I&EC research
Industrial & Engineering Chemistry Research

Article

How To Improve the Model Partitioning in a DSS for Instrumentation Design

Alejandro O. Domancich, Marcelo Durante, Sebastia#n Ferraro,
Patricia Hoch, Ne#lida B. Brignole, and Ignacio Ponzoni

Ind. Eng. Chem. Res., **2009**, 48 (7), 3513-3525• DOI: 10.1021/ie800449t • Publication Date (Web): 06 March 2009

Downloaded from http://pubs.acs.org on March 26, 2009

Industrial & Engineering Chemistry Research is published by the American
Chemical Society. 1155 Sixteenth Street N.W., Washington, DC 20036

How To Improve the Model Partitioning in a DSS for Instrumentation Design

Alejandro O. Domancich,[†,‡] Marcelo Durante,[‡] Sebastián Ferraro,[§,||] Patricia Hoch,[*,†] Nélida B. Brignole,[†,‡] and Ignacio Ponzoni[†,‡]

Planta Piloto de Ingeniería Química – CONICET, Complejo CRIBABB, Camino La Carrindanga km.7 CC 717, Bahía Blanca, Argentina, Laboratorio de Investigación y Desarrollo en Computación Científica (LIDeCC), Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Av. Alem 1253, 8000, Bahía Blanca, Argentina, Departamento de Matemática, Universidad Nacional del Sur, Av. Alem 1253, 8000, Bahía Blanca, Argentina, and Instituto de Matemáticas y Física Fundamental, CSIC, Serrano 123, 28006 Madrid, Spain

In this Article, we present an extension to the observability analysis algorithm known as Direct Method (DM), which is an essential piece of a Decision Support System (DSS), whose purpose is to help engineers in the instrumentation design of industrial process plants. The main goal of this new version of DM algorithm is to improve the global performance of the DSS. The new variant, known as Extended Direct Method (EDM), incorporates a novel module that quantifies the degree of nonlinearity associated with the variables and equations of the mathematical model for the process plant. This new quantification procedure conforms a heuristic search, which guides the traditional DM toward a decomposition of the mathematical model in subsystems with the smallest possible degree of nonlinearity. The obtained results show that the EDM is able to reduce the nonlinearity of the subsystems obtained through the observability analysis procedure. At the same time, it can successfully increase the generated quantity of minimum size subsystems, thus also contributing to improve the DSS's global performance.

1. Introduction

Chemical process plants are usually constituted by a great amount of industrial units, which are interconnected by material streams that enter and leave the equipment. To control these processes, it is necessary to have a well-chosen group of measurement sensors, which are located in diverse sectors of the plant. The quantity and location of the sensors employed to monitor a certain process have direct influence on the safety policies and the degree of knowledge that anybody can have, thus becoming fundamental to obtain an optimal sensor distribution. It is then desirable to reach a high degree of knowledge of the plant, which is necessary to monitor its correct operation in an efficient way, which redounds to economic benefits and leads to the improvement of the plant's safety conditions.[1−3]

To carry out this task, it is necessary to count on mathematical models for each item of equipment that integrates the chemical plant.[4] They relate all of the variables that take part in the process using the algebraic equations coming from mass and energy balances, equilibrium relationships, and thermodynamic equations that represent the plant's steady-state operation.[5,6] The process variables are basically divided into two types: the measurable ones and the unmeasurable ones. Measurable variables are those whose value can be obtained from the installation of a measurement sensor in the appropriate place. Unmeasurable variables are those whose value is directly unknown because there is no sensor capable to measure them. Therefore, if their value is indispensable to be learnt, it should be calculated from the mathematical model together with the process data available. The main goal of the analysis then is to determine the optimal group of sensors (i.e., variables to be measured) that allows the maximization of the degree of knowledge for plant operation, quantified by the amount of model variables whose value will be known starting from a proposed instrumentation layout. The objective also contemplates an attempt to minimize the cost of acquisition and installation of the sensors.

Because of the great amount of chemical units that compose a process plant, the resulting mathematical models are large, making the use of computational tools fundamental to carry out the instrumentation design task.[7−9] In this context, the development of a Decision Support System (DSS) was proposed,[10] which approaches this task by means of two central procedures: the observability analysis (OA) and the redundancy analysis (RA).[11] The objective of the OA is to individualize those variables whose values can be calculated from a set of sensors combined on the basis of the mathematical model of the plant.[12] The classification of the variables into observable (calculable) ones and unobservable ones then is carried out. The RA, however, offers information about the sensors that may be removed from the plant without reducing the degree of plant knowledge that one has achieved.

In general, the techniques proposed to give aid in the OA try to determine an instrumentation design that maximizes the amount of calculable variables of a plant. This can be achieved with a procedure that partitions the mathematical model of the process in a sequence of subsystems of equations. These subsystems allow individualizing those variables that can be calculated with the data obtained from a group of sensors. A feature that distinguishes chemical processes is their strong nonlinear character. In the subsystems obtained by OA, the degree of nonlinearity has a strong impact on the acting of the techniques employed to carry out the RA. These techniques significantly reduce their times and computational efforts when the amount of linear subsystems that should be processed with respect to the number of nonlinear systems becomes greater.

[*] To whom correspondence should be addressed. E-mail: p.hoch@plapiqui.edu.ar.

[†] CONICET.

[‡] Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur.

[§] Departamento de Matemática, Universidad Nacional del Sur.

[||] CSIC.

10.1021/ie800449t CCC: $40.75
Published on Web 03/06/2009

Another characteristic is that the incidence matrices associated with the equation systems conforming the models of the industrial processes are generally rank-deficient. This is due to the presence of redundant equations, whose purpose is to allow the data reconciliation to detect measurement errors in the sensors located in the plant.[12]

As a result, there is more than one way to decompose the equation subsystems. Therefore, various partitions can lead to subsystems with different nonlinearity degrees, which are based on the equations and variables that conform each particular subsystem. In this context, it would turn out desirable to find the best decomposition, that is, the one that tends to reduce the nonlinearity of its subsystems, moving the nonlinear relationships to the redundant or unassigned equation subsets following the equations categories defined in Ponzoni et al.;[13] see section 2.1. Nowadays, none of the search algorithms used in OA keeps this factor in mind.

The purpose of this work was to carry out a feasibility analysis of the extension of some of the existing techniques to incorporate this aspect to the decomposition process. In particular, we present a modified version of the OA algorithm known as Direct Method (DM).[13] This new version, which we called the Extended Direct Method (EDM), favors the reduction of the nonlinearity degree in the resulting decomposition, without affecting the degree of knowledge provided by the DM.

This Article is organized as follows: in section 2, a detailed explanation of how the instrumentation is designed with the help of our DSS, as well as a description of the methodology adopted by the DM for the OA; in section 3, the EDM is proposed, together with the way nonlinearity degrees are determined and incorporated as a search heuristic; in section 4, results of a comparative performance analysis between DM and EDM for three real study cases are discussed; and finally, in section 5, conclusions and future work are analyzed.

2. Decision Support System

The Decision Support System proposed by Vazquez et al.[10] is a friendly software package that helps the process engineer in the instrumentation design of chemical plants. This DSS allows the rigorous and precise modeling of real problems no matter how big the models under analysis are. Thus, it becomes an efficient and robust tool for complex problems treatment. Another characteristic, which is similarly important, is the flexibility the package has for the addition of new algorithms, allowing that, as it will be seen in this Article, any improvement or new algorithm to be implemented can be incorporated to the software in a simple way.

Several modules with specific tasks constitute the DSS. These blocks are integrated through a graphical interface so as to allow the engineer to make the complete instrumentation design. The modules that constitute the DSS are: a data entry module (DEM), a model generator module (MGM),[14] a robust module for OA (OAM), and an RA module (RAM).[11] In Figure 1, a complete scheme together with the DSS interaction is shown.

In the first place, the engineer should define the plant topology, entering the items of process equipment and the streams (pipes) that interconnect them. Once the whole flowsheet is informed, an initial sensor configuration should be proposed. This activity can be carried out manually by the user or through a genetic algorithm specially implemented for this task.[15-18] The following step is the generation of the complete mathematical model, making use of the MGM. The generated model should be entered in the OAM, which determines the identity of the variables that are observable by means of a classification routine. The user, who decides if it is satisfactory, should analyze this first result. If any of the important process variables is left unobservable, it is necessary to revert to DEM. The DEM is processed again after having added sensors for those variables. The OAM is then run once more to obtain a more suitable classification.

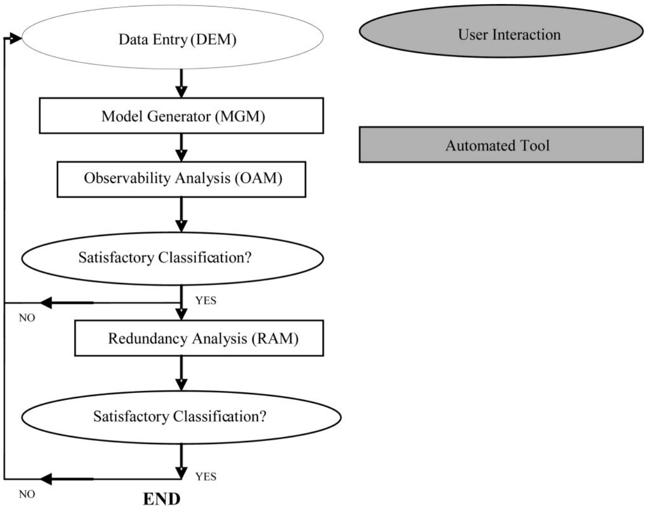

Data Entry (DEM)
User Interaction
Model Generator (MGM)
Automated Tool
Observability Analysis (OAM)
Satisfactory Classification?
YES
NO
Redundancy Analysis (RAM)
Satisfactory Classification?
YES
NO
END

Figure 1. Interaction scheme between the user and the DSS showing how the chemical plant instrumentation design is carried out. Rectangles represent tasks automatically made by the DSS, and ellipses represent tasks the user takes part in.

If the user approves the result yielded by the OAM, the following step is to enter this classification in the RAM. As it was previously explained, this module indicates which variables associated with the sensors are also calculable with the mathematical model. In fact, for the sake of data reconciliation or security issues, it is sometimes desirable to have some of the observable variables measured. If this is not the case, the engineer will have to revert to the DEM, remove some of the redundant sensors, and run the OAM and RAM all over again. Once this iterative process is finished, a satisfactory instrumentation design for the process plant is established.

In particular, the OAM implementation uses the Direct Method (DM).[13] The DM constitutes a very robust and efficient matrix-partitioning approach. The technique is robust because it can be applied to any type of matrices, whatever its structural pattern, and efficient because it allows one to incorporate constraints in the block conformation to guide the ordering according to previously specified guidelines. The method is extremely flexible because it is possible to generate different reorderings for the same problem until finding the most satisfactory one. Because of the previously mentioned reasons, it is possible to state that this technique constitutes a solid base for the development of a methodology for the resolution of sparse nonlinear equations systems. This reordering simplifies the resolution of the original system. Moreover, it is even applicable to numerically singular systems.

2.1. Observability Analysis Using the Direct Method. These algorithms yield four kinds of variables according to feasibility of calculation: (1) Redundant variables are measured variables that can also be calculated from the model equations and the rest of the measured variables. (2) Non-redundant variables are measured variables that cannot be computed from the equations and the rest of the measured variables. (3) Observable variables are unmeasured variables that can be evaluated from the available measurements using the equations.

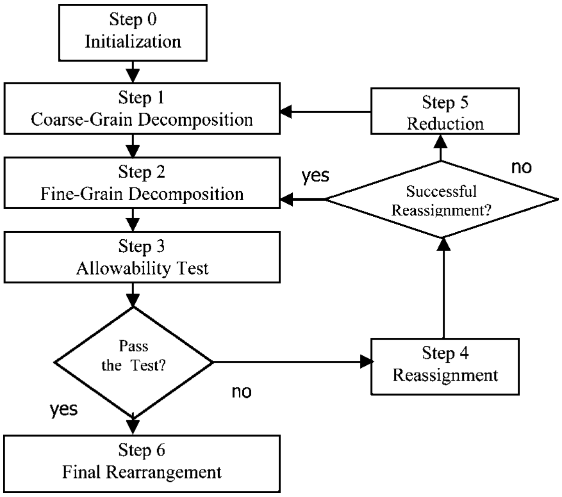


Figure 2. Direct Method architecture.

$$\mathbf{N} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 3. Mathematical model's occurrence matrix.

(4) Unobservable variables are unmeasured variables that cannot be evaluated from the available measurements using the equations.

In turn, the model equations are classified into three categories: (1) Assigned equations are those that will be employed to find the value of the observable variables. (2) Redundant equations are those that have not been assigned and whose variables are either measured or observable. (3) Unassigned equations are those that have not been assigned and contain at least one unobservable variable.

The OA objective is to determine which of the nonmeasured variables can be calculated from either the measured variables or the mathematical model equations. More specifically, its task is to detect the nonmeasured variables included in the group of observable variables and to individualize those in the other group. In particular, the DM observability algorithm consists of seven stages that were outlined in Figure 2. In Ponzoni et al.,[13] there is a detailed explanation of each of the steps through the various algorithms that are applied in each stage. In this Article, we present a complete application example of the DM, to show clearly the task each stage involves.

In the first place, an initialization phase is carried out (step 0), where the Bigraph G(N) associated with the occurrence matrix N (Figure 3), which corresponds to the mathematical model of the plant, is built (Figure 4). This bigraph contains two sets of nodes, called R and C, which represent N rows and columns, respectively.

In the following step (step 1), a G(N) maximum matching, P, is obtained (Figure 4). A matching is constituted by a disjoint-edge set (without common end points) that connects R nodes with C nodes.[19] The number of edges in P is called the cardinality of the matching P. In particular, a matching is called maximum if there is no other matching with higher cardinality. On this matching base, a R and C node classification is employed to carry out a reordering of the matrix N, to the shape shown in Figure 5. This node classification partitions C into SC1, SC2, and HC, and R into SR1, SR2, HR, and VR. The SC1 and SC2 nodes are associated with the observable variables, while the HC nodes correspond to the unobservable variables. In turn, the nodes in SR1 and SR2 represent the assigned equations, the ones in VR indicate the redundant equations, and those in HR correspond to equations that contain unobservable variables.

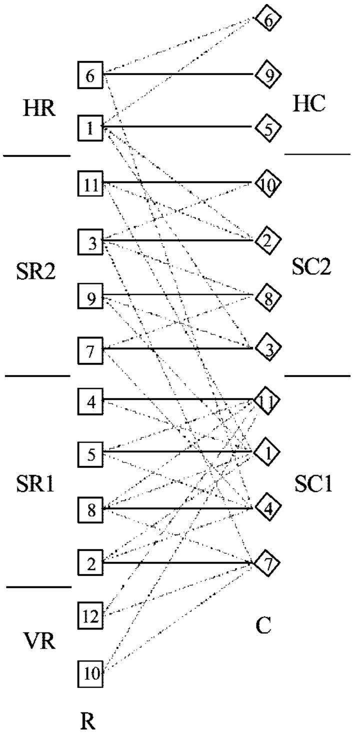
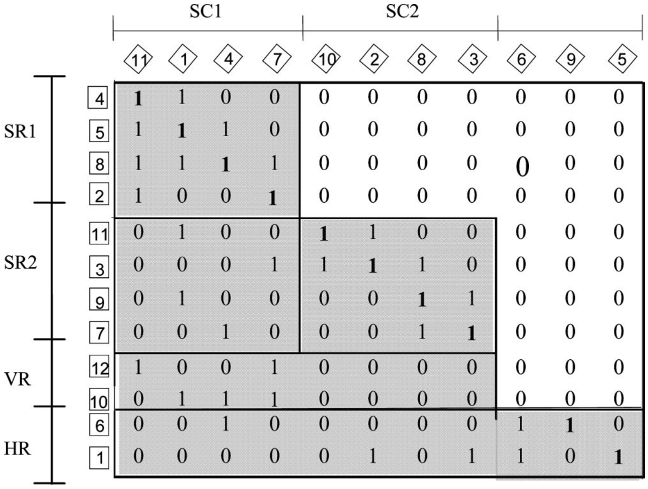


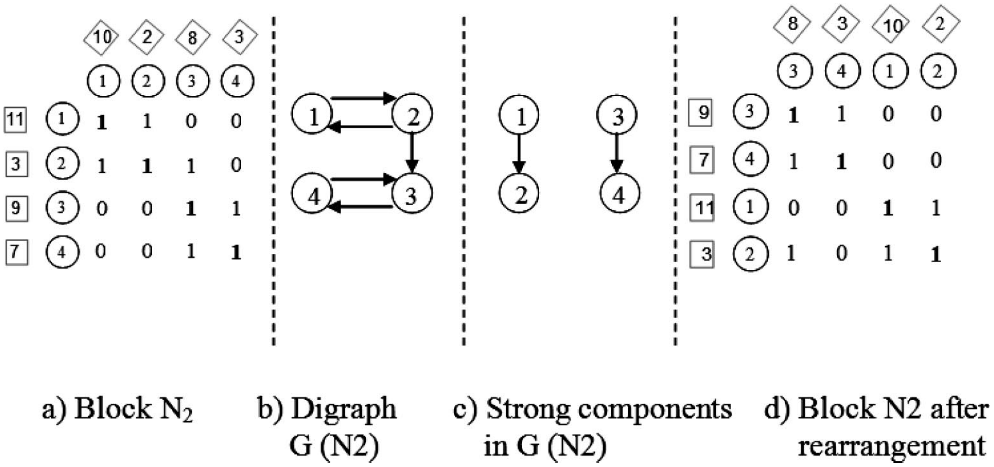
Figure 4. Maximum matching for G(N) bigraph.



		SC1				SC2						
		11	1	4	7	10	2	8	3	6	9	5
SR1	4	**1**	1	0	0	0	0	0	0	0	0	0
	5	1	**1**	1	0	0	0	0	0	0	0	0
	8	1	1	**1**	1	0	0	0	0	0	0	0
	2	1	0	0	**1**	0	0	0	0	0	0	0
SR2	11	0	1	0	0	**1**	1	0	0	0	0	0
	3	0	0	0	1	1	**1**	1	0	0	0	0
	9	0	1	0	0	0	0	**1**	1	0	0	0
	7	0	0	1	0	0	0	1	**1**	0	0	0
VR	12	1	0	0	1	0	0	0	0	0	0	0
	10	0	1	1	1	0	0	0	0	0	0	0
HR	6	0	0	1	0	0	0	0	0	1	**1**	0
	1	0	0	0	0	0	1	0	1	1	0	**1**

Figure 5. Rearranged matrix after the coarse-grain decomposition.

The next step (step 2) is the fine-grain decomposition. The coarse-grain decomposition yields two structurally nonsingular square blocks determined by the sets (SR1,SC1) and (SR2,SC2). The fine-grain decomposition stage partitions these blocks into irreducible square subsystems by means of the classical Tarjan

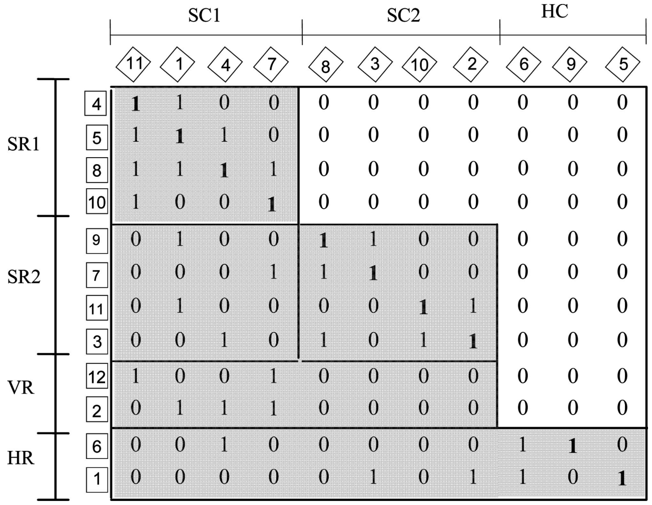


		10	2	8	3
		1	2	3	4
11	1	**1**	1	0	0
3	2	1	**1**	1	0
9	3	0	0	**1**	1
7	4	0	0	1	**1**

		8	3	10	2
		3	4	1	2
9	3	**1**	1	0	0
7	4	1	**1**	0	0
11	1	0	0	**1**	1
3	2	1	0	1	**1**

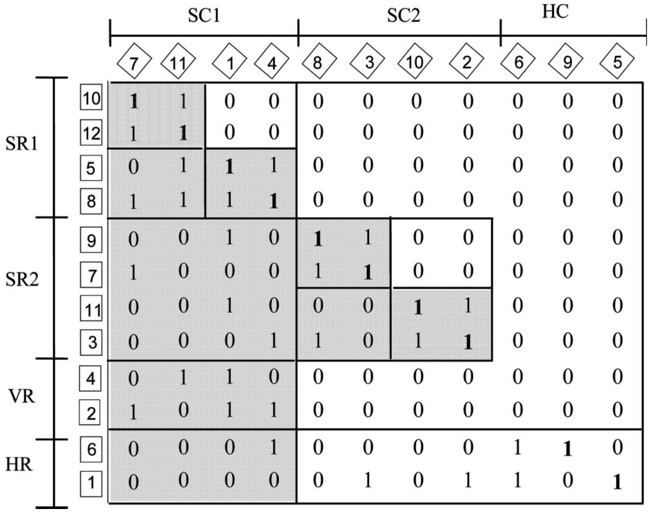
a) Block N_2 b) Digraph G (N2) c) Strong components in G (N2) d) Block N2 after rearrangement

Figure 6. Fine-grain decomposition of N_2 = (SC2, SR2).



		SC1				SC2				HC		
		11	1	4	7	8	3	10	2	6	9	5
SR1	4	**1**	1	0	0	0	0	0	0	0	0	0
	5	1	**1**	1	0	0	0	0	0	0	0	0
	8	1	1	**1**	1	0	0	0	0	0	0	0
	10	1	0	0	**1**	0	0	0	0	0	0	0
SR2	9	0	1	0	0	**1**	1	0	0	0	0	0
	7	0	0	0	1	1	**1**	0	0	0	0	0
	11	0	1	0	0	0	0	**1**	1	0	0	0
	3	0	0	1	0	1	0	1	**1**	0	0	0
VR	12	1	0	0	1	0	0	0	0	0	0	0
	2	0	1	1	1	0	0	0	0	0	0	0
HR	6	0	0	1	0	0	0	0	0	1	**1**	0
	1	0	0	0	0	0	1	0	1	1	0	**1**

Figure 7. Matrix resulting from the first reassignment.



		SC1				SC2				HC		
		7	11	1	4	8	3	10	2	6	9	5
SR1	10	**1**	1	0	0	0	0	0	0	0	0	0
	12	1	**1**	0	0	0	0	0	0	0	0	0
	5	0	1	**1**	1	0	0	0	0	0	0	0
	8	1	1	1	**1**	0	0	0	0	0	0	0
SR2	9	0	0	1	0	**1**	1	0	0	0	0	0
	7	1	0	0	0	1	**1**	0	0	0	0	0
	11	0	0	1	0	0	0	**1**	1	0	0	0
	3	0	0	0	1	1	0	1	**1**	0	0	0
VR	4	0	1	1	0	0	0	0	0	0	0	0
	2	1	0	1	1	0	0	0	0	0	0	0
HR	6	0	0	0	1	0	0	0	0	1	**1**	0
	1	0	0	0	0	0	1	0	1	1	0	**1**

Figure 8. Matrix resulting from the second reassignment.

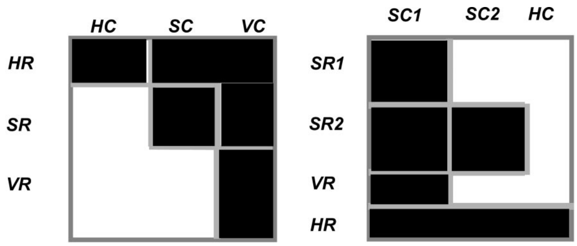
algorithm[20] for the detection of the strong components of a digraph. First, the digraph G(N_1) is decomposed, where N_1 = (SR1, SC1), and, second, the Tarjan algorithm is applied to G(N_2), where N_2 = (SR2, SC2). Figure 6 shows the fine-grain decomposition of block N_2 for the example given in Figure 5. In the first place, the digraph G(N_2) is built (Figure 6a,b). Each node in G(N_2) is associated with two nodes from the bigraph, which come from SR2 and SC2, respectively. Next, the digraph is partitioned into its strong components (Figure 6c). Finally, N_2 is rearranged according to the strong components found through the Tarjan algorithm (Figure 6d).

It might happen that some of the blocks previously found in the fine-grain decomposition do correspond to a numerically singular equations subsystem. In this case, a restriction that prohibits this subsystem generation is imposed (step 3). Let us assume that there is a constraint T_1 composed of the row nodes {4, 5, 8, 2} and the column nodes {11, 1, 4, 7} because the associated equations and variables correspond to a singular subsystem from the mathematical model of the plant. The allowability test then rejects the block, and the algorithm proceeds with the reassignment stage. At this point, we shall assume that the row nodes 2 (from SR1) and 10 (from VR) are chosen and exchanged. Afterward, block N_1 is composed of the row nodes {4, 5, 8, 10}, as shown in Figure 7, while node 2 corresponds to a redundant equation.

If the fine-grain decomposition is executed again, a single strong component that comprises the complete block N_1 is found again. Next, we will assume that the new block, composed of row nodes {4, 5, 8, 10} and column nodes {11, 1, 4, 7}, is checked and rejected as a result of a constraint made up of row nodes {4, 5, 10} and column nodes {11, 1, 4}. This triggers a new reassignment, where, for example, row nodes 4 and 12 are exchanged, yielding a block N_1 with row nodes {5, 8, 10, 12} and column nodes {7, 11, 1, 4}. Next, the fine-grain decomposition is repeated, resulting in the following two strong components: N_{11} = {{10, 12}, {7, 11}} and N_{12} = {{5, 8}, {1, 4}}. Both of them pass the allowability test, which ends in this way for N_1. The resulting matrix is shown in Figure 8.

There are different techniques to carry out matrix permutations to block triangular forms. Some of them were not applicable to rectangular, structurally singular matrices, while others led to useless block distributions. Pothen and Fan[21] introduced a partitioning technique for general matrices. This technique consists of two different stages. In the first place, a coarse-grain decomposition is yielded by applying the Dulmage−Mendelsohn decomposition.[22] The Dulmage−Mendelsohn decomposition is one of the most widespread noncombinatorial techniques to permute a general matrix to block triangular forms (BTF). Once this technique is applied, Pothen's method is used to perform the fine-grain decomposition.

The main difference between the Dulmage−Mendelsohn decomposition and the Direct Method lies in the fact that DM's technique places in the same set all of the nodes in N1 that



a) Dulmage-Mendelsohn classification b) Direct Method classification

Figure 9. Classification of nodes and coarse-grain decomposition for both techniques.

Table 1. Term Pondering by Means of Weights, Which Are in Accordance with the Term's Degree of Nonlinearity

			nonlinear				
term	linear	bilinear	1 variable	2 variable	3 variable	4 variable	5 or more variables
punctuation w	0	1	2.2	2.4	2.6	2.8	3.0

belong either to VR or to SR1, while the classification for the instrumentation design, that is, the Direct Method, requires a distinction to be made between them. Figure 9 shows the contrasts between both classifications. Figure 9a shows the DM classification, whereas in Figure 9b it can be noticed how the nodes should be rearranged to get the desired BTF. Apart from being square, an important characteristic of blocks (SC1, SR1) and (SC2, SR2) is that they have a full transversal because the nodes in SC1, SR1, SC2, and SR2 belong to the maximum matching.

Another important difference between both proposals is the fact that Pothen and Fan's decomposition was presented for solving linear systems only, while the Direct Method is designed for linear and nonlinear systems. It is well-known that most of the mathematical models resulting from a chemical plant are strongly nonlinear equation systems. This constitutes an essential reason why it became necessary to propose a new decomposition technique to be able to carry out a computer-aided instrumentation design. A more detailed comparison between both decompositions is in Ponzoni et al.[13]

3. Extended Direct Method

The extension of the Direct Method, which is proposed in this Article, arises from the incorporation of the nonlinearity degree (NLD) analysis to the original DM, allowing in this way the improvement on its performance and applicability range. The main objective why the NLD analysis has been introduced is to reduce the amount of nonlinear subsystems obtained by the DM decomposition. In this way, the first step to reach this goal consists of defining how to quantify the NLD considering that each equation and variable of the system contributes to the mathematical model. This information is basic to guide the model decomposition process, to favor the conformation of the linear subsystems in contrast to the nonlinear ones.

3.1. NLD Definition and Quantification. a. Nonlinearity Degree − Equation Analysis. The NLD of a given equation is computed by considering the linearity type of each term in the equation. A term can be linear, bilinear, or nonlinear, depending on the algebraic operations that it involves. For example, in the equation $a + b^3 + c \cdot d = 0$, the first term is linear, the second one is nonlinear, and the last one is bilinear.

b. Nonlinearity Degree − Analysis of Variables. The variable NLD is constituted by the contribution of all of the bilinear and nonlinear terms where that variable takes part throughout all of the equations of the system.

To calculate the NLD, we assign a specific weight to each term according to its linearity type. The criterion presented here, which was empirically defined, tends to favor the linear terms in contrast to the bilinear and nonlinear ones, because it assigns a smaller weight to the terms with a lower degree of nonlinearity. Table 1 is completely general, and it specifies the weights to be employed for the determination of the nonlinearity degree for each equation and variable in the system. The most important feature is to choose weights (any number) that increase in their value as the nonlinearity level becomes larger.

As we will show in section 4.1.2, it is essential to respect the weighing criterion to achieve the minimum nonlinear degree in the resulting decomposition. In turn, it can be noticed that, for the same weighing criterion, the scalar used as weight has no influence on the yielded number of nonlinear blocks, but does influence the amount of minimum-size blocks resulting in the partition. It is important to notice that it is unfeasible to define a universal set of weights for any kind of process plants because the structure of the algebraic equations involved in different types of processes can be significantly different. It is advisable that, for a given specific process, the user performs a preliminary experimentation to adjust these weights empirically. An example of this kind of experimental analysis is reported in section 4.1.2 for the study of a distillation column.

Next, we have explained and illustrated how the NLD computation is carried out for the equations and variables of the mathematic model. First, we explain (a) quantification with equations: for each equation, the weights corresponding to its terms are added. To normalize the value, the total weight for a given equation is divided by the number of terms. Second, we have (b) quantification with variables: for each variable, the weights corresponding to each term where that variable takes part are added. The terms are evaluated for all of the system equations. To normalize the value, the total amount is divided by the total number of terms.

Example 1. Given the equation E_1, $a + b^3 + c \cdot d = 0$, ponder the terms.

The calculation should be made in two stages.

(a) NLD value of the equation:

The equation has $t = 3$ terms. The weight w_i given in Table 1 is applied using the following formula:

$$\mathrm{NLD}(E) = \frac{\sum_{i=1}^{t} w_i}{t} \tag{1}$$

Applying the formula to the equation, we obtain:

$$\mathrm{NLD}(E_1) = \frac{0 + 2.2 + 1}{3} = 1.07 \qquad (2)$$

(b) NLD value of all of the variables involved:

Let us consider t_v as the number of terms where the variable v takes part across the whole system. For each variable, the following formula then should be applied:

$$\mathrm{NLD}(v) = \frac{\sum_{i=1}^{t_v} w_i}{t_v} \qquad (3)$$

In this case, we find the following.

Variable a:

$$\mathrm{NLD}(a) = \frac{0}{1} = 0 \qquad (4)$$

Variable b:

$$\mathrm{NLD}(b) = \frac{2.2}{1} = 2.2 \qquad (5)$$

Variable c:

$$\mathrm{NLD}(c) = \frac{1.1}{1} = 1.1 \qquad (6)$$

Variable d:

$$\mathrm{NLD}(d) = \frac{1}{1} = 1 \qquad (7)$$

Example 2. Given the following 3 × 3 system (Figure 10) with a, b, and c as variables and according to the weighting criterion defined above, calculate the NLD value of the equations and variables.

(a) NLD value of the equations:

Equation 1:

$$\mathrm{NLD}(E_1) = \frac{2.2 + 0 + 1}{3} = 1.07 \qquad (8)$$

Equation 2:

$$\mathrm{NLD}(E_2) = \frac{2.4 + 1}{2} = 1.7 \qquad (9)$$

Equation 3:

$$\mathrm{NLD}(E_3) = \frac{0 + 0 + 0}{3} = 0 \qquad (10)$$

Therefore, eq 3 is the “less nonlinear” equation because it presents the smaller NLD, and eq 2 is the “most nonlinear” because its NLD is the greater one.

(b) NLD value of the variables:

Variable a:

$$\mathrm{NLD}(a) = \frac{2.2 + 2.4 + 0}{3} = 1.53 \qquad (11)$$

Variable b:

$$\mathrm{NLD}(b) = \frac{0 + 1 + 1 + 0}{4} = 0.5 \qquad (12)$$

Variable c:

$$\mathrm{NLD}(c) = \frac{1 + 2.4 + 1 + 0}{4} = 1.1 \qquad (13)$$

In a similar way, the variable b is the “least nonlinear” (NLD = 0.5), and the variable a is the “most nonlinear” (NLD = 1.53).

$$\begin{cases} a^2 + b + b.c = 0 \\ \dfrac{3c^2}{a} + b.c = 0 \\ a + 3b + 5c = 0 \end{cases}$$

Figure 10. System of equations that represent an industrial process.

3.2. Implementation of the NLD Computation Using a Syntax Analyzer. Once the rules to compute nonlinearity are fixed, it is mandatory to specify a calculation algorithm. A syntax analyzer (parser) was implemented to compute the NLD. A parser is an algorithm that looks for and recognizes text strings with a certain pattern within a text file.

The system of equations that represents an industrial process is placed in a text file with RTF (Rich Text Format) format, which also contains a series of control headings and labels (tags). To carry out the NLD computation, the parser operates in the following stages:

In the first stage, it removes control tags and it applies a new format to the equations terms. In this way, a new text file is generated.

In the second stage, it scans each equation of the new file and calculates the NLD.

Programming a parser is a complex but systematic task. For this reason, a program called Flex 2.5[23] was used for this purpose. Flex is a fast parser generator, an automatic tool for parser codification. Its behavior is governed by an input similar to a program written in some compilable language. Some patterns or regular expressions that must be recognized, and their associated actions codified in C language, are written in a source file. When processing the source, Flex generates the C code associated with the parser that identifies these expressions.

3.3. Integration of the NLD to the DM. Once the NLD weighing coefficients are all calculated, these values are incorporated to the coarse-grain decomposition of the DM, so that the nodes selected for the bigraph maximum turn out to be those with the smallest NLD. In this way, an EDM that favors the conformation of decomposition blocks with a low degree of nonlinearity is obtained.

In Figure 11, the pseudocode for the modified algorithm (MMB), which is employed to compute the maximum matching considering the NLD, is reported. Additional structures were incorporated to represent the vectors that keep NLD coefficients for the equations (NDE) and variables (NDV). The heuristic we introduced in the MDM search process basically consists of giving priority to the exploration of the adjacent nodes with low NLD over those with greater NLD.

In short, the basic sequence for the calculation of the EDM coarse-grain decomposition is given by the execution of the following procedures.

(1) The first is NLD_BuildFiles: it carries out the parsing of the file that contains the system of equations. It then scans the whole system computing the NLD coefficients that correspond to each equation and variable, saving them in files.

(2) The second is LoadBigraph: it uses the bigraph file and the files created in the previous step and adds the nodes to the bigraph structure. At the same time, it orders the adjacent nodes according to their associated NLD. For this task, it uses the

```
Input: R, C, N, NDE, NDV
Output: $P_M$
$P_M = \varnothing$.   $C_U = \varnothing$.

// Reordering of the Bigraph nodes according to its NLD coefficients
Reorder the row and column nodes, from the lower element to the greater
one according to its NLD coefficients (NDE and NDV).

 // Building an initial matching
For each node $c \in C$, following the numbering established in the previous step, do:
      Match c with an unmatched node r, so that $r \in R$. If there are different possibilities, take the one with the lowest number.

      If node r does not exist, then $C = C \cup \{c\}$
 end-do

 // Looking for augmented paths
$C_{UN} = \varnothing$
Repeat
  Search for augmented path Au from c, only visiting those nodes in R that have not been visited before during that step.
      Label all nodes that are reached as "visited".
      If an augmented path A has been found
         then  Augment $P_M$ with Au.
         else  Include c in $C_{UN}$
      End if
      $C = C_{UN}$.   $C_{UN} = \varnothing$.
Until no augmented paths are found in the loop.
End of Algorithm
```

Figure 11. Modified version of the maximum matching in a bigraph algorithm (MMB).

qsort precompiled routine, provided by the C language. This function makes a quick-sort ordering on an element array, on the basis of a key field.

(3) The third is Maximum Matching: it computes a Maximum Matching of the bigraph loaded in the second step. In comparison with the original version, this procedure code did not suffer any change. From this step on, the method has the same behavior as the original one.

4. Extended Direct Method Performance Analysis

Because the Extended Direct Method represents an evolution of the Direct Method, we compared the performance of both methods for three study cases, which represent industrial processes with different levels of complexity. The study cases represent realistic and intricate models of different dimensions. In this way, we aim at testing the Extended Method so as to assess the magnitude of the improvements with respect to the original Direct Method for any scale and composition systems. The comparison is reported by means of the following formula:

$$\Delta = \frac{\mathrm{EDM}_0 - \mathrm{DM}_0}{\mathrm{DM}_0}100 \qquad (14)$$

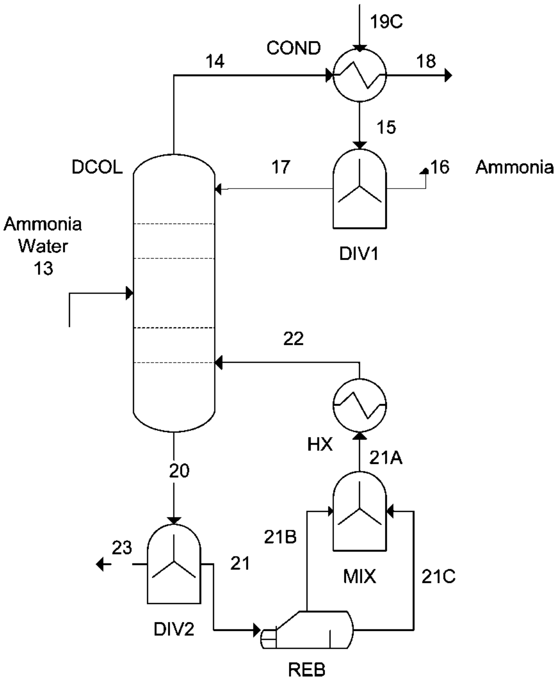


Figure 12. Distillation column scheme.

where Δ is a percentage that quantifies the difference between both methods. DM_0 and EDM_0 represent output variables of interest, to be made precise below, obtained with the original and the Extended Direct Method, respectively.

In particular, for the first study case, we presented a comparative analysis of the result of employing different sets of weights to compute the NLD of the mathematical terms, to substantiate the choice of the values we chose in this Article (Table 1). In turn, in this case we analyzed the impact that the NLD reduction has on the redundancy analysis process. In this way, we were able to quantify the improvements in the DSS global performance due to the incorporation of the Extended Direct Method.

4.1. Case I: Distillation Column. 4.1.1. Description of the Industrial Process. The first example corresponds to the section of an ammonia synthesis plant where ammonia is purified by distillation. A distillation column with two trays is used to separate a stream composed by ammonia and water, which enters at the first tray. A total condenser is used to recover the ammonia product from the top of the column (99.5% of purity), and the bottom product is over 99% water. A column scheme is shown in Figure 12.

The steady-state mathematical model of this section is made up of 104 nonlinear algebraic equations with an initial sensor configuration set with 25 measurements and 85 unmeasured variables.

4.1.2. Experimental Assessment of the Nonlinearity Weights. Various tests were done to define which heuristics should be employed with the Extended Direct Method. These tests can be grouped under three different cases. Because of the complexity and long time required to run and analyze each of the industrial processes, the complete study was carried out just for this column distillation process. Each of the cases presented in Table 2 follows a different general weighting criterion, to be able to analyze the behavior of the decomposition algorithm for diverse scenarios. In turn, for each of the three cases, we changed the scalars used to carry out the computation of the nonlinearity degree, always maintaining the global criterion individually corresponding to the case. In this way,

Table 2. Results Obtained for the Distillation Column Model with EDM for Different Sets of Weights

	parameters								
			nonlinear						
	linear	bilinear	1 V	2 V	3 V	4 V	5+ V	nonlinear blocks	1 × 1 blocks
DM	0	0	0	0	0	0	0	4	48
case A	4	0	2	2.2	2.4	2.6	2.8	4	48
	4	3	2	2.2	2.4	2.6	2.8	4	48
	6	3	1.8	1.6	1.4	1.2	0	4	50
case B	2	3	1.8	1.6	1.4	1.2	0	3	55
	3	5	1.8	1.6	1.4	1.2	0	3	55
	6	7	1.8	1.6	1.4	1.2	0	3	56
case C	0	1	7	7.2	7.4	7.6	7.8	2	56
	0	1	2	3	3.2	3.4	3.6	2	58
	0	1	2.2	2.4	2.6	2.8	3	2	58

Table 3. DM and EDM Observability Results Obtained for the Distillation Column Model

amount of	DM	EDM	Δ
observable variables	63	63	0
unobservable variables	22	22	0
1 × 1 subsystems	48	58	+21%
linear subsystems	47	57	+21%
nonlinear subsystems	4	2	−50%

we were also able to analyze the influence of adopting different scalar values, but following the same general weighting criterion.

In the first case (case A), we completely inverted the originally planned logic, then giving the biggest penalization (weight) to the linear terms, followed by the bilinear terms, and, finally, assigning the lowest weight to nonlinear terms. This penalization assignment, as we expected, led us to the decomposition with the biggest number of nonlinear blocks and, in turn, the lowest number of 1 × 1 blocks. As expected, this outcome reasonably constitutes the worst result that we have obtained. This is a logical result because deep inside the algorithm (DM) we are minimizing a function composed by the various weights, with the purpose of representing the degree of nonlinearity. If we assign the highest value to the linear terms, it is obvious that we would not get the expected result. If in turn, we were maximizing the function, then assigning the highest value to linear blocks should be the case.

Second, we assigned the highest penalization to the bilinear subsystems over the linear ones (case B). We kept, as in case A, the lowest weights for nonlinear terms, continuing with the preference for these over the rest. The fact of assigning a lower penalization to the linear terms with respect to the bilinear ones led to an improvement in the quantity of nonlinear blocks we obtained, as well as in the minimum-size subsystems. It is clear that by a continual favoring of the nonlinear terms above the rest, we are unable to obtain the best possible partition.

Finally, we made the tests by assigning the weights with the criterion we proposed in this Article (case C), that is, giving the highest weight to the nonlinear terms, followed by bilinear ones, and finally assigning the lowest penalization to the linear ones. In this case, taking into account the number of nonlinear blocks we obtained, the resulting partition was the best obtained. In turn, a variation in the scalars yielded different decompositions, which only differed in the number of 1 × 1 subsystems, maintaining the same amount of nonlinear blocks. From this analysis, it emerges as a conclusion that it is imperative to keep the precedence order among the different kinds of terms (linear, bilinear, and nonlinear) established in case C, no matter which specific values have been chosen for the weight definition.

Next, we have incorporated a table that summarizes the results we obtained for each of the cases. In the first row of the table, the original Direct Method decomposition is included. It then can be noticed how, as we descend to lower rows in the table (i.e., analyzing from case A to C), the number of nonlinear blocks decreases, whereas the number of 1 × 1 blocks increases.

It is essential to stress the influence that the selected scalars have on the obtained decomposition, for the same weighting criterion. By analyzing Table 2, we can see that, for the same weighing criterion, the scalar used as weight has no influence on the yielded number of nonlinear blocks, but does influence the amount of minimum-size blocks resulting in the partition. If we analyze each case of the table individually, it can be noticed how the number of nonlinear blocks is kept constant, whereas the quantity of minimum-size subsystems changes, depending on the set of values chosen to carry out the computation of the nonlinearity degree.

It is important to notice that in all cases, and for any set of values, we obtained the highest possible observability degree, the set of observable variables being the same as the one we had found by applying the original Direct Method. Besides, the running times required by the EDM and DM are very similar.

4.1.3. Observability Analysis Using DM and EDM. Next, we included a table with the observability results (Table 3) obtained by the application of both the DM and the EDM, for the mathematical model of the distillation column.

Both methods are able to determine the same number of observable variables, but a remarkable improvement can be appreciated in the EDM application, because an increase of 21% both in the production of low dimension subsystems (1 × 1) and in the generation of linear subsystems is achieved. The most relevant result is the 50% reduction in the quantity of nonlinear subsystems.

4.1.4. Redundancy Analysis Using DM and EDM. The EDM generates a rearrangement of the system that simplifies the task of the algorithm for redundancy analysis (RA) and improves the quality of its output. To show this, we need to explain briefly the main ideas behind the RA algorithm. To analyze if a certain measured variable *m* is redundant, the RA algorithm takes each redundant equation and computes symbolically the derivative of its left-hand side with respect to *m*. This derivative must also take into account the indirect influence of *m* in the expression through observable variables, whose values might depend on that measurement. The algorithm uses the chain rule to compute this indirect influence, and this involves solving a sequence of linear systems whose coefficient matrices are the matrices of symbolic derivatives or Jacobian of the subsystems produced by the observability algorithm. Also, the subsequent steps in the RA algorithm require computing additional symbolic derivatives using the chain rule.

The crucial fact is that nonlinear subsystems generally have a Jacobian whose inverse has a complicated expression. Having such a subsystem at the top of the decomposition has an adverse effect on the efficiency of the redundancy analysis from that point onward. The EDM favors linear blocks, so the nonlinear subsystems tend to appear at the bottom and their impact is minimized. This can greatly improve the performance of the redundancy analysis. Also, if the symbolic expressions are so complex that the RA cannot decide if the measurement *m* is redundant (Red) or nonredundant (N-Red) after a certain time of analysis or number of trials, then *m* is labeled uncategorized (Uncat). So, obtaining simpler expressions generally reduces the number of uncategorized measurements, increasing the level of knowledge of the plant provided by the RA.

In addition, preferring equations with a low nonlinearity degree for building subsystems, as in the EDM, tends to produce

Table 4. For the Decompositions Obtained by the DM and the EDM, Results of the Redundancy Analysis for the Measured Variables[a]

	categorization			
variable	DM		EDM	
F(20)	Uncat		Red	
X(22,20)	Uncat		Red	
P(21)	Red		Red	
T(20)	Uncat		Uncat	
P(23)	Red		Red	
F(22)	Uncat		Red	
T(22)	N-Red		N-Red	
P(HX)	N-Red		N-Red	
P(21A)	N-Red		N-Red	
T(21A)	N-Red		N-Red	
T(21C)	Uncat		Red	
F(14)	Uncat		Uncat	
P(13)	N-Red		N-Red	
T(13)	N-Red		N-Red	
X(22,13)	N-Red		N-Red	
P(14)	N-Red		N-Red	
T(14)	N-Red		N-Red	
T(15)	Uncat		Uncat	
P(15)	N-Red		N-Red	
T(19C)	N-Red		N-Red	
P(18)	N-Red		N-Red	
F(16)	Uncat		Red	
X(20,16)	Red		Red	
X(22,16)	Red		Red	
P(REB)	Uncat		Red	
total uncat. var.	9	(36%)	3	(12%)
total cat. var.	16	(64%)	22	(88%)
total time	7001	s	135	s

[a] Variables *P*(A), *T*(A), and *F*(A) represent the pressure, temperature, and molar flow of stream A, respectively. The variable *X*(B,A) represents the composition of component B in stream A.

a partition where there are more linear and bilinear assigned equations and more nonlinear redundant equations. This is also beneficial for the RA, because having a nonlinear equation as redundant is less harmful than having it as part of a subsystem.

Table 4 summarizes the results for the redundancy analysis. It was carried out for the block decompositions obtained by the application of both the original Direct Method and the Extended Direct Method.

In the first column of the table, a list of the measured variables for the study case of the distillation column is included. In the second column, the categorization that the redundancy software assigned to each variable is shown. As explained above, this categorization is divided into three kinds of variables: redundant (Red), non-redundant (N-Red), and uncategorized (Uncat). In turn, the second column is also split, where it can be seen how the obtained categorization varies depending on the algorithm chosen for the observability analysis. Finally, in the last three rows of the table, global statistics that indicate the total level of knowledge are shown. Also included is the execution time required for the decompositions obtained by applying each algorithm.

Exhibited results show that the application of the EDM leads to a marked improvement in the reached level of knowledge for the measured variables. This can be noticed in the number of measured variables that were uncategorized by applying both observability algorithms. In that sense, for the DM decomposition only a 64% of knowledge level can be reached, against the 88% obtained by applying the EDM.

Another significant factor that must be considered is the execution time required by the redundancy algorithm to perform the analysis of both decompositions. As shown in the table, the required time to complete the analysis from the DM partition (7001 s) was significantly greater than the amount of time required for the treatment of the decomposition found by the EDM (135 s). This significant difference between both executions is directly related to the resulting number of uncategorized variables for each case. Given a measured variable, if the redundancy algorithm reaches a maximum limit of iterations without achieving a classification, it assigns to that variable the Uncat label. It then continues the analysis with a new variable. Therefore, the time used for the frustrated analysis of the Uncat variable significantly impacts on the total time required for the algorithmic execution. Consequently, by tripling the number of Uncat variables from one case to another, the algorithm consumes much more time.

It is important to stress that the classifications obtained for both cases are consistent. In other words, whenever a variable was categorized in both analyses, the category was always the same (Red or N-Red). This feature reflects the more general fact that the category of a measured variable is independent of the observability method applied previous to the redundancy analysis.[11] The only difference in applying different methods can be noticed in the number of variables that could be categorized.

4.1.5. Summary of the EDM Impact on Instrumentation Global Analysis. By analyzing the results, we can see the importance and direct impact that the changes we implemented to the observability analysis algorithm have on the global DSS performance. The main improvements can be summarized by the following items.

The first is increased knowledge of the process redundancy degree: At the end of the instrumentation design task, the user has the possibility of having a higher knowledge level of the redundancy that a specific set of measurements gives to the process under study. This is essential for making decisions related to the removal or incorporation of new sensors. The sensors removal involves taking away redundant measurements that could be unnecessary, thus reducing the monitoring costs of the plant. However, sensors addition involves the measurement of new variables, to achieve critical variables of the process becoming redundant when they were not before.

Next is the reduction of required time in the global process of instrumentation design: As explained in section 2, the whole process of instrumentation design constitutes an iterative task that is comprised of several steps. This forces the user to execute the different steps many times, requiring much time for execution and analysis to achieve a satisfactory sensor configuration. The redundancy analysis, due to the time and computational effort required to be carried out, constitutes one of the bottlenecks of the instrumentation design. The application of the observability algorithm we proposed in this Article lets the user achieve a significant reduction in the time required for the execution of the redundancy analysis of a process plant.

There is also a reduction in the computational effort required for the data reconciliation process: Because the instrumentation-design task does not involve the mathematical resolution of equation systems, one can refer to it as a purely structural methodology. However, a numerical task that constitutes an essential step for the process monitoring is the data reconciliation. This involves comparing data obtained from field measurements with values that are estimated by the resolution of the model equations, to detect possible failures in the measurement instruments. This task requires the resolution of only the equation subsystems that are obtained by the application of the observability algorithm, without taking into account the equations that were not included in the subsystems (redundant

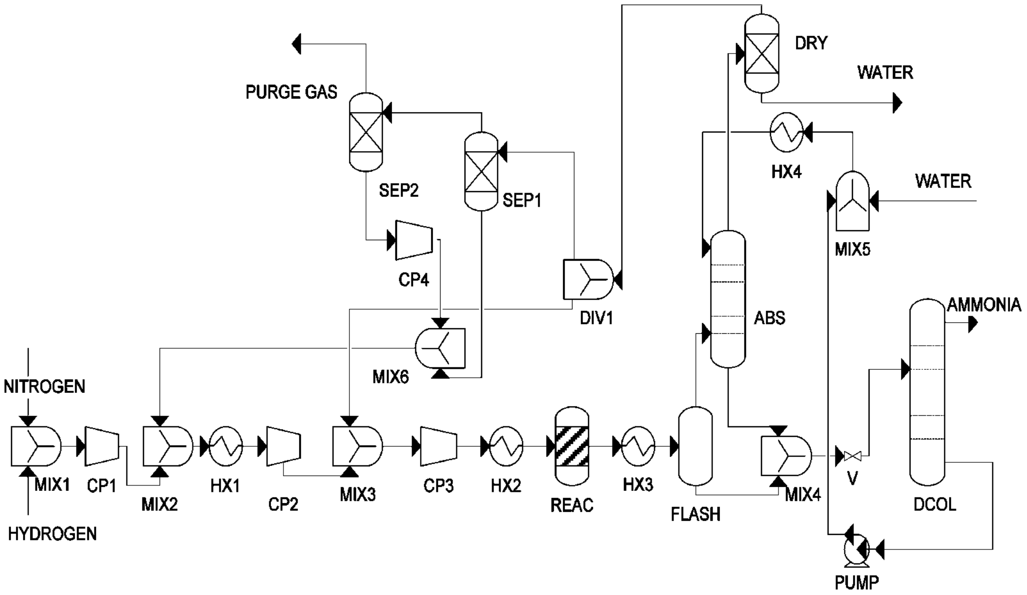


Figure 13. Haber−Bosh ammonia synthesis plant scheme.

or not assigned equations). For this reason, a reduction in the nonlinearity degree in the obtained decomposition involves solving a larger number of linear subsystems. These subsystems can be computed by applying some direct-resolution method, thus decreasing the computational requirements to perform the data-reconciliation task.

Finally, we have simplification of the forbidden-block analysis process: In Figure 2, we included a scheme that describes the DM architecture. In that figure, we can see that the third step of the algorithm consists of an admission test of the blocks that are generated in the decomposition. As presented in Ponzoni et al.,[13] in this step the algorithm carried out an automatic checking of the subsystems to ensure that it does not contain any of the forbidden blocks previously specified by the user. These subsystems are often constituted by linear-dependent equations, for example, a block conformed by component mass balances of a stream and the global mass balance. This kind of analysis is simplified by the application of the EDM, because the decomposition has a larger number of linear and 1 × 1 blocks, which facilitates the detection of singular subsystems.

4.2. Case II: Haber−Bosch Ammonia Synthesis Plant. 4.2.1. Description of the Industrial Process.

The second example corresponds to a complete Haber−Bosch ammonia synthesis plant. This plant was designed by Bike[24] to produce 1500 ton/day of ammonia at 240 and 450 kPa. A minimum purity of 99.5% can be obtained with this process by means of the Haber−Bosch process. The hydrogen and nitrogen feedstocks are provided by a coal gasification facility, and also contain inert gases, like argon and methane.

The kinetics of the ammonia synthesis reaction over a double-promoted iron catalyst can be described on the basis of the following rate equation:

$$N_2 + 3H_2 \Leftrightarrow 2NH_3$$

The reactor product enters an absorption column to remove the ammonia with water. The liquid output the column is fed to a distillation column to obtain pure ammonia as top product. There is also a recovery section, composed of membranes, where the hydrogen is recovered and recycled to the feed. A complete scheme of the plant is shown in Figure 13.

The mathematical model chosen to represent the plant consisted of 557 equations and 587 variables. The analysis was started from a basic configuration with 74 measured variables, with the rest being initially defined as unmeasured.

Table 5. DM and EDM Observability Results Obtained for the Ammonia Synthesis Plant Model

amount of	DM	EDM	Δ
observable variables	216	216	0
unobservable variables	297	297	0
1 × 1 subsystems	144	164	+14%
linear subsystems	143	164	+15%
nonlinear subsystems	8	4	−50%

4.2.2. Observability Analysis with DM and EDM.

Next, we included a table with the observability results (Table 5) obtained by the application of both the DM and the EDM, for the mathematical model of the ammonia synthesis plant.

As in case I, both methods assign the same number of observable variables. The results obtained are similar in quality to the first example. Here, a 14% increment in the quantity of 1 × 1 subsystems and a 15% increment in the number of linear subsystems are achieved. Like in the first example, a reduction of 50% is also obtained in the number of nonlinear subsystems.

4.3. Case III: Ethane Plant. 4.3.1. Description of the Industrial Process.

The last case under study corresponds to an existing ethane plant. The process can be divided into three main sections (Figure 14): gas compression and dehydratation, cryogenic separation, and fractionating. In the first place, the inlet gas is filtered and compressed in three parallel compressors. To avoid the formation of ice and hydrates, the gas is air cooled and dehydrated. Once this process is finished, the feed is divided into two streams, and each of these streams is sent to a cryogenic train (Figure 15), which is basically composed of an air cooler, heat exchangers, Joule−Thompson valves, and a demethanizer column. The demethanizer is a low-temperature distillation column where the separation between methane and ethane is made. The bottom product of the column is sent to a conventional separation process where pure ethane, propane, butanes, and natural gasoline can be obtained. The residual gas (top product) is compressed and delivered as sales gas.

The rigorous mathematical model employed for this analysis contained 1830 equations with 330 measurements and 1602 unmeasured variables.

4.3.2. Observability Analysis with DM and EDM.

Next, we included a table with the observability results (Table 6)

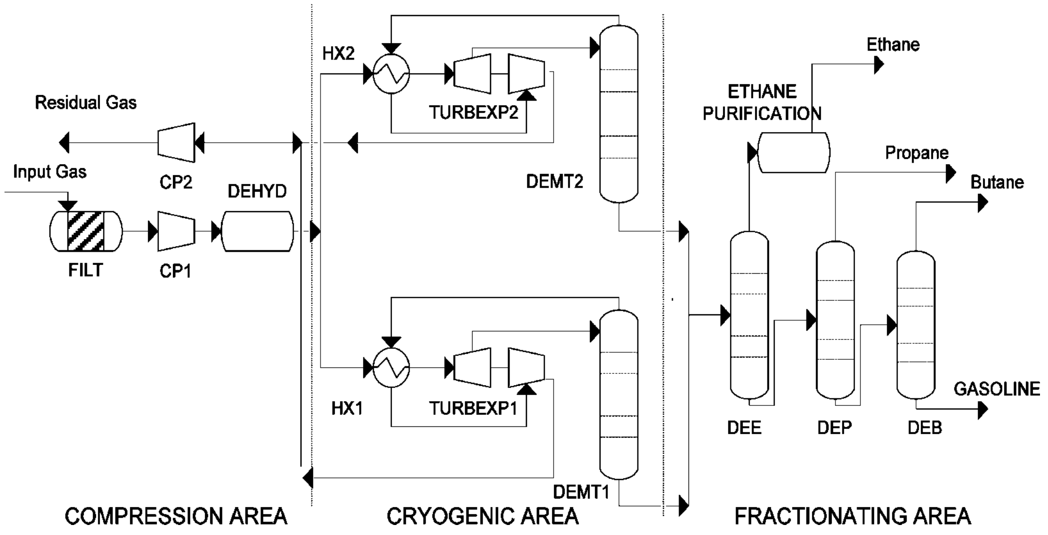


Figure 14. Ethane plant simplified scheme.

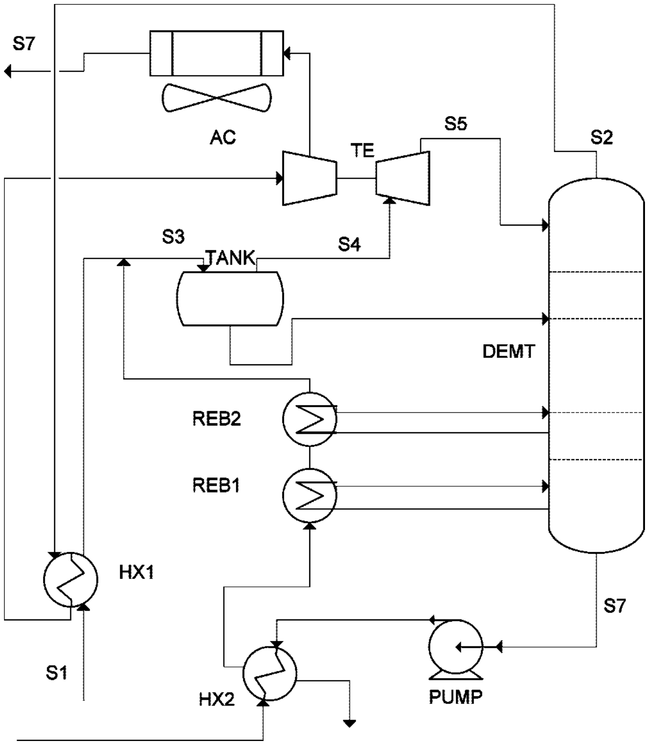


Figure 15. Cryogenic sector of the ethane plant.

Table 6. DM and EDM Observability Results Obtained for the Ethane Plant Model

amount of	DM	EDM	Δ
observable variables	929	929	0
unobservable variables	673	673	0
1×1 subsystems	767	777	+1.3%
linear subsystems	758	776	+2.3%
nonlinear subsystems	26	20	−23%

obtained by the application of both the DM and the EDM, for the mathematical model of the ethane plant.

In this case, an increase of 1.3% in the production of low dimension subsystems (1×1) and 2.3% in the generation of linear subsystems is reached. Meanwhile, the quantity of nonlinear subsystems is reduced by 23%.

If an integral analysis of the three cases is made, a 12% average in the production of both 1×1 dimension subsystems and linear subsystems is obtained. This implies an increase in the subsequent speed of resolution for these systems, because the calculation is reduced to a simple arithmetical operation. On the other hand, and even more important, our hypothesis is verified because a 43% average reduction in nonlinear subsystems generation was achieved.

5. Conclusions

In this work, we have proposed an improvement to the observability analysis algorithm known as Direct Method (DM). This new variant, called Extended Direct Method (EDM), consists of the implementation of a module that quantifies the nonlinearity inherent in a system of equations. This module guides the traditional DM toward a decomposition of the mathematical model into the possible subsystems with the smallest degree of nonlinearity.

The new algorithm was tested for different industrial case studies. The results indicate that in all cases, the EDM was able to yield an ordering that can reduce the nonlinearity of the subsystems obtained through the observability analysis procedure. At the same time, it can successfully increase the generated quantity of minimum size subsystems.

As it was explained above, one of the instrumentation design steps is the redundancy analysis, which is based on the symbolic derivation of functions. It was verified in section 4.1.4 that, as expected, the implemented changes in the observability algorithm produced improvements in the redundancy analysis process. In the first place, we achieved a higher knowledge level of the process redundancy than the one obtained by applying the original DM. This is essential for making decisions related to the addition or removal of sensors in the plant. Second, we achieved a reduction in the time required to perform the global process of instrumentation design, which is carried out by using the DSS. This reduction is possible because the application of the EDM allows one to achieve a significant reduction in the necessary time to perform the redundancy analysis.

There is also another kind of analysis in process plants that could be favored with the use of the EDM. Such is the case of the data reconciliation task. This task, which is carried out after the instrumentation design, involves comparing data obtained by field measurements with values estimated through the model equations, to detect possible failures in measurement instrumentation. To perform this task, it is necessary to carry out the resolution of just the equation subsystems that are obtained by the application of the observability algorithm, without taking into account the equations that were not included in the subsystems (redundant or not assigned equations). In this way, a reduction in the nonlinearity degree of the obtained decomposition involves solving a larger number of linear subsystems, which can be solved by some method of direct resolution, thus reducing the computational requirements to perform the reconciliation task.

Another study that may be significantly favored with the EDM partitioning strategy is the simulation of process plants. This task normally requires the numerical resolution of the mathematical model that represents the process. For complex systems, the resolution involves a great computational effort. In that sense, the EDM could be employed as a partitioning step of the mathematical model, previous to the calculations. In this way, partitioning into subsystems would reduce the computational cost of its numerical resolution. This is feasible because the user has the possibility of applying a direct resolution method for linear subsystems and an iterative one for nonlinear subsystems, taking advantage in this way of the larger number of linear subsystems obtained with the EDM.

In this way, we can appreciate that the favorable impact of incorporating the treatment of the nonlinearity degree to the DM can result in benefits that go beyond the specific task of instrumentation design. This fact broadens the application rank of this system-decomposition methodology.

Acknowledgment

We gratefully acknowledge the economic support given by the Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) through grant PIP 5930 and the Universidad Nacional del Sur through grant PGI 24/N019. We would like to thank the Agencia Nacional de Promoción Científica y Tecnológica, SeCyT, Argentina, for their grants, PICTO UNS 917 and PICT B 2002 11-12778, which are part of the "Programa de Modernización Tecnológica", Loan Contract BID 1728/OC-AR.

Appendix

Nomenclature.

ABS = absorber

AC= air cooler

C = bigraph node set corresponding to columns of the matrix N

COND = condenser

CP = compressor

DCOL = distillation column

DEB = debutanizer

DEE = deethanizer

DEHYD = dehydratation

DEM = data entry module

DEMT = demethanizer

DIV = divider

DM = Direct Method

DRY = dryer

DSS = Decision Support System

EDM = Extended Direct Method

F(A) = molar flow of stream A

FILT = filtering

HC = horizontal-block column set

HR = horizontal-block row set

HX = heat exchanger

MGM = model generator module

MIX = mixer

MMB = maximum matching in a bigraph

N = occurrence matrix

NLD = nonlinearity degree

NDV = nonlinearity degree for variables

OA = observability analysis

P(A) = pressure of stream A

P_{M} = maximum matching

R = bigraph node set corresponding to rows of the matrix N

RA = redundancy analysis

RAM = redundancy analysis module

REAC = reactor

REB = reboiler

SC = square-block column set

SEP = separator

SR = square-block row set

T(A) = temperature of stream A

t = number of terms of an equation

t_v = number of terms where variable v takes part

TE = turbo expander

V = valve

v = variable

VC = vertical-block column set

VR = vertical-block row set

w = terms punctuation

NDE = nonlinearity degree for equations

X(B,A) = molar fraction of component B in stream A

Literature Cited

(1) Worden, K.; Burrows, A. Optimal sensor placement for fault detection. *Eng. Struct.* **2001**, *23*, 885–901.

(2) Muske, K.; Georgakis, C. Optimal measurement system design for chemical processes. *AIChE J.* **2003**, *49*, 1488–1494.

(3) Lohner, R.; Camelli, F. Optimal placement of sensors for contaminant detection based on detailed 3D CFD simulations. *Eng. Comput.* **2005**, *22*, 260–273.

(4) Singh, A.; Hahn, J. Sensor location for stable nonlinear dynamic systems: Multiple sensor case. *Ind. Eng. Chem. Res.* **2006**, *45*, 3615–3623.

(5) Bhushan, M.; Rengaswamy, R. Design of sensor location based on various fault diagnostic observability and reliability criteria. *Comput. Chem. Eng.* **2000**, *24*, 735–741.

(6) Singh, A.; Hahn, J. Determining optimal sensor locations for state and parameter estimation for stable nonlinear systems. *Ind. Eng. Chem. Res.* **2005**, *44*, 5645–5659.

(7) Delattre, C.; Dochain, D.; Winkin, J. Observability analysis of nonlinear tubular (bio) reactor models: a case study. *J. Process Control* **2004**, *14*, 661–669.

(8) Maurya, M.; Rengaswamy, R.; Venkatasubramanian, V. Application of signed digraphs-based analysis for fault diagnosis of chemical process flowsheets. *Eng. Appl. Artif. Intell.* **2004**, *17*, 501–518.

(9) Narasimhan, S.; Rengaswamy, R. Quantification of performance of sensor networks for fault diagnosis. *AIChE J.* **2007**, *53*, 902–917.

(10) Vazquez, G.; Ferraro, S.; Carballido, J.; Ponzoni, I.; Sanchez, M.; Brignole, N. The software architecture of a decision support system for process plant instrumentation. *WSEAS Trans. Comput.* **2003**, *2*, 1074–1079.

(11) Ferraro, S.; Ponzoni, I.; Sanchez, M.; Brignole, N. A symbolic derivation approach for redundancy analysis. *Ind. Eng. Chem. Res.* **2002**, *41*, 5692–5701.

(12) Romagnoli, J.; Sanchez, M. *Data Processing and Reconciliation for Chemical Process Operations*; Academic Press Inc.: San Diego, CA, 1999.

(13) Ponzoni, I.; Sanchez, M.; Brignole, N. A direct method for structural observability analysis. *Ind. Eng. Chem. Res.* **2004**, *43*, 577–588.

(14) Vazquez, G.; Ponzoni, I.; Sanchez, M.; Brignole, N. ModGen: A model generator for instrumentation analysis. *Adv. Eng. Software* **2001**, *32*, 37–48.

(15) Carballido, J.; Ponzoni, I.; Brignole, N. A novel application of evolutionary computing in process systems engineering. *Lect. Notes Comput. Sci.* **2005**, *3448*, 12–22.

(16) Carballido, J.; Ponzoni, I.; Brignole, N. Using computational intelligence and parallelism to solve an industrial design problem. *Lect. Notes Comput. Sci.* **2006**, *4140*, 188–197.

(17) Carballido, J.; Ponzoni, I.; Brignole, N. CGD-GA: A graph-based genetic algorithm for sensor network design. *Inf. Sci.* **2007**, *177*, 5091–5102.

(18) Carballido, J.; Ponzoni, I.; Brignole, N. SID-GA: An evolutionary approach for improving observability and redundancy analysis in structural instrumentation design. *Comput. Ind. Eng.*, available online, doi: 10.1016/j.cie.2008.09.001.

(19) Gibbons, A. *Algorithmic Graph Theory*; Cambridge University Press: Newcastle, UK, 1994.

(20) Tarjan, R. Depth-first search and linear graph algorithms. *SIAM J. Comput.* **1972**, 146–160.

(21) Pothen, A.; Fan, C. J. Computing the block-triangular form of a sparse matrix. *ACM Trans. Math Software* **1990**, *16*, 303–324.

(22) Dulmage, A. L.; Mendelsohn, N. S. Two algorithms for bipartite graphs. *SIAM J. Comput.* **1963**, *11*, 183–194.

(23) Flex: The Fast Lexical Analyzer. http://flex.sourceforge.net/.

(24) Bike, S. *Design of an Ammonia Synthesis Plant*; CACHE Case Study, Department of Chemical Engineering, Carnegie Mellon University, 1985.

Received for review March 19, 2008
Revised manuscript received November 17, 2008
Accepted February 3, 2009

IE800449T