}$ | $F_{34}$ | $F_{35}$ | $F_{45}$ | $F_{xxx}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| 2 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| 3 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 4 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 5 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 6 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

$$F_{xxx} \subset \{F_{123}, F_{124}, F_{134}, F_{135}, F_{145}, F_{234}, F_{235}, F_{245}\}$$

| | OK | $F_1$ | $F_2$ | $F_3$ | $F_4$ | $F_5$ | $F_{12}$ | $F_{13}$ | $F_{14}$ | $F_{15}$ | $F_{23}$ | $F_{24}$ | $F_{25}$ | $F_{34}$ | $F_{35}$ | $F_{45}$ | $F_{xxx}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| VC | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| MD | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

VC and MD values for the observation signature OS $= [0, 0, 1, 1, 1, 1]^T$

The evaluation of the signature matrix is very similar to the FDI methodology. However in our approach, the faults involve nodes instead of components. Hence, the last step is the substitution of each node with one of its components. In the polybox example, fault $F_3$ is equivalent to the faults in $\{\{M_1\}, \{M_7\}, \{A_1\}, \{A_5\}\}$; fault $F_{12}$ is equivalent to faults $\{\{M_6M_5\}, \{M_8M_5\}, \{A_4M_5\}, \{A_6M_5\}\}$; and so on.

The information of all the possible minimal diagnoses is stored in a matrix and as a set of rules. Therefore, it is only necessary to calculate this diagnosis once. As happens in FDI methodology, this work can be done off-line, only the evaluation of the signature matrix is on-line.

## 7 Conclusions and future work

This paper proposes a new approach to automation of and improvement in the determination of minimal diagnosis. The approach is based on FDI and DX theories. The structural pre-treatment in the first phase and the reduction of the model in the second phase enable improvement in the computational complexity. All the possible minimal diagnoses are represented in a signature matrix. The minimal diagnosis is obtained by using an observational model and a set of precompiled rules. Only the evaluation of the signature matrix is on-line, the rest of the process can be done off-line.

The methodology was applied to two different examples, and the results were very promising. As future work we suggest extending the methodology to include dynamic systems and to include more complex and real problems, where the application of the methodology could be more complicated.

## 8 Acknowledgment

## References

[Buchberger, 1985] B. Buchberger. Gröbner bases: An algorithmic method in polynomial ideal theory. *Multidimensional Systems Theory, N. K. Bose, ed.*, pages 184–232, 1985.

[Cassar and Staroswiecki, 1997] J. Cassar and M. Staroswiecki. A structural approach for the design of failure detection and identification systems. In *IFAC-IFIP-IMACS Conf. on Control of Industrial Processes, Belfort, France*, 1997.

[Ceballos *et al.*, 2004] R. Ceballos, M.T. Gomez, R.M. Gasca, and S. Pozo. Determination of possible minimal conflict sets using components clusters and grobner bases. In *DX04*, pages 21–26, Carcassonne, France, June 2004.

[Cordier *et al.*, 2000] M. Cordier, F. Lévy, J. Montmain, L. Travémassuyés, M. Dumas, M. Staroswiecki, and P. Dague. A comparative analysis of ai and control theory approaches to model-based diagnosis. In *14th European Conference on Artificial Intelligence*, pages 136–140, 2000.

[de Kleer *et al.*, 1992] J. de Kleer, A. Mackworth, and R. Reiter. Characterizing diagnoses and systems. *Artificial Intelligence*, 2-3(56):197–222, 1992.

[de Kleer, 1986] J. de Kleer. An assumption-based truth maintenance system. *Artificial Intelligence*, 2(28):127–161, 1986.

[Guernez, 1997] C. Guernez. Fault detection and isolation on non linear polynomial systems. In *15th IMACS World Congress on Scientific, Computation, Modelling and Applied Mathematics*, 1997.

[Pulido and González, 2004] B. Pulido and C. Alonso González. Possible conflicts: A compilation technique for consistency-based diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, 34(5):2192–2206, October 2004.

[Reiter, 1987] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence 32*, 1:57–96, 1987.

[Staroswiecki and Declerk, 1989] M. Staroswiecki and P. Declerk. Analytical redundancy in non linear interconnected systems by means of structural analysis. In *IFAC Advanced Information Processing in Automatic Control (AIPAC-89)*, pages 51–55, Nacy, France, June 1989.