An optimal sensor placement algorithm taking into account diagnosability specifications

Abed Alrahim Yassine[1], Albert Rosich[2], Stéphane Ploix[1]

[1]G-SCOP lab, INPG, CNRS UMR 5272, BP 46, F-38402 Saint Martin d'Hères Cedex, France; e-mail:
first-name.name@g-scop.inpg.fr

[2]Albert Rosich, Universitat Politecnica de Catalunya, Automatic Control Department (ESAII), Spain;
email:albert.rosich@upc.edu

Abstract—Suitable installed sensors in a industrial process is a necessary condition for fault diagnosis. Sensor placement for diagnosis purposes is to study which process variables have to be measured to satisfy diagnosis specifications (detectability, discriminability and diagnosability). This paper presents a method based on the study of the structural model properties and the Dulmage-Mendelsohn decomposition. Due to the use of structural models, the proposed approach can be applied to a wide variety of system (linear, algebraic, dynamics, etc.). Assuming that the cost of placing a sensor for each possible variable is defined, this method finds the minimal cost sensor configuration according to the diagnosability criteria. This method does not require the computation of testable subsystems

Index Terms—Sensor placement, Structural modeling, Fault diagnosis, diagnosability specifications

I. INTRODUCTION

In the scientific literature, many approaches of the fault diagnosis have been proposed since 1980. FDI (fault detection and isolation) approach ([11], [4]) and DX approach ([16], [2]). FDI approach is developed by the Automatic Control community. It is based on the concept of signature table and the residuals based approaches, whereas DX approach is coming from the Artificial Intelligence community and proposes a logical framework for automatic diagnosis processes. But this framework was dedicated only to static binary systems. Recently, a bridge approach between FDI and DX has been proposed ([10], [12]). Thus, tools for solving diagnosis problems are now well established. However, designing an efficient diagnosis system does not start after the system design but it has to be done during the system design. Indeed, the performance of a diagnostic system highly depends on the number and on the location of actuators and sensors. Therefore, designing a system that has to be diagnosed not only require relevant fault diagnosis procedures but also efficient sensor placement algorithms.

[7] has proposed a sensor placement method which deals with linear systems. This approach makes use of the Gauss-Jordan elimination method to find a minimum set of variables to be measured. This ensures the observability of variables while simultaneously minimising the cost of sensors. Another method for sensor placement has been proposed in [8]. This method aims at guaranteeing the detectability and isolability of sensor failures. The sensor placement can be solved by an

analysis of a cycle matrix or using the technique of mixed linear programming.

However, all these methods are not suitable for the design of systems that include a diagnosis system because, in this context, the goal of sensor placement should be to make it possible to monitor hazardous components. The sensor placement algorithm should compute solutions that satisfy detectability and discriminability properties. Few methods have focused on this problem.

[18] has proposed a method based on consecutive additions of sensors, which takes into account diagnosability criteria. This method requires an a priori design of all the ARR for a given set of sensors.

Recently, the sensor placement problem satisfying diagnosability objectives becomes possible without designing **ARR**s. [5] has proposed an efficient method based on a partial order on the well-determined subsets from a structural model. This method finds all the possible minimal sensor sets which makes the problem intractable when the solutions set becomes large.

Another sensor placement method without designing **ARR**s has be presented in ([19]). This method improves the possibility of detecting and localizing faults in systems for which only the structure is known. It considers the complete range of specifications respect to the constraints, i.e. the set of constraints that must be diagnosable, the set of constraints that must be non discriminable but detectable and the set of constraints that must be non detectable.

In continuation of the method presented in [19], this paper presents a sensor placement method which considers practical specifications. Fault detectability and discriminability are handled by means of the Dulmage-Mendelsohn decomposition [3]. Moreover, the search of the optimal solution is performed by using an Combinatorial algorithm.

II. SENSOR PLACEMENT PROBLEM FOR DIAGNOSIS

A. *Sensor placement motivation*

Let us present an intuitive (but not efficient) formulation to understand the sensor placement problem. The solution of a diagnostic problem is generally decomposed into two consecutive steps. The first step relies on consistency tests by means of minimal testable subsets of constraints. Such testable subsets can be obtained from constraints combinations using possible conflict generation [15], bipartite graph [1], Dulmage-Mendelsohn decomposition [6] or elimination rules [13]. An

inconsistency in a testable subset means that, at least, one of the behaviour modes associated to this constraint subset is not actual.The second step is the diagnostic analysis, which provides global conclusions in term of which are the actual system modes. The performance of a diagnostic depends on such testable subsets. Since the generation of all possible testable subsets depends on the data flows (the observation set), then it turns out that diagnostic performance also depends on these observations.

Additional sensors lead to additional testable subsets. Basically, once the testable subset has been generated, it is therefore possible to compute the performance of the diagnostic system given some diagnosis specifications. Whether the performance satisfies the requested specifications or not, the testable subset is modified by adding more sensors and the process is repeated once again until specifications are reached. However, this process requires lots of computations because the generation of testable subsets is time consuming.Another better approach for sensor placement is proposed in this paper. It does not require the computation of testable subsystems, therefore the computation burden is avoided. It directly solves the following problem by studying the properties of the model structure.

B. Diagnosis specifications

The behaviour model of a system Σ can be defined as a set of relations (constraints) among a set of variables V. In a component-oriented model, these relations called primary relations are associated to the components of a system and each physical variable $v \in V$ belongs to a domain $dom(v)$ representing all the possible values for v.

In this section diagnosis specifications are introduced. First, the *minimal testable set* of constraints (TSS) is formally defined and then the concept of TSS is used to define complete diagnosis specifications.

A requirement to test consistency given a TSS, is to be able to compute all the unknown variable involved within this TSS. This motivates Definition 1.

Definition 1 (Solving Constraint Set): Let K be a set of constraints and v a variable in $var(K)$ characterised by its domain $dom(v)$. K is a solving constraint set for v, if using K, it is possible to find a value set S for v such that $S \subset dom(v)$. A solving constraint set K for v is denoted as $K \vdash v$.

From the above definition, the TSS is formally defined in the following definition.

Definition 2 (Minimal Testable Set (TSS)): Let K be a set of constraints. K is a minimal testable set if and only if there are two disjoint subsets $K_1, K_2 \subset K$ (i.e. $K_1 \cap K_2 = \emptyset$) such that $K_1 \vdash v$ and $K_2 \vdash v$ for $v \in var(K)$, and there is no subset in K with such property.

If the property in Definition 2 is satisfied, then it is indeed possible to check whether the value set S_1 deduced from K_1 is consistent with the value set S_2 deduced from K_2: $S_1 \cap S_2 \neq \emptyset$. Moreover, adding any constraint to a testable set leads also to a testable set of constraints. However, only minimal testable sets are interesting. A global testable constraint that can be deduced from a TSS is called an *Analytical Redundancy Relation*.

Let $\mathbb{K} = \{\ldots, K_k, \ldots\}$ be the set of all the minimal testable subsystems that can be deduced from a system constraint set K_Σ according to Definition 2. In this paper, we assume without loss of generality, that there is a one-to-one relationships between constraints and behaviour mode. This allows us to extend the notions of detectability and discriminability to constraints from the set $\mathbb{K}_\Sigma$ of TSS.

Definition 3 (Detectable constraint): Let $\mathbb{K}$ be the set of TSS. A constraint $k \in K_\Sigma$ is detectable [17] in $\mathbb{K}$ iff

$$\exists K_i \in \mathbb{K} : k \in K_i \tag{1}$$

By extension, a set of constraints $K \subseteq K_\Sigma$ is detectable in $\mathbb{K}$ if $\forall k_i \in K$, k_i is detectable in $\mathbb{K}$.

Definition 4 (Discriminable constraints): Two different detectable constraints $k_1, k_2 \in K_\Sigma$ are discriminable [17] in $\mathbb{K}$ iff:

$$\exists K_i \in \mathbb{K} : k_1 \in K_i \wedge k_2 \notin K_i \tag{2}$$

By extension, two sets of detectable constraints $K_1, K_2 \subset K_\Sigma$ such that $K_1 \cap K_2 = \emptyset$ are discriminable in $\mathbb{K}$ iff: $\forall k_i \in K_1$ and $\forall k_j \in K_2$ k_i, k_j are discriminable in $\mathbb{K}$.

Definition 5 (Diagnosable constraint): A detectable constraint $k \in K_\Sigma$ is diagnosable [17] in $\mathbb{K}$ iff:

$$\forall k_j \in (K_\Sigma \setminus k), \exists K_i \in \mathbb{K} : k \in K_i \wedge k_j \notin K_i \tag{3}$$

By extension, a subset of detectable constraints $K \subset K_\Sigma$ are diagnosable in $\mathbb{K}$ iff: $\forall k_i \in K$, k_i is diagnosable in $\mathbb{K}$.

In fault diagnosis, sensor placement has to satisfy specifications dealing with detectability, discriminality and diagnosability. Because each component is assigned to one constraint, the required diagnosis specifications can be defined over the constraint set. Thus, the practical specifications presented in this paper consist in a classification of the constraint set into the following tree constraint subsets in K_Σ:

- the set of constraints K_{diag}^{spec} that must be diagnosable.
- the set $\mathbb{K}_{disc}^{spec}$ of constraints sets that must be discriminable.
- the set of constraints K_{det}^{spec} that must be at least detectable.

The practical specifications K_{diag}^{spec}, $\mathbb{K}_{disc}^{spec}$ and K_{det}^{spec} for sensor placement problems are meaningful if the two following properties are satisfied:

$$K_i \cap K_j = \emptyset$$
$$\forall K_i, K_j \in \mathbb{K}_{disc}^{spec} \cup \{K_{diag}^{spec}\}; K_i \neq K_j \tag{4}$$

$$(\bigcup_{K \in \mathbb{K}_{disc}^{spec}} (K) \cup K_{diag}^{spec}) \subseteq K_{det}^{spec} \subseteq K_\Sigma \tag{5}$$

Let K_Σ' represents the extended system model constraints with the additional sensors, i.e. K_Σ' contains the system model constraints K_Σ plus the additional sensor terminal constraints K_V of V variables, $K_\Sigma' = K_\Sigma \cup K_V$. Therefore, solving a sensor placement problem consists in determining the additional sensor terminal constraints K_V that lead to the satisfaction of practical specifications according Definitions 3, 4 and 5.
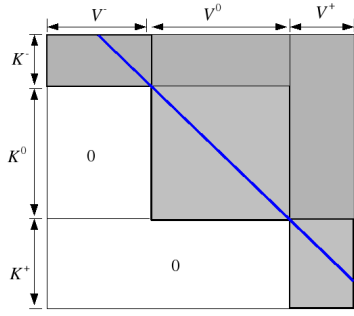
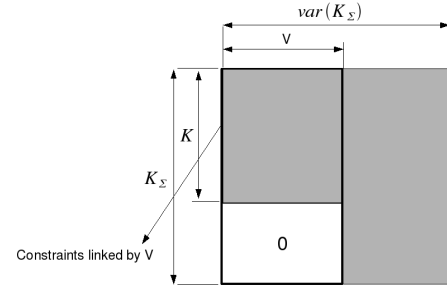Fig. 1.   Dulmage-Mendelsohn decomposition of a bipartite graph

Fig. 2.   Structural matrix of a constraint set K, which is linked by a set of variables V

Fig. 3.   Structural matrix of non detectable constraints

III. Dulmage-Mendelsohn decomposition

The Dulmage-Mendelsohn Decomposition noted "DM Decomposition" is a useful tool for the structural analysis of system to diagnose. La decomposition DM can be obtained by applying classical algorithms of bipartite graph ([3], [9]). This decomposition results by permuting the columns and rows of the incidence matrix which represents the graph in order to obtain a lower triangular block.

A Dulmage-Mendelsohn decomposition ([3]) of a bipartite graph leads to three canonical components called K^+, K^0, K^-. Figure 1 shows these components, where the grey areas means that the matrix contains 0 and 1 and the white areas only contains 0. In this figure, we remark that there are three subsystems:

$$S^+ = (K^+, V^+)$$
$$S^0 = (K^0, V^+ \cup V^0) \qquad (6)$$
$$S^- = (K^-, V^+ \cup V^0 \cup V^-)$$

such that

- (K^+, V^+) is over-constrained,
- (K^0, V^0) is just-constrained,
- (K^-, V^-) is under-constrained.

An efficient algorithm [14] to compute the Dulmage-Mendelsohn decomposition is implemented in Matlab (*dm-perm* function).

IV. Properties of structural models

The required diagnosability properties of structural models are presented in this section. First, the concept of linked constraints has to be formalized because discriminability depends on this concept.

Definition 6 (Linked set of constraints): A set of constraints $K \subseteq K_\Sigma$ is linked by a set of variables $V \subseteq var(K)$ if and only if the three following expressions hold:

$$cons(V) = K \qquad (7)$$

$$(K^0 \cup K^-, V^0 \cup V^-) = (\emptyset, \emptyset) \qquad (8)$$

$$|K| = |V| + 1 \qquad (9)$$

where $cons(V)$ stands for the set of constraints adjacent to the set of variables V.

Note that equation (8) means that the sub-model formed by constraints K and variables V has no just or under-constrained

part.

The shape of an incidence matrix dealing with linked constraints is depicted in figure 2, where the number of constraints in K is one more than the number of variables in V according to (9).

The concept of *linked constraints* is strongly connected with discriminability.

Lemma 1: A set of constraints $K \subset K_\Sigma$ linked by a set of variables $V \subset var(K_\Sigma)$ is necessarily non discriminable.
The proof is presented in [19].

A set of constraints cannot be used to generate a TSS if they are linked and if there are additional variables that cannot be solved. These constraints are qualified as isolated. Detectability depends on this concept.

Definition 7 (Isolated set of constraints): A constraint $k \in K_\Sigma$ is isolated by a set of variables $V \subseteq var(k)$ iff

$$cons(V) = k \qquad (10)$$

By extension, a set of constraints $K \subset K_\Sigma$ is isolated by a set of variables $V \subseteq var(K)$ iff K is linked by $V' \subseteq V$ and there is a non-empty variable subset $V'' \subseteq V \setminus V'$ such that

$$cons(V'') \subseteq K \qquad (11)$$

The shape of a structural matrix dealing with isolated constraints is depicted in figure 3.

The concept of isolated constraints is strongly linked with detectability.

Lemma 2: A set of isolated constraints $K \subseteq K_\Sigma$ is necessarily non detectable.
The proof is presented in [19].

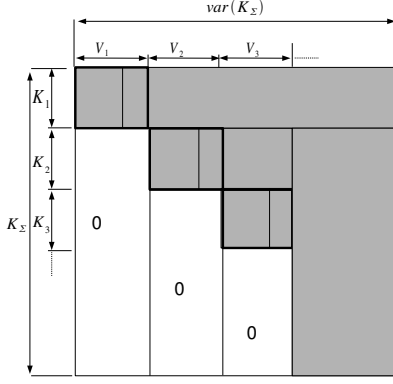In lemma 2, a relationship between isolated constraints and

Fig. 4. Structural matrix of non detectable constraints

the detectability property has been presented. Conversely, a non detectable constraint may not belong to a set of isolated constraints. Next lemma shows the general condition that a non detectable constraints must fulfil according to Definition 7.

Lemma 3: A sufficient condition for a subset of constraints $K \subset K_\Sigma$ to be non detectable is that there is a tuple $(K_1, \ldots, K_m)$ of m sets of constraints making up a partition $\mathcal{P}(K)$ of K such that each K_i is isolated in $K_\Sigma \setminus \bigcup_{j<i} K_j$ (K_1 is a limit case: it should be isolated in K_Σ).
The proof is presented in [19].

Figure 4 indicates the shape of a structural matrix of non detectable constraints.

It is clear that by installing sensors, i.e. adding terminal constraints in the system, a set of linked constraint can become a non-linked (i.e. the non discriminable constraints can become diagnosable) and a set of isolated constraints can become non-isolated (i.e. these non detectable constraints can become non discriminable or diagnosable).

V. COMPUTING DETECTABILITY, DISCRIMINABILITY AND DIAGNOSABILITY FROM A STRUCTURAL MODEL

According to lemma 3, there are several ways to partition a constraint set K which has to be non detectable into several parts $\mathcal{P}(K) = \{K_1, \ldots, K_m\}$ such that each K_i is isolated in $K_\Sigma \setminus \bigcup_{j<i} K_j$. The Dulmage-Mendelsohn decomposition (decompose(K)) provides a unique way to partition the constraint set K. The partition resulting from the Dulmage-Mendelsohn decomposition is included into the partition resulting from lemma 3. In this paper, the Dulmage-Mendelsohn decomposition is used for finding the optimal sensor placement.

Since discriminability and diagnosability concern the detectable constraints, there is no need to search for discriminable or diagnosable faults into the non-detectable constraints, i.e. the corresponding K_Σ^- and K_Σ^0 sets. Furthermore, non-discriminable faults are related to the linked set of constraints according Lemma 1 which means that finding all the linked set of constraints in K_Σ^+ is enough to establish the fault discriminability properties of the system.

Let K be an over-constrained set such that $K^+ = K$. Then for any linked set of constraint $K' \subseteq K$ it holds that

$$K' = (K \setminus \{k\})^0 \cup \{k\} \quad \forall k \in K' \tag{12}$$

This is due to the fact that, by removing the constraint $k \in K'$, the existing complete matching in $var(K')$ [1] turns into a perfect matching with the constraints $K' \setminus \{k\}$ and the corresponding variables, in other words, $K' \setminus \{k\} = (K' \setminus \{k\})^0$.

Algorithm 1 finds all the linked sets of constraints within an over-constrained set by applying (12). The "DM-decompose" function in the algorithm refers to the Dulmage-Mendelsohn decomposition. Note that when a linked set is found by removing a constraint we do not need repeat the procedure with the remaining linked constraints since the same linked set would be found more than once.

Algorithm 1 $\mathbb{K}_{link}$ = findLinkedBlocks(K): Compute the subsets of linked constraints in K

Require: A set of constraints K such that $K = K^+$
Ensure: The subsets of linked constraints $\mathbb{K}_{link}$
  $\mathbb{K}_{link} \leftarrow \emptyset$
  $K' \leftarrow K$
  while $K' \neq \emptyset$ **do**
    $\{k\} \leftarrow \{k\} \in K'$
    $(K^-, K^0, K^+) \leftarrow$ DM-decompose($K \setminus \{k\}$)
    if $K^0 \neq \emptyset$ **then**
      $\mathbb{K}_{link} \leftarrow \mathbb{K}_{link} \cup \{K^0 \cup \{k\}\}$
    end if
    $K' \leftarrow K' \setminus \{\{k\} \cup K^0\}$
  end while

Algorithm 2 is developed in order to determine the detectable K_{det} set, the discriminable $\mathbb{K}_{disc}$ sets and the diagnosable K_{diag} set of constraints within a structural system model. The detectable constraints are the constraints included in the over-constrained part of the the Dulmage-Mendelsohn decomposition i.e. K^+. Once the detectable set is found, then all the discriminable sets are computed by means of Algorithm 1. Finally, since all the possible discriminable sets has been found, the remaining constraintsin the detectable set are the diagnosable ones.

Algorithm 2 $(K_{det}, \mathbb{K}_{disc}, K_{diag})$ = decompose(K): Find the different subsets of constraints in K according definitions3, 4 and 5

Require: A set of constraints K
Ensure: $K_{det}, \mathbb{K}_{disc}, K_{diag}$
  $(K^-, K^0, K^+) \leftarrow$ DM-decompose(K)
  $K_{det} \leftarrow K^+$
  $\mathbb{K}_{disc} \leftarrow$ findLinkedBlocks(K^+)
  $K_{diag} = K^+ \setminus \bigcup_{K \in \mathbb{K}_{disc}^{spec}} K$

VI. OPTIMAL SENSOR PLACEMENT FOR PRACTICAL SPECIFICATIONS

A. Verification of practical diagnosis specifications

Let us consider a system modeled by a constraint set K_Σ. Let K_{det}, $\mathbb{K}_{disc}$ and K_{diag} be the diagnosis properties of a

system K_Σ, and K_{det}^{spec}, $\mathbb{K}_{disc}^{spec}$ and K_{diag}^{spec} the required practical specifications, we say that the system K_Σ fulfils:

1) detectability specifications if

$$K_{det}^{spec} \subseteq K_{det} \tag{13}$$

2) discriminability specifications if

$$|\{K^{spec} \in \mathbb{K}_{disc}^{spec} \mid K^{spec} \cap K \neq \emptyset\}| \leq 1 \quad \forall K \in \mathbb{K}_{disc} \tag{14}$$

3) diagnosability specifications if

$$K_{diag}^{spec} \subseteq K_{diag} \tag{15}$$

According to these expressions, the practical specifications are not fulfilled as long as a specified detectable constraint becomes non-detectable or two specified discriminable constraints are non-discriminable (i.e. are non-detectable or both constraints belongs to the same set $K \in \mathbb{K}_{disc}$) or a specified diagnosable constraint becomes non-diagnosable (i.e. is non-detectable or belongs to any set $K \in \mathbb{K}_{disc}$). Otherwise the practical specifications are fulfilled.

Algorithm 3 verifies whether the practical specifications are fulfilled. Given the set of model constraints K_Σ without installed sensors, a set of variables $V \subseteq var(K_\Sigma)$ to be measured and the practical specifications K_{det}^{spec}, $\mathbb{K}_{disc}^{spec}$ and K_{diag}^{spec}, the algorithm first adds the corresponding terminal constraints of V to the system constraints and computes the diagnosis properties by means of Algorithm 2. Then detectability and diagnosability are verified according (13) and (15). Finally discriminability is verified by ensuring that, for each set in $\mathbb{K}_{disc}$, there is not more that one set in $\mathbb{K}_{disc}^{spec}$ with a non-empty intersection, according to (14).

If the practical specifications are verified then the set of candidate sensors to measure V is a feasible solution for the sensor placement problem. The Algorithm 3 returns a boolean value indicating whether V is a feasible configuration.

B. Optimal sensor search

There may be several sensor placements that satisfy diagnosability specifications. In order to select the most interesting one, a criterion based on the cost of the sensor placement is considered. Let's introduce the following notations. The cost of the measurement of a variable v is denoted $C(v)$. By extension, the cost of the measurement of a set of variables V is denoted: $C(V) = \sum_{v \in V} C(v)$.

In order to describe the sensor placement method, the notions of $Vnode$ and *buffer* of $Vnodes$ are introduced. A $Vnode$ is a couple of variable sets: $Vnode = (V^a, V^b)$, where $Vnode^- = V^a$ and $Vnode^+ = V^b$. A *buffer* is a special First In First Out buffer. The basic functionalities are: *buffer*.$push(Vnode)$ and *buffer*.$pop()$. They respectively correspond to add a $Vnode$ in the buffer and get a $Vnode$ from the buffer.

The optimality criterion for a feasible sensor placement defined by $V_{measured}$ is given by: $C(V_{measured})$. Combinatorial algorithm is implemented in the placeSensor() method (algorithm 4) using a simple First In First Out buffer of nodes of variables.

Algorithm 3 feasible($K_\Sigma, V, K_{det}^{spec}, \mathbb{K}_{disc}^{spec}, K_{diag}^{spec}$) : Check whether the sensor placement satisfies the specifications

Require:
    A structural set of constraints K_Σ.
    A set of candidate sensors $V \subseteq var(K_\Sigma)$.
    The consistent specifications, $K_{det}^{spec}, \mathbb{K}_{disc}^{spec}, K_{diag}$ in K_Σ.
Ensure: true if the candidate sensors satisfy the specifications, **false** otherwise.
    $K_\Sigma' \leftarrow K_\Sigma \cup K_V$
    $(K_{det}, \mathbb{K}_{disc}, K_{diag}) = \text{decompose}(K_\Sigma')$
    if $(K_{det}^{spec} \not\subseteq K_{det})$ **then**
        return false
    end if
    if $(K_{diag}^{spec} \not\subseteq K_{diag})$ **then**
        return false
    end if
    for all $K \in \mathbb{K}_{disc}$ **do**
        $isthesecond \leftarrow$ **false**
        $K' = (\bigcup_{K'' \in \mathbb{K}_{disc}^{spec}} K'') \cap K$
        for all $K^{spec} \in \mathbb{K}_{disc}^{spec}$ **do**
            if $K^{spec} \cap K' \neq \emptyset$ **then**
                if not $isthesecond$ **then**
                    $isthesecond \leftarrow$ **true**
                else
                    return false
                end if
            end if
        end for
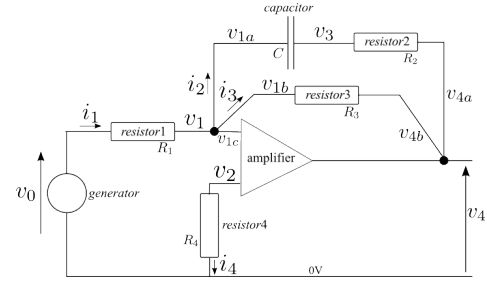   **end for**
    return true

Fig. 5. Scheme of the electronic circuit

VII. APPLICATION

The algorithm of sensor placement summarized in section VI has been applied for an electronic circuit (figure 5). This electronic circuit is modeled by the following constraints:

$k_1 : v_{1c} - v_2 = 0$ $k_8 : v_3 - v_{4a} - R_2 i_2 = 0$
$k_2 : i_1 - i_2 - i_3 = 0$ $k_9 : v_1 - v_{4b} - R_3 i_3 = 0$
$k_3 : v_1 - v_{1a} = 0$ $k_{10} : v_2 - R_4 i_4 = 0$
$k_4 : v_1 - v_{1b} = 0$ $k_{11} : v_4 - v_{4a} = 0$
$k_5 : v_1 - v_{1c} = 0$ $k_{12} : v_4 - v_{4b} = 0$
$k_6 : v_0 - v_1 = R_1 i_1$ $k_{13} : v_0 - \tilde{v_0} = 0$
$k_7 : C(v_{1a} - v_3) - \int_0^t i_2 dt = 0$

$$\tag{16}$$

with $K_\Sigma = \{k_1, k_2, k_3, k_4, k_5, k_6, k_7, k_8, k_9, k_{10}, k_{11}, k_{12}, k_{13}\}$.

Algorithm 4 searchOpt($K_\Sigma, K_{det}^{spec}, \mathbb{K}_{disc}^{spec}, K_{diag}^{spec}$)): a set of variables to be measured

Require: Specifications are consistent in K_Σ.
Require: $C()$ is defined for each variable in V_Σ
  $criteria \leftarrow C(var(K_\Sigma))$
  $V_{measured} \leftarrow var(K_\Sigma)$
  $buffer \leftarrow \emptyset$
  $V_{candidat} \leftarrow var(K_\Sigma)$
  $buffer.push(\emptyset, V_{candidat})$
  while $buffer$ in not empty **do**
    $Vnode \leftarrow buffer.pop()$
    $V_{remaining} \leftarrow Vnode^+$
    for all $v \in Vnode^+$ **do**
      $V_{selected} \leftarrow Vnode^- \cup \{v\}$
      if $C(V_{selected}) < criteria$ **then**
        if isFeasible($K_\Sigma, V_{selected}, K_{det}^{spec}, \mathbb{K}_{disc}^{spec}, K_{diag}^{spec}$)
        then
          $criteria \leftarrow C(V_{selected})$
          $V_{measured} \leftarrow V_{selected}$
        else
          $V_{remaining} \leftarrow V_{remaining} \backslash \{v\}$
          $buffer.push(V_{selected}, V_{remaining})$
        end if
      end if
    end for
  end while
  return $V_{measured}$

Suppose that the costs of the measurements are:

$$C(v_0) = C(i_1) = 1$$
$$C(v_4) = 2$$
$$C(v_1) = C(v_2) = C(v_3) = C(v_{1a}) = C(v_{1b}) =$$
$$= C(v_{1c}) = C(v_{4a}) = C(v_{4b}) = 4$$
$$C(i_2) = C(i_3) = C(i_4) = 8$$

Let consider the following practical specifications:

$$K_{det} = \{k_1, k_6, k_7, k_8, k_9, k_{10}, k_{13}\}$$
$$\mathbb{K}_{disc} = \{\{k_6\}, \{k_7\}, \{k_8\}, \{k_9\}, \{k_{10}\}, \{k_{13}\}\}$$
$$K_{diag} = \{k_1\}$$

In order to find the cheapest sensor placement that satisfies specifications, algorithm 4 is used. It yields the following results:

$$V_{measured} = \{v_0, v_2, v_3, v_4, i_1, i_4, v_{1c}\}$$

with a minimal cost $C(V^*) = 24$.

VIII. Conclusion

A new approach of sensor placement which satisfies practical specifications has been proposed. It is thus possible to specify the performances that a diagnostic system has to meet and then to compute where the sensors should be placed.

Several lemmas and algorithms are used to develop a method for sensor placement that deals with practical specifications: the set of constraints that must be diagnosable, the set of constraint sets that must be discriminable and the set of constraints that must be at least detectable. This method can be applied to a large class of systems since they are based on a structural model approach.

Algorithm for sensor placement managing practical specifications has been presented. Thanks to the proposed algorithm, cost optimal sensor placements satisfying practical diagnosability specifications is possible without designing ARR a priori. It is an important feature since it is no longer necessary to design all the possible ARRs.

References

[1] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki. *Diagnosis and Fault-tolerant Control*. Springer-Verlag, 2006.
[2] J. De Kleer and B.C Williams. Diagnosis multiple faults. *Artificial intelligence*, 32(1):97–130, 1987.
[3] A. L. Dulmage and N. S. Mendelsohn. A structure theory of bi-partite graphs of finite exterior extension. *Transactions of the Royal Society of Canada*, 53(III):1–13, 1959.
[4] P.M. Frank. Analytical and qualitative model-based fault diagnosis - a survey and some new results. *European Journal of Control*, 2:6–28, 1996.
[5] E. Frisk and M. Krysander. Sensor placement for maximum fault isolability. In *The 18th International Workshop on Principles of Diagnosis (DX-07)*, 2007.
[6] M. Krysander, J. Åslund, and M. . Nyberg. An efficient algorithm for finding minimal overconstrained subsystems for model-based-diagnosis. *IEEE Trans. Syst., Man, Cybern. A*, 38(1):197–206, 2008.
[7] F. Madron and V. Veverka. Optimal selection of measuring points in complex plants by lineair models. *AICheE*, 38(2):227–236, 1992.
[8] D. Maquin, M. Luong, and J. Ragot. Fault detection and isolation and sensor network design. *European Journal of Automation*, 31(2):393–406, 1997.
[9] K. Murota. *Systems analysis by graphs and matroids: structural solvability and controlability*. Springer-Verlag, 1987.
[10] M. Nyberg and M. Krysander. Combining ai, fdi, and statistical hypothesis-testing in a framework for diagnosis. In *IFAC Safeprocess'03*, Washington, U.S.A., 2003.
[11] R.J. Patton and J. Chen. A review of parity space approaches to fault diagnosis. In *IFAC SAFEPROCESS Symposium*, Baden-Baden, 1991.
[12] S. Ploix, S. Touaf, and J. M. Flaus. A logical framework for isolation in fault diagnosis. In *SAFEPROCESS'2003*, Washington D.C., U.S.A., 2003.
[13] S. Ploix, A. Yassine, and J.-M. Flaus. An improved algorithm for the design of testable subsystems. In *The 17th IFAC World Congress*, Seoul, Corea, 2008.
[14] A. Pothen and F. Chin-Ju. Computing the block triangular form of a sparse matrix. *ACM Transactions on Mathematical Software*, 16(4):303–324, 1990.
[15] B. Pulido and C. Alonso. Possible conflicts, arrs, and conflicts. In *13th International Workshop on Principles of Diagnosis (DX02)*, pages 122–128, May 2002.
[16] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32:57–95, 1987.
[17] P. Struss, B. Rehfus, R. Brignolo, F. Cascio, L. Console, P. Dague, P. Dubois, O. Dressler, and D. Millet. Model-based tools for the integration of design and diagnosis into a common process- a project report. In *DX'02*, Semmering, Austria, May 2-4 2002.
[18] L. Travé-Massuyès, T. Escobet, and R. Milne. Model-based diagnosability and sensor placement application to a frame 6 gas turbine subsystem. In *12th Int,Workshop on principles of diagnosis*, pages 205–212, Sansicario, Via Lattea, Italie, 2001.
[19] A. Yassine, S. Ploix, and J.-M. Flaus. A method for sensor placements taking into account diagnosability criteria. *Applied Mathematics and Computer Science*, 18(4), 2008.