# Optimal Sensor Placement for FDI using Binary Integer Linear Programming

**Albert Rosich   Ramon Sarrate   Fatiha Nejjari**

*Automatic Control Department, Universitat Politècnica de Catalunya, Rambla de Sant Nebridi, 10, 08222 Terrassa, Spain (e-mail: albert.rosich@upc.edu).*

**Abstract:** This work is devoted to find an optimal set of sensors for model-based FDI. The novelty is that binary integer linear programming is used in the optimization problem, leading to a formulation of the detectability and isolability specifications as linear inequality constraints. Furthermore, a very detailed system model is not needed since the methodology handles structural models. The approach has been successfully applied to a two-tank system, as an illustrative example.

## 1. INTRODUCTION

Fault diagnosis systems are an increasing and important topic in many industrial processes. The number of publications devoted to fault diagnosis has increased notably in the last years, as it can be seen in Blanke et al. (2006) and Gertler (1998). In Model-based Fault Diagnosis, diagnosis is basically performed from the comparison between a process model and on-line process information. Since process information is usually obtained by means of the sensors installed in the process, it is important to develop methodologies to place the correct set of sensors in the process in order to guarantee some diagnosis specifications.

In this paper, diagnosis specifications are detection and isolation of single faults. Two kind of faults are considered: *system faults*, which concern fixed components in the system, and *sensor faults* which concern sensors chosen for installation. No hardware redundancy will be considered, though the method could be easily extended to include it.

Large-scale diagnosis models may consist of many different types of descriptions, for example static/dynamic linear equations, lookup tables, logic rules, non-linear differential-algebraic equations, etc. One way to analyze such a general class of models in a general framework is to analyze the model structure. A structural model is a coarse model description, based on a bi-partite graph, that can be obtained early in the development process, without major engineering efforts. This kind of models is suitable to handle large scale systems since efficient graph-based tools can be used and does not have numerical problems. However, only best case results are obtained. More information about structural modeling applied to fault diagnosis can be found in Blanke et al. (2006).

In model-based *Fault Detection and Isolation* (FDI), faults are modeled as deviations of parameter values or unknown signals and diagnostic models are often brought back to a residual form. The main approaches to construct residuals are based on using *Analytical Redundancy Relations* (ARRs) generated either using the parity space (Staroswiecki and Comtet-Varga, 2001) or observer approaches (Nikoukhah, 1998). In Maquin et al. (1997) the sensor placement problem is solved by the analysis of a set of possible ARRs using algorithms of cycle generation in graphs. Some other results devoted to sensor placement for diagnosis using graph tools can be found in Raghuraj et al. (1999), Krysander and Frisk (2008), Commault et al. (2008), Yassine et al. (2008), Travé-Massuyès et al. (2006) and Rosich et al. (2007). All these works use a structural model-based approach and define different diagnosis specifications to solve the sensor placement problem.

In Sarrate et al. (2007), an optimal sensor placement for model-based FDI requires finding the set of all possible ARRs, considering that all possible candidate sensors are installed. Then, a set of sensors that minimizes the total cost of the network is selected such that the resulting ARRs satisfy that a pre-established set of faults can be detected and isolated. The optimization problem is casted as a Binary Integer Programming problem (Wosley, 1998), where the optimization vector states whether a sensor is installed or not and FDI specifications are translated into constraints. However, the non-linear nature of such constraints lead to a high computational complexity of the resulting optimization problem. An alternative approach is proposed in Fijany and Vatan (2006), which however involves the formulation of a non-linear objective function.

In this work both approaches are enhanced by formulating a *Binary Integer Linear Programming* (BILP) problem. The FDI specifications are formulated as linear constraints and the objective cost function is also linear, so the BILP problem can be efficiently solved by an LP-based branch-and-bound algorithm.

In this paper, as a matter of notational convention, bold and uppercase letters denote matrices, bold and lowercase letters denote vectors, and normal lowercase letters denote matrix o vector elements. Also, the $n \times n$ identity matrix is denoted by $\mathbf{I}_n$, the $n \times m$ null matrix is denoted by $\mathbf{0}_{n\times m}$ and the $i \times j$ ones matrix is denoted by $\mathbf{1}_{i\times j}$. Finally,

⋆ This work was supported in part by the Research Comission of the Generalitat of Catalunya (Grup SAC, ref. 2005SGR00537) and by CICYT (ref. DPI-2008-01996) of Spanish Ministry of Education

the number of $k$-combinations from a set of $n$ elements is denoted by $\mathcal{C}_n^k$.

In Section 2, the sensor placement problem is formulated as a BILP problem. The main contribution of this paper is developed in Section 3, where FDI specifications are formulated as binary linear constraints. Section 4 summarizes the BILP formulation of the sensor placement problem and extends it to include ARRs optimal selection. Next, Section 5 describes an application of the sensor placement methodology to a two-tank system. Finally, Section 6 remarks the conclusions and future extensions of the present work.

## 2. SENSOR PLACEMENT FOR FDI AS A BILP PROBLEM

Similar to optimization based on linear programming, a standard optimization problem using BILP can be formulated as a linear objective function and constrained by linear inequality constraints. This is expressed as:

$$\min_{\mathbf{x}} \quad \mathbf{c}^T\mathbf{x} \quad \text{subject to:} \tag{1}$$

$$\mathbf{A}\mathbf{x} \leq \mathbf{b} \tag{2}$$

$$\mathbf{x} \text{ is binary} \tag{3}$$

The main constraint is that any element of the optimization vector $\mathbf{x}$ must be binary, i.e. $\forall x \in \mathbf{x} : x \in \{0, 1\}$. Moreover, matrix $\mathbf{A}$ and vector $\mathbf{b}$ form the linear inequality constraints. Finally, $\mathbf{c}$ is a cost vector of the linear objective function.

The sensor placement problem for FDI developed in this article can be summarized as finding a minimal set of sensors to be installed in the system such that faults can be detected and isolated.

The sensor placement problem can be formulated as a BILP problem where the set of candidate sensors to be installed is represented by the optimization variable vector. This means that if the entry $x \in \mathbf{x}$ equals 1, the corresponding sensor must be installed whereas if $x$ equals 0, the sensor does not need to be installed.

Furthermore, by means of the $\mathbf{c}$ vector a cost can be assigned to each sensor in order to find an optimal solution based in some criterion, e.g. minimal cardinality, minimal economical price, etc.

Using this formulation, all constraints must be written as linear inequalities. Thus, for sensor placement for FDI, fault detectability and fault isolability constraints must be expressed as in (2). Section 3 is devoted to show how FDI specifications can be expressed as linear inequalities.

For the sake of simplicity, in the following development, it is assumed that there exists a sensor configuration such that all faults concerned are fully detectable and isolable among them. Some guidelines will be given in Section 4 allowing this assumption to be dropped. Note that verifying whether this assumption is fulfilled can be accomplished by just checking which faults are detectable and isolable when all the candidate sensors are installed in the system.

## 3. CONSTRAINTS FORMULATION

In model-based FDI, ARRs are used to check the consistency between the model and system measurements. An ARR can be obtained from a subset of model equations by eliminating unknown variables through the convenient manipulation of the equations. Therefore, an ARR is an expression that only depends on known (measured) variables. Structural analysis theory has been extensively used in model-based FDI to generate the ARRs from the model equations (Blanke et al., 2006, Travé-Massuyès et al., 2006, Krysander et al., 2008).

It is straightforward to establish a relation between the ARRs set and the set of known variables. This relation is represented by a bi-adjacency matrix where the row set is the ARRs set and the column set is the sensors set. Let $n$ be the number of ARRs and $k$ be the number of candidate sensors, then the biadjacency matrix $\mathbf{M} = [m_{ij}]$ is a $n \times k$ matrix defined by:

$$m_{ij} = \begin{cases} 1 & \text{if ARR } i \text{ depends on sensor } j \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Knowing which equations are related to a certain ARR it is possible to determine the set of faults that an ARR is sensitive to. This ARR-fault relation is known in the literature by the *Fault Signature Matrix* (FSM) (Blanke et al., 2006). Let $l$ be the number of system faults to be diagnosed, then the biadjacency matrix $\mathbf{F} = [f_{ij}]$ is a $n \times l$ matrix defined by:

$$f_{ij} = \begin{cases} 1 & \text{if fault } j \text{ may affect ARR } i \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Remark that, the FSM $F$ will refer to system faults. A similar FSM will be considered for sensor faults, which will be denoted by $\mathbf{F_q} = [f_{q_{ij}}]$ and easily deduced from $\mathbf{M}$ as $\mathbf{F_q} = \mathbf{M}$.

In the following sections, the constraint formulation proposed in Sarrate et al. (2007) will be revised and further developed to formulate fault detectability and isolability specifications as linear constraints.

A small example is used through the paper in order to clarify how this formulation should be applied. This example consists of a set of five ARRs, a set of three candidate sensors and a set of two system faults. The corresponding matrices $\mathbf{M}$, $\mathbf{F}$, $\mathbf{F_q}$ are depicted in (6).

$$\mathbf{M} = \mathbf{F_q} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \tag{6}$$

### *3.1 ARR selector constraint*

Given a subset of installed sensors there may be some ARRs that are not valid since they depend on non-installed candidate sensors. Let $\mathbf{q} = [\, q_1 \cdots q_k \,]^T$ be the binary vector that denotes whether a sensor is installed or not.

Then, an ARR $i$ is called *non-selectable* if there is a sensor $j$ such that $q_j = 0$ and $m_{ij} = 1$. This motivates the *ARR selector*:

$$\rho_i = \prod_{j=1}^{k} [m_{ij} q_j + (1 - m_{ij})] \tag{7}$$

Note that $\rho_i$ is a binary variable such that if ARR $i$ is non-selectable then $\rho_i$ equals 0. However, expression (7) is non-linear and can not be casted as a constraint in (2). In order to do so, remark that inequality (8) holds as long as ARR $i$ is non-selectable.

$$\sum_{j=1}^{k} [m_{ij} q_j + (1 - m_{ij})] < k \tag{8}$$

Next, introducing the binary variable $\rho_i$ in inequality (8), the expression (9) is obtained, which is equivalent to the *ARR selector* in (7).

$$\sum_{j=1}^{k} [m_{ij} q_j + (1 - m_{ij})] - k\rho_i \geq 0 \tag{9}$$

Note however that expression in (9) implies that:

$$\text{ARR } i \text{ is not valid} \rightarrow \rho_i = 0$$

whereas the reverse is not true. This means that $\rho_i$ can be viewed as a *dummy* variable in the optimization problem. This variable is forced to zero as long as the corresponding ARR is non-selectable.

Now, equation (9) is linear. Therefore, it is suitable for BILP formulation. Equation (9) can be extended to all the $n$ ARRs, and written in vector form as:

$$\begin{bmatrix} m_{i1} & \cdots & m_{ik} \\ \vdots & \ddots & \vdots \\ m_{n1} & \cdots & m_{nk} \end{bmatrix} \begin{bmatrix} q_1 \\ \vdots \\ q_k \end{bmatrix} - k \begin{bmatrix} \rho_1 \\ \vdots \\ \rho_n \end{bmatrix} + \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} \geq \mathbf{0}_{n\times 1} \tag{10}$$

where $\beta_i = \sum_{j=1}^{k}(1 - m_{ij})$ for $i = \{1, \cdots, n\}$. Finally, equation (10) can be written in a compact form as:

$$\begin{bmatrix} -\mathbf{M} & k\mathbf{I}_n \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\rho} \end{bmatrix} \leq \boldsymbol{\beta} \tag{11}$$

where vector $\boldsymbol{\rho} = \begin{bmatrix} \rho_1 & \cdots & \rho_n \end{bmatrix}^T$ is the set of *ARR selectors* and $\boldsymbol{\beta} = \begin{bmatrix} \beta_1 & \cdots & \beta_n \end{bmatrix}^T$ is a vector of coefficients.

Expression (11) has the same form as (2). Furthermore, the optimization variable vector is augmented by including the *ARR selector*, i.e. $\mathbf{x} = [\mathbf{q}^T \quad \boldsymbol{\rho}^T]^T$.

Given the example in (6), equation (11) becomes:

$$\begin{bmatrix} 0 & -1 & -1 & 3 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 3 & 0 & 0 \\ -1 & -1 & -1 & 0 & 0 & 0 & 3 & 0 \\ -1 & 0 & -1 & 0 & 0 & 0 & 0 & 3 \end{bmatrix} \cdot \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \\ \rho_5 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 2 \\ 2 \\ 0 \\ 1 \end{bmatrix} \tag{12}$$

### *3.2 Fault detectability constraint*

A fault is structurally detectable if there exists at least one ARR that can be affected by this fault. Hence, the *ARR selector* plays an important role since all non-selectable ARRs must be rejected from the detectability study.

Since system and sensor faults are considered in this paper, the number of equations needed to check fault detectability is $l + k$. Next, both type of constraints are deduced.

*System faults detectability*

Given a fault $j$, the following expression holds:

$$\text{a system fault } j \text{ is detectable} \leftrightarrow \sum_{i=1}^{n}(f_{ij}\rho_i) \geq 1 \tag{13}$$

Equation (13) can be extended to all system faults, and written in compact form as:

$$\begin{bmatrix} \mathbf{0}_{l\times k} & -\mathbf{F}^T \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\rho} \end{bmatrix} \leq -\mathbf{1}_{l\times 1} \tag{14}$$

Therefore, the set of system faults is detectable if constraint (14) holds.

Following with the example in (6), equation (14) becomes:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & -1 \end{bmatrix} \cdot \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \\ \rho_5 \end{bmatrix} \leq \begin{bmatrix} -1 \\ -1 \end{bmatrix} \tag{15}$$

*Sensor faults detectability*

A similar expression to (13) is used for sensor fault detectability:

$$\text{a sensor fault } j \text{ is detectable} \leftrightarrow \sum_{i=1}^{n}(f_{q_{ij}}\rho_i) \geq q_j \tag{16}$$

Note that for a non-installed sensor, the right hand side of inequality (16) becomes 0, meaning that no detectability property is expected for this sensor fault. However, as long as a sensor is chosen for installation, equation (16) becomes similar to (13).

Equation (16) can be extended to all sensor faults, and written in compact form as:

$$\begin{bmatrix} \mathbf{I}_k & -\mathbf{F_q}^T \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\rho} \end{bmatrix} \leq \mathbf{0}_{k\times 1} \quad (17)$$

Thus, the sensor fault detectability constraint for the example in (6) is:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 & -1 & -1 \\ 0 & 1 & 0 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & -1 & 0 & -1 & -1 & -1 \end{bmatrix} \cdot \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \\ \rho_5 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (18)$$

*3.3 Fault isolability constraint*

Two faults are structurally isolable if their corresponding signatures in the FSM are different. This is true as long as ARR-based exoneration is assumed (Travé-Massuyès et al., 2006).

Since system and sensor faults are considered in this paper, the number of equations needed to check fault isolability is $\mathcal{C}_2^{l+k}$. Next, three types of constraints are deduced depending on whether system or sensor faults are considered.

*Fault isolability between system faults*

Given two system faults $j_1$ and $j_2$, inequality (19) holds as long as their signatures in the FSM are different.

$$\begin{matrix}\text{two system faults} \\ j_1 \text{ and } j_2 \text{ are isolable}\end{matrix} \leftrightarrow \sum_{i=1}^{n} |f_{ij_1} - f_{ij_2}|\rho_i \geq 1 \quad (19)$$

Equation (19) can be extended to any combination of two system faults, and written in compact form as:

$$\begin{bmatrix} \mathbf{0}_{\mathcal{C}_2^l\times k} & -\mathbf{F_{I_1}}^T \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\rho} \end{bmatrix} \leq -\mathbf{1}_{\mathcal{C}_2^l\times 1} \quad (20)$$

where $\mathbf{F_{I_1}} = [f_{I_1 im}]$ is a $n \times \mathcal{C}_2^l$ matrix with:

$$f_{I_1 im} = |f_{ij_1} - f_{ij_2}| \quad \forall j_1, j_2 \in \{1, \ldots, l\} : j_1 < j_2 \quad (21)$$

where $m$ indexes in lexicographical order the $\mathcal{C}_2^l$ system faults combinations.

Following with the example in (6), equation (20) becomes:

$$\begin{bmatrix} 0 & 0 & 0 & -1 & -1 & -1 & 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \\ \rho_5 \end{bmatrix} \leq \begin{bmatrix} -1 \end{bmatrix} \quad (22)$$

*Fault isolability between system faults and sensor faults*

Isolability involving a sensor fault depends on whether the corresponding sensor is considered for installation or not. So, the condition for isolability between a system fault and a sensor fault can be stated as:

$$\begin{matrix}\text{a system fault } j_1 \text{ and a} \\ \text{sensor fault } j_2 \text{ are isolable}\end{matrix} \leftrightarrow \sum_{i=1}^{n} |f_{ij_1} - f_{q_{ij_2}}|\rho_i \geq q_{j_2} \quad (23)$$

Note that for a non-installed sensor, the right hand side of inequality (23) becomes 0, meaning that no isolability property is expected for this sensor fault. However, as long as a sensor is chosen for installation, equation (23) becomes similar to (19).

Equation (23) can be extended to any pair of system fault and sensor fault, and written in compact form as:

$$\begin{bmatrix} \mathbf{G_2} & -\mathbf{F_{I_2}}^T \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\rho} \end{bmatrix} \leq \mathbf{0}_{l\cdot k\times 1} \quad (24)$$

where $\mathbf{G_2}$ is the following $l \cdot k \times k$ matrix:

$$\mathbf{G_2} = \begin{bmatrix} I_k & I_k & \cdots & I_k \end{bmatrix}^T \quad (25)$$

and $\mathbf{F_{I_2}} = [f_{I_2 ip}]$ is a $n \times l \cdot k$ matrix with:

$$f_{I_2 ip} = |f_{ij_1} - f_{q_{ij_2}}| \quad \begin{cases} \forall j_1 \in \{1, \ldots, l\} \\ \forall j_2 \in \{1, \ldots, k\} \end{cases} \quad (26)$$

where $p$ indexes in lexicographical order the cartesian product of the system faults set and the sensor faults set.

Remark that matrix $\mathbf{G_2}$ is used to involve the corresponding sensor in (23) according to the index $j_2$ used to build matrix $\mathbf{F_{I_2}}$.

Remind that in the example given in (6), $k = 3$ and $l = 2$. Therefore equation (24) becomes:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 1 & 0 & -1 & -1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 0 & 0 & -1 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \\ \rho_5 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (27)$$

*Fault isolability between sensors faults*

Now the isolability condition involves two sensor faults, so it depends on whether both sensors are considered for installation. The condition for isolability between two sensor faults can be stated as the following non-linear inequality:

$$\sum_{i=1}^{n} |f_{q_{ij_1}} - f_{q_{ij_2}}|\rho_i \geq q_{j_1} q_{j_2} \quad (28)$$

Note that as long as both sensors are not selected for installation, the right hand side of inequality (28) becomes 0, meaning that no isolability property is expected between their corresponding sensor faults. However, as long as both sensor are selected for installation, equation (28) becomes similar to (19).

Equation (28) can be transformed into the equation in (29), which is an equivalent linear constraint.

$$\begin{matrix}\text{two sensors faults}\\ j_1 \text{ and } j_2 \text{ are isolable}\end{matrix} \leftrightarrow \sum_{i=1}^{n} |f_{q_{ij_1}} - f_{q_{ij_2}}|\rho_i \geq q_{j_1} + q_{j_2} - 1 \tag{29}$$

Note that the left hand side of the equation in (29) is non-negative, so this constraint will only become active when both sensors are selected for installation.

Equation (29) can be extended to any combination of two sensor faults, and written in compact form as:

$$\begin{bmatrix} \mathbf{G_3} & -\mathbf{F_{I_3}}^T \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\rho} \end{bmatrix} \leq \mathbf{1}_{\mathcal{C}_2^k \times 1} \tag{30}$$

where $\mathbf{G_3}$ is the following $\mathcal{C}_2^k \times k$ matrix:

$$\mathbf{G_3} = \begin{bmatrix} 1 & & & \\ \vdots & & \mathbf{I}_{k-1} & \\ 1 & & & \\ 0 & 1 & & \\ \vdots & \vdots & \mathbf{I}_{k-2} & \\ 0 & 1 & & \\ & & \vdots & \\ 0 & \cdots 0 & 1 & 1 \end{bmatrix} \tag{31}$$

and $\mathbf{F_{I_3}} = [f_{I_3 ir}]$ is a $n \times \mathcal{C}_2^k$ matrix with:

$$f_{I_3 ir} = |f_{q_{ij_1}} - f_{q_{ij_2}}| \quad \forall j_1, j_2 \in \{1, \ldots, k\} : j_1 < j_2 \tag{32}$$

where $r$ indexes in lexicographical order the $\mathcal{C}_2^k$ sensor faults combinations.

Remark that matrix $\mathbf{G_3}$ is used to involve the corresponding pair of sensors in (29) according to the indexes $j_1$ and $j_2$ used to build matrix $\mathbf{F_{I_3}}$ in (32).

Finally, the last constraint is applied to the example in (6), so equation (30) becomes:

$$\begin{bmatrix} 1 & 1 & 0 & -1 & -1 & 0 & 0 & -1 \\ 1 & 0 & 1 & -1 & -1 & -1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & -1 & 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \\ \rho_5 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{33}$$

## 4. PROBLEM FORMULATION

### *4.1 Sensor placement optimization*

Once detectability and isolability constraints have been introduced, the optimal sensor placement for FDI can be formally presented. The full problem is formulated as:

$$\min_{[\mathbf{q}^T \quad \boldsymbol{\rho}^T]} \quad [\mathbf{c}^T \quad \mathbf{0}_{1\times n}] \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\rho} \end{bmatrix} \quad \text{subject to:} \tag{34}$$

$$\begin{bmatrix} -\mathbf{M} & k\mathbf{I}_n \\ \mathbf{0_{l\times k}} & -\mathbf{F}^T \\ \mathbf{I}_k & -\mathbf{F_q}^T \\ \mathbf{0}_{\mathcal{C}_2^l \times \mathbf{k}} & -\mathbf{F_{I_1}}^T \\ \mathbf{G_2} & -\mathbf{F_{I_2}}^T \\ \mathbf{G_3} & -\mathbf{F_{I_3}}^T \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\rho} \end{bmatrix} \leq \begin{bmatrix} \boldsymbol{\beta} \\ -\mathbf{1}_{l\times 1} \\ \mathbf{0}_{k\times 1} \\ -\mathbf{1}_{\mathcal{C}_2^l\times 1} \\ \mathbf{0}_{l\cdot k\times 1} \\ \mathbf{1}_{\mathcal{C}_2^k\times 1} \end{bmatrix} \tag{35}$$

$$[\mathbf{q}^T \quad \boldsymbol{\rho}^T] \text{ is binary} \tag{36}$$

Constraint (35) is the concatenation of (11), (14), (17), (20), (24) and (30) respectively, where all the matrices involved have been previously defined.

The number of rows (*i.e.*, constraints) in (35) is the following:

- The *ARR selector* constraints (11) involve $n$ rows.
- The detectability constraints (14) and (17) involve $l + k$ rows.
- The isolability constraints (20), (24) and (30) involve $\mathcal{C}_2^l + l \cdot k + \mathcal{C}_2^k = \mathcal{C}_2^{l+k}$ rows.

The cost vector of the objective function is extended as a result of including $\boldsymbol{\rho}$ in the variable vector. Since the goal is optimizing the set of sensors, the costs related to the *ARR selector*, $\boldsymbol{\rho}$, are set to zero. Hence, $\boldsymbol{\rho}$ is regarded as a *dummy* vector.

Following with the example in (6), the sensor placement problem for FDI is formulated as the BILP problem in (34)-(36), taking into account the constraints obtained in the previous sections. The following cost vector is considered $[1\ 1\ 1\ 0\ 0\ 0\ 0\ 0]^T$. Thus, the objective function seeks the minimization of the sensor set cardinality. The optimization problem is solved using the `bintprog` function in Matlab. The result is the optimization vector $[1\ 0\ 1\ 0\ 1\ 1\ 0\ 1]^T$. This means that sensors $q_1$ and $q_3$ are required for installation and that selectable ARRs 2, 3 and 5 are sufficient to attain fault detectability and isolability among system and sensor faults.

In this example, a solution has been found that satisfies detectability and isolability of all concerned faults, following the assumption stated in Section 2. However, this assumption could not always hold. If detectability and isolability of all concerned faults are not attainable with any sensor configuration, then maximum detectability and isolability specifications should be determined. Then, the rows in (35) that involve undetectable faults or non-isolable pairs of faults should be removed from the constraints set in order to make the optimization problem feasible.

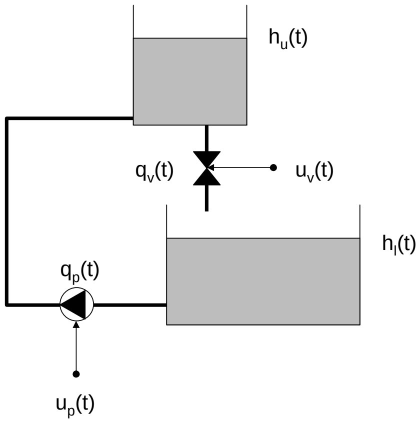


Fig. 1. Two-tank system

### *4.2 Sensor placement optimization with minimal set of ARR*

The BILP optimization stated in (34)-(36) can be extended to optimize the selected ARRs. This requires setting a cost for each ARR in the cost vector of the objective function. Therefore, the optimization problem will be expressed as follows:

$$\min_{[\mathbf{q}^T \quad \boldsymbol{\rho}^T]} \quad [\mathbf{c}^T \quad \mathbf{c}'^T] \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\rho} \end{bmatrix} \quad \text{subject to: (35) and (36)} \tag{37}$$

where $\mathbf{c}'$ is a column-vector of $n$ ARR costs.

There exist several reasons for optimizing the set of ARRs. For instance, minimizing the number of chosen ARRs, minimizing their complexity in terms of the number of equations involved in their computation, or even, maximizing their sensitivity or robustness.

Usually, the main goal will be to optimize the sensors set and then, once a minimal sensor set is ensured, optimize the ARR set. Thus, given $\mathbf{c}' = \begin{bmatrix} c'_1 & \cdots & c'_n \end{bmatrix}^T$, condition (38) must be fulfilled to satisfy this requirement.

$$\sum_{i=1}^{n} c'_i < c_j \quad \forall c_j \in \mathbf{c}, j \in \{1, \ldots, k\} \tag{38}$$

## 5. APPLICATION TO A TWO-TANK SYSTEM

In this section, the BILP formulation developed in the previous sections is illustrated through a two-tank system example. The same aplication has already been used in Sarrate et al. (2007) when solving the sensor placement problem for FDI, but using non-linear constrains.

The system is made up of two tanks interconnected by a pump and a valve. A schematic representation of the system is shown in Fig. 1.

The system can be equipped with two level sensors measuring liquid heights in the tanks $h_u$ and $h_l$, and two flowrate sensors measuring $q_p$ and $q_v$. The inputs variables $u_p$ and $u_v$ can also be measured. The flow in the pump, $q_p$, depends on both levels, $h_u$ and $h_l$, and the input to the pump, $u_p$. The flow in the valve, $q_v$, depends on the upper tank level, $h_u$, and the input to the valve, $u_v$.

Table 1. Incidence Matrix

| | Unknown variables | | | | | |
|---|---|---|---|---|---|---|
| | $h_u$ | $h_l$ | $q_p$ | $q_v$ | $u_p$ | $u_v$ |
| $e_1$ | 1 | | 1 | 1 | | |
| $e_2$ | | 1 | 1 | 1 | | |
| $e_3$ | 1 | 1 | 1 | | 1 | |
| $e_4$ | 1 | | | 1 | | 1 |
| $e_5$ | 1 | | | | | |
| $e_6$ | | 1 | | | | |
| $e_7$ | | | 1 | | | |
| $e_8$ | | | | 1 | | |
| $e_9$ | | | | | 1 | |
| $e_{10}$ | | | | | | 1 |

Table 2. Faults of the Two-Tank System

| Fault | Description | Affected equation |
|---|---|---|
| $f_u$ | upper tank leak | $e_1$ |
| $f_l$ | lower tank leak | $e_2$ |
| $f_{h_u}$ | wrong upper tank level sensor reading | $e_5$ |
| $f_{h_l}$ | wrong lower tank level sensor reading | $e_6$ |
| $f_{q_p}$ | wrong pump flow sensor reading | $e_7$ |
| $f_{q_v}$ | wrong valve flow sensor reading | $e_8$ |
| $f_{u_p}$ | wrong pump control input sensor reading | $e_9$ |
| $f_{u_v}$ | wrong valve control input sensor reading | $e_{10}$ |

The non-linear dynamic equations describing the process behavior and the sensor equations that relate the unknown variables with the measurements are all gathered in (39). $S_u$ and $S_l$ are the upper and lower tank sections respectively. $h_a$ is the height difference between both tanks. $c_p$ and $c_v$ are the pump and valve constants, respectively. Finally, $\rho$ is the liquid density and $g$ is the standard gravity.

$$\begin{aligned}
& e_1 : \dot{h}_u = \frac{1}{S_u}(q_p - q_v) \\
& e_2 : \dot{h}_l = \frac{1}{S_l}(q_v - q_p) \\
& e_3 : q_p = c_p sgn(\Delta p)\sqrt{|\Delta p|} \\
& \qquad \text{with } \Delta p \equiv f_p(u_p) - \rho g(h_a + h_u - h_l) \\
& e_4 : q_v = c_v f_v(u_v)\sqrt{\rho g h_u} \\
& e_5 : h_u = h_{u,measured} \\
& e_6 : h_l = h_{l,measured} \\
& e_7 : q_p = q_{v,measured} \\
& e_8 : q_v = q_{p,measured} \\
& e_9 : u_p = u_{v,measured} \\
& e_{10} : u_v = u_{p,measured}
\end{aligned} \tag{39}$$

The incidence matrix describing the structural model that corresponds to (39) is depicted in Table 1.

Six sensor faults are considered, as well as two system faults: leaks in the upper and lower tanks. Table 2 lists all possible faults along with their corresponding descriptions, and the equation affected by them.

Given the structural model depicted in Table 1, 35 ARRs can be found. For further information on methods devoted to finding ARRs see Krysander et al. (2008), Travé-Massuyès et al. (2006) and Pulido and Gonzalez (2004). Then, taking into account which equation is related with

Table 3. Optimal set of ARRs

| | $e_1$ | $e_2$ | $e_3$ | $e_4$ | $e_5$ | $e_6$ | $e_7$ | $e_8$ | $e_9$ | $e_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $ARR_9$ | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| $ARR_{18}$ | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| $ARR_{23}$ | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| $ARR_{27}$ | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| $ARR_{32}$ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |

Table 4. FSM for the optimal set of ARRs

| | $f_u$ | $f_l$ | $f_{h_l}$ | $f_{q_v}$ | $f_{u_p}$ | $f_{u_v}$ |
|---|---|---|---|---|---|---|
| $ARR_9$ | 0 | 1 | 1 | 1 | 1 | 1 |
| $ARR_{18}$ | 1 | 0 | 1 | 1 | 1 | 1 |
| $ARR_{23}$ | 1 | 1 | 1 | 1 | 0 | 1 |
| $ARR_{27}$ | 1 | 1 | 1 | 1 | 1 | 0 |
| $ARR_{32}$ | 1 | 1 | 0 | 1 | 1 | 1 |

each sensor or fault, matrices $M$, $F$ and $F_q$ can be extracted [1].

If all candidate sensors were installed, it would be straightforward to check that all faults are detectable and isolable (assuming ARR-based exoneration): it suffices to verify that all columns in $F$ and $F_q$ have at least a '1', and that every possible pair of columns is different. So, an optimal sensor placement problem can be posed, since it should have at least that feasible solution.

The following sensor cost $\mathbf{c} = [\,140\ 100\ 135\ 130\ 145\ 110\,]^T$ is taken, with the costs being assigned following the order of measurable variables given in Table 1. Regarding the ARRs set, a cost is assigned according to the number of system equations (*i.e.*, $e_1$,$e_2$,$e_3$,$e_4$) that each ARR relates with. This cost tries to penalize the complexity of ARRs, in terms of number of systems equations involved. Remark that condition (38) is fulfilled by every sensor cost, since for this example $\sum_{i=1}^{35} c'_i = 91$.

The BILP optimization is solved using the bintprog function in Matlab. The result is $\mathbf{q}* = [\,0\ 1\ 0\ 1\ 1\ 1\,]^T$. Therefore, the optimal subset of sensor is $\{h_l, q_v, u_p, u_v\}$. Moreover, an optimal subset of ARRs is found, that depends on these sensors and guarantees fault detectability and isolability, see Table 3. The corresponding FSM is shown in Table 4. Remark that all faults are detectable and isolable.

## 6. CONCLUSIONS

In this work, a new methodology to solve the sensor placement problem for FDI has been addressed. The sensor placement problem has been presented formally as a binary variable problem. The novelty is that BILP standard formulation is used, therefore standard algorithms to solve BILP optimization can be used. The advantage is that these algorithms are deeply developed and their branch and bound search is well-studied, leading to a fast resolution in the majority of cases. Even so, the worst case scenario would lead to a search of all the $2^n$ possible binary vectors of $n$ elements. However, the authors believe that the use of a structural model together with the well-developed optimization solver tools existing in the market make this approach suitable for handling large-scale systems in many cases.

[1] Due to the size of such matrices (*i.e.*, 35 rows), they are not shown in the paper.

Furthermore, the ARRs optimization has been included in the sensor placement problem with no extra effort, so that the most convenient set of ARRs are also selected.

The authors are aware that the ARR-based exoneration assumption may be unrealistic in many cases. However, this method could be easily extended to the case when a residual does not always cross the threshold at fault occurrence. Fault isolability should then be stated as in Krysander and Frisk (2008). It should be reformulated as a linear binary constraint, and included in the optimization problem.

The sensor placement problem based on a BILP formulation could also be extended to diagnostic specifications other than fault detectability and isolability (e.g. robustness), as long as they could be formulated as linear inequality constraints.

The main drawback of the present approach is that the complete ARRs set must be provided beforehand. It is known that generating the whole set of ARRs is a computationally complex task. Future works might be improved by solving the BILP optimization without the need of generating such ARRs set. An incremental ARR generation approach (Rosich et al., 2007) or an approach based on the Dulmage-Mendelsohn decomposition (Krysander and Frisk, 2008) could be applied, for instance.

## REFERENCES


M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki. *Diagnosis and Fault-Tolerant Control*. Springer, 2nd edition, 2006.

C. Commault, J. M. Dion, and S. Y. Agha. Structural analysis for the sensor location problem in fault detection and isolation. *Automatica*, 44(8):2074–2080, aug 2008.

Amir Fijany and Farrokh Vatan. A new efficient algorithm for analyzing and optimizing the system of sensors. In *Proc. 2006 IEEE Aerospace Conference*, Big Sky, Montana, USA, March 4–11, 2006.

Janos Gertler. *Fault Detection and Diagnosis in Engineering Systems*. Marcel Dekker, Inc., New York, 1998.

M. Krysander and E. Frisk. Sensor placement for fault diagnosis. *IEEE Trans. Syst., Man, Cybern. A*, 38(6): 1398–1410, 2008.

M. Krysander, J. Åslund, and M. Nyberg. An efficient algorithm for finding minimal over-constrained subsystems for model-based diagnosis. *IEEE Trans. Syst., Man, Cybern. A*, 38(1), 2008.

D. Maquin, M. Luong, and J. Ragot. Fault detection and isolation and sensor network design. *Europ. J. Autom.*, 31(13):396–406, 1997.

R. Nikoukhah. A new methodology for observer design and implementation. *IEEE Trans. Automat. Contr.*, 43 (2):229–234, 1998.

B. Pulido and C. A. Gonzalez. Possible conflicts: a compilation technique for consistency-based diagnosis. *IEEE Trans. Syst., Man, Cybern. B*, 34(5):2192–2206, October 2004.

R. Raghuraj, M. Bhushan, and R. Rengaswamy. Locating sensors in complex chemical plants based on fault diagnostic observability criteria. *AIChE J.*, 45(2):310–322, February 1999.

A. Rosich, R. Sarrate, V. Puig, and T. Escobet. Efficient optimal sensor placement for model-based FDI using and incremental algorithm. In *Proc. 46th IEEE Conference on Decision and Control*, pages 2590–2595, New Orleans, USA, December 12–14, 2007.

R. Sarrate, V. Puig, T. Escobet, and A. Rosich. Optimal sensor placement for model-based fault detection and isolation. In *Proc. 46th IEEE Conference on Decision and Control*, pages 2584–2589, New Orleans, USA, December 12–14, 2007.

M. Staroswiecki and G. Comtet-Varga. Analytical redundancy relations for fault detection and isolation in algebraic dynamic systems. *Automatica*, 37(5):687–699, 2001.

L. Travé-Massuyès, T. Escobet, and X. Olive. Diagnosability analysis based on component supported analytical redundancy relations. *IEEE Trans. Syst., Man, Cybern. A*, 36(6):1146–1160, 2006.

Laurence A. Wosley. *Integer Programming.* John Wiley & Sons, New York, USA, 1998.

A. A. Yassine, S. Ploix, and J. M. Flaus. A method for sensor placement taking into account diagnosability criteria. *Int. J. Appl. Math. Comput. Sci.*, 18(4):497–512, 2008.