Data Reconciliation as a Framework for Chemical Processes Optimization and Control

by

Chouaib BENQLILOU

March 2004

Submitted to the Department of Chemical Engineering in partial fulfillment of the requirements for the degree of Doctor of Science at the Universitat Politècnica de Catalunya.

Directed by:

Prof. Luis Puigjaner Corbella

Dr. Antonio Espuña Camarasa

Department of Chemical Engineering

Universitat Politècnica de Catalunya

This document has been prepared using LaTeX, LyX, Dia, Grip, Linux and other free software. The author encourages the use of this software in order to prepare high quality documents.
All the names of the different computer programs cited in this thesis (MATLAB, CPLEX, GAMS, etc) are © of their owners.

to my family

... Since all our measurements and observations are nothing more than approximations to the truth, the same must be true of all calculations resting upon them, and the highest aim of all computations made concerning concrete phenomenon must be to approximate, as nearly as practicable, to the truth. But this can be accomplished in no other way than by a suitable combination of more observations than the number absolutely requisite for the determination of the unknown quantities...

From Theory of the Motion of the Heavenly Bodies Moving about the Sun in Conic Sections, Gauss, 1809.

Summary

This thesis presents, discusses and compares a set of methodologies and several appropriate combinations of them, to provide accurate estimation of process variables, either for steady-state or dynamic systems.

Firstly, the accuracy of estimated measurements is improved through the proposal of novel Data Reconciliation techniques. The proposal combines data-based and model-based filtering and also consider the presence of time-delays between sampled data.

Secondly, measuring network design and its optimal use are addressed. Thus, the measuring device number, their type and their location for optimum reliability and accuracy of measurement at lowest possible cost are determined.

The first part of this thesis provides procedures for accuracy estimation in dynamic evolving processes. These procedures rely on combining data-based filtering and model-based filtering. One technique combines a Moving Average filter and a steady-state Data Reconciliation technique sequentially. The resulting estimator presents the important statistic feature of being unbiased. Additionally, this estimator provides high accuracy estimation and good tracking for dramatic dynamic changes of process variables, when compared with other techniques. The other technique performs a wavelet analysis as a former step for reconciling dynamic systems. The wavelet technique catches or extracts the process measurement trends that are later made consistent with the dynamic process model. As a consequence of this technique high estimation accuracy is provided. Additional advantages of applying this technique over the current techniques are the easy handling of distinct sample times and evaluating the variance of dynamic variables. Furthermore, this thesis addresses an important aspect regarding dynamic Data Reconciliation: how to improve the accuracy estimation when the process is faced with the presence of time-delay. This problem was overcome in a simple and efficient way by proposing a time-delay estimation method that works in conjunction with the Measurement Model adopted within the Data Reconciliation technique. The presented time-delay estimation method determines the existing delay by maximizing the correlation of the process variables using genetic algorithms.

Summary

The second part of this thesis addresses the design of sensor networks, the proposed strategy allows the optimal selection and placement of measuring devices. The proposal deal with different sensor placement aspects: variation in design, retrofit, hardware redundancy and available sensor type.

The sensor placement procedure was extended to deal with dynamic systems by taking advantages of dynamic variable classification and dynamic Data Reconciliation. The procedure to locate sensors in dynamic systems aims at maximizing the performance of Kalman filtering using accuracy as its main performance index. To accomplish this, both the measurement noise and the observation matrices are manipulated. The solution strategy has been implemented in academic and in the Tennessee Eastman challenge problems showing promising results. The resulting optimization problem was solved satisfactorily either by exhaustive search or using genetic algorithms based optimization. The profile of the relative increase of the system performance along the sensor network and the associate investment cost gives the designer all the alternatives for making an adequate decision.

Additionally, reliability is considered by combining quantitative process knowledge and fault tree analysis, providing an efficient way to improve its evaluation. It is important to state that the possibility to use inferential sensors based in an Artificial Neural Network model instead of physical sensors, and their incorporation within reliability and reconciliation procedures was a paramount consideration throughout this work.

Finally, this thesis also provides two frameworks, one for sensor placement and the second for Data Reconciliation. Both proposed frameworks have been designed, specified and validated following the guidelines of the new standards and trends in developing component-based application (e.g. UML^{TM}, CAPE-OPEN). These frameworks can include the above mentioned algorithms and can be extended to include other existing or futures approaches efficiently.

Chouaib BENQLILOU

Acknowledgments

I wish to express my gratitude to my thesis supervisors Prof. Luis Puigjaner and Dr. Antonio Espuña. I am truly grateful for the opportunity of having worked under their supervision and for their guidance and support throughout my research work. I sincerely thank Dr. Moisés Graells, Rodolfo V. Tona and Estanislao Musulin for their invaluable suggestions throughout this research.

I would like to express my appreciation for all graduate students who made all together an enjoyable space for work. To all the administrative staff my sincere thanks especially for Francina Solé. I also express my deepest gratitude for all my friends for the aid and understanding I received from them. Additionally, my gratitude to all those who have helped enrich my life.

I heartily thank my sweet girlfriend, Dr. Souad Elaamrani without her love, patience and subtle encouragement yet strict when it was needed, this research would not have been possible. My eternal gratitude goes to my grandparents, parents, sisters, brothers, nieces and nephews for their continued love and support.

Chouaib BENQLILOU

تشكرَاتي

اَلحمدُ وَ الشُّكرُ لِلّٰهِ الَّذِي اعطَاني العَقلَ و الصحَّة لأُ فكر في خَلقِهِ. اَشكرك يَا الله وَ اتمنَا ان تهديني اليك.

أُريد ان اقدم الىَ امي و ابي و جميع افرَاد عَائِلتَى هذَا المجهُود و هذَا الإنجَاز كمقَابل مسَاعدتهم و تشجيعهم علَى تحقُّقه. اِلَى جدي ادريس اتمنَّى ان يكُونَ هذَا العمل رحمة عليه لطمُوحه و ارَادته في تعلّمنَا.

أُريد ايضًا ان اكتب هَذِه الآيَات التَى كَانت ترَافقني طوَال هذه الفترَات من حيَاتي.

شعَيب بن قليلُو

١٨٩ اِنَّ فِي خلقِ السَمٰوٰت و الارض و اختلَاف اليلِ و النَّهَار ءلَايٰت لأُولي الالبٰب
١٩٠ الذين يذكرُونَ اللّٰهَ قِيٰمًا و قُعُودًا و عَلٰى جنُوبهم و يتفكَّرن في خَلق السَمٰوٰت و الَارض ، ربّنَا مَا خلقت هٰذَا بٰطلًا سبحٰنك ، فقنَا عَذٰب النَّار ١٩١

أَلقُرآنٌ الكَريم ، سُورة ءَال عمرَان

Contents

Contents

Contents

Contents

List of Figures

List of Figures

List of Figures

List of Tables

List of Tables

Chapter 1

Introduction

1.1 General overview

Reliable and accurate process measurements are crucial for the improvement of chemical, petrochemical and other material processing industries. In general, the decision-making for the accomplishment of the planned objectives based on Process Modeling, Parameter Estimation and/or Process Optimization performance depends strongly on the data collected from the plant instrumentation and laboratory analyzers.

Unfortunately, data inherently contains inaccurate information since measurements are obtained with imperfect instruments. Additionally, the arduous conditions of the real industrial environment where measurement devices are installed added to the maintenance/calibration and quality of sensors accumulate those measurement errors.

In general, the errors that affect the measured data could be categorized into two main classes:

1. **Random errors**: this type of errors could be arising from fluctuations and/or disturbances in non-controlled conditions that imply an irreproducibility of measurements. These errors are usually considered independent and normally distributed with zero means. Measurements with such errors will be statistically inconsistent with the process model constraints.

2. **Non-random errors** or **gross errors**: this type of error is generated from non-random events and could be subdivided into measurement-related errors (e.g. malfunctioning instruments) and process-related errors (e.g. process leaks, uncounted losses, modeling errors).

Using these data without any filtering technique may affect the achievement of optimal plant performance (e.g. quality, yield or due-date) and even could drive the

plant to an unsafe situation. The objective then is to tight up the instrument confidence region to a level where the resulting information can be accepted for valid decision-making. Therefore, conditioning process measurements must be a common practice, so that random and gross errors affecting data accuracy and reliability are compensated or even eliminated.

1.1.1 Filtering techniques

Filtering of data refers to finding an estimate of the true value of the measured variables based upon some additional information (e.g. process model). Generally, filtering the measurements obtained from a process is mainly based on two main approaches: statistic-based filtering and model-based filtering.

Processes that lack an accurate model might rectify data by using analog or digital filters. Inadequate sampling frequency converts a high frequency signal into an artificial low-frequency signal. This phenomenon is known as *signal aliasing*. Analog filters are used to pre-filter process data before sampling and prevent aliasing. Digital filters are used afterward to further attenuate random errors (high-frequency noise) in process values. These filters (e.g. Exponential, Moving average) are basically used in the pre-processing task as a data conditioning and they are usually incorporated in many Distributed Control Systems (DCS). Furthermore, in the pre-processing procedure the larger outlier measurements can be eliminated by setting the permissible lower and upper bounds of process variables.

In the model-based filtering approach, commonly called ***data reconciliation*** or data rectification, the process model could be either based on first principles (such as energy balances, chemical equilibrium relationships, etc), on an empirical model (such as artificial neural network models) or on a combination of both. The estimation of process variables is obtained by adjusting process measurements to satisfy the process model constraints. By this way, the knowledge on the process model is exploited in terms of redundancy, which restricts the filtering of the interconnected process variables.

Depending on the process model constraints, Data Reconciliation can be classified as steady-state or dynamic and also as linear or nonlinear.

It is important to note that the data pre-processing task should be performed before Data Reconciliation (e.g. it is better to not send any measured value to the Data Reconciliation step instead of sending a negative value of a flow). Thus, data pre-processing and reconciliation should not be considered as competitive procedures but as complementary see Figure 1.1.

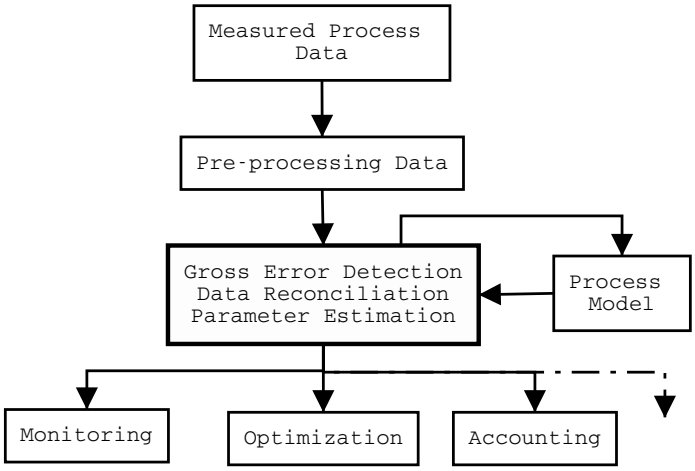


Figure 1.1: Data Reconciliation and plant optimization hierarchy.

1.1.2 Data Reconciliation system

The information obtained from the *statistical model of process measurements* and the *process model* generates redundancy that is the basis of the improvement of process measurements accuracy when a technique such as Data Reconciliation is applied.

Therefore, a fundamental task related with Data Reconciliation is assessing the redundancy of the process under consideration to see whether there is a need for additional instrumentation or any upgrade of the existing sensor network is required.

In the Data Reconciliation (DR) system presented in Figure 1.2 it is shown that the system allows to increase the confidence of the measurements and also provides the most probable value of unmeasured variables (***coaptation***) including model parameters.

Following the reconciliation calculation, all the reconciled measurements and the estimated values of the non-measured variables strictly obey to a pre-determined process model. Additionally, DR also calculates the accuracy of its own results through the computation of the standard deviation of each reconciled or estimated value.

Furthermore, DR system can detect faulty sensors or process leaks and also allows to determine the optimal placement of measuring devices.

Moreover, it is evident that we get better understanding of the process, which allows an operation closer to specification and an estimation of the equipment efficiency, reducing modeling error. These aspects permit an increase in the plant benefits, and justify maintenance tasks for both instruments (calibration) and equipment (cleaning, follow up).

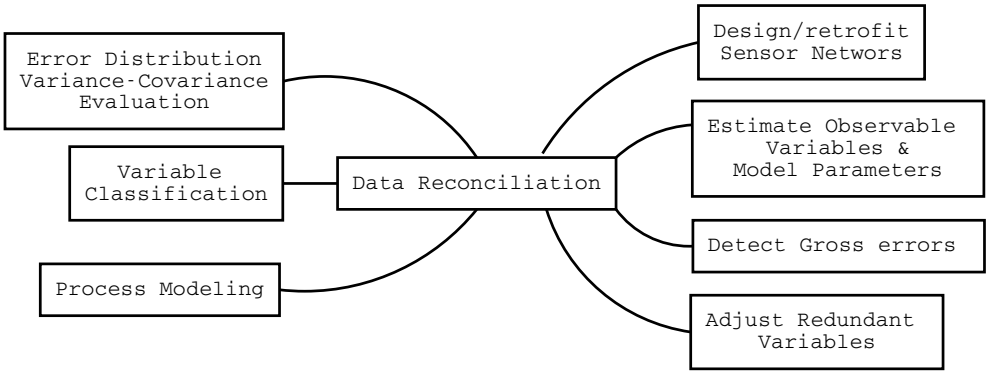


Figure 1.2: Data Reconciliation system.

Mathematically, the associated optimization problem to be solved can be stated as a weighted optimization problem. Data are adjusted to be as similar as possible to the raw measurements but statistically consistent with a set of equality and inequality constraints involving measured, unmeasured and fixed variables.

The equality constraints are in general the mass, composition and energy balances, however, equilibrium or reactions equations could also be used if they are considerably "accurate". The inequality constraints are the variable bounds (e.g. non-negativity constraints imposed on the flow rates and concentrations).

The high dimension of system equations, nonlinearity, bias and missing key variables value complicate the task of matching the process data with the process model. To overcome this, the procedure should start with a reduced process model of the whole plant (e.g. mass balance of steady-state process and after matching this basic balance model, model complexity can be enhanced). Thus, it is wisdom to increase the model complexity gradually since modeling error could affect seriously the reconciliation results.

Data Reconciliation technique is widely applied now days in various chemical, petrochemical, power plant and other material processing industries, it is applied off-line or in connection with on-line applications, such as process optimization, or advanced process control as presented by Abu-el zeet et al. (2002).

When plant managers realized the benefit of using a Data Reconciliation techniques and commercial software for Data Reconciliation being available (e.g. DATREC from Elf Central Research, DATACON^{TM} from Simulation Sciences, RtOpt form Aspentech, VALI from Belsim S.A., Sigmafine from OSI, etc) the number of Data Reconciliation applications increased considerably in a wide range of material processing industries focusing different areas as shown in Figure 1.1.

The analysis of the industrial applications presented in Figure 1.1 is supplied according to the major industrial types of applications for Data Reconciliation techniques (Narasimhan and Jordache (2000); Romagnoli and Sánchez (2000)). Mainly it can be possible to distinguish two major types of applications as follows:

- Process unit reconciliation (material and energy balancing for process units) especially distillation or separation columns such as naphtha cracker. Application of Data Reconciliation to reactors such as a catalytic reformer, catalytic cracker unit or pyrolysis reactors has been also reported in the literature.
- Plant-wide reconciliation production and utilities accounting by the process (e.g. energy consumed). Many refineries are already saving a significant amount of money by this plant-wide reconciliation. Currently, most integrated systems for process simulation, optimization and control include Data Reconciliation system that precedes all applications that make use of process data.

1.2 Background

1.2.1 Data Reconciliation

1.2.1.1 Linear steady-state data reconciliation

Steady-State Data Reconciliation (SSDR) was first addressed in the pioneer work of Kuehn and Davidson (1961). The authors adjust process data to satisfy mass balance. Therefore, they formulated the SSDR as a Weighted Least-Square (WLS) optimization problem (Eq. (1.1)) subject to mass balances (Eq. (1.2)).

$$\min_{\widehat{\mathbf{y}}} \left((\mathbf{y}-\widehat{\mathbf{y}})^T \mathbf{Q}^{-1} (\mathbf{y}-\widehat{\mathbf{y}}) \right) = \min_{\widehat{\mathbf{y}}} \left(\mathbf{a}^T \mathbf{Q}^{-1} \mathbf{a} \right) = \min_{\widehat{\mathbf{y}}} \left(\sum_i \left(\frac{\widehat{\mathbf{y}}_i - \mathbf{y}_i}{\sigma_i} \right)^2 \right) \tag{1.1}$$

subject to:

$$\mathbf{A}\widehat{\mathbf{y}} = \mathbf{0} \tag{1.2}$$

where **A** is the incidence matrix representing steady-state mass balance, it is considered that all process variables involved in the mass balance were measured (i.e. **A** has a full row rank) and the measurement do not contain gross errors. The term **a** is the adjustment vector that is the difference between the measurement vector **y** and estimated vector $\widehat{\mathbf{y}}$ and **Q** is the variance-covariance matrix. The measurement errors follow a normal distribution with zero-mean and a known variance $\mathbf{Q}_{ii} = \sigma_i^2$.

The square of standard deviation σ_i^2 is the weight on the measurement adjustment *i*. Variables known with a high certainty (low variability) are given a large weight

and variables with high variability implies that the measurement is less accurate and received less weight in Data Reconciliation procedure.

The effect of random errors on measurement is modeled as additive contributions. The relation between the measured value, true value and random error in the measurement es expressed by Eq. (1.3).

$$\mathbf{y} = \mathbf{y}^* + \varepsilon \tag{1.3}$$

where $\mathbf{y}^*$ is the vector of true value (noise free) and ε is the vector of random error. The random error ε usually oscillates around zero. Its characteristics can be described using statistical properties of random variables. Its mean or *expected value* is zero and its variance is given by:

$$var(\varepsilon_i) = E[\varepsilon_i^2] = \sigma_i^2 \tag{1.4}$$

where σ_i is the standard deviation of the measurement error ε_i.

The problem above described can be solved analytically by using Lagrange multipliers as shown in Eq. (1.5).

$$\widehat{\mathbf{y}} = \mathbf{y} - \mathbf{Q}\mathbf{A}^T \left(\mathbf{A}\mathbf{Q}\mathbf{A}^T\right)^{-1} \mathbf{A}\mathbf{y} \tag{1.5}$$

The variance-covariance matrix $\widehat{\mathbf{Q}}$ of the reconciled variables $\widehat{\mathbf{y}}$ is given by:

$$\widehat{\mathbf{Q}} = \mathbf{Q} - \mathbf{Q}\mathbf{A}^T \left(\mathbf{A}\mathbf{Q}\mathbf{A}^T\right)^{-1} \mathbf{A}\mathbf{Q} \tag{1.6}$$

Inspired from this work the researchers in this area focused their work in two major investigation lines. Extension of the steady-state linear Data Reconciliation to nonlinear and dynamic systems and secondly to deal with the presence of gross error, which alters the hypothesis of normal error distribution.

1.2.1.2 Precision versus accuracy

Standard deviation is a measure of the measurement precision. The smaller the σ, the more precise is the measurement and the higher the probability that the random error will be close to zero. Accuracy describes the correctness of a measured value, that is, how close is the instrument reading to the "true" value.
Mean-Square Error (MSE) is defined as the expected value of the square of the deviation between the estimate and the true value. MSE can be considered as a measure of accuracy. If no bias is present the MSE is equal to the variance of the estimate. Therefore, in such situations precision and accuracy can be interchangeable.

1.2.1.3 Lagrange multipliers

Since the constrained least-squares estimation will be encountered many times in this thesis, it is worthwhile to explain the minimization through Lagrange multipliers (Mah (1990)). The Lagrangian for the estimation problem is:

$$L = \mathbf{a}^T \mathbf{Q}^{-1} \mathbf{a} - 2\lambda^T (\mathbf{A}\mathbf{y} + \mathbf{A}\mathbf{a}) \tag{1.7}$$

since $\mathbf{Q}$ is positive definite and the constraints are linear, the necessary and sufficient conditions for minimization are obtained by setting the partial derivatives of Eq. (1.7) with respect to variable λ and $\mathbf{a}$ to zero:

$$\frac{dL}{d\lambda} = 0 \tag{1.8}$$

and

$$\frac{dL}{d\mathbf{a}} = 0 \tag{1.9}$$

The differentiation is readily carried out considering that for any vectors $\mathbf{x}$, $\mathbf{y}$, and any matrix $\mathbf{A}$, the product $\mathbf{y}^T\mathbf{A}\mathbf{x}$ is a scalar. The differentiation of a product obeys the usual product rule:

$$\frac{\partial \mathbf{y}^T \mathbf{A}\mathbf{x}}{\partial \mathbf{x}} = \frac{\partial \mathbf{y}^T}{\partial \mathbf{x}} \mathbf{A}\mathbf{x} + \frac{\partial \mathbf{x}^T}{\partial \mathbf{x}} \mathbf{A}^T \mathbf{y} \tag{1.10}$$

In the special case for which $\mathbf{A}$ is symmetric and $\mathbf{y} = \mathbf{x}$:

$$\frac{\partial \mathbf{x}^T \mathbf{A}\mathbf{x}}{\partial \mathbf{x}} = 2\mathbf{A}\mathbf{x} \tag{1.11}$$

Applying these relations to Eqs. (1.8) and (1.9) it is possible to obtain:

$$\mathbf{A}\mathbf{a} = -\mathbf{A}\mathbf{y} \tag{1.12}$$

and

$$\mathbf{a} = \mathbf{Q}\mathbf{A}^T \lambda \tag{1.13}$$

Substituting Eq. (1.13) in Eq.(1.12) it is possible to get:

$$\lambda = -(\mathbf{A}\mathbf{Q}\mathbf{A}^T)^1 \mathbf{A}\mathbf{y} \tag{1.14}$$

Finally, the substitution of Eq. (1.14) in Eq. (1.13) yields the solution in Eq. (1.5).

1.2.2 Redundancy and observability classification

In the case of linear steady-state processes, the presence of unmeasured process variables implies that matrix **A** is not full row rank, which may complicate the inverse calculation during the procedure of reconciliation or coaptation since a singular matrix is generated as it can be seen in Eq. (1.5).

This numerical computation arises also for nonlinear cases: the measured (respectively unmeasured) process variables may not have a unique value through Data Reconciliation (respectively coaptation) thus, the optimization algorithm assigns them whatever values (i.e. possibly meaningless and erroneous).

Then, it is important to classify unmeasured variables as observable and unobservable so that any calculated values of the unobservable quantities will be discarded. Similarly, measured variables are classified as redundant and non-redundant since the non-redundant (just measured) process variables involve that the optimization algorithm might fail.

It is clear that the performance and formulation of the Data Reconciliation optimization problem and statistical interpretation are strongly linked to the **uniqueness of the estimated process variables**. Thus, this must be analyzed by a careful variable classification procedure.

Two main variables classification approaches for linear as well as nonlinear problems were reported in the literature: (i) graph-theory and (ii) quantitative process model.

Defining or classifying variables into non-redundant, redundant, observable and unobservable was first addressed for the steady-state linear processes by Vaclaveck (1969) and defined as:

- An unmeasured variable is defined as unobservable if it cannot be uniquely determined through the measured variables. Thus, its resulting estimated value is meaningless.

- A measured variable is defined as non-redundant if deletion of its measurements makes this variable unobservable, that is in order to be able to estimate a non-redundant variable, its measurement is absolutely necessary since this variable is not related to other measured variables through the process model (e.g. matrix **A** in the linear case).

The classification problem has been also addressed by many other authors applying different mathematical based approaches. In this sense it is worth mentioning the work of Vaclaveck (1969) where a graph-theory approach is provided to generate a *reduced balance scheme* containing only redundant measurements. Kretsovalis and

Mah (1988) reported the application of graph-theory to the identification of observable and unobservable process variables. Crowe et al. (1983) propose for the same class of processes a projection matrix for classifying the variables. Madron and Veverka (1992) use successfully the Gauss-Jordan rearrangement to classify variables for linear systems.

In many industrial applications Data Reconciliation technique is just based in mass balances, but in practice it can also consider energy and component balances, since it is common for temperature and concentration to measure them along with flows. Product of these variables will make these constraints bilinear.

The extension of the matrix projection presented earlier by Crowe et al. (1983) to deal with bilinear processes was addressed by Crowe (1986), where they attempted to transform the bilinear problem to a linear one by process variable substitution (i.e. the product of two variables is substituted by a *unique new variable*).

A completely different approach was proposed by Simpson et al. (1991) to solve the bilinear Data Reconciliation problem. Instead of obtaining a reduced set of constraints involving only measured variables, their approach completely eliminate all constraints. This is accomplished by dividing the set of variables into dependent and independent variables, similar to the approach used by constrained nonlinear optimization techniques. The constraints are utilized to obtain explicit relationships between the dependent and independent variables, which are used to eliminate all dependent variables from the objective function and to obtain a reduced unconstrained problem in the space of independent variables.

Sánchez and Romagnoli (1996) determined the matrix projection for linear and bilinear Data Reconciliation by using orthogonal factorization namely Q-R decomposition approach. Sánchez and Romagnoli (1996) argued that the Q-R decomposition method for constructing the projection matrix is more efficient than the Crowe et al. (1983) approach. Hodouin and Everell (1980) analyzed redundancy and observability of nonlinear processes based on linearization method.

Albuquerque and Biegler (1996) extend the variable classification procedure to dynamic systems by using efficient sparse linear algebra methods and introduce the concept of *collective* redundancy. First they linearize and discretize the dynamic nonlinear model, then they apply the properties of observability and redundancy and derive tools necessary to classify variables.

1.2.3 Probabilistic formulation of the Data Reconciliation problem

1.2.3.1 Maximum Likelihood Estimation

If the measurement error distribution follows a normal distribution, the Data Reconciliation problem can be posed as a Maximum Likelihood Estimation (MLE) prob-

lem, where the probability of the estimated (reconciled) process variables ($\widehat{\mathbf{y}}$) is maximized given the measurement set ($\mathbf{y}$) as shown in Eq. (1.15).

$$\max_{\widehat{\mathbf{y}}} P\{\widehat{\mathbf{y}}|\mathbf{y}\} \tag{1.15}$$

According to Bayes' theorem, the probability of the process variables given the measurements can be written in terms of the probability of the measurements given the reconciled process variables, the probability density function of the process variables $P\{\widehat{\mathbf{y}}\}$ and the probability density function of the measurements $P\{\mathbf{y}\}$.

$$\max_{\widehat{\mathbf{y}}} P\{\widehat{\mathbf{y}}|\mathbf{y}\} = \max_{\widehat{\mathbf{y}}} \frac{P\{\mathbf{y}|\widehat{\mathbf{y}}\}P\{\widehat{\mathbf{y}}\}}{P\{\mathbf{y}\}} \tag{1.16}$$

The denominator term (independent of $\widehat{\mathbf{y}}$) acts as normalizing constant and does not need to be further considered for optimization. The first term in the numerator represents the probability density of the measurements given the reconciled process variables, $\widehat{\mathbf{y}}$, which is the distribution of the measurements errors $P(\widehat{\mathbf{y}}-\mathbf{y})$. Finally, $P\{\widehat{\mathbf{y}}\}$ is a binary assumption, that is equal to 1 if the constraints are satisfied (under this assumption the $P\{\widehat{\mathbf{y}}\}$ term is converted to a set of constraints and the original problem is converted to a constrained optimization) and equal to 0 otherwise.

$$P(\widehat{\mathbf{y}}-\mathbf{y}) = P(\varepsilon) \simeq N(\mathbf{0},\mathbf{Q}) \tag{1.17}$$

If sensor errors are independents the product of this probability over all sensors yields to:

$$P\{\widehat{\mathbf{y}}|\mathbf{y}\} = \prod_i exp\left\{-\frac{1}{2}\left(\frac{\widehat{\mathbf{y}}_i-\mathbf{y}_i}{\sigma_i}\right)^2\right\} = exp\left\{-\frac{1}{2}\sum_i\left(\frac{\widehat{\mathbf{y}}_i-\mathbf{y}_i}{\sigma_i}\right)^2\right\} \tag{1.18}$$

Taking the negative logarithm of the maximization of the objective function represented in Eq. (1.18) results in the minimization of the conventional WLS formulation as is shown in Eq. (1.1*)*. The symmetric and positive definite matrix $\mathbf{Q}$ contains the variance-covariance elements of the measurement errors and thus quantifies the uncertainty in each measured value.
Then the **success of Data Reconciliation technique relies on the hypothesis that the error is normally distributed** and on the evaluation of matrix $\mathbf{Q}$.

1.2.3.2 Variance-covariance calculation

The standard deviation of a measurement error plays an important role in DR. Assuming the process is truly at steady-state, the mean and the covariance matrix can be estimated by:

$$\overline{\mathbf{y}}_i = \frac{1}{n}\sum_{k=1}^{n}\mathbf{y}_{i,k} \tag{1.19}$$

$$cov\left(\mathbf{y}_i,\mathbf{y}_j\right)=\frac{1}{n-1}\sum_{k=1}^{n}\left(\mathbf{y}_{i,k}-\overline{\mathbf{y}}_i\right)\left(\mathbf{y}_{j,k}-\overline{\mathbf{y}}_j\right) \tag{1.20}$$

where n is the width of the window in which the process is truly at steady-state. This procedure based on Eqs. (1.19) and (1.20) is referred as the *direct method* of estimation of the variance-covariance matrix of the measurements error.

When the system is not steady or the steady-state assumption could not be considered, the use of the so-called *direct* method adopting leads to errors in estimating $\overline{\mathbf{y}}$, and so $\mathbf{Q}$, which reduces the performance of Data Reconciliation.

The so-called *Indirect* method presented by Almasy and Mah (1984) tries to overcome this limitation by incorporating additional information of the process, namely linear balance equations. Ideally, measurements have to produce residuals of the balance ($\mathbf{r}$) that are randomly distributed with an expected value of zero.

$$\mathbf{r}=\mathbf{A}\mathbf{y}=\overbrace{\mathbf{A}\mathbf{y}^*}^{0}+\mathbf{A}\varepsilon=\mathbf{A}\varepsilon \tag{1.21}$$

The covariance matrix of the residuals is given by Eq. (1.22) and can be used to estimate matrix $\mathbf{Q}$.

$$\Sigma=cov(\mathbf{r})=E[\mathbf{r}\mathbf{r}^T]=E[\mathbf{A}\varepsilon\varepsilon^T\mathbf{A}^T]=\mathbf{A}E[\varepsilon\varepsilon^T]\mathbf{A}^T=\mathbf{A}\mathbf{Q}\mathbf{A}^T \tag{1.22}$$

Using additional process information in the *indirect* methods is limited to linear constraints, since Eq. (1.22) may present high computational effort if nonlinear constraints are considered.
The presence of gross errors render the *direct* as well as *indirect* methods inefficient, then a more robust method should be considered (see section 1.2.6.3).

Despite the effort done in the calculation of variance-covariance matrix the previous methods are often oriented to steady-state process and little has been done for dynamic processes. In principle the authors assume that the process variable variability is constant and known along the time. This assumption may be seriously affected in a real industrial environment where the variance is continuously undergoing changes. The difficulty of calculating this matrix is associated with the analysis of the different causes affecting the process variables variability (i.e. process dynamics, measurement error, etc).

1.2.4 Nonlinear Data Reconciliation

Commonly, plant processes are represented by mass and energy balances and may also include thermodynamic and physical property correlation and equilibrium relationships. Thus, the process plant is modeled by means of a nonlinear system of equations and nonlinear Data Reconciliation techniques must be used.

These nonlinearities present several difficulties (e.g. solution of the corresponding nonlinear optimization problem) which are not encountered in the linear case.

$$\min_{\widehat{\mathbf{y}}(t)} \Phi[\mathbf{y}(t), \widehat{\mathbf{y}}(t); \sigma] \tag{1.23}$$

subject to:

$$\mathbf{h}\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) = \mathbf{0} \tag{1.24}$$

$$\mathbf{g}\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) < \mathbf{0} \tag{1.25}$$

where $\mathbf{h}()$ set of algebraic equation and $\mathbf{g}()$ set of inequality equations (imposing bounds on the variables, commonly used to limit the effect of gross errors to be spread over all estimates or to impose operation and logical limits).

The nonlinear Data Reconciliation problem is similar to the Error-in-all-Variables Method (EVM) estimation as reported by Kim et al. (1990, 1991). EVM is applied to regress model parameters when there are measurement errors in all process variables. Then, Data Reconciliation is a special case of the EVM estimation where no parameters are to be estimated but the optimization is subject to constraints.

To deal with this problem, two rigorous approaches stand out (1) successive linearization proposed by Knepper and Gorman (1980) and (2) direct use of a nonlinear programming method to solve th NLP problem (Liebman et al. (1992)).

1.2.4.1 Successive linearization methods

The successive linearization relies on the idea that the nonlinear constraints can be linearized as a first-order Taylors series expansion around the current estimate. The analytical solution to the linearly constrained Data Reconciliation problem is then applied.

This approach has the advantage of being relatively simple and fast. However, Liebman and Edgard (1988) demonstrate that the Data Reconciliation using nonlinear programming provides better estimation accuracy than the successive linearization approach.

1.2.4.2 NLP methods

Various methods have been proposed to solve the nonlinear optimization problem (NLP). These include gradient-based methods such as Gauss Newton, Gauss Marquardt or Generalized Reduced Gradient. Nevertheless, the Successive Quadratic Programming (SQP) is in general the most accurate since the objective function in DR is generally a quadratic function.

The NLP methods allow for a general nonlinear objective function not just a WLS and can explicitly handle nonlinear, inequality constrains and variable bounds. The disadvantage of these NLP algorithms is the large amount of computation time required compared to the linearization approach.

1.2.5 Dynamic Data Reconciliation

All the works addressed in previous sections are oriented to steady-state systems described by linear, bilinear and nonlinear constraints, involving in some cases model parameters estimation.

In many practical situations the chemical processes are intrinsically dynamic, although in some significant case they are operated to maintain a nominal steady-state conditions. Even in such cases they are continuously undergoing variations around these desired conditions. Therefore, the model representing the process is a dynamic model and the resulting problem is called Dynamic Data Reconciliation (DDR).

The most widely adopted approaches for solving DDR problems are: (1) stochastic filtering (such as Kalman filtering) and (2) mathematical programming.

1.2.5.1 DDR based on Kalman filtering

Kalman filtering (KF) is a state-space based filtering technique used for the reconciliation of linear dynamic processes as reported by Gelb (1974); Muske and Edgar (1998). KF has been extended to deal with nonlinear systems by linearizing the nonlinear part of the model with a first-order Taylor expansion around the current estimate leading to an Extended Kalman Filter (EKF) (Karjala and Himmelblau (1996); Islam et al. (1994); Chiari et al. (1997)). Nevertheless, Kalman filtering presents a serious drawback due to its limitation to handle inequality constraints such as bounds on process variables or process operation.

Since the Kalman filtering will be encountered many times in this thesis, it is worthwhile to explain the concepts of this estimator. The Kalman Filter is a recursive technique for estimating state variables and their associated error variances. The algorithm uses the following discrete state-space dynamic model:

$$\mathbf{x}_k = \mathbf{A}_k \mathbf{x}_{k-1} + \mathbf{B}_k \mathbf{u}_{k-1} + \mathbf{w}_{k-1} \tag{1.26}$$

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \tag{1.27}$$

where k represents a sample time $t = kT$, being T the sampling period; $\mathbf{x}_k$ is the n_x dimensional vector of state variables; $\mathbf{u}_k$ is the n_u dimensional vector of manipulated input variables (if no control inputs are considered, then $\mathbf{u}_k = 0, \forall k$) and $\mathbf{y}_k$ is the n_y dimensional vector of measured variables.

The state transition matrix $\mathbf{A}_k$, the control gain matrix $\mathbf{B}_k$ and the observation matrix $\mathbf{H}_k$ are matrices of appropriate dimension. If the coefficients of $\mathbf{A}_k$, $\mathbf{B}_k$ and $\mathbf{H}_k$ are assumed time-independent the subscript k can be dropped. This is possible if the model structure and the control strategy are maintained through the time. The Kalman filter assumes error in the process model and in measured data. The process-noise $\mathbf{w}_k$, represents errors in the state transition model. This noise has zero means, and has a variance of $\mathbf{Q}_k$. A random error $\mathbf{v}_k$ representing the measurements noise, which has zero means and a variance $\mathbf{R}_k$, is also added to the measurement model. If the model noise error and measurement errors are constant throughout time, k can also be dropped in $\mathbf{Q}_k$ and $\mathbf{R}_k$.

So there are two ways to estimate $\mathbf{x}_k$: from the model and $\mathbf{x}_{k-1}$ through Eq. (1.26) and from the measurement $\mathbf{y}_k$ through Eq. (1.27). The Kalman filtering balance both ways taking into account the errors (variance) associated to each one in order to asymptotically reduce the error (variance) in the variable estimation along the time.

Assuming an initial estimate of the state variables $\widehat{\mathbf{x}}_0$ and the associated error covariance matrix $\mathbf{P}_0$. These initials value do not affect the final estimation of $\mathbf{x}_{k/k}$ and its corresponding variance $\mathbf{P}_{k/k}$.

$$\widehat{\mathbf{x}}_0 = E\left[\mathbf{x}_0\right] \tag{1.28}$$

$$\mathbf{P}_0 = cov\left[\widehat{\mathbf{x}}_0\right] \tag{1.29}$$

the state variables vector and the associated error covariance matrix $\mathbf{P}$ for the next step in time is given by:

$$\widehat{\mathbf{x}}_{k/k'} = \mathbf{A}_k \widehat{\mathbf{x}}_{k'/k'} \tag{1.30}$$

$$\mathbf{P}_{k/k'} = \mathbf{A}\mathbf{P}_{k'/k'}\mathbf{A}^T + \mathbf{Q}_k \tag{1.31}$$

Using process measurements, these predictions are updated as follows:

$$\widehat{\mathbf{x}}_{k/k} = \widehat{\mathbf{x}}_{k/k'} + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{H}_k\widehat{\mathbf{x}}_{k/k'}\right) \tag{1.32}$$

$$\mathbf{P}_{k/k} = \mathbf{P}_{k/k'} - \mathbf{K}_k\mathbf{H}_k\mathbf{P}_{k/k'} \tag{1.33}$$

where $\mathbf{K}_k$ is the Kalman filter gain given by:

$$\mathbf{K}_k = \mathbf{P}_{k/k'}\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_{k/k'}\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1} \tag{1.34}$$

The estimator gain matrix $\mathbf{K}_k$ take into account the observation matrix $\mathbf{H}_k$, is proportional to the uncertainty in the estimation $\mathbf{P}$ and inversely proportional to the uncertainty in the measurement $\mathbf{R}_k$. If the measurement is very uncertain and the state estimate is relatively precise, then the residual is mainly composed by the noise and

a small change in the state estimates should be made. Whereas, if the uncertainty in the measurement is small and that of the state estimate is high, then the residual indicates considerable errors in the state estimate exists and correspondingly a significant correction should be made.

1.2.5.2 DDR based on mathematical programming

Using a mathematical programming approach, DDR can be formulated as a dynamic optimization problem where the objective is to minimize the deviation between the measured $\mathbf{y}$ and the estimated $\widehat{\mathbf{y}}$ values, weighted by the variance of measurement errors subject to the dynamic model (Eq. (1.36)) and/or nonlinear algebraic model (Eq. (1.37)) and/or inequality constraints (Eq.(1.38)) as follows:

$$\min_{\widehat{\mathbf{y}}(t)} \Phi[\mathbf{y}(t), \widehat{\mathbf{y}}(t); \sigma] \tag{1.35}$$

subject to:

$$\frac{d\widehat{\mathbf{y}}(t)}{dt} = \mathbf{f}\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) = \mathbf{0} \tag{1.36}$$

$$\mathbf{h}\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) = \mathbf{0} \tag{1.37}$$

$$\mathbf{g}\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) < \mathbf{0} \tag{1.38}$$

Despite that DDR can provide accurate estimates to redundant variables as well as to unmeasured observable variables, it is faced with two main difficulties: (1) handling the differential equations and (2) evaluating dynamic evolving process variables variance.

Handling differential equations

1 - The differential equations are approximated by a set of algebraic equations using a weighted residual method. Table 1.1 summarizes some of the more relevant discretization methods adopted within the Data Reconciliation techniques.

Table 1.1: Relevant ODEs discretization methods used by DDR

Orthogonal Collocation	Liebman et al. (1992)
First-Order Euler approximation	Rollins and Devanathan (1993)
Implicit Runge-Kutta	Albuquerque and Biegler (1996)

Using this discretization the differential equations are treated in the same manner as the other constraints The model equations are then solved simultaneously within an infeasible path optimization algorithm.

The performance of these discretization-based methods relies on selecting as small as possible the integration time. In practice the sampling time of getting measurements data is relatively higher than the integration time, which limits the use of these discretization techniques for DDR purposes. Even if it can be assumed that the sampling time is equal to the integration time the above mentioned techniques require that all process measurements are obtained at a regular discrete time; otherwise redundancy is lost.

Additionally, this discretization increases the number of variables and equations in the optimization problem. Therefore, the resolution of the resulting DDR problem involves a large computation effort, which especially complicates the on-line application of these approaches. In practice despite the loose of estimation accuracy the adoption of a moving window horizon strategy that catches only a part of system's dynamic is motivating for reducing the computational effort.

2 - Bagajewicz and Jiang (1997) went on this topic by approximating all the process variables involved in the ODEs system through a polynomial in time with a predefined specific degree; the polynomial coefficients substitute the original variables in the off-line Data Reconciliation formulation. The resulting reconciled coefficients are used to recover the original variables. However, the estimator provides less accuracy since some not reconciled coefficients are used to reconstruct the "reconciled" original variables. Furthermore, this proposal assumes that all the process variables are represented by a polynomial in time with a degree p, and all the polynomials representing the derivative terms have a degree $p+1$ independently of the dynamic behavior of each process variable. Thus, their proposal represents a particular situation where all the variables are represented by the same degree. Additionally, their proposal presents the drawback that a degree higher than the required for a specified variable is assigned, which will undoubtedly increase the number of inflection points, thus involving worse estimations.

Dynamic variance-covariance calculation

In general, variance-covariance matrix is a pre-requisite for performing Data Reconciliation since the objective function in DDR formulation is weighted by its inverse as its shown in Eq. (1.39).

For most applications, the objective function (Eq. (1.39)) is simply a weighted least-squares as follows:

$$\Phi[\mathbf{y}(t),\widehat{\mathbf{y}}(t);\sigma] = \sum_{k=0}^{c} \left[\widehat{\mathbf{y}}(t_k) - \mathbf{y}(t_k)\right]^T \mathbf{Q}^{-1} \left[\widehat{\mathbf{y}}(t_k) - \mathbf{y}(t_k)\right] \tag{1.39}$$

where $\widehat{\mathbf{y}}(t_k)$ represents the values of the estimate variables at discrete time t_k and $\mathbf{Q}$ is the variance-covariance matrix with $\mathbf{Q}_{ii} = \sigma_i^2$. Variables t_0 and t_c represent the initial and current times, respectively.

Evaluating variance-covariance matrix for dynamic systems is relatively complicated. This complication is due to the combined variability of process and measurement errors. In general, it is assumed constant. However, this matrix is not known or known only approximately since measurement errors change frequently. When the measurement error variance is related to the magnitude of the measurement or the statistical distribution having different variances, which is the case of dynamic systems, a heteroscedastic or recursive estimation might be appropriate.

1.2.5.3 DDR regularization

The combination of the DDR problem formulation and its associated numerical solution has recently been focused as the new challenge in the DDR problem (Binder et al. (2002)).

Although several approaches have been proposed for the problem of DDR formulation, the numerical strategy adopted for solving the DDR does not present good performance in terms of stability and converged results (regularization). Kelly (1998) addressed the regularization in the context of steady-state Data Reconciliation.

1.2.6 Gross Error Detection

The validity of the results of the estimation algorithm rests on the assumption that the inconsistency is caused by measurement noise and assumption uncertainty. The presence of undetected gross errors invalidates this assumption and introduces bias into the resulting estimates. Typical gross errors include miscalibrated sensors, failed sensors, process leaks.

1.2.6.1 Statistical tests approaches

Gross errors significantly affect the accuracy of any industrial application using process data. The problem of identifying gross error in data and its importance in Data Reconciliation was pointed early by Ripps (1965). As in Data Reconciliation, Gross Error Detection (GED) can be performed only if process model is present, the availability of process model acts as a counter check of the measurements. There are three steps in gross error handling:

- detection of gross errors, have been adopted
- identification of the sources of those errors (measurements/processes), and
- compensation/elimination of those errors.

If a gross error exists in a measured value, the measurement equation Eq. (1.3) changes to Eq. (1.40).

$$\mathbf{y} = \mathbf{y}^* + \varepsilon + \delta \tag{1.40}$$

where δ is the vector of gross error.

The constraint residual $\mathbf{r}$, Eq. (1.21), is required for GED. If no gross errors are present the expected value of this residual $E[\mathbf{r}]$ is equal to zero, otherwise a gross error is presents. Therefore, the hypothesis of GED can be formulated as:

$$\begin{cases} H_0 : E[\mathbf{r}] = \mathbf{0} \\ H_1 : E[\mathbf{r}] \neq \mathbf{0} \end{cases} \tag{1.41}$$

where H_0 is the null hypothesis that no gross error is presents and H_1 is the alternative hypothesis that either a process leak or a measurement error is presents. H_1 is accepted if the statistic test calculated exceeds a pre-specified threshold value. This threshold is associated to a significance "degree of confidence". These statistical tests are based on the analysis of the constraint residual distribution, Eq. (1.42).

$$\gamma = \mathbf{r}^T \Sigma^{-1} \mathbf{r} \tag{1.42}$$

The Global Test (GT) presented in Reilly and Carpani (1963) is based on the fact that the value of the objective function of DR expressed in Eq. (1.1) at the minimum is distributed as a chi-square variable χ^2 statistic if the measurements are normally distributed about their true values. The magnitude of the DR objective function is then compared to the tabulated chi-square value for a chosen confidence level (e.g. 95%) and for the degrees of freedom equal to the number of independent equations (rank of matrix $\mathbf{A}$).

The vector $\mathbf{r}$ can also be used to derive test statistics one for each constraint j given by:

$$\mathbf{z}_{\mathbf{r},j} = \frac{\left|\mathbf{r}_j\right|}{\sqrt{\Sigma_{jj}}} \tag{1.43}$$

The Nodal Test (NT) presented by Mah et al. (1976) evaluates if $\mathbf{z}_{\mathbf{r},j}$ follows a standard normal distribution under H_0 and decides which equations might contain a gross error. These methods require that the constraints are linear and that all process variables must be measured. The unmeasured variables must be removed from constraints by some classification techniques see section 1.2.2.

The statistical test can also be based on the adjustment distribution by which first process data are reconciled. Then the reconciled data are used to examine if a measurement contains a gross error. These tests allow unmeasured variables in the plant model as presented in Mah and Tamhane (1982). The authors termed this test

Measurement Test (MT). The MT looks at the adjustment of each measurement to identify and rank measurement that may be faulty.

While the NT avoids the computational cost of the reconciliation step, it does require further processing to identify the actual measurement error. The drawback of the MT is that it assumes that the erroneous measurement will have the largest normalized adjustment. However, the measurement adjustment often *smears* the impact of gross errors across several measurements or to one particular measurement.

If the covariance matrix of constraint residuals or measurement adjustments is not diagonal, the assumption that measurement errors are independent of each other is not satisfied, and this affects the power of the statistical tests. The methods of Maximum Power (MP) (Mah and Tamhane (1982)), Principal Component Analysis (PCA) (Tong and Crowe (1995)) and Generalized Likelihood Ratio (GLR) (Narasimhan and Mah (1987)) were developed to overcome this weakness.

In general, the current statistical tests for GED are applied only to linear systems. To address nonlinear systems a suitable linearization of the process constrains has to be done. However, by forcing measurements to conform to the linearized (approximated) model new errors may be introduced.

1.2.6.2 Multiple GED

The tests described in the last section are suitable for the detection/identification of one gross error at a time. When more gross errors exist, multiple GED strategies are needed.

These methods are based on combining some of the statistical tests presented above with a serial or collective elimination or compensation strategy. In the serial elimination strategy presented in Ripps (1965), measurement suspected of containing a gross error is selected on a trial basis. Then, the statistical test is computed and if its value is below the critical value, then the suspected value is declared in gross error and thus eliminated. The measurement elimination is done one at a time recomputing a new statistical test based on the reduced measurement set.

Serial elimination has the drawback of loosing redundancy (since measured variable with gross error are eliminated) and is not applicable to gross error that are not directly associated with measurements (for instance leaks). However, this strategy has now become a standard in multiple GED and is the approach implemented in most commercial software. In the serial compensation strategy illustrated in Serth and Heenan (1986) an estimate of the gross error size is performed using error size estimation "formula" developed by Madron (1985). In this strategy, gross errors are estimated and measurements are compensated one by one, rather than eliminated, thus the redundancy is kept during the procedure. This strategy is applicable to all gross error type (measurement-related error or process-related error). To improve

these methods, Rollins and Davis (1992) propose the estimation of all gross errors simultaneously using UnBiased Estimation Technique (UBET).

Since statistical tests are probabilistic based approaches, two type of errors in GED procedure may potentially rise:

- Error *Type I* : gross error is repaired when actually not present (false alarm).
- Error *Type II* : gross error is not repaired when actually is present (resilience).

Regardless of the power of the statistical tests, there is no guarantee that they can always detect gross errors. Rosenberg et al. (1987) proposes two measures represented by Eqs. (1.44) and (1.45) that may be used to evaluate the performance of GED scheme:

1. The Overall Power (OP) defines the ratio of number of gross errors that are correctly detected to the total number of gross errors in measurements as follows:
$$OP = \frac{number\ of\ gross\ errors\ correctly\ identified}{number\ of\ gross\ errors\ simulated} \tag{1.44}$$
2. The Average Number of Type *I* (AVTI) error, which defines the number of misidentification made by the method, is represented by Eq. (1.45). This measures gives the average numbers of gross error misidentify per application of the method.
$$AVTI = \frac{number\ of\ gross\ errors\ wrongly\ identified}{number\ of\ simulation\ trial\ made} \tag{1.45}$$

1.2.6.3 Simultaneous Data Reconciliation and GED

A promising alternative is to combine data reconciliation and gross error detection. Two main approaches can be adopted for such purpose: the Bayesian Approach and the Robust Approach:

1 - The Bayesian Approach (Tjoa and Biegler (1991)) takes into account the presence and distribution of gross errors in the error distribution function. Thus, the objective function (Eq. (1.46)) contains two terms: one representing the random errors and the second the gross errors, each one multiplied by its probability to occur as follows:
$$P(\varepsilon_i) = (1-\eta_i)\,N\left(0,\sigma_i^2\right) + \eta_i N\left(0, b_i^2.\sigma_i^2\right) \tag{1.46}$$
where η_i is the known probability of the occurrence of gross error in sensor i and $b_i(\gg 1)$ is the ratio of the standard deviation of the gross error distribution of sensor i to the standard deviation of the random-error distribution.

However, the gross error distribution is usually a posteriori information which, may lead to a biased estimation. If the gross error term is higher than the random term for a particular measurement a gross error is identified. In consequence, this approach can only be used if the gross error distribution is known a priori.

2 - The Robust Approach (Huber (1981); Romagnoli and Sánchez (2000)) attempts to make the estimation insensitive in front of the presence of gross errors. The weighted squared residual of the DR formulation is replaced by another function of the residual as shown in Eq. (1.47).

$$\min \sum \rho \left(\mathbf{r}_j \right) \tag{1.47}$$

where $\rho()$ is usually selected as a convex function in order to ensure that the solution is unique, the influence function is the derivative of $\rho()$ with respect to the process variable measurements. This Robust Estimator will give an unbiased estimate when the gross error follows a previously known distribution and still behaves well if they are deviations from ideal situation. Thus, this influence function compensates for the effects that have the residuals on the estimations, given a weight of zero to high value residuals. The crucial step in the Robust Estimation is the choice of these influence functions: different pre-selected choices of the influence function deals to estimations with different robustness.

1.2.6.4 Multivariate Statistical Process Control

The use of statistical tests is the most widely and efficiently procedure for GED but it presents serious limitations when it is difficult to obtain complete and accurate process models or when the process presents high dimensionality and complexity.

In such situations other techniques such as Multivariate Statistical Process Control (MSPC) can be used. The MSPC aim to reduce the number of process variables required to describe significant process variations. The recorded data are thereby compressed into an alternative set containing fewer variables that are more manageable and interpretable. One such MSPC is the Principal Component Analysis (PCA) or the Partial Least Square (PLS).

In the case of nonlinear process an Auto-Associative Neural Network (AANN) could acts as a nonlinear PCA. Nonlinear methods for fault detection are usually based on identifying clusters for normal and faulty operation by methods such as k-means clustering. Although MSPC techniques are very powerful for fault detection their main limitation lies in their ability to identify the faults. Yoon and MacGregor (2000) present the fundamental differences between statistical and causal model based approaches to fault detection and isolation via several simulation studies.

1.2.7 Sensor placement

The principal objective of data reconciliation and gross error detection is to improve the accuracy and consistency of process variables estimated. These techniques certainly reduce the error content in measurements if redundancy exists in the measurements. The improvement that can be achieved depends crucially on: the accuracy of the sensors which is specified by the variance in the measurement errors and number of variables and their type which are measured.

Different sensors may be available for measuring a variable with widely varying capabilities such as the range over which it can measure reliability, and accuracy. The cost of the sensor will be a function of its capabilities. This information must typically be obtained from instrumentation manufacturers or suppliers.

Considering all the different variables such as flow rates, temperature, pressures and compositions of the streams in a process, these could be of the order of several thousands in number. Clearly, from the viewpoint of cost, complexity or technical feasibility it is not possible to measure each and every variable. Only a subset of these variables is usually measured.

The design of sensor networks should provides, the decision regarding which variable should be measured by which instrument and according to which criteria. Although this problem is an important one in design of new plants, it can be used to retrofit the measurement structure of existing plants by identifying new variables that need to be measured for improving monitoring and control of the process for instance.

The problem of sensor networks design for processes has been addressed by several research groups covering different criteria:

- **Observability** = ability to estimate variables using measurement.
- **Estimation accuracy** = ability to obtain accurate estimation through data reconciliation.
- **Reliability** = ability to estimate variables under sensor failures.

Vaclaveck and Loucka (1976) were the first to develop a sensor placement strategy for the steady-state system so as to ensure the observability of a specified set of important variables in a pure mass-flow or multi-component process by using graph-theory. To obtain number of measuring devices is straightforward once known the number of process variables and the number of equations that relates them, from here the sensor placement problem is addressed. Ragot et al. (1992) gave a procedure, which ensured observability of all variables in a bilinear process.

Kretsovalis and Mah (1987) quantified the effect of sensor placement on the accuracy of estimated variables for mass-flow processes, and use the results to develop

a combinatorial search algorithm for sensor network design. Madron and Veverka (1992) extend the work presented in Vaclaveck and Loucka (1976) including the overall cost of sensors. Their method makes use of Gauss-Jordan elimination to identify a minimum set of variables that need to be measured in order to observe all required process variables while simultaneously minimizing the overall cost of sensors. Along this line, Meyer et al. (1994) also developed an algorithm for minimum cost design of sensors for linear processes based on a graph oriented approach. They used a branch-and-bound type strategy to solve the optimization problem formulated.

Ali and Narasimahan (1993) addressed the issue of sensor failure and its effect on observability variables, and took it into account in sensor placement strategies. They went on to tackle the problem of sensor placement strategy for steady-state linear processes when sensors are likely to fail. They also proposed the concept of reliability of estimation of a variable, which gives the probability of being able of estimating a variable value for any given sensor network and specified sensor failure probability. Ali and Narasimhan (1995) extend their work for the optimal design of redundant sensor networks for linear processes.

Bagajewicz (1997) posed the sensor network problem as an optimization problem with minimization of cost as the objective function and requirements of error detectability, resilience and residual precision as the constraints of the optimization problem. Sen et al. (1998) integrated genetic algorithms with graph-theory concepts to solve the problem of optimal design of a sensor network for linear processes. Using genetic algorithms they could solve the problem to optimize different objectives simultaneously such as cost, estimation accuracy and systems reliability. Additionally, the encoding procedure they propose is quite intuitive, that is, in the string the bit represents the characteristics of the solution.

Later, Bagajewicz and Sánchez (1999) merged the concept of degree of redundancy and degree of observability for variable measurements into a single concept: degree of estimability of a variable. They presented a formulation for the design of a sensor network to achieve a required degree of estimability. A minimum overall cost model and a generalized model for the design of a reliable sensor network has been demonstrated to be equivalent as shown in Bagajewicz and Sánchez (2000b). Reallocation and upgrading the existing instruments to achieve maximum precision of selected variables has been dealt with by Benqlilou et al. (2001b); Bagajewicz and Sánchez (2000b).

The methods just cited above do not address the issue of sensor placement for dynamic systems. Chmielewski et al. (2002) extended the static sensor placement problem to linear dynamic processes. They gave a procedure to make the NLP problem independent of the decision variables. Additionally in their contribution they transform the NLP problem into a convex program through the linear matrix inequal-

ity.

Furthermore, a software tool for sensor placement decision-making is still missing, although that some authors have attempted to develop a tool for sensor placement design (Sánchez et al. (1992); Jaroslav and Perris (1992)).

Recently, Heyen et al. (2002) propose a general mathematical formulation of the sensor selection and location problem, in order to reduce the cost of the measurement system while providing estimates of all specified key process parameters within a prescribed accuracy. Narasimhan and Rengaswamy (2003) in a recent work also recognize the need for developing an integrated approach to sensor network design.

1.2.8 The CAPE-OPEN standards

Industrial users of CAPE tools are no longer developing their own proprietary software but the present trends is to use modular and customized programs provided and maintained by specialized vendors. In a similar way, the software engineering has undergone a significant transformation during the last decade.

While the hardware standard has moved from large supercomputers to networked PCs, software developers have given up huge monolithic programs to fully exploit the benefits of distributed computing, such as modularity, maintainability and code re-usability.

These converging changes motivated the CAPE-OPEN and Global CAPE-OPEN (GCO) EU-funded projects, which have provided the mechanisms for ensuring CAPE modules inter-operability across the network. Thus, CAPE-OPEN standards for communication interfaces were produced and published. Hence the term CO-compliant was coined to designate the software modules that correctly implement the CO interfaces, allowing them to interact with other CO-compliant software. Such compliance and the current status of the standard were recently reviewed by Belaud and Pons (2002).

The CO and GCO legacy includes a state-of-the-art monograph on software architectures and tools for CAPE (Braunschweig and Gani (2002)) and the CAPE-OPEN Laboratories Network (CO-LaN) the internationally recognized user-driven organization for the testing and management of the CAPE-OPEN standard (CAPE-OPEN Standard 1.1 (2002)).

The development of the CO standards was carried out on the basis of a technical decision made for the formal description of the interfaces Belaud et al. (2002). The Unified Modeling Language ^{TM} (UML^{TM}) already developed, published and maintained by the Object Management Group ^{TM} (OMG (2003)) was adopted as a convenient tool for such purpose since it allows specifying, visualizing, and documenting models of software systems. This work also uses UML's standard diagram types for defining the architecture and interfaces proposed.

Among the different CO standard interfaces, it is worth noting those related to the architecture presented in this thesis, which allowed incorporating some already developed modules such as mathematical models (CO-ESO, Equation Set Object) and solvers (CO-MINLP). Particularly, the architecture requirements for Parameter Estimation and Data Reconciliation system (PEDR) were addressed by the GCO project and summarized in Arora et al. (2002) and the prototype implementation was also presented in Benqlilou et al. (2002e). However, the sensor placement problem and the architecture of a standardized sensor placement tool, despite being strongly connected to the previous aspects, were not addressed by the GCO project. This thesis is also intended to cover this gap.

1.3 Objectives and thesis contributions

From the concepts and results described in the previous sections, and due to the requirements and potential applications of Sensor Placement, Data Reconciliation, and Gross Error Detection systems and their combinations this thesis is focused in the evaluation and improvement of the afro-mentioned techniques having as main goals:

- Investigating novel approaches for dynamic Data Reconciliation based on moving average filter and SSDR.
- Developing a system for combining wavelets analysis and Dynamic Data Reconciliation.
- Combining Time-Delay Estimation and Dynamic Data Reconciliation.
- Proposing a distributed open architecture for Data Reconciliation able to exploit the new trends of Data Reconciliation techniques.
- Investigating the relationship between Data Reconciliation and sensor placement in the case of dynamic processes.
- Proposing a more precise way to determine the sensor network reliability.
- Incorporating inferential sensors application within the framework of Data Reconciliation and comparing their performance with such of physical sensors.
- Proposing generic design/retrofit framework for placing and selecting sensors for steady-state and dynamic processes.

The most significant contributions of this thesis are highlighted as follows:

In Part I of this thesis the combination of filtering and reconciliation techniques are addressed. A proposal is presented in **Chapter** 2 to combine a moving average filter and a steady-state Data Reconciliation for dynamic cases. The estimator proposed is unbiased and provides competitive results compared with Kalman filter in terms of estimation accuracy and tracking of dramatic dynamic changes of process variables (e.g. step in a set point) as well as variance reduction. This proposal breaks down the traditional view that steady-state Data Reconciliation is used when considering steady-state models and dynamic Data Reconciliation is used when the dynamic model is considered. The advantage of the proposed approach becomes more relevant for on-line industrial application, since this approach requires less computational effort.

Along this line a rigorous Dynamic Data Reconciliation technique is provided in **Chapter** 3. A wavelet analysis technique is used to catch the deterministic features of the process variables. These trends are then represented by a polynomial in time. The coefficients of the resulting polynomial are made consistent with the dynamic process model. An important feature of the proposal is its capability to handle different measuring sampling times and to reduce the problem of estimating the variance of dynamic variables. A promising alternative for estimating state variable of a nonlinear process is also presented in **Chapter** 3. This method is efficient and competes with the current approaches in terms of accuracy and required computational effort.

A frequent problem that operating plants are faced with is the presence of delay between the sampled process variables. Therefore a mismatch between the process model and process measurements is raised. In this thesis, concretely, in **Chapter** 4 a simple but efficient approach is proposed to deal with the presence of such delay. The approach is based on maximizing the correlation between the correlated variables, thus, a time-delay vector is obtained. This vector is used to update the measurement model, which is included in the DR. The resulting approach shows acceptable results and promising application of the proposal in on-line applications.

The use of a simulator-based model within the Data Reconciliation problem is also addressed in this thesis in **Chapter** 5. By using a simulator-based model the construction and the resolution of the process model is performed within the simulator and the optimization problem size is reduced since the simulator does not accept an over-specified system. The synergy between the simulator and the optimizer is the basis for proposing a distributed modular and standard compliant Data Reconciliation application. This approach provides a framework for successful achievement of a simultaneous estimation of the measurements, unmeasured variables and model parameters.

In part II of this thesis the accurate and reliable design/retrofit of sensor networks is addressed. For reasons of cost and technical feasibility not all variables are measured in a given process plant. During the sensor network design phase it is not only important to determine the number of sensors that satisfy some criteria (e.g. accuracy, reliability, etc) but also their types and position in the plant. In **Chapter** 6 a novel methodology is proposed to obtain the network with the minimum number of sensors that ensures the required system accuracy in a dynamic process. In this line the synergy between sensor placement and a dynamic Data Reconciliation technique is investigated. This approach deals with the observability procedure by manipulating the measurements noise matrix in the state-space formulation. The above results present the strong relationship between Data Reconciliation and sensor placement. However, it is also important to appreciate that not all the Data Reconciliation techniques give the same results. In this sense another point addressed in this thesis is the comparison of the different dynamic Data Reconciliation techniques.

Within these contributions an efficient technique is presented in **Chapter** 7 for evaluating the reliability of process variables. With the objective of increasing the system reliability an approach combining quantitative modeling process and fault tree analysis is proposed and validated. The resulting sensor optimization problem is solved by means of genetic algorithms showing good results. With the objective of increasing the system redundancy in order to improve the data quality by means of Data Reconciliation some instrumentation network design has to be realized. The use of inferential sensor as an alternative within Data Reconciliation in a real industrial case is presented demonstrating good performance. A comparison with the other approaches that utilize physical sensors show that inferential sensors present acceptable results.

The decision to update the sensor network in order to reach the new plant performance goals in the case of operating plants is quite difficult. This decision leads to two possibilities: the purchase of new instrumentation or the reallocation of the existing measurement devices. A generic sensor placement formulation extending the current sensor placement paradigm to address the design and retrofit problem as well as to consider dynamic processes by introducing a general framework following the guidelines presented by Benqlilou et al. (2003). Additionally, the framework contemplates hardware redundancy and measuring device selection. Therefore, a tool is provided following the CAPE-OPEN standard and exploiting the current and future trends of sensor placement (**Chapter** 8). The general framework proposed allows addressing these problems from a general decision-making point of view by integrating tools to deal with the necessary cost-performance trade-off as has been recently proposed by Bagajewicz and Cabrera (2000).

Nomenclature

$\mathbf{A}$	incidence matrix with rows corresponding to nodes and columns to streams
$\mathbf{A}_k$	state transition matrix at sampling time k
$\mathbf{a}$	measurement adjustment vector
$\mathbf{B}_k$	control gain matrix at sampling time k
$\mathbf{f}()$	differential equation constraints either linear and/or nonlinear
$\mathbf{g}()$	inequality constraints including simple upper and lower bounds
$\mathbf{h}()$	algebraic equation constraints either linear and/or nonlinear
H_0	null hypothesis
H_1	alternative hypothesis
$\mathbf{H}_k$	observation matrix at sampling time k
k/k'	estimating at sample time $k.T$ using the past $k'.T$ samples
$\mathbf{k}_k$	Kalman filter gain matrix at sampling time k
L	Lagrange function
n	width window
$\mathbf{P}_0$	covariance matrix of initial estimates of state variable $\widehat{\mathbf{x}}_0$
$\mathbf{Q}$	variance-covariance matrix of measurement errors
$\mathbf{Q}_k$	covariance matrix of random variable $\mathbf{w}_k$
$\mathbf{R}_k$	covariance matrix of random variable $\mathbf{v}_k$
$\mathbf{r}$	vector of constraint residuals
$\mathbf{r}_j$	constraint residual at unit j
t	time
T	sampling period
$\mathbf{u}$	vector of input variables
$\mathbf{u}_k$	vector of control input variable at sampling time k
$\mathbf{v}_k$	vector of measurement noise
$\mathbf{w}_k$	vector of process noise model
$\widehat{\mathbf{x}}_k$	vector of state variable at sampling time k
$\widehat{\mathbf{y}}_k$	vector of measured variable at sampling time k
$\mathbf{y}$	vector of process variable measurement
$\overline{y}_i$	arithmetic average of i at a certain number of observations
$\widehat{\mathbf{y}}$	vector of reconciled (estimated) process variable
$\mathbf{y}^*$	vector of true value of process variable

Operators

$E[]$	expected value operator
$cov()$	variance-covariance matrix calculation
$N(\mathbf{0}, \mathbf{Q})$	normal distribution with mean $\mathbf{0}$ and covariance matrix $\mathbf{Q}$

P probability density function

Greek letters

γ distribution of residual (**r**)
δ vector of gross errors
ε vector of random measurement errors
η hypothesized gross error probability occurrence
θ vector of model parameters
λ Lagrange multiplier
$\rho()$ convex function whose derivative is the influence function
σ_i standard deviation of the ε_i
Σ covariance matrix of constraint residuals, **r**
$\Phi()$ objective function of the formulated Data Reconciliation problem
χ^2 chi-square distribution

Subscripts

i process variable
j process unit
k sampling time

Part I

Dynamic Data Reconciliation

Chapter 2

Data Reconciliation and dynamic systems

Abstract

This chapter presents a novel technique for reconciling data of linear dynamic systems. The technique is based on adapting SSDR techniques previously discussed, in section 1.2.1 of Chapter 1 of this thesis.

An estimator is constructed using the reconciled averages (or means) of the process variables measurements. Using the averages, the process dynamics are absorbed and then a SSDR can be carried out to reconcile these averages. Finally, the "reconciled" process variables are recovered from the reconciled averages. An interesting feature of this technique is its low computational effort which renders it a potential alternative for on-line reconciliation applications. Additionally, the proposed estimator presents the important statistical feature of being unbiased. The proposal is competitive with the Kalman filter technique in terms of variance reduction. Furthermore, it is able to deal with steady, pseudo-steady and dynamic linear processes without any important changes.

Thus, the main aim of this work on dynamic Data Reconciliation is to minimize the effort of reconciling dynamic measurements from chemical processes while making the estimations more accurate, precise and faster.

2.1 Introduction

Data Reconciliation is a model-based filtering technique. If the considered process is approximated by a steady-state model, the technique applied should be the SSDR; if the process is represented by a dynamic model DDR techniques are used. Given

that steady-state operation is almost never fulfilled it is better to think of applying DDR even for "steady-state" processes (Narasimhan and Jordache (2000)). Similarly, from the perspective of minimizing the computational effort of reconciling dynamic measurements using DDR techniques it is also reasonable to extend in some cases the range of application of SSDR to deal with dynamic situations. The DR techniques and their range of application are summarized in Figure 2.1; the continuous lines represent the current interconnections whereas the discontinuous line reflects the connection that is addressed in this study.

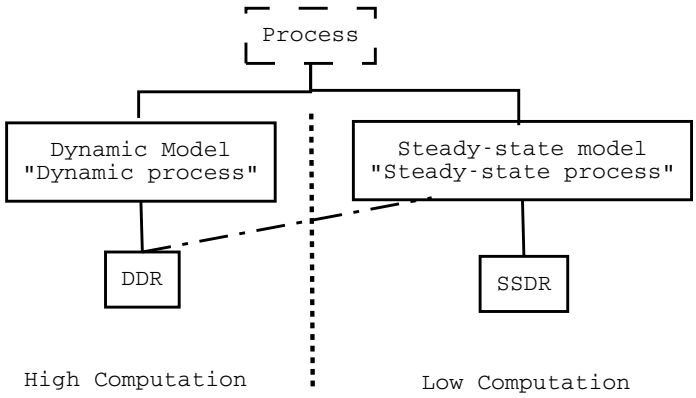


Figure 2.1: Range of application of SSDR and DDR techniques.

Mathematically, a linear dynamic process model can be represented by a set of differential algebraic equations (Eqs. (2.2) and (2.3)). When the process is at steady-state algebraic equations are sufficient to model the correlations between process variables (Eq. (2.3)).

$$\min_{\widehat{\mathbf{x}}} \sum_{i=k-n+1}^{k} \left(\mathbf{x}(t_i) - \widehat{\mathbf{x}}(t_i)\right)^T \mathbf{Q}_{i,\mathbf{x}}^{-1} \left(\mathbf{x}(t_i) - \widehat{\mathbf{x}}(t_i)\right) \tag{2.1}$$

subject to:

$$\mathbf{f}\left(\frac{d\widehat{\mathbf{x}}(t)}{dt}, \widehat{\mathbf{x}}(t), \widehat{\mathbf{u}}(t)\right) = \mathbf{0} \tag{2.2}$$

$$\mathbf{h}\left(\widehat{\mathbf{x}}(t), \widehat{\mathbf{u}}(t)\right) = \mathbf{0} \tag{2.3}$$

In general both SSDR and DDR have the same objective function (Eq. (2.1)), a weighted least-squares equation, where the weights are the variance-covariance matrix of redundant measured process variables. The data reconciliation problem given

by Eqs (2.1), (2.2) and (2.3) can be seen as a dynamic optimization problem. The resolution of these types of problems not only involves an optimization problem but also the solution of the differential equations. However, the DR problem involving only Eqs (2.1) and (2.3) can be seen as a quadratic optimization which presents less complication for its resolution than the former problem and does not present the difficulty of solving the differential equations.

The proposed technique aims to find a way of adapting a SSDR in the case of dynamic processes. This makes the dynamic Data Reconciliation more practical by reducing the computational effort inherently elevated in DDR techniques.

2.2 SSDR approach for dynamic systems

Assuming that the derivative terms are simple process variable, the ordinary differential equation can be transformed to an equivalent set of algebraic equations. Therefore, steady-state data reconciliation techniques can be used rather than DDR techniques.

It is worth mentioning that for a linear dynamic system the balance equations of process variables are also satisfied by the process variable means. Therefore, reconciling the measured variable at each sample time is similar to reconcile the mean of these measured variables.

However, by performing a previous average a data filtering is also achieved which undoubtedly improve the DR performance. Filtering the previously measured data improves the estimation accuracy of DR without increasing the computational effort, making it more appropriate for its use in on-line applications. This approach has been also argued by Kong et al. (2000).

Nevertheless, a difficulty is added for recovery the current reconciled variable from the reconciled averages. Equaling the average of the reconciled variable at each sample time to the reconciliation of the average of the measured variable throughout the sampling time-horizon is one of the points on which is based the proposed Data Reconciliation technique. The proposed approach is shown in Figure 2.2 where the sequential combination of averaging, reconciling and recovering is illustrated. The corresponding DDR is represented by the outer square as is shown in Figure 2.2.

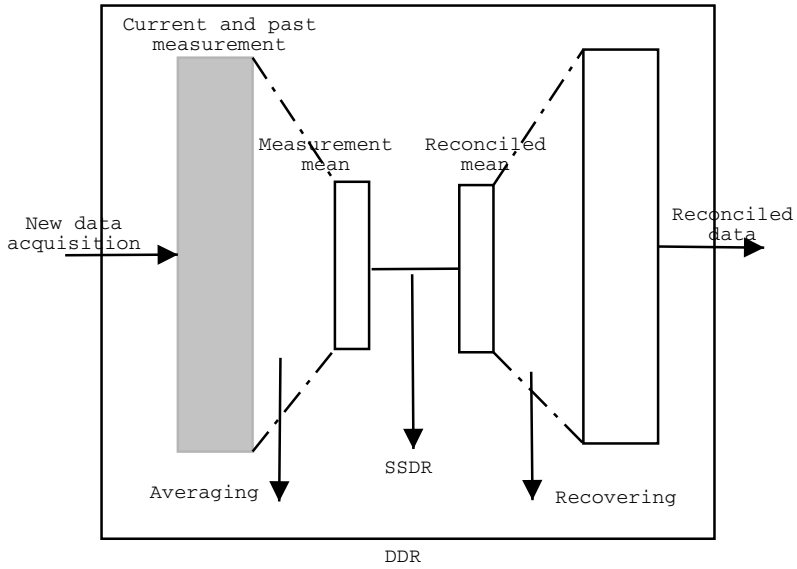


Figure 2.2: MA/SSDR approach for dynamic systems.

2.3 Moving average and SSDR methodology

2.3.1 Averaging step

2.3.1.1 Handling derivative terms

In the proposal it is assumed that no distinction is made among the existing process variables, in the sense that states $\mathbf{x}$, their derivatives $\frac{d\mathbf{x}(t)}{dt}$ and control input variables $\mathbf{u}$ are all included in a unique variable $\mathbf{F} = \left[\mathbf{x}, \frac{d\mathbf{x}(t)}{dt}, \mathbf{u}\right]$.

Taking into account that the considered systems are linear and assuming a small sampling time, the derivative terms can be approximated as follows:

$$\frac{d\mathbf{x}(t)}{dt} = \frac{\Delta\mathbf{x}}{\Delta t} = \mathbf{z} \tag{2.4}$$

By integrating (e.g. cumulative sum) $\mathbf{z}$ over the whole time-horizon allows the recovering of $\mathbf{x}$. This variable substitution (Eq. (2.4)) permits to convert the set of differential equations given by means of Eq. (2.2) to an equivalent set of algebraic ones. Thus, a transformation of the dynamic Data Reconciliation problem to steady-state one is achieved. This is the essence of applying a SSDR to a dynamic system. Therefore, the corresponding canonical form of Eqs. (2.2) and (2.3) can be repre-

sented as follows:

$$\mathbf{AF} = \mathbf{0} \tag{2.5}$$

where $\mathbf{A}$ is the incidence matrix encompassing both $\mathbf{f}()$ and $\mathbf{g}()$ functions.

2.3.1.2 Advantage of averaging

The averaging step is performed over the variable $\mathbf{F} = [\mathbf{x}, \mathbf{z}, \mathbf{u}]$. That is at each sample time k, the average of the latest n samples of measured variable $\mathbf{F}$ is performed as follows:

$$\frac{1}{n} \sum_{i=k-n+1}^{k} \mathbf{F}_i = \overline{\mathbf{F}}_k \tag{2.6}$$

In real systems especially dynamics the most current values tend to better reflect the state of the process. Therefore, the number of sampling points over which the average calculation is done needs to be chosen with care.

In principal Eq. (2.6) can be seen as a moving average filter where the filter weight are all equal to $1/n$. A filter that places most emphasis on the most recent data would be more useful and present better tracking of dynamically changing behavior. From this perspective it is more easy the tuning of n. Too large value of this constant will result in a filtering that does not follow trends quickly enough, whereas small value will lead a poor smoothing. One important advantage of applying the average is that the current measurements are not used directly but are previously de-noised.

Another alternative is to smooth using moving median rather than moving average. The median presents less sensitivity to the presence of outliers as reported by Chen and Romagnoli (1998).

2.3.1.3 Steady-state data reconciliation

SSDR can be formulated as follows:

$$\min_{\widehat{\overline{\mathbf{F}}}_k} \left\{ \left(\widehat{\overline{\mathbf{F}}}_k - \overline{\mathbf{F}}_k \right)^T \overline{\mathbf{Q}}_k^{-1} \left(\widehat{\overline{\mathbf{F}}}_k - \overline{\mathbf{F}}_k \right) \right\} \tag{2.7}$$

subject to:

$$\mathbf{A}\widehat{\overline{\mathbf{F}}}_k = \mathbf{0} \tag{2.8}$$

where $\overline{\mathbf{Q}}_k$ is the variance matrix of the averaged measurement error variables at the current sample time k.

An estimator ($\widehat{\overline{\mathbf{F}}}_k$) can be obtained by solving the above quadratic optimization problem given by Eqs. (2.7) and (2.8) using the Lagrange multiplies see section 1.2.1.3.

The analytical solution is given by the following expression:

$$\widehat{\overline{\mathbf{F}}}_k = \left(\mathbf{I} - \overline{\mathbf{Q}}_k \mathbf{A}^T \left(\mathbf{A}\overline{\mathbf{Q}}_k \mathbf{A}^T\right)^{-1} \mathbf{A}\right) \overline{\mathbf{F}}_k = \Psi \overline{\mathbf{F}}_k \tag{2.9}$$

and the estimator variance is:

$$var\left(\widehat{\overline{\mathbf{F}}}_k\right) = \Psi \overline{\mathbf{Q}}_k \tag{2.10}$$

2.3.1.4 Variance-covariance matrix calculation

Assuming that the distribution of measurements is multivariate normal the *central limit theorem* states that given a distribution with a mean $\mathbf{0}$ and variance $\mathbf{Q}_k$, the sampling distribution of the mean approaches a normal distribution with a mean $\mathbf{0}$ and a variance $\mathbf{Q}_k/n$. Note that the spread of the sampling distribution of the mean decreases as the sample size increases.

The variance of the mean, $\overline{\mathbf{Q}}_k$, can be determined using the measured variable's variance and the time-horizon, n, producing[1]:

$$\overline{\mathbf{Q}}_k = var(\overline{\mathbf{F}}_k) = var\left(\frac{1}{n}\sum_{i=k-n+1}^{k} \mathbf{F}_i\right) = \frac{1}{n^2} var\left(\sum_{i=k-n+1}^{k} \mathbf{F}_i\right) \tag{2.11}$$

assuming the variance of $\mathbf{F}_i$ are constant throughout the time it is possible to get:

$$\overline{\mathbf{Q}}_k = \frac{n}{n^2} var\left(\mathbf{F}_k\right) = \frac{\mathbf{Q}_k}{n} \tag{2.12}$$

where $\mathbf{Q}_k$ is the variance matrix of the measurement error variables at sample time k.

For determining the variance of the derivative terms $\mathbf{z}$, it is taking into account that $\mathbf{x}$ are cumulative sum of the last k value of $\mathbf{z}$, thus, $var\left(\mathbf{x}\right) = k.var(\mathbf{z})$ and it is possible to get:

$$var(\overline{\mathbf{z}}) = \frac{\mathbf{Q}_{k,\mathbf{x}}}{n.k} \tag{2.13}$$

2.3.2 Recovering step

Once the steady-state Data Reconciliation is performed, it is necessary to recover the "reconciled" process variables values from the reconciled variables mean. Equaling the average of the reconciled variable at each sample time to the reconciliation of the average of the measured variable throughout the sampling time-horizon n (Eq. (2.14))

[1] Elemental variance's properties: $var(a.x) = a^2.var(x)$, if a is a constant and $var(\sum_{i=1}^{n} x_i) = \sum_{i=1}^{n} var(x_i)$.

is used to recover the reconciled value at the current time k, $\widehat{\mathbf{F}}_k$ from the reconciled average, $\widehat{\overline{\mathbf{F}}}_k$.

$$\frac{1}{n}\left(\sum_{i=k-n+1}^{k}\widehat{\mathbf{F}}_i\right)=\left(\frac{1}{n}\widehat{\sum_{i=k-n+1}^{k}}\mathbf{F}_i\right)=\widehat{\overline{\mathbf{F}}}_k \tag{2.14}$$

This recovery can also be achieved through Eq. (2.15), as follows:

$$\widehat{\mathbf{F}}_k=n\widehat{\overline{\mathbf{F}}}_k-\sum_{i=k-n+1}^{k-1}\widehat{\mathbf{F}}_i \tag{2.15}$$

$$\widehat{\mathbf{F}}_k=n\left(\widehat{\overline{\mathbf{F}}}_k-\widehat{\overline{\mathbf{F}}}_{k-1}\right)+\widehat{\mathbf{F}}_{k-n} \tag{2.16}$$

Using Eqs. (2.6), (2.9) and (2.14) it can be proved that the estimator is unbiased:

$$\sum_{i=k-n+1}^{k}\widehat{\mathbf{F}}_i=\Psi\sum_{i=k-n+1}^{k}\mathbf{F}_i \tag{2.17}$$

or

$$\widehat{\mathbf{F}}_k=\Psi\sum_{i=k-n+1}^{k}\mathbf{F}_i-\sum_{i=k-n+1}^{k-1}\widehat{\mathbf{F}}_i \tag{2.18}$$

If $\mathbf{F}_i=\mathbf{F}_i^*+\Xi_i,\forall i$, and assuming $\mathbf{A}.\mathbf{F}^*=\mathbf{0}$, Eq. (2.19) is obtained:

$$\left\langle\sum_{i=k-n+1}^{k}\widehat{\mathbf{F}}_i\right\rangle=\sum_{i=k-n+1}^{k}\mathbf{F}_i^*+\Psi\left\langle\sum_{i=k-n+1}^{k}\Xi_i\right\rangle=\sum_{i=k-n+1}^{k}\mathbf{F}_i^* \tag{2.19}$$

Therefore, under deterministic conditions, the proposed estimator does not introduce any bias and possess the desirable statistical property of being unbiased.

2.4 Case study: results and discussion

The Kalman filter is selected as the dynamic Data Reconciliation technique reference for evaluating the performance of the proposed "dynamic" Data Reconciliation MA/SSDR (Moving Average/Steady-State Data Reconciliation) technique. A Kalman filter has been designed to estimate the output based on the noisy measurements.

The process represented in Figure 2.3 is taking from Darouach and Zasadzinski (1991) and selected for evaluating and comparing the proposed technique, where a random error following a normal distribution is added to the measured variables.

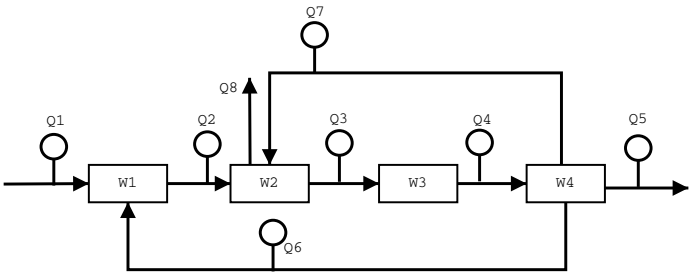


Figure 2.3: Process network.

First the process is considered at steady-state. Then, a sudden but lasting change is introduced when time = 100 by means of a step in Q_1. Since the dominant time constant of the dynamic response is much slower than the period with which disturbances enter to the system, the system displays quasi-steady-state behavior. Later a ramp with a slop of 0.02 is added to the input variable Q_1 leading to some changes in the system dynamics.

As is explained in previous sections, the width of moving time-window directly affects the estimation accuracy as well as the tracking of the dynamic drastic behavior. In Figure 2.4 the estimation of the flow-rate (Q_3) is represented for a moving-horizon (MH) of 5 and 25 time units. It can be seen from the curves that the MA/SSDR estimation accuracy is improved when increasing the moving horizon length however, the tracking of the steps presents an unacceptable behavior, a delay response of almost 20 time unit is observed.

Tables 2.1 and 2.2 represent the variance of the measurement by means of KF and MA/SSDR estimators for input/output variables and states respectively. Since both estimators are unbiased the variance is equal to the Mean-Square Error (MSE). The MSE is often used for evaluating the estimators performance. The second row of both Tables 2.1 and 2.2 represent the variance of the raw measurements.

Table 2.1: MSE of MA/SSDR and KF for input/output variables.

	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7	Q_8
Measurements	2	2	2	2	5	2	2	2
KF	0.32	0.33	0.66	0.53	0.31	0.13	0.26	0.16
MA/SSDR	0.37	0.28	0.30	0.26	0.61	0.20	0.30	0.32

By analysis of the results presented in Tables 2.1 and 2.2 it can be concluded

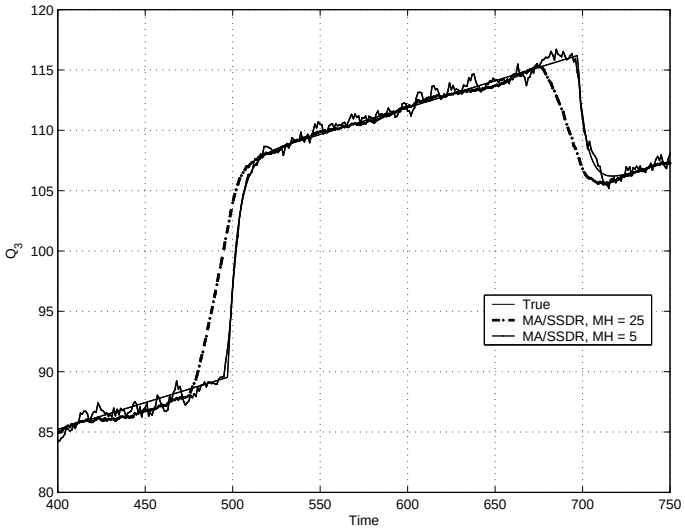


Figure 2.4: Effects of increasing the MH width on the MA/SSDR estimation.

Table 2.2: MSE of MA/SSDR and KF for state variables.

	W_1	W_2	W_3	W_4
Measurements	2	5	2	6
KF	0.45	1.17	0.68	1.35
MA/SSDR	0.44	1.91	0.70	1.63

that the variance reduction using the MA/SSDR approach is very attractive and even better in some cases than the KF. These results are confirmed in Figures. 2.5 and 2.6 representing respectively the behavior of variables Q_1 and W_4. Additionally, from Figure 2.6 it can be seen that the proposal of reconciling and recovering dynamic variables is considerably accurate.

Demonstrating and validating the application of the SSDR for dynamic behavior process variables. Also the tracking of changes in dynamical behavior is better tackled by the proposed method than KF. Finally, Table 2.3 summarizes an average time of the required time for performing the MA/SSDR and KF, from this results it can be noted a slight fastness of the MA/SSDR. This could be crucial for on-line applications.

Table 2.3: Fastness of the MA/SSDR compared with KF.

MA/SSDR	2.19 CPU time
KF	4.15 CPU time

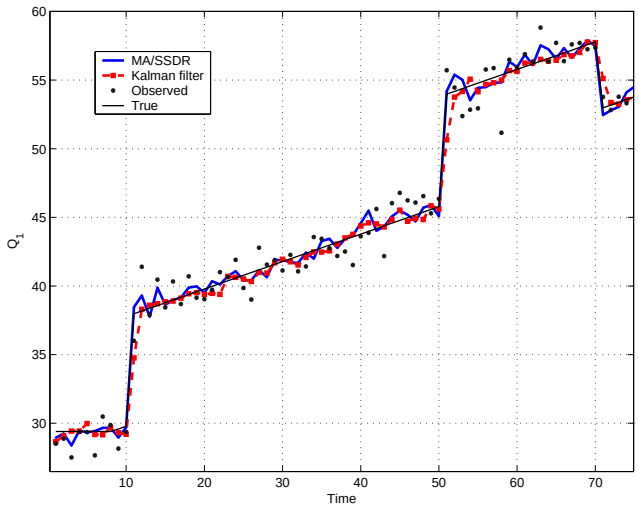


Figure 2.5: Estimation of the input flow-rate Q_1 using MA/SSDR and KF.

2.5 Conclusions

In this chapter a Data Reconciliation technique for dynamically evolving processes is considered from a steady-state point of view. The technique uses previous information by means of the variables averages to generate the current estimates. Thus, not only the temporal redundancy is taken into account but also the quality of estimation is improved by combining a filtering technique and DR in a sequential way. Furthermore, a way for processing the average values to recover the "reconciled" variable is presented and validated. The proposed unbiased estimator is compared with the Kalman filter technique to evaluate its performance in terms of variance reduction and fast dynamical changes tracking.

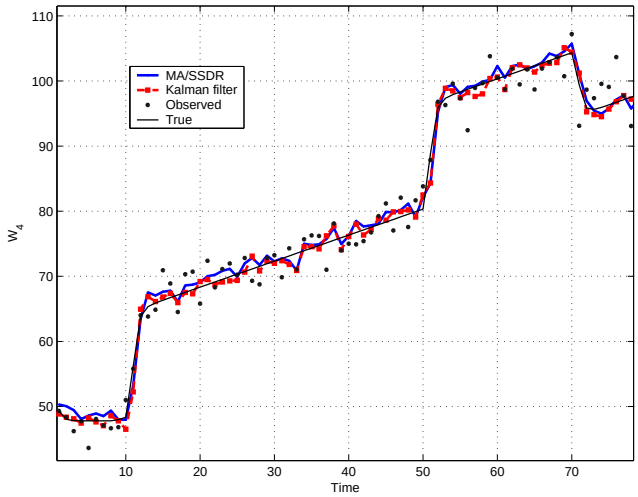


Figure 2.6: Estimation of hold-up W_4 using MA/SSDR and KF.

Nomenclature

$\mathbf{A}$	incidence matrix
$\mathbf{F}$	vector containing both $\mathbf{x}$, $\mathbf{z}$ and $\mathbf{u}$ respectively
$\mathbf{F}^*$	vector of unknown true value of $\mathbf{F}$
$\overline{\mathbf{F}}$	vector of the mean of $\mathbf{F}$
$\mathbf{f}()$	differential equations set
$\mathbf{h}()$	algebraic equations set
$\mathbf{I}$	identity matrix
n	width of the moving time-horizon
$\mathbf{Q}_k$	variance covariance matrix of $\mathbf{F}_k$
$\overline{\mathbf{Q}}_k$	variance covariance matrix of $\overline{\mathbf{F}}_k$ at the sample time k
$\mathbf{Q}_{k,\mathbf{x}}$	variance covariance matrix of $\mathbf{x}$ at sample time k
t	time
$\mathbf{u}$	vector of input variables
$\mathbf{x}$	vector of state variables
$\mathbf{z}$	vector representing the derivative with time of $\mathbf{x}$

Greek letters

σ	standard deviation
Φ	objective function of the DR optimization problem

Ξ vector of random error with mean zero and known variance
Ψ matrix obtained by combining $\mathbf{A}$ and $\overline{\mathbf{Q}}_k$ as shown in Eq. (2.9)

Subscripts

i sample time within n
k current sample time

Operators

$var()$ variance function
$\langle\rangle$ expected value

Chapter 3

Dynamic Data Reconciliation based on wavelet analysis

Abstract

In this proposal wavelet trend analysis is performed as a former step for reconciling dynamic systems. That is the deterministic trends of measured data along a specified time-horizon are identified by means of wavelet analysis and then they are made consistent with the dynamic process model.

Looking for the trend of measurements, a de-noising is carried out. This pre-processing of data leads to an increase in the Data Reconciliation performance. Additionally, the variance estimation of dynamic variables is undertaken considering that the trends contains only the inherent process variability, since the measurement error variability is considerably compensated through wavelet transform. The performance of the proposed strategy is compared with Kalman filtering and discretization techniques for a linear dynamic case. Further extensions to contemplate nonlinear cases show promising results.[1]

3.1 Introduction

Considering that the general DDR optimization problem as discussed in Chapter 1 section 1.2.5 can be expressed as follows:

$$\min_{\hat{\mathbf{y}}(t)} \sum_{k=0}^{c} \left[\hat{\mathbf{y}}(t_k) - \mathbf{y}(t_k)\right]^T \mathbf{Q}^{-1} \left[\hat{\mathbf{y}}(t_k) - \mathbf{y}(t_k)\right] \tag{3.1}$$

[1]A part of this work was proposed in the **REALISSTICO** project under contracts QUI-99-1091, CICYT.

subject to:

$$\frac{d\widehat{\mathbf{y}}(t)}{dt} = \mathbf{f}\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) \tag{3.2}$$

$$\mathbf{0} = \mathbf{h}\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) \tag{3.3}$$

$$\mathbf{0} < \mathbf{g}\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) \tag{3.4}$$

The solution of this dynamic optimization problem involves three sub-problems: (1) the solution of the differential equations (2) the estimation of the variance of the process variables and (3) the proper optimization problem. If the process model presents some nonlinearity the solution of the problem presented above is even more complicated.

In this chapter a novel nonlinear dynamic data reconciliation approach is proposed. It is based on reconciling the trends of measured variables along a specified time-horizon. Before making these trends consistent with the dynamic process model they have to be identified. For such purpose a trend extraction analysis with wavelets has been adopted. Applying a trend analysis technique previously to DDR provides important benefits:

1. Firstly, the complication of evaluating the dynamic process variable variance-covariance matrix is reduced since the variability of the trend is mainly due to the process itself. **Wavelets analysis can de-noise measured variable taking into account the magnitude of the noise's variance**.

2. Secondly, pre-processing raw measured data by means of trend analysis involves a de-noising of data and elimination of abnormal data in measurements which in turn leads to a better estimation accuracy as discussed in Chapter 2.

3. Finally, the proposal can perform reconciliation independently of the variables sampling times.

The proposed approach has been compared with Kalman filter technique in a simulated continuous stirred tank reactor (CSTR) in order to highlight its performance.

3.2 Extended Polynomial Approach (EPA)

3.2.1 Reformulation of DDR optimization problem

3.2.1.1 Nonlinear handling

Mainly the nonlinear optimization problem can be solved directly using nonlinear programming solvers (e.g. GAMS/MINOS). Although these methods provide accu-

rate variable estimation they require large computational time, this might represents a drawback for on-line applications.

An alternative approach is to linearize the nonlinear optimization problem and then linear techniques for solving the linear optimization problem can be applied. However, this approximated linearized model will undoubtedly decrease the accuracy of variable estimation through DR.

This work addresses the NLP problem in a efficient and easy way. The essence of the proposal relies on representing the nonlinear equations as a linear combination of new variables $\mathbf{z}_i(t)$. The new variables $\mathbf{z}_i(t)$ are obtained from variables $\mathbf{y}(t)$, $\mathbf{u}(t)$ and parameters $\widehat{\theta}(t)$ through a well defined functions $f_i(.)$ as follows:

$$\frac{d\widehat{\mathbf{y}}_j(t)}{dt} = \mathbf{f}_j\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) = \sum_{i=1}^{I^z} f_i\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) = \sum_{i=1}^{I^z} \widehat{\mathbf{z}}_i(t) \tag{3.5}$$

The same can be performed for algebraic equations:

$$\mathbf{0} = \mathbf{h}_j\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) = \sum_{i=I^z+1}^{I} f_i\left(\widehat{\mathbf{y}}(t), \widehat{\mathbf{u}}(t), \widehat{\theta}(t)\right) = \sum_{i=I^z+1}^{I} \widehat{\mathbf{z}}_i(t) \tag{3.6}$$

The values of the variables $\mathbf{z}(t)$ and their variances are straightforward obtained by applying the corresponding functions $f_i()$ on the measured variables $\mathbf{y}(t)$ and $\mathbf{u}(t)$ and parameters $\theta(t)$. The reconciliation is carried out both on state and variables $\mathbf{z}(t)$, nevertheless, only state variables are focused since some of the variables $\widehat{\mathbf{z}}(t)$ are meaningless.

3.2.1.2 Differential equation handling

By combining Eqs. (3.5) and (3.6) and assuming that the derivative terms $\frac{d\widehat{\mathbf{y}}(t)}{dt}$ are represent by the process variables $\widehat{\mathbf{x}}(t)$ (see Chapter 2 section 2.3.1.1) it is possible to get a linearized model, Eq. (3.7). This linearized model substitutes Eqs. (3.2) and (3.3) in DDR formulation.

$$\widehat{\mathbf{x}}(t) = \mathbf{A}.\widehat{\mathbf{z}}(t) \tag{3.7}$$

where **A** is the incidence matrix of the process under consideration.

The i^{th} process variable, $\widehat{\mathbf{z}}_i(t)$ and the j^{th} variable $\widehat{\mathbf{x}}_j(t)$ can be fitted by p-degree polynomials in time over a specified time-horizon as follows:

$$\mathbf{z}_i(t) = \sum_{k=0}^{p_i}\left(\alpha_{i,k}t^k\right), \ \forall i = 1\cdots I \tag{3.8}$$

$$\mathbf{x}_j(t) = \sum_{k=0}^{p_j} \left(\omega_{j,k} t^k \right), \ \forall j = 1 \cdots J \tag{3.9}$$

By substituting Eq. (3.8) into Eq. (3.7) and equaling it to Eq. (3.9), the functional and temporal redundancy are combined. Then, it is possible to get for each time t_k the following constraint instead of Eq. (3.7).

$$\omega_{j,k} = \sum_{i=0}^{I} (A_{j,i} \alpha_{i,k}), \ \forall j, \forall k \tag{3.10}$$

or in a matrix form:

$$\Omega^{JxK} = \mathbf{A}.\Lambda^{IxK} \tag{3.11}$$

where each row j in Ω contains the coefficients in time descending order of the variable $\mathbf{x}_j(t)$, while each row i in Λ contains the coefficients in time descending order of the process variables $\mathbf{z}_i(t)$.

The polynomials degree p_i and p_j can be set to a value of two if the moving time window selected is considerably small. That is the dynamic of each involved process variable, $\mathbf{z}_i(t)$ and $\mathbf{x}_j(t)$ might be represented by a polynomial of degree two. The largest polynomial degree of the involved variables (p^{max}) is used to determine the number of columns of Ω and Λ matrices.

This variable substitution Eqs (3.8) and (3.9) affects also the objective function of the DDR formulation presented in Eq. (3.1).

Given an "accurate" process model an ideal data reconciliation scheme would use all information (process measurements) from, the start-up t_0 of the process until the current time t_c. Unfortunately, such a scheme would necessarily result in an optimization problem of ever-increasing dimension. For practical and on-line implementation as focused in this work a moving time window should be used. This will undoubtedly reduce the optimization problem to manageable dimension. The moving time window approach was earlier presented by Jang et al. (1986) and extended by Liebman et al. (1992).

3.2.1.3 Variance-covariance calculation

Calculation of the measurement error's covariance matrix **Q** presents more difficulty for dynamic systems, since the intrinsic variability of the process is combined with measurement errors variability and both variability are continuously changing.

Some important issues are considered to adapt the traditional form of variance estimation (Eq. (1.20)) for dynamic cases.

Firstly, it is assumed that an error is often the sum of a large number of single, elementary errors. According to the *central limit theorem*, under certain generally

acceptable conditions, the distribution of such a sum approaches the normal distribution. Thus, in this work the measurement error variability is assumed to follow a normal distribution.

Secondly, the spread of measured data is calculated over the **estimated** values since neither the expected values nor the sample mean of measured variables are known.

Finally, it is assumed that the measurements are independent, thus, the off-diagonal elements of **Q** are equal to zero.

It is possible to use all the past estimations and measurements, however, if the variance is not constant throughout the time, this might introduce some errors in estimating **Q**. That is, the current measurement should have more similarity to the most recent one in terms of variance. Therefore, a moving time-window approach is also favorable for calculating the measurement error's covariance matrix of dynamic systems.

Based on these assumptions a recursive estimation of matrix **Q** is constructed and continuously updated using past n measured and estimated values according to Eq. (3.12).

$$\mathbf{Q}_{\mathbf{y}(t_k)} = \frac{1}{n-1} \sum_{k=c-n}^{c} \left(\widehat{\mathbf{y}}(t_k) - \mathbf{y}(t_k)\right)^2 \tag{3.12}$$

where n is the width of the moving time-window.

3.2.2 On-line solving of the reformulated DDR

The reformulated DDR optimization problem expressed by Eqs. (3.13) and (3.14) is solved numerically.

$$\min \sum_{k=c-n}^{c} \left[\frac{\widehat{\mathbf{y}}(t_k) - \mathbf{y}(t_k)}{\sigma_{\mathbf{y}(t_k)}}\right]^2 + \left[\frac{\widehat{\mathbf{z}}(t_k) - \mathbf{z}(t_k)}{\sigma_{\mathbf{z}(t_k)}}\right]^2 \tag{3.13}$$

subject to:

$$\widehat{\omega}_{j,k} = \sum_{i=0}^{I} (A_{j,i}\widehat{\alpha}_{i,k}), \ \forall j, \forall k \tag{3.14}$$

The solution permits the easy introduction of inequality constraints Eq. (3.4) and the over-specified polynomial degree of some process variables can be bounded.

To solve the optimization problem a projective pre-conditioned conjugate gradient algorithm available in the MATLAB optimization Toolbox can be used.

The optimized coefficients $\widehat{\omega}_{j,k}$ and $\widehat{\alpha}_{i,k}$ serve for reconstructing the "reconciled" process variables using Eqs. (3.15) and (3.16).

$$\widehat{\mathbf{y}}_j(t) = \mathbf{y}_j(0) + \sum_{k=0}^{p_j} \frac{\widehat{\omega}_{j,k}.t^{k+1}}{k+1}, \ \forall j = 1 \cdots J \tag{3.15}$$

where the integration constant $\mathbf{y}_j(0)$ corresponds to initial conditions. This constant value has to be calculated accurately by an off-line analysis of state variables otherwise the estimation quality may be affected.

$$\widehat{\mathbf{z}}_i(t) = \sum_{k=0}^{p_i} \widehat{\alpha}_{i,k}.t^k, \ \forall i = 1 \cdots I \tag{3.16}$$

The Extended Polynomial Approach (EPA) presented in this study is an extension of the work presented by Bagajewicz and Jiang (1997) to deal with different important features of DDR problems. The EPA procedure can deal with nonlinearity, and can be applied on-line. Furthermore, the calculation of measurement error variance is focused along with the determination of the adequate polynomial degree of process variables. This EPA approach has been applied satisfactorily in a polymerization reactor as reported in Benqlilou et al. (2001c).

3.3 Integrating wavelets analysis in the EPA procedure

The steps involved in the proposed approach start by extracting the de-noised trends of measured data. For such purpose wavelet de-noising utilize the temporally redundant information of measurements so that random errors are reduced and de-noise trends are extracted. Although these trends might be more accurate than the measurements they might be inconsistent with process model constraints. So, the Data Reconciliation is needed along with the de-noising. Thus, the de-noised trend obtained by wavelet transform is assumed as measured data obtained by more accurate instruments. The polynomial degree determination step allows the determination of the polynomial degree of each trend that is the coefficients to be adjusted through Data Reconciliation.

3.3.1 De-noised trend extraction by wavelet transforms

Wavelets transform technology is capable of decomposing any signal into its contributions in different regions of the time-frequency or time-scale[2] space by projection on the corresponding wavelet basis function.

[2]scale means frequency band.

Wavelets transform have been extensively used in chemical engineering during the last ten years (Bakshi (1998); Doymaz et al. (2001); Addison (2002)). In all the existing applications, this capability of wavelets to decompose the raw signal and to extract a filtered approximation of it, is commonly exploited.

Particularly, this has been proved as very useful to signal de-noising and signal trend extraction applications (Doymaz et al. (2001)) where any measured variable signal, $\mathbf{y}(t)$, is assumed as the result of:

$$\mathbf{y}(t) = \mathbf{y}^*(t) + \varepsilon(t) \tag{3.17}$$

where, $\mathbf{y}^*(t)$ vector of true process variables and $\varepsilon(t)$ is its measurement error (noise). This basic idea is also exploited in this work to obtain an estimate of the trend signal.

Different strategies for de-noising and signal estimation have been proposed based on Donoho and Johnstone (1995) proposal:

1. decompose the raw signal taking of the Wavelet Transform (WT). Depending on the signal nature (continuous or discrete) the wavelet transform must be applied continuously (CWT) or discretely (DWT).

2. remove wavelets coefficient below a certain threshold value β, this step is commonly called thresholding.

3. reconstruct processed signal using the inverse of the DWT.

Nevertheless, some difficulties are encountered to define the depth of wavelet's decomposition, to determine the threshold value β or to select the wavelet family.

Firstly, the optimum depth of the decomposition of the wavelets is related to the thresholding as reported by Nounou and Bakshi (1999). Thresholding wavelet coefficients at very coarse scales may result in the elimination of important features, whereas thresholding only at very fine scales may not eliminate enough noise.

Secondly, most used wavelet filter for de-noising and trend extraction are the Daubechies wavelets filters (*dbN*). This is based on their high capability at representing polynomial behaviors within the signal (Rowe and Abbott (1995); Flehmig et al. (1998)). Nevertheless, the *N* order (or vanishing moment), associated to the different *dbN* ($N = 1..., 20, ...$), to be selected can affect the quality of the rectification of the signal (Nounou and Bakshi (1999)). Different subjective criteria or rules of thumb are mentioned by some authors to select the *db*, while other works do not consider this issue. In the next sections a proposal is presented to overcomes these issues.

3.3.1.1 Optimal depth of wavelet's decomposition

The de-noising performance of the wavelet transform can be assessed by calculating the mean-square error (or power) between the raw and the processed signal. The processed signal can be the one obtained by applying the inverse DWT over the approximated coefficients. Therefore, as the scale is increased the MSE will be reduced, showing the de-noising capability of wavelets. However, at a certain scale the wavelet will not only remove the noise but also will eliminate deterministic features of the signal, this will undoubtedly make the MSE higher. Therefore, the optimal decomposition scale is the one corresponding to the first minimum MSE reached.

Mathematically, the different steps required for this optimal depth (dyadic scale) determination are as follow:

1. The scaling and wavelet coefficients, $u_{l,v}$ and $w_{l,v}$ respectively, at various scales l and dilation v are obtained by taking of the DWT over $y(t)$.

2. The *approximation* component, A_l, at each scale l is reconstructed through:

$$A_l = \sum_{l=1}^{L} u_{l,v}\phi_{l,v}(t) \tag{3.18}$$

3. The power P_l contained in the difference between $y(t)$ and A_l at each scale l (da_l) is calculated through, Eq. (3.19).

$$P_l(da) = \sum_{l=1}^{L} \left|y(t) - A_l\right|^2 = \sum_{l=1}^{L} \left|da_l(t)\right|^2 \tag{3.19}$$

4. The variation of power, Eq. (3.19), between successive scales is computed using Eq. (3.20). As the dyadic scale increases the power due to the noise calculated in Eq. (3.19), decreases rapidly until it reaches a first minimum. The optimal scale L_m corresponds to the first minimum encountered

$$\Delta P = P_l(da) - P_{l-1}(da) \tag{3.20}$$

5. At this scale L_m a first thresholding based on setting to zero all the $w_{l,v}$ greater than L_m is performed. Then, a second thresholding over the remaining coefficients is performed through Visushrink methods reported by Nounou and Bakshi (1999).

6. The de-noised signal is obtained by taking the inverse of DWT as shown in Eq. (3.21).

$$\widehat{y}(t) = \sum_{v=-\infty}^{\infty} u_{L_m,v}\phi_{L_m,v}(t) + \sum_{l=1}^{L_m}\sum_{v=-\infty}^{\infty} w^*_{l,v}.\psi_{l,v}(t) \tag{3.21}$$

An additional consideration made about the selection of the wavelets family is the determination of its order or vanishing moment N. An comparative study (Tona et al. (2003)) shows that Daubechies of order $6-7$ (and the combination of its signals) are appropriate to deal with a wide range of signals. Then, Daubechies wavelets of order 6 and 7 have been used in the proposal presented above.

3.3.1.2 Illustrative example

The heavisine signal is used to illustrate the procedure presented in the previous section. As it can be seen in Figure 3.1, the filtering of the reconstructed heavisine signal (Nounou and Bakshi (1999)) using db_6 is optimum by comparing it with the true signal. Therefore, the decomposition of the signal, until the power of the da_l component reaches its first minimum with subsequent shrinkage, provides optimum level L_m and coefficients $\psi_{l,v}$ as it can be seen in Figure 3.2.

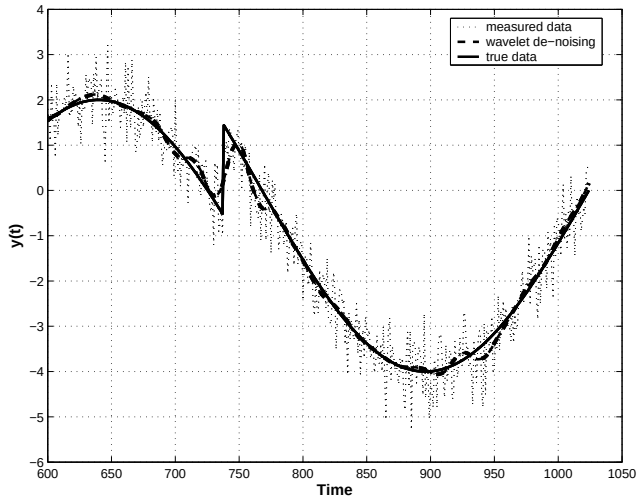


Figure 3.1: The heavisine function filtered by means of wavelet db_6 at scale 4.

3.3.1.3 Moving time-window handling

When a moving time window approach is used in Data Reconciliation each process variable will be reconciled *n* times, due to the fact that each sample will intervene in *n* DR runs.

Mingfang et al. (2003) propose to reconcile each time the already reconciled measured variables by taking advantage of the current reconciliations. This approach can

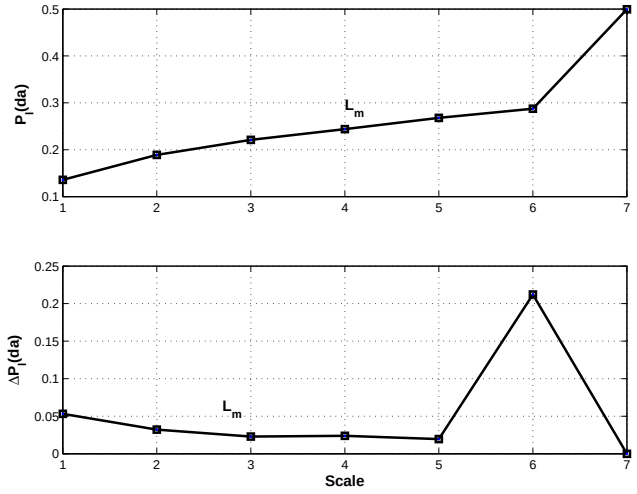


Figure 3.2: Determining the optimal dyadic scale L_m.

undoubtedly improves the estimation as they argued. However, it is not appropriate for on-line applications due to the computation requirements. Furthermore, avoiding to use the previous adjustment in the current estimation is also important from the trend analysis perspective, since, appending window with reconciled values may lead to a distortion of the process variable trend extraction through DWT.

3.3.2 Polynomial degree of the de-noised trend

Using the de-noised trend of measurements obtained in section 3.3.1.1, polynomial degree determination for each process variable can be carried out. This task is repeatedly performed starting by a null degree and increasing it each time by one until the fitting error between polynomial output and trend is less than a specified threshold value.

The resulting polynomial degree for each process variable is introduced in Eq. (3.11) and the values of these coefficients are adopted as the initial solution in the numerical optimization of the DDR problem.

3.3.3 Handling different sampling times

In a real operating plant it is possible to find out that all measurements are not acquired at the same sampling time (e.g. composition analyzer and flow-meters). To deal with this situation the presented proposal performs a reconciliation at the sam-

pling time of the least frequently measured variable or the maximum sample time (T_s^{max}). The estimation accuracy of these measured data is mainly based on the functional redundancy. Whereas the estimation accuracy of process variables with a frequency measurement greater than T_s^{max} is achieved by means of both temporal and function redundancy.

The process variables with a sampling time less than T_s^{max} have an intermediate value which do not present a functional redundancy. Nevertheless, by taking into account all the intermediate samples in the trend extraction procedure the temporal redundancy is taking into account. Moreover, by using the reconciled value at T_s^{max} the accuracy of the extracted trend is improved.

3.4 Case study: results and discussion

3.4.1 Case study - CSTR reactor

A simulated continuous stirred tank reactor with external heat exchange and in which a first-order exothermic reaction (decomposition of a reactant A) occurs (Liebman et al. (1992)) is considered for demonstrating the performance in terms of accuracy at transient times of the proposed DDR techniques.

The equations (3.22-3.25) governing this process are differential equations describing the change in reactor hold-up V, the change in concentration of reactant A, change in temperature of reactor T, and the Arrenhius equation respectively.

$$\frac{dV}{dt} = q_0 - q \tag{3.22}$$

$$\frac{dA}{dt} = \frac{q_0}{V}A_0 - \left(\frac{q}{V} + K\right)A \tag{3.23}$$

$$\frac{dT}{dt} = \frac{1}{V}\left(T_0.q_0 - T.q\right) + \frac{-\alpha_d.\Delta H.A_r.K.A}{\rho.C_p.T_r} + \frac{-U.A_R}{\rho.C_p.V}.\left(T - T_c\right) \tag{3.24}$$

$$K = K_0.\exp\left(\frac{-E_A}{T.T_r}\right) \tag{3.25}$$

where A_0 and T_0 are the feed concentration and temperature; q_0 and q are the feed flow and output flow. K is the rate constant that can be computed according to the Arrhenius expression, E_A is activation energy, A_R is heat transfer area, ρ is density, C_p is heat capacity, ΔH is the heat of reaction, U is the heat transfer coefficient, and T_c is the coolant temperature.

In the following calculation, concentrations and temperatures were scaled by using a nominal reference concentration A_r and nominal reference temperature T_r. The physical constants and parameters value are given in Table 3.1.

Table 3.1: Physical constants and parameters value for the CSTR reactor.

Parameter	*Value*	*Unit*	*ProcessVariable*
q	10.0	cm^3s^{-1}	Output flow
V	1000.0	cm^3	Hold-up
ΔH_r	-27000.0	$calgmol^{-1}$	Heat of reaction
ρ	0.001	gcm^{-3}	Density
C_p	1.0	$cal(gK)^{-1}$	Heat capacity
U	5.010^{-4}	$cal(cm^2sK)^{-1}$	Heat transfer coefficient
A_R	10.0	cm^2	Heat transfer area
T_c	340.0	K	Coolant temperature
T_0		K	Feed temperature
T		K	Tank temperature
K_0	7.8610^{12}	s^{-1}	Arrhenius constant
K		s^{-1}	Arrhenius rate expression
E_A	14090	K	Activation energy
α_d	1.0	–	Catalyst deactivation parameter
A_0		$gmolcm^{-3}$	Feed concentration
A		$gmolcm^{-3}$	Tank concentration

Measurements for state variables A, T, V, and input variables A_0, T_0, V_0, q_0 and q were simulated at time step 2.5 during a time-horizon of 60. A Gaussian noise with standard deviation of 5% was added to these simulated values obtained through numerical integration.

3.4.2 Linear case - CSTR reactor

Focusing the mass balance Eq. (3.22) an open-loop response of the CSTR process is simulated. In Figures 3.3, 3.4, and 3.5 it can be seen the behavior of q_0, V and q respectively when a ramp in the feed q_0 with a slope of 0.02 is added at the sampling time 30. The true value of q_0 is equal to 10 in the first 30 time units then is increased by a slope of 0.02 whereas q is control to maintain a value of 9. The estimation accuracy of the proposal is highlighted. It can be also seen that the EPA/wavelet provides a very high estimation of the signal.

If the behavior of process variables is smooth the proposed approach provides the most accurate estimations of the compared techniques as shown in Tables 3.2 and 3.3. Kalman filtering has been adopted as the DDR reference technique because of its superior performance over the linear dynamic Data Reconciliation techniques

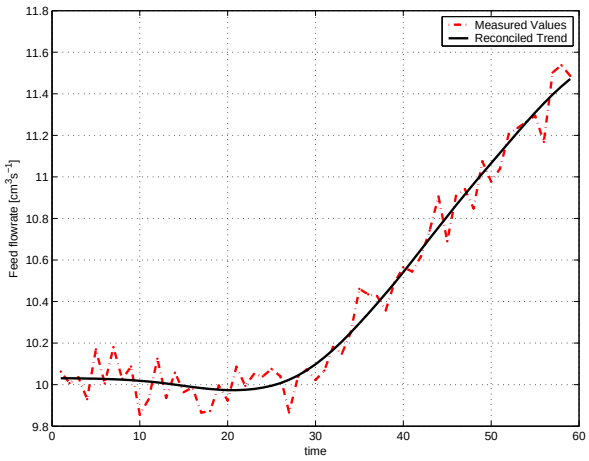


Figure 3.3: Reconciling the feed flow-rate using the EPA/wavelet.

based on first-order either discretization (Rollins and Devanathan (1993)) and the polynomial approach (Bagajewicz and Jiang (1997)) as presented in Benqlilou et al. (2002a).

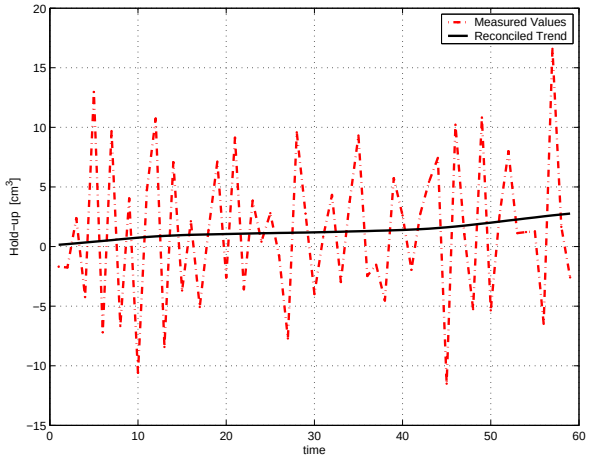


Figure 3.4: Reconciling the hold-up using the EPA/wavelet.

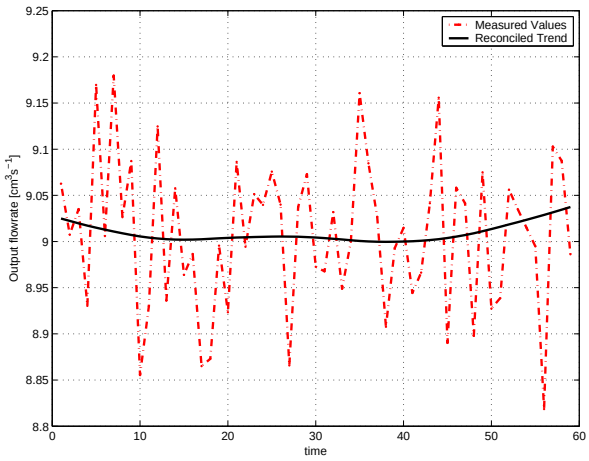


Figure 3.5: Reconciling the output flow-rate using the EPA/wavelet.

Table 3.2: Resulting variance of the different DDR techniques.

	EPA	KF	$EPA/wavelet$
q	0.0053	0.0021	0.0002
q_0	0.0111	0.0086	0.0024

The performance of Kalman filter applied for the CSTR case study can be seen in Figure 3.6.

If the process variable presents a fast changing dynamics (step change in a set point), the polynomial degree of that variable may be considerably high. Additionally, the process variables related to that variable, by means of some correlations, have not necessarily a step behavior thus a more reduced degree is required for their representation. In such situation the proposed approach is faced with a serious draw-

Table 3.3: Percentage deviation of reconciled variables of the DDR techniques.

	EPA	KF	$EPA/wavelet$
q	0.66	0.42	0.17
q_0	0.22	0.17	0.05

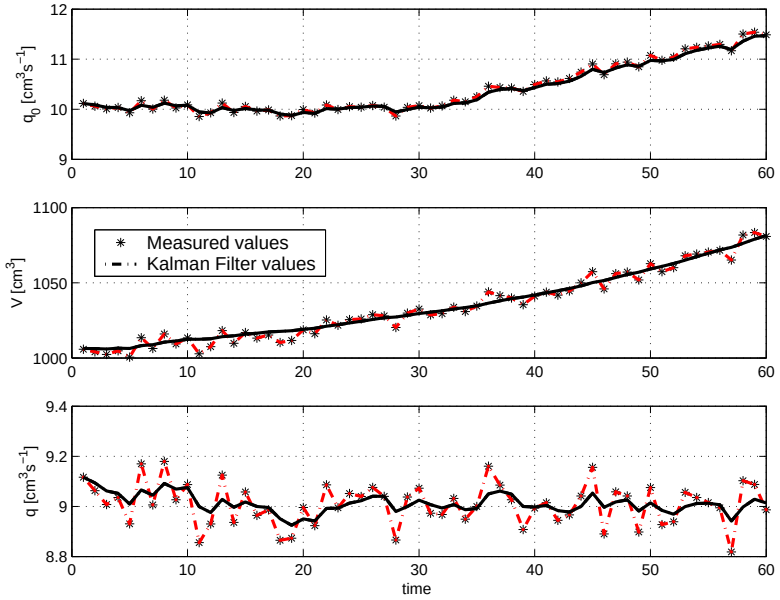


Figure 3.6: Reconciliation of q_0, V and q using Kalman filter.

back: many coefficients are not redundant which leads their to inaccurate estimation. A first proposal to overcome this shortcoming is to use uniquely the wavelet when the step occurs. This approach has proved accurate estimations as seen in Figure 3.7. Once there is sufficient data (6 or 8 samples), the combination approach presented before can be applied.

Additionally, and for on-line Data Reconciliation, the CPU time in an Intel Pentium IV at $2.8\,GHz$ has been 0.6 seconds.

3.4.3 Nonlinear case - CSTR reactor

The nonlinear approximation presented earlier in section 3.2.1.1 is illustrated in this case study. From the right hand side of Eqs. (3.22), (3.23) and (3.24) it is possible to get the new variables $\mathbf{z}(t)$ adopting the following substitutions.

$$\mathbf{z}_1(t) = q_0, \quad \mathbf{z}_2(t) = -q \ , \mathbf{z}_3(t) = \frac{q_0}{V}A_0, \ \mathbf{z}_4(t) = -\left(\frac{q}{V} + K\right)A \tag{3.26}$$

$$z_5(t) = \frac{1}{V}\left(T_0.q_0 - T.q\right), \ \mathbf{z}_6(t) = \frac{-\alpha_d.\Delta H.A_r.K.A}{\rho.C_p.T_r} \tag{3.27}$$

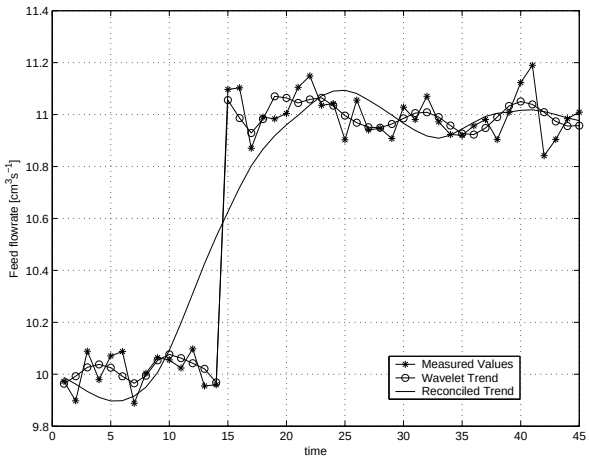


Figure 3.7: Performance of the EPA/wavelet for a step changes.

and

$$\mathbf{z}_7(t) = \frac{-U.A_R}{\rho.C_p.V}.(T - T_c) \tag{3.28}$$

Therfore it is possible to convert the CSTR model governed by Eqs. (3.22-3.24) to Eq. (3.29) which is similar to the required form (Eq. (3.7)).

$$\begin{pmatrix} \frac{dV}{dt} \\ \frac{dA}{dt} \\ \frac{dT}{dt} \end{pmatrix} = \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix} \mathbf{z}(t) \tag{3.29}$$

It is important to note that a simultaneous reconciliation of hold-up, flow-rates, compositions and temperatures provides better estimation that a sequential application of DR technique. This result is supported by the fact that more is the functional redundancy better is the DR performance.

Considering the overall model described in Eq. (3.29) an open-loop response of the CSTR process is simulated. The behavior of reactant A decomposition is presented in Figure 3.8, for a ramp in the feed concentration A_0 with a slope of 0.05. In Figure 3.9 the estimation of reactor temperature is presented.

The approach presented for the nonlinear case provides promising results in a simple way showing a direct advantage over the other approaches mainly in terms of computation requirements.

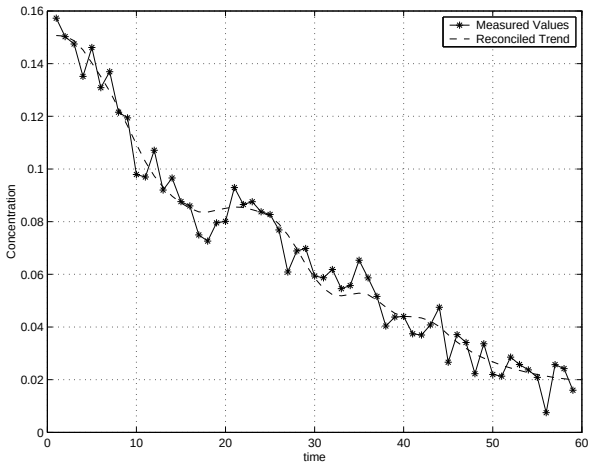


Figure 3.8: Reconciling the decomposition of reactant *A* in the CSTR reactor.

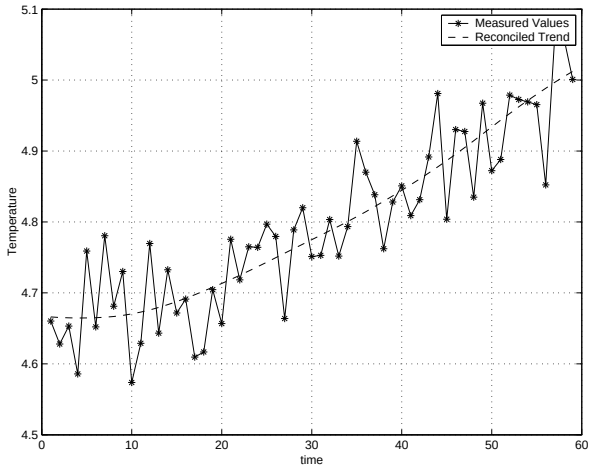


Figure 3.9: Reconciling the temperature T of the CSTR reactor.

3.4.4 Sampling time - CSTR reactor

For the approach proposed for handling the situations where different sampling time are present, it has been assumed that the sampling time of the level-meters is five times greater than the flow-meters used for sensing the flows in the CSTR case presented above.

Additionally, the time-horizon is increased until 112 and the moving time window selected is equal to 8 time intervals. The behavior of the flow-rates and the reactor hold-up are shown in Figures 3.10, 3.11 and 3.12. It can be seen that the data reconciliation estimation is highly acceptable.

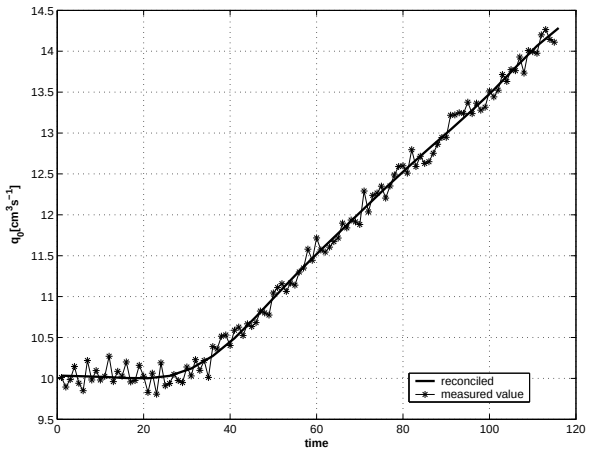


Figure 3.10: Reconciled q_0 with distinct sampling times using EPA/wavelet.

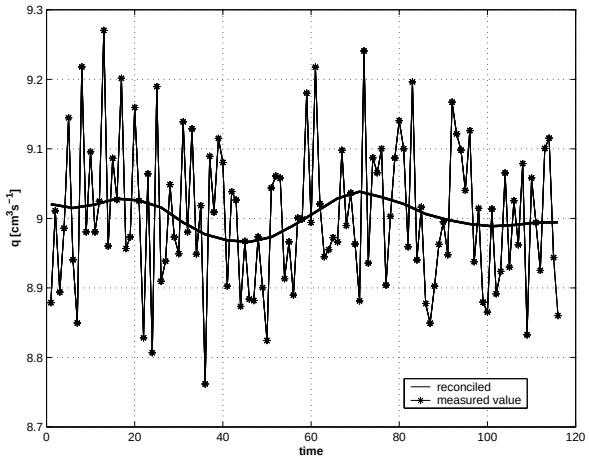


Figure 3.11: Reconciled q with distinct sampling times using EPA/wavelet.

Additionally, in Figure 3.12, it can be seen that the reconstruction of the hold-up is well performed, demonstrating the applicability of the proposal.

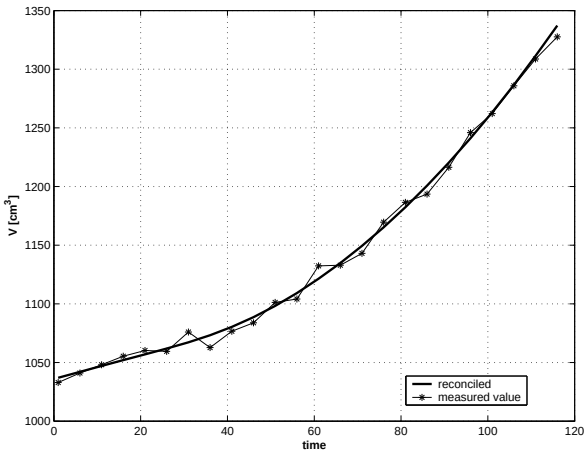


Figure 3.12: Reconciled V with distinct sampling times using EPA/wavelet.

3.5 Conclusions

This chapter presents a dynamic Data Reconciliation technique based on wavelet analysis. Firstly, wavelet is used to catch the deterministic trend of sensor data. Then, these trends are rendered consistent with the process model by optimizing the polynomial coefficients that fit these trends. Therefore, estimations take advantage of both temporal and functional redundancies. The approach proposed in this chapter can be applied on-line as well as off-line in an efficient way, presenting an accurate estimation. Another feature of the proposal is its capability to deal with measurements with different sampling frequency in contrast to the current DDR techniques.

The extension of the proposal to deal with the nonlinear cases has also been presented demonstrating promising results. That is, the way the nonlinearity is undertaken in this chapter is relatively efficient, useful and involves simple and fast calculations.

Nomenclature

$\mathbf{A}$	incidence matrix with rows corresponding to nodes and columns to variables
A_l	approximation signal component at scale l
da_l	difference between raw and reconstructed signal at scale l
$\mathbf{f}()$	linear differential equation constraints set
$\mathbf{h}()$	linear algebraic equality constraints set
$\mathbf{g}()$	inequality constraints set including simple upper and lower limits
L	a certain scale
L_m	optimal dyadic scale for wavelet decomposition
$N(\mathbf{0}, \mathbf{Q})$	normal distribution with mean $\mathbf{0}$ and covariance matrix $\mathbf{Q}$
n	width of moving time-window
N	vanishing moment
$P_l(da)$	power contained in the da_l component at scale l
p_i	polynomial degree of variable $\mathbf{z}_i(t)$
p_j	polynomial degree of variable $\mathbf{x}_j(t)$
$\mathbf{Q}$	covariance matrix of measurement errors
$\mathbf{Q}_{\mathbf{y}(t_k)}$	covariance matrix of measurement errors $\mathbf{y}(t_k)$
$\widehat{\mathbf{Q}}$	covariance matrix of reconciled process variables
t	time
T_s^{max}	least frequency of measured variables
$u_{l,v}$	scaling coefficient at dyadic scale l and dilation v
$w_{l,v}$	wavelet coefficient at dyadic scale l and dilation v
$w^*_{l,v}$	wavelet coefficient at scale l and dilation v that remain after the de-noising
$\mathbf{y}$	vector of process variable measurement
$\widehat{\mathbf{y}}$	vector of reconciled (estimated) process variable
$\mathbf{y}^*$	vector of true process variables
$\mathbf{z}$	new variable obtained by handling the right hand side of process model equations

Greek letters

$\alpha_{i,k}$	k^{th} polynomial coefficient of process variable $\mathbf{z}_i$
α_d	catalyst deactivation parameter
β	thresholding value
ε	error or noise associated to the measured signal
$\phi_{l,v}$	scaling coefficient at dyadic scale l and dilated index v
$\psi_{l,v}$	wavelet coefficient at dyadic scale l and dilated index v

σ standard deviation
$\sigma_{\mathbf{y}(t_k)}$ standard deviation of variable $\mathbf{y}(t)$
$\sigma_{\mathbf{z}(t_k)}$ standard deviation of variable $\mathbf{z}(t)$
$\omega_{j,k}$ k^{th} polynomial coefficient of derivative terms $\mathbf{x}_j$
Ω matrix of $\omega_{j,k}$
Λ matrix of $\alpha_{i,k}$

Subscripts

c current time
i process variable
j derivative term
k sampling time
l dyadic scale
v dilated coefficient

Chapter 4

Data Reconciliation and time-delay estimation

Abstract

The presence of time-delay among measured process variables is frequent in operating plant data due to several causes (e.g. sensor response delay). Thus, reconciling these measurements leads to poor estimation due to the mismatch between the process model and these measurements. Precisely, the problem is that the process model is not adequate since it does not consider the time-delay effects. Therefore, it is essential for a correct performance of data reconciliation to estimate the time-delay parameters. These time-delays parameters can be estimated by searching the maximum correlation among the process variables. The resulting Time-Delay Vector **(TD)** is then integrated with the DDR technique through the measurements model.

In this chapter an efficient Time-Delay Estimation (TDE) procedure based on GA optimization is presented for determining all the existing delays with respect to a reference process signal. That is all the delays are identified in terms of their location, and time-delay parameter value. Additionally, the maximum correlation is tackled from an eigenanalysis perspective. By this way it is possible to determine also the direction of process variability.

It has been found that DDR and TDE are combined in a straightforward way that allows to overcome data/model delay mismatch, this leads to an better estimation.

This novel approach[1] can be applied off-line and on-line as well as for a time-varying or constant time-delay parameter, leading to accurate estimation in the presence of systematic errors. The proposal has been validated using a process operating

[1]A part of this work was proposed in a work-package of the **GICASA-D** Catalan project number I0353, financed by "Generalitat de Catalunya".

at dynamic mode with recycles.

4.1 Introduction

As discussed in previous chapters Data Reconciliation is a model-based filtering technique that attempts to reduce the inconsistency between measured process variables and the corresponding process model. The measurement error, model inaccuracy and so on generate a pseudo-random noise; this noise is superimposed to the real variable values. Thus, by decreasing this inconsistency, more precise variable estimation values are obtained and then they can be used satisfactorily in process control and optimization systems.

Nevertheless, the performance of DR can be seriously affected in the presence of any event (gross error) that increases this inconsistency such as: modeling errors and/or the presence of delays in sampling data. In such situations an error in the DR can be generated by forcing a matching of the process model and data that are in principal incompatible.

To deal with the presence of gross-error(s) several approaches have been presented as discussed in the introduction of this thesis covering steady and dynamic systems as well as different kinds of errors (e.g. bias, process leak). However, less has been developed to handle situations in which delays are present.

There are many applications where several variables are measured and it is known or suspected that they are related with an unknown time lag. The presence of these lags makes highly inconsistent the measurement and their corresponding model. A good estimate of this lag is then essential to match the process model and process data.

Most traditional methods for extracting the delay between variables are based on cross-correlation in the time and frequency domains. This correlation measures can be used to determine delays by applying relative shifts between signals within a process data matrix. Thus, it is possible to find out the position in which the correlation among variables is maximal as proposed in Wachs and Lewin (1999). The optimal shifts are those that minimize the determinant of the associated correlation matrix.

When the number of measured variables is large and/or the maximum relative shift among all variables d_{max} is high in terms of sampling time, exhaustive search of the maximum correlation by direct data matrix manipulation is infeasible due to the high combinatorial size. Wachs and Lewin (1999) realized this problem and proposed an algorithm that reduces the problem size by assuming that:

1. a distinction between input and output variables is possible,

2. output variables are correlated among themselves with no delays present, and

3. inputs variables are independent.

However, these assumptions are not realistic or at least can only deal with a very reduced number of situations, since it is not evident to separate between input and output variables mainly if recycles are present in the process. Furthermore, delays can be present even in output variables. For instance the response of a composition analyzer sensor and a flow-meter both placed in an output variable may be delayed.

4.2 Data Reconciliation

DR is a technique that takes advantage of the redundancy present between the process model and the measurements model. Therefore, the existence of both models is a pre-requisite to reconcile redundant measured data. The measurements model is constructed by assigning to each one of the m redundant measured process variables a normal probability distribution around its true value.

$$\mathbf{y}(t_k) = \mathbf{y}^*(t_k) + \varepsilon\left(t_k\right), \quad \mathbf{y}(t_k) \in R^m \tag{4.1}$$

where $\mathbf{y}$ is the $m \times 1$ discrete measurements vector, $\mathbf{y}^*$ is the $m \times 1$ vector of unknown true values, and ε stands for the $m \times 1$ vector of random measurement errors whose expected value is the null vector and has a certain variance matrix (measurements are assumed as independent). Additional information must be introduced through the process model equations (constraint equations). The process model is in general represented by a set of differential, algebraic and inequality constraints see Eqs. (3.2), (3.3) and (3.4).

Different approaches have been proposed to solve this dynamic optimization problem however Kalman filter technique can be efficiently used since this technique it provides robustness against modeling errors by assuming uncertainties in the process model see Eq. (1.26).

4.3 The Role of time-delays in Data Reconciliation

4.3.1 Time-delay estimation

Time-delays are present in almost all the industrial processes. They affect the performance of control, monitoring and DR systems. Nevertheless, most of these techniques do not take into account delays or simply consider them as known and constant throughout the time.

Two types of time-delays can be distinguished: Process-Related Delays (PRD) are caused by the intrinsic process dynamics, the control algorithm, etc. Generally

these delays are propagated to the subsequent part of the process under consideration. These situation causes a decrease in the DR performance. Whereas Sensor-Related Delays (SRD) such as: the placement of the sensors in the plant (e.g. thermocouple or flow-meter in a pipeline) present a local effect. Nevertheless, from DR perspective, these "local" delays will certainly induce a biased estimation in the rest of the involved redundant process variables (*smearing*).

As it was discussed earlier, the approach presented in Wachs and Lewin (1999) assumes that the output variables are correlated among themselves with no delays present, and that the inputs are independent. In real processes it is difficult to distinguish between input and output variables, since this distinction is process unit dependent (i.e. a variable may be at the same time an output of a process unit and the input of the subsequent unit). These distinctions are even more difficult to be realistic if recycles are presented which is often the case.

This work presents a TDE method that drops all the distinctions discussed above. The proposal is able to deal with real situations without requiring any a priori knowledge about the number, location and size of time-delays in the process. Furthermore, the way of evaluating the correlation is slightly modified as it will be explained in the next section. By adopting such generic approach in processes with an elevated number of variables, may increase considerably the computational effort. With the purpose of reducing this effort a genetic algorithm based optimization is developed.

Despite the characteristics of the stochastic optimization used, the optimum value for the objective function is not ensured. However, a satisfactory solution can be obtained in a short time, which is required for on-line applications.

4.3.2 Genetic algorithm vs. TDE

4.3.2.1 Genetic algorithm formalism

A Genetic Algorithm (GA) (Holland (1975); Goldberg (1989); Musulin et al. (2003); Heyen et al. (2002)) technique has been chosen to solve this high combinatorial optimization problem.

GA constitutes a vast class of stochastic algorithms used to solve optimization problems (Goldberg (1989); Holland (1975)). In comparison with other stochastic procedures for optimization GAs consider many points in the search space simultaneously and therefore have a reduced chance of converging to local optima as demonstrated in Karr (1993). In the classical GA formalism (Figure 4.1), a set of N_{ind} potential solutions (population) is generated randomly. Each potential solution (individual) in the multi-dimensional search space is coded as a vector called chromosome (which consists of a string of genes, each representing a feature).

The goodness of each individual in the population is evaluated by utilizing a pre-

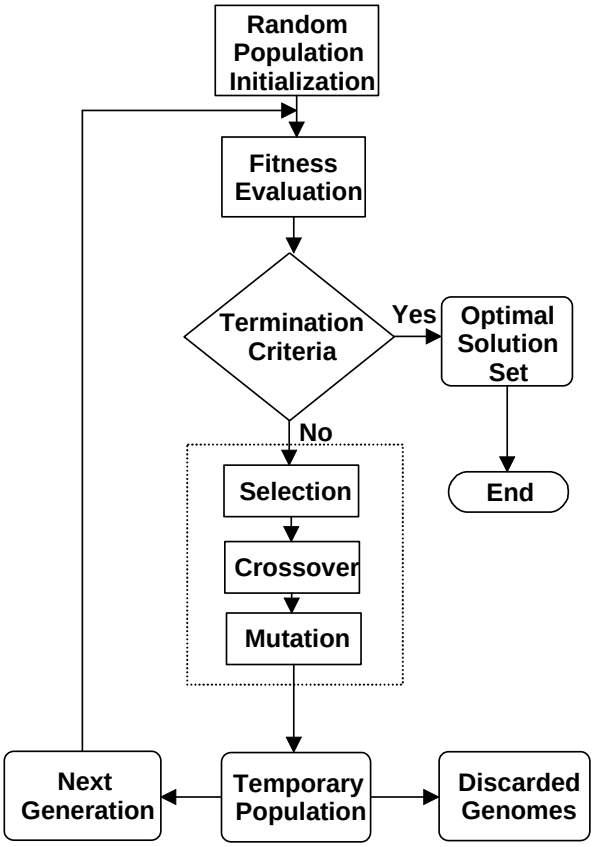


Figure 4.1: Genetic algorithm loop.

specified fitness criterion ϖ. Upon assessing fitness of all the chromosomes in the population, a new generation of individuals is created from the current population by using the selection, crossover and mutation operators.

- The **selection** operator selects pairs of individuals with a view to create a mating pool for reproducing offspring. The operator used in this proposal is the roulette wheel. This operator sets the probability of an individual to be crossed proportionally to its fitness. Half of the individuals ($N_{ind}/2$) are selected in each generation to be crossed.

- The **crossover** operator comprises the basic mechanism for redistributing genetic characteristics between the individuals. It is applied to some pairs of the selected individuals (parents) and creates pairs of new individuals. The selec-

tion is made randomly with a pre-defined crossover probability (P_c). There are several methods in which the parent chromosomes can be combined to generate offspring. The crossover operation used in this work is the so-called: two point crossover.

- The **mutation** operator consists on changing randomly the value of a gen of the individual. This operator introduces the artificial concept of copying error. Its objective is to recover good genetic material that could be lost in the crossover operation and avoid local minima by searching arbitrary points of the solution space. The mutation probability P_m is usually much lower than the crossover probability, in this case it is selected to be $P_m = P_c/N_{ind}$.

Once the mutation has been applied, a new set of $N_{ind}/2$ individuals is available. This new set is then merged with the best fitted $N_{ind}/2$ individuals of the old generation to take a new generation of N_{ind} individuals.

- Finally, the algorithm is terminated using one of these stopping criteria: number of generations reaches a pre-defined maximum value (N_G) or current population does not give sufficient improvement compared with the performance reached a given number of generations before (G).

The MATLAB genetic algorithm Toolbox developed by the University of Sheffield has been used in the following case studies (Sheffield University (2003)).

4.3.2.2 TDE based on GAs

In this approach, each chromosome represents a time-delay vector as defined in Eq. (4.2). It contains the delays present in the process signals with respect to a reference signal. The value of each delay vector element represents a number of sample times, bounded by d_{max}.

$$\mathbf{TD} = [d_1, d_2, ..., d_m] \tag{4.2}$$

The fitness of each individual **TD** is evaluated considering the correlation among the process variables. Let us consider a data matrix:

$$\mathbf{Y} = \mathbf{y}_i(t_k),\ i = 1, ..., m;\ \ k = 1, ..., n \tag{4.3}$$

or

$$\mathbf{Y} = [\mathbf{y}_1(t_k), \mathbf{y}_2(t_k), ..., \mathbf{y}_m(t_k)];\ \ k = 1, ..., n \tag{4.4}$$

containing data corresponding to n samples of m process variables. $\mathbf{y}_i$ represent the k samples of the process variable i. For each variable i the error associated to $\mathbf{y}_i(t_k)$ is

supposed to follow a normal probability distribution. Vector variable $\mathbf{y}_i(t_k)$ must be normalized, let $\mathbf{Y}^{norm}$ be the corresponding normalized data matrix.

Introducing **TD** into $\mathbf{Y}^{norm}$ allows to obtain a new delay adjusted data matrix as:

$$\mathbf{Y}^{norm,corr} = [\mathbf{y}_1^{norm}(t_k - d_1), \mathbf{y}_2^{norm}(t_k - d_2), ..., \mathbf{y}_m^{norm}(t_k - d_m)] \tag{4.5}$$

The corresponding correlation matrix **R** of the redundant measured data is computed as:

$$\mathbf{R} = \frac{\mathbf{Y}^{norm,corr}.(\mathbf{Y}^{norm,corr})^T}{n-1} \tag{4.6}$$

The determinant of the correlation matrix will equal 1.0 only if all correlations equal 0 otherwise the determinant will be less than 1. Therefore, by computing the determinant of **R** it is possible to evaluate the correlation among variables. This result can also be obtained by an eigenanalysis, where each eigenvalue explains or contains the variance in the direction given by the corresponding eigenvector. This similitude between the determinant and the eigenvalue is also confirmed by the property that the continued product of eigenvalues of **R** is equal to the determinant of the correlation matrix. Additionally, by maximizing the largest eigenvalue $\xi(\mathbf{R})$ the correlation is maximized, since all the process variance will be contained in the corresponding direction and the other eigenvalues will have a very small value. Thus, the norm of **R** is calculated as its $\xi(\mathbf{R})$ value. Finally, the fitness of **TD** is considered to be proportional to $\xi(\mathbf{R})$.

It is important to note that it is helpful to scale the process variable measurements in order that they have zero means and unit variances.

4.3.3 TDE proposal validation

To demonstrate the performance of the proposed algorithm an illustrative example is presented in Figure 4.2. It consists of a tank provided with a heating system. The evolution of valve command (V_c), steam flow through the valve (F_v) and the temperature of the tank (T) heated by means of the steam is illustrated in Figure 4.3.

Two hypothetic delays are introduced to this unit process as shown in Figure 4.3: d_2 corresponds to the valve response that is delayed from its command signal (V_c) and d_3 corresponds to the controlled variable T that is delayed from the manipulated variable F_v (i.e. jacket heating).

The reference signal selected in this case has been the most delayed one, the temperature T. This reference selection is convenient for off-line application since all the signals are available when T is available. If d_2 represents the delay between F_v and T and d_3 represents the delay between V_c and T the delay vector is then equal to: $[0, d_2, d_3]$. The number of generations (N_G), mutation and crossover probabilities (P_m

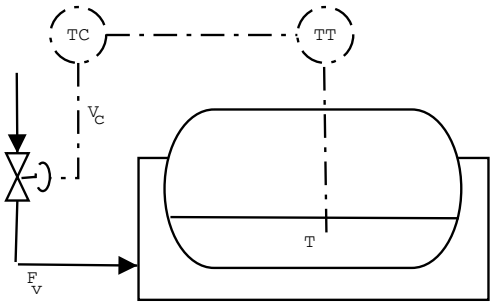


Figure 4.2: Illustrative example for TDE procedure.

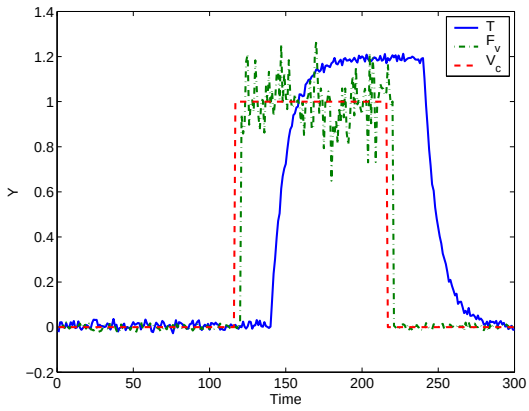


Figure 4.3: Effect of time-delay on process variable behavior.

and P_c respectively) values used in this example are listed in Table 4.1. Finally, the maximum delay between variables is selected to be $d_{max} = 32$. This value is selected by an off-line analysis of the whole process time response.

Table 4.1: GA parameters of TDE used for the illustrative example.

N_{ind}	N_G	P_c	P_m
50	10	0.7	P_c/N_G

Using the GA tuning specified in Table 4.1, the algorithm converges quickly to the

best individual ($\mathbf{TD} = [0, 26, 29]$) in few generations. This **TD** is the same vector as the one obtained by means of an exhaustive search. Nevertheless, a slight difference with the true delay ($[0, 20, 24]$) has been observed due to the process dynamics.

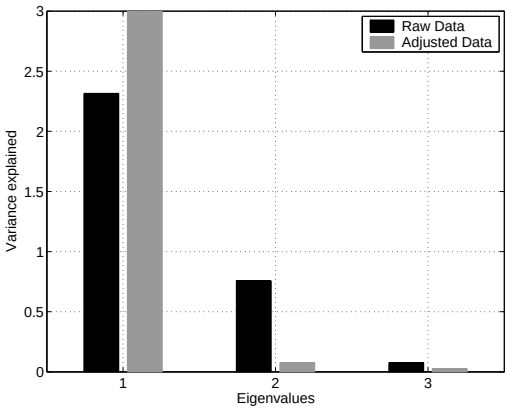


Figure 4.4: Variance explained in the eigenvectors of the correlation matrix.

In Figure 4.4 and Table 4.2 the eigenvalues of the correlation data matrix of the illustrative example are presented. It can be seen how after the delay adjustment, the system variance is concentrated in the first eigenvector (in other words, the process variance is explained in a unique direction). Therefore, by maximizing the first eigenvector, the process correlation is also maximized. From Figure 4.4, it can be noted that the system presents two directions, since there is an evident collinearity between the behavior of V_c and F_v. However, the proposal given in this work also detects that temperature behavior follows the same direction which reduces the system variability to a unique direction represented by the first eigenvalue 98.08%.

Table 4.2: Eigenvalues of the correlation matrix, illustrative example.

Eigenvalues	1^{st}	2^{nd}	3^{rd}
Raw data	2.39	0.54	0.07
Delay adjusted data	3.08	0.04	0.02

Correcting or adjusting the data matrix $\mathbf{Y}^{norm}$ by the use of the best time-delay vector (**TD**) obtained leads to more consistent data ($\mathbf{Y}^{norm,corr}$) as it is shown in Figure 4.5. Thus, delays have been correctly estimated and their effect has been compen-

sated. It is important to note that the combined effect of all process delays is globally compensated.

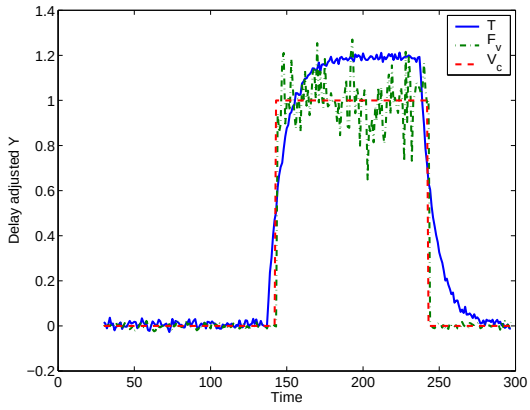


Figure 4.5: Illustrative example, adjusted process data.

4.4 Integrating TDE into DR

4.4.1 Constant time-delay parameter

Integration of the time-delay estimation procedure and DDR can be achieved by updating the measurement model as follows:

$$\mathbf{y}_i(t_k) = \mathbf{y}_i^*(t_k - d_i) + \varepsilon(t_k - d_i), \quad \mathbf{y}(t_k) \in R^m \tag{4.7}$$

In the case of sensor-related delays, DDR can be performed using the process model and modifying the measurement model according to Eq. (4.7). In the case of process-related delays, the reconciliation is also improved by incorporating the TDE method proposed. However, the effect of delays in the process might not be completely compensated. Anyway, the performance of the system is highly enhanced.

4.4.2 Time varying of time-delay parameter

In previous sections, time-delays have been assumed as constant. However they can change throughout the time. Then an extension of the proposed methodology is presented in order to deal with these situations, see Figure 4.6.

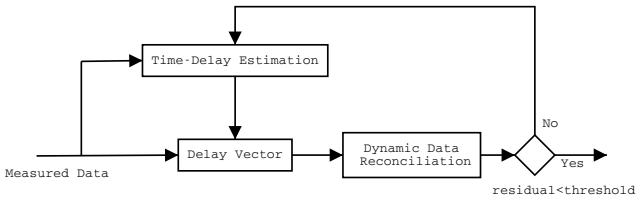


Figure 4.6: Adaptive time-delay estimation procedure.

Initially, **TD** is estimated using the proposed time-delay estimation method. Then. by analyzing the residuals according to a threshold value, the **TD** is maintained to perform DDR or updated. That is, if the threshold is exceeded, the TDE is activated in order to re-calculate **TD**.

The residuals can be evaluated based on the differences between: 1) process measurements and model output, or 2) process measurements and reconciled process.

Seeking to reduce computation time of TDE, the current **TD** vector might be included in the initial GA population of the new delay estimation.

4.5 Case study: results and discussion

The proposed methodology can be applied both for process-related delay (PRD) and for sensor-related delay (SRD). However, a case study presenting PRDs is selected, since it is more complex in terms of generated nonlinearities and delay propagation effects.

Figure 4.7, shows a flow-sheet of the process used to evaluate the proposed DDR methodology. The situation, also used in other sections of this thesis work is taken from the work presented in Darouach and Zasadzinski (1991), and consists of eight streams and four storage tanks.

In this example, the flow-rates and mass hold-ups are the process variables to be monitored.

A disturbance, due to a sudden but lasting change, causes a modification in the operation conditions: the flow Q_1 suddenly rises off by a value of 5 to reach the value of 34.5 at the sampling time 250, maintaining this value during 250 sample times. Then, it returns to drop to its original value of 29.5 (see continuous line in Figure 4.9).

First, the DDR has been applied for estimating flow and hold-up process variables of this case study assuming that no delay will be present. The performance of such application can be seen for the flow Q_3 in Figure 4.8, showing some ability to reduce

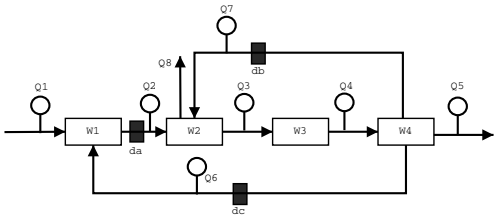


Figure 4.7: Case study for integrating DDR and TDE.

the measurement noise and a notable estimation accuracy. The continuous line shows the "unknown" true data, the points shows the measurements and the dashed line represent the reconciled data.

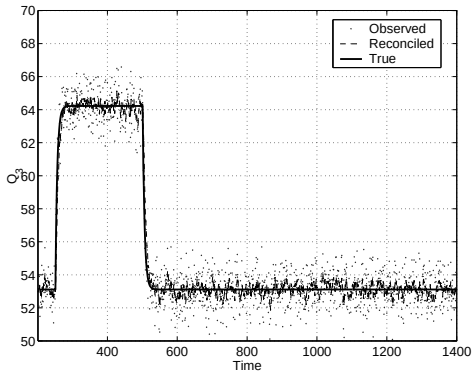


Figure 4.8: Reconciled Q_3 flow assuming that no delay will be present.

This example has been modified by adding three hypothetic delays ($[d_a, d_b, d_c]$ = $[200, 100, 10]$ sample time units) at the positions shown in Figure 4.7. The combined delay effect leads to a considerable discrepancy between the mass balance and the measured values especially around tank W_1, showing unacceptable results and the presence of bias as it can be seen in the left plots of Figures 4.9, 4.10 and 4.12 and 4.13.

Generally, by using the combined methodology (TDE-DDR) proposed, the performance of the reconciliation is considerably enhanced, as can be seen in the right plots of Figures 4.9, 4.10, 4.12 and 4.13.

In the case of incorporating TDE, data reconciliation can also perform correctly when the nonlinearities introduced by the delay effect are present as seen in Figure

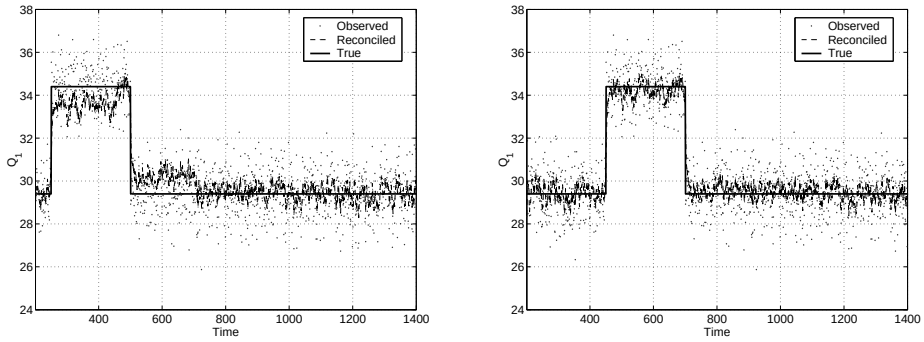


Figure 4.9: Reconciled Q_1 values with delays and with adjusted delays.

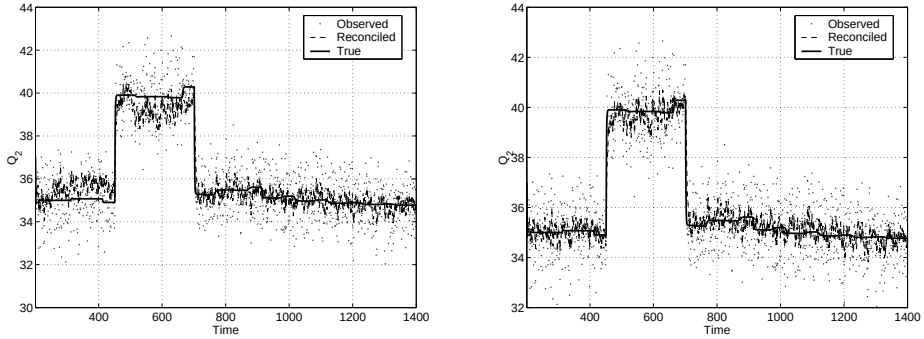


Figure 4.10: Reconciled Q_2 values with delays and with adjusted delays.

4.11.
The parameters of the optimization algorithm used are the same that were used for the illustrative example of section 4.3.3 (Table 4.1) for a d_{max} value of 256 and the best individual found is:

$$\mathbf{TD} = [219, 188, 172, 173, 173, 177, 0, 214, 207, 173, 172, 173] \tag{4.8}$$

The three cases DR without delays, with delays and with adjusted delay (integrated with the TDE) are represented respectively in the three rows (II, III and IV) of Table 4.3. The row I represents the measurement error variance assumed for each measured process variable. It can be noted that the hypothetic delays affect mainly the hold-

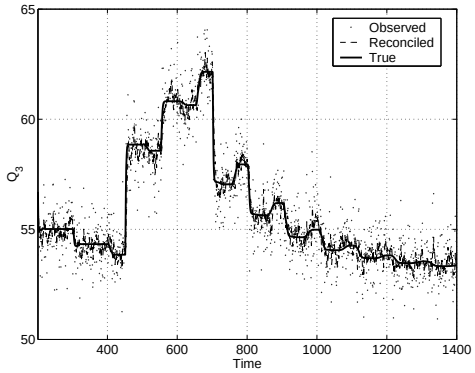


Figure 4.11: Reconciled Q_3 flow values with delays and with adjusted delays.

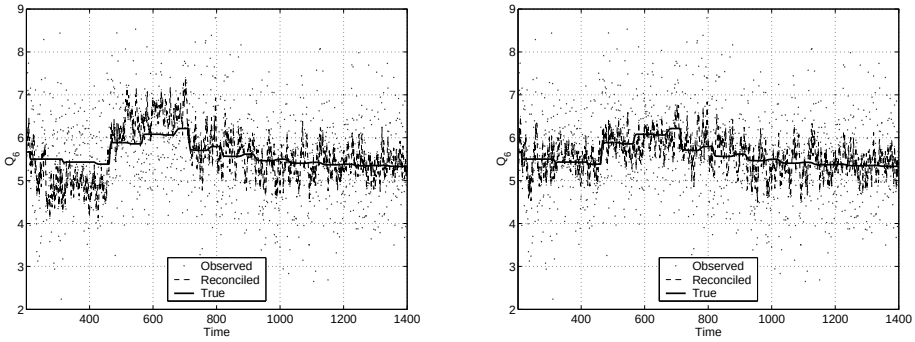


Figure 4.12: Reconciled Q_6 values with delays and with adjusted delays.

up of tank W_1 and a notable bias can be observed when reconciling these delays measured data. However, by integrating the TDE method it can be observed a highly improvement in the DR in all the reconciled variable.

4.6 Conclusions

In most cases of dynamic processes, where some sources of time-delay between causes and effect appear in the system, the performance of DDR can be highly improved by means of the integration with TDE in both on-line and off-line cases. One advantage of the presented TDE method is that no a priori knowledge is necessary

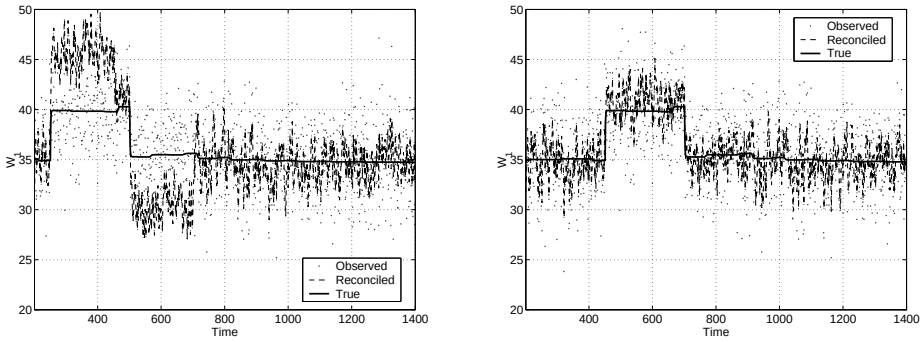


Figure 4.13: Reconciled W_1 values with delays and with adjusted delays.

for the number and location of time-delays in the process, which is in advantage of the application simplicity. On the other hand, although this approach shows good results for the case of process-related delay, it is specially suited for sensor-related delay since it deals with the measurements model. An extension of this approach to time-varying delays has been sketched.

Finally, it has been found that DDR and TDE are combined in a straightforward way that allows to overcome data/model delay mismatch, this leads to an better estimation.

Table 4.3: Effect of delays on DR performance and the role of TDE.

	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7	Q_8	W_1	W_2	W_3	W_4
I	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	10.0	10.0	10.0	10.0
II	0.21	0.21	0.28	0.25	0.21	0.11	0.14	0.12	**3.11**	3.41	3.10	3.03
III	0.58	0.39	0.17	0.18	0.15	0.28	0.16	0.15	**20.02**	3.88	2.93	4.26
IV	0.20	0.20	0.18	0.14	0.15	0.12	0.14	0.19	**3.27**	5.16	2.69	2.71

Nomenclature

d_i	delay of process variable i to the reference variable
d_{max}	maximum relative shift among all variables
m	number of redundant measured process variables
n	number of samples
N_{ind}	number of individuals in each population
N_G	number of generations
P_c	crossover probability
P_m	mutation probability
$\mathbf{Q}$	variance-covariance matrix
$\mathbf{R}$	correlation matrix
t	time
$\mathbf{TD}$	time-delay vector
$\mathbf{Y}$	process data matrix
$\mathbf{Y}^{norm}$	normalized process data matrix
$\mathbf{Y}^{norm,corr}$	corrected (delay adjusted) normalized process data matrix
$\mathbf{y}_i$	process variable vector of i
$\mathbf{y}_i^{norm}$	normalized process variable vector of i
$\widehat{\mathbf{y}}_i$	estimated measurement vector of i
$\mathbf{y}_i^*$	unknown true process variable values of i

Greek letters

ε	random measurement errors
$\xi(\cdot)$	maximum eigenvalue operator
$\widehat{\theta}$	vector of adjusted model parameters
ϖ	fitness function
σ_i	standard deviation of the i^{th} variable

Subscripts

i	process variable
k	sample time

Chapter 5

An open software architecture for Data Reconciliation

Abstract

This chapter proposes a flexible and open architecture design for Parameter Estimation and Data Reconciliation (PEDR) software application. This application is de-coupled according to the functions involved. Then, the application's components and their interactions are specified and validated.

The designed architecture aims to improve the efficiency of the PEDR application by allowing the ex-changeability and connectivity of the components in more efficient and consistent way. These properties are enhanced by the adoption of the CAPE-OPEN standards for the implementation of the required component, which allow the use of optimizers and process models developed under the distributed computing paradigm.

This work adheres to current trends moving towards open systems architectures for process engineering software, this one in the context of Parameter Estimation and Data Reconciliation.[1]

5.1 Introduction

Accurate process measurements are essential to the performance of process system engineering, from modeling to monitoring and optimization. Once gross errors are

[1]This work was proposed in the **Global CAPE-OPEN (GCO)** project funded by the European Community under the Industrial and Materials Technologies Programme (Brite-EuRamIII) under contracts IMS 26691.

filtered, accurate process data may be obtained using Data Reconciliation, a technique that fit measured data to process model by minimizing measurement errors and satisfying model constraints. The same procedure may lead to model adjustment by determining the best model parameter values matching a set of accurate data. Thus, Parameter Estimation (PE) and Data Reconciliation are quite similar in the sense that in general both techniques involve adjustment by constrained optimization.

Moreover, their performance require the same type of elements: mathematical models of the process or processes, sets of measured (or reconciled) process variable and an Optimizer. Therefore, a generic Parameter Estimation and Data Reconciliation application can be conceived to deal either with DR, PE or both problems. Being an aggregation of different techniques, a PEDR application should not be conceived as a monolithic application as the one illustrated in Figure 5.1.

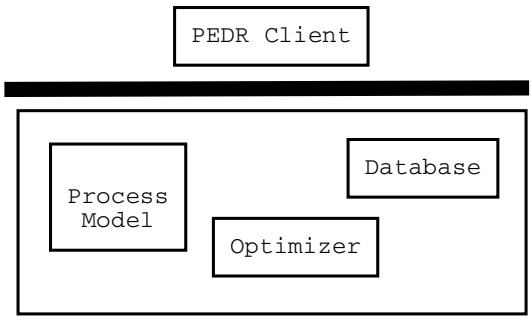


Figure 5.1: Rigid or monolithic PEDR structure

The performance of the whole application or any of its functions can be seriously affected by changing or replacing any particular function. This could be a serious drawback for taking full advantage of the growing number of Computer Aided Process Engineering (CAPE) tools, such as equation-based simulators, optimization algorithms, databases, distributed control systems and so on.

Additionally, the development of a PEDR application that assembles these required CAPE tools faces different challenges. First, software components are generally provided by different suppliers, which involve a certain degree of software and hardware heterogeneity. Second, the existing software components are usually available as black box systems (closed source code) so their integration in an easy and secure way with each other and to any external application (e.g. real-time optimization) may be arduous.

An efficient solution to overcome these drawbacks and to facilitate maintenance and long term further development, is the use of modern component software technology and to standardize PEDR communication with its client applications in a con-

sistent, efficient and secure way by designing a well-defined interfaces that ensures inter-operability and transparency. Moreover, in order to provide a flexible design, all the PEDR components have to be de-coupled as much as possible.

Both the DR and PE can be formulated by an equation oriented simulator with optimization capability, but further efforts are required to use other kind of simulators or optimizers (from other software vendors). Furthermore, this option sets the process model, built in the simulator, as the central point of the problem, instead of leaving it as a inter-changeable element of the general PEDR problem. Thus, a significant advantage over existing commercial software tools may be obtained through standardization.

This chapter introduces an open modular architecture for a PEDR software executive allowing the management of different sets of process data, process models and numerical Optimizers through standardized interfaces. This leads to the exchangeability of the corresponding software components under the plug and play concept. Additionally, the system can also be upgraded by plugging into it new optimization algorithms, models, when available. Finally, such a system provides high flexibility for handling different plant structures and situations by matching various pre-defined or newly introduced process models with the corresponding data sets available.

Case studies have been proposed to test the consistency of the information flows between components and validity of the specified interfaces. The methodology has also been applied to the Data Reconciliation and Parameter Estimation problem in scenarios based on an existing pilot plant.

5.2 PEDR system's architecture

5.2.1 Multi-module vs. monolithic applications

The performance of numerical Optimizers and process modeling tools has a direct impact on the results of PEDR. The continuous improvement of such tools requires a flexible and modular structure for PEDR system so that its upgrading may be easily achieved by replacing old modules under the structure proposed in Figure 5.2, where each bold black bar represents the component's interface. The incorporation and integration of these modules in a rigid structure, and the fact that the modules could be provided from different providers and could have possible software and hardware incompatibility, leads to a significant implementation cost. These drawbacks may be overcome by using separate components that inter-operate through a well-specified interface (see Figure 5.2).

The use of standard interfaces and open communication between software components has emerged as a promising solution to software application incompatibilities

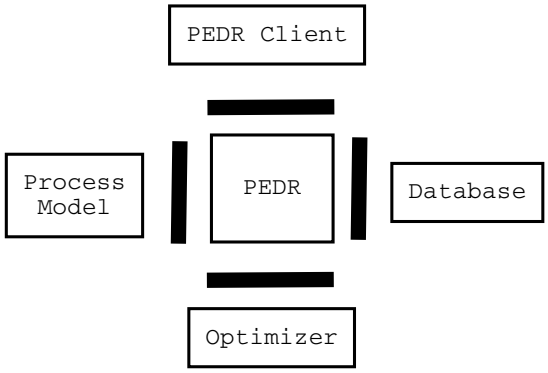


Figure 5.2: Open and flexible PEDR structure.

(CAPE-OPEN Standard 1.1 (2002)) in the CAPE field. For instance Fritz and Engell (1997) proposed an open and flexible architecture for a batch process simulation. Kakhu et al. (1998) designed an open and flexible architecture for process modeling and model-based applications. Benqlilou et al. (2001a) have proposed a distributed architecture for model updating in order to improve Data Reconciliation techniques. Modularity and distributed computing lead to interface standardization, such as the CO standard (CAPE-OPEN Standard 1.1 (2002)) for the CAPE field. Currently, different vendors and industrial companies are already incorporating the CO standard (e.g. Aspen Plus).

5.2.2 Open architecture design

The PEDR system can be de-coupled into four main functions: PEDR Manager, Process Model, Database and an Optimizer as shown in Figure 5.3.

1. The main purpose of the PEDR Manager module is to gain access to the properties of measured data (e.g. sample time values) and to the process model description. By combining this information, a PEDR optimization problem is formulated. This resulting problem will interact with the Optimizer to adjust redundant variables or estimate unknowns parameters and unmeasured but observable variables. The PEDR Manager is also responsible for objective function construction, variable classification and variance calculation.

2. The aim of the process model component is to generate a model description of the process under consideration. The fundamental building block employed for such purpose is an equation-based modeling system, which may be accessed

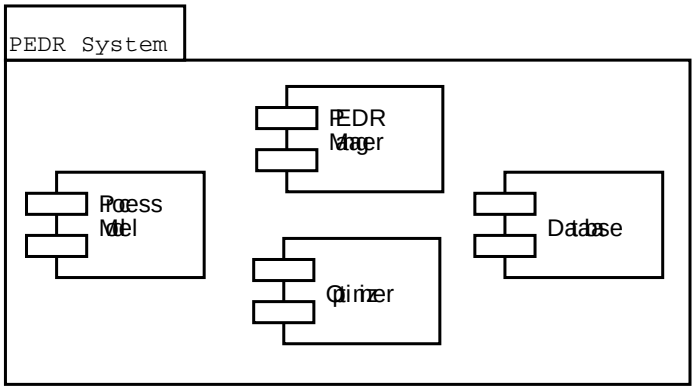


Figure 5.3: Proposed component diagram for the PEDR system.

via the Equation Set Object (ESO) interface defined in the CO standard. The CO standard also allows access to a simulated model through the Sequential-Modular Simulator Tool (SMST) as shown in Benqlilou et al. (2002d).

3. The process data module is responsible for acquiring measured process variable values from on-line as well as from off-line applications. Generally, this process variable data is intermediately stored in a relational database. Additionally, this module holds the data results obtained from PEDR execution.

4. The Optimizer component, is responsible for driving the solution of the optimization problem using all the information of the process model component and the constructed objective function. The Optimizer is able to gain access to the optimization model and run it, as many times as required. This is done by adjusting the model parameters, until the discrepancies between selected model outputs and the measurements is reduced to a minimum value of the objective function. Or to reconcile the redundant process variables until the inconsistency between selected process model and the corresponding measurements is reduced to a minimum, usually in a WLS sense. Interfacing between these modeling tools and optimization algorithms allows not only the implementation of Data Reconciliation and Parameter Estimation efficiently but also their incorporation in real-time applications.

Finally, the PEDR Client could be either an end-user or a process application that requires the system to perform a DR, PE or simultaneous PEDR. Set values to model parameters after DR in a sequential way may be inefficient. Thus, a simultaneous

strategy for PEDR is more efficient, that is minimize the measurement errors subject to model constraints and bounds as presented in Eqs. (3.2), (3.3) and (3.4).

The PEDR Client initiates the communication and interacts with the system to use its functions. Furthermore, it prepares the output generated by the PEDR Manager for easy use by the customer.

5.3 PEDR system's specification

5.3.1 Static view

The PEDR system proposed in this chapter is developed following the CO guidelines provided in CAPE-OPEN Standard 1.1 (2002). Therefore, for the sake of standardization the interfaces for communicating the PEDR Manager with the Optimizer and *process model* components may follow the CO specifications.

5.3.1.1 Process model interface

It is important that the *process model* interface covers the formulation of algebraic, differential, linear and nonlinear problems as well as inequality constraints. Therefore, the *process model* component has to encapsulate the general mathematical description of the physical system. A possible fundamental building block that can be employed for this purpose is a set of continuous equations representing a set of equations and variables. These are the equations that define the physical behavior of the considered process. The *process model* component is intended to serve the needs of PEDR and optimization procedures. Indeed PEDR will create this model and the Optimizer will solve it.

In order to get access to this *process model* component, the later has to expose the interfaces that allow clients to obtain information from it. This information is related to the size, structure of equations as well as current values, lower and upper bounds of process variables. The process modeling based on the ESO available in the CO standard is adopted for such purpose. The process model component can also involves a software component allowing to calculate current thermodynamic properties (enthalpy, viscosity, vapor pressure, LVE, etc) [2].

Additionally, considering that either DR or PE can be seen as optimization problems, an objective function also has to be constructed. This objective function is in general a WLS equation. Nevertheless, any other objective can be conceived and the same ESO can be used for constructing such a new objective function. This point

[2]This work was proposed in a **CO-LaN** project: "Development of CO-Tester for Thermo 1.1 plug in COM technology" under contracts RFB 006.

allows to deal with robust Data Reconciliation since a robust objective function as shown in Eq. (1.47) can be constructed as demonstrated in Arora and Biegler (2000).

5.3.1.2 Numerical Optimizer interface

The Optimizer specification contains three main interfaces. The *ICapeMINLPSolverManager* interface which exposes the methods for creating any kind of Optimizers for a specific optimization problem (either PE or DR, linear or nonlinear). The *ICapeMINLPSystem* interface is exposed by the Optimizer and *ICapeMINLP* interface allows the Optimizer to interact with the MINLP problem to be solved.

Benqlilou et al. (2002c,b) have wrapped successfully two optimization solvers the CPLEX from ILOG and the LP-solve Optimizers using the numerical Optimization specification presented in CAPE-OPEN Standard 1.1 (2002)[3].

5.3.1.3 Database interface

This is specific to the requirement of PEDR system and is not a general-purpose interface for process measurement data. The proposed interface, *ICapePEDRdatabase*, permits both the introduction and retrieve of data using *SetData()* and *GetData()* methods respectively. This data is related to process variable name, values, type and statistical error distribution (e.g. variance or covariance).

The process variable types could be either redundant, non-redundant, observable, or unobservable whereas the measurements are grouped into sets of experiments (corresponding to experiments on the plant in different operating conditions). The variables values can be singular or may include multiple values along a sampling time interval. Usually, steady-state Data Reconciliation handles a single data set at a time, while Parameter Estimation and dynamic Data Reconciliation usually handles several independent sets simultaneously.

5.3.1.4 PEDR Manager interface

Despite its internal modularity, the PEDR Manager has to expose a common interface, *ICapePEDRmanager*, (see Figure 5.4) to be used by any external Client.

Mainly, the interface exposes methods that provides information related to variable type, value and the assessing of the a posteriori variance matrix of adjusted process variables by means of *GetVariablesAttr()* method. The objective function values are also provided by using *GetObjFunAttr()* method. DR as well as PE involve the optimization of an objective function, thus, its construction is required with the use of

[3]This work was proposed in a **CO-LaN** project: "Development of CO-Tester for MINLP 1.0 in COM technology" under contracts RFB 004.

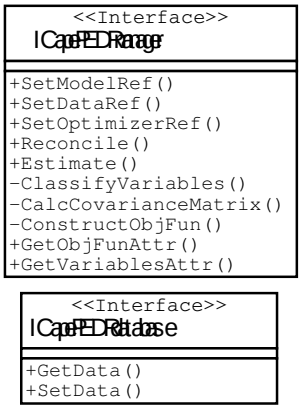


Figure 5.4: Proposed interface diagram for the PEDR system.

the *ConstructObjFun()* method. For example in the case of DR the OF is completely defined from the statistical error distribution thus if that information is not available form the database module it has to be estimated within the proper PEDR Manager using the internal method, *CalcCovarianceMatrix()*.

The optimization is re-started each time a new measurement set becomes available. Provision has to be taken for missing measurements or eliminated measurements with gross error(s). An analysis of the feasibility of estimation or reconciliation, including variable classification (not observable, just calculable, redundant) is also required through the specification of *ClassifyVariable()* method.

Additionally, the PEDR Manager also provides a graphic and user friendly interface, (see Figure 5.5), designed according to the methods that expose the PEDR Manager component.

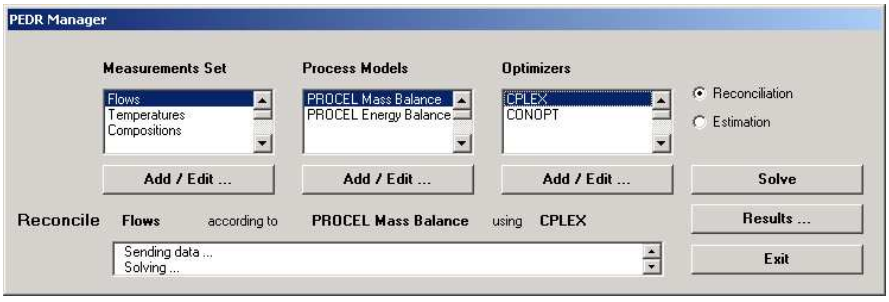


Figure 5.5: Graphical User Interface for PEDR Manager.

So the PEDR Client allows choosing to perform either a DR or a PE task then select the measured data to be reconciled or used for PE, then load the required process model to be used, and decide the appropriate numerical Optimizer for addressing the resulting optimization problem.

5.3.2 Dynamic view

First, Database, Optimizer and DR Manager are registered in order to make them accessible by the PEDR Clients (there can be more than one client, placed locally or even remotely). Once these servers are registered, the Client initiates the communication with the PEDR Manager through the specified interface *ICapePEDRmanager* by asking to reconcile/estimate. Then, the client supplies the references of the following information: process measurements (*SetDataRef()*), process models (*SetModelRef()*) and the appropriate Optimizer (*SetOptimizerRef()*).

The PEDR Manager accesses the *process model* information via the ESO, and process measurement via a Database standard interface, *ICapePEDRdatabase*, and generates an MINLP component exposing the *ICapeMINLP* interface. This created component interacts with the Optimizer to obtain the optimization results, which are then passed to the Client.

The sequence diagrams showing the temporal sequence of steps to be followed in order to perform a DR technique are presented in Figure 5.6.

5.4 PEDR system's validation

The specifications are written in a neutral Interface Definition Language (IDL) for CORBA IDL (Orfali and Harkly (1998)). This specification characterizes all the methods and input/output arguments that determine the entire PEDR specification. However, CORBA IDL is purely a declarative language, which has to be pre-compiled in a programming language in order to provide the code.

Java Development Kit (JDK) version 1.3 has been adopted for the implementation of all the specifications and all the methods within them, since CORBA technology is an integral part of the Java platform and managing the communication between CORBA objects is relatively easy.

The designed architecture, the specified interfaces and the standardization adopted allows total flexibility for constructing the prototype. This also covers the situations in which the involved software components reside in different platforms in a distributed system. Additionally, it also provides the additional benefit of sharing resources as the optimization component for example.

In order to demonstrate this flexibility and openness an absolutely CO distributed

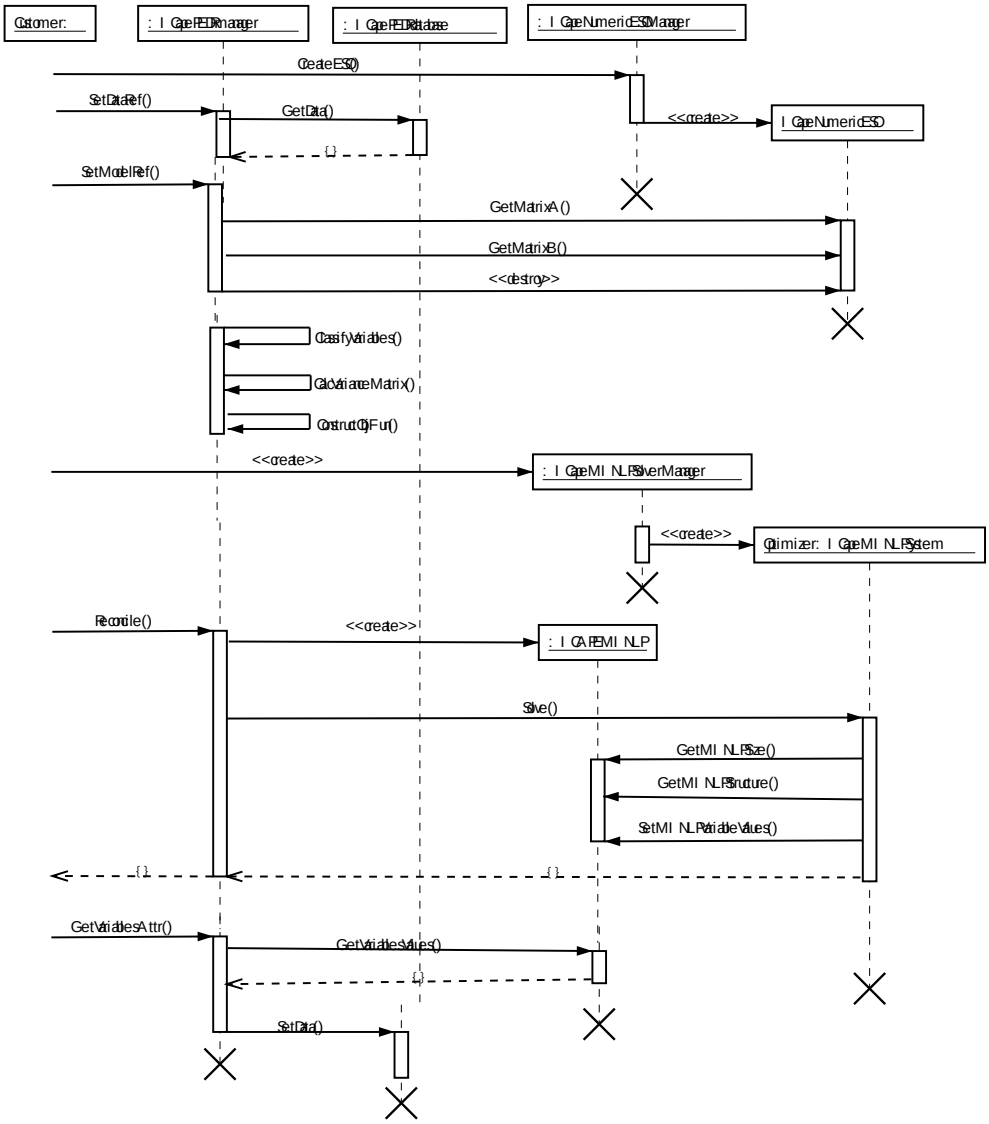


Figure 5.6: Proposed sequence diagrams for the PEDR system.

PEDR application has been implemented. The placement of the different software components of the prototype through the computer network (Optimizer, Database, PEDR Manager and PEDR Client) and their specific implementation is shown in the deployment diagram presented in Figure 5.7, with the corresponding Platforms,

operating system and network connections.

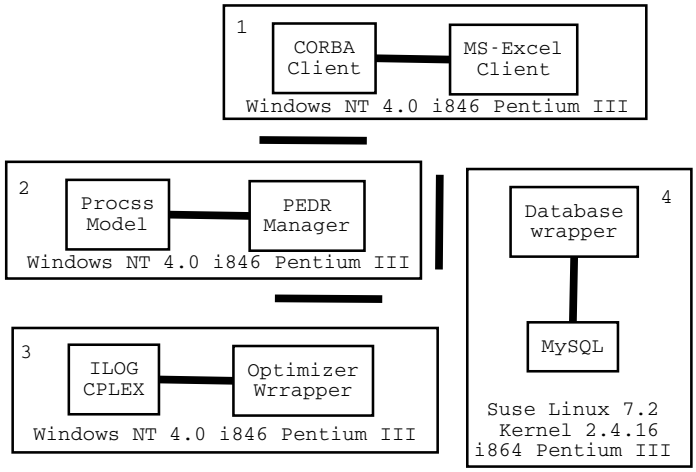


Figure 5.7: Proposed deployment diagram for the PEDR system.

The PEDR Client side, Node (1), contains the CORBA Client that interacts with the end-user through an MS-Excel application (see Figure 5.8). Node (2) is the core of the application where the PEDR Manager and the *process model* (Linear ESO) components reside. Node (3) represents the numerical Optimizer side where the Optimizer wrapper is placed (in this application CPLEX solver from ILOG is wrapped (Benqlilou et al. (2002c))) is accessed through the CO-MINLP standard interface. Finally, the Database wrapper and the database server (MySQL, in this case) represents the Database component in Node (4).

Several Data Reconciliation and Parameter Estimation scenarios have been solved using the developed prototype demonstrating the adequacy and operability of the specification and checking the consistency of the information that flows through software components. Figure 5.8 presents the mass balance reconciliation of PROCEL pilot plant in a continuous mode. This PROCEL configuration contains two reactors inter-connected with two heat exchangers. The measurements and reconciled values of all the process flows are also shown in Figure 5.8.

5.5 Conclusions

The main aim of this chapter has been the design and implementation of a software architecture for distributed Data Reconciliation and Parameter Estimation applica-

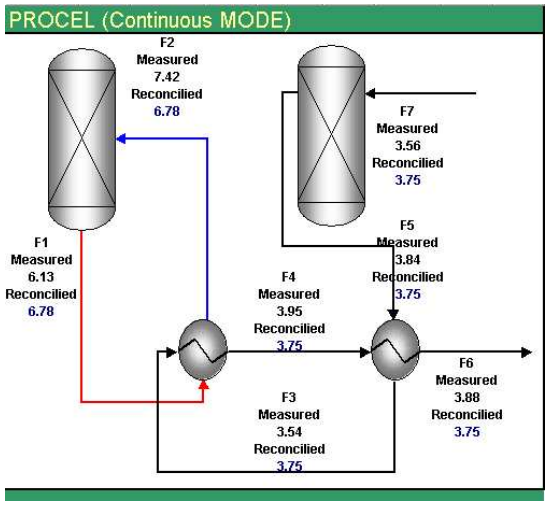


Figure 5.8: steady-state reconciliation on PROCEL plant at continuous mode.

tions.

This contribution is presented towards the standardization of PEDR software packages. The task of maintaining and supporting new process models, databases and/or Optimizers within PEDR system can be justified in terms of cost and implementation effort using the proposed CO software specification and architecture.

A software prototype has been developed to validate and demonstrate the use and benefits in terms of reuse transparency and innovation of the proposed component and the specified interface.

In this chapter the specification of the Parameter Estimation and Data Reconciliation interfaces are conceived as a generic approach, able to deal with dynamic, nonlinear process with relatively low effort and without making any essential change to the other components. The system presented would be very appealing to industry or researchers that already have access to software that can perform the tasks of PE and DR since a most efficient use of such packages would be enhanced through the standardization of communication interfaces (which is the plug and play philosophy proposed by the CO and GCO projects). The generality of the proposed approach allows including present and future algorithms to estimate, for instance, unknown variance and covariance for process variables (Morad et al. (1999)), as well as using new modules for solving the optimization problem.

Part II

Reliable and accurate design/retrofit of sensor networks

Chapter 6

Design of accurate sensor networks for dynamic systems

Abstract

In the previous chapters of this thesis the estimation accuracy of time evolving process variables in steady-state and dynamic systems has been discussed. This chapter presents a methodology to include these estimation techniques in the design of accurate sensor networks.

The method aims at maximizing Kalman filtering performance using accuracy as its main performance index. To accomplish this, both the measurement noise and the observation matrices are manipulated.

The method has been applied to an academic case study and in the Tennessee Eastman challenge problem showing promising results.

6.1 Introduction

If the precision is selected as the goal for designing sensor networks, data reconciliation can be used. Given a set of data measured by a certain sensors, DR provides the precision of each process variable (Eq. (1.6)). Even if not all the process variables are measured the DR can provided an estimated of these unmeasured variables if they are classified as observable.

Based on these concepts several techniques have been developed to design and upgrade precise (or accurate) instrumentation for process plants at steady-state operation mode. Extending these steady-state sensor placement procedures to deal with designing sensor networks schemes of dynamic processes mainly relies on:

- selecting a dynamic data reconciliation that can provide precision at the design phase. Other performance criteria (e.g. gross error detectability, resilience, etc) can also derived from accuracy as reported in Chmielewski et al. (2002). For example Charpentier et al. (1991) defines gross error detectability as follows:

$$d_i = \sqrt{1 - \frac{\sigma^2_{\hat{x}_i}}{\sigma^2_{y_i}}} \tag{6.1}$$

- classifying variables of dynamic processes. Variable classification can help to determine the observability and redundancy needed for reconciliation.

- integrating dynamic data reconciliation with dynamic variables classification procedures.

Several DDR techniques can provide accurate estimate of the measurement error covariance matrix of estimated process variables as discussed in section 1.2.5. However, in some cases, the KF provides superior performance in terms of variance reduction and dynamic tracking, as discussed by Benqlilou et al. (2002a). But the more important point that should be stand out for the use of KF over the existing DDR techniques in the design of precise sensor networks is its capability to calculate the variance of the estimation without any knowledge of the measurements (see Eq. (1.33)). Furthermore, providing the observability requirement, the KF is able to estimate all states using an **incomplete** and noisy measurement set.

The classification of variables for dynamic processes using observability matrices for control point of view was considered in Stanly and Mah (1981). Later, in Albuquerque and Biegler (1996) an alternative was proposed. They first discretize the dynamic model and then apply an LU decomposition to obtain the observability and redundancy properties of the resulting approximated algebraic model.

In this chapter sensor placement problem is addressed from the state-space model identification perspective. An analysis of observability is performed on different approaches and used in the sensor placement formulation. The optimal design is obtained using a genetic algorithm. The methodology is finally applied to several conditions.

6.2 Proposed methodology

6.2.1 Measure of instruments performance

6.2.1.1 Performance measure of variable estimation

The computation of the error covariance matrix $\mathbf{P}_{k/k}$ is independent of the process variable measurements as is apparent from Eq. (1.33). Indeed, the parameters required for computing $\mathbf{P}_{k/k}$ are: $\mathbf{R}^*$, $\mathbf{Q}^*$ and $\mathbf{P}_0$ as well as the sensor network and the associated observation matrix so their evaluation can be performed without any knowledge of the variables values.

1. The measurement error covariance matrix $\mathbf{R}^*$ is given by the intrinsic quality of the measuring devices; thus, the elements of $\mathbf{R}^*$ can be chosen as the design parameters for the sensor placement problem. Assuming that measurement errors are independent (zero covariance), the matrix $\mathbf{R}^*$ has a diagonal form (Liebman et al. (1992)).

2. The process noise covariance $\mathbf{Q}^*$ (modeling error) is generally more difficult to determine due to the difficulties in directly observing the process noise. Assuming a deterministic process, where all process variability sources are included in the process model, then $\mathbf{Q}^* = \mathbf{0}$. However, in real conditions, disturbances affecting the process exist and have to be included. Usually, the diagonal elements of $\mathbf{Q}^*$ are assumed to be positive and fixed and the off-diagonal elements are set to zero.

3. Finally, following a practical initialization of the KF the value of $\mathbf{P}_0$ is selected to be equal to $\mathbf{R}^*$. Nevertheless, any initial value of $\mathbf{P}_0$ will lead to the same asymptotic value of $\mathbf{P}$ if the observability requirement is fulfilled.

Therefore, $\mathbf{P}_{k/k}$ is selected as a basis for assessing the accuracy of the KF estimation. If the measurement errors $\mathbf{R}^*$ are independent of time and normally distributed, the KF is an optimal unbiased estimator. However, if the measurements errors are not normally distributed the KF can not be used because it is biased as reported in Narasimhan and Jordache (2000).
Since $\mathbf{P}_{k/k}$ is not constant along the time horizon $k = 0 \cdots n$, the KF performance of process variable j, k_P^j can be calculated by averaging $\left[P_{k/k}\right]_j$ over the entire time horizon as follows:

$$k_{p,1}^j = \frac{1}{n}\left(\sum_{k=0}^{n}\left[P_{k/k}\right]_j\right) \tag{6.2}$$

Under conditions where $\mathbf{Q}^*$ and $\mathbf{R}^*$ are time-invariant, the covariance $\mathbf{P}_{k/k}$ and the Kalman filter gain $\mathbf{K}_k$ reach a constant value in few iterations. Therefore, the asymptotic value of $\mathbf{P}_{k/k}$ can also be used as a performance measure,

$$k^j_{p,2} = \lim_{k\to n}\left(\left[P_{k/k}\right]_j\right) \tag{6.3}$$

In fact, when the KF is applied to a system that is continuous and dynamic, Eq. (6.2) is preferred, whereas when conditions reflect short lived batch systems Eq. (6.3) is more appropriate. This last one is considered in this analysis.

6.2.1.2 System performance

The performance measure presented in Eq. (6.3) corresponds only to a particular process variable. It is more suitable to maximize the performance of the filter estimations for all variables. However, different sensor networks lead to different values of the elements of $(k^j_p,\ j = 1\cdots J)$ and maximizing the performance for one particular variable may not be compatible with maximizing it for others, causing conflict.
It is proposed to use different performance measurements for the whole system by selecting a function relating the individual variable performance measures given by Eq. (6.3).

One simple alternative is to select the variable with the lowest performance value. A more elaborated option is based on evaluating the distance of sensor network from the "best" sensor network. The "best" sensor network is defined as the one in which all the process variables are completely measured with the most accurate devices available. When only a few process variables are of interest, only these are considered. Indeed, let $\mathbf{S}$ be the set of variables of interest. Then, the performance of the system can be computed by comparing the current system performance ($k^j_{p,c}$) and the "best" system performance ($k^j_{p,b}$) for all the process variables j, as follows:

$$k^{\mathbf{S}}_p = \left(k_0 + \sum_{j\in\mathbf{S}}\left|k^j_{p,c} - k^j_{p,b}\right|\right)^{-1} \tag{6.4}$$

where k_0 is introduced to avoid singularities: indeed, if the value of $k_0 = 1$, the maximum performance value is in the intervale $[0,1]$ and the best performance corresponds to $k^{\mathbf{S}}_p = 1$.

6.2.2 Kalman filter and sensor placement

Clearly, at a design phase the measurement errors ($\mathbf{v}_k$) do not exist since no measurement are available, however, from a sensor placement perspective $\mathbf{v}_k$ are basically

considered the intrinsic errors of the measuring device. Therefore, a possible connection between sensor placement and the Kalman filter can be performed via the measurement error variance matrix $\mathbf{R}^*$, by assigning a large value to any unmeasured variables in this matrix. Finally, the design of accurate sensor network is provided by searching the value of the diagonal of $\mathbf{R}^*$.

In classical control, manipulated variables $\mathbf{u}$ are assumed to be known. Then, the elements of $\mathbf{P}$ are the variance of state variables $\mathbf{x}$. However, from a monitoring perspective, the estimated value and variance of all variables are considered, avoiding the distinction between state and manipulated variables. Therefore, to include the variable set $\mathbf{u}$ in the Kalman filter algorithm, these variables have to be considered as "state" variables, leading to a new n_z dimensional "state" variables, $\mathbf{x}_k^* = \left[\mathbf{u}_k, \mathbf{x}_k\right]$. This change involves an update of the model identification ($\mathbf{A}$, $\mathbf{B}$ and $\mathbf{H}$). Indeed, since the control input variables are considered state variables, the matrix $\mathbf{B}$ is dropped. Next, if we assume that all process variables can be measured directly, matrix $\mathbf{H}$ is the identity matrix of size n_z. This assumption is very important for Kalman filter gain calculation, as shown in Eq. (1.34).

To identify the new state transition matrix $\mathbf{A}^*$ it is assumed that input variables during a given time period are correlated with the input during the previous time period, as follows:

$$\mathbf{u}_k \approx \mathbf{u}_{k-1} + \mathbf{w}_{k-1}^u \tag{6.5}$$

Using the new variable $\mathbf{x}_k^*$ and Eq. (1.26) it is possible to get the new transition state matrix $\mathbf{A}^*$:

$$\mathbf{A}^* = \begin{bmatrix} \mathbf{I}^{nu.nu} & \mathbf{0}^{nu.nx} \\ \mathbf{B}^{nx.nu} & \mathbf{A}^{nx.nx} \end{bmatrix} \tag{6.6}$$

and finally:

$$\mathbf{x}_k^* = \mathbf{A}^* \mathbf{x}_{k-1}^* + \mathbf{w}_{k-1}^* \tag{6.7}$$

$$\mathbf{y}_k^* = \mathbf{H}^* \mathbf{x}_k^* + \mathbf{v}_{k-1}^* \tag{6.8}$$

6.3 Redundancy analysis

It is clear that the system performance $k_p^{\mathbf{S}}$ can lead to acceptable values for any set of sensors if the process variables belonging to $\mathbf{S}$ are all strictly observable either independently $(\mathbf{H}^*)$ or through the transition model equations $(\mathbf{A}^*)$. Thus, it is necessary to assume that for any sensor placement design model only those sensor networks able to guarantee the desired observability should be considered.

The values of the instrument performance measurement will depend not only on the used dynamic data reconciliation technique (in this case KF) but also on the sensor network selected. From the set $\mathbf{S}$ those variables that are redundant are adjusted and

their variances are fed back by DR. The estimation of unmeasured but observable variables and their variances can be obtained by means of DR techniques. That is DR allows to obtain the whole variable estimation with an incomplete and noisy set of observation. By this way it is possible to get the variance of all the process variables in **S** in order to calculate the value of $k_P^{\mathbf{S}}$.

Therefore, observability analysis is an important aid to design instrumentation schemes. An unmeasured variable is defined as unobservable if it cannot be uniquely determined through the measured variables. A measured variable is defined as nonredundant if deletion of its measurements will make this variable unobservable. Thus, both definitions are based on the uniqueness of determining a variable value.

These fundamental properties will serve either for a steady-state process as well as for dynamic processes. Albuquerque and Biegler (1996) developed an efficient method for classifying variables of a dynamic process. After discretizing the differential equations using Implicit Runge-Kutta, they linearize the nonlinearities present in the process model. Finally, they apply the properties of observability and redundancy to derive the tools necessary for such classification using sparse LU decomposition. An important result presented in their work is that the redundancy analysis for steady-state differs to that of dynamic systems.

It is important to mention that the explained *classification variables* procedure gives the same results as those presented by the Kalman filter using the observability matrix calculation (different form the observation matrix $\mathbf{H}^*$), Eq. (6.9). To ensure that the Kalman filter converges to an acceptable and **unique value** the observability constraints must be satisfied.

Therefore, observability and redundancy has a direct impact on the Kalman filter performance. The error variance of the an unobservable variable will be very high, because without measurements the measurement error tends to infinity and thus, the KF gain is equal to zero. This means that the second term of the right hand side of Eq. (1.33) is null while the first term is continuously increasing for each iteration of the KF. Furthermore, if the pair ($\mathbf{A}^*$, $\mathbf{H}^*$) is completely observable, each unmeasured but observable variable is given a unique error variance value.

As it was explained, unmeasured variables are made evident in the sensor placement model formulation by assigning infinite variance to the corresponding positions of $\mathbf{R}^*$. Additionally it is necessary to explicitly handle the unobservable variables by modifying $\mathbf{H}^*$. The rows of $\mathbf{H}^*$ corresponding to unobservable variables $(\mathbf{y})$ are set to zero-row so that the observability requirement is provided by checking if the rank of the observability matrix is equal to the number of state variables as shown in Eq. (6.10).

$$\mathbf{O} = obsv(\mathbf{A}^*, \mathbf{H}^*) = [\mathbf{A}^*, \mathbf{A}^*\mathbf{H}^*, \mathbf{A}^*(\mathbf{H}^*)^2, \cdots, \mathbf{A}^*(\mathbf{H}^*)^{n_z-1}] \tag{6.9}$$

$$rank(\mathbf{O}) = n_z \tag{6.10}$$

Indeed, if the hypothesis of observability is not fulfilled, that is, the observability matrix of the pair ($\mathbf{A}^*$, $\mathbf{H}^*$) is not full-row rank, the error covariance matrix $\mathbf{P}_{k/k}$ will not converge to an unique value from different initial conditions as reported in Maybeck (1979).

6.4 Sensor placement procedure

The objective is to maximize as much as possible the system performance given in Eq. (6.4) by varying the diagonal elements of matrix $\mathbf{R}^*$ subject to a cost bound c^{max} and a set of additional constraints $\psi(\mathbf{S})$ which are related to observability matrix. From the regular elements of the diagonal of $\mathbf{R}^*$, the placement and type (e.g. level-meter, flow-meter, etc) of sensors are directly obtained.

Assume s_{ij} is an integer variable indicating the placement of sensor type i, at network location j. Additionally, the sensor type is given by the variance of its measurement error (σ_i^2). When a variable is not measured a 'dummy' sensor of $(\sigma_i^2 \to \infty)$ is selected having null cost. This directly affects $\mathbf{R}^*$ and $\mathbf{H}^*$. Therefore, the optimal sensor network problem is formulated as follows:

$$\max_{s_{ij}} \left(k_p^{\mathbf{S}} \right) \tag{6.11}$$

subject to:

$$\begin{array}{c} \sum_i \left(c_i . s_{ij} \right) \leq c^{max} \\ \psi(\mathbf{S}) = 1 \end{array} \tag{6.12}$$

where $\psi(\mathbf{S})$ represent a set of constraints. Its value is equal to one when the sensors network allows observing all the process variables belonging to the set $\mathbf{S}$, and zero otherwise. In practice this constraint should ensure that the variable values belonging to $\mathbf{S}$ can be inferred, thus some redundancy should also be allowed to validate those inferences, that is, the function $\psi(\mathbf{S})$ should ensure a minimum redundancy in order that an acceptable precision of variables is provided.

Additionally, different threshold values of cost c^{max} can be used to build a Pareto curve. Since the relationship between investment and performance is generally non-linear, increasing c^{max} by a small amount might lead to a considerable performance improvement and vice-versa.

The steps of the design of accurate sensor networks are:

1. Determine the "best" performance, given when all process variables are measured with the most accurate sensors.

2. Obtain the list of sensor networks from the total set of combination alternatives that satisfies the constraints and sensor characteristics.

3. Obtain the system performance by selecting the maximum value of the objective function given the list obtained in step (2) for a specified cost.

A systematic search can be used to obtain a maximum performance of the sensor network. By looking at the spectrum of solutions it is possible to decide the best trade-off between performance and cost. Moreover, the Pareto optimum space over the different objectives can also be determined. Here in this chapter the resulting optimization problem is solved by means of genetic algorithm. GA shows to be attractive for solving the highly combinatorial sensor placement optimization problem as reported in (Heyen et al. (2002); Musulin et al. (2003))

6.5 Genetic algorithm approach

One of the main steps in a GA approach is the codification of the individuals. In the proposed algorithm each chromosome represents a sensor network. The chromosome has to contain the necessary information to check the feasibility of the sensor network as well as to evaluate the fitness function.

This necessary information includes: the quality, type and cost of the measuring devices as well as the measuring point. With this aim, each gen in the chromosome is associated with a measuring point and implicitly with the sensor type. Additionally, the numerical value of each gen is mapped with a sensor quality and cost. As a result, a generic codification is obtained, which allows the evaluation of the sensor networks performance using different objective functions and restrictions. Once the codification is defined an initial population is generated randomly and the fitness of each feasible individual is evaluated. The sensor network feasibility is a pre-requisite for evaluating the objective function. If the individual is not a feasible solution, the objective function value is considered infinite and the sensor network fitness is set to zero.

6.6 Results and discussion

6.6.1 Academic case study

Figure 6.1 shows a process network used as a motivating example to evaluate the proposed sensor placement methodology.

This case study is taken from the work of Darouach and Zasadzinski (1991) (eight streams and four storage tanks are selected). In this example the flow-rate and mass hold-ups are the desired process variables to be monitored, and it is possible to place flow-rate sensors in all the streams and level sensors in all nodes. In this case the

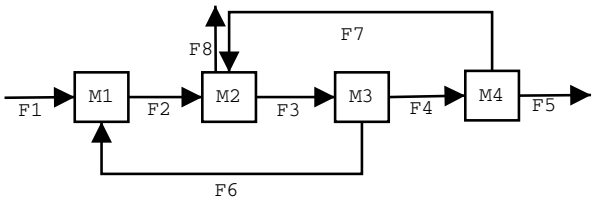


Figure 6.1: Case study for placing sensors under time evolution processes.

dynamic mass balance is adopted for supporting the instrumentation design schemes. Let the accuracy of measurement devices belong to the range [1%, 5%] with a cost of [$2500, $1800], respectively and assume that to any potentially measuring devices can be assigned a sensor with one of these accuracies. The matrixes $\mathbf{A}^*$**, B,** $\mathbf{H}^*$ and the state vector $\mathbf{x}^*$ for this case study are as follows:

$$\mathbf{A}^* = \begin{pmatrix} \mathbf{I}^{8\times8} & \mathbf{0}^{8\times4} \\ \mathbf{B}^{4\times8} & \mathbf{A}^{4\times4} \end{pmatrix}, \mathbf{B} = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & -1 & 0 \end{pmatrix}$$

$$\mathbf{H}^* = \mathbf{I}^{12\times12}$$

$$\mathbf{x}^* = [F1, F2, F3, F4, F5, F6, F7, F8, M1, M2, M3, M4]$$

$$\mathbf{Q}^* = 1.2 * \mathbf{I}^{12x12}$$

The GA approach explained in section 6.5, has been used to solve the sensor placement optimization problem resulting from case study given in Figure 6.1. The GA configuration parameters are listed in Table 6.1.

Table 6.1: GA parameters used for the academic case study.

N_{ind}	N_G	G	P_c	P_m
400	10	5	0.7	0.00175

Table 6.2, presents an example of the chromosome codification adopted for solving this optimization problem. Each gene in this chromosome represents the standard deviation of the instrument assigned, if no sensor is assigned a very high value of standard deviation is assigned (e.g. 10000%).

In order to find the suitable investment-performance trade-off, the sensor network with the optimum performance is searched covering the complete feasible range of

Table 6.2: Illustration of the chromosome codification.

$F1$	$F2$	$F3$	$F4$	$F5$	$F6$	$F7$	$F8$	$M1$	$M2$	$M3$	$M4$
–	1%	5%	–	–	1%	5%	1%	–	1%	1%	–

margin costs c^{max}. This is achieved by solving the problem several times for different values of c^{max}, starting from a low c^{max} and reaching and upper c^{max}.

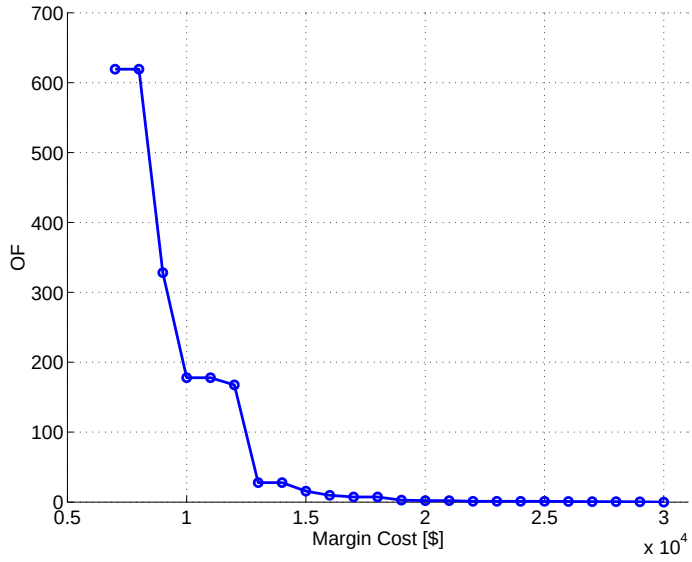
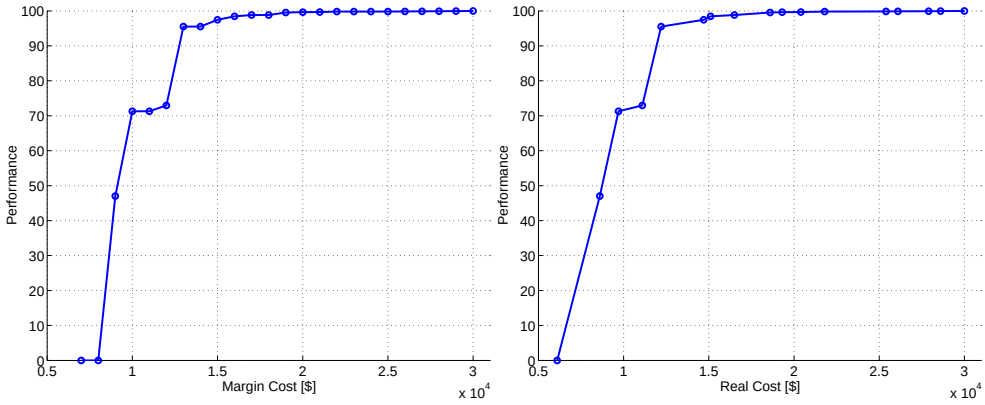


Figure 6.2: Objective function vs. margin cost.

From Figure 6.2 it can be seen the behavior of minimum objective function (OF) value corresponding to the best system performance for a given margin cost. The advantage of these procedures for improving the sensor network design decision-making can be seen from analyzing the obtained profile. The approach shows a significant jump in performance when going from one sensor network alternative to another, and suggests when it is favorable to invest for improving sensor network performance and when the investment just implies a marginal improvement. First, it is clear that for a margin cost lower than $7000 no feasible solutions exist. However, starting from this margin cost, the performance is enhanced by increasing the investment. The best performance is the one obtained when placing the more accurate sensors in all the potential measurement points. This sensor network leads to a

cost of $30000.

The best margin-cost/performance relationship can be obtained by analyzing the break points in the curve presented in Figure 6.3, $10000, $13000, $15000 and $19000.



(a) Perfromance vs. margin cost

(b) Performance vs. real cost

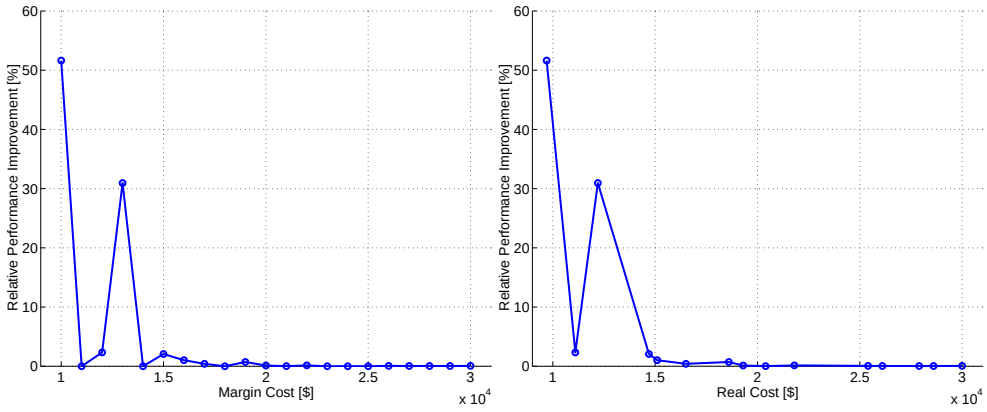
Figure 6.3: Sensor placement performance reached vs. cost.

The relative performance improvement with respect to the margin cost is also plotted in Figure 6.4. This improvement gained from the last performance when investing $1000 more. The peaks in Figure 6.4 illustrate the optimum trade-off between instrumentation investment and the corresponding sensor network performance.

The characteristics of the optimal sensor networks for these parameters in terms of sensor type and location for margin costs are given in Table 6.3.

Table 6.3: Example 1, sensor network characteristics

$F1$	$F2$	$F3$	$F4$	$F5$	$F6$	$F7$	$F8$	$M1$	$M2$	$M3$	$M4$	*Margin Cost*	*Real Cost*	*OF*	k_p^S
–	–	1%	–	–	-	5%	–	5%	–	5%	5%	$10000	$9700	177.8	71.30
–	–	**1%**	–	–	**1%**	–	–	**5%**	**5%**	**5%**	**5%**	**$13000**	**$12200**	**27.8**	**95.51**
–	5%	1%	–	–	–	%5	–	1%	1%	5%	5%	$15000	$14700	15.7	97.47
–	–	1%	–	5%	5%	–	1%	1%	1%	1%	1%	$19000	$18600	2.8	99.54



(a) Relative performance vs. Margin Cost (b) Relative Perfomance vs. real cost

Figure 6.4: Relative performance improvement when increasing cost.

However, the "best" performance given in the last column of Table 6.3 can be obtained by less cost than the margin cost. In some cases, even if the margin cost is increased, the sensor network performance does not change. Thus the investment and the performance remain the same. For example between $10000 and $11000, or between $13000 and $14000. Therefore, for sensor placement cost evaluation purposes real cost has to be used instead of margin cost. The real cost (investment) is the cost of the sensor network, that can be obtained by summing the value of each measuring device belonging to the network. It does not necessarily coincide with the margin cost c^{max}, which is a restriction value. In Figure 6.5 the real cost is represented against the margin cost.

Different networks performance and the relative performance improvement are represented with their real cost in Figures 6.3 and 6.4. It can be noted than the higher the investment the better is the sensor network performance. The performance tends to rise up quickly at first increasing to a break in the curve ($12200, 95.51%), then, presents a slight slope until the break represented by the pair ($18600, 99.54%), and finally shows a quasi horizontal line representing asymptotic performances. These points can also be identified as the peaks in the curve presented in Figure 6.4. These breaks depicted the real "best" trade-off between investment and performance. In principle a very attractive solution is the sensor network corresponding to the more significant break in the curve as is shown by the bold line in Table 6.3.

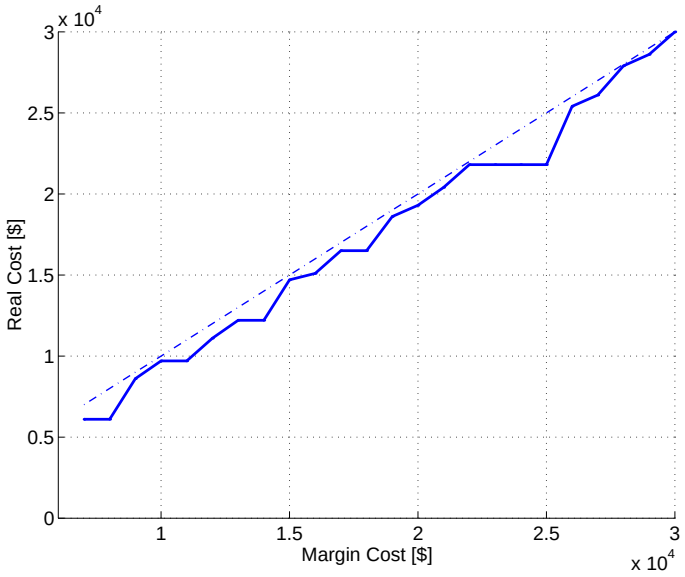


Figure 6.5: Relationship between real-cost and margin-cost.

6.6.2 Tennessee Eastman Problem

The Tennessee Eastman process (TE) (Downs and Vogel (1993)) involves the production of two products, G and H, from four reactants, *A*, *C*, *D* and *E*. Additionally, there are two side reactions that occur and an inert *B*. All the reactions are irreversible and exothermic. There are 41 measured process variables and 12 manipulated variables. Several authors have studied this problem from different points of view in dynamic and steady-state (McAvoy and Ye (1994); Kano et al. (2002)). The process is depicted in Figure 6.6.

The process is composed of five main unit operations: an exothermic 2-phase reactor, a product condenser, a flash separator, a re-boiled stripper, and a recycle compressor.

The gas reactants are fed to the reactor where they react to form liquid products. A non-volatile catalyst dissolved in the liquid catalyzes the reactions in the gas phase. The products leave the reactor in vapor form with part of the reactants that did not react, and the catalyst stays in the reactor. The heat generated in the reactor is extracted by a cooler. The reactor's exit stream goes through a cooler to condense the products, and then to a liquid-vapor separator. The not condensed components are returned back to the reactor through a centrifugal compressor. The condensed components are processed in a separation column (stripper) to extract the remaining reactants.

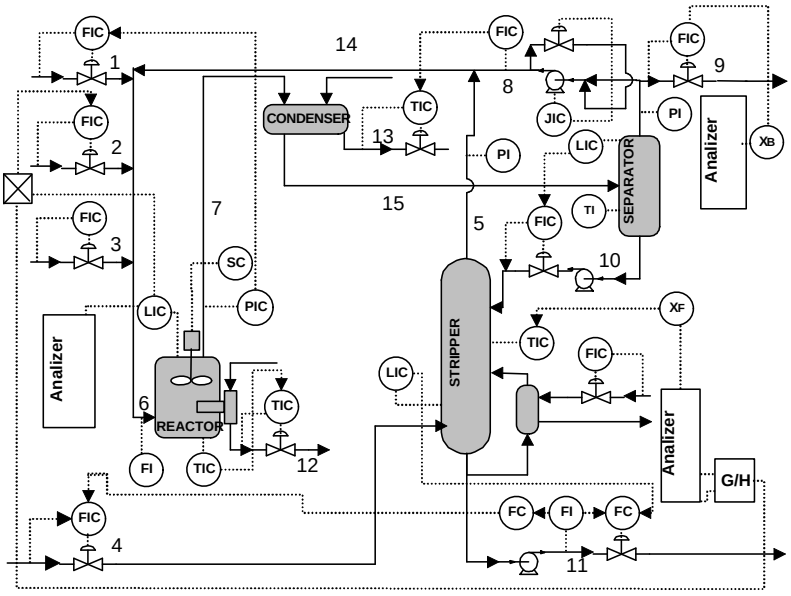


Figure 6.6: Tennessee Eastman process flowsheet.

The products G and H leave the stripper and are separated in other sections of the plant not considered in the problem. The inert and the by-products are extracted from the system as vapors through a purge in the liquid-vapor separator. In this approach the decentralized control system proposed by McAvoy and Ye (1994) has been used (Figure 6.6).

The design of a sensor network with high accuracy at minimum cost has been performed. The application focused in reconcile signals corresponding to material flows and units inventory. For such purpose two sets of sensors, one containing flow-meters and other containing level-meters, have been proposed. The available flow-meter variances are 0.1%, 0.25%, 0.5%, 1% and 2% and their corresponding costs are \$2000, \$1700, \$800, \$500 and \$250. Whereas, the level-meter variances and costs are 0.1%, 0.5%, 1% and 2% and \$1000, \$800, \$500 and \$300 respectively.

The potential measuring points are thirteen streams ($1-11$, 14 and 15) and the hold-ups of the reactor ($L1$), separator ($L2$) and stripper ($L3$). Accordingly to the proposed sensor placement problem statement, the number of sensor networks alternatives is combinatorially high ($13^6 \cdot 3^5 = 1.1729E+9$). The optimization problem has been solved using the approach presented in section 6.6.1. The GA parameters has been set to $N_G = 20$ and $N_{ind} = 500$.

The trade-off between the cost and performance is shown in Figure 6.7. By a

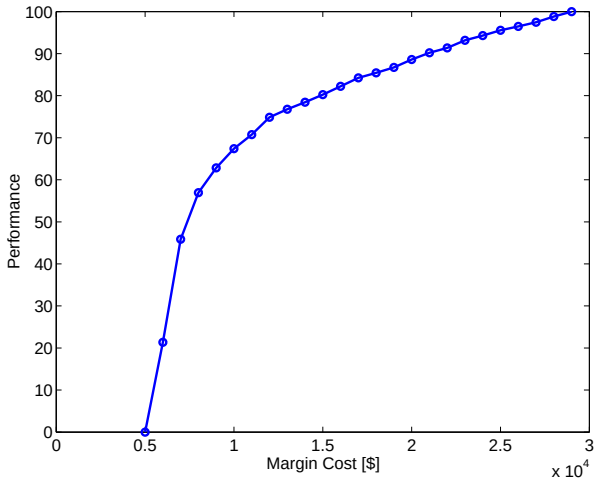


Figure 6.7: Performance vs. margin cost for the Tennessee Eastman case study.

Pareto optimal analysis it is possible to detect the "best" performance that can be achieved given a desired investment. As it was previously explained by analyzing the relative performance improvement it is possible to detect the more convenient investment point (i.e. break points).

In this case, an investment of $10000 is sufficient to get an acceptable performance of almost 70.00% and the corresponding sensor network is shown in Table 6.4.

6.7 Conclusions

The major contribution presented in this chapter is the establishment of a methodology for optimal placing measuring devices satisfying the network performance accuracy of dynamic systems. That use Kalman filter for evaluating the accuracy. The solution strategy has been implemented in academic and industrial case studies showing promising results. The profile of the relative increase of the system performance along the sensor network and the associate investment margin gives the designer all alternatives.

Table 6.4: Proposed sensor networks for the Tennessee Eastman.

1	2	3	4	5	6	7	8	9	10	11	14	15	$L1$	$L2$	$L3$
1%	1%	1%	1%	2%	2%	–	0.5%	1%	0.5%	0.5%	0.5%	0.5%	0.1%	0.1%	0.1%

Nomenclature

$\mathbf{A}_k$	state transition matrix at sampling time k
$\mathbf{B}_k$	control gain matrix at sampling time k
c_i	capital cost of sensor type i
c^{max}	maximum sensor network investment cost
d_i	measure of detectability of an error at variable i
$\mathbf{H}_k$	observation matrix at sampling time k
k/k'	estimating at sample time $k.T$ using the past $k'.T$ samples
k_p^j	KF performance indicator of variable j
$k_{p,1}^j$	KF performance indicator of variable j averaging $\left[P_{k/k}\right]_j$ over n
$k_{p,2}^j$	KF performance indicator of variable j using asymptotic $\left[P_{k/k}\right]_j$ value
$\mathbf{K}_k$	Kalman filter gain matrix at sampling time k
$\mathbf{K}_p$	vector of the $K_{p,2}^j$, $\forall j \in S$
$k_{p,b}^j$	"best" sensor network performance of process variable j
$k_{p,c}^j$	current sensor network performance of process variable j
k_p^S	whole system performance
$k_{p,u}$	upper bound on performance design specification
N_{ind}	number of individual in the population
N_G	maximum number of generation value
P_c	crossover probability
P_m	mutation probability
$\left[P_{k/k}\right]_j$	error covariance matrix at sampling time k of variable j
$\mathbf{P}_0$	covariance matrix of initial estimates of states variables $\widehat{\mathbf{x}}_0$
$\mathbf{Q}_k^*$	covariance matrix of random variable $\mathbf{w}_k$

$\mathbf{R}_k^*$ covariance matrix of random variable $\mathbf{v}_k$
$\mathbf{S}$ set of process variables of interest
s_{ij} integer variable indicating the placement of sensor type i at network location j
T sampling period
t time
$\mathbf{u}_k$ vector of control input variable at sampling time k
$\mathbf{v}_k$ vector of measurement noise
$\mathbf{w}_k$ vector of process model noise
$\mathbf{w}_{k-1}^u$ vector of *measurement model* noise
X chromosome
$\mathbf{x}_k$ vector of state variable at sampling time k
$\mathbf{x}_k^*$ state variable equal to $[\mathbf{u}_k, \mathbf{x}_k]$
$\mathbf{y}_k$ vector of measurements variables at sampling time k

Operators

$E[.]$ expectation operator
$obsv()$ function calculating the observability matrix
$rank()$ function calculating the rank of a matrix

Greek letters

ϖ fitness function
σ_i^2 variance of sensor type i
$\sigma_{\widehat{x}_i}$ standard deviation of the reconciled value i
σ_{y_i} standard deviation of the measurement value i
$\psi()$ observability requirement function

Subscripts

i sensor type
j process variable
k sampling time
n time-horizon

Chapter 7

Design/retrofi t of reliable sensor networks

Abstract

This chapter proposes an approach that combines quantitative process knowledge and fault tree analysis into a new methodology for evaluating the reliability of process variable estimation, taking into account both hardware and functional redundancy.

The reliability of estimating process variable is used to determine the sensor network reliability, which in turn is used for the design/retrofit of sensor networks. Thus, a reliable sensor network design/retrofit problem is formulated as a mathematical programming optimization problem and solved using genetic algorithms.

The performance of such proposal is compared with the current approaches using different case studies and handling several scenarios.

7.1 Introduction

The design of sensor networks is the basis upon which the performance of Monitoring systems (Benqlilou et al. (2004); Musulin et al. (2003)), Fault Diagnosis Systems (Raghuraj et al. (1999)), and/or Optimization systems (Fraleigh et al. (2003) relies. The design of sensor networks includes the determination of sensor characteristics such as: type, number, reliability, placement and so on, while minimizing an objective function such as the instrumentation cost.

Designing a sensor network that allows the observation of all process variables was first addressed by Vaclaveck and Loucka (1976). Later, this problem was solved by Madron and Veverka (1992) regarding the minimum total cost.

However, sensor failure may lead to a reduction of measurements affecting seriously control, monitoring and optimization systems, hence the whole process performance. Thus, it is necessary to ensure that it is still possible to observe key process variables even if one or more sensors fail.

Observing a process variable can be translated mathematically by making non-null the probability of estimating this variable at a given time t (reliability). The evaluation of this probability is tightly related to the different ways to estimate a process variable given a sensor failure probability and a specific sensor network. Based on these concepts, a method for optimal sensor location in a pure flow process was developed using graph-theory by Ali and Narasimahan (1993); Ali and Narasimhan (1995).

In their proposal, the authors first determined the minimum number of sensors that ensure system observability (or system redundancy). Using this number, the complete set of networks that ensure system observability is obtained. Finally, the network that maximizes system reliability is obtained from this set. However, their proposal does not consider directly the cost neither the individual process variable reliability. These points were later addressed by Bagajewicz and Sánchez (2000a). Additionally, the authors transformed the problem presented by Ali and Narasimahan (1993); Ali and Narasimhan (1995) in a mathematical programming problem. Similarly, and based on the analysis of the cycles of process graph and taking into account variable observability and reliability of sensors, Luong et al. (1994) determined an optimal measurement system with regards to reliability and cost analysis in conjunction with observability constraints.

In this chapter an approach is proposed for evaluating the reliability of process variable estimation taking into account all the redundancies that offers the system either hardware or functional. In this evaluation, both quantitative process knowledge and fault tree analysis are considered and combined, which leads to a more suitable and practical evaluation of reliability. The reliability of estimating each one of the key process variables is then used to determine sensor network reliability, which in turn is used as a set of sensor placement constraints in the design/retrofit procedure.

Thus, a general sensor placement formulation is proposed, which considers the number (hardware redundancy or multiplicity) of sensors of a given type (reliability) that are to be assigned to a given process variable while satisfying the reliability requirements at the minimum total cost. This proposal can be applied for the design as well as for retrofitting. Additionally, an analysis of the minimum reliability value that allows both the observability as well as the redundancy of the system is performed as well as the determination of the best trade-off cost/system reliability. The general sensor placement optimization problem formulated is successfully solved using genetic algorithms. The performance of this proposal is compared with the current

approaches considering motivating case studies.

7.2 Reliability evaluation

7.2.1 Sensor reliability

The sensor reliability is an intrinsic quality of the sensor, that can be defined as the probability $r_k^s(t)$ of the non-failure of the sensor k at a time t. Poisson law (see Figure 7.1) can be used to represent this monotonically decreasing probability function (Eq. 7.1) as reported by Luong et al. (1994).

$$r_k^s(t) = \exp\left(-\lambda_k.t\right) \tag{7.1}$$

where λ_k is a constant representing the rate of failure of the sensor k.

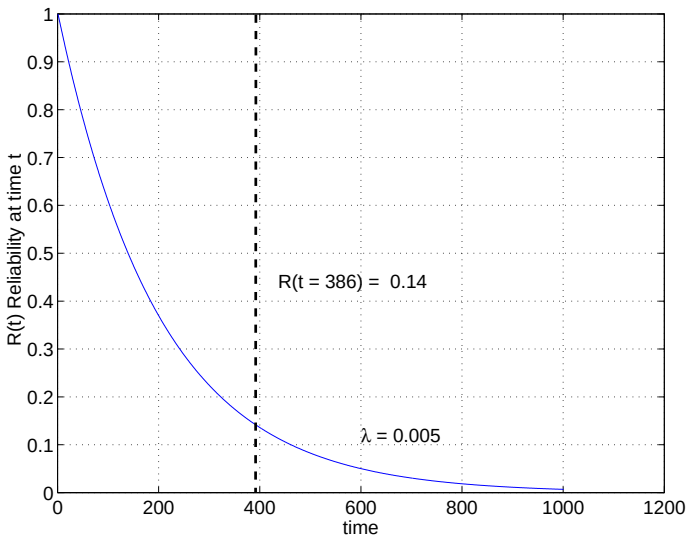


Figure 7.1: Sensor reliability behavior.

The sensor reliability $r_k^s(t)$ is an important parameter for the sensor network design problem, which consists of determining the number n_{jk} of each specific sensor type k to be assigned to each process variable j so that the reliability of estimating j at a time t, $r_j(t)$, is greater than a minimum allowable reliability.

Table 7.1: Reliability $r_{F_1}(t)$ for different sensor networks.

Reliability Equation $r_{F_1}(t)$	Value	Reliability type
$r^s_{s1}(t)$	0.80	Sensor reliability
$r^s_{s3}(t)\, or\, r^s_{s4}(t) = r^s_{s3}(t) + r^s_{s4}(t) - r^s_{s3}(t).r^s_{s4}(t)$	0.96	Hardware reliability
$r^s_{s3}(t)\, and\, r^s_{s4}(t) = r^s_{s3}(t).r^s_{s4}(t)$	0.64	Functional reliability

7.2.2 Process variable estimation reliability

The probability $r_j(t)$ of estimating process variable j at time t must simultaneously consider sensor reliability, hardware reliability and functional reliability.

As an example, consider a simple process unit with one feed stream F_1 and two product streams F_2 and F_3, as shown in Figure 7.2.

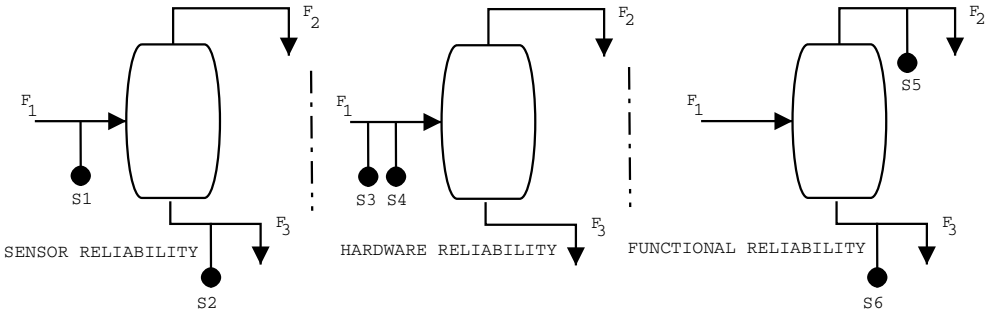


Figure 7.2: Process variable estimation reliability.

For this process unit, the three flow variables are related to each other through the pure mass balance expressed in Eq. (7.2).

$$F_1 = F_2 + F_3 \tag{7.2}$$

Assume that all sensor failures occur independently and randomly and consider that the mass flow of all flows can be measured using sensors (k = $S1$, $S2$, $S3$, $S4$, $S5$, $S6$) whose sensor reliability are all assumed to be 0.8 at time t. Therefore, the reliability[1] of estimating variable F_1 at time t in each case is summarized in Table 7.1.

Table 7.1, shows that in all three cases (sensor reliability, hardware reliability and functional reliability) $r_{F_1}(t) > 0$, thus, variable F_1 is always observable. Moreover, in the case of hardware reliability, the reliability of estimating variable F_1 is greater

[1] If $P1$ and $P2$ are the probabilities of two dependent events then: $(P1) or (P2) = P1 + P2 - P1.P2$

than the reliability of each one of the sensor measuring it, showing the existence of more than one way to estimate F_1. Thus, $r_{F_1}(t)$ is directly related to the degree of redundancy and may still be estimated even if one sensor fails. However, in cases of sensor reliability and functional reliability no sensor fails is admitted. These results illustrate the effect of sensor placement on the reliability of estimating process variable.

7.2.2.1 Hardware reliability

If more than one sensor is placed in a given measuring point, the resulting reliability is calculated by using union of disjoint products. This is possible since the sensors operability are independent. For example, if three similar sensors $S1$, $S2$ and $S3$ are placed in the same location. The resulting reliability is evaluated by analyzing the disjoint events: {(three sensors are operational) or (only 2 sensors are operational and one fails) or (1 sensor is operational and 2 sensors fail)}. A general expression for the hardware reliability evaluation is given by Eq. (7.3).

$$r_j^h(t) = 1 - \left(\prod_{k=1}^{K} (1 - r_k^s(t))^{n_{jk}} \right) \tag{7.3}$$

Where $r_j^h(t)$ is the reliability of estimating j at time t by using n_{jk} sensors of type $k = 1 \cdots K$ available in the catalogue. This set K contains sensors with different "sensor reliability" and cost. If no sensor is assigned to j ($n_{jk} = 0,$) then $r_j^h(t) = 0$, whereas if only one sensor is assigned $r_j^h(t) = r_k^s(t)$.

In the sensor network design/retrofit procedure presented in this chapter, the hardware reliability is first calculated using Eq. (7.3). The resulting reliability is associated to a virtual sensor to be used for the process variables estimation reliability. The cost of this virtual sensor is equal to the sum of the costs all sensors k used for the hardware reliability.

7.2.2.2 Functional reliability

If the relationships between process variables are considered, the value of a process variable j can also be indirectly estimated through the process model. Assuming that:

1 - the equations used for estimating process variable j are linearly independent (i.e. that is each measured variable can be written in such a way that each one belongs to a unique equation) and

2 - the process variable j is not directly measured,

then, the reliability of estimating j at time t is calculated as follows:

$$r_j(t) = \bigcup_{e \in Q_j} \left(\prod_{j' \neq j=1}^{V_e} r_{j'}^h(t) \right) \tag{7.4}$$

where Q_j is the set of independent equations that can be used to estimate the value of process variable j. Each equation e belonging to Q_j has a sensor assigned to each one of its variables except in j. V_e is the cardinal of e minus one. This equation can be also applied if the probabilities of sensor failure are not equal.

However, this is not the general case. It may not be easy to deduce a generic analytical expression because the reliability depends on the process model structure. Moreover, the inter-dependency of process model relationships and the presence of recycling streams generates additional complexity. Therefore, an analytical expression for the functional reliability when dependent equations are involved is quite difficult to generate despite the logic behind is quite clear.

For that reason the reliability evaluation is obtained algorithmically. In the case of linear or linearized process models the quantitative process knowledge can be represented by the incidence matrix **A**, which, in addition to the sensor assignment, n_{jk}, define the reliability $r_j(t)$:

$$r_j(t) = f(n_{jk}, \mathbf{A}) \tag{7.5}$$

This work presents a practical and efficient algorithm to obtain an analytical expression for process variable estimation reliability.

The different steps involved are as follows:

- Identify the model structure through the matrix **A** of the process under consideration.
- Generate the matrix $\mathbf{A}^*$ from **A**. Firstly all the possible combination of the **A**'s rows are obtained. The total number of these combinations is equal to: $\sum_{r=1}^{\mathbf{A}_r} \mathrm{C}(r, A_r)$ where A_r is number of rows in **A**. Secondly, for each generated combination its corresponding rows in **A** are summed.
- Identify the set of key process variables as those whose minimum allowable reliability is non-null.
- Determine the different equations permitting the estimation of variable j (the number of these equations is equal to the degree of redundancy).

  Given matrix $\mathbf{A}^*$ its rows that does not contains explicitly the variable j are directly removed. Furthermore, some infeasible combinations might occur when generating $\mathbf{A}^*$, therefore they have to be discarded. For such purpose the set of variable V_r permitting the estimation of j from row r is stored, then if a set V_{r1} contains completely other set V_{r2}, V_{r1} is discarded. The resulting matrix $\mathbf{A}_j^*$ contains the different way to estimate j.

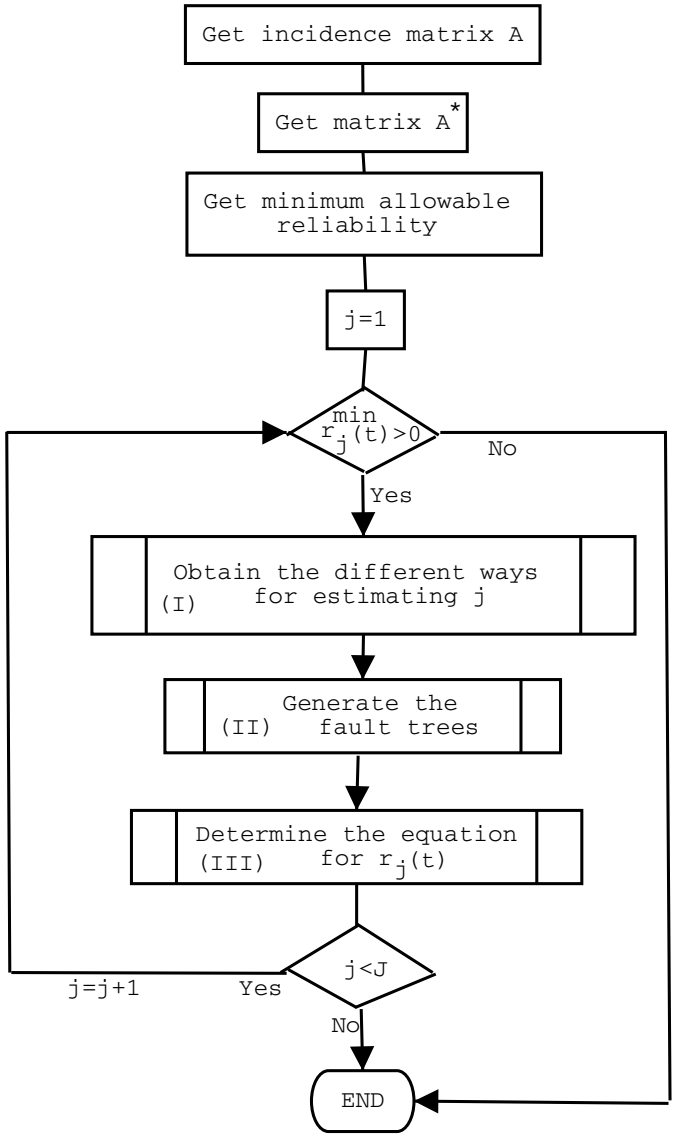


Figure 7.3: The proposed algorithm for variable estimation reliability.

- Generate the fault tree for each matrix $\mathbf{A}^*{}_j$. Fault trees provide a logical modeling framework for analyzing and representing the interactions between component reliabilities. Different programs are available for creating and supporting fault trees. This steps allows to generate a formula for estimating the different way to estimate j.

- Generate the analytical expression for $r_j(t)$ from the logical one.

It is worth mentioning that this analytical expression has to be provided prior to the sensor network design/retrofit procedure. If a process variable is not measured the sensor associated to it has null reliability value. The advantage of the proposal over the existing techniques based on graph-theory is its practical aspects.

For illustrating the steps of the previous algorithm consider the plant presented in Figure 7.4. This case study Ruiz et al. (2001) corresponds to a petrochemical plant consisting of a train of two distillation columns where a group of n-paraphines are separated from kerosene. For this plant the pure flow balance is expressed as follows:

$$F_1 = F_2 + F_3 \tag{7.6}$$

$$F_3 = F_4 + F_5 \tag{7.7}$$

and in a matrix form is given by Eq. (7.8) where the columns represent the flow-rates variables and the rows represent the mass balance equations around the stripper and the re-distillation units.

$$\mathbf{A} = \begin{pmatrix} 1 & -1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 \end{pmatrix} \tag{7.8}$$

The number of combination that can be generated from the above matrix $\mathbf{A}$ are three since $A_r = 2$. These combination are represented by the set $[1,2,(1,2)]$. That is matrix $\mathbf{A}^*$ can be obtained by selecting the first row of $\mathbf{A}$, the second row of $\mathbf{A}$ and summing the first and second row of $\mathbf{A}$.

$$\mathbf{A}^* = \begin{pmatrix} 1 & -1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 \\ 0 & 0 & 1 & -1 & -1 \end{pmatrix} \tag{7.9}$$

If the reliability of process variable F_1 and F_2 are required their corresponding $\mathbf{A}^*{}_j$ have to be generated. In this case their corresponding fault trees are shown in Figure 7.5.

The mapping between the generated trees and the probabilities allows to represent analytically, the reliability of estimating F_1 and F_2 at a time t as expressed in Eqs. (7.10) and (7.11).

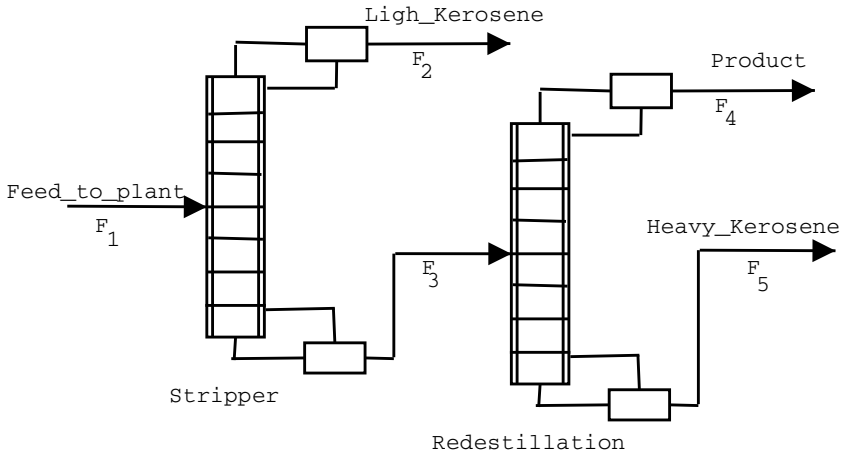


Figure 7.4: A simplified petrochemical plant.

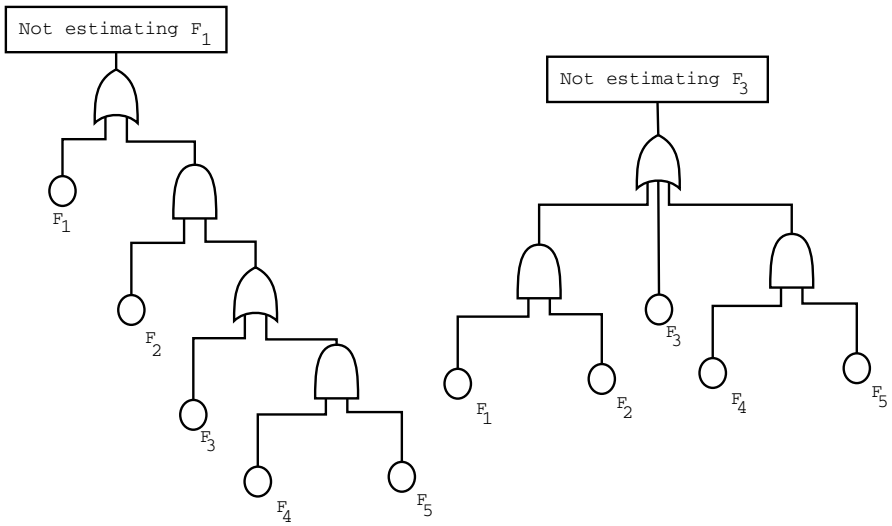


Figure 7.5: Fault trees for process variable estimation reliability.

$$r_{F_1}(t) = (r^h_{F_2}(t) and \left(r^h_{F_3}(t) or \left(r^h_{F_4}(t) and r^h_{F_5}(t)\right)\right)) or (r^h_{F_1}(t)) \tag{7.10}$$

$$r_{F_3}(t) = \left(r^h_{F_1}(t) and r^h_{F_2}(t)\right) or \left(r^h_{F_4}(t) and r^h_{F_5}(t)\right) or (r^h_{F_3}(t)) \tag{7.11}$$

Considering the sensor networks given in Table 7.2 for the plant presented in Figure

Table 7.2: Effect of sensor network on $r_{F_1}(t)$ for functional reliability

process variables	F_1	F_2	F_3	F_4	F_5
sensor reliability at time t	0.8	0.8	0	0.8	0.8

7.4, and taking into account that the equations estimating F_3 are independent, the reliability of estimating F_3 at time t can be obtained using Eq. (7.4) and is equal to 0.87.
Finally, any attempt to maximize the reliability of a particular variable may result in other variables to be unobservable. Indeed, eliminating a sensor that belongs to several equations impacts more negatively in the sensor network reliability than eliminating a sensor that only intervenes in a unique equation.
Consequently, it is necessary to determine the sensor network reliability $R(t)$. For such purpose the sensors reliability and variables estimation reliability should undoubtedly be considered.

7.2.3 Sensor network reliability

Luong et al. (1994) proposed an analytical expression for evaluating the sensor network reliability. Considering that all sensors reliability $r_k^s(t)$ are all equal to p, the reliability of the sensor network is given by Eq. (7.12).

$$R(t) = \sum_{n=0}^{d} \alpha_n . (1-p)^n . p^{d-n} \tag{7.12}$$

where the coefficients α_n give the number of configurations admitting n sensor breakdowns while ensuring the system observability, and d is the minimum degree of redundancy of all variables.
This reliability expression is based on the minimum degree of redundancy of all the redundant process variables, thus might discard some sensor failures despite that the system redundancy admit them. This procedure also presents the impossibility of evaluating the reliability of a subset of the total variables of interest. Alternatively, the sensor network reliability can be given by the minimum of the process variable reliabilities, following the proposal of Ali and Narasimahan (1993).

$$R(t) = \min_j \left(r_j(t) \right), \forall j \tag{7.13}$$

This expression is based on the philosophy that a chain can not be stronger than its weakest link. This conclusion is slightly similar to the proposal of Luong et al.

(1994) considering that network reliability is fixed by the minimum degree of redundancy of the process variables. Ali and Narasimahan (1993) exploit all the system redundancy by using graph-theory; however, their proposal is not applicable for a subset of all variables and requires an elevated computational effort when considered from a mathematical programming perspective (the optimization of the sensor network design problem).

In the proposed approach the network reliability is also defined by the minimum estimation reliability of all variables. The evaluation of the $r_j(t)$ is given in the previous section.

Consider the plant given in Figure 7.4, where the reliability of flow process variables F_1 and F_3 are of interest and assume that all variables are measured with similar sensor $r_j^h(t) = 0.8$.

1. If applying the proposal of Luong et al. (1994) only two scenarios are considered. The first one is that all the five sensors are operational leading to a reliability of 0.327. The second one is that one sensor fail while the four remaining sensors are still operating, thus leading to a reliability of 0.409. Therefore, the reliability $R(t)$ is equal to 0.73, resulting form the summation of the two scenarios.

2. In this work more scenarios are considered admitting more than two sensor breakdowns. These scenarios can be easy deduced from the analysis of the trees of Figure 7.5. Alternatively, by using Eqs. (7.10) and (7.11) it is possible to get $r_{F_1}(t) = 0.96$ and $r_{F_3}(t) = 0.97$ and their minimum is $R(t) = 0.96$.

Table 7.3 summarizes the results of network reliability values. It can be seen that this work provides more accurate value for the network reliability than the proposal of Luong et al. (1994) and is equal to the proposal of Ali and Narasimahan (1993). Despite that both proposals provide the same reliability values the proposal of Ali and Narasimahan (1993) is hard to incorporate into an optimization procedure due to the inherent difficulty to handle the graph-theory concepts.

7.3 Generic design of reliable sensor networks

7.3.1 Sensor networks model

The design of measurement systems addressed with the goal of reliability was earlier considered by Ali and Narasimahan (1993). This proposal is as follows: given a minimum number of sensors N^*, the sensor networks that ensure system observability

Table 7.3: Comparing system reliability evaluation approaches.

Sensor network reliability at time t	$R(t)$ value
Luong et al. (1994) (Eq. (7.12))	0.73
Ali and Narasimahan (1993)	0.96
Proposed approach (Eqs. (7.10,7.11,7.13))	0.96

are determined based on graph-theory. Among all the networks the one that offers the maximum system reliability is selected. This approach does not consider directly the sensor cost and does not guarantee the desired reliability levels on specific variables. These points were addressed later by Bagajewicz and Sánchez (2000a). Additionally, these authors transform the model proposed by Ali and Narasimahan (1993) into a mathematical programming model as follows:

$$\max R(t) \tag{7.14}$$

subject to:

$$\sum_{\forall j} q_j = N^* \tag{7.15}$$

$$E_i(q) = 1, \forall i \tag{7.16}$$

$$q_j = (0,1), \forall j \tag{7.17}$$

In the above sensor placement model, the first constraint fixes the number of sensors. If observability is focused, N^* is equal to the number of process variables minus the number of independent equations describing the process flow-sheet and, if redundancy is focused, a number greater than N^* has to be selected. The observability requirement is mathematically expressed by imposing that the degree of estimability $E_i(q)$, of all variables to be equal to one (see Eq. (7.16)). The extension of this model to include cost and individual variable estimation reliability requirement is proposed in Bagajewicz and Sánchez (2000a) as follows:

$$\min \sum_{\forall j} c_j q_j \tag{7.18}$$

subject to:

$$r_i(t) \geq r_i^{min}(t) \forall i \in M_R \tag{7.19}$$

$$q_j = (0,1), \forall j \in M_J \tag{7.20}$$

Where M_J is the set of locations where sensors can be placed, M_R is the set of variables whose reliability is to be constrained and c_j is the cost of measuring j.

However, this model only allows specifying the possibility to place or not a sensor in a given location ($q_j = 0,1$), it does not consider the hardware redundancy, and assumes that the reliability of the system is implicitly included in the reliability of variable estimation.
These considerations have been included in this work. Additionally the values of $r_j^{min}(t)$ are selected according to the redundancy requirements. If the value of this minimum value $r_j^{min}(t)$ is non-null the variable j is observable and if its value is larger than $r_j^h(t)$ the variable j is redundant. The minimum value of the system reliability $R^{min}(t)$ is selected using a Pareto analysis of the cost/reliability trade-off. Furthermore, the decision variable is not to place or not a sensor, but to select among a catalogue the number n_{jk} of sensors (multiplicity) of a certain type k and reliability $r_k^s(t)$ to be assigned to a given process variable/measuring point j:

$$\min_{n_{jk}} \sum_{\forall k} c_k n_{jk} \tag{7.21}$$

subject to:

$$R(t) \geq R^{min}(t) \tag{7.22}$$

$$r_j(t) \geq r_j^{min}(t) \tag{7.23}$$

A generic objective function that contemplates both the design and retrofit proposed by Benqlilou et al. (2004) can also be used instead of Eq. (7.21) to consider retrofitting case:

$$\min_{n_{jk}} \left(\sum_k c_k^c . \Theta \left(\sum_j n_{jk} - \sum_j n_{jk}^0 \right) + \sum_k \sum_j c_k^i . \left| n_{jk} - n_{jk}^0 \right| \right) \tag{7.24}$$

where n_{jk}^0 is the number of already installed sensors of type k in measuring point j; c_k^c and c_k^i are the capital and installation cost of sensor type k, and Θ is the Heaviside function. The first part of Eq. (7.24) evaluates the capital cost taking into account that for each sensor type k and for all measuring point, j, the sum of allocated sensors of type k is subtracted from the already installed sensors of the same type (if $n_{jk}^0 = 0, \forall j$ $\forall k$, this is a design problem). The second part of Eq. (7.24) calculates the installation cost and the absolute value is applied to allow including the un-installation cost.

Eq. (7.24) allows considering physical sensors and inferential sensors. Inferential sensors have shown to be applicable and have been successfully incorporated to industrial application Ruiz et al. (2001). The reliability of the inferred variable is calculated using the methodology presented earlier in section 7.2. Whereas the installation cost and capital cost of inferential sensors are null.

This MINLP problem is difficult to be addressed by mathematical programming techniques because of the combinatorial tree (type, multiplicity), the form of the objective function and the constraints evaluation. This makes very attractive the use of meta-heuristic techniques such as genetic algorithms. The next section describes

and discusses the use GA to solve the sensor network design/retrofit problem just formulated.

7.3.2 Sensor network solution based on genetic algorithms

Recently, different works (Sen et al. (1998); Musulin et al. (2003)) have shown the ability of GA to solve sensor network optimization problems. The most important aspect for a successful application of the GA techniques is the codification of the individuals (i.e. possible solutions, chromosomes). This codification contains the necessary information to evaluate the objective function (fitness function) as well as to check the feasibility of the individuals.

The fundamental aspects to contemplate in this codification are the selection and placement of measuring devices. Concretely, the codification includes aspects of sensor networks: reliability, cost, location and multiplicity. Additionally, the constraints on the allowable number of sensors per location as well as the technical feasibility of placing a particular sensor in a given point are also managed in this codification. Therefore:

- each chromosome (individual) represents a sensor network,
- each gene in the chromosome corresponds to a measuring point,
- the value of each allele in each gene reflects the multiplicity of a specific sensor type (see Figure 7.6), and
- each sensor type is associated with a sensor cost and reliability.

Based on this codification, an initial population is generated randomly. Then by means of the selection, crossover and mutation operators, new generations are produced. For each iteration the population will contain a fraction $(1/10)$ of the best individual of the precedent generation. The population size is set $N_{ind} = 1000$ using the roulette wheal operator and $9/10$ are selected to be crossed. Two point crossover is applied with a probability of $P_c = 0.7$ and the rate of mutation is $P_m = P_c/(J.K)$ where $J.K$ is the length of the individual.

The feasibility function is evaluated for each individual of the generated population. This function verifies three constraints: network reliability, variables estimation reliability and input parameter consistency. If one of these constraints is not satisfied a very high value is assigned to the objective function (the total cost).

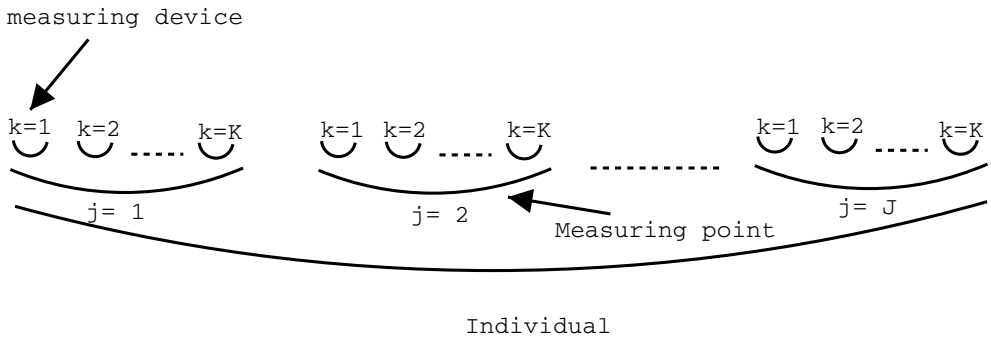


Figure 7.6: Genetic algorithm codification.

7.3.3 Ammonia plant case study

A simplified ammonia synthesis plant presented by Kretsovalis and Mah (1988) is considered for illustrating the performance of the solution methodology proposed when addressing different design/retrofitting situations. This case study (Figure 7.7) has been used in several sensor network design works addressing reliability (Bagajewicz and Sánchez (2000a); Ali and Narasimahan (1993); Ali and Narasimhan (1995)).

This plant consists of six units and eight streams $(F_f,\ f = 1 \cdots 8)$ with node 6 representing the plant mass-balance. Additionally, the presence of different cycles make the case more attractive for sensor network design and retrofitting purposes.

Consider, as in the work presented by Ali and Narasimhan (1995) that all the available sensors $(Sn,\ n = 1 \cdots 5)$ have the same reliability, $r_k^s(t) = 0.9$. Consider also five sensors $S1$, $S2$, $S3$, $S4$, $S5$ having respectively the costs 1500, 1700, 2000, 2300 and 2800. Additionally, assume that some sensor/variable assignation are prohibited as presented by dashes in Tables below. Finally, assume the minimum allowable reliability $r_j^{min}(t)$ for the 8 variables is set to 0.81.

The problem presented by these authors is successfully solved and results are presented in Table 7.4. The same values for the total cost and for the $r_j(t)$ are shown in the penultimate row of Table 7.4. Additionally, the values of n_{jk} obtained by the solution methodology proposed are also equal to their results. This solution is obtained in the first iteration of the GA procedure when minimizing the cost function given by Eq. (7.24).

If different sensor types with lower reliability, $r_k^s(t)$, are considered presented in Table 7.5. In order to satisfy reliability constraints, F_7 and F_8 are now measured instead of F_6 and additional sensors are required, as it should be expected. It can be seen a change in the selection/placement of measuring devices and an increase in

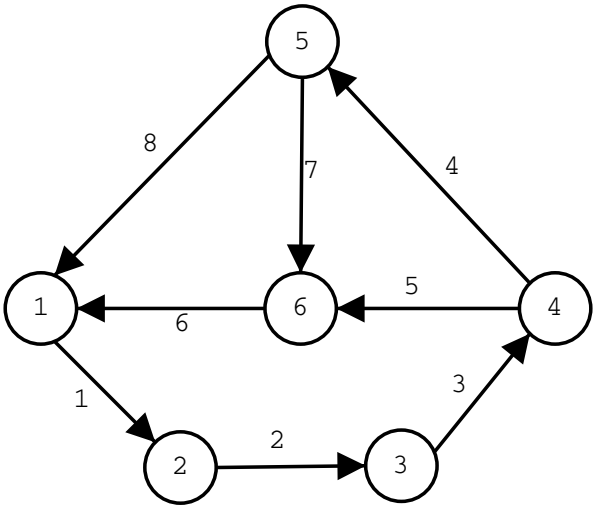


Figure 7.7: Simplified ammonia plant network.

Table 7.4: Design of reliable sensor networks using a unique sensor type.

n_{jk}	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	$r_k^s(t)$
$S1$	**1**	-	-	-	-	-	**0**	-	0.9
$S2$	-	-	-	-	**1**	-	-	-	0.9
$S3$	-	**0**	-	-	-	**1**	-	-	0.9
$S4$	-	-	**0**	-	-	-	-	-	0.9
$S5$	-	-	-	**0**	-	-	-	**0**	0.9
$r_j^{min}(t)$	0.81	0.81	0.81	0.81	0.81	0.81	0.81	0..81	**Total**
$r_j(t)$	**0.9**	**0.9**	**0.9**	**0.81**	**0.9**	**0.9**	**0.81**	**0.81**	**Cost**
c_j	**1500**	**0**	**0**	**0**	**1700**	**2000**	**0**	**0**	**$5200**

the total instrumentation cost. These type of solutions are only obtained when using general models contemplating sensor type selection.

Moreover, if the sensor reliability of these sensors is slightly increased (Table 7.6) the total cost is reduced by a different assignment, n_{jk}, while satisfying reliability constraints. Furthermore, this solution also produces a slight improvement on the variables reliability $r_j(t)$.

From this analysis it can be concluded that by permitting the incorporation of

Table 7.5: Flexibility based sensor type for designing reliable sensor networks.

n_{jk}	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	$r_k^s(t)$
$S1$	**1**	-	-	-	-	-	**1**	-	0.70
$S2$	-	-	-	-	**1**	-	-	-	0.75
$S3$	-	**0**	-	-	-	**0**	-	-	0.80
$S4$	-	-	**0**	-	-	-	-	-	0.85
$S5$	-	-	-	**0**	-	-	-	**1**	0.90
$r_j^{min}(t)$	0.81	0.81	0.81	0.81	0.81	0.81	0.81	0.81	**Total**
$r_j(t)$	**0.84**	**0.84**	**0.84**	**0.82**	**0.86**	**0.82**	**0.84**	**0.93**	**Cost**
c_j	**1500**	**0**	**0**	**0**	**1700**	**0**	**1500**	**2800**	**$7500**

Table 7.6: Flexibility based on selecting sensor type.

n_{jk}	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	$r_k^s(t)$
$S1$	**0**	-	-	-	-	-	**1**	-	0.75
$S2$	-	-	-	-	**1**	-	-	-	0.80
$S3$	-	**1**	-	-	-	**1**	-	-	0.85
$S4$	-	-	**0**	-	-	-	-	-	0.90
$S5$	-	-	-	**0**	-	-	-	**0**	0.95
$r_j^{min}(t)$	0.81	0.81	0.81	0.81	0.81	0.81	0.81	0.81	**Total**
$r_j(t)$	**0.85**	**0.85**	**0.85**	**0.85**	**0.92**	**0.94**	**0.92**	**0.86**	**Cost**
c_j	**0**	**2000**	**0**	**0**	**1700**	**2000**	**1500**	**0**	**$7200**

different sensor types to the design of reliable sensor network it is possible to achieve the reliability requirements at a lower cost. From a general point of view, more flexibility is given to the decision maker.

Following, another scenario is adopted for illustrating the importance of including the hardware redundancy and sensor network reliability for designing reliable sensor networks. For such purpose consider that lower sensor reliability is considered for all the available sensors ($r_k^s(t) = 0.75$) and greater "minimum sensor network reliability" is requested, 0.90 for F_1, F_2, and F_3 and 0.85 for the rest of variables.

The results of such scenario are illustrated in Table 7.7 where the total cost that has to be inverted for satisfying the reliability constraints is $8700.

If multiplicity is next considered (Table 7.8) results are considerably improved.

Table 7.7: System reliability and reliable design of sensor network.

n_{jk}	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	$r_k^s(t)$
$S1$	**1**	-	-	-	-	-	**1**	-	0.75
$S2$	-	-	-	-	**1**	-	-	-	0.75
$S3$	-	**1**	-	-	-	**1**	-	-	0.75
$S4$	-	-	**0**	-	-	-	-	-	0.75
$S5$	-	-	-	**0**	-	-	-	**0**	0.75
$r_j^{min}(t)$	0.90	0.90	0.90	0.85	0.85	0.85	0.85	0.85	**Total**
$r_j(t)$	**0.93**	**0.93**	**0.93**	**0.93**	**0.89**	**0.89**	**0.89**	**0.93**	**Cost**
c_j	**1500**	**2000**	**0**	**0**	**1700**	**2000**	**1500**	**0**	**$8700**

Table 7.8: Hardware redundancy and reliable design of sensor networks.

n_{jk}	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	$r_k^s(t)$
$S1$	**2**	-	-	-	-	-	**1**	-	0.75
$S2$	-	-	-	-	**1**	-	-	-	0.75
$S3$	-	**0**	-	-	-	**1**	-	-	0.75
$S4$	-	-	**0**	-	-	-	-	-	0.75
$S5$	-	-	-	**0**	-	-	-	**0**	0.75
$r_j^{min}(t)$	0.90	0.90	0.90	0.85	0.85	0.85	0.85	0.85	**Total**
$r_j(t)$	**0.93**	**0.93**	**0.93**	**0.86**	**0.89**	**0.89**	**0.89**	**0.86**	**Cost**
c_j	**3000**	**0**	**0**	**0**	**1700**	**2000**	**1500**	**0**	**$8200**

That is for satisfying the same reliability constraints a lower investment is needed ($8200). This is due to assignment of two sensors of type $S1$ to F_1 instead of assigning $S3$ to F_3.

For retrofitting purposes, consider an operating plant whose instrumentation (the one given in Table 7.8) has to be updated to achieve new tighter reliability constraint ($r_j^{min}(t)$). Furthermore, assume that any sensor/location assignation is possible (i.e. each sensor k can be assigned to each location j), thus increasing the number of decision variables.

Table 7.9 presents the solution obtained when the installation costs are assumed zero. Even in this case requiring a more important computational effort, the solution is obtained by the GA in few generations.

Table 7.9: Retrofitting of reliable sensor networks.

n_{jk}	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	$r_k^s(t)$
$S1$	**1**	**0**	**1**	**1**	**0**	**0**	**0**	**1**	0.75
$S2$	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	0.75
$S3$	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	0.75
$S4$	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	0.75
$S5$	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	0.75
$r_j^{min}(t)$	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	**Total**
$r_j(t)$	**0.98**	**0.98**	**0.98**	**0.972**	**0.972**	**0.972**	**0.98**	**0.971**	**Cost**
c_j	**0**	**0**	**1500**	**1500**	**0**	**0**	**0**	**1500**	**$4500**

7.4 Conclusions

The sensor network design is addressed in this chapter at two levels: reliability evaluation and optimization. Firstly, a practical and efficient way for evaluating the reliability of estimating variables is presented based on combining both quantitative process knowledge and fault tree analysis. Secondly, defining the system reliability, a mathematical programming model for plant instrumentation is formulated and solved by means of genetic algorithms. The solution of this problem provides the placement of the number and type of measuring devices that have to be added and/or reallocated in an operating plant in order to lead the plant performance to the desired reliability values at a minimum cost.

The approach of this work provides to the decision maker the flexibility to analyze the different alternatives that could be considered when designing a reliable sensor network. This flexibility is illustrated in the different examples addressed including sensor type selection, hardware reliability, reliability constraints on both, network and individual variables, and the retrofitting capability.

Nomenclature

$\mathbf{A}$	incidence matrix
$\mathbf{A}_j$	incidence matrix focusing variable j
$\mathbf{A}^*$	matrix resulted for linear combination of $\mathbf{A}$'s rows
A_r	number of rows in $\mathbf{A}$
c_j	total cost of sensing variable j
c_k	total cost of sensor k
c_k^j	installation (or un-installation) cost of sensor k
c_k^c	capital cost of sensor k
d	minimum degree of redundancy of all variable
d_j	degree of redundancy of variable j
E_j	degree of estimability of variable j
e	equation that belong to Q where all its variables are measured except one
$\mathbf{J}$	set of process variable
$\mathbf{K}$	set of available sensor in the catalogue
$\mathbf{M}_R$	set of variables for which reliability is required
$\mathbf{M}_J$	set of variables where it is possible to place a sensor
N	number of sensors
N^*	minimum number of sensors
n	number of admitted sensor breakdowns
n_{jk}	number of sensor of type k assigned to location j
n_j^{min}	minimum allowable sensor for location j
n_j^{max}	maximum allowable sensor for location j
p	particular value of sensor reliability
Q_j	set of independent equation for estimating j
q_j	binary variable indicating if it is possible to assign a sensor to j or not
$R(t)$	sensor network (system) reliability
$R(t)^{min}$	minimum value on the sensor network reliability
$r_j(t)$	probability of estimating variable j at time t
$r_k^s(t)$	probability of non-failure of the sensor k at time t
$r_k^h(t)$	reliability resulted from placing a set of sensors in the same location
$r_j^{min}(t)$	minimum reliability value on variable estimation of variable j
t	time at which the sensor network will be designed
V_e	cardinal of e minus one
V_r	set of variables in the row r

Operator

$C(r, A_r)$ combinatorial coefficient defined as: $r!/[A_r!(r-A_r)!]$

Greek letters

α_n number of configuration with a value of n and ensuring system observability
$\Theta(x)$ heaviside function (equal to x if $x \geq 0$ and equal to 0, otherwise)
λ_k constant representing the rate of sensor failure measuring variable k

Subscripts

i variables whose reliability is required
j location where it is possible to place a variable
k sensor type
r row of a matrix

Chapter 8

Decision-making strategy for sensor networks design/retrofi t

Abstract

This chapter presents a decision-making framework for design and retrofit of sensor networks as well as a general strategy to correlate the cost and performance of the different sensor arrangements (number, placement, etc) that may be systematically analyzed.

The analysis of the steps required for coping with the sensor placement problem, in design or retrofit cases, implies the identification of the information flows involved. Such a conception allows a modular design of a CAPE tool for computer aided sensor network design/retrofit. An interface specifications has been proposed for this CAPE tool and a prototype has been developed for validation purposes.

The decision-making strategy presented is independent of the particular algorithms and procedures adopted. Hardware redundancy, steady-state/dynamic processes, design/retrofit and the catalogue of different available sensors are all taken into account, thus leading to a generic framework able to follow different policies for sensor placement. Seeking the synergy given by re-usability and standardization, the sensor placement tool has been developed following the CAPE-OPEN guidelines, allowing the integration of other software modules such as Data Reconciliation and Optimization.

8.1 Introduction

The sensor placement problem may be regarded as a constrained optimization problem for minimizing the sensor network cost. The constraints are inequalities defining

the upper and lower bounds on system performance such as accuracy and/or reliability for each process variable measurements. Thus, given the costs c_j of the sensors required for measuring each process variable j, the sensor placement problem can be formulated as:

$$\min \sum_j c_j b_j \tag{8.1}$$

subject to:

$$\sigma_j^2(j) \leq \sigma_j^{2*} \tag{8.2}$$

$$r_j(t) \geq r_j(t)^* \tag{8.3}$$

where the accuracy σ_j^2 and reliability $r_j(t)$ at a time t are both functions of the sensor assignment:

$$b_j = \begin{cases} 1 & if\ variable\ j\ is\ measured \\ 0 & otherwise \end{cases} \tag{8.4}$$

Alternatively, it is possible to choose to minimize the weighted estimated accuracy (the estimation accuracy can be evaluated through Data Reconciliation) of the involved process variables subjected to the upper bound on the cost of sensor network:

$$\min_{\sigma_j} \sum_j \alpha_j \sigma_j^2 \tag{8.5}$$

subject to:

$$\sum_j c_j b_j \leq c^{max} \tag{8.6}$$

The sensor placement problem presents a particular feature with the difficulty of determining the threshold value of the inequality constraints Eqs. (8.2), (8.3) and (8.6). A partial way to overcome this drawback is considering the problem as a multi-objective unconstrained optimization problem as follows.

$$\min \sum_j \alpha_j \sigma_j^2 + \beta_j c_j b_j \tag{8.7}$$

However, this formulation poses a trade-off between the performance (e.g. accuracy) of each possible sensor network and its cost given by the values of the weight parameters α_j and β_j in Eq. (8.7), which are also difficult to determine.

This work addresses this fundamental trade-off by proposing an information framework for aiding an expert user in the decision-making procedure. Furthermore, the system allows:

- implementation into a modular application an objective function for evaluating performance. The sensor placement constraints (satisfying the desired accuracy or reliability) are usually represented algorithmically. Therefore, generating the corresponding analytical equations for optimization purposes may be very arduous and results in a rigid solution approach to the sensor placement problem since these equations are strongly problem dependent.

- reducing the different sensor placement criteria by developing more generic sensor placement constraints. For example, this work shows that increasing the sensor network reliability indirectly increases the sensor network accuracy.

- selection of a sensor network among a set of possible networks by using the profile of the system performance versus the corresponding cost. This point is useful and practical for evaluating the necessary investment to achieve a desired performance. Thus, this information is very valuable for the decision-making.

Additionally, this work also considers:

- different measuring device types (cost and performance). This consideration is the basis for a more generic sensor placement problem including the reallocation, new purchase and design of sensor placement problems.

- hardware redundancy (more that one measuring device per measurement point), which is undertaken in a novel way by including additional inequality constraints to the sensor placement problem formulation.

- an extension for dealing with dynamic cases, incorporating the results obtained in Chapter 6.

Thus, this work presents a specification for this strategy and a CAPE tool that has been developed following the CO guidelines (CAPE-OPEN Standard 1.1 (2002)). Additionally, the unified modeling language (Muller (1997)) description of sequence, interface and component diagrams for design/retrofit is produced and validated providing the corresponding prototypes. Several CAPE-OPEN standard interface specifications, such as Data Reconciliation and MINLP, have been adopted since the sensor placement problem shares several features with these techniques in terms of the information flows.

8.2 Design strategy

8.2.1 Information flows

8.2.1.1 Input

The process model is a set of equations that correlate and bound process variables and it is a prerequisite for placing sensors in a given process plant. It may be given by an explicit set of mathematical equations[1] or as black box model (e.g. Artificial Neural Network model). This process model allows both the generation of possible sensor networks and the evaluation of the system performance.

The number of possible sensor networks may be limited according to the set of parameters given by Eqs. (8.8) and (8.9).

$$M^{min} = \left\{ m_j^{min} \right\} \tag{8.8}$$

$$M^{max} = \left\{ m_j^{max} \right\} \tag{8.9}$$

Vectors M^{min} and M^{max} both of size J set the minimum and maximum number of measuring devices that may be allocated to the $j(=1...J)$ process variables involved in a given process model (multiplicity or hardware redundancy).
Dealing with multiplicity requires a new integer variable n_j instead of the binary variables b_j for describing the sensor assignment:

$$(b_j = \{0,1\}) \rightarrow (n_j \in N) \tag{8.10}$$

Thus, sensor assignment is limited by maximum and minimum multiplicity values:

$$m_j^{min} \leq n_j \leq m_j^{max}, \forall j \tag{8.11}$$

As an illustrative example, consider the adiabatic process in Figure 8.1, where $F1$ and $F2$ are mass flow rates, L denotes the tank level and, $T1$, $T2$ and T temperatures (heat capacity, density (ρ), and tank section (s), are assumed constants). The process model, given by the mass balance and the enthalpy balance is given by the following equations:

$$F1 - F2 = \rho.s.\frac{dL}{dt} \tag{8.12}$$

$$T1F1 - T2F2 = \rho.s.L.\frac{dT}{dt} + \rho.s.T.\frac{dL}{dt} \tag{8.13}$$

A possible mapping of process variables can be set by the following vector:

$$VariableMapping \Rightarrow \{L, F1, F2, T, T1, T2\}$$

[1] For instance through a standard CO interface: the Equation Set Object (ESO).

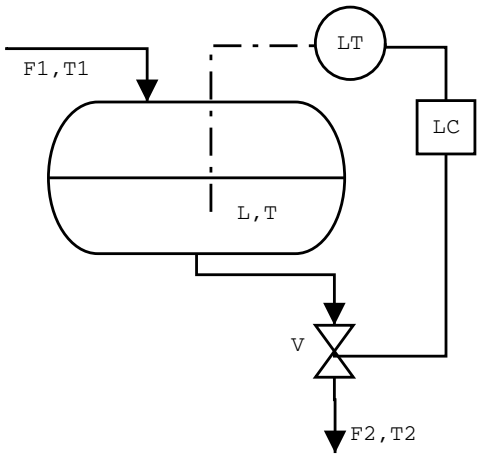


Figure 8.1: Scheme of an adiabatic process for mapping illustration.

lets the corresponding multiplicity bounds be:

$$M^{min} = \{1,0,0,0,0,0\} \tag{8.14}$$

$$M^{max} = \{1,2,1,1,1,1\} \tag{8.15}$$

Vectors M^{min} and M^{max} impose that the measurement of level L is obligatory while process variables $T, T1, T2$ and $F2$ could be either measured once or not. However, flow $F1$ can be measured at maximum twice.

Manipulation of the lower and upper bounds of Eq. (8.11) allows introducing the information about feasible, infeasible and obligatory measurement points and their maximum multiplicity.

In general, the two main cases are as follows:

- if $m_j^{min} = m_j^{max} = p$, where $p \in N$ the measurements of the process variable j is obligatory and has to be measured p times, (due to safety or reliability reasons). In the special case if $p = 0 \Rightarrow m_j^{min} = m_j^{max} = 0$, the measurement of the process variable j is infeasible (due to cost or operation limitations).

- if $m_j^{min} = p$ and $m_j^{max} = q$, where $\{p,q\} \in N, q > p$ the number of sensors used for measuring the j^{th} process variable is not known a priori, it is a decision variable which value pertain to the $\{p,...,q\}$ set. In general, industrial practice (including safety aspects) requires only a couple of sensors per measurement point.

The number of decisions associated to each process variable (i.e. number of possible sensors per measurement point) is given by:

$$\Delta M = \{\delta m_j\} \tag{8.16}$$

where:

$$\delta m_j = m_j^{max} - m_j^{min} + 1, \forall j = 1,..,J \tag{8.17}$$

The a priori calculation of δm_j provides an idea about the dimension of the problem, since the number of "positional" networks, P, that may be generated is given by:

$$P = \prod_j \delta m_j \tag{8.18}$$

Positional network, i, is referred to the assignment of a certain number of sensors of the same type (i.e. assuming all sensors having the same characteristics and cost) to a process variable j. Thus, each positional network i involves a number of sensors of the same type given by the summation:

$$\sum_j n_{ij} \tag{8.19}$$

When incorporating the consideration that different sensors k can be used to measure the same process variable j, then, each sensor network i is characterized by means of s_i, while the superset Υ defines all the networks considered:

$$s_i = \left\{n_{ijk}\right\} \in N; \Upsilon = \{s_i\} \tag{8.20}$$

The set of potential sensors K for measuring j is an information part of the catalogue. This information that should be carefully introduced to the formulation of the sensor placement is in the form of matrix R=$\left\{r_{jk}\right\}$ whose elements are binary variables defined as follow:

$$r_{jk} = \begin{cases} 1 & if\ sensor\ k\ can\ measure\ process\ variable\ j \\ 0 & otherwise \end{cases} \tag{8.21}$$

Assuming there are two types of flow-meters FM^a and FM^b and one type of measuring device for temperature (TM) and level (LM), a matrix R is obtained to characterize the feasibility of assigning measuring devices to measuring points as shown in Table 8.1.

It is important to check the consistency between the information contained in R, M^{min} and M^{max}. Hence:

$$m_j^{min} \leq m_j^{max} \tag{8.22}$$

Table 8.1: Matrix allowing sensor/process variable assignation.

	FM^a	FM^b	TM	LM
L	0	0	0	1
$F1$	1	1	0	0
$F2$	1	1	0	0
T	0	0	1	0
$T1$	0	0	1	0
$T2$	0	0	1	0

$$m_j^{max} \leq r_{jk}.U_1 \tag{8.23}$$

where U_1 is a big number, (a U_1 value higher than 10 should be large enough).
The multiplicity is taken into account by adding the following inequalities:

$$m_j^{min} \leq \sum n_{ijk} \leq m_j^{max}, \forall k \tag{8.24}$$

$$n_{jk} \leq r_{jk}.U_2 \tag{8.25}$$

where U_2 is again a big number, in this case it is assumed equal to $\simeq m_j^{max}$. Finally, the criteria for placing measuring devices (e.g. investment cost, Eq. (8.26)) and/or reliability and/or accuracy, etc) has to be specified to evaluate and rank the different alternatives considered as well as to control the combinatorial explosion if possible.

$$cost\,(s_i) = \sum_j \sum_k c_k n_{ijk} \tag{8.26}$$

$$perf\,(s_i) = \Xi\left(n_{ijk}, ESO, \ldots\right) \tag{8.27}$$

8.2.1.2 Output

The information that allows a technician (or an optimization algorithm) to make reliable decisions when designing a plant instrumentation is composed by the cost (Γ), and performance (Π) of each feasible sensor network ($s_i \in \Upsilon$), defined by the type and number of measuring devices per each measuring point.

$$\Upsilon = \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_i \end{pmatrix} \qquad \Gamma = \begin{pmatrix} cost\,(s_1) \\ cost\,(s_2) \\ \vdots \\ cost\,(s_i) \end{pmatrix} \qquad \Pi = \begin{pmatrix} perf\,(s_1) \\ perf\,(s_2) \\ \vdots \\ perf\,(s_i) \end{pmatrix} \tag{8.28}$$

Knowing the type of process variable j to be measured one can get the corresponding range of available sensors from a catalogue. This range includes information such as: the sensor accuracy $\{\sigma_k^2\}$, reliability $\{r_k(t)\}$ and cost $\{c_k\}$.

8.2.2 Modular structure and modules interaction

8.2.2.1 Involved modules

The analysis of the information flows of the sensor placement problem has followed the existing trends. The specification proposed for the sensor placement module may be easily plugged into other standardized software components or use them if required.

The modules required by an open and flexible architecture can be classified according to their functionality. Mainly, five components have to be considered as illustrated in the component diagram (see Figure 8.2):

- Sensor Networks Generator (module I or M-I),
- System Performance Calculator (module II or M-II),
- Total Cost Calculator (module III or M-III),
- Catalogue (module IV or M-IV), and
- Process Model (module V or M-V).

In the UML^{TM} notation, the component diagram addressing the static implementation view of the sensor placement system is illustrated in Figure 8.2.

Additionally, the dependency relations between the different software components are shown. Module (I) is responsible for generating a set of feasible sensor networks $\Upsilon = \{s_i\}$ from the bounds given by M^{min} and M^{max}, process model, sensor characteristics $\{\sigma_k^2, r_k(t), c_k\}$ and sensor placement criteria. This task can be performed by an elaborated algorithm or by a simple enumeration. Despite the efficiency of these or future algorithms, this work highlights the necessity of such functionality. The networks generated may not necessarily pertain to the feasible space in the sense of meeting the performance objective, it is up to the end-user to remove the infeasible subset since their performances will be unacceptable. Furthermore, the end-user may need to add additional networks to the generated for evaluating their performance and/or cost or remove some of the generated ones.

The objective of module (II) is the evaluation of the system performance of networks generated by the use of the above functionality. This module offers the list

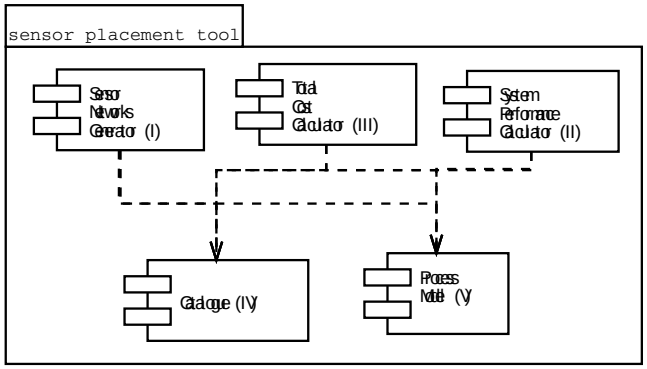


Figure 8.2: Proposed component diagram for the sensor placement system.

of performance or combination of performance it has evaluated and also the evaluation results. These performances may be mostly related to the data monitoring such as reliability, accuracy, gross error detectability, but can also deal with control and optimization by changing the system performance evaluation. Additionally, the performance should merge different criteria or offer a multi-objective solution.

Evaluating total cost of a generated set of networks is the objective of module (III). This module can also evaluate the cost corresponding to a retrofitting case.

The catalogue component or module (IV) acts as a database of measuring devices providing their characteristics. These characteristics include cost and the list of sensors characteristics that are mainly accuracy (σ_k^2), reliability ($r_k(t)$) and measuring range. Therefore, this module has process variable type (e.g. level, flow) as input and has the range of measurement device characteristics as output. It is important to note that this component is needed for evaluating the system performance, system cost as well as for generating the sensor networks.

Finally, module (V) allows to introduce the process model which is required for the performance of this application, since it is this model that represents the plant considered for sensor placement.

In principle the interaction of the specified modules will be initiated by a Client that could be either an end-user or a software application. The component diagram is complemented by the interface diagram (Figure 8.3). This diagram gives a static representation of the interfaces that the sensor placement components are required to expose, as well as their inter-operability. The interface diagram is of major significance for designing, specifying, and implementing the sensor placement architecture since it declares the methods that are externally exposed.

In the next section the dynamic interaction of the components involved will be

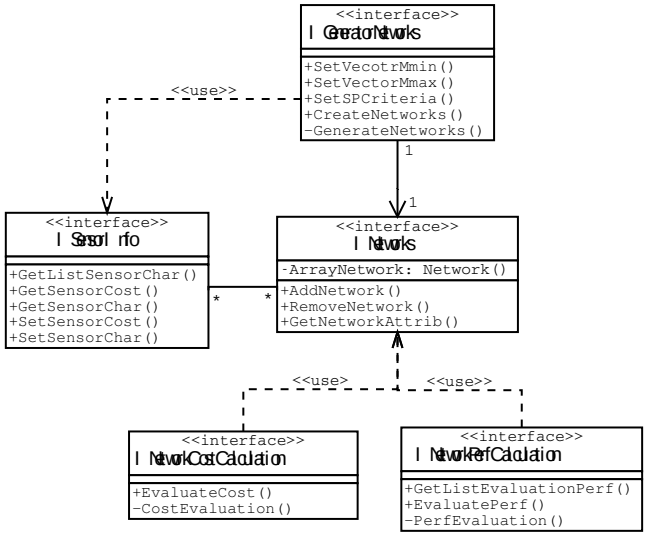


Figure 8.3: Proposed interface diagram for the sensor placement system.

analyzed.

8.2.2.2 Module interaction

First, the input information presented in section 8.2.1.1 is introduced to the sensor network system. This information is used by M-I to generate a set of networks Υ. The set of networks is then available to the end-user who decides to reduce or increment it prior to its evaluation. The performance evaluation M-II returns back to the end-user the system performance corresponding to a given set of networks. To fulfill the task, the component needs to interact with the process model M-V.

The last functionality is the evaluation of the cost of a set of networks (design/retrofit). To do so the total cost calculator M-III interacts with the catalogue M-IV to calculate the cost. The details of this interaction are described by means of the sequence diagrams in Figures 8.4 and 8.5.

The sequence diagram captures time-oriented dynamic behavior and interactions around sensor placement components and their relationships. Figure 8.4 proposes a a scenario for generating sensor networks whereas Figure 8.5 illustrates the temporal interaction for evaluating cost and performance of the generated sensors networks.

Finally, the resulting information is returned to the end-user by the sensor placement system (cost and performance) as summarized in section 8.2.1.2.

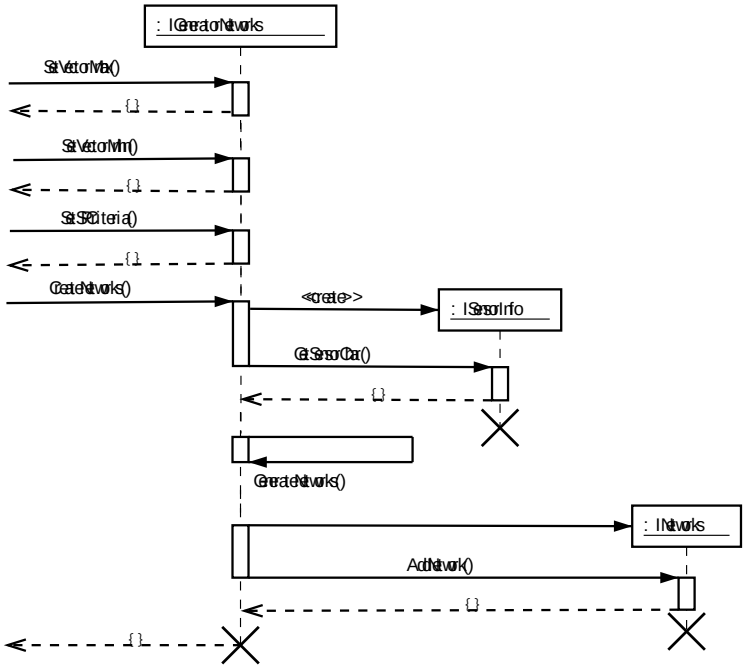


Figure 8.4: Proposed sequence diagram for sensor networks generation at design phase.

8.3 Sensor placement prototype

A prototype has been developed to demonstrate and validate the proposed specifications, the functionality that offers each involved component and the sequential interaction between these components. At a fist step, this prototype has been developed using MATLAB software package. However, the implementation into a set of CO-compliant modules working in a distributed way is straightforward, since the interface specification proposed in this work follows the CO standards and guidelines. The CO option enhances the potential of the adopted approach since this standard is obtaining a growing interest and acknowledgment at the academic as well as industrial environment. However, some simplifications are assumed for implementing the prototype as well as for the selected sensor network design problem:

1. Only system's accuracy is considered in the implemented prototype. A Data Reconciliation system has been wrapped to become the system's performance evaluation module M-II, offering process variables accuracy as the unique criterion to be selected for placing sensors within a given plant. It is important

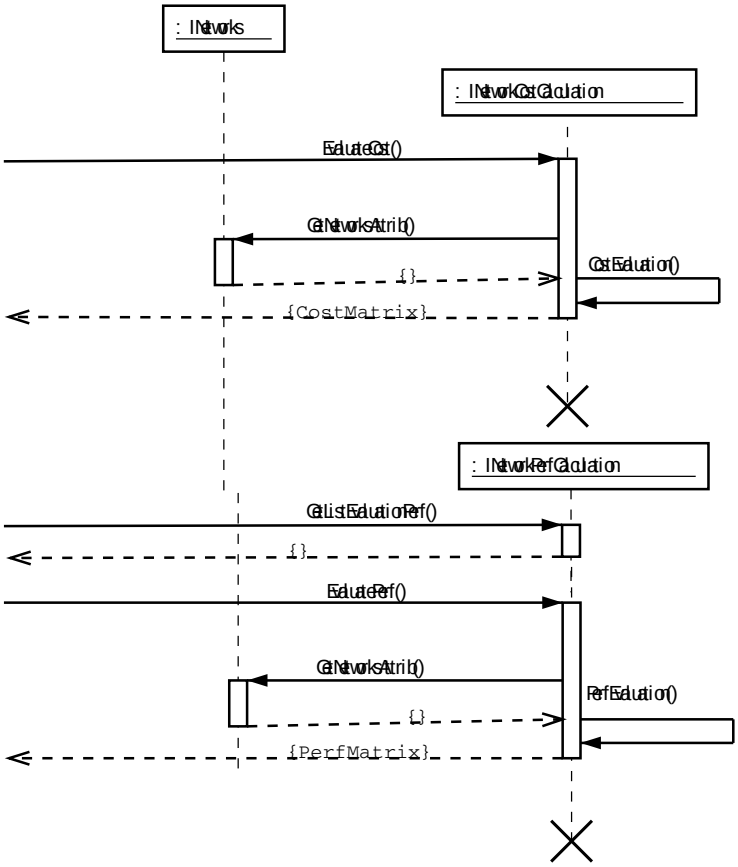


Figure 8.5: Proposed sequence diagram for cost and performance evaluations at the design phase.

to note that variance is calculated without any previous knowledge on process measurement which permit the use of the proposal for the design case. Additionally, since Data Reconciliation is adopted for evaluating the accuracy some degree of observability has to be satisfied. This requirement on observability leads to the minimum allowable number of sensors which in turn reduces the space of feasible combinations, and decreases the computational effort. An enumerative algorithm is adopted for generating the feasible combinations and it has been wrapped to be the sensor network generator module.

2. For sake of simplicity no more than one sensor per potential measurement point is assumed; that is $m_j^{max} = 1, \forall j$. Furthermore, it is assumed that the sensors

involved have the same characteristics and prices. Finally, the sensor network design problem is focused on placing sensors for monitoring the mass balance of a given system.

8.4 Case study for sensor placement design

8.4.1 Sensor network design for PROCEL pilot plant

A case study is proposed for validating the decision-making strategy as well as the prototype developed. As a test plant, PROcess CELL (PROCEL), a Pilot Plant, at the Universitat Politècnica de Catalunya (UPC) has been used[2]. PROCEL consists of three tanks with agitators, heaters, and heat exchangers. The tanks are connected in a highly flexible way so that different configurations are possible. In Figure 8.6 the continuous operation mode of this pilot plant in a specific configuration is shown. The design of the sensor network of the PROCEL plant is undertaken, focusing in the trade-off between maximizing accuracy of estimation through Data Reconciliation and the instrumentation cost.

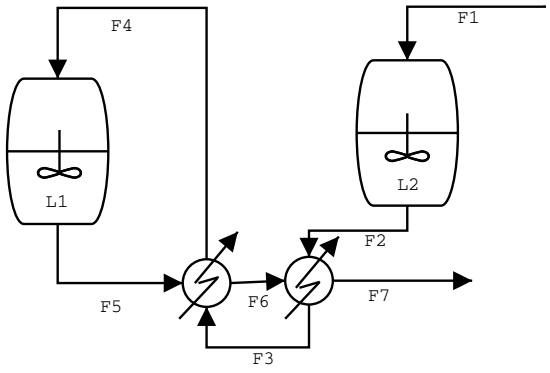


Figure 8.6: PROCEL Pilot Plant.

Commonly, the accuracy shows to be monotonically increasing with the investment. However, different networks might lead to the same performance, which implies a certain degree of degeneracy. This is reduced when the process under consideration is further complicated including recycling or when different sensor types are available. The former case involves that adding a sensor does not imply directly an improvement in the plant performance. Here, the dynamic case of PROCEL is considered. A dynamic mass balance has to be adopted if the sensor placement considers

[2]Further information on PROCEL can be supplied upon requests (luis.puigjaner@upc.es)

the dynamic behavior of the plant. Thus, the level in both tanks is given by:

$$F5 - F4 = \rho.s_1.\frac{dL1}{dt} \tag{8.29}$$

$$F2 - F1 = \rho.s_2.\frac{dL2}{dt} \tag{8.30}$$

If heat exchanger hold-ups are neglected, mass balance around them are represented as follows:

$$\begin{aligned} 0 &= F2 - F3 \\ 0 &= F6 - F7 \\ 0 &= F3 - F4 \\ 0 &= F5 - F6 \end{aligned} \tag{8.31}$$

Given this dynamic process model, assuming that the available sensors in the catalogue all have the same accuracy of 10% and cost of 1, module (I) generates sensors networks. This set is then evaluated by module (II) to produce to corresponding sensor networks accuracy, Kalman filter is used as the DDR (see Chapter 6). Module (III) is used to obtain the corresponding set of costs.
Figure 8.7 plots the performance given by the different sensor networks that can be arranged by acquiring 7 (see the right plot on Figure 8.7) and 8 (see the left plot on Figure 8.7) sensors.

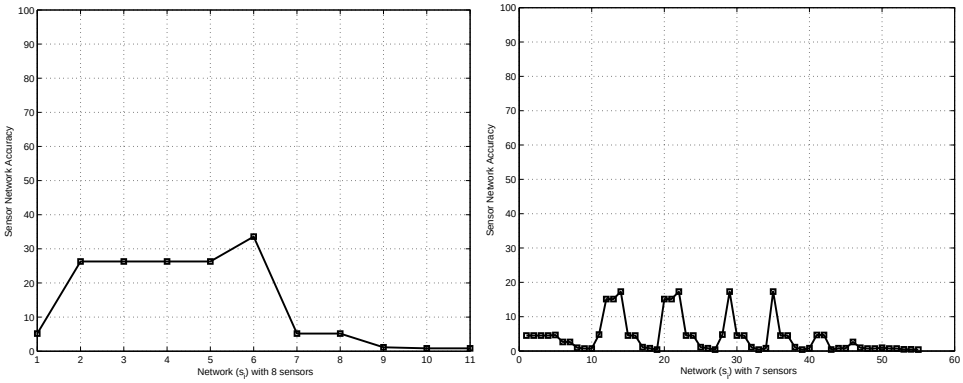


Figure 8.7: Performance vs. number of measuring devices (8 and 7) respectively.

The graph shows the dependence of the performance on the arrangement and clearly indicates that increasing the number of sensors does not necessarily lead to higher performance results but depend highly on the measuring point. The decision-making process is greatly aided by an information system allowing to properly manage the data obtained from the systematic analysis carried out. Thus, the sensor

arrangements evaluated may be sorted by performance to obtain the plots in Figure 8.8.

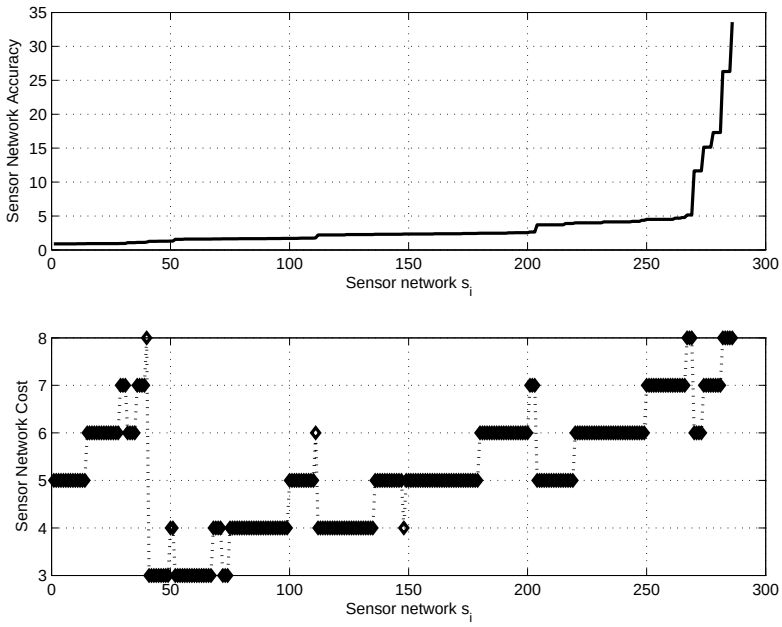


Figure 8.8: Increasing performance profile and related cost for design case.

The jagged performance profile confirms the idea that certain design decisions may have a significant impact on plant operation. In the case of equivalent measuring devices, the cost of the sensor network is given by the number of sensors. This is also plotted in Figure 8.8 for each network.

This second profile does not run parallel to the performance above, but allows learning that certain design decisions may imply a cost reduction simultaneously to performance rise. Once again, the information management allows acquiring deeper knowledge of the problem by gathering the previous data into Figure 8.9, which outlines the best performance that can be obtained at a fixed cost. Figure 8.9 provides useful information for deciding the trade-off cost performance though the comparative data for the basic design options. Moreover, the information system also allows further treatment of the data as a formal approach to the multi-objective problem such as Pareto optimal analysis (see Figure 8.9).

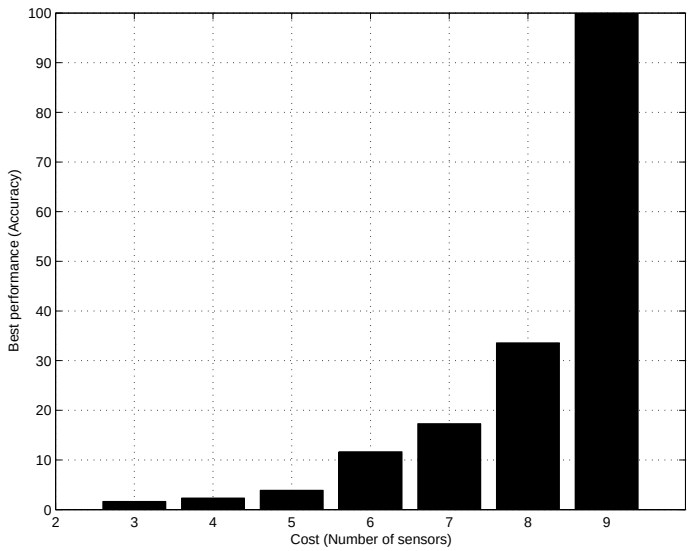


Figure 8.9: Performance vs. cost (the sensor network s_i producing the best performance is indicated for each investment cost).

Table 8.2: Sensor networks obtained for the design.

Network	F_1	F_2	F_3	F_4	F_5	F_6	F_7	L_1	L_2
s_1 (9 sensors)	1	1	1	1	1	1	1	1	1
s_7 (8 sensors)	1	1	1	1	1	-	1	1	1
s_{21} (7 sensors)	1	-	1	1	1	-	1	1	1
s_{76} (6 sensors)	1	-	1	-	1	-	1	1	1
s_{225} (5 sensors)	1	-	1	-	-	-	1	1	1
s_{436} (4 sensors)	1	-	-	-	-	-	1	1	1
s_{689} (3 sensors)	1	-	-	-	-	-	-	1	1

8.4.2 Reliability vs. accuracy

The incidence matrix representing the mass balance of PROCEL operating at steady-state is given by Eq. (8.32).

$$\mathbf{A} = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix} \tag{8.32}$$

The columns represent the seven flow-rates $Fi,\ i = 1 \cdots 7$ respectively while the rows correspond to mass balance around tanks and heat exchangers. It is important to mention that mass balances around heat exchangers considers separately the balance of the cooler flow and the heater flow.

For accuracy calculation, Lagrange Multiplier solution of the Steady-State Data Reconciliation problem is adopted and the variance of the estimated process variables in a matrix form is given by Eq. (8.33) when all variables are measured. The variance of the measured variables is given by $\mathbf{Q}$, where $\mathbf{Q}_{jj} = \sigma_j^2$.

$$\widehat{\mathbf{Q}} = \mathbf{Q} - \mathbf{Q}\mathbf{A}^T \left(\mathbf{A}\mathbf{Q}\mathbf{A}^T\right)^{-1} \mathbf{A}\mathbf{Q} \tag{8.33}$$

As expected, not all the variables are measured. Nevertheless, the actually unmeasured variables will be handled as measured variables with a large standard deviation.

For reliability evaluation all the redundant equations are formed by a disjoint set of measurements. The corresponding reliability is calculated by the probability analysis using the sum of disjoint products. This reliability calculation is treated in more details in Chapter 7 of this thesis. Figure 8.10 shows the accuracy and reliability given by the different sensor networks that can be proposed for PROCEL case, when modeled as a steady-state mass balance.

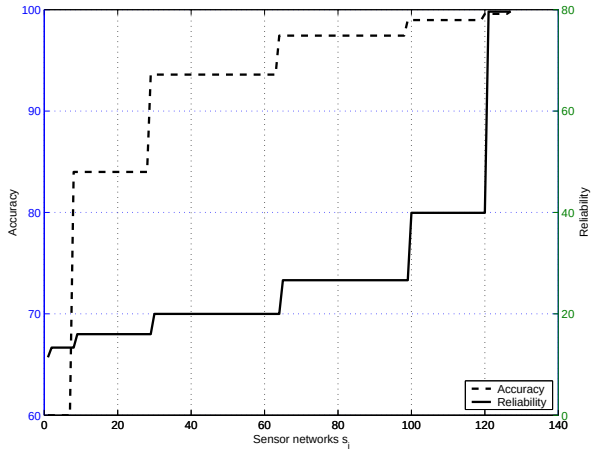


Figure 8.10: Accuracy and reliability trends.

Once again, the data are sorted by accuracy and hence the two graphs confirm for this case the parallel trend of both performance indexes (i.e. accuracy and reliability). Such an information allows reducing a double objective problem to a single one.

8.5 Sensor network upgrading strategy

Retrofitting is regarded as the procedure for providing the number, type and placement of measuring devices (considering already sensor characteristics) that should be added/relocated in the operating plant in order to increase plant performance, within budget considerations. The decisions involved in such a retrofitting can be made according to two main strategies:

1. purchase of new measuring devices.
2. reallocation of existing sensors while permitting new purchases.

Anyway, the starting point for all of these partial approaches is the information related to the current sensor network characteristics represented by $\Upsilon^0 = \left\{ n^0_{jk} \right\}$. Where the parameter set n^0_{jk} is defined as follows:

$$n^0_{jk} = \begin{cases} n & number\ of\ sensors\ of\ type\ k\ installed\ \ for\ measuring\ j \\ 0 & otherwise \end{cases} \tag{8.34}$$

Additionally, since the reallocation of sensors saves the acquisition cost but not some cost related to their installation/un-installation, the cost parameters need to be detached into:

$$c_k = c^c_k + c^i_k \tag{8.35}$$

where c^c_k and c^i_k are respectively the capital and installation costs. Additionally, for further system performance evaluation the characteristics of the installed measuring devices have to be included in the catalogue. In the proposed approach the set Υ^0 are considered as "new" sensors with a cost c^i_k (if the sensor position is changed during the sensor placement retrofitting procedure) and a certain performance (characteristics may be assigned to these old sensors by tuning the current accuracy and reliability values) that can be treated as a design problem.

8.5.1 Purchase strategy

Once this information is set, retrofitting by purchasing new items imposes that the already installed instrumentation is not subject to be changed and the goal is to determine the best set of new measuring devices (number, type and placement) that will satisfy the new performance requirements.

This procedure can be supported in a straightforward way by the design procedure presented in section 8.2 just by indicating to the system the set of sensors already installed (Υ^0).

Additionally, the already installed set of sensors might be prevented from reallocation by properly adjusting the multiplicity (m_j^{min}, m_j^{max}) and assigning an elevated cost to their un-installation.

8.5.2 Reallocation strategy

Retrofitting by means of reallocation is another particular case of retrofitting where the placement of some or all the installed sensors can be changed.

The goal is to determine the set of sensors to be reallocated as well as their new placement. The old sensors can be again considered as "new" sensors with a price corresponding uniquely to their installation cost c_k^i . This point is very important since it favors the already installed sensors to be re-allocated before acquiring new sensors with higher cost $c_k^c + c_k^i$.

Despite of this intrinsic priority, additional constraints could be added to force the re-installation of already existing sensors. Thus, purchase strategy can be seen as a special case of reallocating for which the installed sensor network Υ^0 does not admit any change.

8.5.3 Information Flows

The parameter sets n_{jk}^0 provides information related to the sensors characteristics (number, type and placement) and indirectly the cost. Once this information is provided solving the retrofitting problem will require the same components as those needed for the design case.

At this point, the interfaces designed proved to be robust and general enough to undertake a wide range of sensor placement problems. Thus, given n_{jk}^0, the current system performance can be obtained by calling the method *EvaluatePerf()* in the interface *INetworkPerfCalculation* presented in the interface diagram (see Figure 8.3). A reference or desired system performance may be set by assuming the design of a new sensor network for the same plant ($n_{jk} = 0, \forall j, \forall k$). Then, the investment needed to lead the current system performance to the desired one is evaluated invoking the method *EvaluateCost()* in the interface *INetworkCostCalculation*. To evaluate the investment, direct costs (as instrument purchase and installation) as well as benefits resulting from additional measuring devices have to be contemplated, therefore, M-III has to include an algorithm for such task. Thus, the extension of the sensor placement framework presented in previous section to deal with retrofitting is straightforward, although additional information should be provided into the system.

Additionally, cost evaluation may be extended to contemplate the purchase of new items and the reallocation of old ones through the related cost parameters. Such a change does not require any revision of the framework and interfaces proposed but

just plugging a new evaluation function (Eq. (8.36)) instead of Eq. (8.26). Eq. (8.36) considers capital cost and installation cost, but when $n^0_{jk} = 0, \forall j, \forall k$ becomes:

$$Total\,Cost = \sum_k c^c_k . \Theta \left(\sum_j n_{jk} - \sum_j n^0_{jk} \right) + \sum_k \sum_j c^i_k \left| n_{jk} - n^0_{jk} \right| \tag{8.36}$$

where:

$$\Theta(x) = \begin{cases} 1 & if \quad x \geq 0 \\ 0 & if \quad x < 0 \end{cases} \tag{8.37}$$

It is important to note that the retrofitting cost evaluation presented above can handle the situations where the already installed sensors are similar but different to the sensors provided in the catalogue (because of decreasing efficiency). This can be done in a parametric way by providing their performances (i.e. the set K is increased by including these sensor types).

Furthermore, once a general problem definition and solution framework have been set, the design case can be considered as a special retrofitting case that is parametrically described by setting $\Upsilon^0 = \left\{ n^0_{jk} \right\}$ to zero, thus meaning no sensors are already installed.

8.6 Case study for sensor placement retrofit

For the retrofit case assume that the PROCEL pilot plant considered (Figure 8.6) is operating at steady-state, thus the model given by the set of Eq. (8.31) can be used. Additionally, assumes that the plant already contains two flow-meters for measuring flows $F1$ and $F2$, and no more than one sensor per measuring point can be admitted. Finally, assumes flow-meters available in the catalogue present the characteristics given in Table 8.3.

The plant instrumentation combination (Υ) that fulfills the sensor placement goals is generated using the information of Table 8.3. The sensor network accuracy can be calculated using Eq. (8.33).

Figures 8.11 and 8.12 show cost and performance profiles given by the different sensor networks generated as instrumentation upgrading opportunities. Once again, information management allows plotting these alternatives in order of increasing performance, which in turn permits to observe that some retrofitting options may result in higher performance as well as lower cost.

Figure 8.12 summarizes the previous information in a graph relating the best performance that can be obtained up to a given cost. It is worth noting that in this case the cost is not only related to the number of sensors, but includes the relative costs

Table 8.3: Sensor characteristics for validating the retrofit case.

sensor type	installed $(k=1)$	new purchase $(k=2)$
σ_k^2	1.2	0.2
c_k^c	0.0	2.25
c_k^i	5.25	0.25

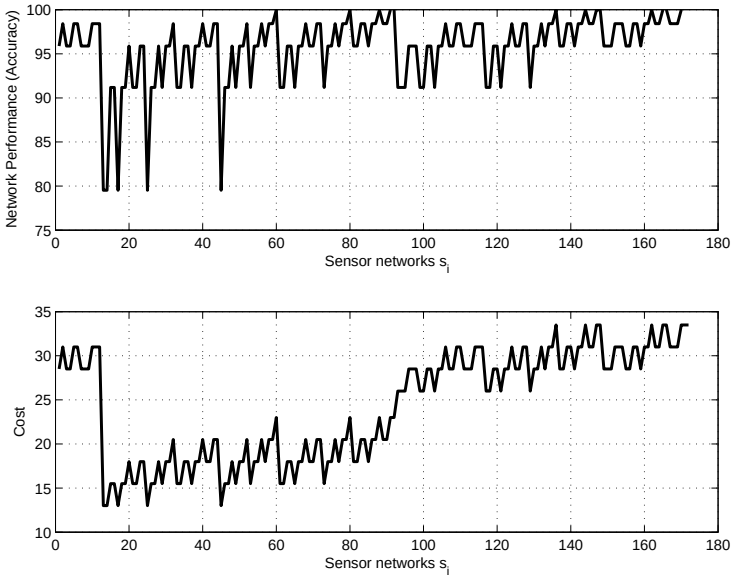


Figure 8.11: Increasing performance profile and related cost for retrofit case.

associated to rearranging plant instrumentation. This cost definition also reduces the problem degeneracy given by the simpler cost definition case study in section 8.4. Figure 8.12 provides useful information to to discard some level of investment.

Maintaining the already flow-meters in measuring point F_1 and F_2 and each time and adding at a distinct point a sensor of type 2 the plant performance is passed for an sensor network accuracy 62% corresponding to 0.58 to a 100%. Therefore, the sensor network presented in Table 8.4 provides a local maximum at cost 12.5 indicates that it makes no sense to consider arrangements of cost superior to 12.5, since no better performance may be obtained with any other sensor network.

A new minimum appear if any one of the flow-meters of type 1 placed in F_1 or F_2

Table 8.4: Most reliable sensor network for PROCEL at steady-state mode

	F_1	F_2	F_3	F_4	F_5	F_6	F_7
k	1	1	2	2	2	2	2

is placed elsewhere in F_3, F_4, F_5, F_6 or F_7. This situation can be enhanced each time an additional sensor is added until all the flows are sensed. Finally, a third minimum appear when no sensor of type one is placed in $F_1$and F_2 as shown in Figure 8.12.

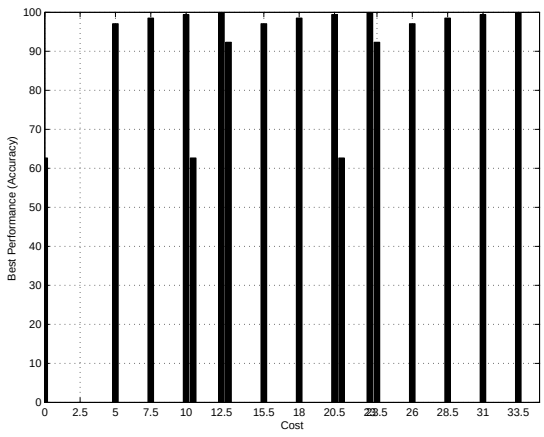


Figure 8.12: Best performance vs. cost for retrofit case.

This example shows the capability of the framework proposed to address the general sensor placement problem and to provide practical results for the decision-maker. Additionally, a main advantage of the framework is its openness and the possibility to incorporate any calculation module (plug-in) for solving the problem or part of it, thus allowing to increase the system performance with no code rewriting.

8.7 Conclusions

This chapter addresses the sensor placement problem through the analysis of the information flows involved in a comprehensive problem definition and a general purpose solution approach. The problem statement considers dynamic as well as steady-state systems and contemplates multi-objective sensor placement optimization, hardware redundancy and the catalogue of available measuring devices. Furthermore, in

this proposal in a unique formulation "all" the possible sensor placement problems (design, reallocation, new purchase) have been merged in a unique formulation.

As a result, beyond the specific chances given by the diverse optimization techniques that may be used, a decision-making framework for instrumentation design and retrofitting is presented. This framework is described in terms of the UML™ and thus, the interface, sequence and component diagrams showing the strategy proposed are also presented. The resulting specification has been implemented in a software prototype and using some illustrative examples. The use of the CO guidelines ensures an easy adaptation of the prototype to specific scenarios and accepts further integration to other chemical engineering tools such as Data Reconciliation system presented previously in Chapter 5.

The case studies presented also proved the usefulness and potential of a computer-aided decision-making tool allowing the comparative analysis of several instrumentation alternatives (networks) and the management of the information required for solving the cost-performance trade-off. The "what if" analysis provides a deeper insight into the sensor placement problem and provides a learning procedure that supports the decision-making through a more valuable knowledge of the problem.

Nomenclature

b_j	binary variable setting if the variable j is measured or not
c_j	cost of measuring j
c_k	total cost of sensor k
c_k^c	capital cost of sensor k
c_k^i	installation cost of sensor k
c^{max}	maximum cost allowable for instrumentation investment
J	set of measuring points
K	set of sensor type available in the catalogue
L	tank level
M^{max}	vector specifying maximum number of sensors per measuring point j
M^{min}	vector specifying minimum number of sensors per measuring point j
m_j^{max}	maximum number of measuring devices per measuring point j
m_j^{min}	minimum number of measuring devices per measuring point j
n_j	number of sensor in measuring point j
n_{jk}	number of sensors of type k measuring variable j
n_{jk}^0	number of already installed sensor type k in measuring point j
P	number of positional network
$\mathbf{Q}$	variance-covariance matrix of measured data
$\widehat{\mathbf{Q}}$	variance-covariance matrix of reconciled data
$\mathbf{R}$	matrix allowing sensor/measuring point assignment
r_{jk}	binary variable indicating if sensor type k can measure variable j or not
$r_j(t)$	reliability of process variable j at time instant t
$r_k(t)$	reliability of sensor type k at time instant t
$r_j(t)^*$	lower bound reliability for process variable j at time t
s_i	sensor network i
s	tank section
T	temperature

Greek letters

α_j	weight parameter for the accuracy
β_j	weight parameter for the instrumentation cost
Γ	cost matrix
δm_j	number of possible (decision) sensor per measurement point j
Θ	Heaviside function
Π	performance matrix
ρ	density

σ_j standard deviation of process variable j
σ_j^2 accuracy of process variable j
σ_k^2 accuracy of sensor type k
σ_j^{2*} upper bound accuracy for process variable j
Υ matrix of generated sensor networks
Υ^0 sensor network already installed in an operating plant

Subscripts

i sensor networks $i = 1...I$
j measurement point/measured variables $j = 1...J$
k sensor types $k = 1...K$

Chapter 9

Conclusions

9.1 Dynamic data Reconciliation

The first part of this thesis is focused on the investigation of dynamic Data Reconciliation. In **Chapter** 2 a Data Reconciliation technique for a dynamically evolving process is considered from a steady-state point of view. The technique uses previous information by means of the redundant measured variables averages to generate the current estimates. Thus, not only the temporal redundancy is taken into account but also the quality of estimation is improved by combining a filtering technique and SSDR sequentially. Moreover, a way for processing the average values to recover the "reconciled" variable is presented and validated. The proposed unbiased estimator is compared with the Kalman filter technique and shows high performance in terms of variance reduction and tracking of dramatic dynamic changes. Additionally, in **Chapter** 3 a dynamic Data Reconciliation based on wavelet trend analysis was presented. Firstly, wavelet is used to catch the deterministic trend of sensor data. Then, these trends are rendered consistent with the process model, optimizing the polynomial coefficients that fit these profiles. Therefore, estimations take advantage of both temporal and spatial redundancies. The approach proposed within this work can be applied on-line as well as off-line in an efficient way, presenting an accurate estimation. Another feature of this proposal is its capability to deal with measurements with different sampling frequencies in contrast to the current DDR techniques that can not address. The extension of this proposal to deal with nonlinear cases has also been presented demonstrating promising results. That is, the way the nonlinearity is undertaken in this work is relatively efficient, useful and involves simple and fast calculations.

One of the main problems encountered in operating plants is the presence of delays. In most cases of dynamic processes, where some sources of time-delays

between cause and effects appear in the system. The performance of DDR can be highly improved by means of its integration with time-delay estimation in both on-line and off-line cases as discussed in **Chapter** 4. One advantage of the presented TDE method is that a priori knowledge for the number and location of time-delays in the process is not necessary, which is an advantage of the application simplicity. On the other hand, although this approach shows good results for the case of process-related delay, it is specially suited for sensor-related delay since it deals with the measurements model. An extension of this approach to time-varying delays has been sketched.

The main aim of the work presented in **Chapter** 5 has been the design and implementation of a software architecture for distributed Data Reconciliation and parameter estimation applications. This contribution is presented towards the standardization of PEDR software packages. The task of maintaining and supporting new process models, databases and/or optimizers within the PEDR system can be justified in terms of cost and implementation effort using the proposed CO software specification and architecture.

A software prototype has been developed to validate and demonstrate the use and benefits in terms of reuse transparency and innovation of the proposed component and the specified interface. In this work the specification of the parameter estimation and Data Reconciliation interfaces are conceived as a generic approach, able to deal with dynamic, nonlinear processes with relatively low effort and without the need to make any essential change to the other components. The framework presented would be very appealing to industry and researchers that already have access to software that can perform the tasks of PE and DR. Since the most efficient use of such packages would be enhanced through the standardization of communication interfaces (which is the plug and play philosophy proposed by the CAPE-OPEN and Global CAPE-OPEN). The adaptability and flexibility of the proposal allows the inclusion of present and future algorithms, estimating unknown variance and covariance for process variables, as well as using new solvers for the resulting optimization problems.

9.2 Design of reliable and accurate sensor networks

The second part of this thesis addresses the design of accurate and reliable sensor networks. The major contribution of **Chapter** 6 is the establishment of a method that uses a Kalman filtering techniques for placing the minimum number of measuring devices in the "optimal" measurement points satisfying the network performance accuracy of dynamic systems. The solution strategy has been implemented in academic and industrial case studies showing promising results. The profile of the relative

increase of the system performance along the sensor network and the associated investment gives the designer all alternatives (Pareto space of optimal solution). The extension of the proposed approach to deal with nonlinear systems can be performed using an extended Kalman filter as the performance accuracy calculator.

For the sensor placement algorithm investigated in this thesis, a practical and efficient way for evaluating the reliability of process variables is presented in **Chapter** 7 based on combining both quantitative process knowledge and fault tree analysis. After mathematically formalizing the definition of the system reliability, a mathematical programming model for plant instrumentation was formulated and solved by means of genetic algorithms. The solution of this problem provides the number, type and optimal placement of measuring devices that has to be added and/or reallocated in an operating plant in order to upgrade plant performances to the desired precision and reliability. The formulation allows hardware redundancy and different sensor characteristics.

Finally, in this thesis the sensor placement problem is addressed through the analysis of the information flows involved, through a comprehensive problem definition and a general purpose solution approach. Problem statement considers steady-state systems and contemplates multi-criteria sensor placement, hardware redundancy and the catalogue of available measuring devices. Additionally, in this thesis "all" the possible sensor placement problems (Design, reallocation, new purchase) has been successfully merged in a unique formulation as shown in **Chapter** 8.

As a result, beyond the specific choices given by the diverse optimization techniques, a decision-making framework for instrumentation design and retrofitting is presented. This framework is described in terms of the UML^{TM} representation and thus, the interface, sequence and component diagrams showing the strategy proposed are also presented. The resulting interface specifications have been validated through the development of a software prototype that has been used for solving some illustrative examples. The interface specification has been developed following the CAPE-OPEN guidelines and thus it can be easily adapted to fit this standardization trend for a further integration of the sensor placement module to other chemical engineering tools such as Process Simulators and Data Reconciliation systems.

The case studies results also proved the usefulness and potential of a computer-aided decision-making tool allowing the comparative analysis of several instrumentation alternatives (networks) and the management of the information required for solving the cost-performance trade-off. The "what if" analysis provides deeper insight into the sensor placement problem and allows a learning procedure that supports the decision-making through more valuable knowledge of the problem.

Acronyms

AANN = Auto-Associative Neural Networks
ANN = Artificial Neural Networks
AVTI = Average Number of Type I
CAPE = Computer Aided Process Engineering
CWT = Continuous Wavelet Transform
CO = CAPE-OPEN
COLaN = CAPE-OPEN Laboratory Networks
COM = Component Object Model
CORBA = Common Object Request Broker Architecture
CSTR = Continuous Stirred Tank Reactor
DAEs = Differential-Algebraic Equations
DCS = Distributed Control System
DR = Data Reconciliation
DDR = Dynamic Data Reconciliation
DWT = Discrete Wavelet Transform
EKF = Extended Kalman Filter
EPA = Extended Polynomial Approach
ESO = Equation Set Object
EVM = Error-in-all-Variables Method
FDS = Fault Diagnosis System
GAMS = General Algebraic Modeling System
GAs = Genetic Algorithms
GED = Gross Error Detection
GLR = Generalized Likelihood Ratio

GCO = Global CAPE-OPEN
GT = Global Test
IDL = Interface Definition Language
JDK = Java Development Kit
LU = Lower and Upper triangular (matrix decomposition)
LVE = Liquid-Vapor Equilibrium
KF = Kalman Filter
MA = Moving Average
MA/SSDR = Moving Average/Steady-State Data Reconciliation
MH = Moving Horizon
MILP = Mixed Integer Linear Programming
MINLP = Mixed Integer Non Linear Programming
MLE = Maximum Likelihood Estimation
MSE = Mean-Squares Error
MSPC = Multivariate Statistical Process Control
NLP = Non Linear Programming
MP = Maximum Power
MT = Measurement Test
MySQL = My Structured Query Language (database query language)
NLP = Non Linear Programming
NT = Nodal Test
ODEs = Ordinary Differential Equations
OF = Objective Function
OMG = Object Management Group
OP = Overall Power
PCA = Principal Component Analysis
PE = Parameter Estimation
PEDR = Parameter Estimation and Data Reconciliation
PLS = Partial Least-Squares
PRD = Process-Related Delay
PROCEL = PROcess CELL
QP = Quadratic Programming
SMST = Sequential-Modular Simulator Tool

SQP = Successive Quadratic Programming
SRD = Sensor-Related Delay
SSDR = Steady-State Data Reconciliation
TDE = Time-Delay Estimation
TE = Tennessee Eastman
UBET = UnBiased Estimation Technique
UML = Unified Modeling Language
WLS = Weighted Least-Squares
WT = Wavelet Transform

Publications

Journal and Conference Proceeding Articles

Decision-Making Strategy and Tool for Sensor Networks Design and Retrofit

C. Benqlilou, M. Graells, L. Puigjaner
Universitat Politècnica de Catalunya, Chemical Engineering Department, ETSEIB, Diagonal 647, 08028 Barcelona, Spain.
Journal: Industrial and Engineering Chemical Research , Vol 43, 1711-1722, **2004**.
This work presents a decision-making framework for design and retrofit of sensor networks as well as a general strategy to correlate the cost and performance of the different sensor arrangements (number, placement) that may be systematically analyzed.

The analysis of the steps required for coping with the sensor placement problem, for design or retrofitting cases, implies the identification of the information flows involved. Such a conception allows a modular design of a tool for computer aided sensor network design/retrofit. An interface specification has been proposed for this tool and a prototype has been developed for validating purposes.

The decision-making strategy adopted is independent of the particular algorithms and procedures adopted. Hardware redundancy, steady-state/dynamic processes, design/retrofit and the catalogue of different available sensors are all taken into account, thus leading to a generic framework able to follow the future trends of sensor placement.

Seeking the synergy given by reusability and standardization, the sensor placement tool has been developed following the CAPE-OPEN (CO) guidelines, allowing the integration of other software modules such as Process Model and Data Reconciliation.

Design optimization of constructed wetlands for wastewater treatment

R. Pastor*, **C. Benqlilou***, D. Paz^{+}, G. Cardenas^{+}, A. Espuña*, L. Puigjaner*

*Universitat Politècnica de Catalunya, Chemical Engineering Department, ETSEIB, Diagonal 647, 08028 Barcelona, Spain.
+Estación Experimental Agroindustrial Obispo Colombes, Tucuman, Argentina.
Journal: Resources Conservation & Recycling, 37, 193, **2003.**
In this work the combination of mathematical programming based optimization strategy and hybrid neural network models is presented in the framework of wastewater minimization. In the optimization strategy, the objective function is composed by three terms: fresh water cost, wastewater treatment cost and the discharge taxes, while the constraints are the balance equations of all the production units. Once all production units that generate wastewater and the divers wastewater treatment systems are specified, the model automatically identified the best treatment option for each water stream: reuse or recycle with or without regeneration. the formulated optimization problem is solved using mathematical programming techniques and details about the optimum treatment for each stream are obtained. In general, for effluent treatment from municipal and food industrial wastewater, the most suitable process identified is the biological treatment. Among this treatment it has been chosen the constructed wetland. The proposed model for representing the dynamic of the wetlands is based on the combination of a first principles model and an artificial neural network. The hybrid model resulted from combining both modeling strategy has been used to optimize the design of the wetland. Finally, the methodology is applied to two case studies where the characterization of influent and effluent water flows is emphasized.

A Petri nets- based scheduling methodology for multipurpose batch plants

D. Riera+, M. Narciso+, **C. Benqlilou***
*Universitat Politècnica de Catalunya, Chemical Engineering Department, ETSEIB, Diagonal 647, 08028 Barcelona, Spain.
+Universitat Autònoma de Bracelona, Systems Engineering and Automatic Control Group, Bellaterra, 08193 Barcelona, Spain.
Journal: SIMULATION (Accepted, December **2003**).
This article presents an optimization methodology of batch production processes assembled by shared resources which rely on a mapping of state-events into time-events allowing in this way the straightforward use of a well consolidated scheduling policies developed for manufacturing systems. A technique to generate the timed Petri net representation from a continuous dynamic representation (Differential-Algebraic Equations (DAEs) systems) of the production system is presented together with the main characteristics of a Petri nets-based tool implemented for optimization purposes. This paper describes also how the implemented tool generates the coverability tree and how it can be pruned by a general purpose heuristic. An example of a distil-

lation process with two shared batch resources is used to illustrate the optimization methodology proposed.

Proposal To speed Up the Implementation of Abnormal Situation Management in the Chemical Process Industry

D. Ruiz*, **C. Benqlilou***, J. M. Nougués*, C. Ruiz+, L. Puigjaner*
*Universitat Politècnica de Catalunya, Chemical Engineering Department, ETSEIB, Diagonal 647, 08028 Barcelona, Spain.
+Advanced Control Group, SOTEICA SRL, Alvarez Thomas 796, 3 C, 1427 Buenos Aires, Argentina.
Journal: Industrial and Engineering Chemical Research, Vol 41, No. 4, **2002.**
The aim of this work is to present a proposal for implementation of a support framework for abnormal situation management in the chemical process industry. A main feature of the technology developed is that it takes advantage of existing software packages that are familiar to plant engineers (e.g. Plant Information System) and a commercial process simulator. On the basis of three sources of information (a historical database, a HAZOP analysis, and a first principles plant model), the support framework is developed and easily implemented into the real plant. It consists of a preprocessing module, which performs a variety of keys tasks using plant data such as data reconciliation, filtering, and de-noising. Some of the outputs of this preprocessing module are the inputs of the fault diagnosis system (FDS). This FDS is a combination of a pattern recognition approach based on neural networks and a fuzzy logic system (FLS) in a block oriented configuration. The case study to demonstrate the FDS implementation corresponds to a real petrochemical plant.

Sistema inteligente de soporte para la optimización en tiempo real de procesos de fabricación quimicos y petroquímicos

L. Puigjaner, A. Espuña, R. V. Tona, **C. Benqlilou,** S. E. Sequeira
Universitat Politècnica de Catalunya, Chemical Engineering Department, ETSEIB, Diagonal 647, 08028 Barcelona, Spain.
Book chapter: Ciencia & Tecnolgía, Vol. 1, pp. 261, Septiembre **2001,** Tibidabo Ediciones, Barcelona (ISBN: 84-8033-145-3).

El objetivo de este artículo es la descripción de las functionalidades requeridas por un sistema de Soporte de la Decisión avanzada para la monitorización, análisis y optimización de un proceso de fabricación, tomando como base los requerimientos de los sistemas de fabricación Químicos y Petroquímicos. Se trata de realizar una diagnosis en tiempo real del sistema que permite mantener la integridad operacional del proceso y mejore la calidad de productividad, todo ello al menor coste.

Publications

Para poder considerar adecuadamente los aspectos dinámicos del proceso a largo, medio e incluso a corto plazo es necesario introducir un ajuste progresivo del modelo en tiempo real que responda a la situación actual de la planta. Se debe conocer las incidencias que ocurren, analizarlas y tomar decisiones como reacción a dichas incidencias. Pero sobre todo se debe tener en cuenta el papel del operario para crear una estructura adecuada de soporte a la toma de decisiones.

Towards the Standardization of Thermodynamic and Physical Properties Packages

C. Benqlilou[1], S. Bel[1], M. González[1], M. Pons[2], R. Szczepanski[3], A. Espuña[1], L. Puigjaner[1]
[1]Universitat Politècnica de Catalunya, Chemical Engineering Department, Av. Diagonal 647, E-08028 Barcelona, Spain.
[2]ATOFINA Centre Technique de Lyon BP 32 69492 Pierre-Benite Cedex, France.
[3]InfoChem Computer Services Ltd, 13 Swan Court, 9 Tanner Street, London, SE1 3LE, U.K.
Book serie: ESCAPE-14, A. P. Póvoa and H. Matos (Editors) **2004** Elsevier Science B. V.

Adaptation of a Thermodynamic and Physical Properties (Thermo) Package to the increasing improvements of computer performance and the development of new, efficient, and robust thermodynamic methods, remains expensive in time, cost and implementation effort. An efficient solution to overcome these drawbacks is to standardize Thermo communications with client applications in a consistent, efficient and secure way by designing a well-defined interface that ensures interoperability and transparency. CAPE-OPEN (CO) standard provides such a facility. To enhance the existing standard, a revision of the CAPE-OPEN Thermo interface specification (version 1.1) has been proposed. A thermodynamic package has been wrapped to comply with this interface specification using COM technology. Considered as a help to developers, a Tester to verify standard compliance has been specified and developed. This Tester, a part of the CO-Tester suite, will be freely distributed and used to validate the consistency and compliance of the pieces of software implementing the new version of the CO Thermo interface specification. Version 1.1 of the CO interface specification for Thermo is operational.

Sensor-Placement for Dynamic Processes

C. Benqlilou[1] , J. M. Bagajewicz[2], A. Espuña[1], L. Puigjaner[1]
[1]Universitat Politècnica de Catalunya, Chemical Engineering Department, Av. Diagonal 647, E-08028 Barcelona, Spain.

[2]ATOFINA Centre Technique de Lyon BP 32 69492 Pierre-Benite Cedex, France.

[3]InfoChem Computer Services Ltd, 13 Swan Court, 9 Tanner Street, London, SE1 3LE, U.K.

Book serie: ESCAPE-13, A. Kraslawski and I. Turunen (Editors) **2003** Elsevier Science B. V.

This article presents a methodology to design instrumentation networks for dynamic systems where Kalman filtering is the chosen monitoring technique. Performance goals for Kalman filtering are discussed and minimum cost networks are obtained.

An Open Software Architecture for Steady-State Data Reconciliation and Parameter Estimation

C. Benqlilou , M. Graells, A. Espuña, L. Puigjaner

Universitat Politècnica de Catalunya, Chemical Engineering Department, Av. Diagonal 647, E-08028 Barcelona, Spain

Book serie: ESCAPE-12, J. Grievink and J. van Schijndel (Editors) **2002** Elsevier Science B. V.

In this paper a flexible and open architecture design for Parameter Estimation and Data Reconciliation (PEDR) software application is proposed by de-coupling it according to the functionalities involved.

In the proposed approach the different components that are involved in this application and their interactions are specified and tested. The proposed architecture aims at an improved efficiency and upgrading of the PEDR application by allowing the exchangeability and connectivity of the present components in an easy and consistent way.

Modelado y Simulación de un Proceso de Destilación

C. Benqlilou

Universitat Autònoma de Bracelona, Enginyeria de Sistemes i Automàtica, Bellaterra, 08193 Barcelona, España.

Memoria presentada como trabajo de investigación de Tercer Ciclo de Informática, **1998.**

Participation in Congresses and Workshops

American Institute of Chemical Engineers (AIChE)

1. **C. Benqlilou**, C. M. Graells, L. Puigjaner, A Sensor-Placement Framework: Design and Retrofit **2003.** San Francisco, California.

2. **C. Benqlilou**, C. A. Espuña, L. Puigjaner, Sensor Placement Optimization for Robust Control **2001.** Reno, Nevada.

3. D. Ruiz D., J.M., Nougués, **C. Benqlilou**, C. Ruiz, L. Puigjaner, A proposal to speed-up the implementation of an Abnormal Situation Management framework into real chemical plants, **2000.** Los Angeles, California.

Canadian Chemical Engineering Conference (CSCHE)

1. **C. Benqlilou**, M. Graells, L. Puigjaner. A Sensor-Placement Framework: Design and Retrofit. 53rd CSCHE, **2003**. Hamilton, Ontario, Canada.

2. **C. Benqlilou**, S. Bel, A. Espuña, M. Pons, and L. Puigjaner, Towards a Standardization of Optimization Software. 52nd CSCHE, **2002**. Vancouver, Canada.

3. **C. Benqlilou**, A. Espuña, and L. Puigjaner, Dynamic Data Reconciliation Based Software Simulator Models (***KEYNOTE LECTURE***). 52nd CSCHE, **2002**. Vancouver, Canada.

4. **C. Benqlilou**, A. Espuña, and L. Puigjaner, Improved data reconciliation by redundant model updating. A distributed architecture. 51st CSCHE, **2001**. Halifax, Nova Scotia, Canada.

Mediterranean Congress of Chemical Engineering (MCCE) Session: Process systems engineering.

1. **C. Benqlilou**, E. Musulin, J. M. Nougués, M. J. Bagajewicz, A. Espuña, L. Puigjaner, Synergy between Dynamic Data Reconciliation and Sensor-Placement. 9th MCCE, **2003**, Barcelona, Spain.

2. **C. Benqlilou**, M. J. Bagajewicz, A. Espuña, L. Puigjaner, A Comparative Study of Linear Dynamic Data Reconciliation Techniques. 9th MCCE, **2003**, Barcelona, Spain.

3. **C. Benqlilou**, S. Bel, M. Pons, A. Espuña, L. Puigjaner, Open systems architectures for process engineering software: MINLP Solver Specification, Prototype and Testing. 9th MCCE, **2003**, Barcelona, Spain.

4. J. Cantón, D. Ruiz, **C. Benqlilou**, J. M. Nougués, L. Puigjaner, An integrated architecture for information management in batch chemical processes, 8th MCCE, **1999**, Barcelona, Spain.

5. **C. Benqlilou**, D. Ruiz, J. M. Nougués, L. Puigjaner, A Hybrid Neural Network-First Principles Approach for Process Modeling. 8th MCCE, **1999**, Barcelona, Spain.

Process Integration, Modeling and Optimization for Energy Saving and Pollution Reduction (PRES) Conference

1. **C. Benqlilou**, C., R.V. Tona, A. Espuña, L. Puigjaner, On-Line Application of Dynamic Data Reconciliation. 4th PRES, **2001**. Florence, Italy.

2. R. Pastor, **C. Benqlilou**, D. Paz, G. Cardenas, A. Espuña, L. Puigjaner, Design Optimization of Constructed Wetlands for wastewater treatment, 4th PRES, **2001**. Florence, Italy.

3. **C. Benqlilou**, D. Ruiz, A. Espuña, L. Puigjaner, Enhancement of Prediction and Extrapolation Properties by using Hybrid Neural Network Model, 3rd PRES, **2000**. Prague, Republic Cseck.

Other conferences

1. J. Cantón, D. Ruiz, **C. Benqlilou**, J. M. Nougués, L. Puigjaner, Integrated information system for monitoring, scheduling and control applied to batch chemical processes, *Proceedings 7th IEEE International Conference on Emerging Technologies and Factory Automation* (Editor: J. M. Fuertes), Vol. 1, pp 211-217, **1999**. Bracelona, Spain.

2. **C. Benqlilou**, M. Graells, L. Puigjaner. Decision-Making Strategy and Tool for Sensor-placement, *4th European Congress of Chemical Engineering* P-9.2-041, **2003**. Granada, Spain.

Bibliography

Bibliography

Z. H. Abu-el zeet, P. D. Roberts, and V. M. Becerra. Enhancing Model Predictive Control Using Dynamic Data Reconciliation. *AIChE J.*, 48(2):324–333, 2002.

P. S. Addison. *The Illustrated Wavelet Transform Handbook: Applications in Science, Engineering, Medicine and Finance*. Institute of Physics Publishing, 2002.

J. S. Albuquerque and L. T. Biegler. Data Reconciliation and Gross-Error Detection for Dynamic Systems. *AIChE J.*, 42(10):2841, 1996.

Y. Ali and S. Narasimahan. Sensor Network Design For Maximizing Reliability of Linear Processes. *AIChE J.*, 39(5):820–826, 1993.

Y. Ali and S. Narasimhan. Redundant Sensor Network Design for Linear Processes. *AIChE J.*, 41(10):2237–2249, 1995.

G. A. Almasy and R. S. H. Mah. Estimation of measurement error variances from process data. *Ind. Eng. Chem. Proc. Des. Dev.*, (23):779, 1984.

N. L. Arora and L. T. Biegler. Redescending Estimation for Data Reconciliation and Parameter Estimation. In *AIChE Annual Meeting*, Los Angeles, CA, 2000.

N. L. Arora, L. T. Biegler, and G. Heyen. *Software Architecture and Tools for Computer Aided Process Engineering*, chapter Data Reconciliation Framework, pages 193–212. Elsevier Science, 2002.

M. J. Bagajewicz. Design and Retrofit of Sensor Networks in Process Plants. *AIChE J.*, 43(9):2300–2306, 1997.

M. J. Bagajewicz and E. Cabrera. Pareto Optimal Solutions Visualisation Techniques for Multiobjective Design and Upgrade of Instrumentation Networks. *Ind. Eng. Chem. Res.*, (42):5195–5203, 2000.

M. J. Bagajewicz and Q. Jiang. Integral approach to plant Linear Dynamic Reconciliation. *AIChE J.*, 43(10):2546–2558, 1997.

M. J. Bagajewicz and M. C. Sánchez. Design and Upgrade of Nonredundant and Redundant Linear Sensor Networks. *AIChE J.*, 45(9):1927–1938, 1999.

M. J. Bagajewicz and M. C. Sánchez. Cost-optimal design of reliable sesnor networks. *Comp. Chem. Engng.*, 23:1757–1762, 2000a.

M. J. Bagajewicz and M. C. Sánchez. Reallocation and Upgrade of Instrumentation in Process Plants. *Comp. Chem. Engng.*, (24):1945, 2000b.

B. R. Bakshi. Multiscale PCA with Application to Multivariate Statistical Process Monitoring. *AIChE J.*, 44(7):1596–1610, 1998.

J. P. Belaud, B. Braunschweig, and M. White. *Software Architecture and Tools for Computer Aided Process Engineering*, chapter The CAPE-OPEN Standard: Motivations, Development Process, Technical Architecture and Examples, pages 303–332. Elsevier Science, 2002.

J. P. Belaud and M. Pons. Open software architecture for process simulation: the current status of the CAPE-OPEN standard. In Elsevier Science J. Grievink and J. van Schijndel, editor, *12th European Symposium on Computer Aided Process Engineering*, pages 847–852, 2002.

C. Benqlilou, M. J. Bagajewicz, A. Espuña, and L. Puigjaner. A Comparative Study of Linear Dynamic Data Reconciliation Techniques. In *9th Mediterranean Congress of Chemical Engineering*, pages 8–31, Barcelone, Spain, November 2002a.

C. Benqlilou, M. J. Bagajewicz, A. Espuña, and L. Puigjaner. Sensor Placement for Dynamic Systems. In A. Kraslawski and I. Turunen Eds, editor, *13th European symposium on computer aided process engineering*, pages 371–376, Lappeenranta, Finland, 2003.

C. Benqlilou, S. Bel, M. Pons, A. Espuña, and L. Puigjaner. Open Systems Architectures for Process Engineering Software: MINLP Solver Specification, Prototype and Testing. In *9th Mediterranean Congress of Chemical Engineering*, pages 8–8, Barcelone, Spain, November 2002b.

C. Benqlilou, A. Espuña, S. Bel, and L. Puigjaner. Towards a Standarization of Optimization Software. In *52nd Can. Chem. Eng. Conf.*, Vancouver, British Columbia, Canada, October 2002c.

C. Benqlilou, A. Espuña, and L. Puigjaner. Improved Data Reconciliation by Redundant Model Updating: Distributed Architecture. In *51th Can. Chem. Eng. Conf.*, Halifax, Nova Scotia, Canada, October 2001a.

C. Benqlilou, A. Espuña, and L. Puigjaner. Sensor-Placement Optimization for Robust Control. In *AIChE Annual Meeting*, Reno, USA, October 2001b.

C. Benqlilou, A. Espuña, and L. Puigjaner. Dynamic Data Reconciliation Based Software Simulator Models. In *52nd Can. Chem. Eng. Conf.*, Vancouver, British Columbia, Canada, October 2002d.

C. Benqlilou, M Graells, A. Espuña, and L. Puigjaner. An Open Software Architecture for Steady-State Data Reconciliation and Parameter Estimation. In Elsevier Science J. Grievink and J. van Schijndel, editor, *12th European Symposium on Computer Aided Process Engineering*, pages 847–852, The Hague, The Netherlands, 2002e.

C. Benqlilou, M. Graells, and L. Puigjaner. Decision-making Strategy and Tool for sensor networks Design and Retrofit. *Ind. Eng. Chem. Res.*, 43:1711–1722, 2004.

C. Benqlilou, R. V. Tona, A. Espuña, and L. Puigjaner. On-Line Application of Dynamic Data Reconciliation. In *4th Conference on Process Integration, Modelling, and Optimisation for Energy Saving and Pollution Reduction*, pages 403–406, Florence, Italy, May 2001c. Jiri Klemes.

T. Binder, L. Blank, W. Dahmen, and W. Marquardt. On the regularization of dynamic data reconciliation problems. *Journal of Process Control*, (12):557–567, 2002.

B. Braunschweig and R. Gani. *Software Architecture and Tools for Computer Aided Process Engineering*. Braunschweig, B R. Gani, Elsevier Science, 2002.

CAPE-OPEN Standard 1.1. CO-LaN (the CAPE-OPEN Laboratories Network). Delivering the power of component software and open standard interfaces in computer-aided process engineering. [on-line] Site web, URL http://www.colan.org., 1 May 2002.

V. Charpentier, L. J. Chang, G. M. Schwenzer, and M. C. Bardin. An On-line Data Reconciliation System for Crude and Vacuum Units. In *NPRA Computer Conf*, 1991.

J. Chen and J. A. Romagnoli. A Strategy for Simultaneously Dynamic Data Reconciliation and Outlier Detection. *Comp. Chem. Engng.*, (22):559–562., 1998.

M. Chiari, G. Bussani, M. G. Grottoli, and S. Pierucci. On-Line Data Reconciliation and Optimization: Refinery Applications. In S. Skogestad, editor, *7th European Symposium on Computer Aided Process Engineering*, pages 1185–1190, 1997.

D. J. Chmielewski, T. Palmer, and V. Manousiouthakis. On the Theory of Optimal Sensor Placement. *AIChE J.*, 48(5):1001–1012, 2002.

C. M. Crowe. Reconciliation of process flow rates by matrix projection. ii: nonlinear case. *AIChE J.*, 32:616–623, 1986.

C. M. Crowe, Y. A. G. Campos, and A. Hrymak. Reconciliation of process flow rates by matrix projection. Part I: linear Case. *AIChE J.*, 29(6):881–888, 1983.

M. Darouach and M. Zasadzinski. Data Reconciliation in Generalized Linear Dynamic Systems. *AIChE J.*, 37:193, 1991.

D. L. Donoho and I. M. Johnstone. Adapting to Unknown Smoothness via Wavelet Shrinkage. *J. Am. Stat. Assoc.*, 90(432):1200:1224, 1995.

J. Downs and E. Vogel. A plant-wide industrial process control problem. *Comp. Chem. Engng.*, (17):245–255, 1993.

F. Doymaz, A. Bakhtazad, J. A. Romagnoli, and A. Palazoglu. Wavelet-based robust filtering of process data. *Comp. Chem. Engng.*, (25):1549–1559, 2001.

F. Flehmig, R. V. Watzdorf, and W. Marquardt. Identification of Trends in Process Measurements using the Wavelet Transformation. *Comp. Chem. Engng.*, 22:491–496, 1998.

L. M. Fraleigh, M. Guay, and J. F. Forbes. Sensor selection for model-based real-time optimization: relating design of eperiments and design cost. *Journal of Process Control*, 13:667–678, 2003.

M. Fritz and S. Engell. An Open Software Architecture for Batch Process Simulation. In S. Skogestad, editor, *7th European Symposium on Computer Aided Process Engineering*, pages 769–773, 1997.

A. Gelb. *Applied Optimal Estimation*. Cambridge, Mass.: MIT Press, 1974.

D. Goldberg. *Genetic algorithms in search, optimization machine learning, machine learning*. Addison Wesley, 1989.

G. Heyen, M. Dumont, and B. Kalitventzeff. Computer-Aided Design of Redundant Sensor Networks. In Elsevier Science. J. Grievink and J. van Schijndel, editor, *12th European Symposium On Computer Aided Process Engineering*, pages 685–690, 2002.

D. Hodouin and M. D. Everell. A Hierarchical Procedure for Adjustment and Material Balancing of Mineral Processes Data. *Int. J. Miner. Proc.*, (7):91–116, 1980.

J. Holland. *Adaptation in natural and artificial systems*. University of Michigan Press, 1975.

P. J. Huber. *Robust Statistics*. J. Wiley & Sons, New York., 1981.

K. A. Islam, G. H. Weiss, and J. A. Romagnoli. Nonlinear Data Reconciliation For An Industrial Pyrolysis Reactor. In H. J. Bart F. Moser, H. Schnitzer, editor, *4th European Symposium on Computer Aided Process Engineering*, pages 218–221, 1994.

S. S. Jang, B. Josepth, and H. Muhai. Comparison of two approaches to on-line parameter and state estimation problem of non-linear systems. *Ind. Eng. Che,. Process Des. Dev.*, (25):809–814, 1986.

K. Jaroslav and T. Perris. Quality Information for Production Management, A Performance Analysis Tool for Interactive use by plant Personal. In Elsevier Science R. Gani, editor, *1st European Symposium on Computer Aided Process Engineering*, pages 507–511, 1992.

A. Kakhu, B. Keeping, Y Lu, and C. Pantelides. An Open Software Architecture for Process Modeling and Model-Based Applications. In Ed Pekny J. F and G. E. Blau, editor, *3rd Foundations of Computer-Aided Process Opearation conference*, pages 518–524, 1998.

M. Kano, K. Nagao, S. Hasebe, I. Hashimoto, H. Ohno, R. Strauss, and B. Bakshi. Comparison of multivariate statistical process monitoring methods with applications to the Eastman challenge problem. *Comp. Chem. Engng.*, (26):161–174, 2002.

T. W. Karjala and D. M. Himmelblau. Dynamic Rectification of Data via Recurrent Neural Network and the Extended Kalman Filter. *AIChE J.*, (42):2225, 1996.

C. Karr. Fuzzy control of pH using genetic algorithms. *IEEE Trans. Fuzzy Syst.*, (1): 146–153, 1993.

J. D. Kelly. A regularization approach to the reconciliation of constrained data sets. *Comp. Chem. Engng.*, 22(12):1771–1778, 1998.

I. M. Kim, M. J. Liebman, and T. F. Edgar. A sequential Error in Variables Estimation Method for Nonlinear Dynamic Systems. *Comp. Chem. Engng.*, (15):663–670, 1991.

I. W. Kim, M. J. Liebman, and T. F. Edgar. Robust Error in Variables Estimation using Nonlinear Programming Techniques. *AIChE J.*, (36):985–993, 1990.

J. C. Knepper and J. W. Gorman. Statistical Analysis of Constrained Data Sets. *AIChE J.*, 26(2):260–264, 1980.

M. Kong, B. Chen, and B. Li. An Integral approach to dynamic data rectification. *Comp. Chem. Engng.*, 24(2/7):749–753, 2000.

A. Kretsovalis and R. S. H. Mah. Effect of Redundancy on Estimation Accuracy in Process Data Reconciliation. *Chem. Eng. Sci.*, 42:2115, 1987.

A. Kretsovalis and R. S. H. Mah. Observability and Redundancy Classification in Generalized Process Network. *Comp. Chem. Engng.*, 12(7):671–703, 1988.

D. R. Kuehn and H. Davidson. Computer control II: mathematics of control. *Chem. Engng. Prog.*, 57(6):1653–1672, 1961.

M. J. Liebman, T. F. Edgar, and L. S. Lasdon. Efficient data reconciliation and estimation for dynamic processes using nonlinear programming techniques. *Comp. Chem. Engng.*, 16(10/11):963–986, 1992.

M. J. Liebman and T. F. Edgard. Data reconciliation for nonlinear processes. In *AIChE Annual Meeting, Washington*, 1988.

M. Luong, D. Maquin, C. T. Huynh, and J. Ragot. Observability, Redundancy, Reliability and Integrated Design of Measurement Systems. In *IFAC Symp. on Inteligent Components and Instrument Control Applications, Budapest*, 1994.

F. Madron. A new approach to the identification of gross errors in chemical engineering measurements. *Chem. Eng. Sci*, (40):1855–1860, 1985.

F. Madron and V. Veverka. Optimal Selection of Measuring Points in Complex Plants by Linear Models. *AIChE J.*, 38(2):227–236, 1992.

R. S. H. Mah. *Chemical Process Structures and Information Flows*. Boston: Butterworths, 1990.

R. S. H. Mah, G. M. Stanley, and D. W. Downing. Reconciliation and Rectification of Process Flow and Inventory Data. *Ind. & Eng. Chem. Proc. Des. Dev.*, (15): 175–183, 1976.

R. S. H. Mah and A. C. Tamhane. Detection of Gross Errors in Process Data. *AIChE J.*, (28):828, 1982.

P. S. Maybeck. *Stochastic models, estimation, and control*, volume 141. series. Mathematics in Science and Engineering, ACADEMIC PRESS, INC New York, 1979.

T. McAvoy and N. Ye. Base control problem for the Tennessee Eastman problem. *Comp. Chem. Engng.*, (18):383–413., 1994.

M. Meyer, J. M. L. Le Lann, B. Koehert, and M. Enjalbert. Optimal selection of sensor location on a complex plant using graph-oriented approach. In H. J. Bart F. Moser, H. Schnitzer, editor, *4th European Symposium on Computer Aided Process Engineering*, pages 535–540, 1994.

K. Mingfang, C. Bingzhen, H. Xiaorong, and H. Shanying. Approach to dynamic data reconciliation based on extended warm-start technique. In Ed Pekny J. F and G. E. Blau, editor, *3rd Foundations of Computer-Aided Process Operations Conference*, pages 439–442, January 2003.

K. Morad, W. Y. Svrcek, and I. McKay. A robust direct approach for calculating measurement error covariance matrix. *Comp. Chem. Engng.*, (23):889–897, 1999.

P. Muller. *Modélisation Object avec UML*. Editions Eyrolles París, 1997.

K. Muske and T. F. Edgar. *Nonlinear State Estimation*, chapter 6. M. A. Henson and D. E. Seborg, Prentice Hall, Englewood Cliffs, NJ, 1998.

E. Musulin, M. J. Bagajewicz, J. M. Nougués, and L. Puigjaner. Instrumentation design and upgrade for principal components analysis monitoring. *Accepted in Ind. Chem. Eng. Res.*, 2003.

S. Narasimhan and C. Jordache. *Data Reconciliation & Gross Error Detection: An Intelligent Use of Process Data*. Gulf Publishing Company, Houston, Texas, 2000.

S. Narasimhan and R. S. H. Mah. Generalized Likelihood Ratio Method for Gross Error Identification. *AIChE J.*, 33(9):1514–1521, 1987.

S. Narasimhan and R. Rengaswamy. Sensor Network Design for Integrated Process Management: An Overview and Recent Results. In *53th Can. Chem. Eng. Conf., Hamilton, ON*, October 2003.

M. N. Nounou and B. R. Bakshi. On-Line Multiscale Filtering of Random and Gross Errors without Process Models. *AIChE J.*, 45(5):1041–1058, 1999.

OMG. OMG (the Object Management Group). [on-line] Site web, URL http://www.omg.org, 1 May 2003.

R. Orfali and D. Harkly. *Client/Server programming with java and CORBA*. J.Wiley & Sons, Inc. Captions, 1998.

R. Raghuraj, M. Bhushan, and R. Rengaswamy. Locating Sensors in Complex Chemical Plants based on Faults Diagnosis Observability Criteria. *AIChE J.*, 45(2):310–322, 1999.

J. Ragot, D. Maquin, and G. Bloch. Sensor Positioning for Processes Described by Bilinear Equations. *Diagnostic et surete de fonctionment*, (2):115–132, 1992.

P. M. Reilly and R. E. Carpani. Application of Statistical Theory to Adjustment of Material Balances. In *13th Can. Chem. Eng. Conf., Montreal, Quebec*, 1963.

D. L. Ripps. Adjustment of Experimental Data. *Chem. Eng. Prog. Symp. Ser.*, (5561): 8–13, 1965.

D. K. Rollins and J. F. Davis. Unbiased Estimation of Gross Errors in Process Measurements. *AIChE J.*, (38):563, 1992.

D. K. Rollins and D. Devanathan. Unbiased Estimation in Dynamic Data Reconciliation. *AIChE J.*, 39(8):1330–1334, 1993.

J. A. Romagnoli and M. C. Sánchez. *Data Processing and Reconciliation for Chemical Process Operations*, volume Volume 2 of PROCESS SYSTEM ENGINEERING. Academic Press, 2000.

J. Rosenberg, R. S. H. Mah, and C. Jordache. Evaluation of Schemes for Detecting and Identification of Gross Errors in Process Data. *Ind. & Eng. Chem. Proc. Des. Dev.*, (26):555–564, 1987.

A. C. H. Rowe and P. Abbott. Daubechies Wavelets and Mathematics. *Computer in Physics*, 9(6):635–648, 1995.

D. Ruiz, C. Benqlilou, J. M. Nougués, C. Ruiz, and L. Puigjaner. Proposal To Speed Up the Implementation of an Abnormal Situation Management in the Chemical Process Industry. *Ind. Eng. Chem. Res.*, 41(4):817–824, 2001.

M. C. Sánchez, A. Bandoni, and J. A. Romagnoli. PLADAT: A Package for Process variable Classification and Plant Data Reconciliation. In Elsevier Science R. Gani, editor, *1st European Symposium on Computer Aided Process Engineering*, pages 499–506., 1992.

M. C. Sánchez and J. A. Romagnoli. Use of orthogonal transformation in data classification-reconciliation. *Comp. Chem. Engng.*, (20):483–493, 1996.

M. Sen, S. Narasimhan, and K. Deb. Sensor Network Design of Linear Processes Using Genetic Algorithms. *Comp. Chem. Engng.*, (22):385–390, 1998.

R. W. Serth and W. A. Heenan. Gross Error Detecting and Data Reconciliation in Stream-Metering Systems. *AIChE J.*, (30):743–747, 1986.

Sheffield University. Genetic Algorithms Toolbox. [on-line] web site, URL http://www.shef.ac.uk/uni/projects/gaipp/ga-toolbox/, 1 August 2003.

D. E. Simpson, V. R. Voller, and M. G. Everett. An efficient algorithm for mineral processing data adjustment. *Int. J. Miner. Proc.*, 31:73–96, 1991.

G. M. Stanly and R. S. H. Mah. Observability and Redundancy in Process Data Estimation. *Chem. Eng. Sci.*, 36:259–272, 1981.

I. B. Tjoa and L. T. Biegler. Simultaneous strategies for data reconciliation and gross error detection of nonlinear systems. *Comp. Chem. Engng.*, (15):679, 1991.

R. V. Tona, A. Espuña, and L. Puigjaner. Standarizing approaches for wavelet filtering. Technical report, University Politechnic of Catalunya, Chemical Engineering Department, 2003.

H. Tong and C. M. Crowe. Detection of Gross Errors in Data Reconciliation by Principal Component Analysis. *AIChE J.*, 41(7):1712–1722, 1995.

V. Vaclaveck. Studies on system engineering III. Optimal choice of the balance measurments in complicated chemical systems. *Chem. Eng. Sci.*, 24:974, 1969.

V. Vaclaveck and M. Loucka. Selection of Measurements Necessary to Achieve Multicomponent Mass Balances in Chemical Plants. *Chem. Eng. Sci.*, (31):1199–1205, 1976.

A. Wachs and D. R. Lewin. Improved PCA Methods for Process Disturbance and Failure Identification. *AIChE J.*, 45:1688–1700, 1999.

S. Yoon and J. F. MacGregor. Statistical and causal model based approaches to fault detection and isolation. *AIChE J.*, 46(9):1813–1824, 2000.