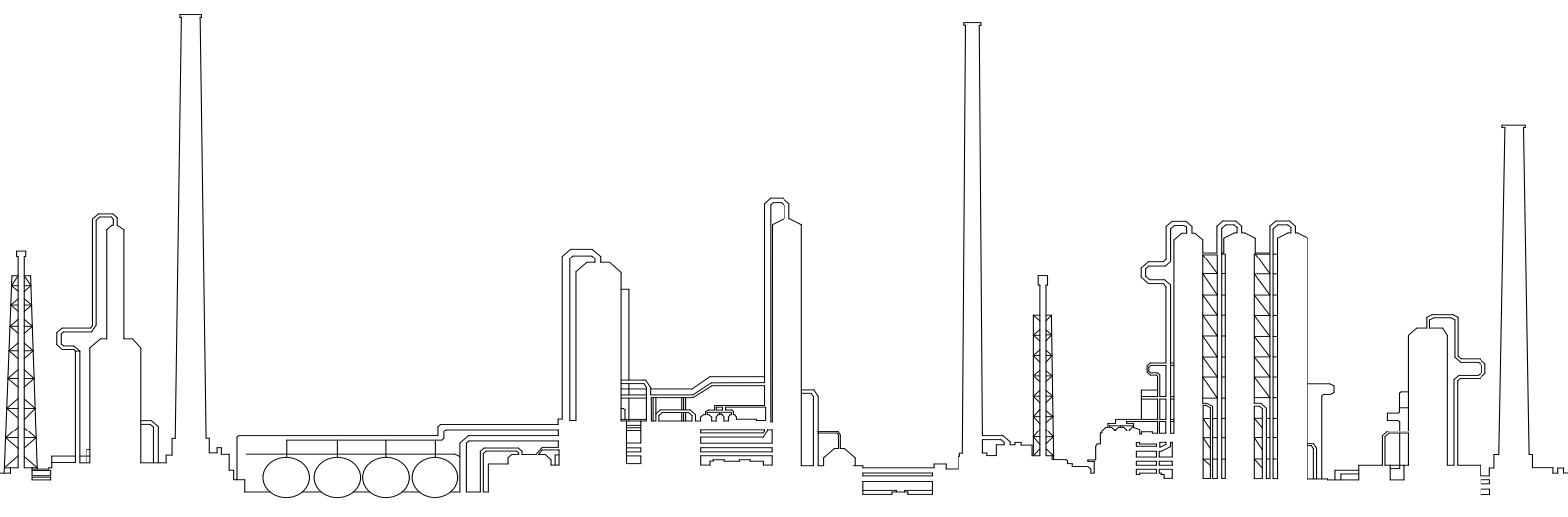




Universidad de Valladolid

Ph.D. Thesis

Modifier Adaptation for process optimization with uncertainty



Tania Rodríguez Blanco



**PROGRAMA DE DOCTORADO EN
INGENIERÍA INDUSTRIAL**

TESIS DOCTORAL:

**MODIFIER ADAPTATION FOR
PROCESS OPTIMIZATION
WITH UNCERTAINTY**

Presentada por Tania Rodríguez
Blanco para optar al grado de
Doctor/a por la Universidad de
Valladolid

Dirigida por:
D. César de Prada Moraga
D. Daniel Sarabia Ortiz

Agradecimientos

A la financiación recibida del Ministerio de Economía, Industria y Competitividad del Gobierno de España bajo la beca predoctoral FPI de formación de personal investigador BES-2013-062737 asociada al proyecto “Operación óptima en tiempo real de sistemas de planta completa”.

A mis directores de tesis, César de Prada y Daniel Sarabia, por todo el conocimiento que me han aportado, consejos y por confiar en mí para la realización de esta tesis.

Al personal investigador del Laboratorio de Control Automático de la Escuela Politécnica Federal de Lausanne, en especial a Dominique Bonvin y Alejandro Marchetti por brindarme la oportunidad de trabajar con ellos y de mantener una fructífera colaboración.

A los profesores del grupo de investigación “Control y supervisión de procesos” por la formación recibida. A mis compañeros del Departamento de Ingeniería de Sistemas y Automática: José Luis Pitarch, Carlos Gómez y Álvaro Serna, por todos los buenos momentos pasados juntos. En especial, quisiera agradecer a Daniel Sarabia y Rubén Martí, porque además de ayudarme en los momentos más difíciles me han brindado su amistad, gracias.

A mi amiga desde la infancia, Ana, gracias por estar siempre a mi lado a pesar de la distancia y por hacer la mejor portada que podría tener esta tesis. Porque volver a nuestra tierra siempre es mejor si sé que estarás allí.

A mis padres, Conchi y Claudino, porque gracias a su esfuerzo, apoyo y cariño he llegado hasta aquí y gracias también por darme los mejores hermanos que se pueden tener, Javier y Laura.

Y por supuesto, a Igor, mi compañero de vida, por todo su apoyo, cariño y comprensión.

Contents

Abstract / Resumen	11
Abstract	13
Resumen	15
1 Introduction.....	19
1.1 Process Optimization	21
1.1.1 Real Time Optimization	23
1.1.2 Problem formulation	24
1.2 Handling uncertainties in process optimization	31
1.2.1 Stochastic programming	32
1.2.1.1. Two-stage programming	33
1.2.1.2. Multistage programming	33
1.2.2 Robust optimization	35
1.2.3 Model Parameter Adaptation	36
1.2.4 Direct Input Adaptation	36
1.2.5 Modifier Adaptation methodology	38
1.3 Motivation	39
1.4 Objectives and working plan	40
1.5 Organization of the thesis	41
1.6 Contributions and publications	42
2 Modifier Adaptation methodology for Real Time optimization.....	47
2.1 State of the art: Modifier Adaptation methodology	49
2.1.1 Integrated System Optimization and Parameter Estimation (ISOPE)	50
2.1.2 Modifier Adaptation methodology	52
2.1.3 Dual Modifier Adaptation	58
2.1.4 Iterative gradient-modification optimization combined with derivative free optimization	60
2.1.5 Nested Modifier Adaptation	62
2.1.6 Directional Modifier Adaptation	65
2.1.7 Summary of MA applications in literature	67

2.2	Implementation issues	69
3	Handling operational constraints in Modifier Adaptation	73
3.1	Introduction	75
3.2	Alternative MA using the Lagrangian gradients	76
3.3	Efficient Nested Modifier Adaptation for dealing with operational constraints	79
3.4	Implementation in the Otto-Williams reactor	80
3.4.1	Process description	80
3.4.2	Steady-state RTO model	82
3.4.3	Optimization problem	83
3.4.4	Results	85
3.5	Conclusions	90
4	Modifier Adaptation using transient measurements	93
4.1	Introduction	95
4.2	Modifier Adaptation computing plant gradients from neighbouring extremal control	96
4.3	Combination of transient and steady-state measurements	100
4.4	Modifier Adaptation approach using Recursive Extended Least Squares to compute process gradients	101
4.4.1	BIBO stability of the closed-loop system	107
4.4.2	Implementation in the Otto-Williams reactor	109
4.4.3	Implementation in the laboratory-scale flotation column	113
4.4.3.1	Process description	113
4.4.3.2	Experimental set-up	116
4.4.3.3	Metallurgical model	118
4.4.3.4	RTO model and optimization problem	119
4.4.3.5	Results	121
4.5	Conclusions	122
5	Case study I: Depropanizer distillation column	125
5.1	Modelling and simulation	127
5.1.1	Dynamic rigorous model	130

5.1.2	Steady-state rigorous model	137
5.1.3	Steady-state simplified model	137
5.1.4	Simulation and comparison between models	143
5.2	Basic control	146
5.3	Model predictive control DMC	146
5.4	RTO problem formulation	147
5.5	Implementation of MA in the depropanizer distillation column ..	151
5.5.1	Static MA	155
5.5.2	Alternative NMA for dealing with operational constraints	160
5.5.3	MA combining transient and steady-state measurements	168
5.5.4	MA based on transient measurements	175
5.5.4.1	Evaluation of the dual constraint in transient MA	181
5.6	Conclusions	185
6	Case study II: Natural gas networks	187
6.1	Natural gas networks	189
6.2	Gas networks library	189
6.3	Gas networks modelling	192
6.3.1	Compressor station modelling	192
6.3.2	Gas pipelines dynamic rigorous modelling	200
6.4	Implementation of MA in Reganosa network	207
6.4.1	Reganosa network	207
6.4.2	Simplified RTO model	209
6.4.3	RTO problem formulation	210
6.4.4	Alternative NMA based on the Lagrangian function	212
6.5	Conclusions	216
7	Final conclusions and future work	219
7.1	Final conclusions	221
7.2	Future work	223
	References	227

ABSTRACT / RESUMEN

Abstract

In the chemical industry, the optimal management of processes has become a topic of great interest in recent years since it enables to operate and makes decisions efficiently with respect to economic, environmental criteria or product qualities. Within the process industry, this management is usually performed at a top layer in the hierarchical control structure called Real Time Optimization (RTO), which is based on process models and uses optimization methods to provide the optimal operation of the process.

However, optimal operation is not guaranteed since the process models used by RTO are always inaccurate, so the optimum computed from the model may not be the same as the optimum of the process. In addition, the layers of control structure use different models for making decisions, RTO typically involves nonlinear first-principles models that describe the steady-state behaviour of the plant, whereas MPC is usually based on dynamic linear models, creating inconsistencies between them that affect the final result.

The Modifier Adaptation methodology (MA) was developed to deal with these problems. This technique uses the process measurements to estimate the gradients of the process and, with this information, computes certain corrective terms that are added to the cost function and constraints of the optimization problem to lead the process to its optimum operating point despite the presence of structural uncertainty between the model used in the RTO layer and the process.

Nevertheless, this methodology has some limitations. One of the main drawbacks is related to the dimension of the problem with respect to the number of decision variables and constraints, which increase significantly the number of required modifiers, thus slowing down the convergence of the method considerably, making the implementation of MA in this type of problems infeasible. To try to solve this issue, in examples with many operational constraints, this thesis presents an alternative MA formulation that makes the number of modifiers depend solely on the number of process inputs, reducing the dimension of the problem. In addition, this formulation is implemented following the Nested Modifier Adaptation methodology (NMA), which does not require the calculation of gradients to obtain the modifiers, thus making the implementation of the algorithm easier and speeding up the convergence towards the plant optimum.

Another important limitation of the MA methodology is the necessity of waiting for the steady state of the process to update the modifiers at each iteration of the algorithm. In many real applications, for example the operation of distillation columns, this steady-state can be achieved after several hours of operation, which means that the convergence of the method is very slow and the optimum operating point is reached after several days of operation. This problem makes the implementation of this methodology ineffective in real processes, especially in those that have a long settling time. Therefore, this thesis has worked on the use of transient measurements to estimate the plant gradients and compute the modifiers during the transient, running the RTO layer with a higher frequency, without waiting for the steady-state of the plant, so the convergence rate to the optimum is sped up.

Furthermore, after reviewing the existing literature on MA, it can be seen that there are not many large scale applications or realistic problems with a hierarchical control structure. For this reason, this thesis also focuses on showing the potential of this tool and the great benefits that could be obtained from applying it in the process industry. The considered case studies are very realistic, large-scale problems, such as the transport of natural gas through gas networks and the operation of a depropanizer distillation column present in any refinery.

Resumen

La gestión óptima de procesos dentro de la industria química se ha convertido en un tema de gran interés a lo largo de los últimos años, ya que permite operar y tomar decisiones de forma eficiente con respecto a criterios económicos, medioambientales o de calidades de producto. Dentro de la industria de procesos, esta gestión suele realizarse en una capa superior dentro de la estructura jerárquica de control denominada Optimización en tiempo real (RTO), que en base a modelos del proceso y utilizando métodos de optimización proporciona las directrices óptimas de operación del proceso.

Sin embargo, los modelos utilizados nunca reflejan fielmente la realidad por lo que el óptimo calculado en base a estos modelos puede no corresponder al óptimo real del proceso. Además, las diferentes capas de la estructura de control usan modelos distintos, mientras que la capa RTO está basada en modelos estacionarios no lineales, la capa MPC utiliza modelos dinámicos lineales, de modo que podría existir una falta de coherencia entre los mismos que afecte al resultado final.

Para tratar esta problemática surge la metodología de adaptación de modificadores (MA) que utiliza las medidas del proceso para estimar los gradientes del proceso y con esta información calcular ciertos términos correctores que se añaden a la función de coste y a las restricciones del problema de optimización para conducir el proceso a su punto óptimo de operación a pesar de la presencia de incertidumbre estructural entre el modelo utilizado en la capa RTO y el proceso real.

Sin embargo, dicha metodología presenta ciertas limitaciones. Una de sus principales desventajas está relacionada con la dimensión del problema con respecto al número de variables de decisión y de restricciones que aumentan considerablemente el número de modificadores necesarios ralentizando mucho la convergencia del método, llegando a hacer inviable la aplicación de MA. Para tratar de resolver este problema en el caso de que existan numerosas restricciones operacionales, la presente tesis presenta una formulación alternativa de MA que hace que el número de modificadores dependa únicamente del número de entradas del proceso reduciendo así la dimensión del problema. Además, esta formulación se implementa siguiendo la metodología de adaptación de modificadores anidados (NMA) que evita el cálculo explícito de gradientes para la

obtención de los modificadores facilitando aún más la implementación del algoritmo y acelerando la convergencia hacia el óptimo de la planta.

Otra importante limitación de la metodología MA es la necesidad de esperar al estado estacionario del proceso para actualizar los modificadores en cada iteración del algoritmo. En muchas aplicaciones reales, por ejemplo, la operación de columnas de destilación, dicho estado estacionario puede alcanzarse después de varias horas de operación lo que supone que la convergencia del método sea muy lenta y el punto óptimo de operación se alcance después de varios días de operación. Éste problema hace que la implementación de esta metodología en procesos reales, especialmente en aquellos que presentan largos tiempos de asentamiento, no sea eficaz. Por ello, en esta tesis se ha trabajado en el uso de medidas transitorias del proceso para estimar los gradientes de la planta y estimar los modificadores durante el transitorio, ejecutando la capa RTO con una frecuencia mayor, sin esperar al estado estacionario de la planta, de modo que la velocidad de convergencia al óptimo se vea acelerada.

Además, después de revisar la literatura existente sobre la metodología MA uno puede darse cuenta que no existen muchas aplicaciones en problemas con un considerable número de variables o sobre ejemplos realistas que cuenten con una estructura jerárquica de control. Por ello esta tesis también está enfocada en demostrar el gran potencial que tiene esta metodología y los enormes beneficios que se podrían obtener si fuera aplicada en la industria de procesos. Los casos de estudio sobre los que se ha trabajado son ejemplos realistas y de gran escala como, el transporte de gas natural o la operación de una columna de destilación despropanizadora presente en todas las refinerías.

1 INTRODUCTION

This chapter introduces the key role that process optimization plays in the chemical industry, the basic ideas to formulate optimization problems and the way to carry out the optimal operation in practice through Real Time Optimization. The issue of model uncertainty to manage processes efficiently is also addressed, describing the current methods to solve it and providing the motivation of this thesis to contribute to the research and development of new methods in this field. Finally, the chapter concludes with the main contributions of the present thesis and its organization.

1.1 Process Optimization

The growing competition in the chemical industry has led to a determination to find the optimal way to operate processes so as to reduce costs, increase benefits, maximize process efficiency and improve the quality of the obtained products. For this reason, process optimization has evolved from a methodology of academic interest into a technology with a significant impact in engineering practice.

In the chemical industry, optimal operation is typically addressed by a hierarchical structure, as shown in Figure 1.1.

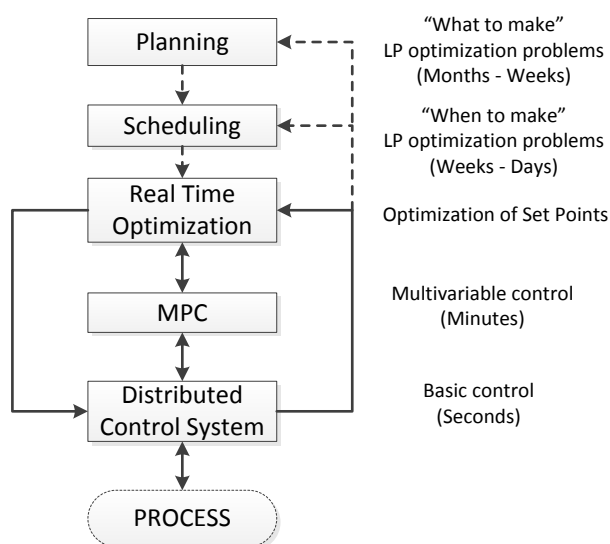


Figure 1.1. Hierarchical control structure.

This control structure is formed by several layers with different objectives that are defined as follows:

- **Planning**: the aim of planning is to answer the question "what to make", that is, decide what feedstock to purchase, which products to make, and how much of them. In almost all large scale chemical plants, a linear program (LP), or successive LPs, are used for planning and are based on economics, typically an overall plant profit objective function, as well as forecasts of some aspects such as price variations or future demands.

- Scheduling: this is a decision-making process to determine “when”. Given a plan in a specific time horizon, scheduling addresses the timing of the necessary actions and events to execute that plan, optimizing work and workloads in the production process.
- Real Time Optimization (RTO): this provides the bridge between plant scheduling and process control. At this level, medium-term decisions are made, typically on a time scale of hours, by explicitly considering economics in operations decisions. This step relies on a real-time optimizer that determines the optimal operating point under changing conditions, such as changes in raw material quality or in the operating conditions. The operating point computed by this layer is characterized by set points for a set of controlled variables that are passed on to the lower-level controllers. RTO typically involves nonlinear first-principle models describing the steady-state behaviour of the real process. These models are often relatively large, and the model-based optimization may require substantial computing time (Marlin & Hrymak, 1997).
- Multivariable predictive control (MPC): this provides multivariable dynamic control of the plant, giving some amount of optimization capability. The associated time scale is usually in the order of minutes.
- Distributed control system (DCS): at the lowest level, measurements made by sensors are collected from the plant, and basic flow, pressure and temperature control is implemented on a time scale of seconds. The DCS layer is typically the main operator interface for monitoring and controlling the plant, including generating alarms for abnormal situations. Also implemented in this layer are advanced regulatory controls, such as cascade, ratio and constraint controllers, as well as sequencing controls.

The multilevel structure leads to a vertical decomposition of the automation tasks. However, this has some drawbacks. The main limitation is that sampling and optimization have to be delayed until the controlled plant has settled to a new steady state (Engell, 2007). This delay occurs at each RTO step after a change in the input variables, and worse, after the occurrence of disturbances, so the adaptation of the operating conditions can be slow. Inconsistencies may also arise from the use of different models at the different levels. For instance, RTO typically involves nonlinear first-principles models that describe the steady-state behaviour of the plant, whereas MPC is usually based on dynamic linear models obtained

empirically. So, there might be inconsistencies between them that affect the final result.

Furthermore, RTO presents important challenges that have to be addressed, such as proposing algorithms that manage the existing model uncertainty, handling the interaction between the RTO and MPC layers, or speeding up the execution rate of RTO to adapt the operating conditions faster. These issues, and some others, will be explained in more detail throughout this thesis.

1.1.1 Real Time Optimization

RTO emerged in the seventies together with model predictive control (MPC), at the time when on-line computer control of chemical plants became available. Since then, RTO has been widely applied in the process industry. The applications go far beyond the chemical and petrochemical industries, and include food production or biological processes, among many other applications. These plants are made up of a large number of interconnected units, so achieving an optimal management is not a trivial task because of the inherent difficulties of the process itself, as well as the uncertainties and disturbances that continuously modify the operating conditions. Therefore, the development of a systematic mechanism, such as RTO, to achieve optimal operation has been most useful for the process industry. Thanks to RTO, important economic payoffs can be obtained since RTO aims to reduce costs and improve profitability. Its success has meant that many companies have developed RTO solutions and related software, whose use is increasing due to the highly competitive market.

RTO consists of solving an optimization problem by considering economic objectives through the explicit use of models for making decisions in real time that are passed to lower level controllers. One of the main challenges in RTO systems comes from the fact that models are simplified representations of reality and are thus subject to uncertainty. The most intuitive way to deal with this problem is to use process measurements to update the process model in order to give a better prediction of the plant outputs. This is the classical “two-step” approach, which consists of a parameter estimation step followed by an economic optimization step to compute the new operating point to be applied to the process in an iterative scheme until no further improvements in the cost function are observed (Behrens, et al., 2014).

However, in the presence of structural plant-model mismatch and unmeasured disturbances, the solution provided by the “two-step” approach might result in suboptimal, or even infeasible, plant operation. Even in the presence of parametric uncertainty, this methodology does not always provide correct results, since it is not straightforward to decide which model parameters to adapt by means of parameter estimation and which ones to keep at fixed values. In addition, these parameters should be identifiable from the available measurements, but this is not always the case (Yip & Marlin, 2002).

Hence, there is a clear interest in devising and studying RTO methodologies that are tailored to enforcing feasibility and optimality, while alleviating model accuracy and updating requirements. With this in mind, this thesis describes the main advances in RTO and develops new techniques to improve its applicability in the process industry.

1.1.2 Problem formulation

From a practical point of view, the optimization task consists of finding the best operating point that satisfies process specifications and constraints on many variables. To carry out this task a mathematical optimization problem that contains the following elements must be formulated (Biegler, 2010):

- **Objective function:** a quantitative performance measure to be minimized or maximized. This can for instance be operational costs, yield or profit.
- **Model:** set of equations that describe the behaviour of the process. For the optimization problem, this translates into a set of equations and inequalities named constraints which comprise a feasible region that defines the performance limits for the system.
- **Decision variables:** these are degrees of freedom in the process, that is, variables that can be adjusted to satisfy the constraints and find the optimal value for the objective function.
- **Operating constraints:** the limitations on the values of some variables that can be divided mainly into four groups: equipment constraints (valve position, compressor speed ...), safety constraints (explosive limit, critical pressure...), quality constraints (reaction yield, purity...) and environmental constraints (pollutant emissions).

RTO typically uses an approximate nonlinear steady-state model of the process to make decisions:

$$\mathbf{F}(\mathbf{x}, \mathbf{u}, \boldsymbol{\beta}) = 0, \quad \mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u}) \quad (1.1)$$

where \mathbf{F} and $\mathbf{h} \in \mathcal{R}^{n_h \times 1}$ are vectors of nonlinear functions, $\boldsymbol{\beta} \in \mathcal{R}^{n_\beta \times 1}$ the model parameters, $\mathbf{u} \in \mathcal{R}^{n_u \times 1}$ the decision variables, $\mathbf{y} \in \mathcal{R}^{n_y \times 1}$ the outputs predicted by the model and $\mathbf{x} \in \mathcal{R}^{n_x \times 1}$ accounts for some internal model variables.

Then, RTO involves solving a constrained, steady-state optimization problem whose general formulation is given by (1.3) where ϕ is the cost function to be minimized and g_i represents the inequality constraints ($i = 1, \dots, n_g$) which can be approximated from the knowledge of \mathbf{u} and \mathbf{y} as (1.1) shows. \mathbf{h} is the set of n_h equality constraints given by the model, $\boldsymbol{\beta} \in \mathcal{R}^{n_\beta \times 1}$ the model parameters, and $\mathbf{u} \in \mathcal{R}^{n_u \times 1}$ the decision variables which present lower and upper limits \mathbf{u}^l and \mathbf{u}^u .

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi(\mathbf{u}, \mathbf{y}(\mathbf{u}, \boldsymbol{\beta})) \\ \text{s.t.} \quad & g_i(\mathbf{u}, \mathbf{y}(\mathbf{u}, \boldsymbol{\beta})) \leq 0 \quad i = 1, \dots, n_g \\ & \mathbf{h}(\mathbf{u}, \mathbf{y}(\mathbf{u}, \boldsymbol{\beta})) = 0 \\ & \mathbf{u}^l \leq \mathbf{u} \leq \mathbf{u}^u \end{aligned} \quad (1.2)$$

For simplicity, notation $\phi(\mathbf{u}, \boldsymbol{\beta})$ will stand for $\phi(\mathbf{u}, \mathbf{y}(\mathbf{u}, \boldsymbol{\beta}))$ in the rest of the thesis with no loss of generality.

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi(\mathbf{u}, \boldsymbol{\beta}) \\ \text{s.t.} \quad & g_i(\mathbf{u}, \boldsymbol{\beta}) \leq 0 \quad i = 1, \dots, n_g \\ & \mathbf{h}(\mathbf{u}, \boldsymbol{\beta}) = 0 \\ & \mathbf{u}^l \leq \mathbf{u} \leq \mathbf{u}^u \end{aligned} \quad (1.3)$$

Assume ϕ and g_i are twice continuously differentiable in a neighbourhood of \mathbf{u}^* , the gradients of the active constraints $g_i(\mathbf{u}^*, \boldsymbol{\beta})$ are linearly independent, and that \mathbf{u}^* is a local minimizer. The first-order necessary conditions of optimality for the problem (1.3), also known as the Karush-Kuhn Tucker (KKT) conditions, must be hold at \mathbf{u}^* , where \mathbf{u}^* is a local optimum of problem (1.3) and they are defined as follows:

$$g_i \leq 0 \quad \mathbf{u}^l \leq \mathbf{u} \leq \mathbf{u}^u \quad (1.4)$$

$$\mu_i g_i = 0 \quad \zeta^{u^T} (\mathbf{u} - \mathbf{u}^u) = 0 \quad \zeta^{L^T} (\mathbf{u}^L - \mathbf{u}) = 0 \quad (1.5)$$

$$\mu_i \geq 0 \quad \zeta^{u^T} \geq 0 \quad \zeta^{L^T} \geq 0 \quad (1.6)$$

$$\frac{\partial L}{\partial \mathbf{u}} = \frac{\partial \phi}{\partial \mathbf{u}} + \sum_{i=1}^{n_g} \mu_i^T \frac{\partial g_i}{\partial \mathbf{u}} + \zeta^u - \zeta^L = 0 \quad (1.7)$$

where the Lagrangian L is defined as:

$$L(\mathbf{u}, \boldsymbol{\mu}, \boldsymbol{\beta}, \zeta^u, \zeta^L) := \phi + \sum_{i=1}^{n_g} \mu_i g_i + \zeta^{u^T} (\mathbf{u} - \mathbf{u}^u) + \zeta^{L^T} (\mathbf{u}^L - \mathbf{u}) \quad (1.8)$$

where $\boldsymbol{\mu} \in \mathfrak{R}^{n_g}$, $\zeta^u, \zeta^L \in \mathfrak{R}^{n_u}$ are the Lagrange multiplier vectors.

The necessary conditions of optimality can be divided into: (1.4), the primal feasibility conditions; (1.5), the complementary slackness conditions; and (1.6), (1.7) that are referred to as the dual feasibility conditions. Any point \mathbf{u}^* for which there exist Lagrange multipliers $\boldsymbol{\mu}^*, \zeta^{u^*}, \zeta^{L^*}$, such that, $(\mathbf{u}^*, \boldsymbol{\mu}^*, \zeta^{u^*}, \zeta^{L^*})$ satisfies the KKT conditions is called a KKT point.

These conditions are necessary conditions that must hold at each local minimum; however, they may be satisfied by a point that is not a local minimum. Sufficient conditions must be satisfied for a KKT point to be a strict local optimum.

In order to assure that the solution given by the first order KKT conditions is a minimum, and not a maximum or a saddle point of the Lagrangian L in $(\mathbf{u}^*, \boldsymbol{\mu}^*)$, the second order conditions are formulated.

Second order conditions provide sufficient optimality conditions for the solution and are given by the following expression involving the Hessian $\nabla_{\mathbf{u}}^2 L$:

$$\mathbf{z}(\mathbf{u}^*)^T \nabla_{\mathbf{u}}^2 L \mathbf{z}(\mathbf{u}^*) > 0 \quad (1.9)$$

So, the Hessian of L with respect to \mathbf{u} is positive definite in relation to all vectors \mathbf{Z} orthogonal to the gradients of the active constraints at \mathbf{u}^* .

Although, RTO is traditionally based on solving nonlinear steady-state problems, it is important to note that there are many other classes of optimization problems, depending on several factors, such as the kind of functions and variables involved in the problem or the dependence of these functions on time. A possible classification is shown in Figure 1.2.

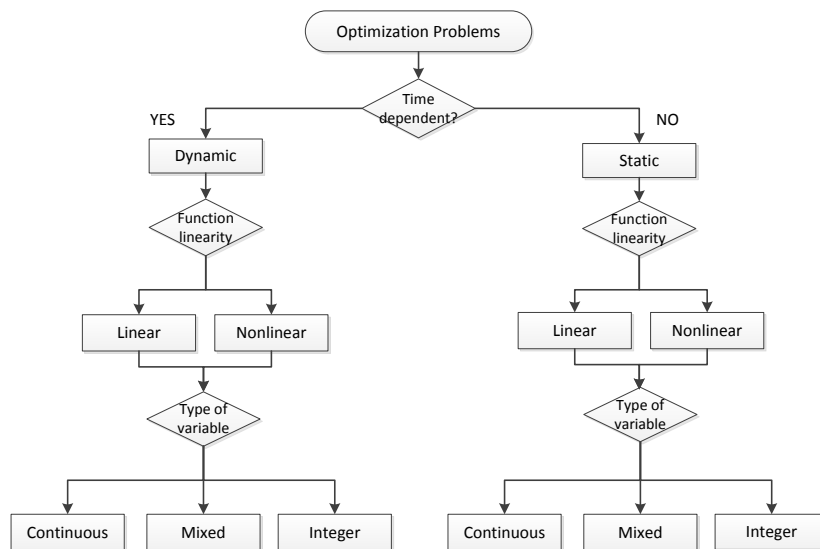


Figure 1.2. Classes of optimization problems.

The first separation in the classification considers whether model equations depend on time or not, distinguishing between dynamic and static problems. The linearity of the equations divides the optimization problems into two groups: linear (all the functions involved are linear) or nonlinear (any of the functions are nonlinear).

If the variables involved in the problem are continuous, there is a continuous problem; if all of them are discrete, the problem is called integer, and finally, if there are both types of variables, the problem is mixed. The discrete variables are often restricted to taking 0 or 1 values to define logical or discrete decisions, such as the assignment of equipment and sequencing of tasks.

There are other classes of optimization problems that have not been shown in Figure 1.2, such as deterministic or stochastic problems. In deterministic problems, the input data for the given problem is known accurately, whereas stochastic problems are solved under uncertainty, involving random variables. There is also multiobjective optimization, which involves minimizing or maximizing multiple objective functions subject to a set of constraints.

There is an important distinction in whether they are convex or nonconvex. A problem is considered convex if the following conditions are satisfied:

- A set $\mathbf{S} \in \mathfrak{R}^{nu}$ is convex if and only if all points on the straight line connecting any two points in this set are also within this set. For all $\alpha \in (0,1)$ and $u_1, u_2 \in \mathbf{S}$, this can be stated as:

$$u(\alpha) = \alpha u_1 + (1 - \alpha) u_2 \in \mathbf{S} \quad (1.10)$$

- A function $\phi(\mathbf{u}, \boldsymbol{\beta})$ is convex if its domain \mathbf{U} is convex and

$$\alpha \phi(u_1) + (1 - \alpha) \phi(u_2) \geq \phi(u(\alpha)) \quad (1.11)$$

holds for all $\alpha \in (0,1)$ and $u_1, u_2 \in \mathbf{X}$

- Convex feasible regions require $g_i(\mathbf{u}, \boldsymbol{\beta})$ to be convex functions and $\mathbf{h}(\mathbf{u}, \boldsymbol{\beta})$ to be linear.
- A function ϕ is concave if $-\phi$ is convex.

If (1.3) is a convex problem, any local solution, that is, a feasible solution that cannot be improved within a neighbourhood around this one, is guaranteed to be a global solution to (1.3), i.e., a better solution cannot be found. However, nonconvex problems may have multiple local solutions.

Other important kind of optimization problems are the dynamic ones which play an important role in disciplines such as chemical industry or aerospace applications. A dynamic optimization problem (DAE) is formulated as follows:

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi(\mathbf{u}, \boldsymbol{\beta}) = \int_{t_0}^t \Gamma(\mathbf{x}, \mathbf{u}, \boldsymbol{\beta}) \\ \text{s.t.} \quad & \frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \boldsymbol{\beta}), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \\ & g_i(\mathbf{x}, \mathbf{u}, \boldsymbol{\beta}) \leq 0 \quad i = 1, \dots, n_g \\ & \mathbf{h}(\mathbf{x}, \mathbf{u}, \boldsymbol{\beta}) = 0 \\ & \mathbf{u}^l \leq \mathbf{u} \leq \mathbf{u}^u \end{aligned} \quad (1.12)$$

where $\mathbf{x}(t) \in \mathfrak{R}^{n \times 1}$ are the state variables, which are functions of time, $t \geq 0$, $\mathbf{u}(t) \in \mathfrak{R}^{nu \times 1}$ are the decision variables and $\boldsymbol{\beta} \in \mathfrak{R}^{n\beta \times 1}$ are parameters that are independent of time. The main characteristics of this optimization problem are that some of the equations are given as differential equations

like \mathbf{f} , equality constraints \mathbf{h} , inequality constraints \mathbf{g} and decisions variables are made over time.

A wide variety of approaches have been developed to address the solution of (1.12). These strategies can be loosely classified as “Optimize then discretized” and “Discretized then optimize” (Biegler, 2010). A schematic of the different optimization strategies is shown in Figure 1.3.

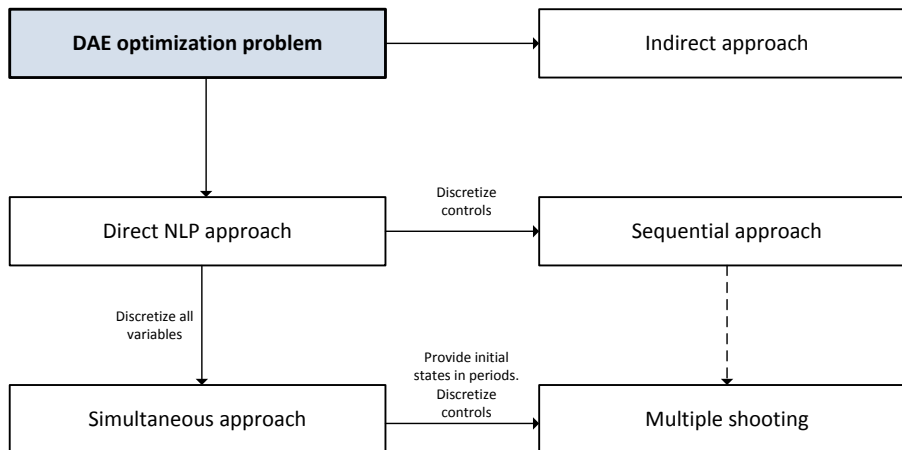


Figure 1.3. Schematic of DAE solver methods.

- Indirect methods:** indirect methods attempt to solve optimal control problems by seeking a solution to the necessary conditions of optimality, such as those presented earlier in (1.4)-(1.7). Many indirect methods use iterative procedures based on successive linearization to find a solution to the system of NCO. A nominal solution is chosen that satisfies part of the NCOs, then this nominal solution is modified by successive linearization so as to meet the remaining NCOs. Popular indirect methods for optimal control include quasi-linearization methods, gradient methods such as control vector iteration, and indirect shooting methods (see (Bryson & Ho, 1975)).
- Direct methods:** Nonlinear programming solvers NLP are a kind of optimization tool based on the availability of nonlinear models and first and second derivatives. As seen in Figure 1.3, methods which apply NLP solvers can be classified into two groups, the sequential and the simultaneous approaches.

Sequential approach: In the sequential methods, the control variables are discretized with control vector parameterization (CVP) methods, generating an equivalent problem that can be solved as one NLP, where the cost function and the constraints are obtained by integrating the system. For more information see

Examples of control profile of various degrees and continuity orders are shown in Figure 1.4 represented as piecewise polynomials.

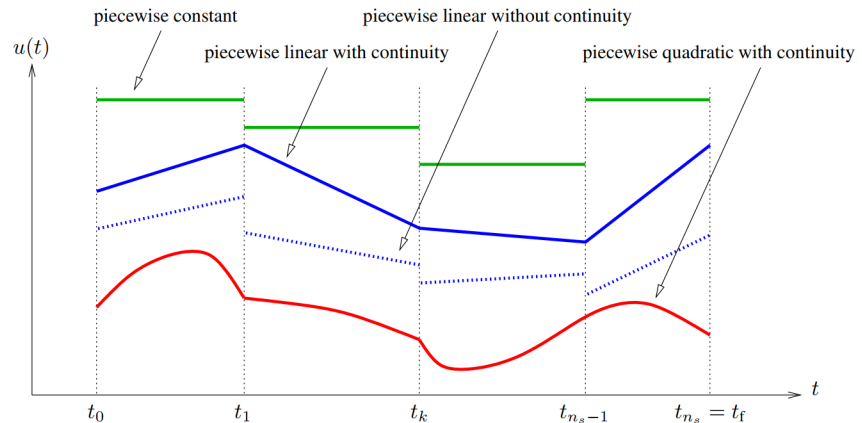


Figure 1.4. Examples of control variable profiles.

Simultaneous approach: In this case, a full discretization of the state and control profiles, is performed, generating an algebraic set of equations, so that, the dynamic optimization can be solved as a large NLP one. For more information see

Multiple shooting approach: Optimization with multiple shooting serves as a bridge between sequential and direct transcription and was developed to deal with unstable DAE systems. In this approach, the time domain is divided into smaller time elements and the DAE models are integrated separately in each element. In addition, the initial values of the state variables at every interval \mathbf{x}_i are incorporated as new decision variables. For an extended review of multiple shooting, (Bock & Plitt, 1984) and (Leineweber, 1999) are recommended.

1.2 Handling uncertainties in process optimization

Process optimization typically involves significant uncertainty due to imprecise knowledge of the plant operation, transforming deterministic problems into stochastic ones, the solution of which undoubtedly remains challenging and with a great practical importance. The uncertainties in process optimization can arise from two sources: those due to the random behaviour of the process variables and those due to the presence of plant-model mismatch, which can be divided into structural mismatch arising from the approximation of the underlying process mechanisms, parametric mismatch that stems from uncertain process parameters and process disturbances.

To deal with uncertainty, several approaches have been developed such as stochastic programming or robust optimization. Other approaches focus on modifying the model or the optimization problem from process measurements to cope with modelling errors.

There are basically three groups of methods for dealing with uncertainty depending on the use of process measurements. The first approach, the most intuitive, consists of using the process measurements to update the model parameters and then using this corrected model to carry out the economic optimization (Chen & Joseph, 1987). This formulation, the “two-step” approach, is solved iteratively until the algorithm converges to an optimal solution, i.e. once a steady-state of the process is achieved, both problems are solved (parameter estimation and economic optimization). However, the presence of significant structural uncertainty about the model and the real process means the described methodology is not able to achieve the plant optimum. A second technique uses the process measurements to estimate the gradients of the objective function of the process with respect to the decision variables. It does so in order to compute some modifiers for the cost function and the constraints of the optimization problem to enforce the necessary optimality conditions (NCO) of the modified optimization problem, matching them to the ones for the real process. In this way, the algorithm will converge to the plant optimum. This method is named ISOPE (Integrated System Optimization and Parameter Estimation) (Roberts, 1979) and represents the base for the Modifier Adaptation methodology (MA). The MA technique uses the process measurements to estimate the plant gradients and compute some corrector terms for the cost function and the constraints of the optimization problem without updating the model parameters. Instead of updating the model or

considering all the possible values of the uncertain parameters, MA changes the optimization problem, incorporating information from process measurements, so that the solution of the modified problem coincides with the solution obtained with a perfect model. The main advantage of MA is the capacity to converge to the plant optimum despite the presence of structural plant-model mismatch. The third group of methods incorporate the plant information so as to directly update the process inputs, replacing the optimization problem by a feedback control problem which tries to satisfy the process optimality conditions. Some examples are Extremum-Seeking Control (ESC) (Krstic & Wang, 2000) and Neighboring-Extremal Control (NEC) (François, et al., 2005).

1.2.1 Stochastic programming

In process optimization, an important source of uncertainty is the random behaviour of some parameters. When the parameters, or significant disturbances, are known only within certain bounds, one approach to tackling such problems is to take decisions that cover the different realizations of the uncertainty. In this case, the goal is to find a solution which is feasible and optimal for all such data in some sense. Stochastic programming models take advantage of the fact that the random variables present expected bounds and belong to a probability space with a given probability distribution function (PDF). The goal here is to find some policy that is feasible for all (or almost all) the possible data instances and maximizes the expectation of some function of the decisions and the random variables.

The first works concerning linear stochastic programming date back to the mid-1950s (Dantzig, 1955), (Beale, 1955). From then on, a huge amount of works have followed this research, making stochastic programming a very active field with applications in many different disciplines, such as operations research, finances and engineering.

One important formulation of stochastic programming considers a two- or multi-stage problem based on different scenarios of the uncertainty (Birge, 1997) as will be explained below. First, it is important to understand a general concept for stochastic programming called *recourse*. This concept refers to the fact that the decision-maker can adapt future decisions to future observations, which is usually illustrated in the separation of the decisions between *here-and-now* decisions that have to be fixed at a certain

time instant and the *wait-and-see* decisions, which can be adapted according to future observations.

1.2.1.1 Two-stage programming

The most widely applied and studied stochastic programming models are two-stage (linear) programs. Here the decision-maker takes some action in the first stage, after which a random event occurs affecting the outcome of the first-stage decision. A recourse decision can then be made in the second stage that compensates for any bad effects that might have been experienced as a result of the first-stage decision. The optimal policy from such a model is a single first-stage decision and a collection of recourse decisions (a decision rule) defining which second-stage action should be taken in response to each random outcome. Its formulation involves the assumption that the second-stage uncertain data can be modelled as a random vector with a known probability distribution (Shapiro & Philpott, 2007).

1.2.1.2 Multi-stage programming

The stochastic programming model discussed before is static in the sense that a decision (supposedly optimal) is made at one point in time, while accounting for possible recourse actions after all uncertainty has been resolved. But there are many problems where decisions should be made sequentially, at certain periods of time based on information available at each time period. Such multi-stage stochastic programming problems can be viewed as an extension of two-stage programming to a multi-stage setting. In each stage, decisions must be taken considering the previous ones and the probability distribution function of the random variable in the future, which is called a multi-stage recursion problem. It is assumed that, after a certain number of time periods or stages (robust horizon), it is possible to have information on the uncertainties, so the corresponding actions to be carried out at that time can be computed, assuming a known value of the uncertainty (Martí, 2015).

Quite often, multi-stage stochastic optimization is performed where, in the first stage, the uncertain variable (with robust horizon equal to one) can have any value, but in the second stage, they can be considered to be known. Notice that, in the first stage, a single decision must be taken in spite

of the different realizations of the uncertainty; while different decisions can be taken for every value of the uncertainty variables in the second stage.

In terms of the probability distribution function of the random variables, the most widely used method considers a *scenario representation* of the random space as a finite-dimensional approximation of the original problem, transforming the original stochastic optimization problem into a large scale deterministic equivalent (Dupacova, 1995).

The model uncertainty is taken into account by considering a tree of discrete scenarios for each possible value of the uncertainty, as depicted in Figure 1.5, where \mathbf{x}_k^j represents the state-vector at stage k in the position j , \mathbf{u}_k^j the vector of control inputs, and \mathbf{d}_k^j are the disturbances. The formulation of a scenario tree makes it possible to take into account, explicitly, the fact that future decisions can depend on new information (measurements) that will become available in the future.

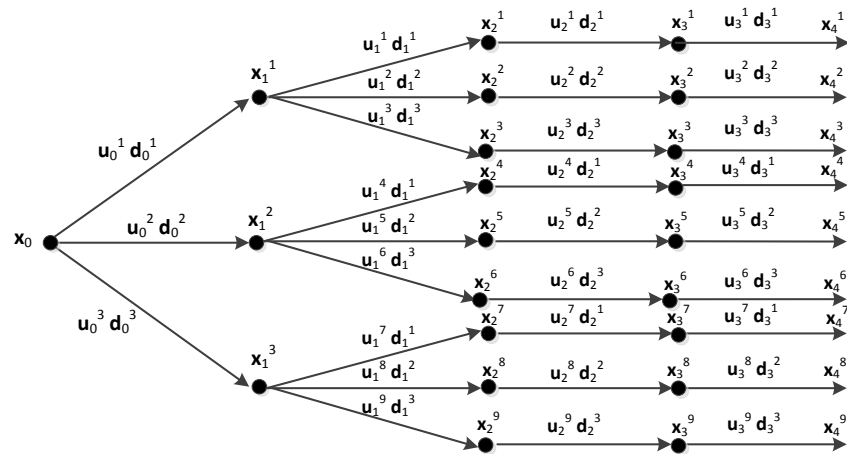


Figure 1.5. Scenario tree presentation of the uncertainty evolution for multi-stage programming.

In this way, the future control inputs are adapted according to the future realizations of the uncertainty and the conservativeness of this approach is reduced in comparison to other robust methods that search for a single sequence of control inputs to satisfy the constraints for all the possible values of the uncertainty.

1.2.2 Robust optimization

Robust optimization is an approach to optimization under uncertainty, in which the uncertainty model is not stochastic, but rather deterministic and set-based. Instead of seeking to immunize the solution in some probabilistic sense to stochastic uncertainty, here the decision-maker constructs a solution that is optimal for any realization of the uncertainty in a given set (Bertsimas, et al., 2011). In the early seventies, Soyster (Soyster, 1973) was one of the first researchers to investigate explicit approaches to Robust Optimization. The general formulation for Robust Optimization is described by (1.13).

$$\begin{aligned}
 \min_{\mathbf{u}} \quad & \phi(\mathbf{u}, \mathbf{s}_i) \\
 \text{s.t.} \quad & g_i(\mathbf{u}, \mathbf{s}_i) \leq 0 \quad \forall \mathbf{s}_i \in \mathbf{S}_i, \quad i = 1, \dots, n_g \\
 & \mathbf{h}(\mathbf{u}, \mathbf{s}_i) = 0 \\
 & \mathbf{u}^l \leq \mathbf{u} \leq \mathbf{u}^u
 \end{aligned} \tag{1.13}$$

Here $\mathbf{s}_i \in \mathfrak{R}^{n_s}$ are disturbance vectors or parameter uncertainties, and $\mathbf{S}_i \in \mathfrak{R}^{n_s}$ are uncertainty sets, which will always be closed. The objective of (1.13) is to find the decision variables that minimize the cost function among all the solutions which are feasible for all realizations of the disturbances \mathbf{s}_i within \mathbf{S}_i . This problem offers some measure of feasibility protection for optimization problems containing parameters which are not known exactly, since the solutions are computed to ensure feasibility when the problem parameters vary within the prescribed uncertainty set.

While this technique achieves the desired effect of immunizing the problem against parameter uncertainty, it is overly conservative for practical implementation (Bertsimas & Thiele, 2006). This issue was addressed by Ben-Tal and Nemirovski (Ben-Tal & Nemirovski, 1998), (Ben-Tal & Nemirovski., 1999) and El-Ghaoui and Lebret (El Ghaoui & Lebret, 1997), (El Ghaoui, et al., 1998) who proposed restricting the uncertain parameters to belong to ellipsoidal uncertainty sets, which removes the most unlikely outcomes from consideration and yields tractable mathematical programming problems. Other ways for modelling the uncertainty have also been proposed, for example, the use of polyhedral uncertainty sets (Bertsimas & Sim., 2004), or uncertainty sets described by more general norms (Bertsimas, et al., 2004).

1.2.3 Model Parameter Adaptation

Model parameter adaptation updates the parameters of the process model and repeats the optimization. It refers to the standard way of implementing RTO, called the “two-stage” approach (Chen & Joseph, 1987), also referred to in the literature as repeated identification and optimization. In the first step, the values of the adjustable model parameters β are estimated using the process output measurements. This is typically done by minimizing the lack of closure in the steady-state model equations, such as the weighted sum of squared errors between the measured outputs \mathbf{y}_p and the predicted outputs \mathbf{y} from the model. In order to implement this method, one has to select the set of model parameters to be adjusted, as these parameters should represent actual changes in the process and contribute to approaching the process optimum. Clearly, the smaller the subset of parameters, the better the confidence in the parameter estimates, and the lower the required excitation. However, too few adjustable parameters can lead to completely erroneous models, and thereby to a false optimum.

In the second step, the updated model is used to determine a new operating point by solving an economic optimization problem.

The interaction between the model-parameter adaptation and reoptimization steps must be considered carefully for the two-step approach to be able to achieve optimality. This is due to the fact that the objective of the parameter adaptation might be inconsistent with the economic optimization problem, since minimizing the mean-square error in \mathbf{y} may not help to find feasibility and optimality.

The convergence of this methodology has been addressed by several authors (Forbes & Marlin, 1994), (Biegler, et al., 1985); showing that optimal operation may be reached if model adaptation leads to matched Karush-Kuhn-Tucker (KKT) conditions for the model and the plant, which is not possible if the model presents structural uncertainty.

1.2.4 Direct Input Adaptation

Direct input adaptation turns the optimization problem into a feedback control problem and implements optimality via tracking of the necessary conditions of optimality.

The aim of this feedback control is to compute the set points for the manipulated variables while trying to maintain certain measures of

optimality fixed. The challenge is to find the optimality functions calculated from the measured variables that must be fixed by changing the decision variables to enforce the NCO of the plant. In other words, the goal of the control structure is to achieve a similar steady-state performance as would be realized by a fictitious, on-line optimizing controller.

In self-optimizing control (Skogestad, 2000), the idea is to use a process model to select linear combinations of the output variables, the tracking of which results in good performance. However, tracking linear combinations of the output variables, although very convenient, hardly provides any guarantee that a plant optimum is reached upon convergence. To circumvent this difficulty, a rather natural idea is to choose the controlled variables as the NCO components with the corresponding set points equal to 0, thereby enforcing the plant NCO. Two classes of approaches fall within this category: extremum-seeking control (Guay & Zhang, 2003), and NCO tracking (François, et al., 2005). Tracking the NCO of the plant consists of three steps: (i) determining the active set (positivity condition on Lagrange multipliers), (ii) enforcing the active constraints, and (iii) pushing the sensitivity to zero (Chachuat, et al., 2009). These techniques are briefly described below:

- **Extremum seeking** (Guay & Zhang, 2003): this is based on the assumption that the input (decision variable) to output (objective function) map is static. Under this assumption, one possibility to attain an extremum is to first estimate the gradient of the input–output map and then design a control law to keep it as close to zero as possible. This extremum-seeking controller is derived from the NCO. A popular strategy for estimating the process gradients consists of exciting the plant with a sinusoidal input in order to extract the information to compute the required gradients.
- **NCO tracking** (François, et al., 2005): this is a general framework that turns a (dynamic or static) optimization problem into a control problem. It uses the fact that, at the optimal operating point, the first order necessary optimality conditions must hold. Basically, the NCO are the controlled variables and the NCO tracking procedure adapts the inputs at given sample times. Instead of controlling “normal” measurements, the gradient is measured (or estimated), and used as a controlled variable. When a disturbance enters the process, the NCO tracking control scheme adapts the inputs iteratively such that the NCO are satisfied after some iterations.

1.2.5 Modifier Adaptation methodology

Modifier Adaptation methodology is one of the most used for real-time optimization of uncertain processes. MA modifies the cost and constraints of the economic optimization problem by adding some extra terms called modifiers and then repeating the optimization so that the KKT conditions of the model and the plant can match. These modifiers are computed using plant data and can be divided into three groups: constraint bias terms, cost-gradient modifiers and constraint-gradient modifiers.

The idea of adding a gradient modifier to the cost function of the optimization problem dates back to the work of Roberts in the late 1970s. Note that it was originally proposed in the framework of two-step methods to better integrate the model update and optimization sub-problems, which led to the so-called ISOPE approach (Roberts, 1979).

Gao and Engell (Gao & Engell., 2005) proposed adding first-order modifiers to the process-dependent constraints, in addition to the constraint bias terms. This modification allows, not only the values of the constraints, but also their gradients to be matched.

The adaptation methodology was finally formalized by Marchetti and co-workers, who presented the Modifier Adaptation Methodology (Marchetti, et al., 2009). The authors analyzed the convergence to the real optimum when the adaptation in the economic optimization is performed by correcting the gradients of the objective function and the constraints, from the point of view of the KKT matching between the model and the real process.

The modified optimization problem to be solved in the RTO layer is formulated as follows:

$$\begin{aligned}
 \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \boldsymbol{\beta}) + \boldsymbol{\lambda}_k^T (\mathbf{u} - \mathbf{u}_{k-1}^*) \\
 \text{s.t.} \quad & g_{M,i} = g_i(\mathbf{u}, \boldsymbol{\beta}) + \mathbf{v}_k^T (\mathbf{u} - \mathbf{u}_{k-1}^*) + \varepsilon_{i,k} \leq 0 \\
 & \mathbf{h}(\mathbf{u}, \boldsymbol{\beta}) = 0 \\
 & \mathbf{u}^l \leq \mathbf{u} \leq \mathbf{u}^u \\
 & i = 1, \dots, n_g
 \end{aligned} \tag{1.14}$$

where ϕ_M and g_M are the modified cost function and constraints; \mathbf{u}_{k-1}^* is the input applied in the previous steady state, that is, the optimal solution of

the previous RTO; and the modifiers λ , γ and ε are computed from the following equations:

$$\begin{aligned}\lambda_k^T &= \nabla_{\mathbf{u}} \phi_P|_{\mathbf{u}_{k-1}^*} - \nabla_{\mathbf{u}} \phi|_{\mathbf{u}_{k-1}^*} & \mathbf{V}_k^T &= \nabla_{\mathbf{u}} g_P|_{\mathbf{u}_{k-1}^*} - \nabla_{\mathbf{u}} g|_{\mathbf{u}_{k-1}^*} \\ \varepsilon_{i,k} &= g_{P,i}(\mathbf{u}_{k-1}^*) - g_i(\mathbf{u}_{k-1}^*)\end{aligned}\quad (1.15)$$

where the subscript P indicates that the cost function and constraint gradients, as well as the output variables, are all evaluated from process measurements.

This methodology will be explained in more detail in chapter two.

1.3 Motivation

The optimization of an operating process is not a trivial task, since the processes are very complex and there are a lot of disturbances that can affect the process operation. In addition, these processes are not perfectly known, so the presence of uncertainty is inherent to process optimization. Hence, the existence of a tool like RTO makes sense, since it is able to find the optimal operating point, despite uncertainties, using plant measurements; thus increasing the added value of a given process.

RTO is based on steady-state plant models which are generally simplified representations of reality and are thus subject to uncertainty. Therefore, the optimum computed from the model may not be the same as the optimum of the process. For this reason, the study and development of RTO methods able to overcome this mismatch is an important area for research in order to better deal with the uncertainties that affect process optimization, overcoming the partial knowledge of the process.

This thesis focuses on the use of the Modifier Adaptation methodology to deal with uncertainty in RTO. MA has become a very useful tool and several applications have shown the effectiveness of this methodology. Important features of MA are the fact that it uses a fixed model in the RTO layer and its ability to deal with structural uncertainty. However, it is subject to some limitations. More specifically, after the study of the different MA approaches, it has been concluded that the main disadvantages of this technique are three:

- It requires the gradients of the plant cost and constraints to be estimated with respect to the decision variables.

- When dealing with problems with many operational constraints and decision variables, the number of modifiers increases and therefore makes the computational task more complicated.
- The necessity of waiting for the steady state of the process to update the modifiers in each RTO iteration, since this optimization layer is based on stationary models.

These two problems slow down the convergence rate of this methodology, making its implementation in large-scale, real processes infeasible. To overcome these issues, this thesis tries to speed up the convergence of MA in three different ways:

- Reducing the number of required modifiers.
- Using transient measurements to estimate plant gradients.
- Improving the applicability of methods that do not require the computation of plant gradients like Nested-Modifier Adaptation (NMA) in order to speed up the convergence.

Furthermore, after reviewing the existing literature about MA, it is clear that there are not many large scale applications or realistic problems with a hierarchical control structure that could really be useful to show the potential of this tool. One of the objectives of this thesis, therefore, is to test MA in realistic case studies to show how powerful it is, as well as the enormous benefits it can bring to the chemical industry.

1.4 Objectives and working plan

The general objective of this thesis is to contribute to the development of MA by proposing methods that facilitate its applicability in real processes, speeding up the convergence rate to the optimum operating point. Moreover, the applicability of MA in large scale systems is demonstrated with examples of the implementation of the developed methods in such simulated systems as; the transportation of natural gas through a gas pipeline network, or the operation of a depropanizer distillation column of a petrol refinery. In addition, an experimental flotation column has also been used to test the new methods.

To reach this general objective, the following particular objectives have been considered:

- The study and implementation of different MA approaches found in the literature.
- Building large scale dynamic models, based on first principles, to be considered as real processes in simulation for the natural gas network and the depropanizer distillation column.
- Applying the existing MA techniques in large scale problems with a hierarchical control structure.
- Model reduction to be used in the RTO layer.
- The study of the reduction in the number of modifiers in MA, formulating an alternative way to compute the modifiers using Lagrangian functions.
- The study of different plant gradients estimation techniques proposing a method based on transient information to speed up the convergence rate of the Modifier Adaptation.
- Testing the proposed approaches in realistic problems, such as the operation of a depropanizer distillation column and the transportation of natural gas through gas networks. A well-known benchmark, the Otto-Williams reactor, has also been considered.
- Testing the proposed approach based on transient information in real applications, such as a laboratory-scale flotation column for copper concentration.

1.5 Organization of the thesis

The thesis is organized as follows: chapter two reviews the existing MA approaches, describing how to apply them to real processes and analysing the implementation issues. Chapter three explains a new methodology to handle operational constraints in MA and its application to the Otto Williams reactor. In chapter four, different MA techniques based on transient information are described and implemented in the Otto Williams reactor and in a laboratory-scale flotation column. These approaches are the following:

- Modifier Adaptation computing plant gradients from neighbouring extremal control.

- Modifier Adaptation approach using Recursive Extended Least Squares to compute process gradients.
- A combination of transient and steady-state measurements.

The considered case studies are described in chapter five and chapter six showing the results obtained after the implementation of the presented MA approaches. Two case studies have been used that corresponds to the transport of natural gas through gas pipelines and the operation of a depropanizer distillation column of a petrol refinery. The complete models to simulate the real processes and the reduced ones to be used in the RTO layer are described. Finally, conclusions and future work are presented in chapter seven.

1.6 Contributions and publications

Journal papers:

- F. Blasco, E. Pareja, T. Rodríguez, D. Sarabia, C. de Prada. Sisgas, un proyecto español para la optimización de la medición en redes de transporte de fluidos preciosos, gas natural y derivados petrolíferos. *Revista Industria Química*, 39, Octubre 2016.
- T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Optimización en Tiempo Real utilizando la Metodología de Adaptación de Modificadores. *RIAI: Revista Iberoamericana de Automática e Informática industrial*, 2017.
- T. Rodríguez-Blanco, D. Sarabia, J.L. Pitarch, C. de Prada. Modifier Adaptation methodology based on transient and static measurements for RTO to cope with structural uncertainty. *Computers & Chemical Engineering*, 2017.

Book chapters:

- Proceedings of the 24th European Symposium on Computer Aided Process Engineering – ESCAPE 24 June 15-18, 2014, Budapest, Hungary. Elsevier B.V., 2014.
- Proceedings of the 27th European Symposium on Computer Aided Process Engineering – ESCAPE 27 October 1-5, 2017, Barcelona, Spain. Elsevier B.V., 2017.

Congress contributions:

- T. Rodríguez-Blanco, D. Sarabia, J.L. Morales, C. de Prada. Modelling and supervision of natural gas networks. European Gas Technology Conference (EGATEC 2013), May 30-31, 2013. Paris (France). Poster.
- T. Rodríguez-Blanco, D. Sarabia, C. de Prada, J.L. Morales. Simulación y control de compresores centrífugos usando EcosimPro. XXXIV Jornadas de Automática, September 4-6, 2013. Terrassa (Barcelona). Poster.
- T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Optimización de planta completa con incertidumbre. XIII Simposio CEA-IFAC en Ingeniería de Control. February 6-7, 2014. Universidad de Valladolid. Oral presentation.
- T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modelling and optimization of natural gas networks. ESCAPE24 (European Symposium on Computer Aided Process Engineering). June 15-18, 2014. Budapest (Hungary). Poster.
- R. Martí, T. Rodríguez-Blanco, J.L. Pitarch Pérez, D. Sarabia, C. de Prada. Optimización dinámica mediante diferenciación automática usando EcosimPro y Casadi. XXXV Jornadas de Automática, September 3-5, 2014, Valencia (Spain). Poster.
- T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier-Adaptation methodology for RTO applied to distillation columns. ADCHEM 2015. June 7-11, 2015. British Columbia (Canada). Oral presentation.
- M. Rannany, T. Rodríguez-Blanco, C. de Prada. Operación óptima en EcosimPro de estaciones de regulación y medida en la red de gas natural. XXXVI Jornadas de Automática, September 2-4, 2015, Bilbao (Spain). Poster.
- J.L. Pitarch, D. Sarabia, T. Rodríguez-Blanco, C. G. Palacín, C. de Prada. Energy consumption optimization in an evaporation process using NLP and a reformulated modifier-adaptation methodology. 10th European Congress of Chemical Engineering, ECCE 2015. September 27 – October 1, 2015. Nize (France). Oral presentation.
- T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Optimización jerárquica de planta completa con incertidumbre. XIV Simposio CEA-IFAC en

Ingeniería de Control. March 10-11, 2016. Universidad de La Rioja. Oral presentation.

- T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier-Adaptation approach to deal with structural and parametric uncertainty. DYCOPS-CAB 2016. June 4-9, 2016. Trondheim (Norway). Oral presentation.
- F. Blasco, E. Pareja, T. Rodríguez, D. Sarabia, C. de Prada. SIGGAS: Un proyecto español para la optimización de la medición en redes de transporte de fluidos preciosos, gas natural y derivados petrolíferos. Conferencia ISA anual, June 16, 2016. Madrid (Spain). Oral presentation.
- T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier-Adaptation methodology for RTO applied to Distillation Columns using a simplified steady-state model. MSC 2016. September 19-22, 2016. Buenos Aires (Argentina). Oral presentation.
- T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier-Adaptation approach using RELS to compute process gradients. FOCAPO-CPC 2017. January 8-12, 2017. Tucson (Arizona), United States. Poster.
- D. Navia, A. Puen, L. Bergh, T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier-Adaptation based on transient measurements applied to a laboratory-scale flotation column. Symposium on Computer Aided Process Engineering-ESCAPE 27, Barcelona, Spain, 2017. Poster.
- T. Rodríguez-Blanco, D. Sarabia, D. Navia, C. de Prada. Efficient Nested Modifier Adaptation for RTO using Lagrangian functions. Symposium on Computer Aided Process Engineering-ESCAPE 27, Barcelona, Spain, 2017. Oral presentation.
- Fernando Blasco, Elena Pareja, Tania Rodríguez, Daniel Sarabia, César de Prada. SIGGAS, a Spanish Project to optimize measurement in the transport networks of precious fluids, Natural Gas and Petroleum derivatives. Automation Instrumentation Summit 2017, Pavia, Italy, July 5th-6th 2017. Oral presentation.

2 MODIFIER ADAPTATION METHODOLOGY FOR RTO

Modifier Adaptation methodology has become a powerful tool for RTO of uncertain processes. This chapter presents an overview of the different approaches that have been developed in this field and the way to implement them. Then, a summary of applications is made followed by the explanation of some implementation issues that remain challenging within this methodology.

2.1 State of the art: Modifier Adaptation methodology

RTO comprises a set of optimization methods that incorporate plant measurements in the optimization framework to drive a real process to the optimum operating point, while guaranteeing constraint satisfaction. Process optimization is typically carried out by following a sequence of steps that include, firstly, process modelling, secondly, numerical optimization using the process model, and lastly, the application of optimal inputs to the plant (Marchetti, et al., 2016).

RTO typically involves solving a constrained, steady-state optimization problem whose general formulation is given by (2.1), where ϕ is the cost function to be minimized, g_i represents the inequality constraints ($i = 1, \dots, n_g$), \mathbf{h} the set of n_h equality constraints given by the model, $\boldsymbol{\beta} \in \mathcal{R}^{n_{\beta} \times 1}$ the model parameters, and $\mathbf{u} \in \mathcal{R}^{n_u \times 1}$ the decision variables which present lower and upper limits \mathbf{u}^L and \mathbf{u}^U .

$$\begin{aligned}
 \min_{\mathbf{u}} \quad & \phi(\mathbf{u}, \boldsymbol{\beta}) \\
 \text{s.t.} \quad & g_i(\mathbf{u}, \boldsymbol{\beta}) \leq 0 \quad i = 1, \dots, n_g \\
 & \mathbf{h}(\mathbf{u}, \boldsymbol{\beta}) = 0 \\
 & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U
 \end{aligned} \tag{2.1}$$

However, the model optimum obtained by solving (2.1) may not be the same as the plant optimum unless the model is a perfect representation of the process and the value of all disturbances is known. This often results in suboptimal plant operation and even in constraint violation.

RTO has emerged over the past forty years and uses diverse strategies to overcome the difficulties associated with plant-model mismatch by incorporating process measurements in the optimization framework to combat the adverse effect of uncertainty. This uncertainty can have three main sources; namely, parametric uncertainty, when the values of the model parameters do not correspond to the reality of the process; structural plant-model mismatch, when the structure of the model is not perfectly known; and the third type, process disturbances.

The most intuitive strategy to combat the uncertainty is to use process measurements to update the model. This is the main idea behind the “two-stage” approach (Chen & Joseph, 1987), (Darby, et al., 2011). Here,

deviations between predicted and measured outputs are used to update the model parameters, and new inputs are computed on the basis of the updated model. This iterative process is repeated until convergence is reached. However, the “two-stage” scheme works well only when the plant-model mismatch is parametric and the operating conditions provide sufficient excitation to efficiently estimate the model parameters.

This difficulty of converging to the plant optimum has motivated the development of a modified “two-stage” approach, known as Integrated System Optimization and Parameter Estimation (ISOPE) (Roberts, 1979) and (Roberts, 1995), that will be explained in the following subsection. ISOPE requires both output measurements and estimates of the gradients of the plant outputs with respect to inputs. These gradients will be used to modify the cost function of the optimization problem in order to enforce NCO matching between the model and the process.

MA has its origins in this technique, but differs in the definition of the modifiers and in the fact that no parameter estimation is required. MA uses measurements of the plant constraints, measurement of the objective function and estimates of the plant gradients to modify the cost and constraint functions in the model-based optimization problem without updating the model parameters (Marchetti, et al., 2009) to match the first-order NCO upon convergence. As will be explained later, the main advantage of MA is that it is able to reach plant optimality upon convergence despite the presence of structural plant-model mismatch.

MA has evolved from its beginnings in the seventies, with numerous schemes emerging, some of which will be described hereafter. Its implementation in several case studies, such as the Otto-Williams reactor (Marchetti, et al., 2010), (Navia, et al., 2013), or the operation of a depropanizer distillation column (Rodríguez-Blanco, et al., 2015), has shown that MA is a great tool for optimizing the process operation in the presence of a high degree of uncertainty. A review of the fundamentals of MA and the principle approaches developed in this field is made in this section, describing the main features and how to implement them.

2.1.1 Integrated System Optimization and Parameter Estimation (ISOPE)

To take into account the plant-model mismatch in RTO, a two-stage algorithm emerged called ISOPE. This technique adds a parameter to the

cost function of the economic optimization problem solved in the RTO layer that makes the model gradient match the process gradients in such a way that the KKT conditions (defined in equations (1.2)-(1.5)) of the modified problem match those of the real plant. The following steps, described in Algorithm 2.1, have to be followed to implement this methodology. For simplicity, an unconstrained economic optimization problem has been considered, although ISOPE has also been developed and implemented in the presence of operational constraints (Brdys, et al., 1986).

ALGORITHM 2.1: ISOPE (Roberts, 1979)

Step 1: Once the steady state of the process is reached, the k execution of the RTO is carried out. The inputs applied to the plant being \mathbf{u}_{k-1} , and measuring the output variables $\mathbf{y}_p(\mathbf{u}_{k-1})$, a parameter estimation problem is solved to update the uncertain model parameters $\boldsymbol{\beta}_k$, whose lower and upper limits are defined by $\boldsymbol{\beta}^l$ and $\boldsymbol{\beta}^u$, respectively. This problem (2.2) is solved under the condition that the outputs computed from the model match those measured from the process $\mathbf{y}(\mathbf{u}_{k-1}, \boldsymbol{\beta}_k) = \mathbf{y}_p(\mathbf{u}_{k-1}) \in \mathfrak{R}^{n_y}$, obtaining the optimal solution for the uncertain parameters $\boldsymbol{\beta}_k^*$, where $\mathbf{h} \in \mathfrak{R}^{n_h}$ are the set of equality constraints given by the model.

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & (\mathbf{y}_p - \mathbf{y})^T (\mathbf{y}_p - \mathbf{y}) \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{u}_{k-1}, \mathbf{y}(\mathbf{u}_{k-1}, \boldsymbol{\beta})) = 0 \\ & \boldsymbol{\beta}^l \leq \boldsymbol{\beta} \leq \boldsymbol{\beta}^u \end{aligned} \quad (2.2)$$

Step 2: Assuming that the output plant gradient $\nabla_{\mathbf{u}} \mathbf{y}_p|_{\mathbf{u}_{k-1}}$ is available, the first order modifier $\boldsymbol{\lambda}_k \in \mathfrak{R}^{n_u}$ is computed from steady state information of the process by applying (2.3).

$$\boldsymbol{\lambda}_k^T = \nabla_{\mathbf{y}} \phi|_{(\mathbf{u}_{k-1}, \mathbf{y}(\mathbf{u}_{k-1}, \boldsymbol{\beta}_k^*))} [\nabla_{\mathbf{u}} \mathbf{y}_p|_{\mathbf{u}_{k-1}} - \nabla_{\mathbf{u}} \mathbf{y}|_{(\mathbf{u}_{k-1}, \boldsymbol{\beta}_k^*)}] \quad (2.3)$$

Step 3: Once the model has been updated and the modifier has been computed, the modified economic optimization problem (2.4) of the RTO layer is solved, where ϕ_M is the modified cost function, in order to obtain the new inputs \mathbf{u}_k^* that will be applied to the process until the next RTO iteration, which will be carried out when the process reaches a new steady state.

$$\begin{aligned}
\min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \mathbf{y}(\mathbf{u}, \boldsymbol{\beta}_k^*)) + \boldsymbol{\lambda}_k^T \mathbf{u} \\
\text{s.t.} \quad & \mathbf{h}(\mathbf{u}, \mathbf{y}(\mathbf{u}, \boldsymbol{\beta}_k^*)) = 0 \\
& \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U
\end{aligned} \tag{2.4}$$

In this way, the gradient of the modified cost function will coincide with the plant gradient at \mathbf{u}_k^* , so that the qualification condition of the model described in the literature of the ISOPe method (Brdys & Roberts, 1987) is satisfied; i.e., at the optimum, the KKT conditions of the modified problem are equal to those of the process. These KKT conditions have already been defined in chapter one, equations (1.2)-(1.5) and (1.7).

Step 4 (optional): Smooth input changes and improve the convergence to the optimum by filtering the inputs obtained by (2.4) by means of a first order exponential filter (2.5), where $\mathbf{K} \in \mathfrak{R}^{n_u \times n_u}$ is a diagonal matrix of damping factors whose eigenvalues vary from 0 to 1. The filtered inputs are applied to the plant.

$$\mathbf{u}_k = \mathbf{u}_{k-1} + \mathbf{K}(\mathbf{u}_k^* - \mathbf{u}_{k-1}) \tag{2.5}$$

Step 5: Check that the convergence criterion is satisfied. This criterion can be defined as $|\mathbf{u}_k - \mathbf{u}_{k-1}| \leq \text{tol}$, i.e., there are no changes in the RTO decision variables higher than a fixed tolerance. If this is satisfied, stop the algorithm, otherwise, $k = k + 1$, and return to step 1 when the next steady state is reached.

Thus, by applying this methodology, each RTO execution corresponds to a steady state of the process.

2.1.2 Modifier Adaptation methodology

In 2002, Tatjewski proved that the convergence to the optimum of the ISOPe method does not depend on the estimation of the uncertain parameters, but on the equality between the process and model outputs in each RTO execution (Tatjewski, 2002). For this reason, he introduced a new modifier, $\mathbf{b}_k \in \mathfrak{R}^{n_y}$, that takes into account the difference between the process and model outputs ensuring that, upon convergence, these values will be the same. The resultant optimization problem solved in the RTO layer is described by (2.6), where ϕ_M is the modified cost function, \mathbf{u}_{k-1}^* is the input applied in the previous steady state, i.e., the optimal solution of the

previous RTO, \mathbf{h} represents the set of equality constraints imposed by the model, and the subscript P indicates that the cost function gradient and the output variables are evaluated from process measurements.

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{y} + \mathbf{b}_k, \mathbf{u}, \boldsymbol{\beta}) + \boldsymbol{\lambda}_k^T \mathbf{u} \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{y} + \mathbf{b}_k, \mathbf{u}, \boldsymbol{\beta}) = 0 \\ & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \end{aligned} \quad (2.6)$$

where

$$\boldsymbol{\lambda}_k^T = \nabla_{\mathbf{u}} \phi_P|_{\mathbf{u}_{k-1}^*} - \nabla_{\mathbf{u}} \phi|_{\mathbf{u}_{k-1}^*} \quad \mathbf{b}_k = \mathbf{y}_P(\mathbf{u}_{k-1}^*, \boldsymbol{\beta}) - \mathbf{y}(\mathbf{u}_{k-1}^*, \boldsymbol{\beta}) \quad (2.7)$$

One extension of this method was developed by Gao and Engell, who defined new modifiers for the process constraints (Gao & Engell., 2005). The main contribution of this work is the way they correct the process constraints. In addition to the bias corrector added to the constraints ε_k that results from the difference between the value of the process and model constraints (Forbes & Marlin, 1994), they add correction terms to the constraint gradients. These modifiers, $\mathbf{v}_{i,k} \in \mathfrak{R}^{n_{\text{u}} \times n_{\text{g}}}$, are computed as the difference between the constraint gradients estimated from the model and those from the real plant. In this way, upon convergence, the finding of an operating point that satisfies the NCO of the process is guaranteed. In this case, $g_{M,i}$ are the modified constraints and $g_{P,i}$ are the process constraints.

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{y} + \mathbf{b}_k, \mathbf{u}, \boldsymbol{\beta}) + \boldsymbol{\lambda}_k^T \mathbf{u} \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{y} + \mathbf{b}_k, \mathbf{u}, \boldsymbol{\beta}) = 0 \\ & g_{M,i} = g_i(\mathbf{u}, \boldsymbol{\beta}) + \mathbf{v}_{i,k}^T (\mathbf{u} - \mathbf{u}_{k-1}^*) + \varepsilon_{i,k} \leq 0 \quad i = 1, \dots, n_g \\ & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \end{aligned} \quad (2.8)$$

where

$$\begin{aligned} \boldsymbol{\lambda}_k^T &= \nabla_{\mathbf{u}} \phi_P|_{\mathbf{u}_{k-1}^*} - \nabla_{\mathbf{u}} \phi|_{\mathbf{u}_{k-1}^*} \quad \mathbf{v}_{i,k}^T = \nabla_{\mathbf{u}} g_{P,i}|_{\mathbf{u}_{k-1}^*} - \nabla_{\mathbf{u}} g_i|_{\mathbf{u}_{k-1}^*} \\ \varepsilon_{i,k} &= g_{P,i}(\mathbf{u}_{k-1}^*, \boldsymbol{\beta}) - g_i(\mathbf{u}_{k-1}^*, \boldsymbol{\beta}) \quad \mathbf{b}_k = \mathbf{y}_P(\mathbf{u}_{k-1}^*, \boldsymbol{\beta}) - \mathbf{y}(\mathbf{u}_{k-1}^*, \boldsymbol{\beta}) \end{aligned} \quad (2.9)$$

The MA methodology was developed from the equality of the KKT conditions (defined in equations (1.2)-(1.5)) of both the process and the model. This technique was presented by Chachuat in 2009 (Chachuat, et al., 2009) and formalized by Marchetti (Marchetti, et al., 2009). The authors removed the modifier \mathbf{b}_k , with the justification that by using the rest of the

modifiers, upon convergence, the KKT conditions of the modified optimization problem would match those of the process. The problem (2.10) is iteratively solved by implementing the optimal inputs \mathbf{u}_k^* to the process and updating the modifiers when a steady state has been reached, as shown in Figure 2.1:

$$\begin{aligned}
\min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \boldsymbol{\beta}) + \boldsymbol{\lambda}_k^T (\mathbf{u} - \mathbf{u}_{k-1}^*) \\
\text{s.t.} \quad & g_{M,i} = g_i(\mathbf{u}, \boldsymbol{\beta}) + \boldsymbol{\nu}_{i,k}^T (\mathbf{u} - \mathbf{u}_{k-1}^*) + \varepsilon_{i,k} \leq 0 \\
& \mathbf{h}(\mathbf{u}, \boldsymbol{\beta}) = 0 \\
& \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \\
& i = 1, \dots, n_g
\end{aligned} \tag{2.10}$$

where the modifiers are computed as shown in (2.9). In this formulation, the number of modifiers (n_{mod}) is given by (2.11), where it can be observed that it depends on both the number of decision variables (n_u) and the number of constraints (n_g):

$$n_{mod} = n_g + n_u(n_g + 1) \tag{2.11}$$

By using these modifiers, if $(\mathbf{u}^*, \boldsymbol{\mu}^*)$ is a KKT point of the modified problem it will be a KKT point of the process as it can be seen in the following expressions. Under convergence $\mathbf{u}^* = \mathbf{u}_k^* = \mathbf{u}_{k-1}^*$ and therefore $\mathbf{u}_k^* - \mathbf{u}_{k-1}^* = 0$:

$$\frac{\partial L_M}{\partial \mathbf{u}^*} = \frac{\partial \phi_M}{\partial \mathbf{u}^*} + \boldsymbol{\mu}^{*T} \frac{\partial \mathbf{g}_M}{\partial \mathbf{u}^*} = \frac{\partial \phi}{\partial \mathbf{u}^*} + \boldsymbol{\lambda}^T + \boldsymbol{\mu}^{*T} \left(\frac{\partial \mathbf{g}}{\partial \mathbf{u}^*} + \boldsymbol{\nu}^T \right) = 0 \tag{2.12}$$

$$\boldsymbol{\mu}^{*T} \mathbf{g}_M = \boldsymbol{\mu}^{*T} (\mathbf{g} + \boldsymbol{\varepsilon}) = 0, \quad \boldsymbol{\mu}^{*T} \geq 0 \tag{2.13}$$

$$\mathbf{g}_M = \mathbf{g} + \boldsymbol{\varepsilon} \leq 0 \tag{2.14}$$

Replacing the first- and zeroth-order modifiers given by (2.9) in equations (2.12), (2.13) and (2.14):

$$\frac{\partial L_M}{\partial \mathbf{u}^*} = \frac{\partial \phi}{\partial \mathbf{u}^*} + \left(\frac{\partial \phi_p}{\partial \mathbf{u}^*} - \frac{\partial \phi}{\partial \mathbf{u}^*} \right) + \boldsymbol{\mu}^T \left(\frac{\partial \mathbf{g}}{\partial \mathbf{u}^*} + \left(\frac{\partial \mathbf{g}_p}{\partial \mathbf{u}^*} - \frac{\partial \mathbf{g}}{\partial \mathbf{u}^*} \right) \right) = \frac{\partial L_p}{\partial \mathbf{u}^*} \tag{2.15}$$

$$\boldsymbol{\mu}^{*T} \mathbf{g}_M = \boldsymbol{\mu}^{*T} (\mathbf{g} + (\mathbf{g}_p - \mathbf{g})) = \boldsymbol{\mu}^{*T} \mathbf{g}_p, \quad \boldsymbol{\mu}^{*T} \geq 0 \tag{2.16}$$

$$\mathbf{g}_M = \mathbf{g} + (\mathbf{g}_p - \mathbf{g}) = \mathbf{g}_p \tag{2.17}$$

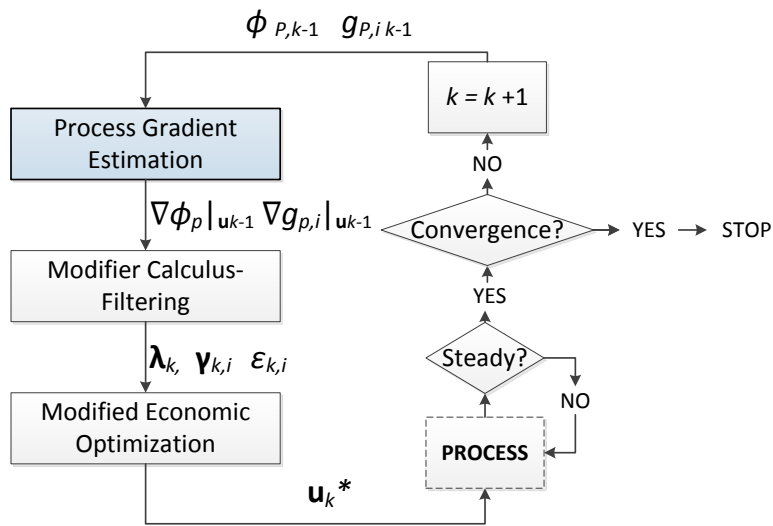


Figure 2.1. General formulation of MA methodology.

A graphical interpretation of the modifiers for the constraint g_i is shown in Figure 2.2. The modifier $\epsilon_{i,k}$ corresponds to the gap between the plant $g_{p,i}$ and the predicted constraint values g_i at u_k , whereas $\gamma_{i,k}$ represents the difference between the slopes of $g_{p,i}$ and g_i at u_k .

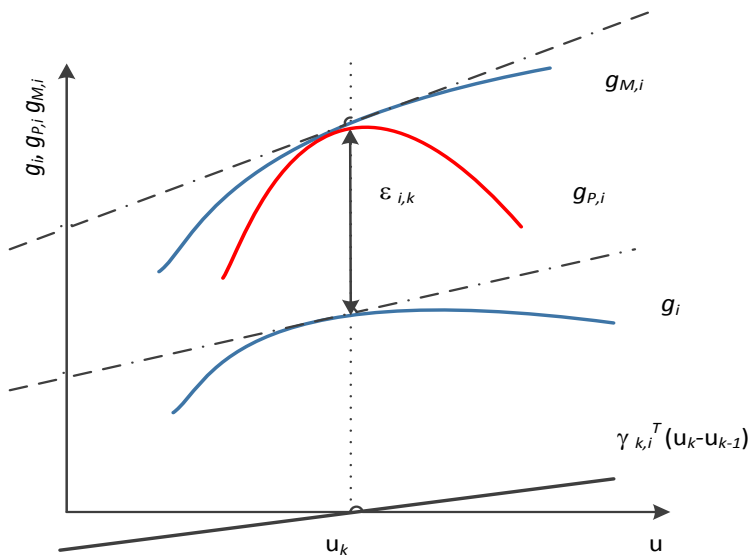


Figure 2.2. Modification of the constraint g_i at u_k .

This algorithm has the advantage that it is not necessary to know a priori the set of active constraints, adding as many $\mathbf{v}_{i,k}$ modifiers as operational constraints, at the expense of slowing down the convergence of the method. Furthermore, the method achieves the process optimum, even in the presence of structural uncertainty. The main disadvantage is that, although the method converges to an operating point that satisfies the NCO of the process, it may do so following an infeasible way, since there is no guarantee of feasibility in the intermediate iterations. Therein lies the importance of using first order exponential filters for the modifiers, as those shown in (2.18), to smooth changes and to try to follow a feasible path to the optimum.

$$\begin{aligned}\boldsymbol{\lambda}_k &= (\mathbf{I} - \mathbf{K}_\lambda)\boldsymbol{\lambda}_{k-1} + \mathbf{K}_\lambda \left(\nabla_{\mathbf{u}} \phi_P \Big|_{\mathbf{u}_{k-1}^*} - \nabla_{\mathbf{u}} \phi \Big|_{\mathbf{u}_{k-1}^*} \right) \\ \mathbf{v}_{i,k} &= (\mathbf{I} - \mathbf{K}_v)\mathbf{v}_{i,k-1} + \mathbf{K}_v \left(\nabla_{\mathbf{u}} g_P \Big|_{\mathbf{u}_{k-1}^*} - \nabla_{\mathbf{u}} g \Big|_{\mathbf{u}_{k-1}^*} \right) \\ \boldsymbol{\varepsilon}_k &= (\mathbf{I} - \mathbf{K}_\varepsilon)\boldsymbol{\varepsilon}_{k-1} + \mathbf{K}_\varepsilon \left(g_{P,i}(\mathbf{u}_{k-1}^*) - g_i(\mathbf{u}_{k-1}^*) \right)\end{aligned}\tag{2.18}$$

where \mathbf{K}_λ , \mathbf{K}_v y \mathbf{K}_ε , are diagonal matrixes whose eigenvalues are within the interval (0,1].

The model used for RTO must satisfy certain adequacy conditions for MA, which guarantees that the real optimum \mathbf{u}_∞ can be found. These conditions are the following.

First, a process model is said to be adequate for use in an RTO scheme if it is capable of producing a local minimum for the RTO problem and a fixed point for the RTO algorithm at the plant optimum \mathbf{u}_∞ . In other words, \mathbf{u}_∞ must be a local minimum of the RTO problem when the RTO algorithm is applied at \mathbf{u}_∞ . The plant optimum \mathbf{u}_∞ is an operating point that satisfies the first-and second-order NCO of the plant.

The adequacy criterion requires that \mathbf{u}_∞ must also satisfy the first- and second-order NCO for the modified optimization (2.10), with the modifiers (2.18) evaluated at \mathbf{u}_∞ . As MA matches the first-order KKT elements of the plant (defined in equations (1.2)-(1.5)), only the second-order NCO remains to be satisfied. That is, the reduced Hessian of the Lagrangian $\nabla^2_r L_M(\mathbf{u}_\infty)$ must be positive definite at \mathbf{u}_∞ (1.7), then the process model is adequate for use in the MA scheme (Marchetti, et al., 2016). This positive-definiteness requirement is independent of the modifier values themselves.

In general, the MA can be implemented by following the next steps:

ALGORITHM 2.2: Modifier Adaptation (Marchetti, et al., 2009)

Step 1. Initialize $k = 0$ and the initial value for the modifiers λ_0, γ_0 and ϵ_0 .

Step 2: Once the process has achieved the steady state, the process cost function $\phi_{P,k-1}$ and the constraints $g_{P,i,k-1}$ are measured. Compute the zeroth order modifier $\epsilon_{i,k}$ from (2.9) for each constraint.

Step 3: Estimate the process gradients required to compute the first order modifiers. This step will be different in each of the described methodologies since each one considers a different estimation method, as will be seen later in this chapter.

Step 4 (optional): Filter the modifiers given by the upper optimization layer using (2.18).

Step 5: Compute the new decision variable \mathbf{u}_k^* that will be applied to the process until the next steady state by solving the modified optimization problem (2.10).

Step 6: Check if the convergence criterion is satisfied, for instance, $|\mathbf{u}_k^* - \mathbf{u}_{k-1}^*| \leq \mathbf{tol}$, that is, there are no changes in the RTO decision variables

higher than a fixed tolerance. Another stop criteria can be considered, for example, there are no changes in the process cost function higher than a specified value $|\phi_{P,k}^* - \phi_{P,k-1}^*| \leq \mathbf{tol}$. If any of these criteria is satisfied, stop the algorithm, otherwise, $k = k + 1$, and return to step 2 when the next steady state is reached.

As can be seen in Figure 2.1, this methodology requires the computation of model and experimental gradients. Whereas there are tools to compute the model gradients accurately, the plant gradients are more difficult to estimate. Because of the necessity of estimating the process gradients, several approaches have emerged to compute them properly, implemented in step 3, which are combined with the general MA methodology described by the Algorithm 2.2. The most intuitive method consists in estimating the plant gradients by forcing an increase in each manipulated variable, waiting for the steady state and computing each gradient by finite differences. However, this requires a high number of intermediate experiments to estimate all the gradients (n_u experiments) before each RTO execution, which makes this type of estimation impractical.

2.1.3 Dual Modifier Adaptation

The Dual Modifier Adaptation method (DMA), initially proposed by Brdyś and Tatjewski (Brdyś & Tatjewski, 1994), (Brdyś & Tatjewski, 2005) and named Dual ISOPE, is an MA approach where an extra constraint is added to the RTO problem (dual constraint) to guarantee that the actions applied in the past will have generated enough excitation to accurately estimate the plant gradients. These experimental gradients are estimated from past operating points generated by the previous RTO iterations using the definition of directional derivative. Assuming that there is a collection of n_u+1 points $\mathbf{u}_k, \mathbf{u}_{k-1}, \dots, \mathbf{u}_{k-n_u}$ applied in the past RTO iterations, the vectors of differences with respect to previous points $\mathbf{s}_{k,i}$ are defined by (2.19).

$$\mathbf{s}_{k,i} = \mathbf{u}_k - \mathbf{u}_{k-i} \quad \forall i = 1, \dots, n_u \quad (2.19)$$

Supposing that the vectors $\mathbf{s}_{k,i}$ are linearly independent, it is possible to formulate a nonsingular square matrix $\mathbf{S}_k \in \mathbb{R}(n_u \times n_u)$ whose condition number will give the degree of excitation of the process:

$$\mathbf{S}_k = [\mathbf{s}_{k,1} \quad \dots \quad \mathbf{s}_{k,n_u}]^T \quad (2.20)$$

The process gradients are obtained from the definition of directional derivative as shown in (2.21), where the variable z would represent either the process cost function ϕ_P or the constraints $g_{p,i}$.

$$\frac{\partial z}{\partial \mathbf{u}} \approx \frac{dz}{d\mathbf{u}} = (\mathbf{S}_k)^{-1} \begin{bmatrix} z(\mathbf{u}_k) - z(\mathbf{u}_{k-1}) \\ \vdots \\ z(\mathbf{u}_k) - z(\mathbf{u}_{k-n_u}) \end{bmatrix} \quad (2.21)$$

To ensure that gradients are obtained accurately, a new constraint ($\kappa^{-1}(\mathbf{S}_k) \geq \alpha$) is added to the optimization problem, where κ represents the condition number ($\kappa(\mathbf{S}_k) = \|\mathbf{S}_k^{-1}\| \|\mathbf{S}_k\|$ for any consistent norm) of \mathbf{S}_k , which is the matrix formed by the vectors of the differences of the decision variables with respect to n_u samples before, as shown in (2.20), and the parameter α indicates the minimum degree of excitation (Gao, et al., 2016). This constraint represents the dual characteristic of the method: while the rest of the optimization aims to converge to the optimum of the modified model (primal objective), the dual constraint ensures that, in the next RTO iteration, the system will have enough excitation to estimate the process gradient adequately (dual objective). The introduction of this constraint

reduces the feasible region, which may imply a loss of optimality with respect to solving the original modified optimization problem (2.10), but increases the quality of the process gradient estimation. Following the described methodology, the modified optimization problem to be solved in the RTO layer is given by (2.22).

$$\begin{aligned}
 \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \boldsymbol{\beta}) + \boldsymbol{\lambda}_k^T (\mathbf{u} - \mathbf{u}_{k-1}^*) \\
 \text{s.t.} \quad & g_{M,i} = g_i(\mathbf{u}, \boldsymbol{\beta}) + \mathbf{v}_{i,k} (\mathbf{u} - \mathbf{u}_{k-1}^*) + \varepsilon_{i,k} \leq 0 \\
 & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \\
 & i = 1, \dots, n_g \\
 & \kappa^{-1}(\mathbf{S}_k) \geq a
 \end{aligned} \tag{2.22}$$

The implementation of DMA is carried out following Algorithm 2.2, in which steps 1, 3 and 5 have been replaced by the ones described below in Algorithm 2.3:

ALGORITHM 2.3: Dual Modifier Adaptation (Brdyś & Tatjewski, 2005)

Step 1: Execute $n_u + 1$ steady states to acquire enough information to initialize the computation of plant gradients and, therefore, the initial value for the modifiers $\boldsymbol{\lambda}_0$, $\mathbf{v}_{i,0}$ and $\boldsymbol{\varepsilon}_0$. Initialize $k = 0$.

Step 3: Estimate the plant gradients from (2.21).

Step 5: Choose the tuning parameter a and compute the new decision variables \mathbf{u}_k^* by solving the modified optimization problem (2.22). These inputs will be applied to the process until the next steady state.

The main advantage of this algorithm is that the process gradients are computed from information taken in the previous RTO executions, so extra excitation is not required, as in the case of estimating the gradients by using finite differences. However, the evolution of the algorithm is very sensitive to the value of parameter a , whose optimal value is not easily known a priori. This tuning parameter must be carefully chosen since a high value implies a high process excitation and this may drive the process to an operating point that satisfies the NCO of the process, but following an infeasible way. On the other hand, a small value may cause the gradients to be inaccurately estimated.

2.1.4 Iterative gradient-modification optimization combined with derivative free optimization

One of the major challenges of MA in practice is the estimation of the plant gradients from noisy measurement data. The iterative gradient-modification method (IGMO) explores the inherent smoothness of the plant mapping to enable an efficient and reliable optimization, even if the data is subject to a high level of noise. The idea behind this approach is to combine the quadratic approximation approach used in derivative-free optimization techniques with the iterative gradient-modification optimization scheme.

The IGMO scheme computes plant gradients based on the data collected at the previous operating points, in the same way as DMA does from the expression (2.21). In this method, the dual constraint defined in (2.22) was used to decide whether to perturb the process additionally. The perturbation \mathbf{u}^{addi} is optimized by maximizing the inverse of the condition number of \mathbf{S}_k^a subject to the adapted process constraints.

$$\begin{aligned} \max_{\mathbf{u}^{addi}} \quad & \kappa^{-1}(\mathbf{S}_k^a) \\ \text{s.t.} \quad & g_{M,i}(\mathbf{u}^{addi}) = g_i(\mathbf{u}^{addi}) + \mathbf{v}_{i,k}(\mathbf{u}^{addi} - \mathbf{u}_{k-1}) + \varepsilon_{i,k} \leq 0 \end{aligned} \quad (2.23)$$

where

$$\mathbf{S}_k^a = [(\mathbf{u}_k - \mathbf{u}^{addi}) \dots (\mathbf{u}_k - \mathbf{u}_{k-nu+1})]^\top \quad (2.24)$$

To provide reliable convergence in the presence of measurement noise, IGMO extracts the gradients from quadratic approximations. In this way, this scheme is complemented by the following elements (Gao, et al., 2016):

- Selecting points from the collected data set for a well-poised regression set.
- Introducing a constrained search space for the next move based on a covariance analysis of the regression set.
- Tracking the prediction accuracies of the adapted model-based mappings ($\phi_{ad}(\mathbf{u})$ and $g_{ad}(\mathbf{u})$) and the approximating quadratic functions, and also switching between model-based and data-based optimizations according to the observed accuracies.

How to choose the regression set and the evaluation of the search space are discussed in detail in (Gao, et al., 2016).

Let U_k and B_k represent the regression set and the search space at the k iteration. $\rho_{m,k}$ and $\rho_{q,k}$ are the prediction accuracies of the adapted model-based mappings and the quadratic functions (see (Gao, et al., 2016) to know how to compute these values). ϕ_q and $g_{q,i}$ are the regressed objective and constraint functions from the quadratic approximation. The implementation of this method is carried out by following the steps set out below:

ALGORITHM 2.4: Iterative gradient-modification optimization (Gao, et al., 2016)

Step 1: Choose an initial operating point \mathbf{u}_0 and apply that point and $\mathbf{u}_0 + c \mathbf{e}_i$ to the plant, where c is a suitable step size and $\mathbf{e}_i \in \mathcal{R}^{n_u}$ ($i=1, \dots, n_u$) are mutually orthogonal unit vectors. Compute the gradients at \mathbf{u}_0 by applying the finite difference approach and run IGMO until $(n_u + 1)(n_u + 2)/2$ set points have been generated. Run the screening algorithm (see (Gao, et al., 2016)) to obtain the regression set U_k . Initialize $\rho_{m,k} = \rho_{q,k} = 0$.

Step 2: Compute the quadratic functions ϕ_q and $g_{q,i}$ and determine the search space B_k .

Step 3: Extract the gradients from the quadratic functions. Adapt the model-based optimization problem and determine $\hat{\mathbf{u}}_k$ as follows:

- a) If $\rho_{m,k} \leq \rho_{q,k}$, run the adapted model-based optimization (2.10) under the constraint $\mathbf{u} \in B_k$.
- b) Otherwise, perform an optimization based on the quadratic approximation.

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi_{q,k}(\mathbf{u}) \\ \text{s.t.} \quad & g_{q,i,k}(\mathbf{u}) \leq 0 \\ & \mathbf{u} \in B_k \end{aligned} \quad (2.25)$$

Step 4: If $\|\hat{\mathbf{u}}_k - \mathbf{u}_k\| \leq \Delta \mathbf{u}$, where $\Delta \mathbf{u}$ is the parameter used by the screening algorithm to handle the influence of measurement noise, and there is at least one point $\mathbf{u}_j \in U_k$ such that $\|\mathbf{u}_j - \mathbf{u}_k\| > 2\Delta \mathbf{u}$, set $\hat{\mathbf{u}}_k = (\mathbf{u}_j + \mathbf{u}_k)/2$. This step is used to improve the quadratic approximation.

Step 5: Evaluate the process at \mathbf{u}_k to obtain ϕ_p and $g_{p,i}$. The next iteration

is defined depending on the following conditions:

- a) Successful iteration. If $\hat{\phi}_{p,k} < \phi_{p,k}$, where $\hat{\phi}_{p,k} = \phi_p(\hat{\mathbf{u}}_k)$, define $\hat{\mathbf{u}}_{k+1} = \mathbf{u}_k$ and run the screening algorithm to define the next regression U_{k+1} .
- b) Update the quality indices $\rho_{m,k+1}$ and $\rho_{q,k+1}$. Check the termination criterion, if it is not satisfied, set $k = k+1$ and go to step 2.

Unsuccessful iteration. If $\hat{\phi}_{p,k} \geq \phi_{p,k}$, run the screening algorithm to update the regression set for \mathbf{u}_k with the value $\hat{\mathbf{u}}_k$ in the collected data. Due to the screening algorithm, $\hat{\mathbf{u}}_k$ will be included in the regression set to achieve an improved quadratic approximation around $\hat{\mathbf{u}}_k$. Return to step 2.

Compared to the finite-difference calculation of the plant gradients, the quadratic approximation method decreases the influence of noise by capturing the curvature information from more distant points to provide a smooth mapping. These points are chosen by the screening algorithm based on the collected data. However, the previously collected data might be invalid in the presence of varying disturbances. This issue was discussed by Wenzel (Wenzel, et al., 2015), studying the effect of considering only recently collected data in the regression set for quadratic approximations.

2.1.5 Nested Modifier Adaptation

Although the MA techniques based on the computation of experimental gradients guarantee an operating point that satisfies the process NCO, regardless of the method used to estimate the gradients, the fact is that, it might be a problem in real applications; either because it is expensive, for instance, in the case of interconnected processes where the disturbances needed for the computation affect other units, or because this estimation is not accurate enough, as happens in the presence of measurement noise. To avoid the calculation of gradients, a new formulation was developed, Nested Modifier Adaptation (NMA) (Navia, et al., 2015). This method uses nested optimization architecture with a gradient-free optimization algorithm, for example, the Nelder-Mead algorithm, to directly update the modifiers, using them to iterate over the modified optimization until the process optimum is found. In this way, the process gradient estimation is replaced by another method that takes into account the minimization of the Lagrangian function measured directly from the process. The steps that have to be followed to implement NMA are shown in Algorithm 2.2, replacing step 3 with the one described below in Algorithm 2.5. In this formulation, it has been supposed

that the gradient-free algorithm implemented in the upper layer is the Nelder-Mead, but other direct search methods can be considered. This algorithm has been chosen because it is particularly parsimonious in function evaluations per iteration. This is because, in practice, it typically requires only one or two function evaluations to create a new iteration, whereas other direct search methods require n_u or more function evaluations to build a new simplex (Griffiths & Watson, 1995). This property is very important, as each function evaluation implies a change in the operating point of the plant (Navia, et al., 2015). The Nelder-Mead algorithm tries to find the process optimum by exploring the cost surface by means of a geometric figure with $n_\gamma + n_\lambda + 1$ vertices, the simplex, where $n_\gamma + n_\lambda$ is the number of first-order modifiers. Each vertex corresponds to a set of values of the decision variables and is associated with its corresponding value of the measured cost function. Then, with four basic operations: reflection, expansion, contraction and shrinking, the algorithm iterates with the set of decision variables, looking for the optimum, as shown in Figure 2.3 for the case of two decision variables, where p_{min} is the point with the lowest cost function, p_{max} is the point with the highest cost function, p_r is the reflected point, p_e is the expanded point and p_c is the contracted point.

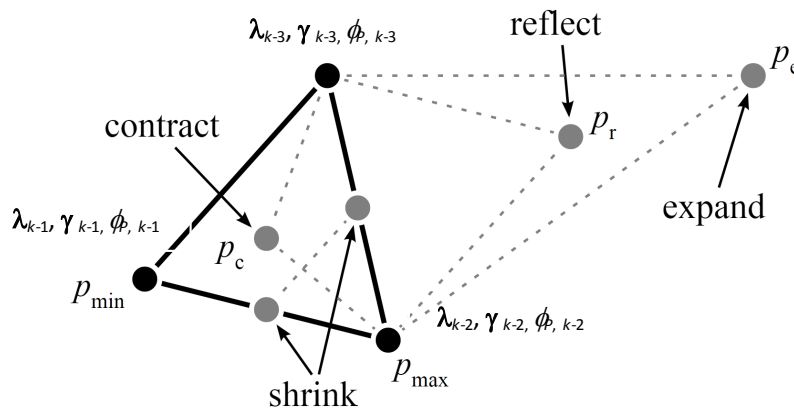


Figure 2.3. Nelder-Mead optimization.

The implementation of NMA is graphically shown in Figure 2.4.

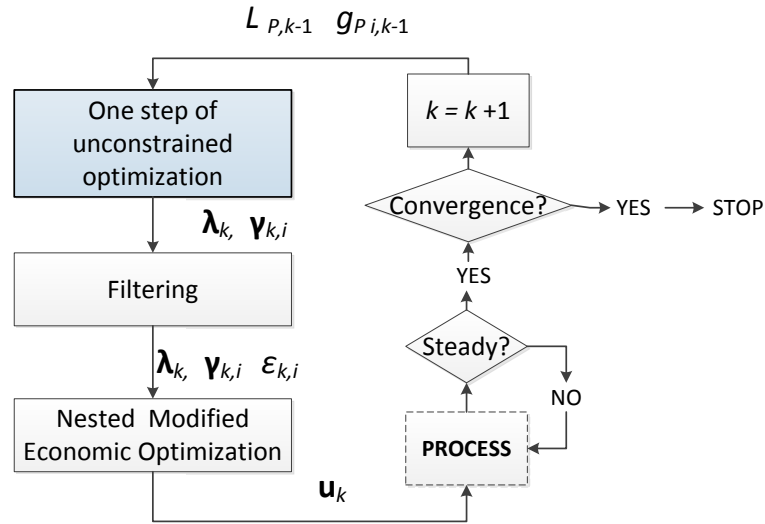


Figure 2.4. Implementation of Nested Modifier Adaptation.

ALGORITHM 2.5: Nested Modifier Adaptation (Navia, et al., 2015)

Add to Step 1: The initialization of the Nelder-Mead algorithm requires information about $n_\gamma + n_\lambda + 1$ previous steady states to construct the initial simplex, whose vertices correspond to different values for the first order modifiers obtained from the initial λ_0, \mathbf{v}_0 .

Step 3: The upper unconstrained optimization updates the value of the first order modifiers $\lambda_k, \mathbf{v}_{k,i}$ with the objective of minimizing the measured Lagrangian by solving (2.26), as its stationary point corresponds to the KKT conditions of the process. The corresponding multiplier $\boldsymbol{\mu}_{k-1}^T$ associated with the modified optimization problem solved in the previous RTO.

$$\min_{\lambda_k, \mathbf{v}_k} L_{P,k-1} = \phi_{P,k-1} + \boldsymbol{\mu}_{k-1}^T \mathbf{g}_{P,k-1} \quad (2.26)$$

The main advantage of NMA is that it does not require the calculation of experimental gradients that can be quite complicated in practice; in addition, it is less sensitive to noise than other methods such as DMA, because, as has already been said, it does not need the computation of gradients. However, this algorithm needs a good initial estimation of the modifiers; if it does not have this, then the way to the optimum might be infeasible and not very direct. Furthermore, the use of Nelder-Mead in the

upper optimization layer presents certain drawbacks, such as the proper choice of many tuning parameters that may affect the convergence to the optimum, or the fact that each iteration can require several steady-states. In addition, the unconstrained optimization layer works well if the uncertainty remains constant, which is difficult to guarantee if it takes several steady-states to converge, if not, the optimal modifiers will not be found. Finally, when there are a high number of modifiers due to the presence of many decision variables or operational constraints, the convergence to the optimum may be very slow, since the number of decision variables in the upper optimization layer increases considerably and its convergence is slowed down. To solve this problem, a new NMA formulation for dealing with a high number of operational constraints, described in chapter three, has been developed in this thesis. Since NMA measures the process Lagrangian, whose value incorporates the measurement of the plant constraints, this alternative approach consists of using the NMA methodology but reducing the number of required modifiers since both, cost function and constraints are taking into account in the measurement of the Lagrangian. In this way, it is necessary only to add one modifier for each process input in the cost function, independently of the number of constraints. These modifiers are obtained by the upper optimization layer with the objective of minimizing the value of the process Lagrangian function.

2.1.6 Directional Modifier Adaptation

One of the main disadvantages of the previously described methods is related to the problem size, since the number of required modifiers depends on both the number of decision variables and the number of operational constraints, as can be seen in (2.11).

The estimation of experimental gradients is the most difficult task in the application of MA. This experimental work increases with the number of decision variables, with the implementation of MA becoming impossible in processes with many inputs. For instance, applying DMA, a high number of decision variables implies a great number of previous steady states to initialize the algorithm, making the convergence rate of the method slower. In addition, all the changes applied to the process inputs in previous sample times must be linearly independent, that is, all the possible directions in the process inputs space must be explored so DMA cannot drive the process directly to the optimum but will converge slowly exciting all the process

inputs. A new formulation called Directional MA (D-MA) was developed to solve this problem. This method obtains the plant derivatives only in certain privileged directions.

D-MA has been developed for dealing with problems that present many input variables. This methodology overcomes the limitation of standard MA in these cases by estimating the experimental gradients only in $n_r < n_u$ privileged directions. In this way, the convergence to an operating point that satisfies the NCO of the process is speeded up, since fewer experiments are required to estimate the plant gradients in each RTO iteration. D-MA defines $\mathbf{U}_r = [\delta\mathbf{u}_1 \dots \delta\mathbf{u}_r]$ as a matrix ($n_u \times n_r$) whose columns contain the n_r privileged directions in the input space, in which the directional derivative is estimated for the cost function and the constraints at the operating point \mathbf{u}_k [38]. The directional derivative is considered a partial gradient that contains information about how a function varies (locally) in certain directions r of the input space and is given by (2.27).

$$\nabla_{\mathbf{u}_r} j_p := \left. \frac{\partial j_p(\mathbf{u}_k + \mathbf{U}_r \mathbf{r})}{\partial \mathbf{r}} \right|_{\mathbf{r}=0} \quad j_p \in \{\phi_p, g_{p,1}, g_{p,2}, \dots, g_{p,n_g}\} \quad (2.27)$$

where $r \in \mathfrak{R}^{n_r}$. The plant gradients are approximated by the following expressions (2.28):

$$\begin{aligned} \nabla \phi_p(\mathbf{u}_k) &\approx \frac{\partial \phi}{\partial \mathbf{u}}(\mathbf{u}_k) (\mathbf{I}_{n_u} - \mathbf{U}_r \mathbf{U}_r^+) + \nabla_{\mathbf{u}_r} \phi_p \mathbf{U}_r^+ \\ \nabla g_{p,i}(\mathbf{u}_k) &\approx \frac{\partial g_i}{\partial \mathbf{u}}(\mathbf{u}_k) (\mathbf{I}_{n_u} - \mathbf{U}_r \mathbf{U}_r^+) + \nabla_{\mathbf{u}_r} g_{p,i} \mathbf{U}_r^+ \\ i &= 1 \dots n_g \end{aligned} \quad (2.28)$$

where the superscript $(.)^+$ represents the Moore-Penrose pseudoinverse and \mathbf{I}_{n_u} is the identity matrix of size $n_u \times n_u$. By applying D-MA, the gradients of the process cost function and the constraints needed to estimate the modifiers by (2.9) are replaced by the approximations given by (2.28) for each value of \mathbf{u}_k .

Costello describes, in (Costello, et al., 2016), how to select the privileged directions for the case with parametric uncertainty. This choice is carried out by a sensitivity analysis of the gradient of the Lagrangian function with respect to the uncertain model parameters $\boldsymbol{\beta}$. The main idea is that if the parameter variations significantly affect the gradient of the Lagrangian function in only a few directions of the input space, it is enough to estimate the plant gradients in these directions.

D-MA is easy to implement and has important theoretical properties which allow a fast convergence rate and guarantee that the plant optimum is achieved.

The D-MA algorithm is described below in Algorithm 2.6. The methodology to be followed is the same as presented in Algorithm 2.2, adding new variables to initialize and replacing the computation of experimental gradients by the estimation of the directional derivatives:

ALGORITHM 2.6: Directional Modifier Adaptation (Costello, et al., 2016)

Add to Step 1: Choose a matrix of privileged directions in the input space \mathbf{U}_r , in which the plant gradients are estimated. To know how to obtain this matrix, see (Costello, et al., 2016). Therefore, to start the algorithm, it is necessary to collect the information about n_r previous operating points.

Step 3: Estimate the directional derivative of the process cost function $\nabla_{\mathbf{u}_r} \phi_p(\mathbf{u}_k)$ and the constraints $\nabla_{\mathbf{u}_r} g_{p,i}(\mathbf{u}_k)$ from (2.27) and then compute the plant gradients $\nabla \phi_p(\mathbf{u}_k)$ and $\nabla g_{p,i}(\mathbf{u}_k)$ from (2.28). These derivatives must be estimated using the measurements collected in at least n_r successive operating points near to \mathbf{u}_k .

The main advantage of this method is that if the number of selected privileged directions is much smaller than the number of decision variables, the task of estimating experimental gradients is drastically reduced; however, there is the drawback of choosing these directions properly.

2.1.7 Summary of MA applications in literature

MA has many useful features and a high potential that make it very interesting for investigation and application. The methods previously described have been successfully implemented in numerous applications in realistic simulations, but few of them have reached the industrial stage. The Otto-Williams reactor (Roberts, 1979) has been the principal benchmark problem for testing MA approaches in the literature. This problem is quite challenging due to the presence of significant structural plant-model mismatch, as the plant is simulated as a 3-reaction system, whereas the model includes only two reactions with adjustable kinetic parameters. The different MA variants such as ISOPE (Roberts, 1979), DMA (Marchetti, et al., 2010) and NMA (Navia, et al., 2013), have been applied to this example and it will be used to test the different approaches developed in this thesis. All

these methods can achieve an operating point that satisfies the process NCO.

A more realistic application will also be used in this thesis that considers the operation of a depropanizer distillation column with a model predictive control layer (Rodríguez-Blanco, et al., 2015). This problem presents two types of uncertainty; the first is due to the use of an erroneous steady state model in the RTO layer, and the other originates from the presence of different kinds of models in the control structure (RTO is based on nonlinear steady state models, whereas MPC uses dynamic linear models). This case study can be considered as a large scale problem from the point of view of the model size, since the dynamic model used to represent the process very precisely is formed by 2152 variables (1976 explicit, 129 derivative, 40 algebraic and 7 boundaries) and 2145 equations (129 differential equations and 2023 algebraic equations). However, the RTO layer uses a simplified steady state model with only 32 algebraic equations and 39 variables (29 explicit, 3 algebraic and 7 boundaries), which supposes a great structural plant-model mismatch. This case study will be explained in more detail in chapter five, since the approaches developed during this thesis have been implemented on this example. For comparison, static methodologies like DMA and NMA have been implemented on this example (Rodríguez-Blanco, et al., 2015), observing that the optimum is achieved after 60 hours, which implies 10 steady states. To speed up the convergence to the plant optimum, new methods based on transient measurements developed during this thesis were implemented, (Rodríguez-Blanco, et al., 2016), (Rodríguez-Blanco, et al., 2017), achieving plant optimum in only one steady state, after approximately 8 hours of operation.

Additionally, D-MA was applied to the challenging problem of optimizing the flight path of a power-generating kite (Costello, et al., 2016). This problem deals with 40 decision variables, which makes the implementation of standard MA infeasible. In the presence of parametric and structural uncertainty, D-MA is able to find the optimal path, estimating the gradients in two directions ($n_r = 2$) after only 10 iterations. The performance comparison was made considering a greater number of directions ($n_r = 4$), concluding that the convergence is much slower in this case, since gradients are estimated in more directions than necessary.

Other interesting case studies where this methodology has been implemented are: semi-batch reactors (Jia, et al., 2016), batch chromatography (Gao & Engell., 2005), chromatographic separation

(Behrens, et al., 2014), sugar and ethanol plant (Serralunga, et al., 2013), solid-oxide fuel cell (Bunin, et al., 2012), leaching process (Zhang, et al., 2015), parallel compressors (Milosavljevic, et al., 2016) and experimental flotation columns (Navia, et al., 2016) .

This latter will be used to test the methodology experimentally, based on transient measurements developed in this thesis (Navia, et al., 2017). The results obtained and the process description are explained in chapter four.

2.2 Implementation issues

This chapter has summarized the main MA approaches that have been developed within this methodology. However, these techniques present some drawbacks for its implementation. One of them is the application of MA in problems with many operational constraints or many input variables. As has been shown in (2.11), the bigger the number of constraints, the bigger the number of modifiers that have to be estimated. A high number of modifiers require the computation of more experimental gradients, which requires more excitation to estimate them or more previous operating points to start the estimation, depending on the applied methodology. D-MA has been developed to solve problems with many inputs, but handling many operational constraints is still an open issue. To overcome this problem, chapter three describes a modification of the NMA approach to deal with operational constraints. This NMA approach is based on the minimization of the Lagrangian function that includes, in only one modifier, the correction in the cost function and the constraints. Therefore, this new method requires only one modifier to be added to the RTO cost function for each input, independently of the number of constraints.

Another issue is the convergence rate of MA. As has been noted, MA proceeds by iteratively adjusting the optimization problem with first and zeroth order corrections, calculated from steady state information at each RTO execution. This fact implies that each correction in the RTO has to wait until a new steady-state has been reached, which implies a long convergence time. To solve this problem, chapter four presents several MA approaches based on the computation of plant gradients from transient measurements to speed up the convergence to an operating point that satisfies the process NCO.

However, there are some remaining problems that have not been dealt with in this thesis. For instance, MA does not guarantee the feasibility of the

successive RTO iterates. So, an open issue is to guarantee the feasibility to the process optimum without slowing down the convergence rate.

The Sufficient Conditions for Feasibility and Optimality (SCFO) have been used, combined with RTO, to enforce plant feasibility (Bunin, et al., 2013). However, SCFO fails to enforce sufficiently fast convergence because of the necessity to upper bound uncertain plant constraints using Lipschitz constants. Feasibility in DMA has also been studied in (Navia, et al., 2012), where the authors propose the use of a PI controller that is only activated when the measurements show a violation in the constraints. The use of convex upper-bounding functions to guarantee feasibility has also been discussed in (Marchetti, et al., 2017). Despite the existence of numerous studies in this field, there is still a trade-off between the convergence rate and plant feasibility which means that guaranteeing feasibility in MA is still an open issue that requires further investigation.

One possibility for dealing with this problem is to apply MA in Dynamic RTO. In this way, path and terminal constraints are considered in the resolution of the optimization problem, so MA would ensure convergence following a feasible path. In addition, inconsistencies between control layers would be reduced, since both the MPC and the RTO would use dynamic models. The main difficulty of this approach would be to obtain the modifiers from the NCO formulated for dynamic optimization problems.

The robustness of MA to gradient uncertainty is another open issue, since the estimated gradients have inherent errors. The extent to which MA is robust for this type of uncertainty should therefore be defined. Several works in this field point out that MA is quite robust for this uncertainty (Marchetti, et al., 2009), (Marchetti, et al., 2010), but in reality, it has to be assumed that gradient uncertainty is additive and bounded.

Taking into account the important benefits that may be obtained by applying MA in the industry and the fact that there remain several open issues, MA has become an interesting topic on which further research should be done.

3 HANDLING OPERATIONAL CONSTRAINTS IN MODIFIER ADAPTATION

MA has been successfully implemented in several applications; however, the size of the problem with respect to the number of decision variables and the number of constraints is still an open issue, as it increases the number of modifiers and the time required to estimate them, slowing down convergence to the optimum. For dealing with process-dependent constraints without increasing the number of modifiers, an alternative MA methodology can be implemented. This consists in estimating the modifiers from information provided by the Lagrangian function, adding only one gradient modifier for each process input to the cost function. These modifiers are used in the NMA technique, which does not require the computation of experimental gradients, thus making its implementation in practice easier.

3.1 Introduction

Although the MA techniques presented in the previous chapters have been successfully implemented in several applications, the size of the problem with respect to the number of decision variables (n_u) and the number of constraints (n_g) is still an open issue, since it increases the number of modifiers (n_{mod}) and therefore the number of experimental gradients that must be estimated, as shown in (3.1), making the convergence of the MA to the real optimum slower.

$$n_{mod} = n_g + n_u(n_g + 1) \quad (3.1)$$

In particular, DMA requires $n_u + 1$ operating points to make the first estimation of the plant gradients and initialize the algorithm. Moreover, the greater the number of constraints, the larger the number of experimental gradients that have to be estimated, thus increasing the computational work. In the case of NMA, the influence of the problem size is more direct, as the main drawback of this methodology is that it requires $n_{mod} + 1$ previous operating points to construct the initial simplex and initialize the algorithm. In this way, the greater the number of constraints, the larger the number of required initial operating points and, therefore, the slower the convergence to the optimum, since the number of decision variables of the unconstrained optimization layer increases.

This chapter proposes an alternative MA methodology for dealing with process-dependent constraints without increasing the number of modifiers. The key feature consists in computing the modifiers from the gradient of the Lagrangian function, thereby requiring a single gradient modifier per process input. In this way, the first-order modifiers that are usually included in the inequality constraints are not necessary because these corrections are included in the expression for the gradient of the Lagrangian.

In the context of NMA, this feature represents a great advantage as it reduces the number of decision variables in the outer optimization that directly affect the convergence speed. To overcome this problem, this thesis presents a new NMA approach to deal with problems with many operational constraints so as to get a faster convergence and make the implementation of this technique easier in practice.

The performance of the proposed methodology is illustrated through a case study corresponding to the operation of the Otto-Williams reactor, in the presence of both parametric and structural plant-model mismatch.

In addition, another two case studies will be considered, as will be shown in chapters five and six: the operation of an industrial depropanizer distillation column of a petrol refinery and the transport of natural gas through gas pipeline networks.

3.2 Alternative MA using the Lagrangian gradients

In order to deal with process-dependent constraints, Brdyś proposed using a modifier for the gradient of the Lagrangian function in the context of the ISOPE method (Brdys, et al., 1986). An equivalent approach can be applied in MA. The following equation shows the alternative modified optimization problem, correcting the Lagrangian gradients, described by (3.2), which can be compared to the original problem given by (2.10):

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \boldsymbol{\beta}) + \boldsymbol{\lambda}_k^{L,T} \mathbf{u}_k \\ \text{s.t.} \quad & \mathbf{g}_M = \mathbf{g}(\mathbf{u}, \boldsymbol{\beta}) + \boldsymbol{\varepsilon}_k \leq 0 \\ & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \end{aligned} \quad (3.2)$$

where ϕ_M and $\mathbf{g}_M \in \mathfrak{R}^{\text{ng}}$ are the modified cost function and constraints, $\boldsymbol{\varepsilon}_k \in \mathfrak{R}^{\text{ng}}$ are the zero-order constraint modifiers, and $\boldsymbol{\lambda}_k^L \in \mathfrak{R}^{\text{nu}}$ is the Lagrangian gradient modifier which represents the difference between the Lagrangian gradients of the plant and the model, as shown in (3.3). The subscript “P” indicates that the variable is evaluated from the process measurements.

$$\begin{aligned} \boldsymbol{\lambda}_k^{L,T} &= \nabla_{\mathbf{u}} L_p \Big|_{\mathbf{u}_{k-1}, \boldsymbol{\mu}_{k-1}} - \nabla_{\mathbf{u}} L \Big|_{\mathbf{u}_{k-1}, \boldsymbol{\mu}_{k-1}} \\ \boldsymbol{\varepsilon}_k &= \mathbf{g}_p(\mathbf{u}_{k-1}) - \mathbf{g}(\mathbf{u}_{k-1}) \end{aligned} \quad (3.3)$$

where the plant and model Lagrangians, L_p and L , are computed as follows, and $\boldsymbol{\mu}_k \in \mathfrak{R}^{\text{ng}}$ are the Lagrange multipliers of the modified problem given by (3.2):

$$L_p = \phi_p + \boldsymbol{\mu}_k^T \mathbf{g}_p, \quad L = \phi + \boldsymbol{\mu}_k^T \mathbf{g} \quad (3.4)$$

If the MA algorithm converges, the Lagrangian gradient of the modified problem (3.2) matches the Lagrangian gradient for the plant, as can be seen next. The modifier $\boldsymbol{\lambda}_\infty^{L,T}$ is obtained from the equality of the first order NCO at $k = \infty$ of the process and the modified problem (3.2) as shown by equations (3.5) to (3.8), where L_M , defined by (3.5), is the Lagrangian of the

modified problem. $\mathbf{U}_\infty^T = (\mathbf{u}_\infty^*(\boldsymbol{\Lambda}_\infty), \boldsymbol{\mu}_\infty^*(\boldsymbol{\Lambda}_\infty))$ is the solution of the problem defined by (3.2) with modifiers $\boldsymbol{\Lambda}_\infty^T = [\boldsymbol{\lambda}_\infty^{L^T}]$, where $\boldsymbol{\mu}_\infty^*$ is the value of the Lagrange multipliers for \mathbf{g}_M and $\boldsymbol{\Lambda}_\infty^T$ is the value of the modifiers at this point:

$$L_M = \phi_M + \boldsymbol{\mu}_k^T \mathbf{g}_M \quad (3.5)$$

$$\nabla_{\mathbf{u}} L_p(\mathbf{u}_\infty, \boldsymbol{\mu}_\infty) = \nabla_{\mathbf{u}} L_M(\mathbf{u}_\infty, \boldsymbol{\mu}_\infty, \boldsymbol{\Lambda}_\infty) \quad (3.6)$$

$$\nabla_{\mathbf{u}} \phi(\mathbf{u}_\infty^*, \boldsymbol{\beta}) + \boldsymbol{\lambda}_\infty^{L^T} + \boldsymbol{\mu}_\infty^T \nabla_{\mathbf{u}} \mathbf{g}(\mathbf{u}_\infty^*, \boldsymbol{\beta}) = \nabla_{\mathbf{u}} \phi_p(\mathbf{u}_\infty^*) + \boldsymbol{\mu}_\infty^T \nabla_{\mathbf{u}} \mathbf{g}_p(\mathbf{u}_\infty^*) \quad (3.7)$$

$$\boldsymbol{\lambda}_\infty^{L^T} = \nabla_{\mathbf{u}} \phi_p(\mathbf{u}_\infty^*) - \nabla_{\mathbf{u}} \phi(\mathbf{u}_\infty^*, \boldsymbol{\beta}) + \boldsymbol{\mu}_\infty^T (\nabla_{\mathbf{u}} \mathbf{g}_p(\mathbf{u}_\infty^*) - \nabla_{\mathbf{u}} \mathbf{g}(\mathbf{u}_\infty^*, \boldsymbol{\beta})) = \nabla_{\mathbf{u}} L_p(\mathbf{u}_\infty, \boldsymbol{\mu}_\infty) - \nabla_{\mathbf{u}} L_M(\mathbf{u}_\infty, \boldsymbol{\mu}_\infty, \boldsymbol{\Lambda}_\infty) \quad (3.8)$$

If the scheme converges, then, under the following assumptions, it will do so to a KKT point of the plant. It is assumed that functions $\phi(\mathbf{u}, \boldsymbol{\beta})$ and $\mathbf{g}(\mathbf{u}, \boldsymbol{\beta})$ are twice continuously differentiable with respect to $(\mathbf{u}, \boldsymbol{\mu})$ in a neighbourhood of \mathbf{U}_∞^T , the Lagrangian function $L_p(\mathbf{u}, \boldsymbol{\mu}) = \phi_p(\mathbf{u}) + \boldsymbol{\mu}_k^T \mathbf{g}_p(\mathbf{u})$ is continuously differentiable with respect to $(\mathbf{u}, \boldsymbol{\mu})$ in a neighbourhood of \mathbf{U}_∞^T , and $(\mathbf{u}^*(\cdot), \boldsymbol{\mu}^*(\cdot))$ are continuously differentiable with respect to $\boldsymbol{\Lambda}$ in a neighbourhood of $\boldsymbol{\Lambda}_\infty^T$. The following conditions for the model-based problem have to be satisfied:

1. \mathbf{u}_∞^* is a regular point for the active constraints. This means that the gradient vectors of the active constraints $\nabla_{\mathbf{u}} \mathbf{g}_{M,act}(\mathbf{u}_\infty^*)$ are linearly independent.
2. The strict complementary slackness for problem (3.2) holds at \mathbf{u}_∞^* ($\boldsymbol{\mu}_\infty^T \mathbf{g}_M(\mathbf{u}_\infty^*, \boldsymbol{\beta}) = 0$).
3. $\boldsymbol{\mu}_\infty \geq 0$, $\mathbf{g}_M(\mathbf{u}_\infty^*) \leq 0$
4. The gradient of the Lagrangian is equal to zero $\nabla_{\mathbf{u}} L_M(\mathbf{U}_\infty^T, \boldsymbol{\Lambda}_\infty) = 0$ and the Hessian of L_M , $\nabla_{\mathbf{u}}^2 L_M(\mathbf{U}_\infty^T, \boldsymbol{\Lambda}_\infty)$ is positive definite.

Then is a KKT point associated with the plant corresponding to a local optimum. For more details see (Navia, et al., 2015).

As usual in MA methodology, the modifiers can be applied after using a first-order exponential filter, as shown in (3.9) where \mathbf{K}_λ and \mathbf{K}_ε are diagonal matrices with eigenvalues in the interval (0, 1], in order to smooth changes in the decision variables and to ensure convergence.

$$\begin{aligned}\boldsymbol{\lambda}_k^L &= (\mathbf{I} - \mathbf{K}_\lambda) \boldsymbol{\lambda}_{k-1}^L + \mathbf{K}_\lambda \left(\nabla_{\mathbf{u}} L_P \big|_{\mathbf{u}_{k-1}, \boldsymbol{\mu}_{k-1}} - \nabla_{\mathbf{u}} L \big|_{\mathbf{u}_{k-1}, \boldsymbol{\mu}_{k-1}} \right) \\ \boldsymbol{\varepsilon}_k &= (\mathbf{I} - \mathbf{K}_\varepsilon) \boldsymbol{\varepsilon}_{k-1} + \mathbf{K}_\varepsilon \left(\mathbf{g}_P(\mathbf{u}_{k-1}) - \mathbf{g}(\mathbf{u}_{k-1}) \right)\end{aligned}\quad (3.9)$$

The Lagrange multipliers can also be adjusted as follows:

$$\mu_{i,k} = \max \left\{ 0, \mu_{i,k-1} + b_i (\mu_{i,k}^* - \mu_{i,k-1}) \right\} \quad i = 1, \dots, n_g \quad (3.10)$$

where $\mu_{i,k}^*$ are the optimal values of the Lagrange multipliers of the modified problem (3.2) and b_i is the gain of the first order filter.

The number of required modifiers is drastically reduced by using the corrections computed by (3.9), as shown in equation (3.11), compared to the number of modifiers for the standard version given by (3.1).

$$n_{mod} = n_g + n_u \quad (3.11)$$

Compared to traditional gradient based MA techniques, this approach has the advantage that it only requires one gradient modifier $\boldsymbol{\lambda}_k^L$ for each process input; but it has the disadvantage that the modified cost and constraint functions do not provide first-order approximations of the plant constraint at each RTO iteration, that is, the slope of the modelled constraint is not being adjusted by first-order modifiers that are required to achieve matched first-order optimality conditions. This fact may make the convergence rate slower if gradient-based MA techniques are applied (Marchetti, et al., 2016). In addition, during the intermediate iterations, constraints may be violated as they are not being properly corrected. This problem, however, is common to all the MA approaches, despite the introduction of first-order corrections in the constraints. Nevertheless, it will converge to the optimum because the KKT point will be satisfied ($\nabla_{\mathbf{u}} L_P = \nabla_{\mathbf{u}} L_M = 0$) L_M being the Lagrangian of the modified problem given by (3.2), as has been demonstrated before.

3.3 Efficient Nested Modifier Adaptation for dealing with operational constraints

This section shows the use of the previously defined Lagrangian gradient corrections in NMA. For this methodology, the use of these corrections implies a clear advantage, since it is directly affected by the size of the problem. The larger the number of modifiers, the bigger the number of decision variables of the upper optimization layer, which may imply a slow convergence to the optimum. Figure 3.1 describes the alternative NMA method that considers the minimization of the Lagrangian function measured from the process and only one modifier for the cost function:

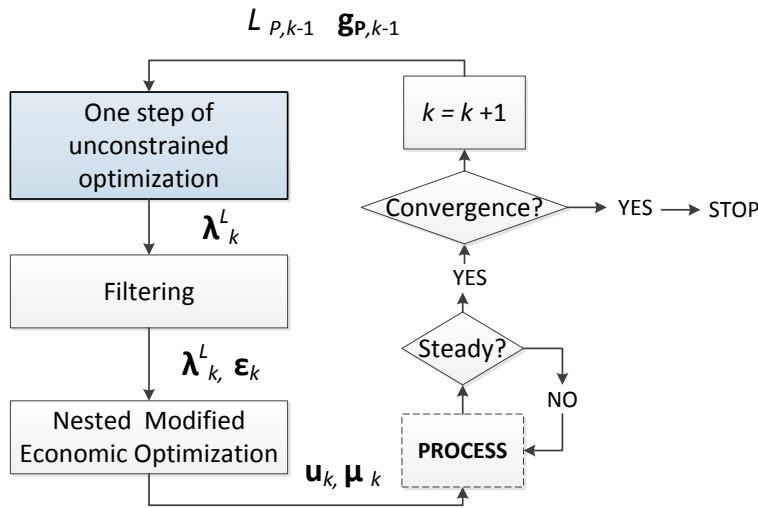


Figure 3.1. Schematic alternative NMA.

As stated in chapter 2, the key point of NMA is to avoid the calculation of experimental gradients by using a gradient-free optimization algorithm to directly update the modifiers. In this case, the optimization problem solved in the upper layer takes into account the minimization of the process Lagrangian as shown in (3.12), with the idea of reducing the number of required modifiers:

$$\min_{\lambda_k^L} L_p(\mathbf{u}_k^*, \boldsymbol{\mu}_k^*) = \phi_p(\mathbf{u}_k^*) + \boldsymbol{\mu}_k^{\top} \mathbf{g}_p(\mathbf{u}_k^*) \quad (3.12)$$

The alternative NMA approach can be implemented as shown in Algorithm 3.1:

ALGORITHM 3.1: Efficient NMA for dealing with operational constraints
 (Rodríguez-Blanco, et al., 2017)

Step 1. Set $k = 0$ and initialize λ_0^L , μ_0^* , u_0^* and ϵ_0 . If the Nelder-Mead algorithm is used in the upper optimization layer, it is necessary to take information from $n_{mod} + 1$ previous operating points to construct the initial simplex. These points will be generated by solving n_{mod} RTOs, where the value of the modifiers is changed from those λ_0^L initially given.

Step 2: Given u_{k-1}^* and μ_{k-1}^* , apply u_{k-1}^* to the process and wait for a steady state to be achieved to measure $g_{p,k-1}$ and $\phi_{p,k-1}$. Estimate the process Lagrangian via $L_p(u_{k-1}^*, \mu_{k-1}^*) = \phi_p(u_{k-1}^*) + \mu_{k-1}^{*T} g_p(u_{k-1}^*)$ and the zeroth order modifier ϵ_k .

Step 3: Perform one step of the upper optimization problem described by (3.12) to update the modifiers λ_k^L , in order to minimize L_p .

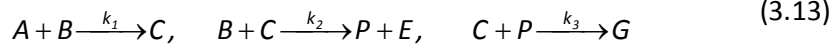
Step 4: Solve the modified optimization problem described by (3.2) to obtain u_k^* and μ_k^* . μ_k^* is given by the optimizer at u_k^* . Compare u_k^* with u_{k-1}^* and if the convergence criterion is satisfied, that is, $|u_k^* - u_{k-1}^*| \leq \text{tol}$, then stop. Otherwise, set $k = k + 1$ and go to step 2.

3.4 Implementation in the Otto-Williams reactor

The effectiveness of the proposed methodology is illustrated through a case study corresponding to a simulated Otto-Williams reactor presented in Figure 3.2. It is a continuous stirred-tank reactor (CSTR) that has been widely used in the literature to study the performance of different RTO approaches with modelling mismatch (Williams & Otto, 1960), (Roberts, 1979), (Navia, et al., 2013) and (Marchetti, et al., 2010). It will also be used in this thesis to study the performance of the developed MA methods. The obtained results are compared to those obtained by applying the traditional DMA and NMA techniques.

3.4.1 Process description

The system consists of a CSTR that is fed with two sources of raw material, A and B , by means of the streams F_A (kg/s) and F_B (kg/s), respectively. Inside the vessel, three parallel reactions take place forming 4 new compounds: C , G , E and P , as (3.13) shows.



These compounds, along with the unused reactive, leave the reactor from the bottom of the vessel in a single stream F_R . X_i represents the mass fraction of the i compound ($i = A, B, C, E, G, P$) inside the reactor and T_R ($^{\circ}\text{C}$) is the reactor temperature.

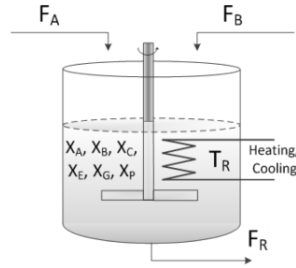


Figure 3.2. Diagram of the Otto-Williams reactor.

The system can be described using a first principles model, where the mass balance for each compound in the reactor is defined as follows:

$$V_R \frac{dX_A}{dt} = F_A - F_R X_A - V_R r_1 \quad (3.14)$$

$$V_R \frac{dX_B}{dt} = F_B - F_R X_B - V_R r_1 \frac{M_B}{M_A} - V_R r_2 \quad (3.15)$$

$$V_R \frac{dX_C}{dt} = -F_R X_C + V_R r_1 \frac{M_C}{M_A} - V_R r_2 \frac{M_C}{M_B} - V_R r_3 \quad (3.16)$$

$$V_R \frac{dX_E}{dt} = -F_R X_E + V_R r_2 \frac{M_E}{M_B} \quad (3.17)$$

$$V_R \frac{dX_G}{dt} = -F_R X_G + V_R r_3 \frac{M_G}{M_C} \quad (3.18)$$

$$V_R \frac{dX_P}{dt} = -F_R X_P + V_R r_2 \frac{M_P}{M_B} - V_R r_3 \frac{M_P}{M_C} \quad (3.19)$$

$$F_R = F_A + F_B \quad (3.20)$$

where M_i represents the molecular weight of the compound i , and r_j is the molecular reaction rate of the chemical reaction j defined with respect to its limiting reactive. V_R is the volume expressed as kg, since density is considered a constant in the reactor. As we are dealing with pseudo-compounds it is necessary to define the relation between their molecular

weights. This can be obtained by assuming that $M_A = M_B = M_C$. Under this consideration the ratios from (3.15)-(3.19) are:

$$\frac{M_P}{M_B} = 1 \quad \frac{M_C}{M_A} = \frac{M_C}{M_B} = \frac{M_E}{M_B} = 2 \quad \frac{M_G}{M_C} = 1.5 \quad \frac{M_P}{M_C} = 0.5 \quad (3.21)$$

Regarding the reaction rate, it can be calculated as follows:

$$r_1 = k_1 X_A X_B \quad (3.22)$$

$$r_2 = k_2 X_B X_C \quad (3.23)$$

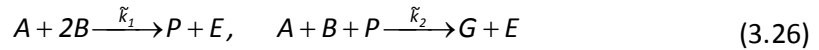
$$r_3 = k_3 X_C X_P \quad (3.24)$$

where k_j is the kinetic constant of the reaction j that can be obtained using an Arrhenius expression, and E_{Aj} is the activation energy from reaction j .

$$k_j = k_j^0 e^{\left(\frac{E_{Aj}}{T_R}\right)}, \quad j = 1, 2, 3 \quad (3.25)$$

3.4.2 Steady-state RTO model

The mass fraction of the product C is one order of magnitude below the rest of the compounds. Therefore, a common choice in a gross representation of the process is to consider only the other five species, with the corresponding modelling mismatch. Then, only two parallel reactions inside the reactor are considered:



with this source of modelling mismatch, the steady state model to be used in the RTO layer is the following:

$$F_A - F_R X_A - V_R \tilde{\gamma}_1 - V_R \tilde{\gamma}_2 = 0 \quad (3.27)$$

$$F_B - F_R X_B - 2V_R \tilde{\gamma}_1 - V_R \tilde{\gamma}_2 = 0 \quad (3.28)$$

$$-F_R X_E + 2V_R \tilde{\gamma}_1 = 0 \quad (3.29)$$

$$-F_R X_G + 3V_R \tilde{\gamma}_2 = 0 \quad (3.30)$$

$$-F_R X_P + V_R \tilde{\gamma}_1 - V_R \tilde{\gamma}_2 = 0 \quad (3.31)$$

$$F_R = F_A + F_B \quad (3.32)$$

$$\tilde{\gamma}_1 = \tilde{k}_1 X_A (X_B)^2 \quad (3.33)$$

$$\tilde{\gamma}_2 = \tilde{k}_2 X_A X_B X_P \quad (3.34)$$

$$\tilde{k}_j = \tilde{k}_j^0 e^{\left(\frac{E_{A,j}}{T_R}\right)}, \quad j = 1, 2 \quad (3.35)$$

where the tilde represents the parameters used in the model that includes mismatch.

3.4.3 Optimization problem

The optimization objective, as described in the benchmark (Williams & Otto, 1960), is to maximize the operating profit expressed as the cost difference between the product and reactant flowrates, so the process cost function is defined by (3.36):

$$\phi = F_R (X_P P_P + X_E P_E) - F_A X_A C_A - F_B X_B C_B \quad (3.36)$$

where P_P and P_E are the prices for the products P and E , respectively, and C_A and C_B the cost of the reactants A and B . The flowrate of reactant A (F_A) is fixed at 1.8275 kg/s. The flowrate of reactant B (F_B) and the reactor temperature (T_R) are the decision variables, thus $\mathbf{u} = [F_B, T_R]$.

So, the model based optimization can be summarized as: finding the decision variables F_B and T_R that maximize the profit, subject to a model that only takes into account five compounds and two chemical reactions, corresponding to the simulated modelling mismatch. In addition, for the implementation of the alternative NMA method described in this chapter, two operational constraints have been added to the original benchmark. These constraints fix the upper limits for the composition of compounds A and G ($X_A^U = 0.085$, $X_G^U = 0.105$):

$$\max_{\mathbf{u}=[F_B, T_R]} \phi$$

s.t

steady – state model with mismatch (3.27) – (3.35) (3.37)

$$X_A - X_A^U \leq 0 \quad X_G - X_G^U \leq 0$$

$$F_B^L \leq F_B \leq F_B^U \quad T_R^L \leq T_R \leq T_R^U$$

The optimal steady-state solution for the plant (simulated reality) is presented in Table 3.1. The mass fractions obtained at the optimal solution are given in Table 3.2.

Table 3.1. Process Optimum.

F_B^* (kg/s)	T_R^* (°C)	ϕ^* (\$/s)
4.899	89.985	191.017

Table 3.2. Mass Fractions at the Optimum.

X_A^*	X_B^*	X_C^*	X_E^*	X_G^*	X_P^*
0.085	0.397	0.015	0.288	0.105	0.110

The values of the parameters used from (3.14) to (3.35) are summarized in Table 3.3.

Table 3.3. Value of the Model Parameters.

Parameter	Value	Parameter	Value
F_A (kg/s)	1.8725	\tilde{E}_{A1} (°C)	-8077.6
V_R (kg)	2105	\tilde{E}_{A2} (°C)	-12438.5
k_1^0	1.6599×10^6	F_B^L (kg/s)	3
k_2^0	7.2177×10^8	F_B^U (kg/s)	6
k_3^0	2.6745×10^{13}	T_R^L (°C)	70
E_{A1} (°C)	-6666.7	T_R^U (°C)	100
E_{A2} (°C)	-8333.3	P_P (\$/kg)	1143.38
E_{A3} (°C)	-11111	P_E (\$/kg)	25.92
$\tilde{\kappa}_1^0$	2.611×10^{12}	C_A (\$/kg)	76.23
$\tilde{\kappa}_2^0$	1.655×10^8	C_B (\$/kg)	114.34

By applying traditional MA, the modified optimization problem solved at each RTO iteration is described by (3.38), where ϕ_M is the modified cost function and $g_{M,1}, g_{M,2}$ are the modified constraints:

$$\begin{aligned}
& \max_{u=[F_B, T_R]} \phi_M = \phi + \boldsymbol{\lambda}_k^T (\mathbf{u} - \mathbf{u}_{k-1}) \\
& \text{s.t} \\
& \text{steady - state model with mismatch (3.27) - (3.35)} \\
& \mathbf{g}_{M,1} = X_A - X_A^U + \mathbf{v}_{1,k}^T (\mathbf{u} - \mathbf{u}_{k-1}) + \varepsilon_{1,k} \leq 0 \\
& \mathbf{g}_{M,2} = X_G - X_G^U + \mathbf{v}_{2,k}^T (\mathbf{u} - \mathbf{u}_{k-1}) + \varepsilon_{2,k} \leq 0 \\
& F_B^L \leq F_B \leq F_B^U \quad T_R^L \leq T_R \leq T_R^U
\end{aligned} \tag{3.38}$$

where the first order modifiers $\boldsymbol{\lambda}_k, \mathbf{v}_k$ are computed as the difference between the model and experimental gradients as shown in (3.3), obtained from past operating points using DMA, or given by the upper optimization layer when NMA is applied. In this problem, according to (3.1), the number of modifiers is eight, two first-order modifiers for the cost function, another two for each constraint plus the zeroth-order modifiers for both constraints.

However, if the alternative MA technique described previously is applied, the modified optimization problem will be formulated as (3.39), with only two Lagrangian modifiers $\boldsymbol{\lambda}_k^L$, one for each decision variable, and two zeroth-order modifiers $\varepsilon_{i,k}$, one for each constraint. These modifiers are given by solving the unconstrained optimization problem (3.12), taking into account the minimization of the process Lagrangian.

$$\begin{aligned}
& \max_{u=[F_B, T_R]} \phi_M = \phi + \boldsymbol{\lambda}_k^{L\ T} (\mathbf{u} - \mathbf{u}_{k-1}) \\
& \text{s.t} \\
& \text{steady - state model with mismatch (3.27) - (3.35)} \\
& \mathbf{g}_{M,1} = X_A - X_A^U + \varepsilon_{1,k} \leq 0 \\
& \mathbf{g}_{M,2} = X_G - X_G^U + \varepsilon_{2,k} \leq 0 \\
& F_B^L \leq F_B \leq F_B^U \quad T_R^L \leq T_R \leq T_R^U
\end{aligned} \tag{3.39}$$

3.4.4 Results

The results obtained, as a function of the iteration number, are shown in the following figures, where NMA indicates the results of applying NMA with the traditional modifiers (correcting the cost and constraint gradients) solving (3.38). NMA Lag shows the results obtained by correcting the Lagrangian gradients, solving (3.39), and DMA shows the results given by the

implementation of the Dual Modifier Adaptation (section 2.1.3). RTO is solved every hour, the time required for the process to achieve a new steady-state. Modifiers are filtered with a gain of 0.3 for the first order modifiers and 1 for the zero order modifiers.

As explained in the previous chapter, the DMA methodology requires the addition of a new constraint on the optimization problem, ($\kappa^{-1}(\mathbf{S}_k) \geq a$), which takes into account the grade of excitation of the process, so as to ensure sufficient information in the measurements to guarantee gradient accuracy. This is equivalent to choosing the parameter a (lower limit of $\kappa^{-1}(\mathbf{S}_k)$), whose selected value in this case is 0.10.

Both, the simulation and optimization have been carried out in the modelling and simulation software EcosimPro (Int, 2013), using an external library for the optimization called SNOPT (Gill, et al., 2008), a software package for solving large-scale optimization problems.

The following graphs, in particular Figure 3.3, show that the three approaches are able to reach the real optimum. For the case of NMA, the use of the Lagrangian based modifier reduces the number of required iterations by four times, since the number of modifiers is smaller; so the upper layer optimization has fewer decision variables, which makes it easier to achieve the optimum. In addition, the time required to achieve the plant optimum is also considerably reduced, making the implementation of NMA in real applications easier.

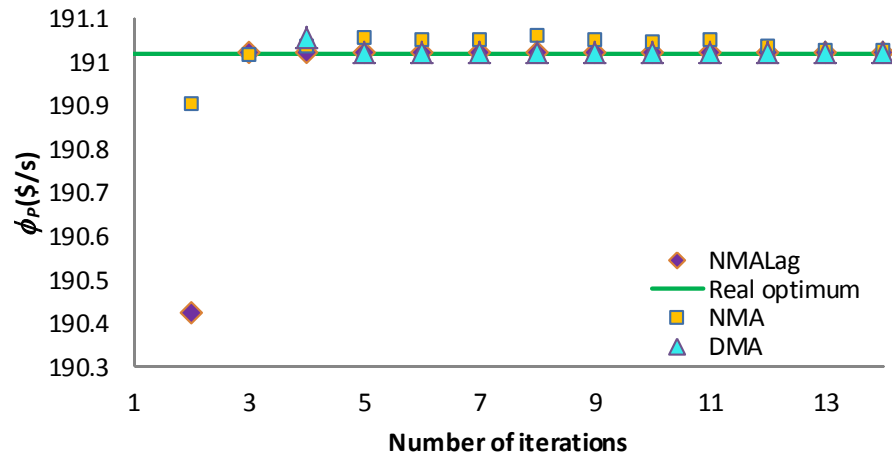


Figure 3.3. Evolution of the process cost function ϕ .

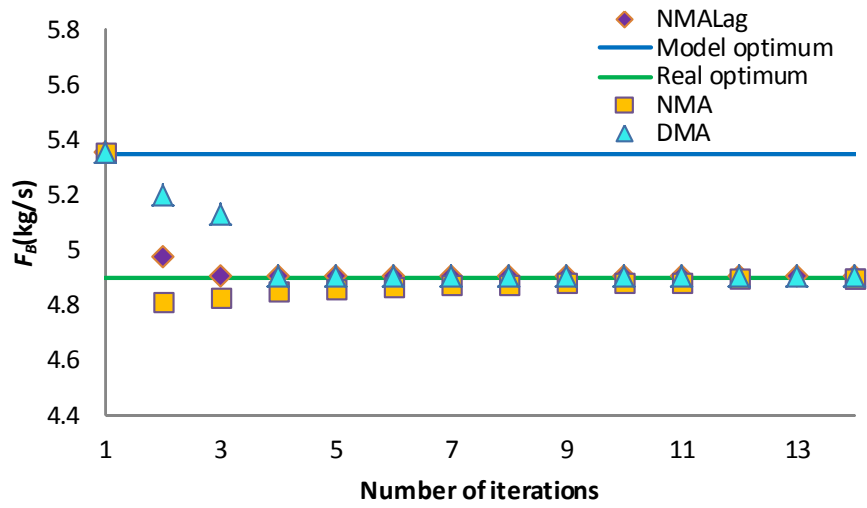


Figure 3.4. Evolution of the decision variable F_B .

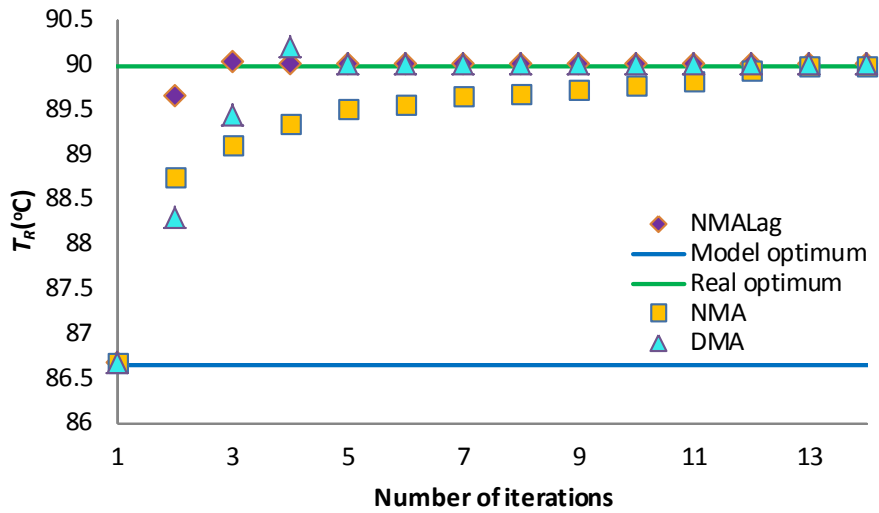


Figure 3.5. Evolution of the decision variable T_R .

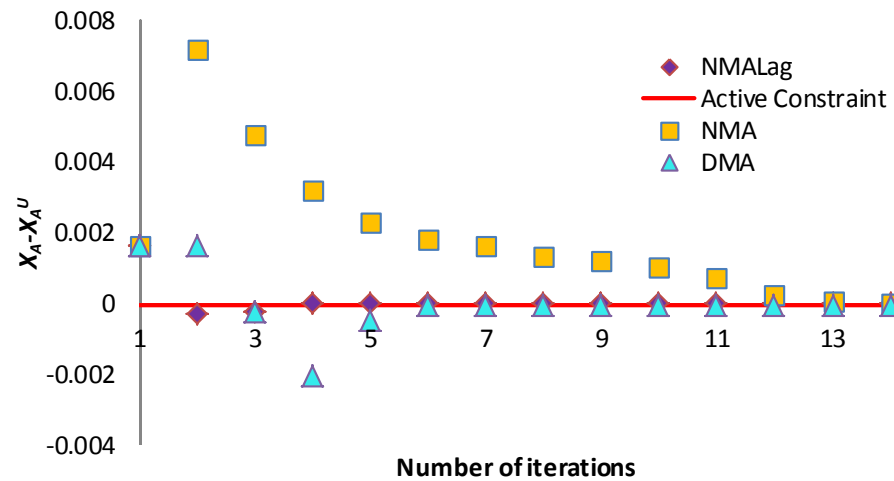


Figure 3.6. Evolution of the process constraint on composition of component A.

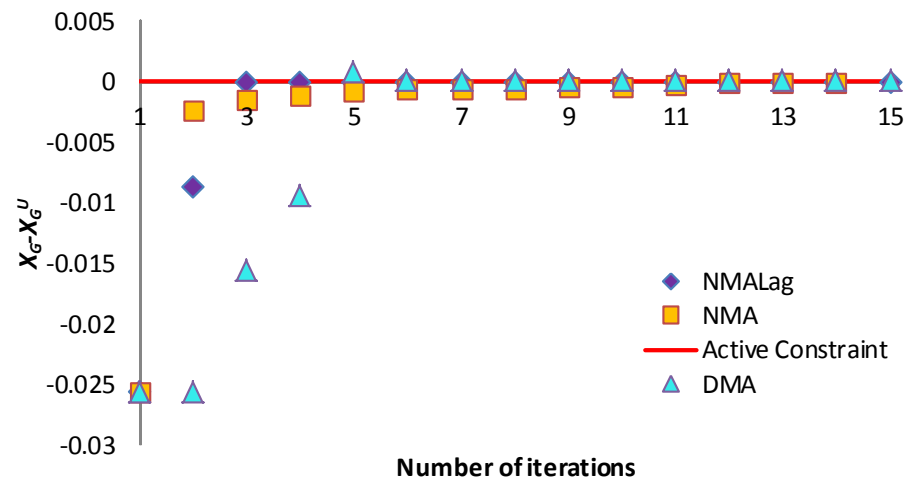


Figure 3.7. Evolution of the process constraint on composition of component G.

Fixing a tolerance band of 0.5% with respect to the optimal value of the cost function, the obtained results can be summarized in Table 3.5. The differences between the process optimum, the nominal solution and the model optimum that is, the results obtained by applying the nominal solution (solution of the RTO without modifiers) to the process are shown in Table 3.4. For the DMA approach, the number of steady-states needed was

9, the first three to initialize the computation of the process gradients ($n_u + 1 = 3$) and 6 related to RTO solutions. NMA requires 46 steady-states: first, 3 initial steady-states to estimate plant gradients for the computation of initial modifiers, then 7 to construct the first simplex corresponding to the number of first-order modifiers plus 1, and finally 36 RTO executions. The alternative NMA formulation only requires 11 steady-states: 3 for the initial estimation of the modifiers, another 3 to construct the simplex and, finally, 5 RTO executions to converge to the optimum.

Table 3.4. Model and process optimum.

	Process optimum	Model optimum	Nominal solution
F_B (kg/s)	4.899	5.349	5.349
T_R (°C)	89.985	86.648	86.648
X_A	0.085	0.086	0.085
X_G	0.105	0.079	0.090
ϕ (\$/s)	191.017	176.678	199.141

Table 3.5. Summary Results.

	NMA	NMALag	DMA
Decision variables	2	2	2
Constraints	2	2	2
Number of Modifiers	8	3	8
First-order modifiers	6	2	6
Iterations	14	4	6
Steady state points	46	11	9

As can be seen in Table 3.5, the use of the corrections based on the Lagrangian gradients makes the results obtained from NMA very similar to the ones obtained after applying DMA. This fact supposes an important advantage since, traditionally, DMA drives the process to the optimum in a faster and more direct way than NMA, as it deals with the information of the plant and process gradients. This formulation allows the performance of both approaches to be comparable and even NMA will be faster as the number of constraints increases.

3.4.5 Conclusions

This chapter has proposed an alternative MA formulation, whereby the traditional modifiers that correct the cost and constraint gradients are replaced by modifiers for a single Lagrangian gradient. As with all other MA schemes, this formulation also guarantees that a point which satisfies the first-order optimality conditions of the plant will be reached upon convergence, since the KKT conditions will be satisfied, $(\nabla L_p|_{\mathbf{u}} = \nabla L_M|_{\mathbf{u}} = 0)$. The main advantage of this approach is that the number of required modifiers is reduced, which significantly speeds up convergence.

In the context of NMA, this feature represents a major advantage since it is directly affected by the size of the problem. The larger the number of modifiers, the bigger the number of decision variables of the upper optimization layer, which may imply a slow convergence to the optimum. In this chapter, the alternative method has been implemented in the Otto-Williams reactor, but other applications will be shown in chapters five and six. Realistic case studies have been considered corresponding to a depropanizer distillation column of a petrol refinery and the transport of natural gas through the gas pipelines. It can be concluded that, using the described formulation, the number of iterations is considerably reduced, compared to standard MA formulations, and therefore the time required to achieve the real optimum, which makes the implementation of NMA in real applications easier.

The use of these modifiers allows NMA to achieve the plant optimum with a similar convergence rate to DMA. Using the traditional modifiers, DMA drives the process to the optimum in a faster and more direct way than NMA, since it deals with the information of the model and process gradients. However, using the Lagrangian gradients, the convergence rates of both methods, NMA and DMA, are very similar, NMA can be even faster if it is applied in a case study with a large number of operational constraints, since the benefits of the proposed approach will be larger, the greater the number of process-dependent constraints.

4 MODIFIER ADAPTATION USING TRANSIENT MEASUREMENTS

Traditionally, MA proceeds by iteratively adjusting the optimization problem with modifiers calculated from steady-state information at each RTO execution. This implies a long convergence time. This chapter presents several approaches to speed up the convergence to the optimum by using transient information of the process. Among them, a new method is described that is based on a recursive identification algorithm to estimate process gradients from transient measurements, achieving the plant optimum faster than traditional MA techniques.

4.1 Introduction

MA is normally applied based on static information, as happens in the gradient based method called DMA (Brdyś & Tatjewski, 1994) or the gradient free method, NMA (Navia, et al., 2015). However, the implementation of these methods is sometimes impractical, especially in processes with a long settling time, as the process needs to reach the steady-state at each RTO execution to estimate the process gradients. Over these long periods of time, the operating conditions or differences between process and model may change and the method may not converge to the real optimum, that is, an operating point that satisfies the NCO of the process. This issue makes the application of this methodology impractical in these cases. Besides, and not less important, this kind of method does not guarantee a feasible path during convergence to the real optimum, only when this point is achieved are all the process constraints satisfied. So, the longer it takes to achieve the optimal operation, the bigger the probability of violating operational constraints, which may lead to the loss of benefits or to the violation of safety and environmental constraints that may result in a dangerous operation.

In order to speed up the convergence of the MA methodology for slow dynamic processes and reduce the risk of process constraint violations, several researchers have suggested the use of transient measurements to estimate the variables required by the steady-state optimization solved in the RTO layer. This idea was pursued by Zhang and Roberts in 1990 (Zhang & Roberts., 1990), who combined the ISOPE scheme with a linear dynamic model identification to compute process gradients for the steady-state optimization of nonlinear constrained processes with slow dynamics. However, this work did not address the problem of shortening the gradient estimation time. In contrast, in 2014, François and Bonvin (François, et al., 2014) proposed an approach that uses transient measurements to compute process gradients by the neighbouring extremal method, NE, which relies on the accuracy of the linearization resulting from a variational analysis of the nominal model (see section 4.2). However, none of these techniques work well in the presence of strong structural plant-model mismatch.

Two approaches are developed in this thesis to overcome this problem. The first combines the NE-based gradient estimation during the transient with the estimation of gradients based on stationary data. In this way, it is possible to deal with both parametric and structural plant-model mismatch (see section 4.3).

The second aims to extend the idea of using transient measurements to speed up convergence to the optimum of the MA methodology by estimating the process cost and constraint gradients directly using a recursive identification method. Thus, waiting for the steady-state at each RTO iteration is no longer necessary (see section 4.4). This identification method does not require any assumption about the type of uncertainty of the model, parametric or structural, or the knowledge of the parameter responsible for the plant-model mismatch. For this reason, it is able to deal with both parametric and structural uncertainties. The key here is to estimate the plant gradients at the steady-state from process measurements taken during the transient. The estimated gradients are then used to update the value of the modifiers and solve the modified economic optimization problem every sample time during the transient, so as to drive the plant operation to a point that satisfies the NCO of the plant faster than traditional MA, as can be observed in the results obtained from the implementation of this technique on the Otto-Williams reactor and a laboratory-scale flotation column (see sections 4.4.2 and 4.4.3).

4.2 Modifier Adaptation computing plant gradients from neighbouring extremal control

The idea of speeding up the convergence of MA was pursued by Francois and co-workers (François & Bonvin, 2014), who developed an MA scheme that, under certain assumptions, satisfies this aim. The philosophy behind this framework is inspired from Neighbouring-Extremal techniques (NEC), which use transient information for steady-state optimization. The main difference is that the control update is not obtained by computing a control law, but rather by solving a modified optimization problem.

NEC aims to maintain process optimality in the presence of disturbances by implementing a control law that forces the satisfaction of the process NCO. It relies on linear approximations around the nominal optimum, namely the first-order variations of the necessary optimality conditions, to estimate the variation of the parametric uncertainty, $\delta\beta$, from the changes in the output measurements, $\delta\mathbf{y}$, and the inputs, $\delta\mathbf{u}$. For this reason, one condition that must be satisfied is $n_y \geq n_\beta$, that is, there are at least as many output measurements \mathbf{y} as there are uncertain parameters β .

Consider the following static optimization problem:

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi(\mathbf{u}, \boldsymbol{\beta}) \\ \text{s.t.} \quad & g_i(\mathbf{u}, \boldsymbol{\beta}) \leq 0 \quad i = 1, \dots, n_g \end{aligned} \quad (4.1)$$

where ϕ is a smooth function that represents the cost depending on the unknown parameters $\boldsymbol{\beta}$ and the decision variables represented by \mathbf{u} and g_i are the inequality constraints. Moreover, the model includes the output equation (4.2):

$$\mathbf{y} = \mathbf{H}(\mathbf{u}, \boldsymbol{\beta}) \quad (4.2)$$

Here, \mathbf{u} and \mathbf{y} represent the inputs and outputs, respectively, at steady-state.

From the variational analysis of the first-order variations of the NCO, considering the parametric variations, $\delta\boldsymbol{\beta}$, around the nominal values of the parameters, $\boldsymbol{\beta}_{\text{nom}}$, a first-order approximation to the cost gradient can be expressed as (4.3)

$$\nabla_{\mathbf{u}}\phi = \mathbf{A} \delta\mathbf{u} + \mathbf{B} \delta\boldsymbol{\beta} \quad (4.3)$$

with \mathbf{A} being the Hessian matrix ($n_u \times n_u$) and \mathbf{B} the ($n_u \times n_\beta$) matrix given by (4.4)

$$\mathbf{A} = \nabla_{\mathbf{u}\mathbf{u}}^2\phi \quad \mathbf{B} = \nabla_{\mathbf{u}\boldsymbol{\beta}}^2\phi \quad (4.4)$$

Output variables can be linearized with respect to \mathbf{u} and $\boldsymbol{\beta}$:

$$\delta\mathbf{y} = \mathbf{Q}\delta\mathbf{u} + \mathbf{P}\delta\boldsymbol{\beta} \quad (4.5)$$

with the ($n_y \times n_u$) matrix \mathbf{Q} and the ($n_y \times n_\beta$) matrix \mathbf{P} computed by (4.6)

$$\mathbf{Q} = \nabla_{\mathbf{u}}\mathbf{y} \quad \mathbf{P} = \nabla_{\boldsymbol{\beta}}\mathbf{y} \quad (4.6)$$

Let us assume $n_y \geq n_\beta$, that is, there are at least as many output measurements as there are uncertain parameters. Using (4.5), the parametric variations $\delta\boldsymbol{\beta}$ can be inferred from $\delta\mathbf{y}$ and $\delta\mathbf{u}$ as follows:

$$\delta\boldsymbol{\beta} = \mathbf{D}(\delta\mathbf{y} - \mathbf{Q}\delta\mathbf{u}) \quad (4.7)$$

where \mathbf{D} is an ($n_\beta \times n_y$) pseudoinverse of \mathbf{P} .

Equation (4.3) provides a first-order approximation of the cost gradient, which can be estimated from $\delta\mathbf{y}$ and $\delta\mathbf{u}$, using (4.7) to eliminate $\delta\boldsymbol{\beta}$, obtaining (4.8).

$$\nabla_{\mathbf{u}}\phi = \mathbf{G}_{\mathbf{y}}\delta\mathbf{y} + \mathbf{G}_{\mathbf{u}}\delta\mathbf{u} \quad (4.8)$$

with

$$\mathbf{G}_{\mathbf{y}} = \mathbf{BD} = \nabla_{\mathbf{u}\boldsymbol{\beta}}^2\phi(\nabla_{\boldsymbol{\beta}}\mathbf{y})^+ \quad (4.9)$$

$$\mathbf{G}_{\mathbf{u}} = \mathbf{A} - \mathbf{BDQ} = \nabla_{\mathbf{u}\mathbf{u}}^2\phi - \nabla_{\mathbf{u}\boldsymbol{\beta}}^2\phi(\nabla_{\boldsymbol{\beta}}\mathbf{y})^+\nabla_{\mathbf{u}}\mathbf{y} \quad (4.10)$$

The first-order RTO modifiers $\boldsymbol{\lambda}$ can be expressed by the following equations, where $\mathbf{y}_p(t)$ and \mathbf{y} are the vector of output measurements from the process (measured during the transient) and the model, \mathbf{u}_p and \mathbf{u} the vector of inputs applied to the process and the model, and finally, \mathbf{y}_{nom} the vector of model outputs for the nominal solution \mathbf{u}_{nom} , that is, the solution of RTO without modifiers. The parametric uncertainty causes the optimal inputs and outputs, as well as the gradient, to deviate from their nominal values. For this reason, the plant gradients are estimated using the approximation shown in (4.8), but adding the deviation in the process inputs and outputs with respect to the nominal values to extract information about the effect of the uncertain parameters on the estimated gradients. Since this method considers that there is only parametric plant-model mismatch, $\mathbf{G}_{\mathbf{y}}$ and $\mathbf{G}_{\mathbf{u}}$ will be equal for the RTO model and the plant.

$$\begin{aligned} \boldsymbol{\lambda} = \nabla_{\mathbf{u}}\phi_p - \nabla_{\mathbf{u}}\phi &= \mathbf{G}_{\mathbf{y}}\delta\mathbf{y}_p + \mathbf{G}_{\mathbf{u}}\delta\mathbf{u}_p - (\mathbf{G}_{\mathbf{y}}\delta\mathbf{y} + \mathbf{G}_{\mathbf{u}}\delta\mathbf{u}) = \\ &\mathbf{G}_{\mathbf{y}} \cdot (\delta\mathbf{y}_p - \delta\mathbf{y}) + \mathbf{G}_{\mathbf{u}} \cdot (\delta\mathbf{u}_p - \delta\mathbf{u}) \end{aligned} \quad (4.11)$$

where,

$$\delta\mathbf{y}_p = \mathbf{y}_p(t) - \mathbf{y}_{nom} \quad (4.12)$$

$$\delta\mathbf{y} = \mathbf{y} - \mathbf{y}_{nom} \quad (4.13)$$

and

$$\delta\mathbf{u}_p = \mathbf{u}_p - \mathbf{u}_{nom} \quad (4.14)$$

$$\delta\mathbf{u} = \mathbf{u} - \mathbf{u}_{nom} \quad (4.15)$$

The same estimation procedure is followed to compute the constraint modifiers $\boldsymbol{\gamma}_i$, formulating (4.8) as the estimation of constraint gradients and

computing \mathbf{G}_y and \mathbf{G}_u as a function of the constraint value instead of the cost value.

$$\nabla_{\mathbf{u}} g_i = \mathbf{G}_{y,i} \delta \mathbf{y} + \mathbf{G}_{u,i} \delta \mathbf{u} \quad i = 1, \dots, n_g \quad (4.16)$$

$$\mathbf{G}_{y,i} = \mathbf{B}_i \mathbf{D}_i = \nabla_{\mathbf{u}\beta}^2 g_i (\nabla_{\beta} \mathbf{y})^+ \quad (4.17)$$

$$\mathbf{G}_{u,i} = \mathbf{A}_i - \mathbf{B}_i \mathbf{D}_i \mathbf{Q}_i = \nabla_{\mathbf{u}\mathbf{u}}^2 g_i - \nabla_{\mathbf{u}\beta}^2 g_i (\nabla_{\beta} \mathbf{y})^+ \nabla_{\mathbf{u}} \mathbf{y} \quad (4.18)$$

$$\begin{aligned} \mathbf{y}_i = \nabla_{\mathbf{u}} g_i - \nabla_{\mathbf{u}} g_i &= \mathbf{G}_{y,i} \delta \mathbf{y}_p + \mathbf{G}_{u,i} \delta \mathbf{u}_p - (\mathbf{G}_{y,i} \delta \mathbf{y} + \mathbf{G}_{u,i} \delta \mathbf{u}) = \\ & \mathbf{G}_{y,i} \cdot (\delta \mathbf{y}_p - \delta \mathbf{y}) + \mathbf{G}_{u,i} \cdot (\delta \mathbf{u}_p - \delta \mathbf{u}) \end{aligned} \quad (4.19)$$

The main advantage of this method is that the process cost gradient is only estimated by using the model offline, through the expression for \mathbf{G}_y and \mathbf{G}_u , and $\mathbf{G}_{y,i}$ and $\mathbf{G}_{u,i}$, whereas $\delta \mathbf{y}_p$ is measured from the process during the transient. Nevertheless, this method also presents strong limitations, such as:

- The method works well only if there is parametric uncertainty or little structural plant model mismatch.
- The user has to know what the uncertain parameters are, although their exact values are unknown.
- There have to be at least as many available output measurements as uncertain parameters ($n_y \geq n_\beta$).

This methodology has been implemented by the developers of the method for the optimization of a simulated continuously stirred tank reactor (CSTR) (François & Bonvin, 2014), (François, et al., 2014), where the goal is to maximize the productivity of the most value-added reaction product at steady-state. By applying this methodology, the authors conclude that the time needed for convergence to the plant optimum varies from about two-thirds to twice the plant settling time, i.e., a factor of 2–6 reduction compared to static MA under the same uncertainty. However, this example only considers parametric plant-model mismatch. In this thesis, the application of this method in the presence of structural uncertainty has been studied (Rodríguez-Blanco, et al., 2016) through the optimization of a simulated depropanizer distillation column. The presence of parametric and structural uncertainty led to the development of two extensions of the previous method that will be presented in sections 4.3 and 4.4. The first one combines transient and steady-state measurements. The second uses

identification algorithms to estimate the process gradients, also using transient information. Both approaches will be explained below and the results obtained by its implementation in the distillation column will be shown in chapter five.

4.3 Combination of transient and steady-state measurements

Using the method described in the last section, waiting for the steady state at each iteration is not necessary since the modifiers are updated every RTO sample time during the transient. In this way the convergence of the MA approach is sped up. Nevertheless, this method also presents strong limitations that have been mentioned in the previous section. The main drawback is the fact that it does not work well in the presence of structural plant-model mismatch.

To overcome this limitation, a first extension is presented in this section that combines MA based on transient measurements with the traditional methodology which estimates process gradients from steady-state information. In that way, both parametric and structural plant-model mismatch can be dealt with.

This methodology consists of two steps:

1. First, MA is applied by using transient measurements, as explained in section 4.2, to overcome the parametric plant-model mismatch between the process and the model used in the model-based optimization layer.
2. Once the steady-state is achieved, the method based on transient measurements will not be able to find a better operating point, so static MA is applied at the end of the transient to deal with the structural plant-model mismatch, computing the process gradients from steady-state information, applying a traditional static MA technique, for instance the DMA methodology.

An example of the application of this methodology, which considers the optimal operation of a depropanizer distillation column in the presence of parametric and structural uncertainty, will be shown in chapter five. The use of this approach reduces the time required to achieve the process optimum

by 60% compared to traditional MA techniques based on steady-state information.

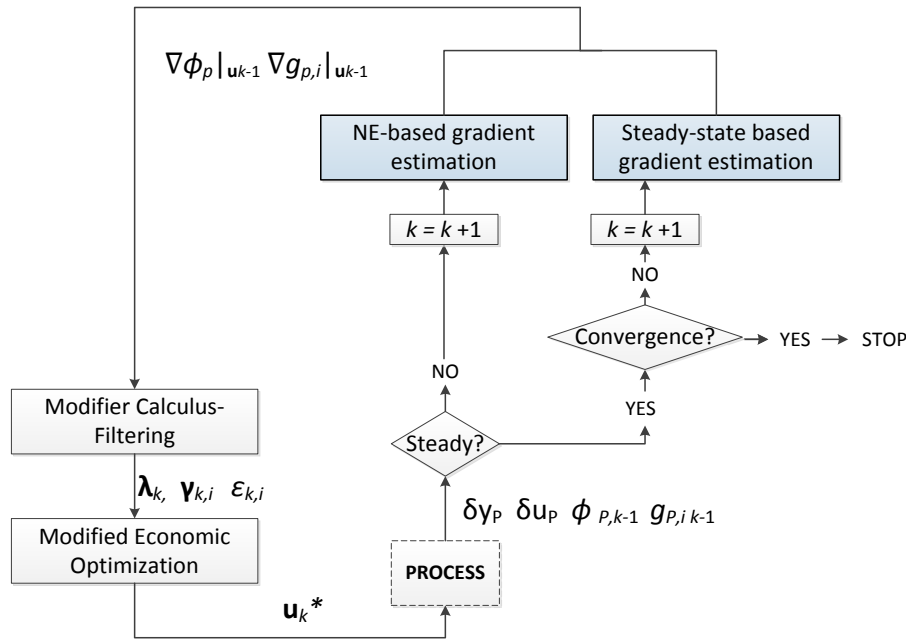


Figure 4.1. Schematic of MA based on transient and steady-state data.

4.4 Modifier Adaptation approach using Recursive Extended Least Squares to compute process gradients

The second alternative approach that has been developed is also based on the use of transient measurements. This technique considers that process gradients can be estimated from input-output data during the transient using adaptive estimation techniques. The proposed method uses transient measurements to estimate the plant gradients, and therefore the value of the modifiers, thus allowing faster convergence to the steady-state plant optimum than traditional MA, being valid in the presence of both parametric and structural uncertainty.

To estimate the process gradients, it is necessary to suppose an approximation for the variation of the process cost function $\Delta\phi_{P,k} = \phi_{P,k} - \phi_{P,k-1}$.

In this thesis, it is proposed to approximate this variation at time k as a quadratic Taylor polynomial that depends on the input applied until the present sample time \mathbf{u}_{k-1} and the one applied in the previous sample instant \mathbf{u}_{k-2} (see Figure 4.3). Other simpler functions can be supposed, for example a first-order approximation. However, this approximation would be realistic only for linear systems, which are rarely met in practice.

This estimator for the variation of the cost function is given by (4.20) where the hat “^” indicates that the variable is estimated from the process measurements. $\Delta\mathbf{u}_{k-1}$ is the column vector of the moves of the input variables (decision variables of the RTO problem) and the index k represents the RTO execution which, in transient methods, does not coincide with the steady-state number, as happens in static MA. $\hat{\boldsymbol{\theta}}_{k-1}$ is the column vector of the estimated parameters, which contains the required process gradients to compute the modifiers as the first element of this vector, i.e., the gradients of the process cost function with respect to the decision variables \mathbf{u}_{k-1} .

$$\Delta\hat{\phi}_{p,k} = \varphi_{k-1}^T \hat{\boldsymbol{\theta}}_{k-1} = \nabla_{\mathbf{u}_{k-1}} \phi \Delta\mathbf{u}_{k-1}^T + \nabla^2_{\mathbf{u}_{k-1}\mathbf{u}_{k-1}} \phi 1/2 \Delta\mathbf{u}_{k-1} \Delta\mathbf{u}_{k-1}^T \quad (4.20)$$

$$\varphi_{k-1}^T = [\Delta\mathbf{u}_{k-1}^T \quad 1/2 \Delta\mathbf{u}_{k-1} \Delta\mathbf{u}_{k-1}^T] \quad (4.21)$$

$$\hat{\boldsymbol{\theta}}_{k-1}^T = [\nabla_{\mathbf{u}_{k-1}} \phi \quad \nabla^2_{\mathbf{u}_{k-1}\mathbf{u}_{k-1}} \phi] \quad (4.22)$$

The gradients contained in $\boldsymbol{\theta}_k$ are estimated by using the recursive extended least squares algorithm (RELS) with forgetting factor α . The concept of forgetting means that older data are gradually discarded in favour of more recent information (Vahidi, et al., 2005).

This algorithm is based on the difference between the current input \mathbf{u}_{k-1} and the previous ones and the difference between the measured $\Delta\phi_{p,k}$ and the predicted $\Delta\hat{\phi}_{p,k}$ change in the cost function for the gradient estimation (Guay, 2014). Hence, the parameter estimation update approach is given as follows:

$$\hat{\boldsymbol{\theta}}_k = \hat{\boldsymbol{\theta}}_{k-1} + \frac{1}{\alpha} (\boldsymbol{\Sigma}_{k-1})^{-1} \varphi_{k-1} \left(1 + \frac{1}{\alpha} \varphi_{k-1}^T (\boldsymbol{\Sigma}_{k-1})^{-1} \varphi_{k-1} \right)^{-1} (\mathbf{e}_k) \quad (4.23)$$

where the terms involved in (4.23) are defined as follows:

$$\Sigma_0 = \frac{1}{\alpha} \mathbf{I} \quad (4.24)$$

$$\Delta \mathbf{u}_{k-1} = \mathbf{u}_{k-1} - \mathbf{u}_{k-2} \quad (4.25)$$

$$\Delta \hat{\phi}_{p,k} = \varphi_{k-1}^\top \hat{\boldsymbol{\theta}}_{k-1} \quad (4.26)$$

$$e_k = \Delta \phi_{p,k} - \Delta \hat{\phi}_{p,k} \quad (4.27)$$

$$\begin{aligned} (\Sigma_k)^{-1} &= \frac{1}{\alpha} (\Sigma_{k-1})^{-1} - \\ &\frac{1}{\alpha^2} (\Sigma_{k-1})^{-1} \varphi_{k-1} \left(1 + \frac{1}{\alpha} \varphi_{k-1}^\top (\Sigma_{k-1})^{-1} \varphi_{k-1} \right)^{-1} \varphi_{k-1}^\top (\Sigma_{k-1})^{-1} \end{aligned} \quad (4.28)$$

where Σ is the covariance matrix of the estimated error, whose initial value is Σ_0 , e_k is the output prediction error, and \mathbf{u}_k is the solution of the modified RTO problem for each sampling instant k during the transient. On the other hand, $\Delta \phi_{p,k} = \phi_{p,k} - \phi_{p,k-1}$ is the difference between the current process cost function and the cost function measured one sampling time before.

An important drawback of using passive identification with transient measurements is the degree of excitation that the system must have to estimate the gradients accurately. For this reason, a dual constraint is added to the RTO problem (Brdyś & Tatjewski, 1994), (Marchetti, et al., 2010), ($\kappa^{-1}(\mathbf{S}_k) \geq a$), which is described by (2.13) and (2.14). This constraint represents the dual characteristic of the method: while the rest of the optimization aims to converge to the optimum of the modified model (primal objective), the dual constraint ensures that, in the next RTO iteration, the system will have enough excitation to estimate the process gradient again (dual objective). However, this technique also identifies second order terms included in $\hat{\boldsymbol{\theta}}_k$ from second order data included in φ_k whose excitation is not guaranteed by the dual constraint. For this reason, the persistent excitation condition (4.29), based on the theory of system identification, must be checked in order to guarantee that all the plant gradients (included the second-order terms) have been properly estimated (Goodwin & Sin, 1984), (Ljung, 1987). This condition, given by (4.29), indicates that the square matrix $\Psi_k \Psi_k^\top$ must be positive definite in order to guarantee that there is enough process excitation. c_1 (a positive constant) fixes a lower limit for the excitation and Ψ_k is defined by (4.30).

$$\frac{1}{k} \sum_{k=1}^l \Psi_k \Psi_k^T \geq c_1 \mathbf{I} \quad (4.29)$$

$$\Psi_k^T = [\Delta \mathbf{u}_k^T \quad \Delta \mathbf{u}_{k-1}^T \quad \Delta \mathbf{u}_{k-2}^T \quad \dots \quad \Delta \mathbf{u}_{k-l}^T \quad 1/2 \Delta \mathbf{u}_k \Delta \mathbf{u}_k^T \quad 1/2 \Delta \mathbf{u}_{k-1} \Delta \mathbf{u}_{k-1}^T \quad 1/2 \Delta \mathbf{u}_{k-2} \Delta \mathbf{u}_{k-2}^T \quad \dots \quad 1/2 \Delta \mathbf{u}_{k-l} \Delta \mathbf{u}_{k-l}^T] \quad (4.30)$$

This condition can be satisfied if there are at least as many data sets measured from the process as estimated parameters, that is, $k \geq l = \text{size}[\hat{\boldsymbol{\theta}}_k]$.

The persistent excitation condition can be checked after the end of the algorithm, once the steady-state of the process has been achieved, or by adding a constraint to the RTO problem at the k^{th} execution, where $k = \text{size}[\hat{\boldsymbol{\theta}}_k]$, in order to guarantee that there is enough excitation. If this constraint is satisfied at the k^{th} iteration, this condition will be satisfied during the whole experiment since the more information we get, the better the estimation of the plant gradients.

This constraint would be formulated as shown in (4.29), where Ψ_k includes the decision variables of the RTO ($\Delta \mathbf{u}_k = \mathbf{u}_k - \mathbf{u}_{k-1}$). In this way, the optimizer will decide the next inputs to be applied, \mathbf{u}_k , forcing the process to be sufficiently excited.

On the other hand, taking into account the fact that, sometimes, the inputs of the model are not the same as the inputs of the process, for example, when there is an intermediate control layer that is not considered in the RTO model, or the fact that applying RTO during the transient implies that the input variable may not be able to reach the fixed set points; then the plant gradients must be estimated with respect to the set points of the process. Therefore, the variable \mathbf{u}_k that represents the solution of the RTO problem would indicate the set points computed for the input variables. This issue will be shown in the case study described in chapter five, which consists of a depropanizer distillation column with a MPC layer that determines the set points for the PID controllers. In this case, the decision variables in the RTO do not correspond to those of the process due to an intermediate MPC layer that is not considered in the RTO model.

One advantage of this technique, compared to that described in subsection 4.2, is that the new formulation does not require any assumption about the type of uncertainty of the model, parametric or structural, or the knowledge of the parameter responsible for the plant-model mismatch. Consequently, the proposed method can be applied to problems with both

parametric and structural uncertainty, without modifying the RTO model or identifying what the uncertain parameters are.

If enough measurements are taken during the transient, and therefore, the estimated gradients have been sufficiently corrected during this time, the true value will be achieved at the steady-state. This implies that an operating point that satisfies the NCO of the process may be reached in only one steady-state. However, this steady-state will be extended over time because the process inputs are very frequently manipulated, every sample time during the transient.

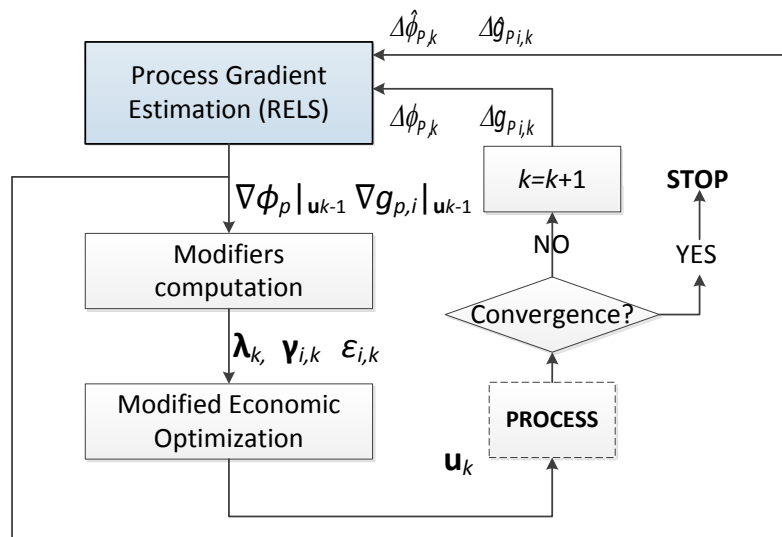


Figure 4.2. Schematic of MA based on the direct estimation of process gradients over the transient.

The steps needed to implement the previously described method are given next:

ALGORITHM 4.1: Modifier Adaptation using RELS (Rodríguez-Blanco, et al., 2017)

Step 1: Set $k = 0$ and initialize the vector of estimated parameters $\hat{\theta}_0$, the value of the estimation error e_0 and the covariance matrix Σ_0 , whose initial value is given by (4.24). Then, choose the tuning parameters α (the forgetting factor), the lower limit a for the dual constraint, and the lower limit for the persistent excitation c_1 . To start the algorithm, it is necessary to apply two operating points, u_{k-1} and u_{k-2} , to make the first estimation

for the process gradients. Only two previous points are necessary to estimate the first gradient, regardless of the number of decision variables.

Step 2: At the sample instant k , the process cost function ϕ_p and the constraints $g_{p,i}$ are measured or calculated directly through other measurements. From these measurements, and those taken in the previous sample time $k-1$, the variation terms $\Delta\phi_{p,k}$ and $\Delta g_{p_i,k}$ are computed and the data vector is formed by:

$$\begin{aligned}\Delta\phi_{p,k} &= \phi_{p,k} - \phi_{p,k-1} \\ \Delta g_{p_i,k} &= g_{p_i,k} - g_{p_i,k-1} \quad i = 1, \dots, n_g \\ \Delta \mathbf{u}_{k-1} &= \mathbf{u}_{k-1} - \mathbf{u}_{k-2}\end{aligned}\quad (4.31)$$

Figure 4.3 shows how the time domain is divided into several sample periods, which are the time intervals between two consecutive RTO executions. The yellow points indicate the time instants when the measurement was taken and the blue points the instants when the new inputs obtained by solving the RTO are applied.

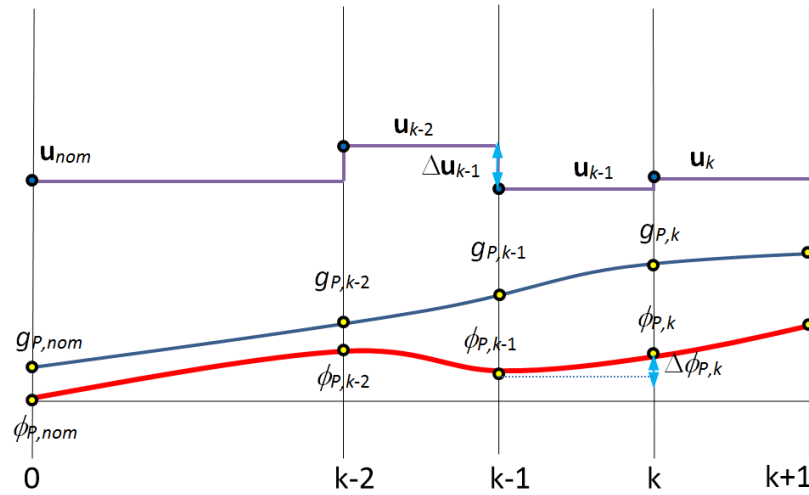


Figure 4.3. Sample periods for the estimation of plant gradients.

Step 3: The estimation error for the variation of the process cost function $e_{\phi,k}$ and the constraints $e_{g_{i,k}}$ is computed as follows:

$$\begin{aligned} e_{\phi,k} &= \Delta\phi_{p,k} - \varphi_{k-1}^T \hat{\theta}_{\phi,k-1} \\ e_{g_i,k} &= \Delta g_{p_i,k} - \varphi_{k-1}^T \hat{\theta}_{g,k-1} \end{aligned} \quad (4.32)$$

Step 4: Estimate the process cost and constraint gradients, $\nabla_{\mathbf{u}}\phi_p|_{\mathbf{u}_{p,k-1}}$ and $\nabla_{\mathbf{u}}g_{p,i}|_{\mathbf{u}_{p,k-1}}$ contained in $\hat{\theta}_{\phi,k}$ and $\hat{\theta}_{g,k}$ using the RELS algorithm described by (4.24)-(4.28).

Step 5: Compute the modifiers λ_k , γ_k and $\varepsilon_{i,k}$ for the RTO problem by applying the following expressions:

$$\begin{aligned} \lambda_k &= \nabla_{\mathbf{u}}\phi_p|_{\mathbf{u}_{p,k-1}} - \nabla_{\mathbf{u}}\phi|_{\mathbf{u}_{k-1}^*} \\ \gamma_{k,i} &= \nabla_{\mathbf{u}}g_{p,i}|_{\mathbf{u}_{p,k-1}} - \nabla_{\mathbf{u}}g_i|_{\mathbf{u}_{k-1}^*} \quad i = 1, \dots, n_g \\ \varepsilon_{i,k} &= g_{p,i}(\mathbf{u}_{p,k-1}) - g_i(\mathbf{u}_{k-1}^*) \end{aligned} \quad (4.33)$$

Step 6: Solve the RTO modified optimization problem (4.34) to obtain \mathbf{u}_k^* , which is applied to the process until the next RTO execution at $k+1$.

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \theta) + \lambda_k^T (\mathbf{u} - \mathbf{u}_{p,k-1}) \\ \text{s.t.} \quad & g_{M,i} = g_i(\mathbf{u}, \theta) + \gamma_{i,k}^T (\mathbf{u} - \mathbf{u}_{p,k-1}) + \varepsilon_{i,k} \leq 0 \quad i = 1, \dots, n_g \\ & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \\ & \kappa^{-1}(\mathbf{S}_k) \geq a \end{aligned} \quad (4.34)$$

Step 7: Check whether the convergence criterion is satisfied, for instance, $|\mathbf{u}_k^* - \mathbf{u}_{k-1}^*| \leq \mathbf{tol}$, that is, there are no changes in the RTO decision variables higher than a fixed tolerance. Other stop criteria can be considered, for example, there are no changes in the process cost function higher than a specified value $|\phi_{p,k}^* - \phi_{p,k-1}^*| \leq \mathbf{tol}$. If any of these criteria is satisfied, stop the algorithm, otherwise, $k = k + 1$ and return to step two when the next sampling time is reached.

4.4.1 BIBO stability of the closed-loop system

The gradients estimated during the transient using RELS will be equal to the steady-state gradients only when the system converges to a steady state. This fact, together with the manipulation of the control set points

during the transient, could make the closed-loop system (controlled plant + RTO) become unstable. However, according to the small gain theorem (SGT) for nonlinear systems (Haddad & Chellaboina, 2008), (Khalil, 1996), the following derivations can be made to ensure closed-loop bounded-input, bounded-output (BIBO) stability. Figure 4.4 shows the basic scheme concerning the interconnection of two systems, in this case, the controlled plant (input \mathbf{u} , output ϕ_p) and the RTO (input ϕ_p , output \mathbf{u}).

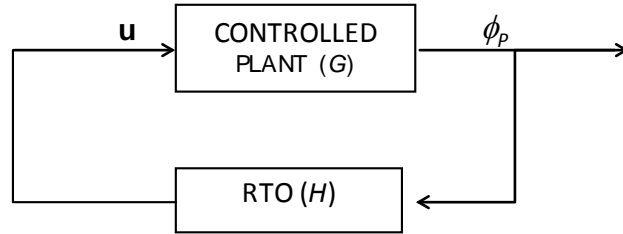


Figure 4.4. Closed-loop setup.

The condition provided by the SGT, to ensure the BIBO stability of the closed-loop system $G \times H$, is given by (4.35) for any norm of the involved signals:

$$\|G\| \|H\| < 1 \quad (4.35)$$

In order to fulfil (4.35) for the induced $L_\infty \rightarrow L_\infty$ norm (denoted by the L_1 system norm below), the following assumptions are made.

Assumption 1:

According to (4.34), \mathbf{u} is assumed bounded, so $\|\mathbf{u}\|_\infty \leq \sigma$, where $\sigma > 0$. Therefore, the modified optimization problem described by H is BIBO stable, that is, $\|H\|_1 < \infty$.

Assumption 2:

The controlled plant, represented by G , is properly designed, stable and well posed, such that the system is BIBO stable with $\|\phi_p\|_\infty \leq \chi \|\mathbf{u}\|_\infty$, for some $1 > \chi > 0$. So, $\|G\|_1 < 1$. Then, two possible scenarios appear:

1. The output of the modified RTO problem, i.e. \mathbf{u} , is not on the constraints. Then, nothing can be concluded about $\|H\|_1$, that is, its value may vary between $(0, \infty)$, so the overall closed-loop system

behaviour may be unstable (ϕ_p increases as \mathbf{u} does). If this worst case happens, then we will enter scenario 2.

2. \mathbf{u} is on the constraints. Then, $\|\mathbf{u}\|_\infty \leq \sigma$ independently of how large ϕ_p is. Therefore, $\|H\|_1 < 1$ applies for large ϕ_p , since $\|\mathbf{u}\|_\infty \leq \xi \|\phi_p\|_\infty$, with $0 < \xi < 1$. In this case, considering Assumption 2, the closed loop system is BIBO stable by the SGT.

Remark 1:

The behaviour of the closed-loop system may lie on the stability limit in the worst case ($\|G\| \cdot \|H\| = 1$), i.e., the system may be continuously jumping between scenario 1 and 2, but without becoming unstable in the BIBO sense.

Remark 2:

Note that assumption 2 about $\|G\|_1 < 1$ is strong, but it can indeed be relaxed to be $\|G\|_1 < \xi^{-1}$ and still fulfil (4.35). Furthermore, the filter parameters and the sampling time of the MA algorithm can act as tuning parameters to smooth the system response, thus reducing $\|H\|_1$.

Nevertheless, it is important to note that the closed-loop operation does not normally present any stability problems, since the upper-level control layers, such as the MPC layer, provides the required system stability. In addition, the first-order modifier filters are applied after filtering to smooth changes in the plant and prevent it from becoming unstable. Note also that the SGT is very conservative (the worst case may appear at a high sampling frequency, worst modelling errors and system conditions which excite the nonlinearities), so many closed-loop systems whose induced $L_\infty \rightarrow L_\infty$ norms do not strictly satisfy (4.35) will, in practice, be stable.

4.4.2 Implementation in the Otto-Williams reactor

The simulated Otto-Williams reactor described in chapter three, section 3.4, is used again to show the performance of the new MA approach based on RELS to estimate the plant gradients. The obtained results have been compared to those obtained by applying traditional static MA, in particular, DMA is applied to this example (see section 2.1.3). In this implementation, the original problem without operational constraints formulated in the

original benchmark (Williams & Otto, 1960) has been considered. Only the limits in the decision variables are considered as the original benchmark was presented. As explained in chapter three, this example presents a high degree of structural plant-model mismatch.

Therefore, the RTO optimization problem which maximizes the operating profit is described by (4.36)

$$\begin{aligned} \max_{\mathbf{u}=[F_B, T_R]} \phi &= F_R (X_P P_P + X_E P_E) - F_A X_A C_A - F_B X_B C_B \\ \text{s.t} & \\ \text{steady - state model with mismatch} & \quad (3.27) - (3.35) \\ F_B^L \leq F_B \leq F_B^U \in T_R^L \leq T_R \leq T_R^U & \end{aligned} \quad (4.36)$$

By applying MA, the modified optimization problem is given by (4.37)

$$\begin{aligned} \max_{\mathbf{u}=[F_B, T_R]} \phi_M &= \phi + \boldsymbol{\lambda}_k^T (\mathbf{u} - \mathbf{u}_{k-1}) \\ \text{s.t} & \\ \text{steady - state model with mismatch} & \quad (3.27) - (3.35) \\ F_B^L \leq F_B \leq F_B^U \in T_R^L \leq T_R \leq T_R^U & \end{aligned} \quad (4.37)$$

where the first-order modifiers $\boldsymbol{\lambda}_k$ are computed as the difference between experimental gradients (obtained from past operating points using DMA and estimated by RELS using the MA approach based on transient information) , and the model gradients as (4.38) shows:

$$\boldsymbol{\lambda}_k^T = \nabla \phi_p|_{\mathbf{u}_{k-1}} - \nabla \phi|_{\mathbf{u}_{k-1}} \quad (4.38)$$

Over the simulated Otto-Williams reactor, a traditional static method, DMA, and the recently developed technique based on RELS, in future TMA, have been applied. Both approaches are implemented with a value of α (lower bound of the dual constraint that appears in (4.34)) and c_1 (lower bound of the persistent excitation constraint (4.29)) equal to 0.10. For DMA, RTO is executed every hour, the time required for the process to achieve a new steady-state. However, TMA has been implemented with an RTO sample time of 600 seconds, which is one sixth of the process stabilization time. The forgetting factor α , considered for the estimation of the process gradients has been 0.90.

The obtained results are represented graphically in Figure 4.5, which shows the evolution of the process cost function and Figure 4.6 and Figure

4.7, which show the evolution of the RTO decision variables as a function of time:

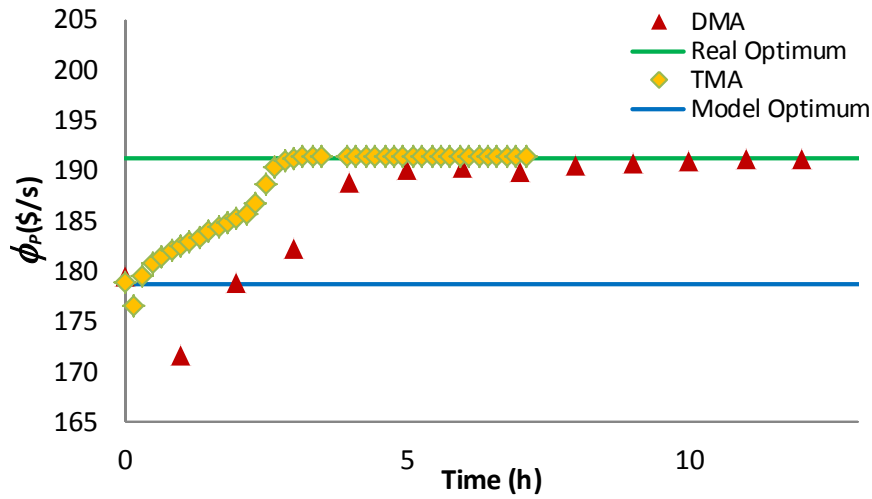


Figure 4.5. Evolution of the process cost function ϕ_p .

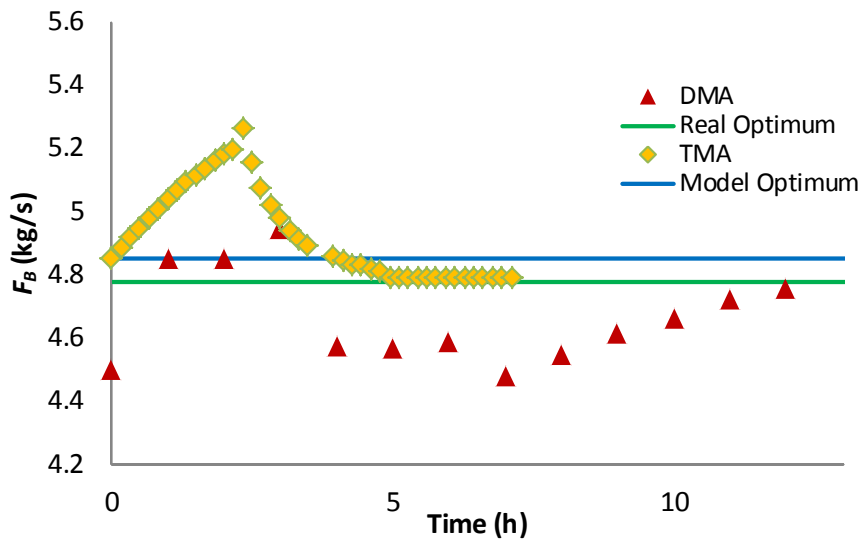


Figure 4.6. Evolution of the decision variable F_B .

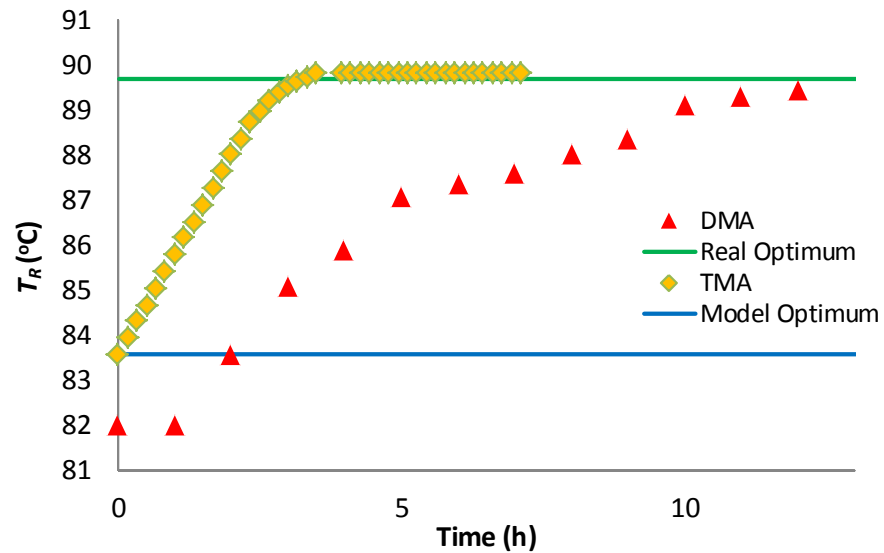


Figure 4.7. Evolution of the decision variable T_R .

For the implementation of DMA, $n_u + 1$ measurements are required to initialize the estimation of the experimental gradients. In this case, we have information of any n_u operating points and then the nominal solution, i.e., the optimal solution for the stationary model with uncertainty is applied at time three in order to have enough information to make the first gradient estimation.

For the identification of plant gradients using RELS, the first operating point is the nominal solution and then, in the second sample period, a change in the input variables is imposed to obtain the measurements at a different operating point. In this way, data from two operating points are obtained and the algorithm has the required information to start the estimation of the gradients at the third sample period during the transient.

Fixing a tolerance band of 0.5% with respect to the optimal value of the cost function, the graphs (in particular, Figure 4.5) show that the optimum of the process is achieved after approximately 46800 seconds using static MA (13 hours). This involves 13 steady states, 11 RTO executions, plus n_u initial steady-states required to estimate the gradients, where n_u is the number of decision variables. For this long period of time, operating conditions and plant-model mismatch could change, resulting in a loss of

optimality since the MA method would not converge to the process optimum.

By applying the new approach, updating the modifiers during the transient and fixing the same tolerance band, the optimum operating point is achieved after 12000 seconds, approximately 3.5 hours, which means a considerable time reduction when compared to the 46800 seconds of the static MA approach. It involves 20 RTO solutions executed during the transient.

A comparison of the performance of the two implemented approaches is shown in Table 4.1.

Table 4.1. Summary Results.

	DMA	TMA
Convergence time (h)	13	3.3
RTO sampling time (s)	3600	600
RTOs executed (#)	11	20

In view of the results, it can be concluded that the new method developed to speed up the convergence of RTO-MA to an operating point that satisfies the plant NCO is able to effectively perform the optimization in the presence of structural plant-model mismatch, considerably reducing, by a factor of 4 (about 25% of the time required for DMA) in this example, the time required to achieve the process optimum as compared with standard static MA techniques, such as DMA.

4.4.3 Implementation in a laboratory-scale flotation column

The MA approach based on the direct estimation of the process gradients using RELS has also been implemented in a laboratory-scale flotation column developed and operated at the department of Ingeniería química y ambiental, at the San Joaquín campus of the Universidad Técnica Federico Santa María (Chile). In this real-life set-up, the obtained results show that the use of transient measurements results in a time saving of around 64% when compared to the application of static MA techniques. The process implementation of the RTO and the experimental results are shown next.

4.4.3.1 Process description

Most of the copper is found in sulphide ores in copper mines, with an average concentration of approximately 0.9% by weight. The copper is obtained from these ores through a process that mainly involves three stages: comminution, concentration and refinement. In the comminution stage, copper is extracted from sulphide ores using size reduction methods. Then, copper minerals are separated from the remaining materials in froth flotation circuits because of the selective hydrophobicity produced by the chemical reagents (collectors). Finally, the copper enriched solution is refined through smelting and electrochemical processes.

The application considered in this thesis focuses on the flotation units used in the concentration stage. The flotation column is a multivariable process whose main control objective is to guarantee the metallurgical yield set for the process operation, expressed by the recovery of the valuable mineral in the concentrate (Navia, et al., 2016). For this task, a supervisory layer is added to the process to find the optimal decision variables.

Figure 4.8 shows a diagram of a flotation column. The flotation column is divided into two zones: the collection zone and the froth or cleaning zone. The pulp feed enters below the froth–pulp interface and descends against a rising group of bubbles generated by the injection of air through a bubbler. Hydrophobic particles of the mineral copper collide and adhere to the bubbles, moving upwards in the collection zone (Finch & Dobby, 1990). At the top, the froth is formed because of the addition of chemical reagents. The froth is stabilized by the addition of wash water passing through a shower. The wash water plays an important role in eliminating fine particles entering from the concentrate. The column has two outflows: concentrate and tail. Concentrate is the stream enriched in floatable minerals (copper sulphides in this case) that goes to the refinement process or to other cleaning stages with the final purpose of being refined. In contrast, the tail is used in scavenger concentration units to recover part of the remaining floatable minerals. The metallurgical yield of the column can be defined by two indicators, recovery and law (4.39). Recovery (R) is the percentage of copper from the feed that is recovered in the concentrate, whereas the law of copper (L) represents its concentration in the concentrate.

$$R = \frac{F_c C_c^{Cu}}{F_f C_f^{Cu}}, \quad L = C_c^{Cu} \quad (4.39)$$

where F_f and F_c are the feed and concentrate flows, respectively, and C_c^{Cu} and C_f^{Cu} are the concentration of copper in the concentrate and the feed streams, respectively.

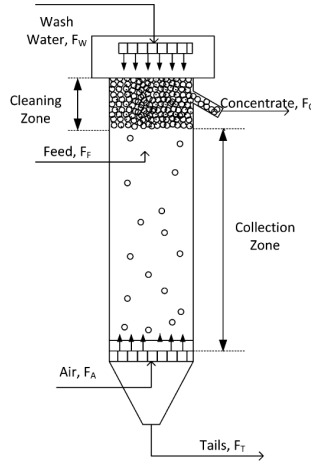


Figure 4.8. Diagram of a flotation column.

The operation of the flotation column has three typical controlled variables: bias, air hold-up and froth depth. Bias (B) is defined as the net flow of water passing through the interface froth-pulp. It is calculated as shown in (4.40)

$$B = F_c - F_f \quad (4.40)$$

The hold-up (ε_G) corresponds to the air fraction inside the column. This variable depends on the air flow, the diameter of the bubbles produced in the bubbler, the pulp density and the concentration of chemical reagents. For an air-water system with a constant density, the hold-up can be estimated as shown in (4.41) (Finch & Dobby, 1990).

$$\varepsilon_G = 1 - \frac{\Delta P}{\rho_{SL} \Delta L} \quad (4.41)$$

where ΔP is the pressure difference across the collection zone, ΔL is the distance between the pressure sensors and ρ_{SL} is the pulp density.

Finally, the third decision variable is the froth depth H_f , that is, the height of the froth measured from the froth-pulp interface to the top of the foam.

The manipulated variables are the wash water, air and tail flows. Each of these flows affects the metallurgical objectives of the system as follows:

- Wash water: it is added to drag the entering hydrophilic gangue to the bottoms to increase L . However, this action also removes part of the floatable mineral due to the mechanical action of water, thus reducing R .
- Air: the addition of air increases R because of the availability of bubbles, but it also produces a decrease in L due to the fluid dynamics in the collection zone.
- Tail: this variable is related to the residence time of both zones in the column. An increase in this flow reduces the volume of the collection zone and increases the froth depth, producing a reduction in R and an increase in L (Navia, et al., 2016).

Other variables that affects the operation of the column are: the characteristics of the feed and the concentration of flotation reagents. However, these variables have not been considered in this work.

4.4.3.2 Experimental set-up

A supervisory layer to optimize the process operation has been implemented in the laboratory-scale flotation column shown in Figure 4.9. The column is made up of an acrylic cylinder with an internal diameter of 9.2 cm and a height of 3.27 m, with a system to collect the froth at the top by overflow. The feed and wash water are pumped using peristaltic pumps, whereas the compressed air is injected at the bottom of the column using a porous diffuser. Both the air and tail flows are manipulated using globe valves. The column has two differential pressure sensors, one located in the collection zone, a volumetric flowmeter for the tail, and a mass flowmeter for the air. The P&ID described in Figure 4.9 shows the control loops that follow the typical pairing for this type of unit: B -wash water (FYC), ε_G -air (DYC) and H_f -tail (LYC). The optimal decisions made in the supervisory layer are passed to the regulatory layer that consists of PI controllers. For simplicity, the experimental set-up follows a hybrid approach (Bergh, 2007), (Bergh, 2012). The main concept behind this approach is that the essential phenomena of a process can be divided into two aspects: hydrodynamics and physicochemical mechanisms.

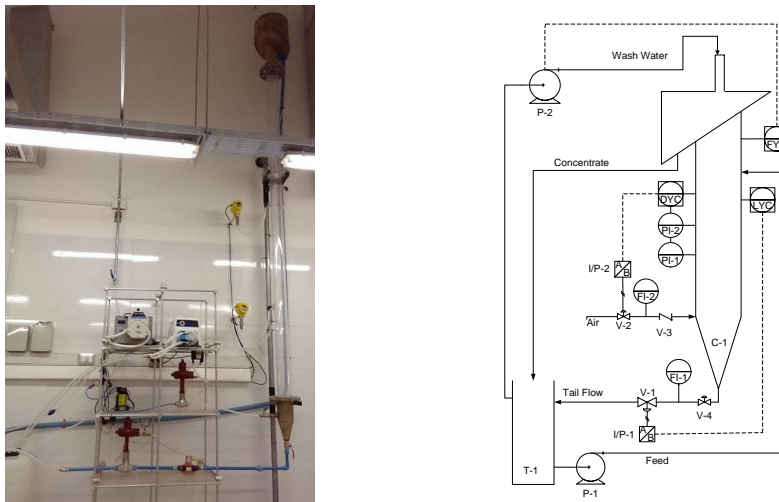


Figure 4.9. Experimental set-up and P&ID of the column.

The hydrodynamics of the flotation column can be represented in a proper manner for a laboratory-scale unit considering an air–water system (Bergh, 2012). However, the physicochemical mechanisms involved in the flotation process require the use of expensive instrumentation, such as sensors or actuators, and also high maintenance costs, making the experimentation infeasible. For this reason, a hybrid experimental set-up was developed that, in addition to the physical components above mentioned, includes a virtual part with a first-principles model that has been developed to emulate the phenomenology of the flotation of copper sulphur minerals. The system takes into account two types of degrees of freedom: physical and virtual. The physical degrees of freedom are the variables measured from the experimental air–water set-up and the design parameters of the column, whereas the virtual ones are the properties of an emulated feed pulp, that is, law of copper, slurry density, solid percentage and particle size, together with the kinetic relations necessary to calculate the recovery and law of copper (Bergh, 2007), (Bergh, 2012). Figure 4.9 shows that the tail flow is produced due to the pulp level, as there is no pump in the outflow of the bottom. The dynamic of the discharge is in the order of minutes. Gas enters at the bottom and is discharged to the atmosphere at the top of the column with a dynamic on the order of a few seconds. This means that the dominant time constant is related to H_f and the tail flow. Because the tail flow affects the net flow of water, the bias is also affected. Therefore, the dynamic of B also has important effects on the transient, as the control loop modifies the wash water, thus affecting H_f .

This interaction can produce important drawbacks in a system with an RTO-based supervisory layer related to the time needed to reach the next steady state. For this reason, it would be desirable to reduce the number of steady states needed to find the NCO of the process. For this task, MA based on transient information has been applied to speed up convergence to the optimum.

4.4.3.3 Metallurgical model

A first-principles model has been implemented to represent the phenomenology of flotation (Bergh, et al., 1998). This model considers that flotation can be described as an elementary reaction between the floatable particles and the bubbles. Considering that the concentration of bubbles should be larger than the mineral, the kinetic rate of flotation can be approximated with a pseudo-first-order expression (Polat & Chande, 2000). The recovery in the cleaning (R_F) and collection zones (R_{Ci}) is given by (4.42) and (4.43), respectively. In these equations, N_d is the vessel dispersion number, τ_s is the residence time of the solid which is a function of J_b , defined as the superficial velocity of the bias. In the same manner, J_g and J_w are the superficial velocities of the air and wash water, respectively. k_{Ci} represents the kinetic expression of species i .

$$R_F = 0.95 \exp\left(\frac{-0.0144 H_f (1 + 3 \cdot J_w)}{J_g^3}\right) \quad (4.42)$$

$$R_{Ci} = 1 - \frac{4a_i \exp\left(\frac{1}{2N_d}\right)}{(1+a_i)^2 \exp\left(\frac{a_i}{2N_d}\right) - (1+a_i)^2 \exp\left(\frac{-a_i}{2N_d}\right)} \quad (4.43)$$

where the constant a_i is computed as follows:

$$a_i = \sqrt{1 + (4k_{Ci} \tau_s (J_b) 60N_d)} \quad (4.44)$$

The kinetic expressions that represent the probability of reactant collisions, in this case air flow and floatable particles with a diameter D_p equal to 40 μm , are described by (4.45), where the parameters A_1 and A_2 have been estimated using operational data from a column located in the cleaning zone of a concentration circuit (Bergh, et al., 1998).

$$\begin{aligned} k_{C_1} &= A_1 J_g^{0.75} (0.5 + 1.5D_p + 1.1D_p^2) \\ k_{C_2} &= A_2 J_g^{0.75} (2.5 + 5D_p + 6D_p^2) \end{aligned} \quad (4.45)$$

4.4.3.4 RTO model and optimization problem

The uncertainty in the model-based optimization has two sources. One is the experimental uncertainty related to the precision of the instruments and the behaviour of the PI controllers. Another is the plant-model mismatch due to the use of a simplified and uncertain RTO model.

To emulate this modelling mismatch, the dependencies from (4.45) have been neglected in the model-based optimization, assuming that k_{Ci} is constant.

The model-based optimization problem, solved in the RTO layer, is formulated by (4.46), where the decision variables are H_f , ε_G and B , that is, $\mathbf{u}^T = [H_f, \varepsilon_G, B]$.

$$\begin{aligned} \min_{\mathbf{u}=[H_f, \varepsilon_G, B]} \phi &= C_E + C_L + C_R - I \\ \text{s.t.} &: \text{first - principles stationary model} \\ \mathbf{g}^L &\leq \mathbf{g} \leq \mathbf{g}^U, \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \end{aligned} \quad (4.46)$$

The economic objective function has four terms: C_E is the energy cost of pumping the slurry and compressing the air, C_L is the cost of the metallurgical losses in the scavenger stage (proportional to the copper that cannot be recuperated in this stage), C_R is the cost associated to the regrinding process, and I represents the incomes of the process related to the specific selling price of concentrate and the amount produced as shown next.

$$I = \left(\frac{\rho_{sl} F_F x_F R}{x_C} \right) P_C(x_C) \quad (4.47)$$

where ρ_{sl} is the density of the pulp, x_F is the w/w percentage of copper in the feed, x_C is the w/w percentage of copper in the concentrate, R is the total recovery of copper, and P_C the specific selling price of concentrate, which depends on the copper concentration.

The operational constraints \mathbf{g} are related to metallurgical objectives and limits imposed by the experimental set-up, therefore, $\mathbf{g}^T = [L, R, J_G, J_w, J_b]$.

The limits for the decision variables and the constraints are shown in Table 4.2.

Table 4.2. Lower and upper bounds for the optimization problem.

Variable	Lower bound	Upper bound
L (%w/w)	25	35
R (%)	70	100
J_G (cm/s)	1	3.5
J_W (cm/s)	0.04	0.40
J_b (cm/s)	0	0.28
H_f (cm)	80	120
ε_G (%)	0.07	0.15
B (l/min)	0	0.8

The described plant-model mismatch makes that the process optimum does not match the model optimum, as can be seen in Table 4.3.

Table 4.3. Model and process optimum.

Optimum	J^* (USD/h)	H_f^* (cm)	ε_G^* (%)	B (l/min)
Process	2.74	80	11.3	0.8
Model	3.19	80	9.24	0.8

The modified optimization problem, which is solved by applying MA is described by (4.48), where ϕ_M is the modified cost function and g_M^L and g_M^U are the modified constraints. Taking into account the fact that the number of decision variables is two (since it has been supposed that B is not updated by the RTO layer) and the number of constraints is five, the size of the modifier vectors is $\lambda_k^T \in \mathfrak{R}^2$, $\mathbf{v}_{k,g}^T \in \mathfrak{R}^{2 \times 5}$ and $\boldsymbol{\varepsilon}_{k,g} \in \mathfrak{R}^5$.

$$\begin{aligned}
 \min_{\mathbf{u}} \phi_M &= \phi + \boldsymbol{\lambda}_k^T (\mathbf{u} - \mathbf{u}_k) \\
 \text{s.t. : Uncertain} \quad &\text{first-principles} \quad \text{model} \\
 \mathbf{g}_M^L &= \mathbf{g}^L + \mathbf{v}_{k,g}^T (\mathbf{u} - \mathbf{u}_k) + \boldsymbol{\varepsilon}_{k,g} \leq 0 \\
 \mathbf{g}_M^U &= \mathbf{g}^U + \mathbf{v}_{k,g}^T (\mathbf{u} - \mathbf{u}_k) + \boldsymbol{\varepsilon}_{k,g} \leq 0 \\
 \mathbf{u}^L &\leq \mathbf{u} \leq \mathbf{u}^U
 \end{aligned} \tag{4.48}$$

4.4.3.5 Results

Starting from the points summarized in Table 4.4, the behaviour of TMA (Navia, et al., 2017), (Rodríguez-Blanco, et al., 2017) has been compared with the DMA (Brdyś & Tatjewski, 1994). Because of the fact that the flotation column presents important controllability problems, the decision variable B has not been updated by the supervisory layer (Navia, et al., 2016).

The RTO sample time is about 4 minutes for the implementation of DMA, the time required for the process to achieve a new steady-state and 36 seconds when TMA is applied.

Figure 4.10 shows three experiments with the evolution of the decision variables using TMA, while Figure 4.11 presents the same, but using the DMA. In each figure, the dashed lines are the set-points of the controllers proposed by the supervisory layer (u -SP), the evolution of the manipulated variables (u) are represented by solid lines. The optimum of the process and the model (u_p^* and u_m^*) and the limits for the decision variables (u^U and u^L) are shown with horizontal grey dashed lines. The evolution from the three points is represented with the following colours: black lines start from point 1, blue lines from point 2 and red lines from point 3.

Table 4.4. Starting Points.

No. of Starting Point	H_f (cm)	ϵ_G (°/1)	B (l/min)
1	80	0.0924	0.8
2	80	0.07	0.8
3	90	0.07	0.8

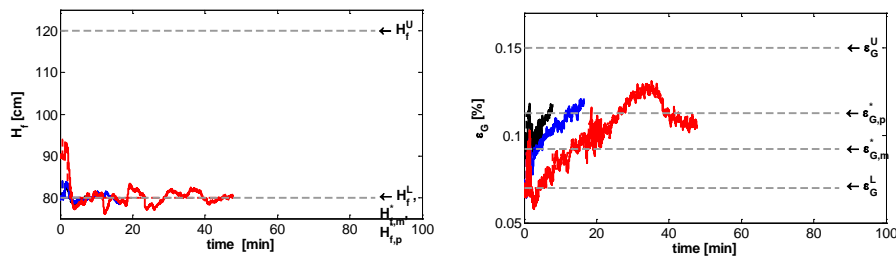


Figure 4.10. Evolution of the decision variables using TMA.

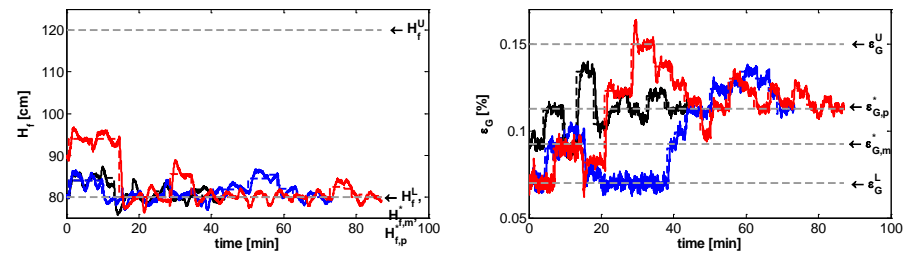


Figure 4.11. Evolution of the decision variables using DMA.

From Figure 4.10 and Figure 4.11, it can be seen that the use of MA allows a point to be found that satisfies the NCO of the flotation column, despite the structural uncertainty associated with the kinetic model of the flotation phenomenon and the ability of the regulatory layer to reach the set point of the process. The time needed to reach the desired point depends on the starting point for both TMA and DMA methodologies. However, TMA reduced this time by 82%, starting from point 1, 77.5% from point 2 and 45.2% from point 3. Considering that the time needed to reach the steady state of this laboratory equipment is about 4 min, and the average time to reach the NCO of the process with DMA was 65 min, it can be said that on average, TMA allowed approximately 10 steady states to be saved. As this process interacts with side units and the quality of the feed changes periodically, this reduction of time can be seen as an improvement from the point of view of the applicability of MA in real processes.

4.5 Conclusions

In this chapter, different MA approaches based on transient information have been presented. It is important to highlight the description of a new method that supposes one of the main contributions of this thesis. The mentioned technique obtains the process gradients directly from truncated Taylor expansions of the process cost and gradients combined with adaptive filtering estimation techniques.

The method has been tested in the simulated Otto-Williams and the results obtained show that it is possible to effectively perform the optimization of the operation of the reactor in the presence of structural plant-model mismatch. This reduces by a factor of 4 the time required to achieve the process optimum as compared with traditional static MA techniques, such as DMA.

In addition, this approach has been tested in a laboratory-scale flotation column, reducing the time needed to detect the NCO of a real process, converging to the optimum in only one steady-state, whereas 10 steady-states are used by the classic MA. The implementation of this technique in other case studies will be shown in the following chapter.

To conclude it can be said that the combination of the modifier adaptation methodology, the least squares algorithm using transient measurements to compute process gradients, and the use of simplified models in the RTO layer, is a really powerful approach to achieve the optimal operating point of processes, reducing the convergence time.

5 CASE STUDY I: DEPROPANIZER DISTILLATION COLUMN

The performance of the proposed MA schemes will be illustrated through the case study of an industrial depropanizer distillation column found in petrol refineries, and the results will be compared to those obtained with standard NMA and DMA. This example is an industrial example, very realistic and with a high complexity. The successful implementation of MA in large-scale simulated processes like this, shows that MA is a powerful tool that can be implemented in real plants. We use both a rigorous model to simulate the column and a simplified model to implement RTO, that is, there is considerable plant-model mismatch. The case study also considers the industrially relevant situation where the decision variables in the RTO problem do not correspond to those in the plant. This often results from the presence of an intermediate MPC layer that is not considered in the RTO problem formulation.

5.1 Modelling and simulation

The case study considers the optimal operation of a continuous depropanizer distillation column of a petrol refinery that is used to separate propane (C_3H_8) from a mixture containing butane (C_4H_{10}) and other components. In general terms, distillation columns use differences in temperature and pressure conditions along the column to get the more volatile components at the top of the column, propane in this case, and the less volatile component at the bottom of the column.

The main equipment where separation occurs is the distillation column and reboiler. The liquid feed usually enters the distillation column after preheating. The liquid flows to the reboiler where steam is used as the heating medium. The reboiler provides the initial vaporisation of the liquid phase to create the vapour phase. Part of the liquid in the reboiler is vaporised and the vapour is returned to the distillation column. The remaining liquid leaves the column as bottoms and provides the heat source to preheat the feed.

The vapour flows up the column and leaves as overhead vapour. The overhead vapour is condensed in the condenser and collected in the accumulator.

Part of the liquid from the overhead accumulator is withdrawn as the distillate, and the rest is returned to the column as reflux liquid. The reflux liquid flows down the column, combining with the feed stream in the column, and the combined liquid continues to flow down the column. Normally, the section of the column above the feed entry is known as the rectifying section, and the section below the feed entry is known as the stripping section.

Inside the column, the down-flowing liquid comes into contact with the up-flowing vapour. The distillation column is equipped with trays and/or packings, to facilitate the vapour-liquid contact. During this contact, the vapour is enriched in the more volatile component and stripped of less volatile components.

The control objective for the distillation column is to maintain the composition of propane in the distillate stream within the desired specifications, satisfying such operational constraints as maximum pressure drop to prevent the column from flooding, or limits on head and bottom temperature (Acedo Sánchez, 2003).

Basic control includes PID loops to maintain the liquid levels in the accumulator and the bottom of the column at their desired values. There are also controllers to regulate the steam and reflux flows and the head pressure. The steam flow controller manipulates the opening of the valve located in the steam input, whereas the reflux flow controller manipulates the valve located in the reflux input. In the considered case study, the level in the accumulator is controlled by manipulating the distillate flow, whereas the level at the bottom of the column is controlled by manipulating the bottom flow. The pressure at the top of the column is controlled by changing the outlet flow of the condenser.

Control of head and bottom composition can be difficult, both because of a lack of reliable measurements and the multivariable nature of the process. Because of this, in many refineries, MPC controllers are used to control column temperatures instead of composition. In the same way, in our case study, a DMC controller, which manipulates the references for the steam and reflux flow controllers, tries to maintain head and bottom temperatures close to their set points, as an indirect way of controlling distillate and bottom compositions. An upper RTO layer improves the economic performance of the column computing optimal set points for the DMC, while maintaining the composition of propane in the distillate within the desired specification. This control structure is shown in Figure 5.1, where it can be observed that the decision variables of the RTO layer are the head and bottom temperatures that are passed as set points to the DMC controller.

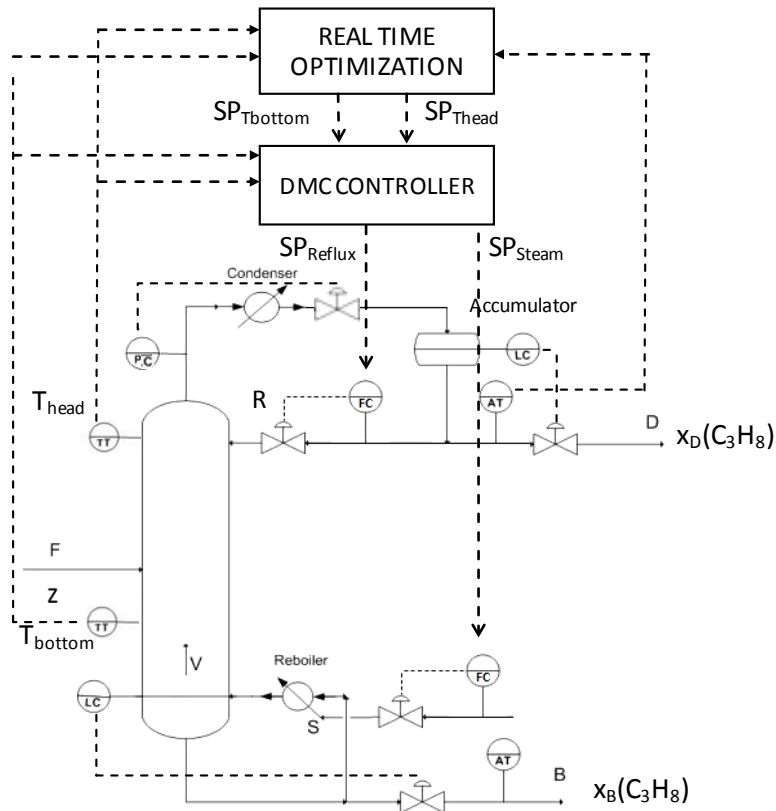


Figure 5.1. Control structure of a depropanizer distillation column.

The simulated depropanizer column is based on an industrial example located in the Repsol Tarragona Refinery (Spain). The column is made up of a total condenser, a top accumulator, a partial reboiler and 37 equilibrium stages or trays. In the described case study, the feed mixture enters at stage 19 with a flow rate of 468 kmol/h and 330.42 K. The composition of the feed is 45.55 mol% propane, 44.67 mol% butane and 9.77 mol% ethane. The tray efficiency is 60%. A rigorous first-principles model was developed to represent the process, the main equations of which are described next.



Figure 5.2. Typical refinery distillation column.

5.1.1. Dynamic rigorous model

The dynamic rigorous model of the distillation column, developed to simulate the process, is based on material and energy balances on each tray and vapour-liquid equilibrium equations that define the mass transfer between liquid and vapour stream on each tray (Rueda, 2015). The reboiler and the condenser of the column are also modelled.

Material and energy balances:

The mass balances for each tray read (see schematic in Figure 5.3 and nomenclature in Table 5.2):

$$\frac{dmol_i}{dt} = l_{i+1} + v_{i-1} - l_i - v_i \quad i = 2, \dots, n-1 \quad (5.1)$$

$$\frac{dmol_{n_feed}}{dt} = f + l_{n_feed+1} + v_{n_feed-1} - l_{n_feed} - v_{n_feed} \quad (5.2)$$

$$\frac{dmol_n}{dt} = l_{ref} + v_{n-1} - l_n - v_n \quad (5.3)$$

$$\frac{dmol_i}{dt} = l_2 + v_1 - b \quad (5.4)$$

$$\frac{dmol_{accum}}{dt} = l_{accum} - l_{ref} - d - l_{overflow} \quad (5.5)$$

where $dmol/dt$ is the molar accumulation rate (kmol/h), i is the tray number, n is the total number of trays, l , v , f stand for liquid, vapour and feed molar flows respectively (kmol/h), l_{ref} is the molar reflux flow (kmol/h), and b and d are the molar flow of bottom and distillate streams, also in kmol/h. Equation (5.1) presents the overall mass balance around the i^{th} tray, shown schematically in Figure 5.3, where the involved terms are the liquid and the vapour leaving the tray i , l_i and v_i , the liquid flowing down from the upper tray, l_{i+1} , and the vapour flowing up from the below tray, v_{i-1} . Equations (5.2), (5.3), (5.4) and (5.5) show the mass balances in the feed tray (Tray n_{feed}), the top of the column (Tray n), the bottom (Tray 1) and the top accumulator. In (5.5), l_{accum} is the flow that comes from the condenser and $l_{overflow}$ is the excess liquid that overflows from the accumulator (Rodríguez-Blanco, et al., 2015), (Coulson & Richardson, 2002):

The mass flow rates (in kg/h) of the different streams (reflux R , distillate D , bottom B and feed F) can be obtained from the molar flow rates as follows, where P_m (kg/kmol) indicates the molecular weight of each stream, computed from the stream composition $x_{j,i}$ and the individual molecular weight of each component $M_{m,j}$ as (5.6) shows:

$$P_{m,i} = \sum_{j=1}^3 x_{j,i} M_{m,j} \quad (5.6)$$

$$R = l_{ref} P_{m,l_{ref}} \quad (5.7)$$

$$D = d P_{m,d} \quad (5.8)$$

$$B = b P_{m,b} \quad (5.9)$$

$$F = f P_{m,f} \quad (5.10)$$

The component mass balances are expressed by (5.11)-(5.15). These balances are defined only for the two components with less composition, in this case, butane and ethane, and the majority one, propane, is obtained by (5.15)

$$\frac{dx_{j,i}}{dt} = l_{i+1}x_{j,i+1} + v_{i-1}y_{j,i-1} - l_i x_{j,i} - v_i y_{j,i} \quad i = 2, \dots, n-1 \quad (5.11)$$

$$\frac{dx_{j,n_feed}}{dt} = fz_j + l_{n_feed+1}x_{j,n_feed+1} + v_{n_feed-1}y_{j,n_feed-1} - l_{n_feed}x_{j,n_feed} - v_{n_feed}y_{j,n_feed} \quad (5.12)$$

$$\frac{dx_{j,1}}{dt} = l_2x_{j,2} + v_1y_{j,1} - bx_{j,B} \quad (5.13)$$

$$\frac{dx_{j,accum}}{dt} = l_{accum}x_{j,accum} - l_{ref}x_{j,D} - dx_{j,D} - l_{overflow}x_{j,D} \quad (5.14)$$

$$\sum_{j=1}^3 z_j = 1 \quad \sum_{j=1}^3 x_{j,i} = 1 \quad \sum_{j=1}^3 x_{j,D} = 1 \quad (5.15)$$

$$\sum_{j=1}^3 x_{j,B} = 1 \quad \sum_{j=1}^3 x_{j,accum} = 1 \quad \sum_{j=1}^3 y_{j,i} = 1 \quad i = 1, \dots, n$$

where $x_{j,i}$ and $y_{j,i}$ are the composition of component j (butane, propane and ethane) in the liquid and vapour streams through the i^{th} tray ($0/1$), z_j is the feed composition, $x_{j,D}$ is the composition of the distillate stream and $x_{j,B}$ the composition of the bottom stream.

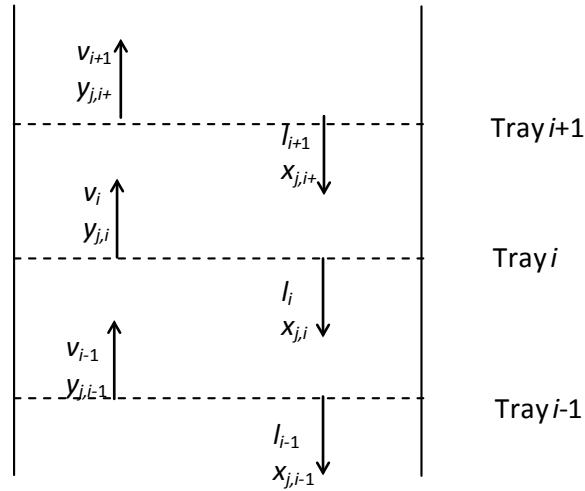


Figure 5.3. Schematic of material exchange on trays.

Due to their fast dynamics, the energy balances around the i^{th} tray can be modelled as steady-state equations:

$$H_{v,i}v_i = h_{l,i+1}l_{i+1} + H_{v,i-1}v_{i-1} - h_{l,i}l_i \quad i = 2, \dots, n-1 \quad (5.16)$$

$$H_{v,n_feed}v_{n_feed} = h_f f + h_{l,n_feed+1}l_{n_feed+1} + H_{v,n_feed-1}v_{n_feed-1} - h_{l,n_feed}l_{n_feed} \quad (5.17)$$

$$H_{v,i} = \sum_j y_{j,i} h_j(T_i) \quad i = 1, \dots, n \quad (5.18)$$

$$h_{l,i} = \sum_j x_{j,i} h_j(T_i) \quad i = 1, \dots, n \quad (5.19)$$

where H_v is the vapour enthalpy (kJ/kg), h_l is the liquid enthalpy (kJ/kg), and h_j is the specific enthalpy of each component (kJ/kg) that depends on the tray temperature T_i (K), since it is computed as the product of the specific heat at the constant pressure of the component $C_{p,j}$ (kJ/kg K) and the temperature T_i .

The key temperatures in the operation of the distillation columns are the head temperature (T_{head}) and the bottom temperature (T_{bottom}). These temperatures correspond to the sensitive trays of the column where the sensors are located. In this case, the sensitive trays are the number 34 for T_{head} and 4 for T_{bottom} .

Vapour-liquid equilibrium:

The concentration of vapour in contact with liquid at equilibrium $y_{eq,j,i}$ in each tray is expressed by Raoult's law (5.20).

$$y_{eq,j,i} = E_f K_{j,i} X_{j,i} \quad i = 1, \dots, n \quad (5.20)$$

$$K_{j,i} = \frac{P_{sat,j,i}}{P_i} \quad i = 1, \dots, n \quad (5.21)$$

$$P_{sat,j,i} = 10^{A_j - \frac{B_j}{T_i + C_j}} \quad i = 1, \dots, n \quad (5.22)$$

where the equilibrium constant $K_{j,i}$ is given by the ratio between the component j vapour pressure $P_{sat,j}$ (Pa) (computed by the Antoine equation (5.22) that includes specific constant values A , B , C for each component j) and the total pressure in the i^{th} tray P_i (Pa). The parameter E_f indicates the tray efficiency.

Along the column, there is a pressure drop that is a function of the vapour stream passing through the tray and the pressure in the upper tray. c is a constant value to compute the pressure drop through the column. Pressure at the top of the column P_n is controlled by a PID which manipulates the outlet flow of the condenser.

$$P_i = P_{i+1} + (v_i / c)^2 \quad (5.23)$$

Energy balance of the reboiler:

A static energy balance is considered in the reboiler:

$$H_{v,1} v_1 = Q_b - h_{l,1} b \quad (5.24)$$

$$Q_b = S \Delta h(P_{reboiler}) \quad (5.25)$$

where Q_b (kJ/h) is the heat flow added to the reboiler, S (kg/h) the steam flow and Δh the latent heat of vaporization (kJ/kg) that depends on the reboiler pressure $P_{reboiler}$ (bar).

Energy balance of the condenser:

The steady-state energy balance corresponding to the condenser is expressed by the equation (5.26).

$$Q_c = v_n [H_{v,n}(T_{head}) - h_{l,n}(T_{head})] \quad (5.26)$$

where Q_c is the heat removed in the condenser (kJ/h), which is computed next:

$$Q_c = F_w C_{p,w} (T_{out,w} - T_{in,w}) \quad (5.27)$$

$$T_{out,w} = \frac{T_n - (T_n - T_{in,w})}{e^{(U_{cond}(1-\alpha_{cond})A_{cond}(F_w C_{p,w}))}} \quad (5.28)$$

where F_w (kg/s) is the flow rate of cooling water, $T_{out,w}$ ($^{\circ}$ C) and $T_{in,w}$ ($^{\circ}$ C) are the outlet and inlet water temperature respectively, A_{cond} (m^2) is the heat exchange area of the condenser, U_{cond} (kJ/h m^2 $^{\circ}$ C) is the overall heat transfer coefficient, α_{cond} $^{\circ}$ /1 of A_{cond} that is being used and $C_{p,w}$ (kJ/kg $^{\circ}$ C) is the specific heat capacity for water.

The nonlinear dynamic model described above has been simulated using the software EcosimPro (EAInt, 2013), a modern object oriented simulation environment. The model consists of 2145 equations (129 differential equations and 2016 algebraic ones) and 2152 variables (1976 explicit, 129 derivative, 40 algebraic and 7 boundary variables).

Table 5.1. Boundary variables in the dynamic model.

Variable	Meaning	Units
F	Feed molar flow rate	kg/h
SP_{Thead}	Set point for the head temperature	K
$SP_{Tbottom}$	Set point for the bottom temperature	K
SP_{Pn}	Set point for the head pressure	K
$Z_{C_2H_6}$	Molar fraction of C_2H_6 in the feed stream	$^{\circ}$ /1
$Z_{C_4H_{10}}$	Molar fraction of C_4H_{10} in the feed stream	$^{\circ}$ /1
T_{feed}	Feed temperature	$^{\circ}$ C

Table 5.2. Nomenclature used in the dynamic model.

Variable	Meaning	Units
l	Liquid molar flow rate	kmol/h
v	Vapour molar flow rate	kmol/h
f	Feed molar flow rate	kmol/h
$dmol/dt$	Molar accumulation rate	kmol/h
l_{ref}	Reflux molar flow rate	kmol/h
l_{acum}	Condenser outlet molar flow rate	kmol/h
$l_{overflow}$	Liquid molar flow rate overflowing from the accumulator	kmol/h
b	Bottom molar flow rate	kmol/h
d	Distillate molar flow rate	kmol/h
R	Reflux mass flow rate	kg/h
D	Distillate mass flow rate	kg/h
B	Bottom mass flow rate	kg/h
F	Feed mass flow rate	kg/h
S	Steam flow to the reboiler	kg/h
P_m	Molecular weight	kg/kmol
$M_{m,j}$	Molecular weight of each component	kg/kmol
$x_{j,i}$	Molar fraction of component j in the liquid stream leaving the tray i	$^{\circ}/1$
z_j	Molar fraction of component j in the feed stream	$^{\circ}/1$
$y_{j,i}$	Molar fraction of component j in the vapour stream leaving the tray i	$^{\circ}/1$
$y_{eq,j,i}$	Equilibrium concentration	$^{\circ}/1$
$K_{j,i}$	Equilibrium constant of component j in the tray i	
H_v	Vapour enthalpy	kJ/kg
h_l	Liquid enthalpy	kJ/kg
h_j	Specific enthalpy of component j	kJ/kg
T_i	Tray temperature	K
n_{feed}	Feed tray	#
T_{head}	Head temperature	K
T_{bottom}	Bottom temperature	K
$P_{sat,j}$	Vapour pressure of component j	Pa
P_i	Tray pressure	Pa
E_f	Tray efficiency	$^{\circ}/1$
Q_b	Heat flow added to the reboiler	kJ/h
Q_c	Heat removed in the condenser	kJ/h
Δh	Latent heat of vaporization	kJ/kg
$P_{reboiler}$	Reboiler pressure	bar
F_w	Flow rate of cooling water	kg/s
$C_{p,w}$	Specific heat capacity for water	kJ/kg $^{\circ}C$
$T_{out,w}$	Outlet water temperature	$^{\circ}C$
$T_{in,w}$	Inlet water temperature	$^{\circ}C$
U_{cond}	Overall heat transfer coefficient	kJ/h m^2 $^{\circ}C$
A_{cond}	Heat exchange area of the condenser	m^2
α_{cond}	Portion of condenser area used	$^{\circ}/1$

5.1.2. Steady-state rigorous model

While the previous dynamic model will be used to represent the real process, the RTO layer uses a steady-state model to make decisions. A rigorous steady-state model may be developed directly by removing derivatives from the dynamic column described above, and removing the auxiliary equipment that does not affect the optimization problem, such as the energy and mass balances at the accumulator (at steady-state, the inlet flow is the same as the outlet flow and the same happens with the temperature).

However, the resulting model has a large number of variables, equations and algebraic loops that make the initialization and convergence of the optimization difficult. In addition, this complexity makes the optimization problem solved in the RTO run slowly (see Table 5.4 and Table 5.7). The simulation of this model is shown in section 5.1.4.

For this reason, a simplified steady-state model has been developed specifically in this thesis to be used at the RTO layer. This reduced model is described next.

5.1.3. Steady-state simplified model

The simplified model takes into account the fact that distillation columns can be divided into three sections: namely, the rectifying or enriching section, the stripping section and the feeding tray, as shown in Figure 5.4.

The feeding tray is the tray where the feed is introduced. The rectifying section consists of the trays above the feed tray and is so named because, in this section, rising vapours are enriched in the more volatile component. The section below the feeding tray is called the stripping section because liquid is stripped off the more volatile component there.

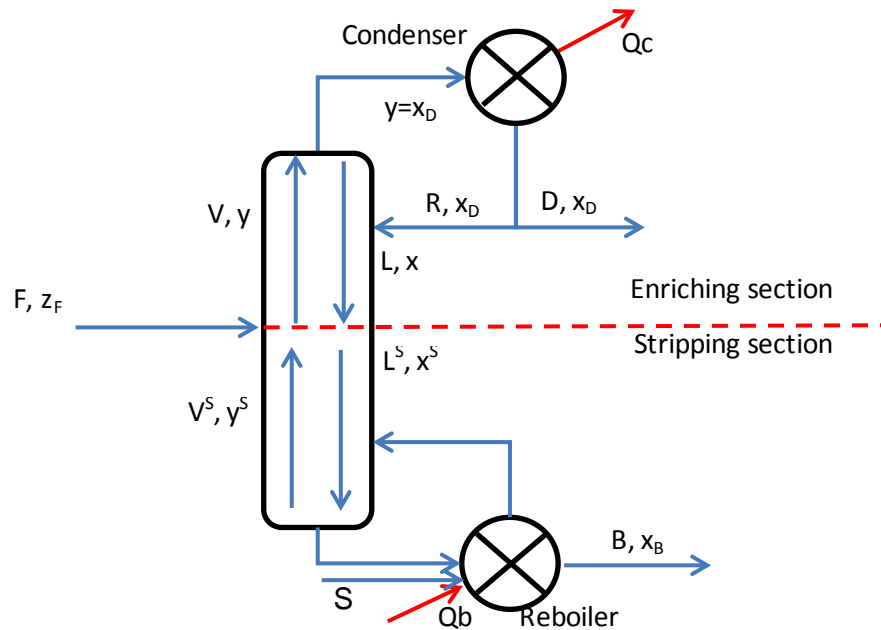


Figure 5.4. Schematic diagram of the three sections of a continuous distillation column used in the steady-state simplified model.

In order to obtain a fairly small reduced model, the rigorous column model has been simplified by considering only three trays and a global efficiency. The equations that describe the simplified steady-state model for RTO are presented next.

The overall mass balance of a distillation column is given by (5.29)

$$f = d + b \quad (5.29)$$

where f , d and b are the feed, distillate and bottom molar flow rate (kmol/h); whereas the mass balances on the enriching and stripping sections are given by (5.30) and (5.31) respectively

$$v = l_{ref} + d \quad (5.30)$$

$$l^S = v^S + b \quad (5.31)$$

where v and v^S are the vapour molar flow rate in the enriching and stripping sections (kmol/h), l^S is the liquid molar flow rate in the stripping section (kmol/h), and l_{ref} is the reflux molar flow rate (kmol/h).

The mass flow rates (in kg/h) of the different streams (reflux R , distillate D , bottom B and feed F) can be obtained from the molar flow rates as shown in (5.6)-(5.10).

The component mass balances are expressed by (5.32)-(5.34)

$$f x_{j,n_feed} = d x_{D,j} + b x_{B,j} \quad (5.32)$$

$$l^s x_j^s = v^s y_j^s + b x_{B,j} \quad (5.33)$$

$$y_j = x_{D,j} \quad (5.34)$$

$$\sum_{j=1}^3 x_{j,n_feed} = 1 \quad \sum_{j=1}^3 x_j^s = 1 \quad \sum_{j=1}^3 y_j^s = 1 \quad (5.35)$$

$$\sum_{j=1}^3 x_j = 1 \quad \sum_{j=1}^3 y_j = 1 \quad \sum_{j=1}^3 x_{B,j} = 1$$

where x_j^s and y_j^s are the composition of component j (butane, propane and ethane) in the liquid and vapour streams through the stripping section, and x_j and y_j are the same compositions for the enriching section. These equations are applied for two components, in this case butane and ethane, which are those present in lower concentrations; while the composition of propane is obtained by (5.35). The feed enters the column with a composition of x_{j,n_feed} and distillate and bottom streams leave the column with a composition equal to $x_{D,j}$ and $x_{B,j}$ respectively. Assumption (5.34) is valid since there is a total condenser, so the vapour stream leaving the column is totally condensed and there is no accumulator considered in this model, so the composition of the reflux stream is equal to the composition of the vapour.

The energy balances are expressed by (5.36)-(5.38)

$$Q_c = v(H_D(T_{head}) - h_D(T_{head})) \quad (5.36)$$

$$Q_b = S \Delta h(P_{reboiler}) \quad (5.37)$$

$$l^s h^s(T_{bottom}) + Q_b = v H^s(T_{bottom}) + b h_B(T_{bottom}) \quad (5.38)$$

where Q_c is the heat removed in the condenser (kJ/h), Q_b is the heat flow added to the reboiler (kJ/h), Δh is the latent heat of vaporization (kJ/kg) that depends on the reboiler pressure $P_{reboiler}$ (bar), H^s and h^s are the enthalpies for liquid and vapour streams in the stripping section (kJ/kmol), h_B is the enthalpy of the bottom stream (kJ/kmol), h_D is the enthalpy of the distillate stream (kJ/kmol), H_D is the enthalpy of the inlet vapour stream to the condenser (kJ/kmol) and S (kg/h) is the steam entering the reboiler. The enthalpy values are functions of the corresponding temperatures ($^{\circ}\text{C}$), T_{head}

and T_{bottom} since they are computed as the product of the specific heat at constant pressure of the stream $C_{p,j}$ (kJ/kg K) and the temperature T_i .

Along the column, there is a pressure drop that is a function of the vapour stream passing through the column and the number of trays n_{tray} present in the real column. As V and V^S have approximately the same value, both of them could be used to compute this pressure drop and, therefore, the value of the bottom pressure P_{bottom} . The head pressure P_{head} is maintained by a controller, while c is a constant value to compute the pressure drop through the column.

$$P_{bottom} = P_{head} + (v^S / c)^2 \cdot n_{tray} \quad (5.39)$$

The concentration of vapour and liquid streams leaving each zone are at equilibrium, expressed by Raoult's law (5.40), which depends on the equilibrium constants K . The parameter E_f indicates the tray efficiency, which varies from 0 to 1, which is the same as saying that it varies from 0 to 100%. At the reboiler, the efficiency is equal to 1, since it is considered an ideal tray:

$$y_j^S = K^S(P_{bottom}, T_{bottom})x_j^S \quad (5.40)$$

$$y_j = E_f(K(P_{head}, T_{head})x_j - y_j^S) + y_j^S \quad (5.41)$$

Due to the absence of energy balances for each tray of the column, head and bottom temperatures were obtained from experimental data using the ALAMO (Automated Learning of Algebraic Models for Optimization) software for model building, (Cozad, et al., 2014), (Cozad, et al., 2015). ALAMO generates algebraic models of simulations or experiments.

The ALAMO software was developed at Carnegie Mellon University with the purpose of generating algebraic models which are accurate, simple in functional form and generated from a minimal experimental data set. It is based on using an iterative design of experiments to build low-complexity surrogate models. It works following the algorithmic flowchart shown in Figure 5.5, following a three step iterative process:

1. Initial design of experiment: define an initial training set.
2. Model building: build a low complexity surrogate model based on the current training, solving an MILP/MINLP problem.
3. Adaptive sampling: by interrogating the system using Error Maximization Sampling (EMS), the training set is updated with new

points sampled where the current model breaks down. To do that, ALAMO uses a derivative-free optimization solver called SNOBFIT.

- Repeat steps two and three until no points can be found that violate the model.

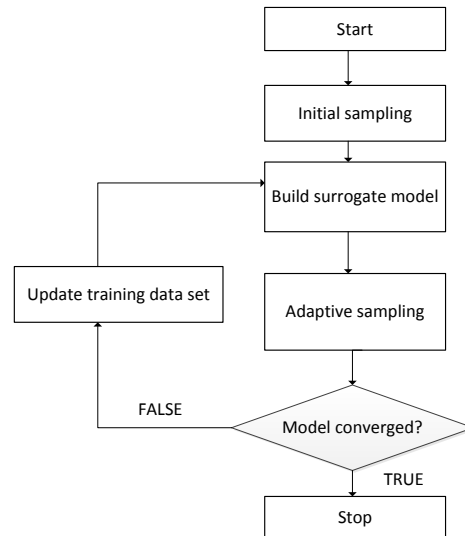


Figure 5.5. Algorithmic flowchart of ALAMO.

In this study, since it is assumed that the feed rate and the composition are kept constant, the models for the head and bottom temperatures were generated as functions of the model inputs, namely, the steam flow to the reboiler S (kg/h) and the reflux flow R (kg/h). These expressions are obtained from the simulation of the rigorous dynamic model of the distillation column, where f_1 and f_2 represent second order polynomial functions. For this task, 82 experiments at different operating points, varying steam and reflux flows within the allowed range, were carried out to generate the data used to fit the following second-order polynomial functions:

$$T_{head} = f_1(S, R) \quad (5.42)$$

$$T_{bottom} = f_2(S, R) \quad (5.43)$$

In addition, it is assumed that the light key component, ethane, is removed completely from the column via the distillate stream, that is,

$$x_B(C_2H_6) = 0 \quad (5.44)$$

The new variables included in the simplified model are presented in Table 5.3, the rest of the variables have been previously defined in Table 5.2.

Table 5.3. Nomenclature used in the steady-state simplified model.

Variable	Meaning	Units
v	Vapour molar flow rate in the enriching section	kmol/h
v^s	Vapour molar flow rate in the stripping section	kmol/h
l^s	Liquid molar flow rate in the stripping section	kmol/h
V	Vapour mass flow rate in the enriching section	kg/h
V^s	Vapour mass flow rate in the stripping section	kg/h
L	Liquid mass flow rate in the enriching section	kg/h
L^s	Liquid mass flow rate in the stripping section	kg/h
x_j^s	Molar fraction of component j in the liquid stream through the stripping section	$^0/1$
x_j	Molar fraction of component j in the liquid stream through the enriching section	$^0/1$
y_j^s	Molar fraction of component j in the vapour stream through the stripping section	$^0/1$
y_j	Molar fraction of component j in the vapour stream through the enriching section	$^0/1$
x_{j,n_feed}	Molar fraction of component j in the feed stream	$^0/1$
$x_{D,j}$	Molar fraction of component j in the distillate stream	$^0/1$
$x_{B,j}$	Molar fraction of component j in the bottom stream	$^0/1$
H^s	Enthalpy for liquid stream in the stripping section	kJ/kg
h^s	Enthalpy for vapour stream in the stripping section	kJ/kg
h_B	Enthalpy of the bottom stream	kJ/kg
h_D	Enthalpy of the distillate stream	kJ/kg
H_D	Enthalpy of the input vapour stream to the condenser	kJ/kg
P_{bottom}	Bottom pressure	Pa
P_{head}	Head pressure	Pa
n_{tray}	Number of trays	#

The differences in the number of variables and computation time between the rigorous model and the simplified steady-state model are presented in Table 5.4, where it can be seen that the model size has been reduced by 28 times, and Table 5.7, where the reduction in the

computation time is shown. Both models share the same boundary variables as Table 5.5 shows.

Table 5.4. Size comparison between steady-state models, rigorous and reduced.

	Rigorous	Reduced
Equations	1076	39
Explicit variables	921	29
Algebraic variables	148	3
Boundary variables	7	7

5.1.4. Simulation and comparison between models

In order to test the reduced model, the same changes in the input variables, reflux and steam flow rates have been applied to the reduced and full models in parallel, as shown in Figure 5.6, starting from a steady-state to check the behaviour of the output variables.

The initial steady-state corresponds to the boundary conditions shown in Table 5.5 that include the composition, flow and temperature of the feed stream, the steam and reflux flows and the pressure at the top of the column.

Table 5.5. Boundary conditions for the simulation.

	Rigorous
$z(\text{C}_2\text{H}_6)$ ($^{\circ}/1$)	0.099
$z(\text{C}_4\text{H}_{10})$ ($^{\circ}/1$)	0.446
F (kg/h)	22950
S (kg/h)	5550
R (kg/h)	8750
T_{feed} ($^{\circ}\text{C}$)	57.27
P_{head} (bar)	15.74

Some significant output variables have been represented in Figure 5.7, Figure 5.8, Figure 5.9, Figure 5.10 and Figure 5.11, where it can be observed that the achieved steady-states are not the same, due to the strong structural differences.

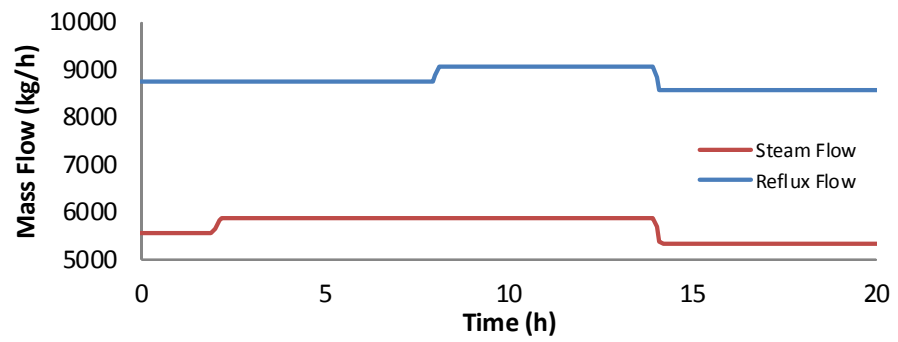


Figure 5.6. Changes in the reflux and steam flows.

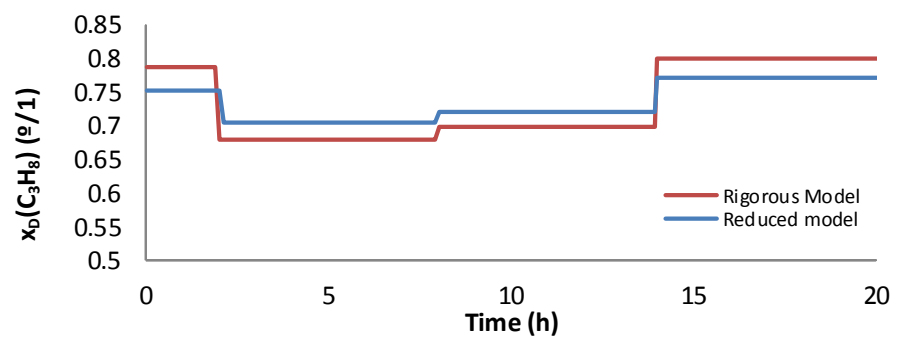


Figure 5.7. Evolution of the composition of propane in the distillate.

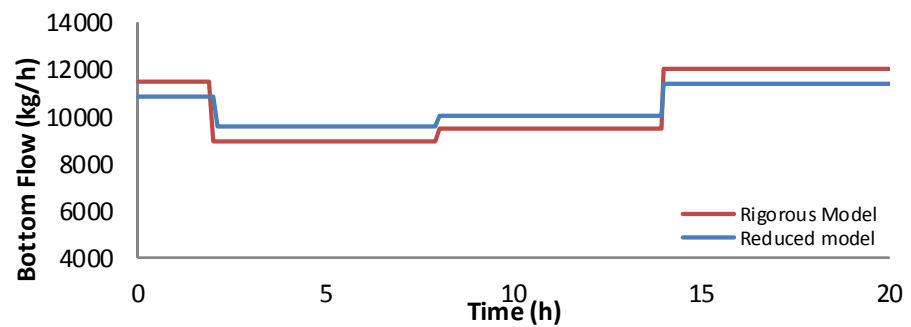


Figure 5.8. Evolution of the bottom flow rate.

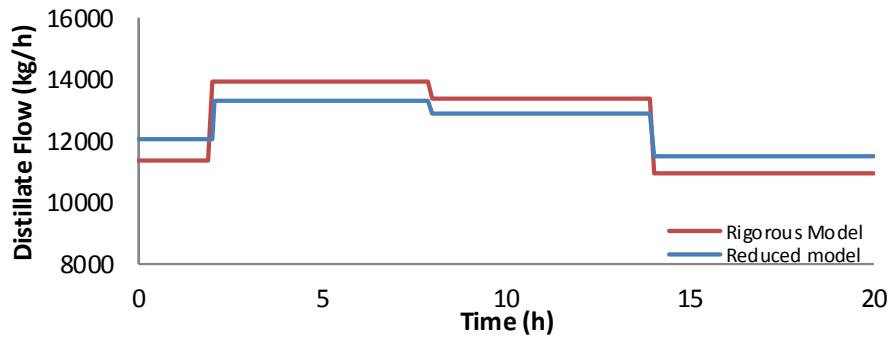


Figure 5.9. Evolution of the distillate flow rate.

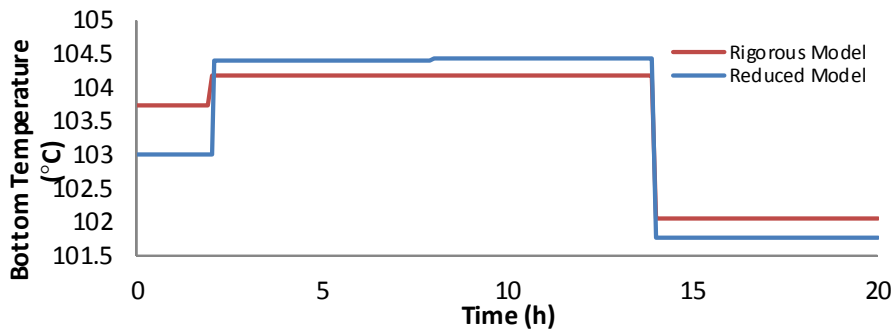


Figure 5.10. Evolution of the bottom temperature.

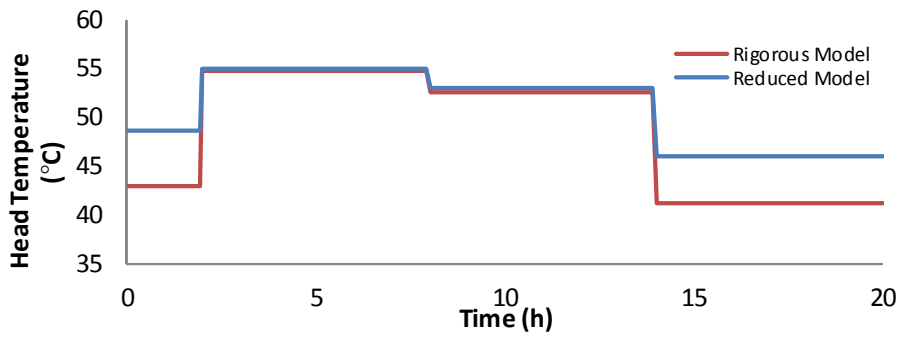


Figure 5.11. Evolution of the head temperature.

5.2 Basic control

The basic control for the distillation column has also been modelled in the rigorous dynamic model. This basic control includes continuous PI controllers to control steam and reflux flow rates, the level at the bottom of the column, the level in the condenser and the head pressure.

The steam flow controller manipulates the opening of the valve located in the steam inlet to the reboiler. The reflux flow controller manipulates the opening of the valve located in the reflux inlet to the column.

The level in the accumulator is controlled by manipulating the distillate flow, whereas the level at the bottom of the column is controlled by manipulating the bottom flow. The pressure at the top of the column is controlled by changing the outlet flow of the condenser.

Linear valves have been considered, that is, the flow passing through it, F_v , is proportional to the input signal to the valve, u_v , and the range of the controlled variable ($u_{min} : u_{max}$):

$$F_v = u_v(u_{max} - u_{min}) + u_{min} \quad (5.45)$$

5.3 Model predictive control DMC

As shown in Figure 5.1, a DMC controller is responsible for maintaining head and bottom temperatures close to the set points given by the RTO layer. To do that, the DMC controller manipulates the references for the steam and reflux flow controllers.

DMC uses the step response (5.47) to model the process, only taking into account the first N terms, therefore assuming the process to be stable and without integrators. In this case, 40 model coefficients have been considered. As regards the disturbances (\hat{n}_t), their value will be considered to be equal to the measured value of the output ($y_{m,t}$) minus the one estimated by the model (\hat{y}_t) (de Prada, et al., 1993) (Camacho & Bordons, 2000).

$$\hat{n}_t = y_{m,t} - \hat{y}_t \quad (5.46)$$

and therefore the predicted value of the output will be

$$\hat{y}_t = \sum_{i=1}^N g_i \Delta u_{t-i} + g_N u_{t-N} + \hat{n}_t \quad (5.47)$$

where g_i are the coefficients of the step response model and Δu the input changes with respect to the inputs applied before.

The rigorous dynamic model described in section 5.1.1 served as the basis to identify the linear dynamic model used in the DMC controller, which consists of the step responses of the head and bottom temperatures to changes in the set points of the reflux and steam controllers (see Figure 5.1). No more models are required, since it is an unconstrained DMC, as in the commercial controller from AspenTech.

In this example the linearization operating point corresponds to: $T_{bottom}=93$ °C, $T_{head}=49.5$ °C, $S=5220$ kg/h and $R=8620$ kg/h.

The objective of a DMC controller is to drive the output as close to the set point w_t as possible, in a least-squares sense, with the possibility of the inclusion of a penalty term ν on the input moves. Therefore, the manipulated variables are selected to minimize a quadratic objective J that can consider the minimization of future error alone and can include the control effort, in which case it presents the generic form.

$$J = \sum_{i=1}^p (\hat{y}_{t+i} - w_{t+i})^2 + \sum_{i=1}^m \nu (\Delta u_{t+i-1})^2 \quad (5.48)$$

The tuning parameters of the DMC controller of the distillation column are: a prediction horizon (p) equal to 40, a control horizon (m) of 3 and a sample time of 6 minutes.

5.4 RTO problem formulation

The optimization problem solved in the RTO layer is based on the non-linear steady-state simplified model described in the previous section that can be considered as a grey-box model, since it combines a partial theoretical structure with experimental data to complete the model. This model reduction implies a strong structural plant-model mismatch; in addition, the real process is controlled by an MPC layer (the DMC controller) which is not considered in either the RTO or the steady-state model. So, the inputs of the real process, DMC set points for bottom and head temperatures, are not the same as the model inputs used by the RTO, that is, steam and reflux flows.

Another source of uncertainty is due to the identification of the linear dynamic models that are required by the DMC controller. This identification is carried out around a certain operating point. So these models could be incorrect if the process is operated away from this point, resulting in more uncertainty the further the decision variables are moved away from the identification point. The presence of this strong uncertainty implies that the optimal solution of the RTO model will not be the same as the process optimum, making it necessary to apply MA methodology to achieve the optimal operating point of the process.

The economic objective of RTO is to maximize the operation profit, considering the benefits obtained from producing distillate D (kg/h) with a given purity specification and minimizing the steam consumption S (kg/h) for a given feed (Porru, et al., 2015). Prices P_S and P_D have been fixed for both streams, 20 €/ton for the steam consumed and 80 €/ton for propane in distillate, assuming that it is above the composition specification. Below this target, the price decreases as a function of the composition following the sigmoid behaviour shown next.

$$P_D = 50 + \frac{9000}{300 + e^{-(X_D(C_3H_8) - 0.8)600}} \quad (5.49)$$

The objective function is represented by the value of ϕ (€/h) calculated by (5.50) and the constraint over the composition of propane in the distillate stream g is expressed by (5.51), whereas the RTO problem is formulated as:

$$\max_{S,R} \phi = P_D(x_D(C_3H_8))D - P_S S \quad (5.50)$$

s.t

$$\text{Steady - state simplified model of the process} \quad (5.22) - (5.38) \quad (5.51)$$

$$g = x_D(C_3H_8) \geq 0.80$$

Lower and upper limits are fixed for the model inputs, that is, steam and reflux set points.

$$4000 \leq S \leq 6000 \quad (5.52)$$

$$6000 \leq R \leq 11000$$

It might happen that, due to plant-model mismatch, the computed set points for the bottom and head temperatures cannot be enforced exactly. Hence, the values of the temperatures T_{head} and T_{bottom} (°C) in the RTO problem are constrained as follows:

$$40 \leq T_{head} \leq 60 \quad (5.53)$$

$$80 \leq T_{bottom} \leq 110$$

Another constraint is the maximum loss of pressure ΔP to prevent the flooding of the column:

$$\Delta P \leq 0.25 \quad (5.54)$$

In this example, as mentioned above, the inputs of the RTO model are not the same as the plant inputs due to the presence of the DMC layer. The inputs of the RTO model are (S, R) , whereas the process inputs are the references of the DMC controller $(SP_{T_{bottom}}, SP_{T_{head}})$, as shown in Figure 5.12. The DMC controller manipulates the references for the steam and reflux flow controllers to maintain head and bottom temperatures close to the set points given by the RTO layer.

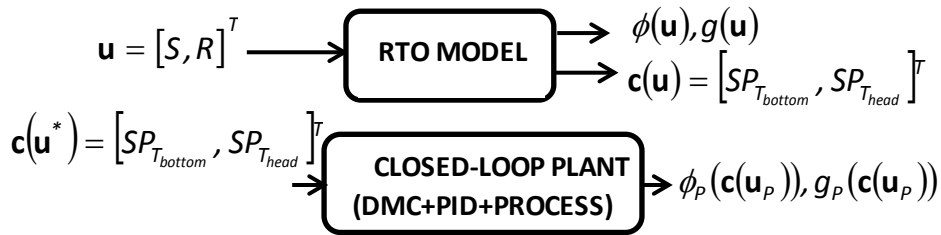


Figure 5.12. Comparison between plant and model inputs/outputs.

As shown previously, the reduced model presents a strong structural plant-model mismatch, so the model optimum is not the same as the real optimum (see Table 5.6). The real optimum has been obtained by solving the RTO problem (5.55)-(5.56) based on the rigorous steady-state model, assuming that this optimum is the same as the process optimum, since the rigorous steady-state model has been obtained by directly removing the derivative terms. The model optimum is obtained by solving the steady-state simplified model and applying the obtained solution to the real process.

Table 5.6. Comparison between Process and Model Optimum.

	Real Optimum	Model Optimum
<i>S</i> (kg/h)	5558.20	5000.00
<i>R</i> (kg/h)	9389.67	9265.70
<i>D</i> (kg/h)	11179.53	9965.52
<i>SP</i>_{thead} (°C)	50.98	42.58
<i>SP</i>_{Tbottom} (°C)	102.02	96.33
<i>x</i>_D(C₃H₈) (°/1)	0.800	0.819
<i>φ</i>_P (€/h)	783.19	714.50

Solving the nominal RTO problem without modifiers implies a decrease of 8.77 % in the profit obtained. To solve this issue, the MA schemes developed in this thesis and the traditional MA approaches are implemented. The NLP optimization problems have been solved using a sequential approach with a sequential quadratic programming (SQP) algorithm implemented in the SNOPT library (Gill, et al., 2008) and executed in EcosimPro software (EAInt, 2013).

The size reduction in the model used in the RTO layer (see Table 5.4) implies that the optimization problem could be solved faster and more easily than by using a more complicated one, as can be observed in Table 5.7.

Table 5.7. Time comparison between steady-state models, rigorous and simplified.

	Rigorous	Simplified
Optimization time (s)	7.594	0.235
Number of iterations	56	50
Time per iteration (s)	0.135	0.0047

The execution of one RTO problem is 32 times faster using the reduced model, which allows the use of RTO in real applications, where decisions must be implemented within a time scale of a few seconds. In this example, the use of the rigorous model does not imply a problem with respect to the optimization time, but it is important to note that this issue could happen using large scale models, where the optimization time could be prohibitive and the use of reduced models is the only way to deal with them.

5.5 Implementation of MA in the depropanizer distillation column

The performance of the proposed MA schemes will be illustrated through the case study of the industrial depropanizer distillation column and the results will be compared to those obtained with standard MA techniques, such as, NMA and DMA.

The rigorous dynamic model (see section 5.1.1.), including the basic control layer (see section 5.2) and the MPC layer (see section 5.3), has been simulated in EcosimPro software (EAInt, 2013), to be used as a substitute for the real process, in order to study the performance of the different MA approaches. First, static MA methods are applied to optimize the operation of the depropanizer distillation column. Notice that this process has a long settling time, approximately 6 hours, which makes the implementation of static approaches impractical for industrial use, since the plant optimum is achieved after several days of operation. This long period of time can result in loss of optimality, since operating conditions and plant-model mismatch could change and the method would not converge to the process optimum. To overcome this problem, the proposed MA approaches based on transient measurements could be applied to speed up the convergence to the real optimum.

By applying MA, the original optimization problem formulated by (5.50) and (5.51) is changed by adding some modifiers in the cost function and constraints, giving rise to the modified problem (5.55). In principle, only the constraint on the distillate composition will be modified, although other constraints will be considered when certain techniques are applied. The subscript “P” indicates that the variable is measured from the process and the subscript “k-1” is the measurement taken in the previous RTO iteration:

$$\max_u \phi_M = P_D(x_D(C_3H_8))D - P_S S + \lambda_k^T(\mathbf{c}(\mathbf{u}) - \mathbf{c}(\mathbf{u}_{P,k-1})) \quad (5.55)$$

s.t

Stationary model of the process (5.22) - (5.38)

$$g_M = -x_D(C_3H_8) + 0.80 + \mathbf{v}_k^T(\mathbf{c}(\mathbf{u}) - \mathbf{c}(\mathbf{u}_{P,k-1})) + \varepsilon_k \leq 0 \quad (5.56)$$

$$4000 \leq S \leq 6000$$

$$6000 \leq R \leq 11000$$

$$\begin{aligned} 40 &\leq T_{head} \leq 60 \\ 80 &\leq T_{bottom} \leq 110 \end{aligned}$$

The number of modifiers required to adapt the RTO problem n_k is given by (5.57), where n_u is the number of decision variables which are head and bottom temperatures (T_{bottom} , T_{head}) and n_g is the number of constraints (distillate composition); so, in our problem, $n_k = 5$:

$$n_k = n_g + n_u(n_g + 1) \quad (5.57)$$

The modifiers λ , γ and ε are given by (5.58)-(5.60) and represent the difference between experimental and model gradients. Several techniques to compute process gradients have been presented in the previous sections. However, the calculation of model gradients requires special attention in this case because they cannot be computed directly, since T_{bottom} and T_{head} are not considered independent variables of the RTO model. So the modifiers are computed as follows (Costello, et al., 2013), (Rodríguez-Blanco, et al., 2015):

$$\lambda_k^T = \nabla_{\mathbf{c}} \phi_P |_{\mathbf{c}(\mathbf{u}_{p,k-1})} - \nabla_{\mathbf{c}} \phi |_{\mathbf{c}(\mathbf{u}_{k-1}^*)} = \nabla_{\mathbf{c}} \phi_P |_{\mathbf{c}(\mathbf{u}_{p,k-1})} - \nabla_{\mathbf{u}} \phi (\nabla_{\mathbf{u}} \mathbf{c})^+ |_{\mathbf{u}_{k-1}^*} \quad (5.58)$$

$$\gamma_k^T = \nabla_{\mathbf{c}} g_P |_{\mathbf{c}(\mathbf{u}_{p,k-1})} - \nabla_{\mathbf{c}} g |_{\mathbf{c}(\mathbf{u}_{k-1}^*)} = \nabla_{\mathbf{c}} g_P |_{\mathbf{c}(\mathbf{u}_{p,k-1})} - \nabla_{\mathbf{u}} g (\nabla_{\mathbf{u}} \mathbf{c})^+ |_{\mathbf{u}_{k-1}^*} \quad (5.59)$$

$$\varepsilon_k = g_P(\mathbf{c}(\mathbf{u}_{p,k-1})) - g(\mathbf{u}_{k-1}^*) \quad (5.60)$$

where (*) indicates the Moore-Penrose pseudo-inverse.

The choice of evaluating the gradients at $\mathbf{c}(\mathbf{u}_{k-1})$ does not pose any conceptual difficulty, since the set points are applied to the plant. However, this would be problematic if the gradients were evaluated at \mathbf{u}_{k-1} , since the computed \mathbf{u}_{k-1} is not applied to the plant and typically differs from the plant inputs achieved by the controlled plant at steady-state (François, et al., 2016) (Papasavvas, et al., 2017).

As usual in MA methodology, the modifiers are applied filtered using a first-order exponential filter, where \mathbf{K}_λ , \mathbf{K}_γ and \mathbf{K}_ε are diagonal matrices with eigenvalues in the interval (0, 1], (5.61)-(5.63) to smooth changes in the decision variables and to ensure convergence.

$$\lambda_k = (\mathbf{I} - \mathbf{K}_\lambda) \lambda_{k-1} + \mathbf{K}_\lambda \lambda_k \quad (5.61)$$

$$\gamma_k = (\mathbf{I} - \mathbf{K}_\gamma) \gamma_{k-1} + \mathbf{K}_\gamma \gamma_k \quad (5.62)$$

$$\boldsymbol{\varepsilon}_k = (\mathbf{I} - \mathbf{K}_\varepsilon) \boldsymbol{\varepsilon}_{k-1} + \mathbf{K}_\varepsilon \boldsymbol{\varepsilon}_k \quad (5.63)$$

Different MA methodologies have been implemented, using different models in the RTO layer and considering different control structures and operational constraints.

In particular, we aim to test the proposed method based on the direct estimation of the process gradients (section 5.5.4), comparing it with standard DMA and NMA formulations (section 5.5.1), as well as with the one based on NEC and transient measurements (section 5.5.4).

The developed method, based on the combination of transient and steady-state measurements, will also be implemented in this example (section 5.5.3).

Another considered approach to speed up the convergence of the NMA methodology is that based on the reduction in the number of modifiers by computing those modifiers which adapt the optimization problem and whose solution minimizes the Lagrangian function, measured directly from the process (section 5.5.2). In this way, the implementation of NMA is easier and faster, allowing the use of this formulation in real processes. A summary of the applications developed during this thesis is given in Table 5.8.

Table 5.8. Summary of applications.

Section	MA technique	Control structure	RTO model (uncertainty)	Operational constraints	Objective
5.5.1	Static MA: - DMA (section 2.1.3) - NMA (section 2.1.5)	Column + PID + MPC + RTO (model inputs \neq plant inputs)	Nonlinear steady-state simplified model (structural)	$x_0(C_3H_8) \geq 0.80$	Test MA in realistic case studies with a MPC layer and a long settling time
5.5.2	NMA for dealing with operational constraints (section 3.3)	Column + PID + MPC + RTO (model inputs \neq plant inputs)	Nonlinear steady-state simplified model (structural)	$x_0(C_3H_8) \geq 0.80$ $\Delta P \leq 0.25$	Speed up the convergence of NMA by reducing the number of modifiers
5.5.3	MA combining transient (NEC-MA) and steady-state measurements (DMA) (section 4.3)	Column + PID + RTO (model inputs = plant inputs)	Nonlinear steady-state rigorous model (parametric (E) + structural (E_{cont}))	$x_0(C_3H_8) \geq 0.80$	Speed up the convergence of MA by combining transient and steady-state measurements to deal with parametric and structural uncertainty
5.5.4	MA based on transient measurements: - NEC-based (section 4.2) - RELS-based (section 4.4)	Column + PID + MPC + RTO (model inputs \neq plant inputs)	Nonlinear steady-state simplified model (structural)	$x_0(C_3H_8) \geq 0.80$	Speed up the convergence of MA by using transient measurements to estimate the plant gradients

5.5.1 Static MA

The DMA (Brdyś & Tatjewski, 1994), (Marchetti, et al., 2010) and NMA (Navia, et al., 2015) methods have been implemented on the depropanizer column solving the RTO optimization problem every 6 hours, the time required for the process to achieve a new steady-state. In both approaches, DMA and NMA, the first three iterations were used to estimate the process gradients by finite differences; in the second approach this is necessary to provide a good starting point for the upper optimization layer. Two initial operating points are imposed on the real process to take data and then the third operating point applied to the process corresponds to the nominal solution, that is, the result obtained by solving the RTO problem without modifiers. In this way, data from three different operating points are obtained to make the first gradient estimation.

The DMA methodology requires the addition of the dual constraint ($\kappa^{-1}(S_k) \geq a$), which guarantees a sufficient level of excitation to estimate the gradients accurately. The selected value in this case is 0.10. This dual constraint is evaluated using the set point values since, as shown in (5.58)-(5.60), the modifiers are computed as functions of these references. Equation (5.64) shows how to compute the vectors of differences with respect to the previous points:

$$\mathbf{s}_{k,i} = \mathbf{c}(\mathbf{u}_k) - \mathbf{c}(\mathbf{u}_{k-i}) \quad \forall i = 1, \dots, n_u \quad (5.64)$$

Figure 5.13 shows the evolution of the cost function during the process operation, whereas Figure 5.14 and Figure 5.15 show the bottom and head temperatures. The evolution of the composition of the distillate is shown in Figure 5.16, where it can be observed that the achieved operating point corresponds to an active constraint represented by the dotted red line. Figure 5.17, Figure 5.18 and Figure 5.19 represent the steam, reflux and distillate flows, respectively.

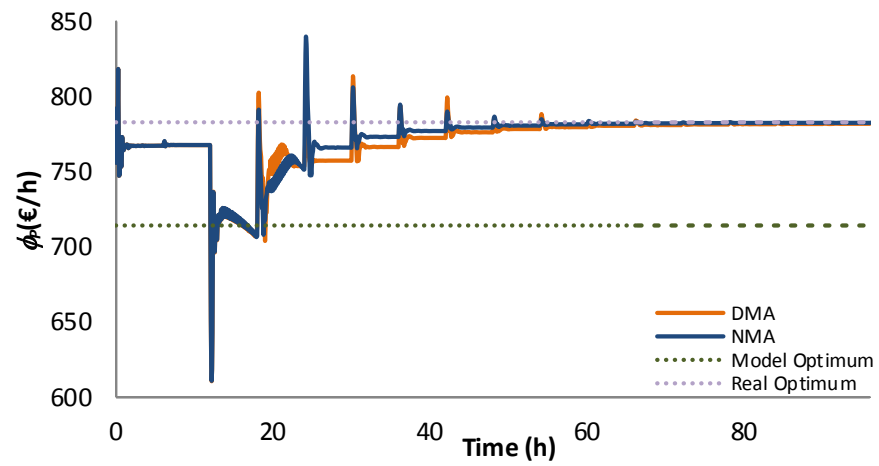


Figure 5.13. Evolution of the process cost function ϕ_p .

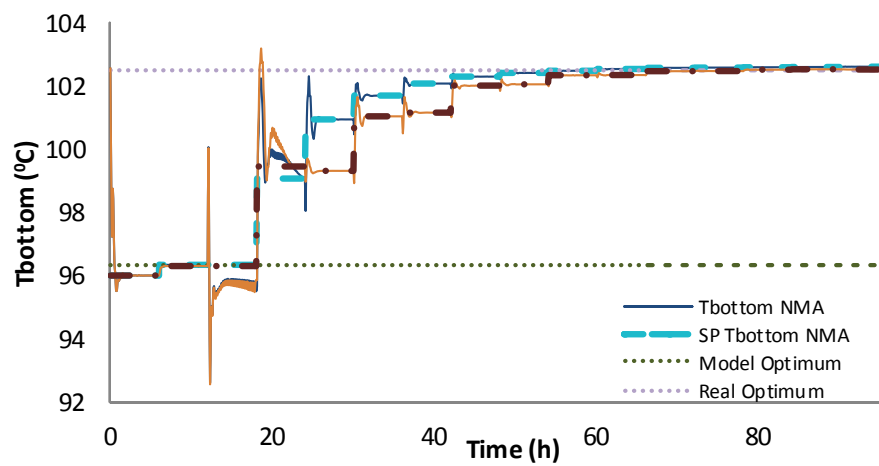


Figure 5.14. Evolution of the bottom temperature T_{bottom} .

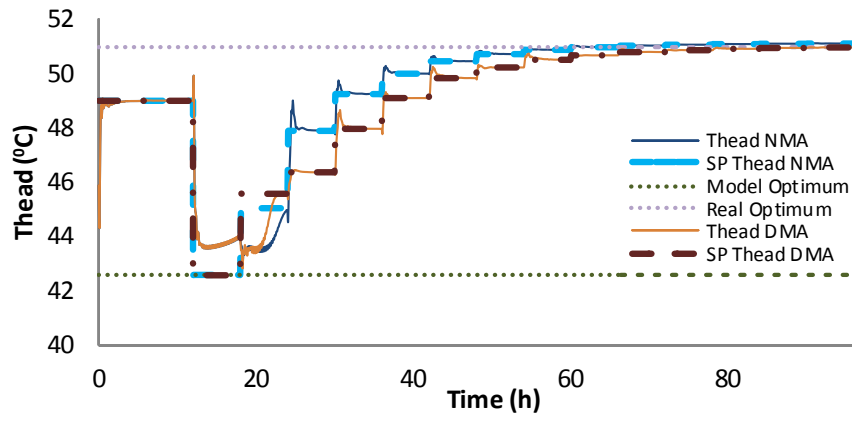


Figure 5.15. Evolution of the head temperature T_{head} .

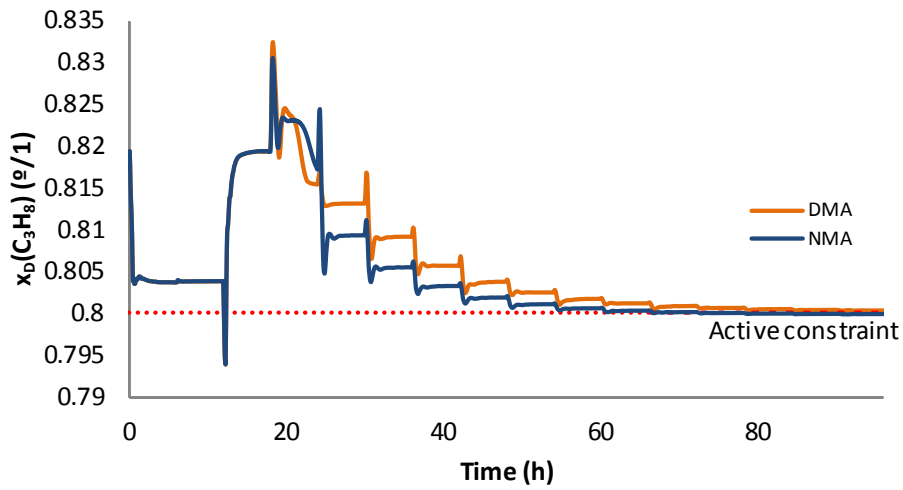


Figure 5.16. Evolution of the composition of propane in the distillate $x_D(C_3H_8)$.

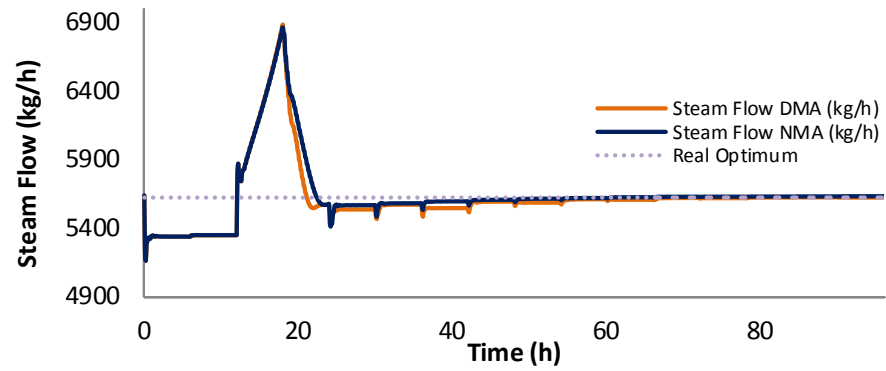


Figure 5.17. Evolution of the steam flow S .

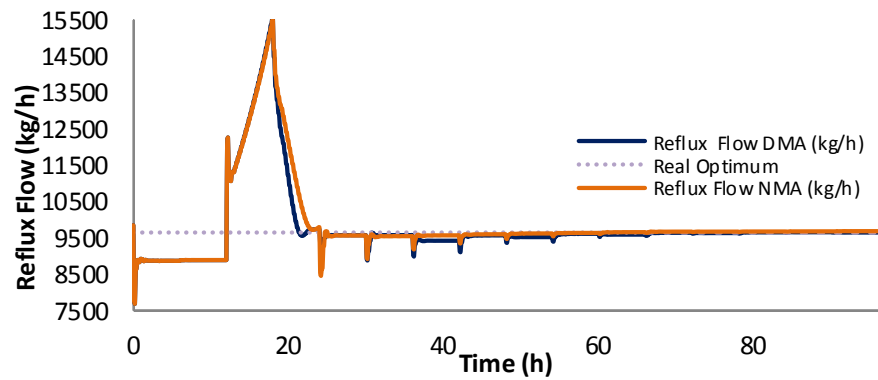


Figure 5.18. Evolution of the reflux flow R .

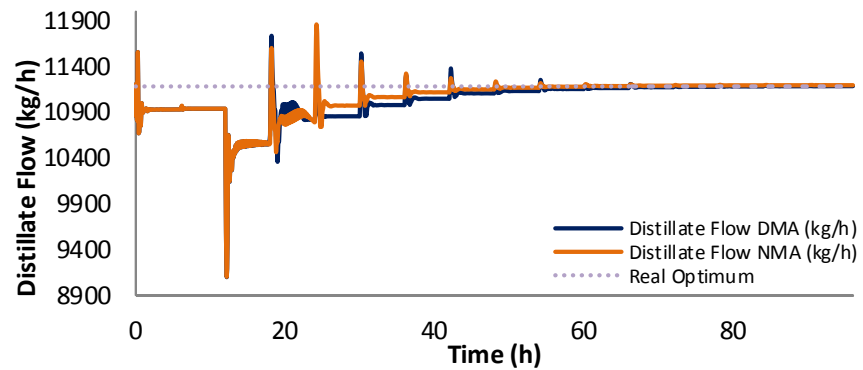


Figure 5.19. Evolution of the distillate flow D .

Fixing a tolerance band of 0.5% with respect to the optimal value of the cost function, the graphs (in particular, Figure 5.13) show that the optimum of the process is achieved after approximately 60 hours, using both DMA and NMA approaches. For the DMA approach, the number of steady-states needed was 10, the first two to initialize the computation of the process gradients ($n_u = 2$) and 8 related to RTO solutions. NMA also requires 10 steady-states, first, two initial steady-states ($n_u = 2$) to estimate plant gradients, then five to construct the first simplex corresponding to the number of first-order modifiers plus 1, and finally 3 RTO executions.

DMA only requires one tuning parameter a , the bigger this parameter the greater the excitation of the system, driving the process to the optimum following a less optimal path. However, the Nelder Mead algorithm used in NMA to estimate the modifiers has many tuning parameters, which strongly affect the speed of convergence and the path followed to achieve the real optimum. It is a difficult task to tune these parameters, since the effect of each one is not really known. So, a bad choice could mean reaching the optimum slowly, implying many steady-states. In addition, the estimation of the initial modifiers also affects the behaviour of the method.

An overshoot is observed in Figure 5.17 and Figure 5.18; this is due to the saturation of the valves that manipulate the reflux flow and the steam flow, which happens because the reference given by the RTO layer to the DMC cannot be reached by the process (see Figure 5.14 and Figure 5.15) because the valves for the steam and the reflux flows become saturated. This solution corresponds to the nominal one, that is, the result of solving the RTO problem without modifiers. In spite of this unfeasible first solution, MA is able to bring the process to the real optimum, giving feasible solutions in the following iterations.

One important disadvantage of this kind of method is that they can work well with few decision variables, but when the number of inputs increases, the number of modifiers also grows, involving a bigger number of steady-states to converge to the optimal solution, which supposes a problem, especially for processes with long settling times like in the depropanizer column. To deal with this problem, NMA based on the minimization of the Lagrangian function will be implemented to reduce the number of required modifiers and therefore, speed up the convergence rate of the method. Results are shown in section 5.5.2.

As shown in Figure 5.13 to Figure 5.19, the convergence to the optimum using static MA techniques takes two and a half days, which is a problem if

the methodology is implemented in real problems. To deal with this issue, transient measurements have been incorporated in the MA methodology to speed up the convergence to the real optimum. The corresponding results are shown in section 5.5.3 and 5.5.4.

5.5.2 Alternative NMA for dealing with operational constraints

For dealing with process-dependent constraints without increasing the number of modifiers, an alternative MA methodology can be implemented. It consists of computing the modifiers from the gradient of the Lagrangian function which incorporates information about the constraints, and thus only one gradient modifier for each process input is required. These modifiers are used in the Nested Modifier Adaptation (NMA) technique (presented in section 3.3), which has been implemented in the simulated depropanizer distillation column. The use of these modifiers implies a clear advantage over the NMA formulation (presented in section 2.1.5) in terms of problem size (number of decision variables in the outer optimization layer).

The obtained results show that the convergence rate of the alternative NMA approach is faster than applying standard NMA and even faster than applying DMA (presented in section 2.1.3).

To take into account more constraints, an active constraint on the column pressure drop has been added, so, the resulting optimization problem is given by (5.65) and (5.66):

$$\max_{S,R} \quad \phi = P_D(x_D(C_3H_8)D) - P_S S \quad (5.65)$$

s.t

Stationary reduced model of the process

$$g_1 = x_D(C_3H_8) \geq 0.80 \quad (5.66)$$

$$g_2 = \Delta P \leq 0.25$$

$$4000 \leq S \leq 6000$$

$$6000 \leq R \leq 11000$$

$$40 \leq T_{head} \leq 60$$

$$80 \leq T_{bottom} \leq 110$$

$$\Delta P \leq 0.25$$

In this case, the optimum operating point for the process and the RTO model is given by Table 5.9, where the real optimum corresponds to the solution obtained by solving the RTO problem (5.65) and (5.66), based on the rigorous steady-state model, whereas the model optimum is obtained by solving the steady-state simplified model and applying the obtained solution to the real process.

Table 5.9. Comparison between Process and Model Optimum.

	Real Optimum	Model Optimum
S (kg/h)	5558.20	5000.00
R (kg/h)	9389.67	9265.70
D (kg/h)	11179.53	9965.52
SP_{Thead} (°C)	50.98	42.58
$SP_{Tbottom}$ (°C)	102.02	96.33
$x_D(C_3H_8)$ (°/1)	0.800	0.822
ΔP (bar)	0.250	0.408
ϕ_P (€/h)	783.19	726.39

The modified problem applying the standard MA formulation is given by (5.67) and (5.68). The subscript “ P ” indicates that the variable is measured from the process and the subscript “ $k-1$ ” is the measurement taken in the previous steady state:

$$\max_u \phi_M = P_D(x_D(C_3H_8))D - P_S S + \lambda_k^T(\mathbf{c}(\mathbf{u}) - \mathbf{c}(\mathbf{u}_{P,k-1})) \quad (5.67)$$

s.t

Stationary model of the process (5.22) - (5.38)

$$g_{M,1} = -x_D(C_3H_8) + 0.80 + \mathbf{v}_{k,1}^T(\mathbf{c}(\mathbf{u}) - \mathbf{c}(\mathbf{u}_{P,k-1})) + \varepsilon_{k,1} \leq 0 \quad (5.68)$$

$$g_{M,2} = \Delta P - 0.25 + \mathbf{v}_{k,2}^T(\mathbf{c}(\mathbf{u}) - \mathbf{c}(\mathbf{u}_{P,k-1})) + \varepsilon_{k,2} \leq 0$$

$$4000 \leq S \leq 6000$$

$$6000 \leq R \leq 11000$$

The modifiers λ_k , \mathbf{v}_k and ε_k are given by (5.61)-(5.63) and represent the difference between experimental and model gradients.

However, if the alternative NMA technique described in chapter 3, section 3.3 is applied, using the Lagrangian formulation, the modified optimization problem will be formulated as follows, where the modifiers λ_k^L , ε_k are given by (5.71) and (5.72), where the first-order modifiers are directly

computed by the unconstrained upper optimization layer using the Nelder-Mead algorithm:

$$\max_{\mathbf{u}} \phi_M = P_D(x_D(C_3H_8))D - P_S S + \boldsymbol{\lambda}_k^{L^T} (\mathbf{c}(\mathbf{u}) - \mathbf{c}(\mathbf{u}_{p,k-1})) \quad (5.69)$$

$$\text{s.t} \quad (5.70)$$

Stationary model of the process (5.22) - (5.38)

$$g_{M,1} = -x_D(C_3H_8) + 0.80 + \varepsilon_{k,1} \leq 0$$

$$g_{M,2} = \Delta P - 0.25 + \varepsilon_{k,2} \leq 0$$

$$4000 \leq S \leq 6000$$

$$6000 \leq R \leq 11000$$

$$\boldsymbol{\lambda}_k^{L^T} = \nabla_{\mathbf{u}} L_p|_{\mathbf{u}_{k-1}, \boldsymbol{\mu}_{k-1}} - \nabla_{\mathbf{u}} L|_{\mathbf{u}_{k-1}, \boldsymbol{\mu}_{k-1}} \quad (5.71)$$

$$\varepsilon_{k,i} = g_{p,i}(\mathbf{u}_{k-1}) - g_i(\mathbf{u}_{k-1}) \quad i = 1, \dots, n_g \quad (5.72)$$

where the Lagrangian for the process (L_p) and the Lagrangian for the model (L) are defined as follows:

$$L_p = \phi_p + \boldsymbol{\mu}_k^T \mathbf{g}_p, \quad L = \phi + \boldsymbol{\mu}_k^T \mathbf{g} \quad (5.73)$$

In this example, using the Lagrangian formulation, the number of modifiers is reduced from 8 to 4, namely, two first-order modifiers corresponding to the Lagrangian gradient with respect to the two inputs and two zeroth-order modifiers for the two constraints.

As has been mentioned before, the RTO layer is executed every 6 hours, which is the time required for the process to reach steady state. In both dual MA and standard NMA, the first three operating points ($n_u + 1$) are used to estimate the process gradients by finite differences; in NMA, this is necessary to provide a good starting point for the outer optimization layer. Note that the starting point for the algorithms corresponds to the nominal model solution, that is, the result obtained by solving the RTO problem without modifiers.

DMA has been applied with a lower value for the dual constraint equal to 0.10. In DMA and NMA, three previous operating points ($n_u + 1$) are required to initialise the estimation of the modifiers, the third operating point corresponding to the nominal solution, that is, the result obtained by solving the RTO problem without modifiers.

The following figures show the obtained results as a function of the iterations, where NMA indicates the results of applying standard NMA, NMA_{Lag} indicates those obtained by using the Lagrangian formulation, and DMA the results for the implementation of DMA. Figure 5.20 shows the evolution of the cost function during the process operation, whereas Figure 5.21 and Figure 5.22 show the bottom and head temperatures. The evolution of the composition of the distillate is shown in Figure 5.23 and the pressure drop is presented in Figure 5.24, where it can be observed that the achieved operating point corresponds to the active constraints represented by the dotted red line. In this case, the modifiers have been applied without filtering, that is, $\mathbf{K}_\lambda = \mathbf{K}_\nu = \mathbf{K}_\epsilon = \mathbf{I}$.

Note that, since the model optimum violates the constraint on the pressure drop as given in Table 5.9, the first RTO iteration is performed with a non-zero value for the zeroth-order modifier $\epsilon_{0,2}$ so as to start the algorithm from the feasible region.

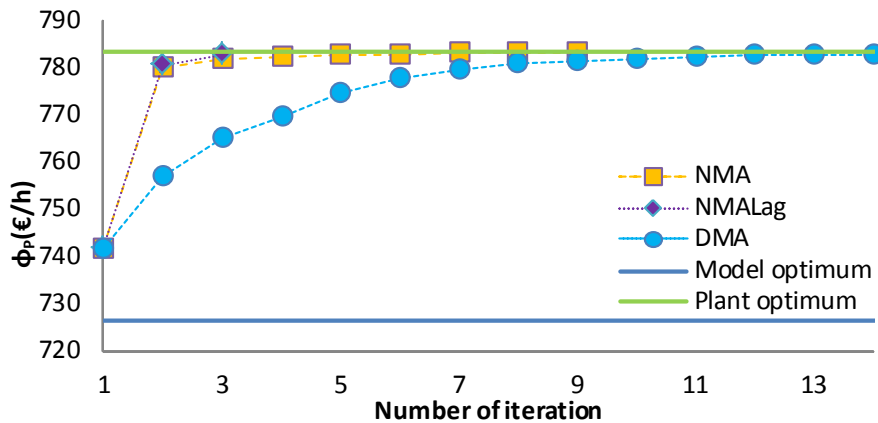


Figure 5.20. Evolution of the process cost function ϕ_p .

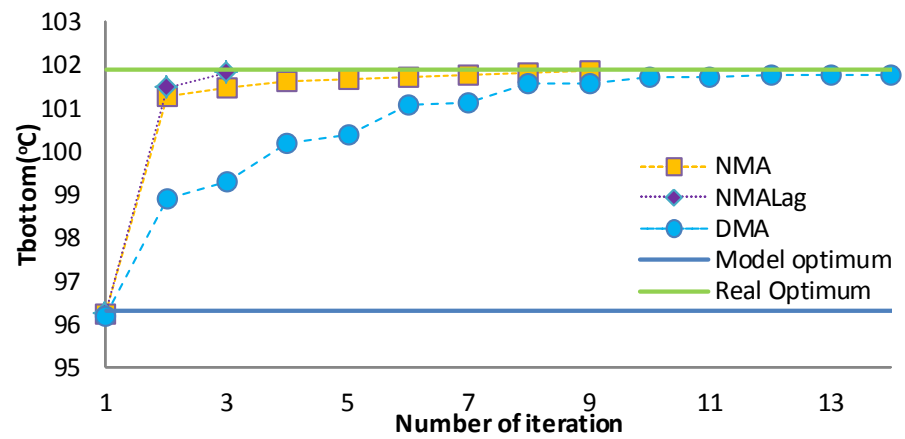


Figure 5.21. Evolution of the bottom temperature T_{bottom} .

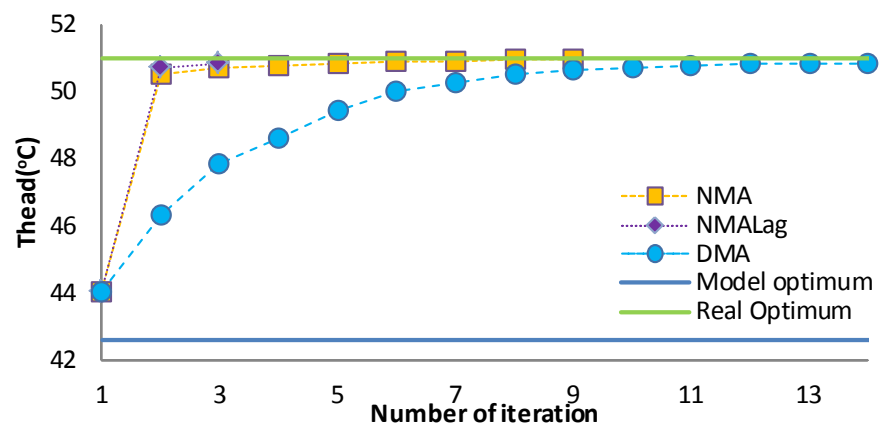


Figure 5.22. Evolution of the head temperature T_{head} .

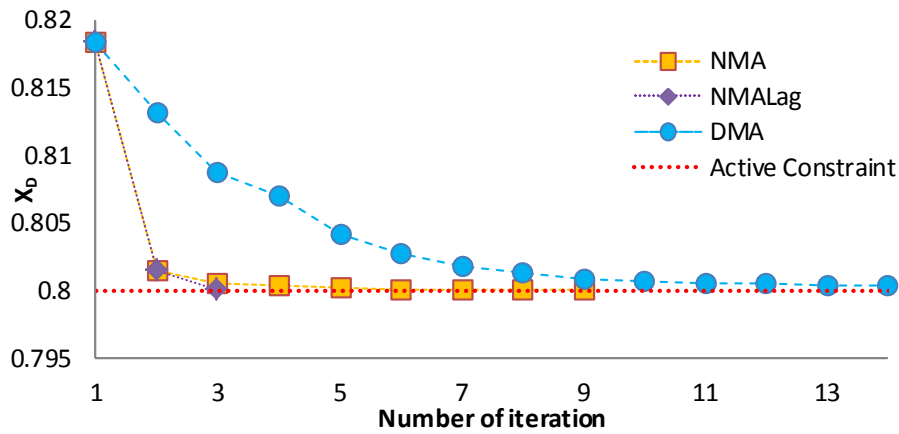


Figure 5.23. Evolution of the composition of propane in the distillate $x_D(C_3H_8)$.

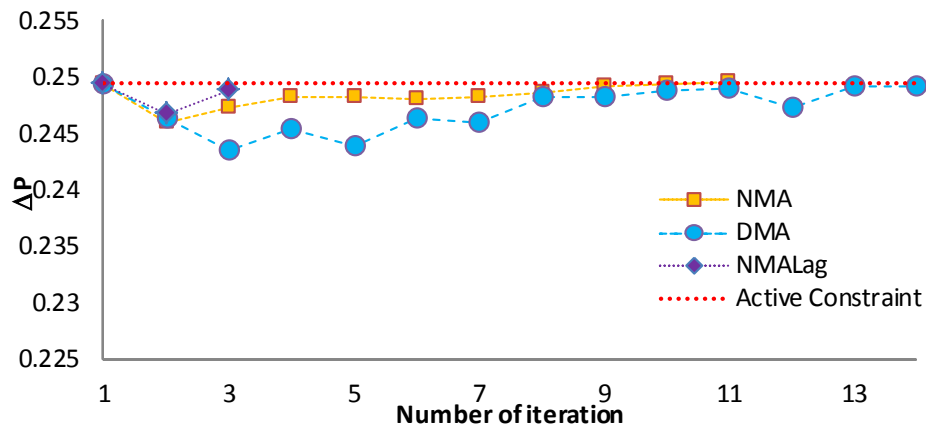


Figure 5.24. Evolution of the pressure drop ΔP .

The previous figures show that the number of iterations is drastically reduced by using the alternative NMA method, based on the computation of the Lagrangian gradients, compared to standard DMA and NMA, which converges to the optimum in a similar number of iterations. However, each NMA iteration may require several steady-states, as will be shown next. Even so, the alternative NMA presents a faster convergence than the other methods.

The following figures show the obtained results as a function of the RTO solutions, that is, the steady-state points achieved by the process. The red points are $n_u + 1$ previous operating points used to estimate the initial plant

gradients for both DMA and NMA. (5.24) shows the evolution of the process cost function, whereas (5.25) and (5.26) show the bottom and head temperatures. (5.29) shows the evolution of the composition of the distillate and (5.30) the pressure drop in the column during the operation corresponding to the active constraints represented by the dotted red line at the optimum:

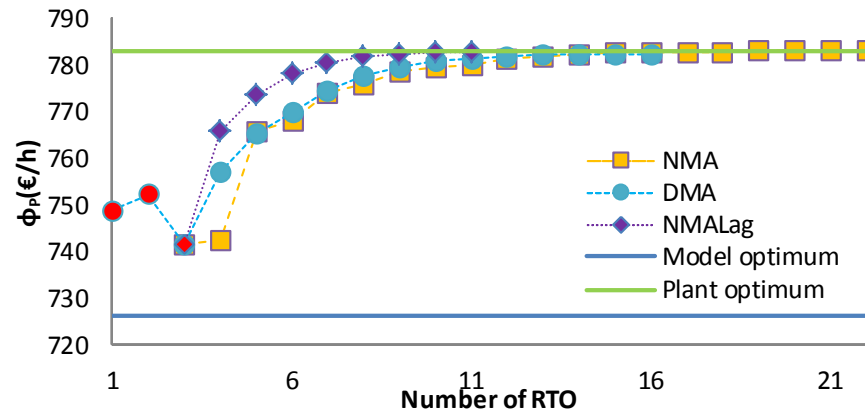


Figure 5.25. Evolution of the process cost function ϕ_p .

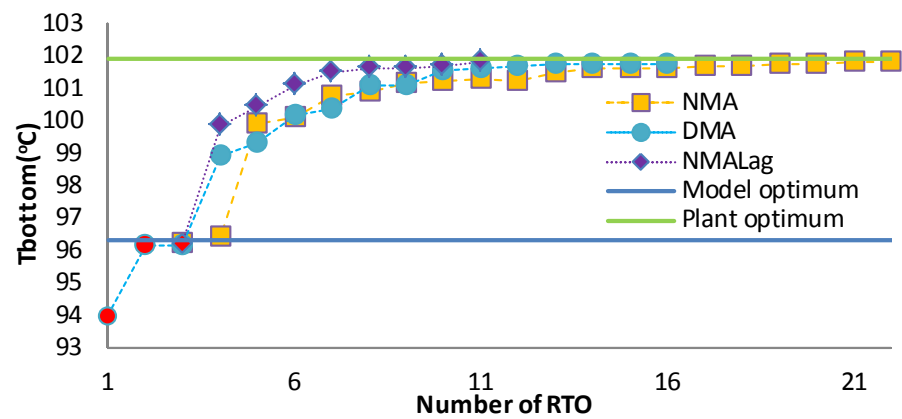


Figure 5.26. Evolution of the bottom temperature T_{bottom} .

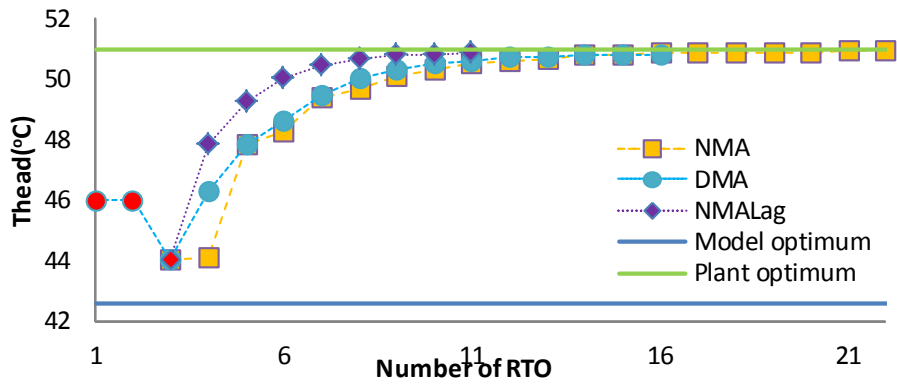


Figure 5.27. Evolution of the head temperature T_{head} .

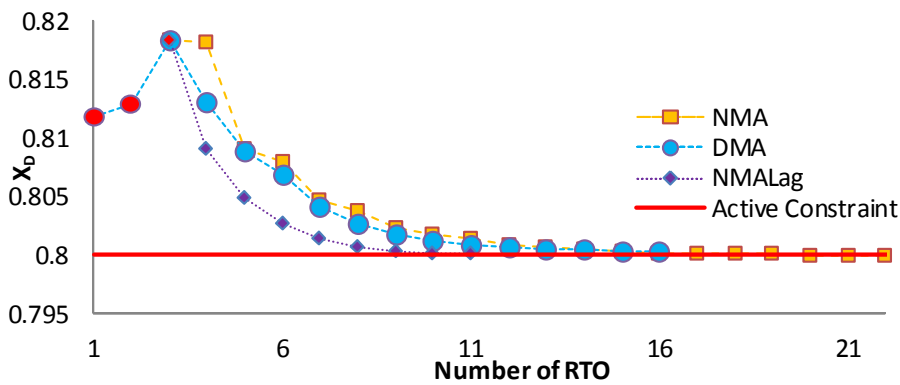


Figure 5.28. Evolution of the composition of propane in the distillate $x_D(C_3H_8)$.

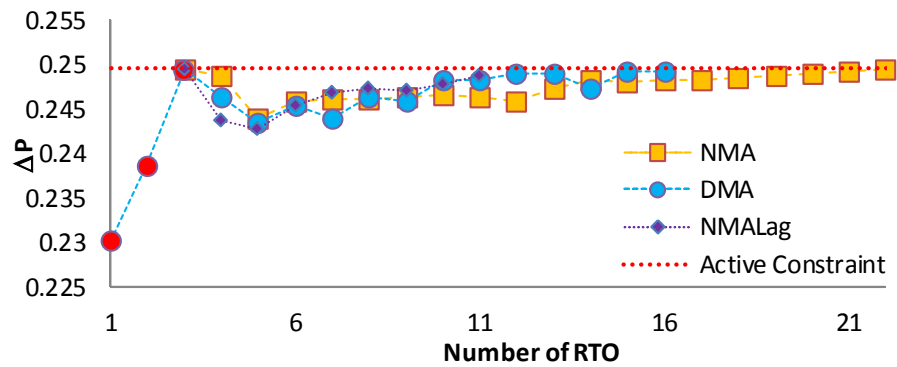


Figure 5.29. Evolution of the pressure drop ΔP .

These figures show that all three approaches are able to reach the plant optimum. The fastest convergence is achieved with NMALag. Indeed, the use of the Lagrangian formulation reduces the number of modifiers and thus also the number of decision variables in the outer optimization layer. This ultimately reduces the number of steady-states needed to reach plant optimality. However, it is important to keep in mind that the Nelder-Mead algorithm may require, in some cases, more than one steady-state value for each iteration, since it may need to evaluate several operating points to decide on the next simplex move.

The performance of each scheme to converge within 0.5% of the optimal cost is given in Table 5.10. For the DMA approach, the number of experiments is 16, namely, the first 3 to initialize the computation of the plant gradients ($n_u + 1 = 2$) and 13 related to RTO iterations. NMA requires 22 experiments, namely, 3 to estimate the plant gradients used in the computation of the initial modifiers, 6 to construct the first simplex (corresponding to the number of first-order modifiers plus 1), and 12 RTO iterations. The NMALag scheme requires only 11 experiments, namely, 3 for the initial estimation of the modifiers, 3 to construct the simplex and finally 5 RTO iterations to converge to the optimum.

Table 5.10. Summary results.

	DMA	NMA	NMALag
Number of Modifiers	8	8	4
First-order modifiers	6	6	2
RTO iterations	13	12	5
Total steady state points	16	22	11

5.5.3 MA combining transient and steady-state measurements

To speed up the convergence of the MA method, it is possible to use transient measurements to compute plant gradients during the transient. However, as explained in section 4.2, the methodology based on NEC only works if there is parametric uncertainty. For this reason, to deal with parametric and structural plant-model mismatch, an extension is presented in section 4.3 that combines the estimation of process gradients from the transient using NEC-MA and steady-state information using DMA.

In this example, the rigorous steady-state model obtained directly from removing derivative terms in the dynamic model is used in the RTO layer. This model has been chosen in this case because the MA method based on NEC works well only if there is parametric uncertainty, since it is necessary to know what the uncertain parameters are. Then structural uncertainty will be added to this model to check that MA based on NEC is not able to find the process optimum in the presence of structural plant-model mismatch.

The parametric modelling mismatch, which is added between the stationary RTO model and the dynamic process, is on the tray efficiency E_f (+20%) that affects the vapour-liquid equilibrium (5.20). The MPC layer is omitted in this example so as not to introduce more plant-model mismatch. In this case, then, the inputs of the process are the same as the inputs of the model, the references for the steam and reflux flow controllers.

Then, structural uncertainty is added by assuming that pressure is constant along the distillation column, that is, there is no pressure drop across the column and also, significant energy losses E_{lost} (kJ/h) in each tray are considered in the model used by the RTO layer as a function of the difference between tray temperature T_{tray} and ambient temperature T_{amb} multiplied by a global heat transfer coefficient U . This term has been considered in the steady-state model because it is easier to modify the RTO model than the model describing the real process. The real process is considered to be perfectly insulated, so there are no energy losses to the environment.

$$E_{lost} = U(T_{tray} - T_{amb}) \quad (5.74)$$

The RTO problem is defined as follows:

$$\max_{\mathbf{u}=[S,R]} \phi = P_D(x_D(C_3H_8))D - P_S S \quad (5.75)$$

s.t

$$\text{Rigorous steady – state model of the process} \quad (5.76)$$

$$g = x_D(C_3H_8) \geq 0.80$$

$$4000 \leq S \leq 6000$$

$$6000 \leq R \leq 11000$$

The modified RTO problem solved at each iteration is given by the following equations:

$$\max_{\mathbf{u}=[S,R]} \phi_M = P_D(x_D(C_3H_8))D - P_S S + \boldsymbol{\lambda}_k^T (\mathbf{u} - \mathbf{u}_{p,k-1}) \quad (5.77)$$

s.t

Rigorous steady – state model of the process

$$g_M = -x_D(C_3H_8) + 0.80 + \boldsymbol{\gamma}_k^T (\mathbf{u} - \mathbf{u}_{p,k-1}) + \varepsilon_k \leq 0 \quad (5.78)$$

$$4000 \leq S \leq 6000$$

$$6000 \leq R \leq 11000$$

The differences between the model optimum obtained by solving the RTO based on the rigorous steady-state model with uncertainty and the real optimum obtained by solving the RTO based on the rigorous model without uncertainty are shown in Table 5.11

Table 5.11. Comparison between Process and Model Optimum.

	Real Optimum	Model Optimum
$S(\text{kg/h})$	5558.20	5473.42
$R(\text{kg/h})$	9389.67	8549.26
$T_{head} (\text{°C})$	50.98	59.68
$T_{bottom} (\text{°C})$	102.02	103.29
$x_D(C_3H_8) (\text{°}/1)$	0.80	0.76
$\phi_P (\text{€}/h)$	783.19	491.36

As Table 5.11 shows, if the operating point obtained by solving the RTO without modifiers is applied to the process, the obtained distillate does not meet the specification. Below this target, the price decreases as a function of the composition following a sigmoid behaviour and, therefore, its value and the obtained profit are considerably reduced, to be precise, by about 37 % in this case.

First, it has been checked that the MA methodology using transient measurements is not useful for the case where structural uncertainty is present. This can be observed when this approach is applied in this case study, which presents both parametric and structural uncertainty.

In this case, RTO is executed every half an hour so the modifiers are also updated every half an hour. Figure 5.30 shows the evolution of the process cost function where TMA indicates the results obtained by applying transient MA based on NEC. Figure 5.31 and Figure 5.32 show the evolution of the decision variables, the set point of the steam and reflux flow

controllers in this case, while the evolution of the constraint on propane composition in the distillate stream is shown in Figure 5.33.

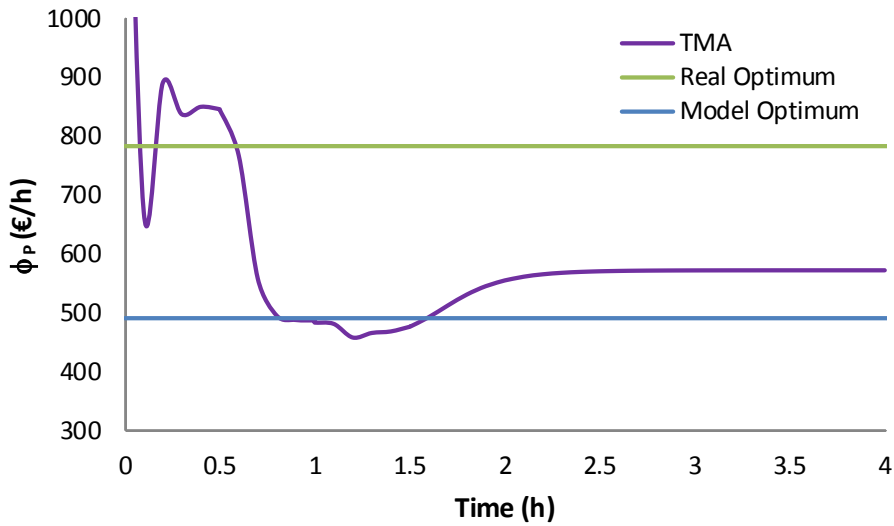


Figure 5.30. Evolution of the process cost function ϕ_P .

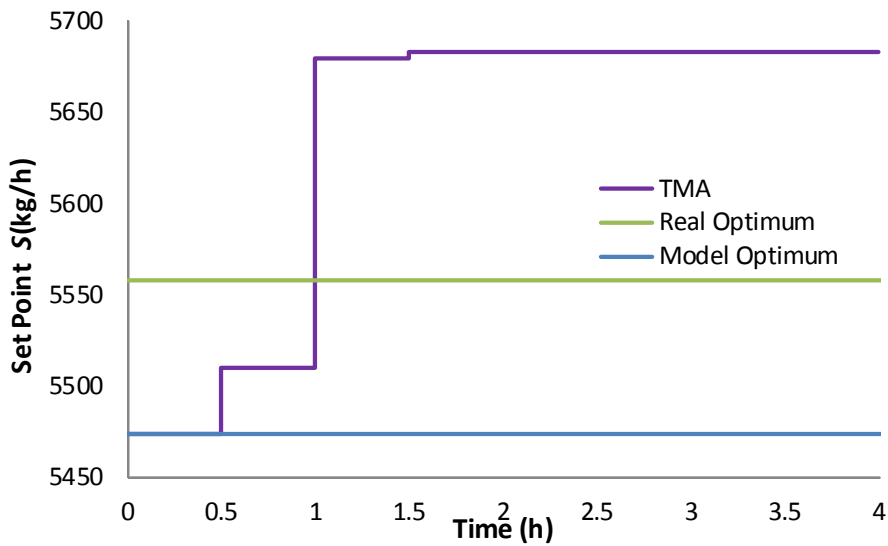


Figure 5.31. Evolution of the steam flow controller set point S.

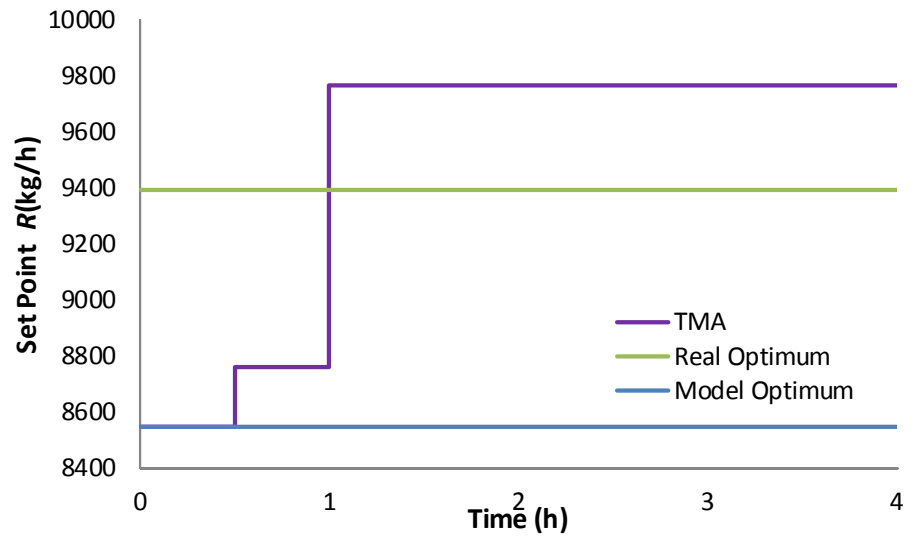


Figure 5.32. Evolution of the reflux flow controller set point R .

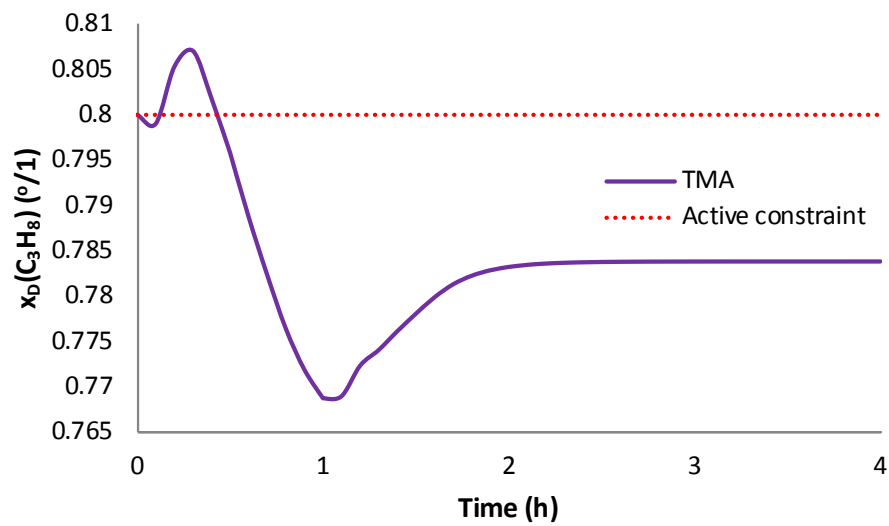


Figure 5.33. Evolution of the composition of propane in the distillate $x_D(\text{C}_3\text{H}_8)$.

Comparing the values obtained before (see Table 5.12) with the ones of the real optimum (see Table 5.11), the previous graphs show that the MA approach using transient measurements is not able to achieve the real optimum of the process when such a structural uncertainty has been

considered or even satisfy the process constraint on the propane composition (≥ 0.80) at steady state.

Now, the approach explained in section 4.3 is implemented. MA using transient measurements is applied until the steady-state is achieved and this method is not able to find a better operating point, in this case 4 hours, so a step of traditional modifier adaptation from steady state information is implemented until the real optimum is achieved as seen in Table 5.12.

Figure 5.34 shows the evolution of the process cost function. Figure 5.35 and Figure 5.36 show the evolution of the RTO decision variables, the set point of the steam and reflux flow controllers, with the evolution of the constraint on propane composition in distillate stream shown in Figure 5.37.

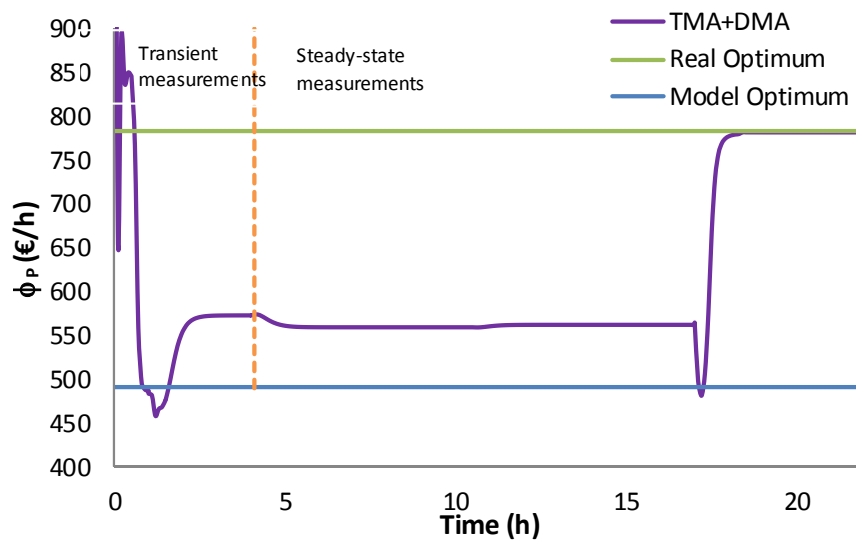


Figure 5.34. Evolution of the process cost function ϕ_p .

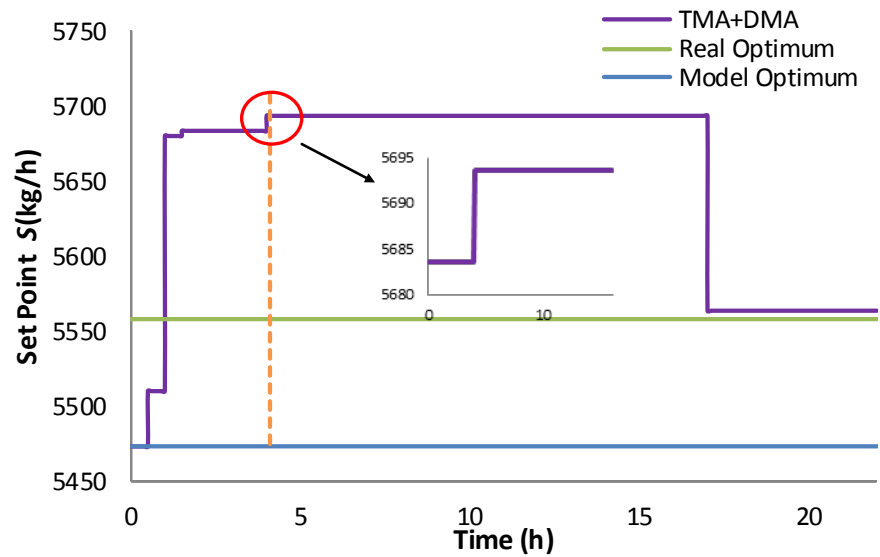


Figure 5.35. Evolution of the steam flow controller set point S .

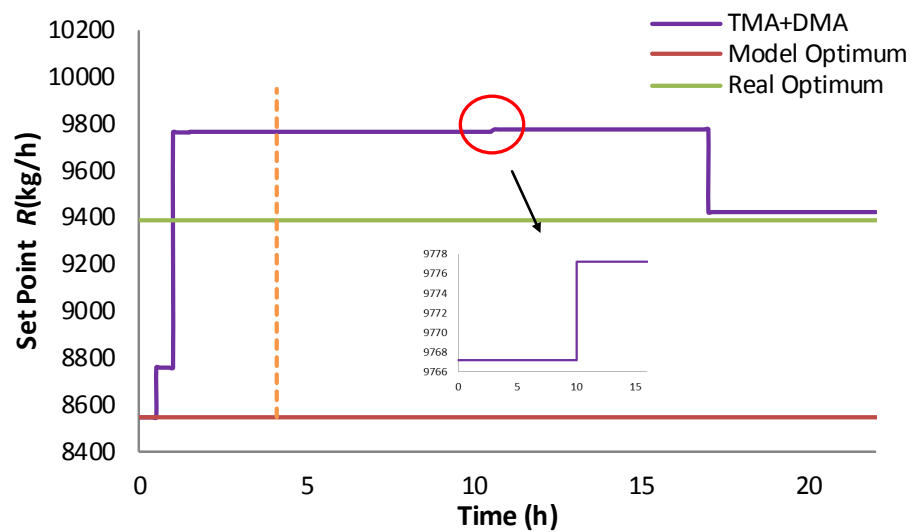


Figure 5.36. Evolution of the reflux flow controller set point R .

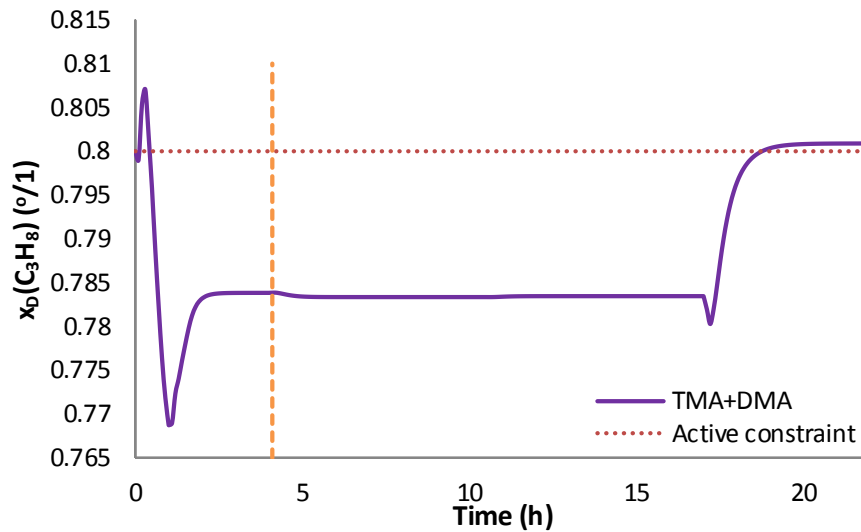


Figure 5.37. Evolution of the composition of propane in the distillate $x_D(\text{C}_3\text{H}_8)$.

The optimum operating point is achieved after 22 hours, which means a considerable time reduction compared to the traditional static MA applied in section 5.5.1. This period of time involves eight RTO during the transient and only one RTO executed during the steady state. However, three steady states are required, two to estimate process and model gradients and one steady state that is achieved after applying the obtained RTO solution. This approach shows a good performance, reducing the time required to achieve the real optimum of the process by 60% compared to the implementation of traditional static MA.

Table 5.12. Summary results.

	Real Optimum	TMA	TMA+DMA
$S(\text{kg/h})$	5558.20	5679.83	5590.26
$R(\text{kg/h})$	9389.67	9753.26	9523.33
$T_{head} (\text{°C})$	50.98	55.54	50.79
$T_{bottom} (\text{°C})$	102.02	103.34	102.03
$x_D(\text{C}_3\text{H}_8) (\text{°}/1)$	0.80	0.78	0.80
$\phi_P (\text{€/h})$	783.19	577.66	783.17

5.5.4 MA based on transient measurements

In this section, the results obtained by applying the MA techniques based on transient measurements to estimate process gradients are presented. In particular, the proposed method described in section 4.4 will be compared to the method presented in section 4.2 based on NEC. In this example, plant measurements are taken every hour without waiting for a steady-state and the RTO problem is updated at the same rate. This sample time corresponds to one sixth of the settling time of the process.

First, we present the results of the proposed method based on estimating process gradients using a recursive parameter identification algorithm, to be precise, Recursive Extended Least Squares (RELS), with a forgetting factor α equal to 0.99 and the lower bound for the degree of excitation, c_1 , of the system is fixed at 0.05. The lower bound for the degree of excitation, a , of the system is fixed at 0.02. The results obtained using this approach are shown in the figures as EstMA. In this example, the size of the data vector and the estimated parameters, given by (5.79) and (5.80), is five, so, after the fifth iteration, there is enough collected data to check if the persistent excitation condition is satisfied.

$$\varphi_k^T = [\Delta T_{bottom,k} \quad \Delta T_{head,k} \quad \Delta T_{bottom,k}^2 \quad \Delta T_{head,k}^2 \quad \Delta T_{bottom,k} \Delta T_{head,k}] \quad (5.79)$$

$$\theta_k^T = \left[\nabla_{T_{bottom,k}} \phi \quad \nabla_{T_{head,k}} \phi \quad \frac{1}{2} \nabla^2_{T_{bottom,k} T_{bottom,k}} \phi \quad \frac{1}{2} \nabla^2_{T_{head,k} T_{head,k}} \phi \quad \nabla^2_{T_{bottom,k} T_{head,k}} \phi \right] \quad (5.80)$$

For comparison purpose, the approach described in section 4.2, based on NEC to estimate process gradients is also applied. It needs to know where the parametric uncertainty between process and model is located. The use of the reduced steady-state model implies a strong structural plant model mismatch; for this reason, this mismatch has been adjusted to the rigorous one that represents the real process by adding some offsets that take into account the structural mismatch. In this case, the steady-states achieved by the dynamic process can be matched with those of the model by changing three uncertain parameters that have been considered as parametric uncertainty: two constants (c_{head} and c_{bottom}) added in the equations used to compute head and bottom temperatures, (5.42) and (5.43) with a nominal value of 0, being the easiest way to deal with it, but not the most efficient, and the global efficiency (E_f) of the column (5.41), whose nominal value is equal to 0.98, a value which has been adjusted from simulation results.

However, these parameters are time variant, so the implementation of this technique is possible and its performance is considerably improved, although the real optimum is not achieved. This task, to translate the structural uncertainty into parametric, could be a difficult task in some cases, so it is a clear disadvantage of this technique:

$$T_{head} = f_1(S, R) + c_{head} \quad (5.81)$$

$$T_{bottom} = f_2(S, R) + c_{bottom} \quad (5.82)$$

Therefore, when the NEC-based method is applied (see section 4.2), the vector of uncertain parameters is $\beta = [c_{head}, c_{bottom}, Ef]$, the decision variables (the model inputs) are $u = [R, S]$ and the vector of output variables $y = [B, D, x_B(C_3H_8)]$. It is possible to choose a different set of output variables, but the selected ones are the most sensitive to the uncertain parameters. The results using this method are shown in the figures as TMA.

The process starts from a certain operating point and, after one hour, the value of the optimal manipulated variables of the nominal solution is applied to the process, that is, the RTO solution without modifiers. Thus, both methodologies have the same transient information when they start working. In this example, modifiers have been applied without filtering, that is, $K_\lambda = K_\gamma = K_\epsilon = I$.

Figure 5.38 presents the evolution of the process cost function for both methods, whereas Figure 5.43 and Figure 5.42 show the bottom and head temperatures. The evolution of the composition of the distillate is shown in Figure 5.41, observing that the achieved operating point corresponds to an active constraint indicated by the dotted red line. Figure 5.42 and Figure 5.43 represent the steam, reflux and distillate flows respectively.

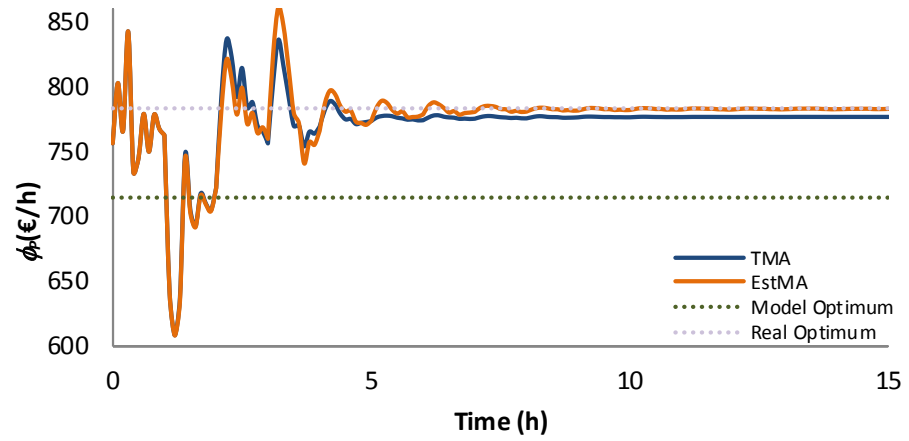


Figure 5.38. Evolution of the process cost function ϕ_p .

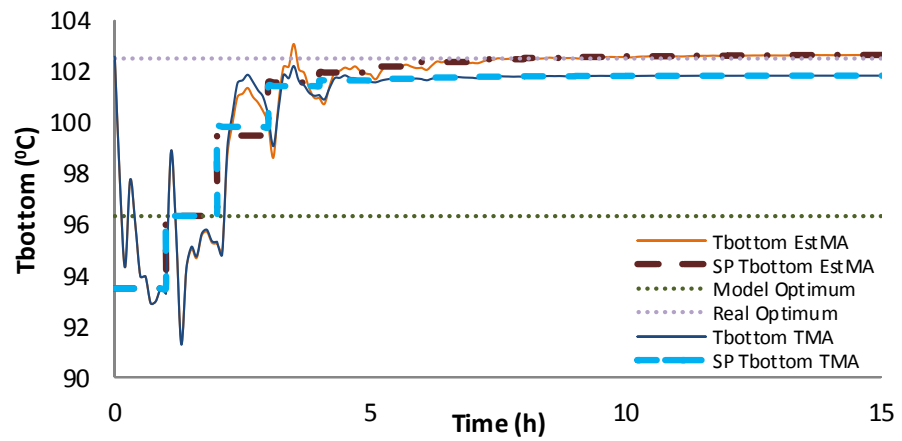


Figure 5.39. Evolution of the bottom temperature T_{bottom} .

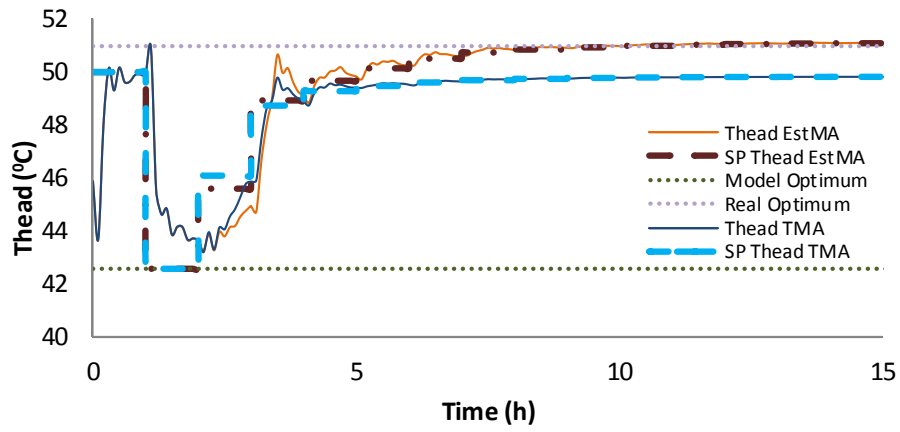


Figure 5.40. Evolution of the head temperature T_{head} .

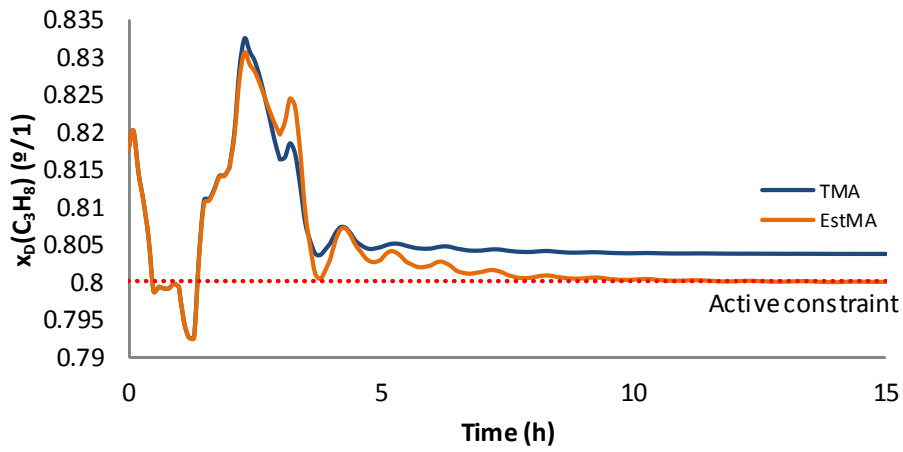


Figure 5.41. Evolution of the composition of propane in the distillate $x_D(C_3H_8)$.

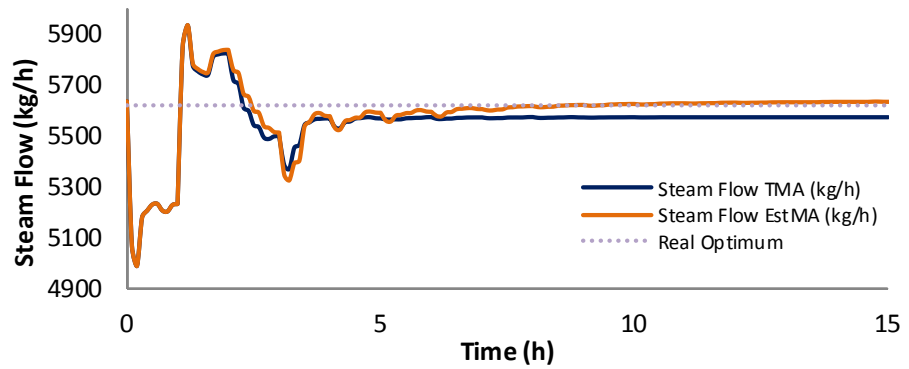


Figure 5.42. Evolution of the steam flow S .

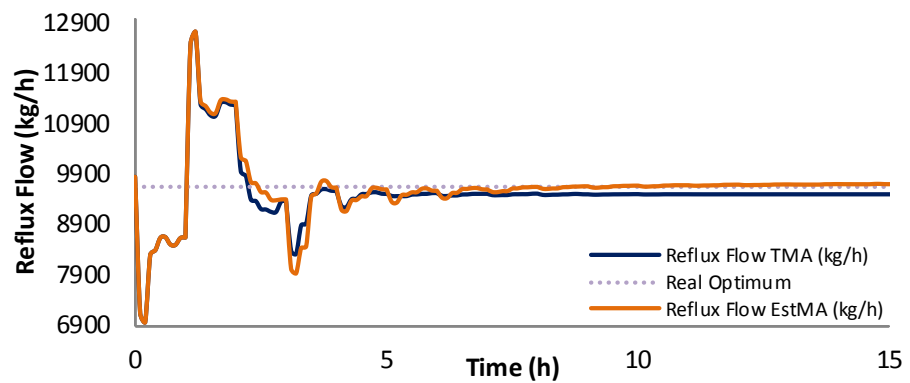


Figure 5.43. Evolution of the reflux flow R .

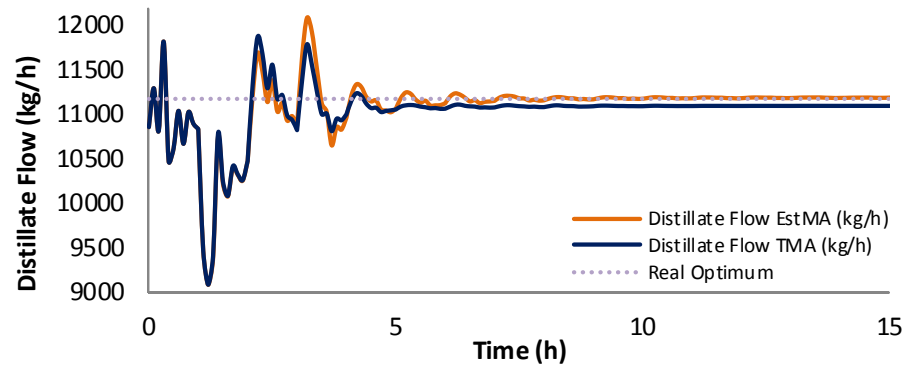


Figure 5.44. Evolution of the distillate flow D .

Observing the previous graphs (in particular Figure 5.38), one can see that the convergence of both methods to a region near the optimum (a tolerance band of 0.50% with respect to the process cost function has been considered) is faster than in the previous section, around 8 hours in both cases, which is a clear advantage that the approaches based on transient information present. However, notice that, due to the presence of structural uncertainty, there is an offset with respect to the real process optimum in the solution provided by the method based on NEC. This offset could be removed if once the steady-state is achieved a step of static MA is applied, as shown in section 5.5.3.

By contrast, when estimating the process gradients directly with the proposed recursive identification algorithm, the real optimum is achieved after one steady-state. Nevertheless, it must be pointed out that the settling time of the process has grown, since the DMC set-points were being changed continuously (every hour) during the transient.

By applying the RELS algorithm to update the modifiers during the transient, the optimum operating point is achieved after 8 hours, which involves 8 RTO solutions (one per hour), reducing by approximately 8 times the time required to achieve the optimum as compared with traditional static MA techniques, such as DMA or NMA.

5.5.4.1 Evaluation of the dual constraint in transient MA

A dual constraint has been added to the MA technique based on the estimation of the process gradients using RELS to ensure that, in the next RTO iteration, the system will have enough excitation to estimate the process gradients again. Figure 5.45 shows the evolution of the dual constraint in the implementation described in the section before. In this case, the lower limit for the dual constraint is 0.02.

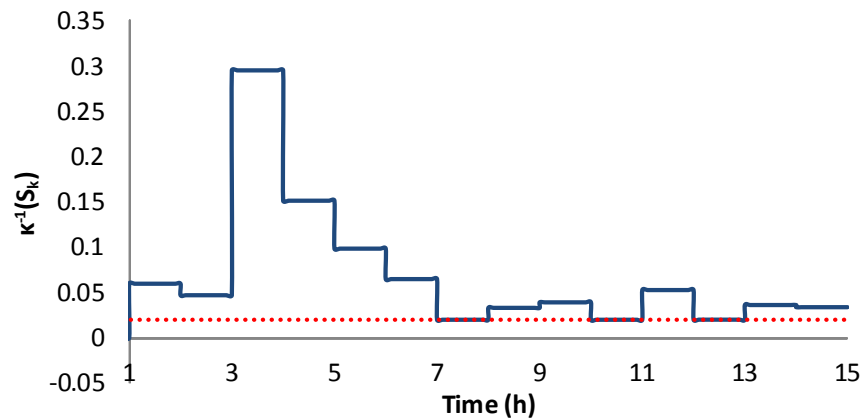


Figure 5.45. Evolution of the dual constraint.

In view of the results, it can be concluded that the system has enough excitation during the whole operation. In some iterations, the dual constraint becomes active. This is because, the RTO solution does not provide enough excitation, so the dual constraint acts to guarantee that the gradients will be properly estimated in the next RTO.

A second example is shown below to prove the effect of the dual constraint in a system that not is adequately excited. A very small constant filter has been considered, $\mathbf{K}_\lambda = \mathbf{K}_y = \mathbf{K}_\varepsilon = 0.15\mathbf{I}$, to force that situation, that is, the changes in the variables proposed by the RTO do not provide enough excitation to estimate the plant gradients accurately. In this situation, if excitation is forced by the dual constraint, an operating point that satisfies the NCO of the process can be achieved. In the following figures, RTO filtered, indicates the results obtained without adding the dual constraint and RTO dual, present the results adding the dual constraint to ensure excitation. Figure 5.46 presents the evolution of the process cost function, whereas Figure 5.47 and Figure 5.48 show the evolution of head and bottom temperatures, respectively. Figure 5.49 shows the value of the dual constraint whose lower limit has been fixed at 0.10.

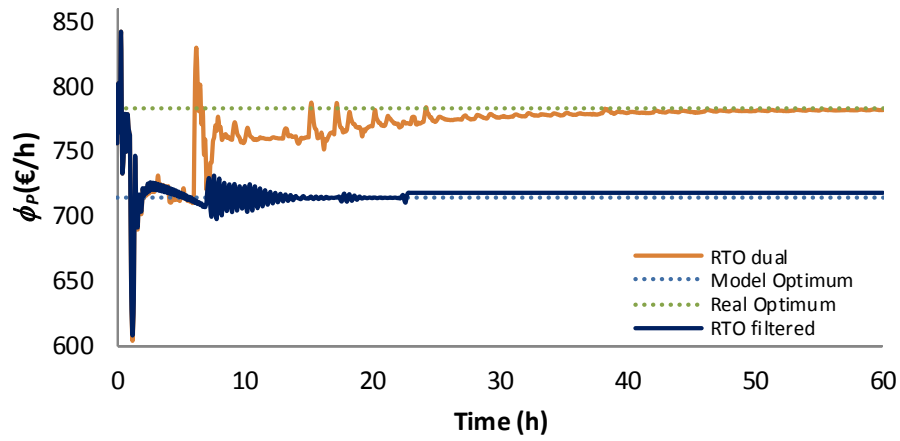


Figure 5.46. Evolution of the process cost function ϕ_p .

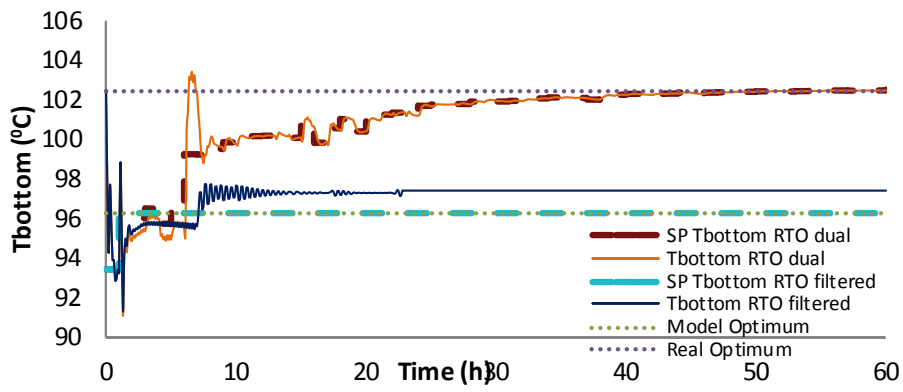


Figure 5.47. Evolution of the bottom temperature T_{bottom} .

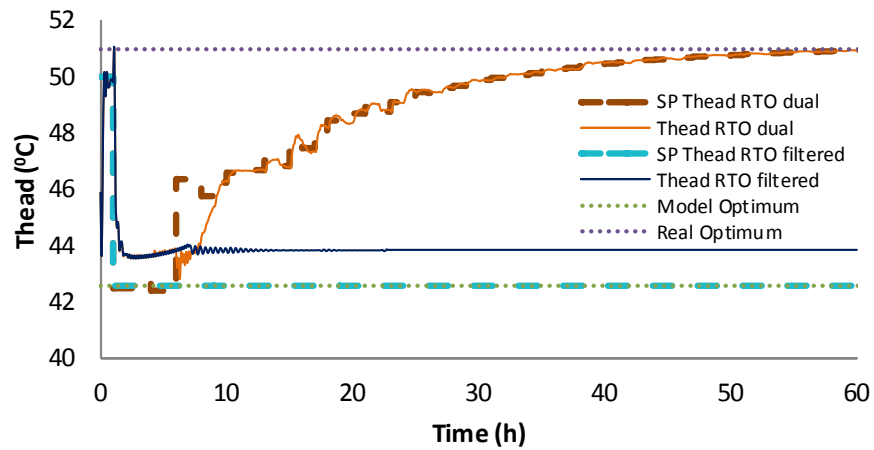


Figure 5.48. Evolution of the head temperature T_{head} .

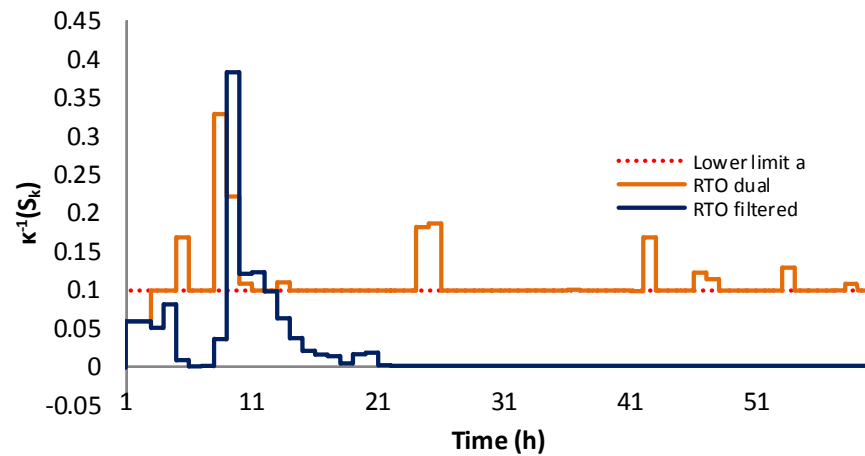


Figure 5.49. Evolution of the dual constraint.

The previous results show that the addition of the dual constraint, with an appropriate lower limit, provides enough excitation to the system. In this way, the transient MA method presented in this paper will be able to achieve an operating point that satisfies the NCO of the process, even if the RTO does not induce important changes in the decision variables by itself.

It is important to note that the convergence rate is strongly affected by the constant of the filters. In this last example, the optimum is achieved after 60 hours; this is due to the modifiers having been strongly filtered,

slowing down the convergence of the method compared with Figure 5.38-Figure 5.44.

5.6 Conclusions

In this chapter, we have analysed the performance of different MA approaches. One first conclusion is that most of them work well, but the developed method based on RELS is able to speed up the convergence of RTO-MA to the real plant optimum. It is based on transient information, obtaining process gradients directly from truncated Taylor expansions of the process cost and gradients combined with adaptive filtering estimation techniques. The method has been tested in a realistic case study corresponding to a depropanizer distillation column of a petrol refinery and the results obtained show that it is possible to effectively perform the optimization of the distillation column in the presence of structural plant-model mismatch and the presence of an MPC layer not considered explicitly in the RTO model. In the considered case study, the method reduces by a factor of 8 the time required to achieve the process optimum as compared with traditional static MA techniques, such as DMA or NMA.

This application also shows the clear advantages of using simplified steady-state models in the RTO layer; the solution time is significantly reduced during the optimization of the depropanizer column operation (approximately 28 times in this case), which allows the implementation of RTO in real, large-scale processes. In addition, the reduced model is easier to understand and easier to update and maintain if necessary.

Therefore, the combination of modifier adaptation methodology, the least squares algorithm to use transient measurements to compute process gradients, and the use of simplified models in the RTO layer, is really a powerful approach to achieve the optimal operating point of processes, reducing the convergence time.

Another effective way of speeding up the convergence of static MA consists of reducing the number of modifiers to be adapted at each RTO iteration. In this sense, the new NMA approach, based on the Lagrangian function, has been tested in this example, getting a faster convergence than usual NMA and DMA, and making the implementation of NMA easier in practice.

6 CASE STUDY II: NATURAL GAS NETWORKS

The second case study considered in this thesis is the optimal transport of natural gas through gas pipelines. The results obtained after applying MA to the RTO optimization problem are shown in this chapter.

6.1 Natural gas networks

Natural gas is becoming one of the most widely used sources of energy in the world due to its environmental friendly characteristics. Usually, the location of natural gas resources and the place where the gas is supplied to the consumer are far apart. This transport is carried out through pipeline network systems where the gas flows through pipes and various devices such as regulators, valves and compressors. During transport, the pressure of the gas is reduced mainly due to the friction with the wall of the pipe. The gas temperature also varies due to the heat transfer between the gas and the surroundings (Woldeyohannes & Majid, 2011).

A library of dynamic and rigorous components to simulate the behavior of natural gas flowing through pipelines has been developed in the simulation software EcosimPro (EAInt, 2013). The library is composed of natural gas pipelines, compressor stations, regulation and measurement stations, valves, turbines, etc.

This library can be used for different purposes such as: the optimal management of gas transport, training operators, detecting instrumentation failures, or predicting the behavior of the gas before changes in the operating conditions. The developed models are complex, since they are formed by partial differential equations that include variables that depend on the time and position along the pipeline. Due to the large size of these networks, according to the number of variables and equations needed to simulate the behavior of the gas, a simplified model has also been developed to be used for optimization purposes.

6.2 Gas network library¹

The natural gas network library comprises the following elements:

- Gas pipelines: high-pressure gas pipelines are channels for the transport of gas, made from carbon steel and with high levels of elasticity, the joints of which are welded. The maximum pressure in the pipeline is about 80 bar and the minimum 30 bar. The only exceptions are underwater sections that have a design pressure of 220 bar.

¹This library was developed at the Dpt. of Systems Engineering and Automatic control of the University of Valladolid within the project: "Modelado y supervision de redes de gas natural" together with Intergeo Tecnología and Aplein Ingenieros.

- Compressor stations: these are facilities located along a natural gas pipeline which compress the gas to a specified pressure, thereby allowing it to continue flowing along the pipeline to the intended recipient. The compression process is normally carried out by centrifugal compressors.
- Measurement and regulation stations: these are located at delivery points. The pressure is reduced to 16 bar in these stations as a means of starting the adaptation process to the final pressure used by companies and private individuals, which can be as low as 20 millibar. In this installation, the natural gas is filtered and heated if necessary, to compensate the future loss of temperature due to the gas expansion in the pressure regulator; the gas pressure is reduced in the reducer and immediately after that, the required variables are measured at fixed pressure and temperature conditions.

A natural gas network library has been developed in the modelling and simulation software EcosimPro (EA Int, 2013) that includes all the elements described above. A picture of the appearance of this library is shown in Figure 6.1. In addition to the gas pipelines model, other essential components for the simulation of the natural gas transport, such as the compressor stations (Figure 6.2), used to recover the pressure loss due to the transport through the gas networks, or the regulation and measurement stations (Figure 6.3), located in the supply points, where the gas pressure is reduced in order to initialize the adaptation process to the final pressure of the gas as used by companies and individuals.

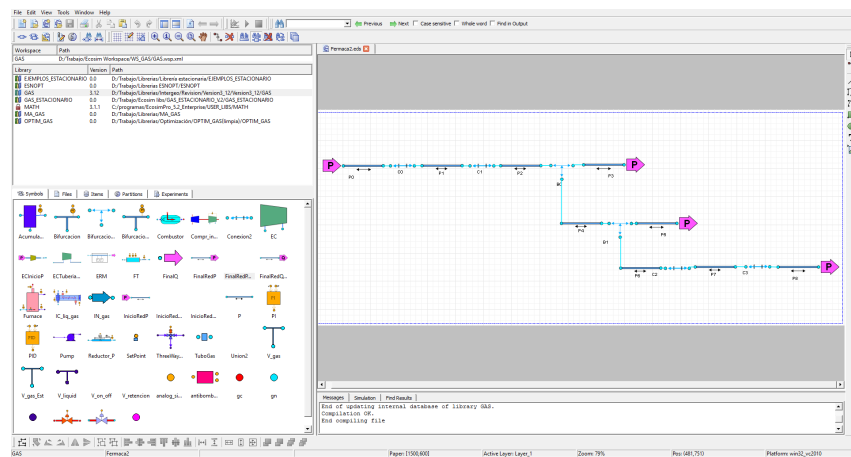


Figure 6.1. Dynamic library of natural gas networks in EcosimPro.

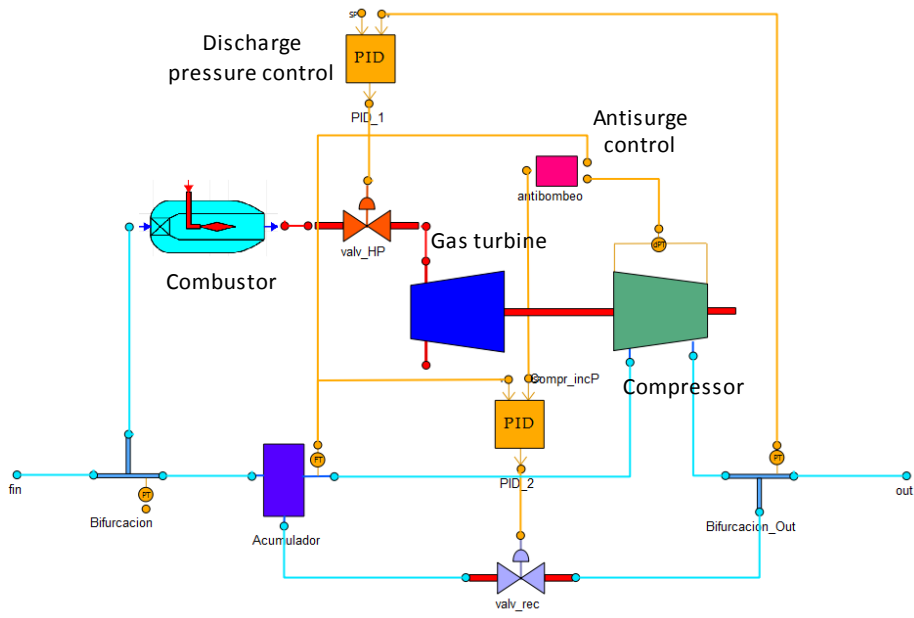


Figure 6.2. Schematic of the compressor station.

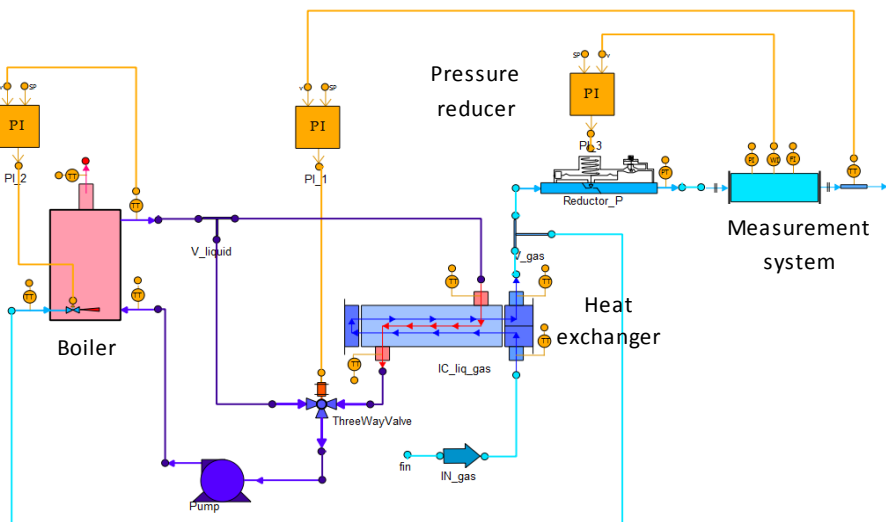


Figure 6.3. Schematic of the regulation and measurement station.

6.3 Gas network modelling

6.3.1 Compressor station modelling

One important part of the library is the modelling of the compressor stations. As natural gas flows along a pipeline, it slows due to friction between it and the pipeline. This results in a loss of pressure along the pipeline. In order to make the gas flow continuously at a desired flow rate, it is re-pressurized at suitable locations along the pipeline. This is done by mechanically compressing the gas at sites connected to the pipeline known as compressor stations. The location and quantity of compressor stations required in a pipeline system is dependent on a number of factors, including the operating pressure of the pipeline, the diameter of the pipe used, elevation changes along the pipeline route and the desired volume of gas to be transported.

The two main components of the compressor station are the gas turbine and the compressor, see Figure 6.2. The compressor mechanically re-pressurizes the gas in the pipeline using an impeller that converts the mechanical energy into kinetic, then a diffuser converts the kinetic energy into potential in pressure form. The energy required to drive the compressors is provided by the gas turbines that are mechanically coupled to the compressor impeller. The gas turbines are powered by a portion of the natural gas that flows through the pipeline.

In this section, we focus our attention on the modelling of the centrifugal compressor. Each compressor has a characteristic curve that is given by the manufacturer. This curve represents the iso-efficiency and iso-speed lines represented in the axis of the pressure difference (called height if it is expressed in meters of column of fluid) and the flow in abscissa.

The operating point of the compressor will be given by the intersection of the characteristic curve for a given speed of rotation, of the increase in pressure in the compressor and the flow passing through it.

However, from a practical point of view, not all points of the characteristic curve can be achieved. Figure 6.4 shows the operating margins that must be respected for the proper operation of the compressor (Mirsky, et al., 2012).

1. Surge line. Surge is the left hand boundary of the compressor map. Operation to the left of this line represents a region of flow instability.

It indicates the minimum load for each rotational speed. When the flow is reduced below the surge limit, the pressure at the discharge of the compressor exceeds the pressure-making capability of the compressor, causing a momentary reversal of flow. When this flow reversal occurs, the pressure of the discharge system is reduced, allowing the compressor to resume delivering flow until the discharge pressure again increases, and the surge cycle repeats. Surging usually creates a clearly audible noise. Prolonged operation in this unstable mode can cause serious mechanical damage to the compressor.

2. Maximum speed line. This is the maximum speed at which the compressor can work properly.
3. Minimum speed line. This is the minimum speed at which the compressor can work properly. Continuous operation beyond maximum and minimum speed is not allowed as mechanical strength limits may be reached when above maximum speed and the possible, unacceptably high levels of vibration beyond both maximum and minimum speeds.
4. Choke line. Choking of the centrifugal compressor occurs when the compressor is operating at low discharge pressure and very high flowrates. These high flowrates at compressor choke point are actually the maximum that the compressor can push through.
5. Maximum power line. This represents the limit above which the turbine is not able to give more power to the compressor.

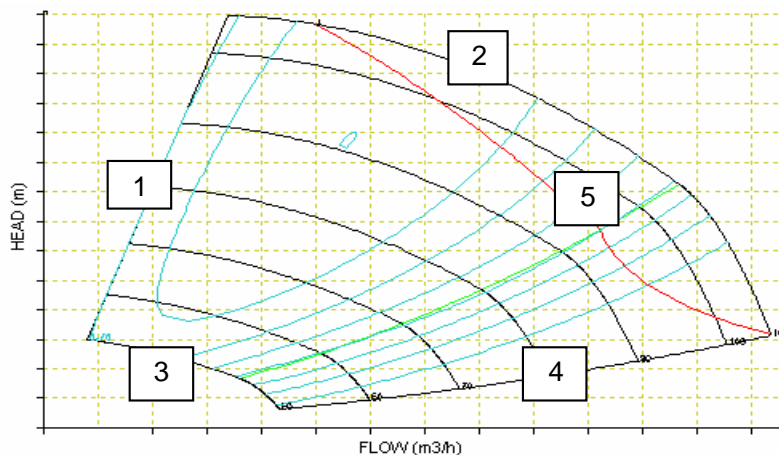


Figure 6.4. Characteristic curve of a centrifugal compressor.

A dynamic model has been developed for the centrifugal compressor. This model is based on mass and energy balances and the pressure profiles.

Mass balance

It is assumed there is no mass accumulation in the compressor, so mass balance is expressed by (6.1).

$$F_{in} = F_{out} = F \quad (6.1)$$

where F_{in} and F_{out} (kg/s) are the inlet and outlet mass flow rates of natural gas at the compressor.

Energy balance

The compression process is considered an adiabatic transformation, since the heat transmission process is very slow compared with the compression process so, the energy balance does not consider any term of heat loss to the ambient ($Q_{amb}=0$).

$$\frac{dE}{dt} = Q_{amb} + P + F_{in} \left(h_{in} + \frac{1}{2} v_{in}^2 + gz_{in} \right) - F_{out} \left(h_{out} + \frac{1}{2} v_{out}^2 + gz_{out} \right) \quad (6.2)$$

where E (kJ) is the compressor energy, Q_{amb} (kW) is the heat exchanged with the ambient, P (kW) is the power supplied to the compressor, h (kJ/kg) is the natural gas enthalpy, v is the gas velocity (m/s), z (m) is the height and g (m/s²) is the gravity.

The variation of the compressor speed is computed as the difference between the turbine (τ_t) and compressor torque (τ_c):

$$J \frac{dN}{dt} = \tau_t - \tau_c = \frac{P_t}{w} - \frac{P}{N} \quad (6.3)$$

where J (kg m²) is the moment of inertia, P_t is the turbine power (kW), and w (rad/s), N (rps) are the rotational speed of the compressor expressed in different units.

The changes in the kinetic and potential energy are negligible, as well as the heat loss to the environment, so the resulting energy balance is given by (6.4).

$$\frac{dE}{dt} = P + F_{in}(h_{in} - h_{out}) \quad (6.4)$$

Besides, the compressor energy E is a combination of mechanical and thermal energy:

$$E = \frac{1}{2} J W^2 + mU = 2\pi^2 J N^2 + mU \quad (6.5)$$

where U (kJ/kg) is the internal energy of the natural gas and m (kg) is the compressed gas mass.

Then, differentiating E from equation (6.5) and substituting in (6.4), equation (6.6) is obtained:

$$4\pi^2 J \frac{dN}{dt} + \frac{d(mU)}{dt} = P + F(h_{in} - h_{out}) \quad (6.6)$$

In addition, taking into account the definitions for the internal energy (6.7) and the variation of enthalpy (6.8), the global energy balance of the compressor is defined by (6.9):

$$\frac{d(mU)}{dt} = mC_v \frac{dT}{dt} + mT \frac{dC_v}{dt} \quad (6.7)$$

$$h_{in} - h_{out} = C_p(T_{in} - T_{out}) \quad (6.8)$$

$$4\pi^2 J \frac{dN}{dt} + mC_v \frac{dT}{dt} + mT \frac{dC_v}{dt} = P + F(h_{in} - h_{out}) \quad (6.9)$$

where C_p (kJ/kmol) is the calorific capacity of the natural gas at constant pressure, C_v (kJ/kmol) is the calorific capacity of the natural gas at constant volume and T (K) is the gas temperature.

The power supplied to the compressor (P) is obtained from (6.10).

$$P = W_a \cdot F \quad (6.10)$$

where W_a (kJ/kg) is the real work of the compressor computed by

$$-W_a = \frac{-W_p}{\eta_p} \quad (6.11)$$

where η_p is the polytropic efficiency.

The work performed by the compressor is computed from (6.12).

$$-W_p = Z_m \cdot \frac{R \cdot n \cdot T_{in}}{M \cdot (n - 1)} \cdot \left[\left(\frac{P_c}{P_{in}} \right)^{\frac{n-1}{n}} - 1 \right] \quad (6.12)$$

where W_p (kJ/kg) is the polytropic work absorbed by the gas, Z_m is the average compressibility factor (explicitly computed by the Sarem method (Pitzer, 1955)), R (J/K mol) is the ideal gas constant, M (g/mol) is the molecular weight of the gas and n is the polytropic coefficient that normally varies from 1.5 to 1.6 and whose value has been chosen as 1.5 in this model.

Pressure variation at the compressor

The variation in the pressure increase at the compressor is given by (6.13). The pressure increase produced by the compressor is a function of the compressor speed and the flow passing through the compressor. This variation of pressure is obtained by interpolating the gas volumetric flow Q and the speed in the characteristic map of the compressor.

$$\Delta P = f(Q, N) \quad (6.13)$$

Finally, the discharge pressure of the compressor P_c (Pa) is defined by (6.14) where P_{in} (Pa) is the suction pressure and ΔP the pressure increase at the compressor.

$$P_c = P_{in} + \Delta P \quad (6.14)$$

Gas turbine modelling

The energy required to drive the compressor is provided by the gas turbine, which is mechanically coupled to the compressor impeller. The gas turbine is powered by a portion of the natural gas that flows through the pipeline. A simplified model has been developed for the turbine that includes the mass balance (6.15) and the computation of the turbine power P_t (6.16).

$$F_{gc,in} = F_{gc,out} = F_{gc} \quad (6.15)$$

$$P_t = F_{gc} C_{p,gc} (T_{gc.in} - T_{gc.out}) \quad (6.16)$$

where $F_{gc,in}$ and $F_{gc,out}$ (kg/s) are the inlet and outlet mass flow rates of combustion gas at the turbine, $C_{p,gc}$ (kJ/kg) is the calorific capacity at constant pressure and $T_{gc,in}$, $T_{gc,out}$ ($^{\circ}$ C) are the inlet and outlet temperatures of the combustion gases.

The gas turbines have a characteristic curve which relates the gas flow passing through the turbine F_{gc} , the rotational speed of the compressor N_t (rpm) and the pressure difference between the inlet and the outlet ΔP_t (Pa). Assuming that this pressure decrease is perfectly controlled, it has been considered a curve that relates the rotation speed as a function of the gas consumed.

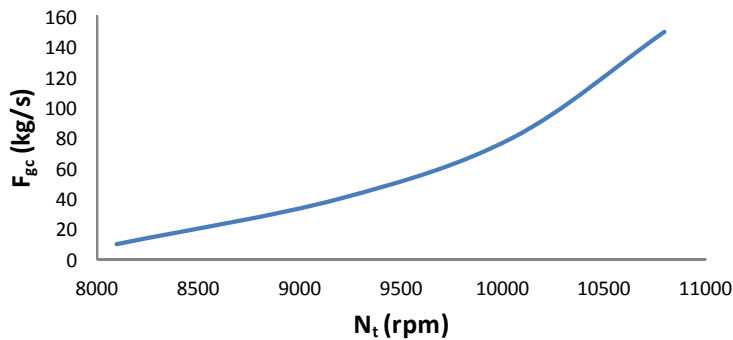


Figure 6.5. Characteristic curve of the turbine.

The combustion between the natural gas taken from the gas pipeline and the compressed air is carried out at the combustor. Therefore, the inlet gas comes from the gas network and the outlet combustion gases are expanded in the turbine.

The fraction of natural gas taken from the gas network required to make the combustion is obtained taking into account a typical air/fuel mass ratio whose value is 16:1. Another typical mass ratio relates the mass flow rate of combustion gases with the mass flow rate of natural gas entering the turbine. This ratio is supposed to be 17:1.

Valve modelling

The equation (6.17) is used to compute the mass flow circulation through a valve, where K_v (m^3/h) is the valve flow coefficient, u_v (%) is the valve opening, $P_{v,in}$ and $P_{v,out}$ (Pa) are the inlet and outlet pressure and ρ_v (kg/m^3) is the fluid density. The volumetric flow Q_v (m^3/s) passing through the valve is given by (6.18).

$$F_v = K_v \cdot (u_v / 100) \cdot \sqrt{P_{v,in} \cdot \rho_v \cdot (1 - P_{v,out}/P_{v,in})} \quad (6.17)$$

$$Q_v = F_v / \rho_v \quad (6.18)$$

There are two main valves in the compressor station, the valve located at the inlet of the turbine which circulates the combustion gases, whose opening is u_{valv_HP} and the recycle valve of the antisurge control, whose opening signal is u_{valv_rec} .

Compressor discharge pressure control

The typical control structure for compressor stations has also been modelled. This structure includes the control of the discharge pressure and the antisurge control (Acedo Sánchez, 2003).

The discharge pressure of the centrifugal compressor is controlled by varying the rotational speed of the turbine. This speed is adjusted by modifying the valve opening that allows the inlet of combustion gases to the turbine (u_{valv_HP}), as shown in Figure 6.2.

When the discharge pressure decreases, the control system will open the valve to allow a larger flow of gas entering the turbine, increasing the rotational speed of the turbine, also therefore increasing also the compressor speed.

Compressor antisurge control

As mentioned above, the phenomenon of surge occurs when the flow is reduced to below the surge limit. The compressor surge is dangerous because it causes the compressor to vibrate and is detrimental because it causes damage to the compressor parts. To prevent this problem, centrifugal compressors have an antisurge controller.

Antisurge is an inferential control, so the first step in designing the control system is to obtain the surge line equation. The equation of the line

with slope K and independent term $B0$ is obtained by computing the corresponding straight line passing through two points on the map, located near the working rotational speed, for example, 80-100% of the nominal speed, as can be observed in Figure 6.6.

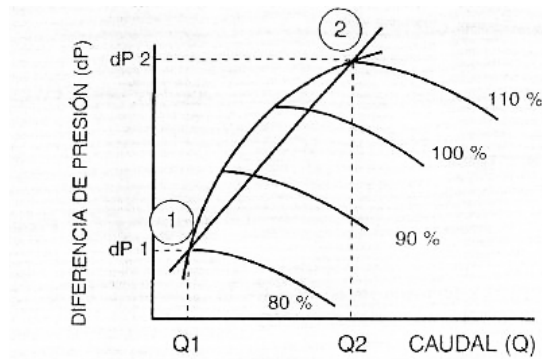


Figure 6.6. Surge line.

The equation for the surge line is defined as follows:

$$Q = K \cdot \Delta P + B0 \quad (6.19)$$

The control system detects when a process compression stage is approaching surge and subsequently takes action to reverse the movement of the operating point towards the surge line. This decreases the pressure and increases the flow through the compressor, resulting in stable working conditions. It is normally achieved by opening a control valve in a recycle line (Figure 6.2), returning the discharge gas to the inlet of the compressor. The resulting increase in compressor inlet volume flow moves the operating point away from surge.

Due to inaccuracies in measurements and response times of transmitters and valves, antisurge control achieves a surge control line parallel to the surge limit line. The control line is offset to the right of the surge line by a margin; typically equal to 3-10% of inlet volume flow at surge (Figure 6.7). However, a lower margin is also desirable, as higher efficiency could be obtained by closing the recycle valve (Ghazanfarihashemi & Ghanbariannaeni, 2012).

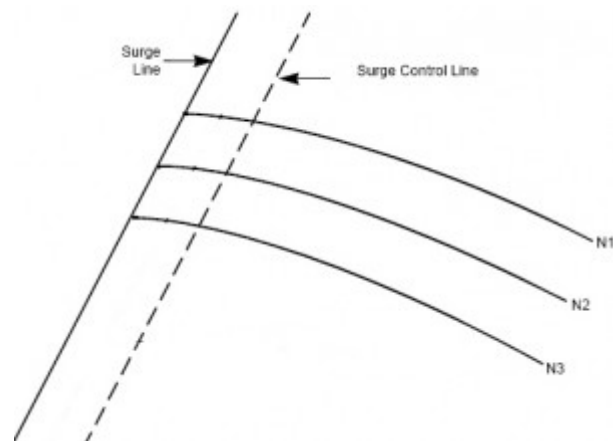


Figure 6.7. Surge control line.

This control system comprises a PID controller, shown in Figure 6.2, which receives the measurement of the controlled variable (gas flow entering the compressor). This is compared to the set point (minimum inlet flow) and sends the corresponding control signal to the recycle valve (`valv_rec`) to modify its opening and recycle the required flow to guarantee compressor safety.

6.3.2 Gas pipeline dynamic rigorous modelling

This section is focused on the modelling of gas pipelines, the main element of natural gas transport. A rigorous gas pipeline model must include the following mass balances, energy and momentum.

General assumptions

To know the dynamic of the model variables (mass, composition, velocity, pressure and temperature) as a function of both time t and the longitudinal coordinate of the gas pipeline x , the corresponding mass balance, momentum and energy are considered. It is supposed that the variables remain constant in the radial direction and that natural gas is composed of 12 components: methane (CH_4), ethane (C_2H_6), propane (C_3H_8), *i*-butane ($\text{i-C}_4\text{H}_{10}$), *n*-butane ($\text{n-C}_4\text{H}_{10}$), *i*-pentane ($\text{i-C}_5\text{H}_{12}$), *n*-pentane ($\text{n-C}_5\text{H}_{12}$), hexane (C_6H_{14}), nitrogen (N_2), heptane (C_7H_{16}), carbon dioxide (CO_2) and hydrogen sulfide (H_2S). The model yields an accurate prediction, since all these components are considered. To know this composition is very

important to compute important variables, such as the calorific power of natural gas that determines its economic value (Rodríguez, et al., 2013).

Mass balances

The global mass balance for the gas pipeline described by (6.20) is formed by an accumulation term and a convection term. The natural gas mass m (kg) and the velocity v (m/s) depend on both the time t (s) and the longitudinal coordinate of the pipeline x (m):

$$\frac{\partial m}{\partial t} + \frac{\partial(mv)}{\partial x} = 0 \quad (6.20)$$

The individual mass balance for each component is given by (6.21), which takes into account the accumulation and convection terms for all components $i = 1, \dots, M-1$, except one (normally the one with a high concentration, in this case, methane) whose composition is computed by (6.22):

$$\frac{\partial(my_i)}{\partial t} + \frac{\partial(mvy_i)}{\partial x} = 0 \quad \forall i = 1, \dots, M-1 \quad (6.21)$$

$$1 = \sum_{i=1}^M y_i \quad (6.22)$$

where M is the total number of components and y_i ($^{\circ}/1$) is the mass fraction of each component i .

Energy balance

The total energy balance is given by (6.23) (internal, kinetic and potential energy), assuming that the temperature loss is due to the conduction between the pipe wall and the gas along the pipeline together with the conduction between soil and pipe wall. In addition, heat transmission by convection along the pipeline is also considered.

$$\frac{\partial}{\partial t} \left[m \left(c_v T + \frac{v^2}{2} + gh \right) \right] + \frac{\partial}{\partial x} \left[mv \left(c_v T + \frac{p}{\rho} + \frac{v^2}{2} + gh \right) \right] = kV \frac{\partial^2 T}{\partial x^2} + U_{ground} A (T_{soil} - T) \quad (6.23)$$

where ρ (kg/m³) is the gas density, P (Pa) the gas pressure, T (°C) the gas temperature, T_{soil} (°C) the soil temperature, C_v (J/kg °C) the gas heat capacity at constant volume, k (W/m °C) the thermal conductivity of the natural gas, U_{ground} (W/m² °C) the heat transfer coefficient through the pipe wall, A (m²) the heat transmission surface, h (m) the altitude difference between pipeline ends, V (m³) the volume of the pipeline and g (m/s²) is the gravity.

Momentum balance

The momentum balance is described by (6.24), which takes into account the accumulation term, convection, head loss due to the pressure, head loss due to the friction (F_{roz}) and head loss due to the slope of the pipeline (F_{grav}). All these forces are represented in Figure 6.8.

$$\frac{\partial(mv)}{\partial t} + \frac{\partial(mv^2)}{\partial x} + V \frac{\partial P}{\partial x} = -m \frac{v|v|}{2d} 4\phi - mg \frac{h}{L} \quad (6.24)$$

where d (m) is the inside diameter, L (m) is the length of the pipeline and ϕ is the friction factor computed by using the Chen equation defined by (6.25).

$$\frac{1}{\sqrt{\phi}} = -4 \log \left(\frac{rg/d}{3.7065} - \frac{5.0452}{Re} \log \left(\frac{(rg/d)^{1.1096}}{2.8257} + \left(\frac{7.149}{Re} \right)^{0.8961} \right) \right) \quad (6.25)$$

where Re is the Reynolds number computed by (6.26) and rg (m) the pipe roughness.

$$Re = \frac{dv\rho}{\mu} \quad (6.26)$$

where μ (kg/m s) is the gas viscosity.

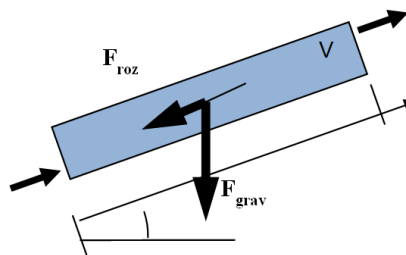


Figure 6.8. Scheme of forces in the pipeline.

Equation of state for real gases

Natural gas is a compressible fluid with a non-ideal behavior described by the equation of state for real gases defined by (6.27):

$$PV = ZnRT \quad (6.27)$$

where Z is the compressibility factor, which in this case is computed using the Sarem method (Pitzer, 1955), n (mol) is the number of moles and R (J/K mol) is the universal gas constant.

Spatial discretization. Finite volume method

To solve the partial differential equations it is necessary to use an integration method. In this case, the finite volume method has been chosen. This technique integrates the original differential equations on a finite volume obtaining a balance for each of the discrete zones into which the pipeline is divided, called nodes (Baliga & Patankar, 1983).

When applying this method for the discretization, it has been supposed that the volumetric flow circulating through each node takes the value of the steady-state flow given by (6.28), the fundamental gas flow equation at steady state (Schroeder, 2001), since the considered volumes are sufficiently small to make this assumption. Thus, the momentum balance is replaced by (6.28), simplifying the use of the model because of ignoring the complex partial differential equation (6.24). The mass and energy balances described above are applied over each control volume.

$$Q = C \cdot \frac{T_b}{P_b} \cdot d^{2.5} \cdot e \cdot \left(\frac{P_{in}^2 - P_{out}^2 - H_c}{L \cdot G \cdot T_a \cdot Z_a \cdot \phi} \right)^{0.5} \quad (6.28)$$

C : constant; 0.011493 (metric units)

e : pipeline efficiency

G : specific gravity

d : inside diameter (mm)

L : pipeline length (km)

P_{base} : base pressure (kPa)

P_{in} : inlet gas pressure (kPa)

P_{out} : outlet gas pressure pipeline (kPa)

Q : volumetric flow (m³/day, in standard conditions)

T_a : average temperature (K) (geometric mean between the inlet and outlet temperatures)

T_{base} : base temperature (K)

Z_a : average compressibility factor (computed from T_a and P_a)

Pipes are not usually horizontal (inlet height $H_1 \neq$ outlet height H_2). So long as the slope is not too great, a correction for the static head of fluid H_c is incorporated and determined as follows:

$$H_c = \frac{0.06835g(H_2 - H_1)P_a^2}{ZT_a} \quad (6.29)$$

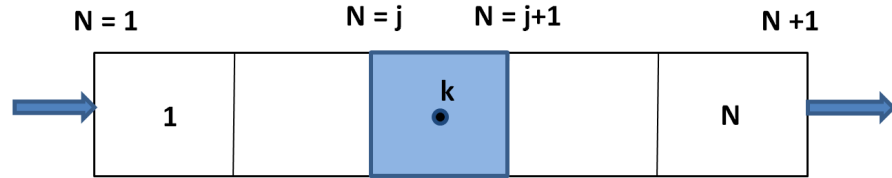


Figure 6.9. Pipeline dividing the spatial domain into finite volumes.

By applying the finite volume method, the model variables can be divided into two groups. The variables that are computed at the borders of the volume N , $x[j]$ ($j = 1, \dots, N+1$), and those that are calculated at the centre of the finite volume $x[k]$ ($k = 1, \dots, N$).

In this way, the mass balance described in (6.20) is replaced by (6.30), which is applied to each finite volume and represents the mass accumulation $dm[k]/dt$ resulting from the difference between the inlet mass flow $W[j]$ (kg/s) and the outlet mass flow $W[j+1]$.

$$\frac{dm[k]}{dt} = W[j] - W[j+1] \quad (6.30)$$

The individual mass balance, given by (6.21), is replaced by the solution of (6.31) at each volume, where $yc_i[k]$ is the composition of each component i ($i=1, \dots, M$) at each volume, and $ycb_i[k]$ is the composition at the volume border.

$$\frac{dm[k]}{dt} \cdot yc_i[k] + m[k] \frac{dyc_i[k]}{dt} = W[j]ycb_i[j] - W[j+1]ycb_i[j+1] \quad (6.31)$$

The individual mass balance (6.31) is applied for each component i at each volume, except for the component of highest composition, which is computed by (6.32)

$$1 = \sum_1^M y_{c_i}[k] \tag{6.32}$$

The modelled gas pipelines allow the inverse flow that can happen in some situations. Because of this issue, the variables at the border y_{cb} , T_b ($^{\circ}\text{C}$) are included. These variables can take one or other value, depending on the flow direction.

If the flow direction is positive, that is, the flow goes from volume 1 to N ($W[j] > 0$), then:

$$y_{cb_i}[j] = y_{c_i}[k - 1] \tag{6.33}$$

If the flow direction is negative, that is, the flow goes from volume N to 1 ($W[j] < 0$), then:

$$y_{cb_i}[j] = y_{c_i}[k] \tag{6.34}$$

To sum up, the variables at the volume border take the value of the previous volume, and this volume is chosen as a function of the flow direction, as shown in Figure 6.10 and Figure 6.11. This also happens for the temperature $T_b[j]$ and the gas density $\rho_b[j]$ (kg/m^3).

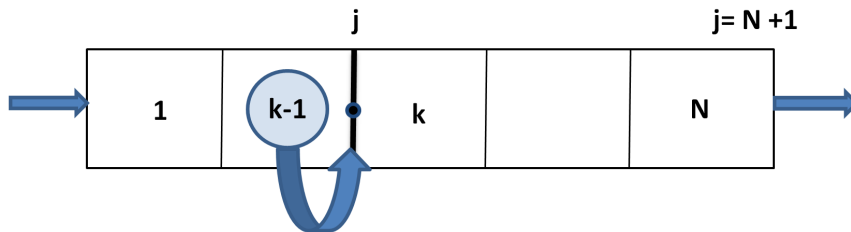


Figure 6.10. Value election when the flow direction is positive.

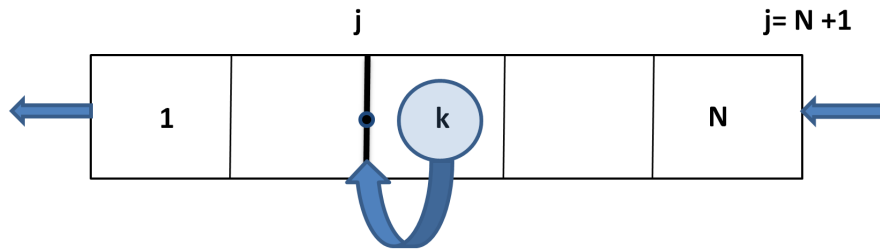


Figure 6.11. Value election when the flow direction is negative.

A certain dynamic, (6.35) and (6.36), is added to smooth the changes in the average values $x_b[j]$ when a change of flow direction happens. The time

constant ζ corresponds to the inverse of the absolute value of the gas velocity $v[j]$ (m/s).

$$W[j] > 0 \quad \zeta \frac{dx[j]}{dt} = x[k-1] - x_b[j] \quad (6.35)$$

$$W[j] < 0 \quad \zeta \frac{dx[j]}{dt} = x[k] - x_b[j] \quad (6.36)$$

The mass flow passing through each volume is computed using the fundamental gas flow equation at steady state (6.28), which is applied to each volume in the following form:

$$W[j] = Pm_a[k] \cdot C \cdot \frac{T_{base}}{P_{base}} \cdot d^{2.5} \cdot e \cdot \left(\frac{P[k-1]^2 - P[k]^2 - H_c}{L / N \cdot G \cdot T_a[j] \cdot Z_a[j] \cdot \phi[j]} \right) \quad (6.37)$$

The energy balance described by (6.23) results in (6.38) after discretization:

$$\begin{aligned} m[k] \left(\frac{dC_v[k]}{dT} T[k] + C_v[k] \right) \frac{dT[k]}{dt} = \\ W[j] \left(C_{va}[j] T_b[j] - C_v[k] T[k] + \left(\frac{Q[j]^2}{2} A^2 \right) - gh_a[j] \right) \\ + (P_a[j] Q[j] - P_a[j+1] Q[j+1]) + U_{ground} \text{areal} (T_{soil} - T[k]) \\ - W[j+1] \left(C_{va}[j+1] T_b[j+1] - C_v[k] T[k] + \left(\frac{Q[j+1]^2}{2} A^2 \right) + gh_a[j+1] \right) \end{aligned} \quad (6.38)$$

This library has been used to simulate real gas networks in the simulation software EcosimPro. These networks are constructed by adding the different components (compressors, gas pipelines, valves,...) linked by ports in a schematic. The characteristics of the gas networks, such as length, diameter or roughness and feed gas composition can be fixed into the schematic whereas the boundary conditions required to simulate the gas behavior through the gas pipelines are defined in the experiment.

6.4 Implementation of MA in Reganosa network

As natural gas pipeline systems have grown larger and more complex, the importance of the optimal operation and planning of these facilities has increased. The investment costs and operation expenses of pipeline networks are so large that even small improvements in system utilization can involve substantial amounts of money (Borraz-Sánchez & Mercado, 2015). For this reason, the implementation of MA in this field would bring important benefits.

In a gas transmission network, the overall operating cost of the system is highly dependent upon the operating cost of the compressor stations in a network. This operating cost is generally measured by the fuel consumed at the compressor station. According to (Luongo, et al., 1989), the operating cost of running the compressor stations represents between 25% and 50% of the total operating budget of the company. Hence, the objective for a transmission network is to minimize the total fuel consumption of the compressor stations, while satisfying specified delivery flow rates.

6.4.1 Reganosa network

The case study considered in this thesis involves a transmission gas network where there are consumers that need a certain amount of gas at a specified quality and pressure. There are also gas sources where the fluid is supplied at a determined pressure; in addition, there are two compressors operating.

Figure 6.12 shows the natural gas network whose operation will be optimized to reduce the costs due to gas consumption in the compressor stations. This example is based on a real gas network managed by Reganosa S. A., situated in the north of Spain, with a total length of 130 km and four measurement stations (Reganosa S.A., 2013). It has one source and six demand points; two compressor stations have been included, one of them at the supply point. They are the controllable units to get the normalized flow demanded by every consumer while minimizing the transport cost.

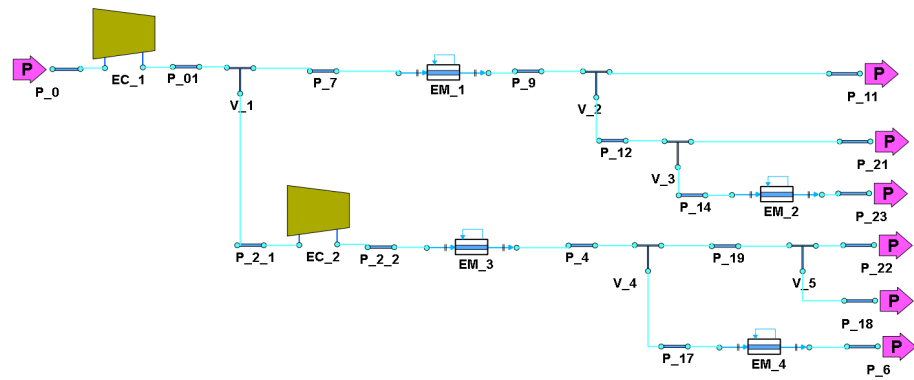


Figure 6.12. Reganosa gas network topology.

The main characteristics of the Reganosa network are shown in Table 6.1.

Table 6.1. Dimensions of the Reganosa gas network.

Pipe name	L (m)	d (m)	rg (m)	H_2-H_1 (m)
P_0	10000	0.7620	$4.6 \cdot 10^{-5}$	250
P_2	6650	0.6604	$4.6 \cdot 10^{-5}$	240
P_4	15650	0.6604	$4.6 \cdot 10^{-5}$	10
P_6	150	0.5080	$4.6 \cdot 10^{-5}$	0
P_7	9500	0.6604	$4.6 \cdot 10^{-5}$	-210
P_9	19700	0.6604	$4.6 \cdot 10^{-5}$	40
P_11	50	0.5080	$4.6 \cdot 10^{-5}$	30
P_12	14900	0.4064	$4.6 \cdot 10^{-5}$	320
P_14	26250	0.4064	$4.6 \cdot 10^{-5}$	-300
P_17	18000	0.5080	$4.6 \cdot 10^{-5}$	120
P_18	50	0.6604	$4.6 \cdot 10^{-5}$	0
P_19	50	0.6604	$4.6 \cdot 10^{-5}$	0
P_21	4600	0.2540	$4.6 \cdot 10^{-5}$	0
P_22	1550	0.4064	$4.6 \cdot 10^{-5}$	-10
P_23	5650	0.4064	$4.6 \cdot 10^{-5}$	10

6.4.2 Simplified RTO model

The model described in the previous section is too complex to be used for optimization. For this reason, the rigorous model for the gas pipelines has been simplified for use in the RTO layer.

A steady-state model has been considered for the gas pipelines, so this model only considers one finite volume, that is, only the value of the variables at the inlet and outlet are taken into account, without considering the dynamic of the variables along the pipe.

Since the computation of the compressibility factor using the Sarem method slows down the simulation considerably, the RTO model assumes that an ideal gas is transported, so the compressibility factor is equal to one.

The model for the compressor stations has also been simplified, since the control structure of the installation has been omitted. Antisurge control and discharge pressure control are omitted in the RTO model, so the input of the model is the signal to the valve of the combustion gases entering the turbine whose value, in the real process, is controlled by a PID controller. The opening for the recycle valve is determined as a function of the difference between the gas flow passing through the compressor and the minimum flow determined by the antisurge line. In this way, the degree of freedom of the RTO model is the opening of the valve of the combustion gases entering the turbine in each compressor.

In addition, a different value of the pipe efficiency e has been considered in both models. This constant value directly affects the computation of the gas flow (6.28). Its value in the rigorous model is 0.42, whereas the value of e in the RTO model is 0.48.

The size comparison between the rigorous model and the simplified model used in the RTO layer is presented in Table 6.2.

Table 6.2. Size comparison between the rigorous and the reduced model.

	Rigorous model	RTO model
Number of equations	11744	2391
Number of explicit variables	7694	2276
Number of derivative variables	4050	115
Number of boundaries	22	22

As can be seen in Table 6.2, the model size is reduced by approximately 5 times.

The boundary variables for both models are: inlet gas composition (12), inlet temperature (1), inlet pressure (1) and the pressure at each outlet (6).

The rigorous model requires two more boundaries, the set point for the discharge pressure controller at each compressor station (2); whereas, since the control layer has been omitted, the reduced model requires two more boundaries corresponding to the opening signal of the combustion gas valves (2).

6.4.3 RTO problem formulation

The optimization objective of the RTO layer is to minimize the fuel gas consumed by the compressor stations. A small portion of natural gas from the pipeline is burned to power the turbine, so compressor stations consume about 2 % of the natural gas running through them.

The optimization problem solved in the RTO layer is defined as follows:

$$\min_{u_{valv_HP1}, u_{valv_HP2}} \phi_p = Q_{fuel_EC1} + Q_{fuel_EC2} \quad (6.39)$$

s.t.

$$\text{Simplified RTO model} \quad (\text{see section 6.1.1.4}) \quad (6.40)$$

$$g_i = Demand_j - Q_j \leq 0 \quad j \in \{P_6, P_11, P_18, P_21, P_22, P_23\}$$

$$68 \leq P_{c_EC1} \leq 80$$

$$68 \leq P_{c_EC2} \leq 80$$

where ϕ_p (m^3/s) is the cost function to minimize, P_{c_EC1} (bar) is the discharge pressure of the compressor, u_{valv_HP} (%) is the signal to open the valve of combustion gases entering the turbine, Q_{fuel} (m^3/s) is the natural gas flow consumed by compressor stations, Q_j (Nm^3/s) is the flow rate at each delivery point and $Demand_j$ (Nm^3/s) is the minimum flow demanded by every consumer.

The values of the boundary conditions for the Reganosa network are defined in Table 6.3. The differences between the real optimum, obtained by solving the optimization problem based on the rigorous dynamic model and the model optimum, i.e., the solution of the RTO problem based on the simplified model whose solution is applied to the process, are shown in Table 6.4.

The NLP optimization problems have been solved using a sequential approach with a sequential quadratic programming (SQP) algorithm implemented in the SNOPT library (Gill, et al., 2008) and executed in EcosimPro software (EAInt, 2013).

Table 6.3. Boundary conditions.

	Value
P_{in, P_0} (bar)	68
T_{in, P_0} ($^{\circ}$ C)	9
P_{out, P_6} (bar)	68
$P_{out, P_{11}}$ (bar)	68
$P_{out, P_{18}}$ (bar)	68
$P_{out, P_{21}}$ (bar)	68
$P_{out, P_{22}}$ (bar)	68
$P_{out, P_{23}}$ (bar)	68
y_{CH_4} ($^{\circ}$ 1)	0.82
$y_{C_2H_6}$ ($^{\circ}$ 1)	0.05
$y_{C_3H_8}$ ($^{\circ}$ 1)	0.005
$y_{i-C_4H_{10}}$ ($^{\circ}$ 1)	0.0001
$y_{n-C_4H_{10}}$ ($^{\circ}$ 1)	0.0001
$y_{i-C_5H_{12}}$ ($^{\circ}$ 1)	0.001
$y_{n-C_5H_{12}}$ ($^{\circ}$ 1)	0.001
$y_{C_6H_{14}}$ ($^{\circ}$ 1)	0.001
y_{N_2} ($^{\circ}$ 1)	0.025
$y_{C_7H_{16}}$ ($^{\circ}$ 1)	0.008
y_{CO_2} ($^{\circ}$ 1)	0.033
y_{H_2S} ($^{\circ}$ 1)	0.0558

Table 6.4. Comparison between process and model optimum.

	Real Optimum	Model Optimum	Lower-Upper Limits
$P_{c_EC_1}$ (bar)	73.43	71.67	68-85
$P_{c_EC_2}$ (bar)	79.85	76.32	68-85
$u_{valv_HP\ EC_1}$ (%)	11.02	7.93	0-100
$u_{valv_HP\ EC_2}$ (%)	27.35	10.54	0-100
Q_{P_6} (Nm ³ /s)	5.90	5.00	5
Q_{P_11} (Nm ³ /s)	43.58	40.90	10
Q_{P_18} (Nm ³ /s)	101.45	88.08	50
Q_{P_21} (Nm ³ /s)	0.21	0.18	0.20
Q_{P_22} (Nm ³ /s)	10.00	8.64	10
Q_{P_23} (Nm ³ /s)	0.25	0.22	0.20
ϕ_P (m ³ /s)	0.0471	0.0725	-

The set of active constraints is not the same for both solutions, whereas at the nominal solution, that is the solution obtained by solving RTO without modifiers, the Q_{P_21} and Q_{P_22} are actives at the real optimum only Q_{P_22} is an active constraint.

6.4.4 Alternative NMA based on the Lagrangian function

The new approach described in section 3.3, the alternative NMA method that considers the minimization of the Lagrangian function measured from the process, is applied to this process.

This application is a good example to show the performance of this methodology, since there are many constraints, one for each output of the gas network, that is, one for each consumer.

The implementation of NMA in this case study would be problematic since the high number of constraints ($n_g = 6$) and the number of decision variables ($n_u = 2$) imply the requirement of 20 modifiers, according to (6.41), from which 14 are first-order modifiers whose value is given by the unconstrained optimization layer (section 2.1.5). Therefore, the initial simplex would require 14 + 1 initial operating points to be constructed.

$$n_{mod} = n_g + n_u(n_g + 1) \quad (6.41)$$

So, it would take 14 steady-states of the process just to initialize the algorithm. Taking into account the fact that the settling time of the process is about 1 hour, it would therefore take 14 hours to start to work.

By using the new approach, the number of modifiers is reduced from 20 to 8 (6.42), of which only two are first-order modifiers, one for each RTO decision variable. Thus, the initial simplex is constructed after only 2 + 1 steady-states.

$$n_{mod} = n_g + n_u \quad (6.42)$$

So, the resulting modified optimization problem is described as follows where the subscript “k-1” indicates the reference applied one sample time before.

$$\begin{aligned} \min_{u_{valv_HP1}, u_{valv_HP2}} \phi_M &= Q_{fuel_EC1} + Q_{fuel_EC2} \\ &+ \lambda_1^L (P_{c_EC1} - P_{c_EC1,k-1}) + \lambda_2^L (P_{c_EC2} - P_{c_EC2,k-1}) \end{aligned} \quad (6.43)$$

s.t.

$$\text{Simplified RTO model} \quad (\text{see section 6.1.1.4}) \quad (6.44)$$

$$g_{M,i} = Demand_j - Q_j + \varepsilon_j \leq 0 \quad j = 1, \dots, 6$$

$$68 \leq P_{c_EC1} \leq 80$$

$$68 \leq P_{c_EC2} \leq 80$$

The modifiers are given by the upper unconstrained optimization layer which solves the following problem:

$$\min_{\lambda_1^L, \lambda_2^L} L_p = \phi_p + \sum_1^6 \mu_{k,i}^T g_{P,i} \quad (6.45)$$

The obtained results are shown in the following figures. Figure 6.13 shows the evolution of the process cost function; Figure 6.14 and Figure 6.15 show the evolution of the discharge pressure for each compressor station. Figure 6.16 and Figure 6.17 show the variation of the flow corresponding to the active constraint at the real optimum and the inlet gas flow to the gas network.

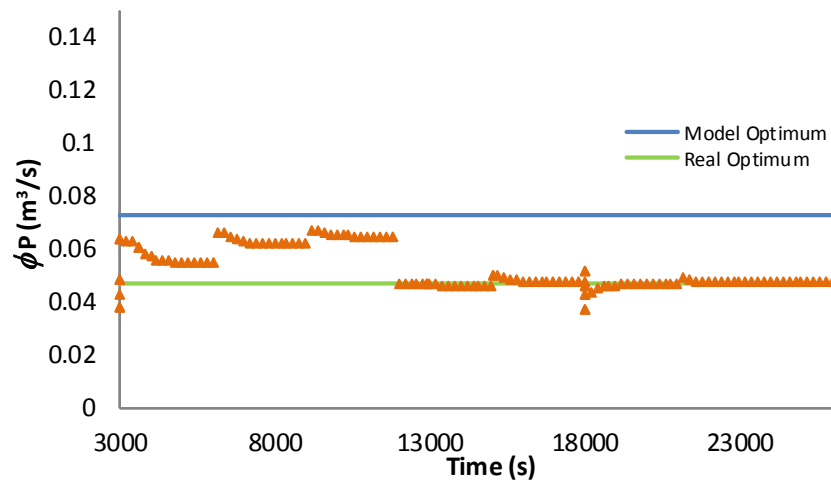


Figure 6.13. Evolution of the process cost function ϕ_P .

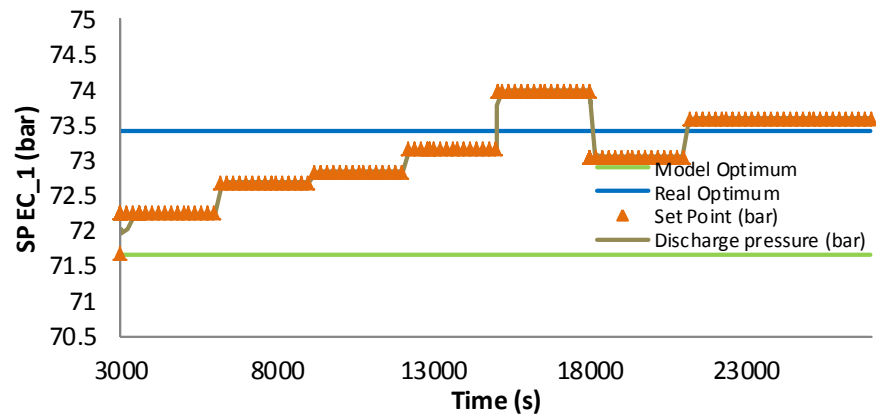


Figure 6.14. Evolution of the set point for the discharge pressure of EC_1 .

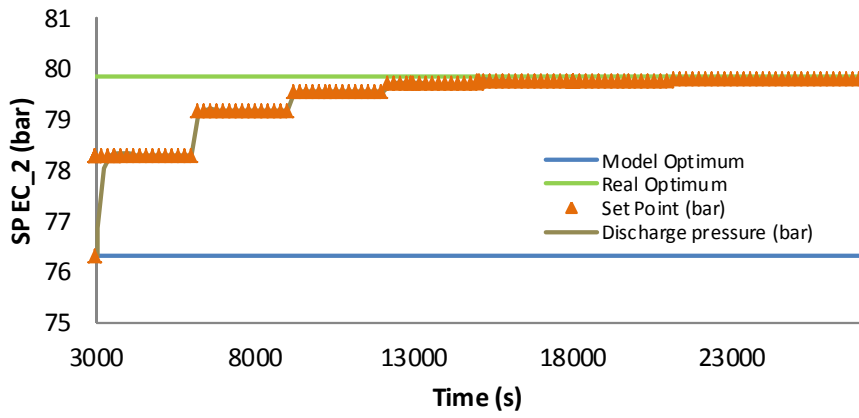


Figure 6.15. Evolution of the set point for the discharge pressure of EC_2 .

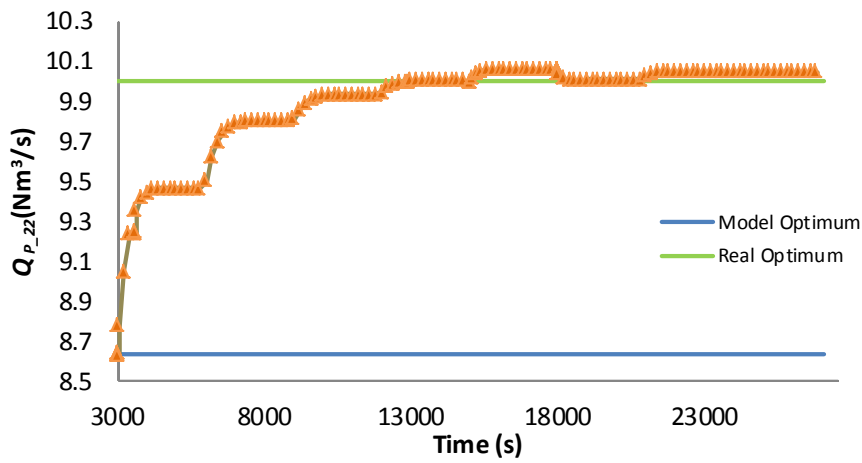


Figure 6.16. Evolution of active constraint $Q_{P_{22}}$.

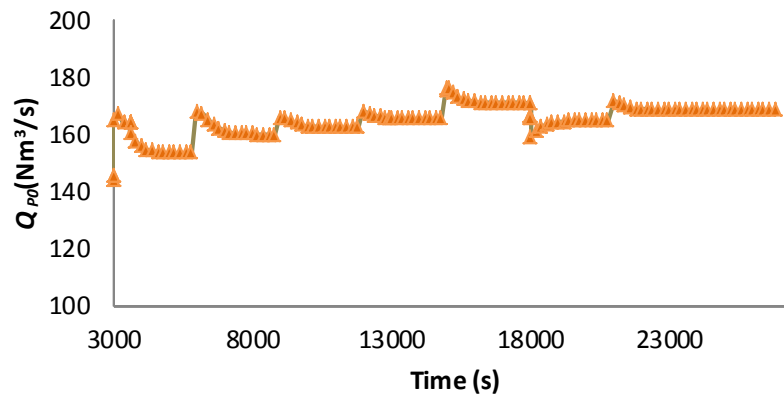


Figure 6.17. Evolution of the inlet volumetric flow Q_{p_0} .

6.5 Conclusions

In this chapter, another successful implementation of MA has been shown to deal with the optimization of gas transmission networks with the objective of minimizing fuel gas consumption of compressor stations while satisfying the demands of the consumers.

MA has been applied to a large-scale problem in the presence of numerous constraints showing that this methodology is a powerful tool that can be implemented in industrial cases.

In particular, an easily implementable method like NMA (section 2.1.5) has been presented. Since it does not require the computation of plant and model gradients, all it needs is the measurement of the process cost function and the constraints. To further simplify the implementation of this methodology to deal with problems with many operational constraints, such as the example described above, the alternative NMA method developed in this thesis (section 3.3) has been used to reduce the number of required modifiers, showing that it is able to achieve an operating point that satisfies the NCO of the plant faster than applying traditional NMA.

The results show that when applying the alternative NMA formulation only 7 steady-states are required, which implies a considerable time reduction compared to applying the standard NMA which would need, at least, 14 steady-states to start to work.

7 FINAL CONCLUSIONS AND FUTURE WORK

To sum up, some general conclusions are presented in this chapter followed by an analysis of the future research that could be carried out in the field of MA for RTO.

7.1 Final conclusions

This thesis has presented several contributions on the MA methodology oriented towards speeding up the convergence to an operating point that satisfies the NCO of the plant. Two different practical MA issues have been studied: the dimension problem with respect to the number of operational constraints and the slow convergence rate. A summary of the main contributions is given next:

- For dealing with processes with a large number of constraints without increasing the number of modifiers, an alternative NMA methodology has been developed in this thesis. This approach consists in estimating the modifiers from information of the Lagrangian function, adding only one gradient modifier for each process input to the cost function (see section 3.3). This reduction in the number of modifiers speeds up the convergence rate of the NMA methodology and makes the implementation of this technique easier in practice.

This contribution has been presented in: “*T. Rodríguez-Blanco, D. Sarabia, D. Navia, C. de Prada. Efficient Nested Modifier Adaptation for RTO using Lagrangian functions. Symposium on Computer Aided Process Engineering-ESCAPE 27, Barcelona, Spain, 2017*”.

DMA usually drives the process to the optimum in a faster and more direct way than NMA, since it deals directly with the information of the plant and model gradients. However, using the modifiers obtained by minimizing the Lagrangian function, the convergence rate of both methods, NMA and DMA, is very similar. NMA can even be faster if it is applied in a case study with a large number of operational constraints, since the benefits of the proposed approach will be larger, the greater the number of process-dependent constraints.

- The previous methodology is based on solving the RTO problem once a steady-state of the process is achieved which implies a slow convergence to the optimum of the process. To deal with this issue, two approaches have been developed with the aim of speeding up the convergence of the MA to the optimum by using transient information of the process. The first combines the NE-based gradient estimation during the transient with the estimation of gradients based on stationary data. Both parametric and structural plant-model mismatch (see section 4.3) could be dealt with in this way. The second aims to extend the idea of using transient measurements by

estimating the process cost and constraint gradients directly through a recursive identification method. In this way, waiting for the steady state at each RTO iteration is no longer necessary (see section 4.4). By implementing both approaches, the time to achieve the plant optimum is considerably reduced, as has been shown in the different contributions that have been made during this thesis.

The combination of NE-based gradient estimation with the traditional stationary MA methodology has been presented in: "*T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier-Adaptation approach to deal with structural and parametric uncertainty. DYCOPS-CAB 2016. June 4-9, 2016. Trondheim (Norway)*".

The method based on the direct estimation of process gradients over the transient has been presented in the paper: "*T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier Adaptation methodology based on transient and static measurements for RTO to cope with structural uncertainty. Computers & Chemical Engineering, 2017*" and also in the FOCAPO conference: "*T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier-Adaptation approach using RELS to compute process gradients. FOCAPO-CPC 2017. January 8-12, 2017. Tucson (Arizona), United States*". Its implementation in a laboratory-scale flotation column was presented in: "*D. Navia, A. Puen, L. Bergh, T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier-Adaptation based on transient measurements applied to a laboratory-scale flotation column. Symposium on Computer Aided Process Engineering-ESCAPE 27, Barcelona, Spain, 2017*".

- This thesis has also tested the performance of MA in realistic problems, such as the operation of a depropanizer distillation column and the transportation of natural gas through gas networks (see chapter six). The obtained results have shown that MA is a powerful tool that enables large-scale processes to be operated in an optimal way, maximizing the benefits obtained.

The implementation of several MA approaches on the depropanizer distillation column was presented in: "*T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier-Adaptation methodology for RTO applied to distillation columns. ADCHEM 2015. June 7-11, 2015. British Columbia (Canada)*" and "*T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Modifier Adaptation methodology based on transient and static measurements*".

for RTO to cope with structural uncertainty. Computers & Chemical Engineering, 2017”

- An exhaustive review of the fundamentals of MA and the different techniques developed until now, analyzing their features and the way to implement them, has been made during this thesis.

From this study a review paper was presented in: “*T. Rodríguez-Blanco, D. Sarabia, C. de Prada. Optimización en Tiempo Real utilizando la Metodología de Adaptación de Modificadores. RIAI: Revista Iberoamericana de Automática e Informática industrial, 2017”*.

7.2 Future work

In the near future, we plan to continue the work developed during this thesis on MA in different ways, as this methodology still presents some limitations that require further research. Among them, we aim to investigate the following:

- Ensure feasibility for all the intermediate RTO iterations. A possible solution could be to apply Dynamic MA. It is necessary to formulate the computation of modifiers for dynamic optimization problems that will be solved in the dynamic RTO.
- The implementation of MA in batch processes, whose optimization typically involves solving a dynamic optimization problem for which the solution consists of time-varying input profiles. Dynamic optimization problems have two types of constraints: the path constraints limit the inputs and states during the batch, while the terminal constraints limit the outcome of the batch at final time. Hence, the path constraints are modified using time-varying modifiers, while the terminal constraints are modified using terminal modifiers. The analysis of this approach could be the subject of future work.
- The MA family of methods is relatively well developed from a theoretical perspective. At this point, it is necessary to apply the methodology on an industrial scale. On the one hand, this will reveal where the real challenges lie and provide motivation for further improving the method. On the other hand, industrial practitioners will be more likely to adopt these methods if they have already been

shown to work on real systems. We plan to test our methodologies in an oil refinery.

REFERENCES

References

Acedo Sánchez, J., 2003. *Control Avanzado de procesos (Teoría y práctica)*. Díaz de Santos.

Bamberger, W. & Isermann, R., 1978. Adaptive on-line steady-state optimization of slow dynamic processes. *Automatica*, Volume 14, pp. 223-230.

Beale, E. M. L., 1955. On minimizing a convex function subject to linear inequalities. *Journal of the Royal Statistical Society*, 17(2), pp. 173-184.

Behrens, M., Khobkhun, P., Potschka, A. & Engell, S., 2014. *Optimizing set point control of the MCSGP process*. Strasbourg, France, pp. 1139-1144.

Ben-Tal, A. & Nemirovski, A., 1999. Robust solutions of uncertain linear programs. *Operations Research Letters*, 25(1), pp. 1-13.

Ben-Tal, A. & Nemirovski, A., 1998. Robust convex optimization. *Math. Oper. Res.*, Volume 23, pp. 769-805.

Bergh, L., 2007. *Combining on-line process measurements and models to empirically test strategies for process monitoring, diagnosis and control*. Quebec City.

Bergh, L. G., 2012. A hybrid approach to empirically test process monitoring, diagnosis and control strategies. *Advanced information technology in education*, Volume 126, pp. 215-222.

Bergh, L. G., Yianatos, J. B. & Leiva, C. A., 1998. Fuzzy supervisory control of flotation columns. *Miner Eng*, Volume 11, pp. 739-748.

Bertsimas, D., Brown, D. B. & Caramanis, C., 2011. Theory and applications of robust optimization. *SIAM review*, Volume 53(3), pp. 464-501.

Bertsimas, D., Pachamanova, D. & Sim, M., 2004. Robust linear optimization under general norms. *Operations Research Letters*, pp. 32:510-516.

Bertsimas, D. & Sim, M., 2004. The price of robustness. *Operations Research*, Volume 52(1), pp. 35-53.

Bertsimas, D. & Thiele, A., 2006. *Robust and data-driven optimization: modern decision making under uncertainty*, pp. 95-122.

Biegler, L. T., 2010. *Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.

Biegler, L. T., Grossmann, I. E. & Westerberg, A. W., 1985. A note on approximation techniques used for process optimization. *Computers & Chemical Engineering*, Volume 9(2), pp. 201-206.

Birge, J. R., 1997. Stochastic programming computation and applications. *INFORMS Journal on Computing*, 9(2), pp. 111-133.

Bock, H. G. & Plitt, K. J., 1984. *A Multiple Shooting Algorithm for Direct Solution of Optimal Control Problems*. 9th IFAC World Congress.

Borraz-Sánchez, R. Z. & Mercado, C. R., 2015. Optimization problems in natural gas transportation systems: A state-of-the-art review. *Applied Energy*, Volume 147, pp. 536-555.

Brdys, M., Chen, S. & Roberts, P. D., 1986. An extension to the modified two-step algorithm for steady-state system optimization and parameter estimation. *International Journal of Systems Science*, 17(8), pp. 1229-1243.

Brdys, M. & Roberts, P. D., 1987. Convergence and optimality of modified two-step algorithm for integrated system optimisation and parameter estimation. *Int. Journal of Systems Science*, 18(7), pp. 1305-1322.

Brdyś, M. & Tatjewski, P., 1994. *An algorithm for steady-state optimizing dual control of uncertain plants*. Smolenice, Slovakia, pp. 249-254.

Brdyś, M. & Tatjewski, P., 2005. *Iterative algorithms for multilayer optimizing control*. London, UK: Imperial College Press.

Bryson, A. E. & Ho, Y. C., 1975. *Applied Optimal Control: Optimization, Estimation and Control*. Washington, D.C.: Hemisphere Publishing Corporation.

Bunin, G. A., Francois, G. & Bonvin., D., 2013. Sufficient conditions for feasibility and optimality of real-time optimization schemes - I. Theoretical foundations. *ArXiv: 1308.2620*.

Bunin, G. A. et al., 2012. Experimental real-time optimization of a solid fuel cell stack via constraint adaptation. *Energy*, Volume 39, pp. 54-62.

Camacho, E. F. & Bordons, C., 2000. *Model predictive control*. Springer.

Chachuat, B., Srinivasan, B. & Bonvin, D., 2009. Adaptation strategies for real-time optimization. *Computers & Chemical Engineering*, Volume 33, pp. 1557-1567.

Chen, C. Y. & Joseph, B., 1987. On-line optimization using a two-phase approach: An application study. *Industrial & Engineering Chemistry Research*, Volume 26, pp. 1924-1930.

Costello, S., François, G. & Bonvin, D., 2013. *Real-time optimization when the plant and the model have different inputs*. Mumbai, pp. 39-44.

Costello, S., François, G. & Bonvin, D., 2016. A directional modifier adaptation algorithm for real-time optimization. *J. Process Control*, Volume 39, p. 64–76.

Coulson, J. M. & Richardson, J. F., 2002. *Coulson and Richardson's Chemical Engineering, Volume 2, 5th edition*. Butterworth-Heinemann.

Cozad, A., Sahinidis, N. & Miller, D., 2014. Learning surrogate models for simulation- based optimization. *AIChE Journal*, Volume 60, pp. 2211-2227.

Cozad, A., Sahinidis, N. & Miller, D., 2015. A combined first- principles and data-driven approach to model building. *Computers and Chemical Engineering*, Volume 73, pp. 116-127.

Dantzig, G., 1955. Linear programming under uncertainty. *Management Science*, 1(3-4), pp. 197-206.

Darby, M. L., Nikolaou, M., Jones, J. & Nicholson, D., 2011. RTO: An overview and assessment of current practice. *Journal of Process Control*, Volume 21, pp. 874-884.

de Prada, C., Serrano, J., Vega, P. & Piera, M. A., 1993. A comparative study of GPC and DMC controllers. *Advances in Model Based Predictive Control, CIM-Europe*, pp. 220-232.

Dupacova, J., 1995. Multistage stochastic programs. The state of the art and selected bibliography. *Kybernetika*, Volume 31, pp. 151-174.

EAInt, 2013. *User Manual* (<http://www.ecosimpro.com>). [Online].

El Ghaoui, L. & Lebret, H., 1997. Robust solutions to least-squares problems with uncertain data. *SIAM J. Matrix Analysis and Applications*, 18(4), pp. 1035-1064.

El Ghaoui, L., Oustry, F. & Lebret., H., 1998. Robust solutions to uncertain semidefinite programs. *Siam J. Optimization*, 9(1), pp. 33-52.

Engell, S., 2007. Feedback control for optimal process operation. *Journal of Process Control*, Issue 17, pp. 203-219.

Finch, J. A. & Dobby, G. S., 1990. *Column flotation*. Oxford: Pergamon Press.

Forbes, J. F. & Marlin, T. E., 1994. Model accuracy for economic optimizing controllers: the bias update case. *Industrial & Engineering Chemistry Research*, Volume 33, pp. 1919-1929.

François, G. & Bonvin, D., 2014. Use of Transient Measurements for the Optimization of Steady-State Performance via Modifier Adaptation. *Industrial & Engineering Chemistry Research*, Volume 53(13), p. 5148–5159.

François, G., Costello, S., Marchetti, A. & Bonvin, D., 2016. Extension of modifier adaptation for controlled plants using static open-loop models. *Computers and Chemical Engineering*, Volume 93, pp. 361-371.

François, G., Srinivasan, B. & Bonvin, D., 2005. Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty. *Journal of Process Control*, Volume 15(6), pp. 701-712.

François, G., Srinivasan, B. & Bonvin, D., 2014. Equivalence between Neighboring-Extremal Control and Self-Optimizing Control of dynamical systems. *Industrial & Engineering Chemistry Research*, Volume 53, pp. 7470-7478.

Gao, W. & Engell., S., 2005. Iterative set-point optimization of batch chromatography. *Computers & Chemical Engineering*, Volume 29, pp. 1401-1409.

Gao, W., Wenzel, S. & Engell., S., 2016. A reliable modifier-adaptation strategy for real-time optimization. *Computers & Chemical Engineering*, Volume 91, pp. 318-328.

Ghazanfarihashemi, A. & Ghanbariannaeni, A., 2012. Protecting a centrifugal compressor from surge. *Pipeline & gas journal*, 239(3).

Gill, P. E., Murray, W. & Saunders, M. A., 2008. *User's Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming*.

Goodwin, G. C. & Sin, K. S., 1984. *Adaptive filtering prediction and control*. New Jersey: Prentice-Hall, Information and system sciences series.

Griffiths, D. F. & Watson, G. A., 1995. *Numerical Analysis, Volume 344 of Chapman & Hall/CRC Research Notes in Mathematics Serie*. CRC Press.

Guay, M., 2014. A time-varying extremum-seeking control approach for discrete-time systems. *Journal of Process Control*, Volume 24, pp. 98-112.

Guay, M. & Zhang, T., 2003. Adaptive extremum seeking control of nonlinear dynamic systems with parametric uncertainty. *Automatica*, Volume 39, p. 1283–1294.

Haddad, W. M. & Chellaboina, V., 2008. *Nonlinear Dynamical Systems and Control: A Lyapunov-based approach*. Princeton University Press.

Jia, R., Mao, Z. & Wang, F., 2016. Self-correcting modifier-adaptation strategy for batch-to-batch optimization based on batch-wise unfolded PLS model. *The Canadian J. Chem. Eng*, Volume 94, pp. 1770-1782.

Khalil, H. K., 1996. *Nonlinear systems*. Prentice-Hall.

Krstic, M. & Wang, H., 2000. Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica*, Volume 36, pp. 595-601.

Leineweber, D. B., 1999. *Efficient Reduced SQP Methods for the Optimization of Chemical Processes Described by Large Sparse DAE Models*. Heidelberg: University of Heidelberg.

Ljung, L., 1987. *System Identification. Theory for the user*. New Jersey: Prentice-Hall, Information and system sciences series.

Luongo, C. A., Gilmour, B. J. & Schroeder, D. W., 1989. Optimization in natural gas transmission networks: A tool to improve operational efficiency. *Technical report, Stoner Associates, Inc., Houston*.

Marchetti, A., Chachuat, B. & Bonvin, D., 2009. Modifier-Adaptation Methodology for Real-Time Optimization. *Industrial & Engineering Chemistry Research*, Volume 48, pp. 6022-6033.

Marchetti, A., Chachuat, B. & Bonvin, D., 2010. A dual modifier-adaptation approach for real-time optimization. *Journal of Process Control*, Volume 20, pp. 1027-1037.

Marchetti, A., François, G., Faulwasser, T. & Bonvin, D., 2016. Modifier Adaptation for Real-Time Optimization—Methods and Application. *Processes*, 4(55).

Marchetti, A. G., Faulwasser, T. & Bonvin, D., 2017. A feasible-side globally convergent modifier-adaptation scheme. *J. Process Contr.*

Marlin, T. E. & Hrymak, A. N., 1997. Real-time operations optimization of continuous processes. *AIChE Symposium Series- CPC- V*, Volume 93, pp. 156-164.

Martí, R., 2015. *Ph.D. thesis: Price coordination strategies in large-scale process control.*

Milosavljevic, P. et al., 2016. *Optimal load sharing of parallel compressors via modifier adaptation.* Buenos Aires, Argentina.

Mirsky, S. et al., 2012. *Development and design of antisurge and performance control systems for centrifugal compressors.* Houston, Texas.

Navia, D., Briceño, L., Gutiérrez, G. & de Prada, C., 2015. Modifier-adaptation methodology for real-time optimization reformulated as a nested optimization problem.. *Industrial & Engineering Chemistry Research*, Volume 54, pp. 12054-12071.

Navia, D., Gutiérrez, G. & de Prada, C., 2013. *Nested Modifier- Adaptation Methodology for RTO in the Otto Williams Reactor.* Mumbai, India.

Navia, D. et al., 2012. *Handling infeasibilities in dual modifier-adaptation methodology for real-time optimization.* Singapore, pp. 537–542.

Navia, D. et al., 2017. *Modifier- Adaptation based on transient measurements applied to a laboratory-scale flotation column.* Barcelona, Spain.

Navia, D., Villegas, D., Cornejo, I. & de Prada, C., 2016. Real-time optimization for a laboratory-scale flotation column. *Computers & Chemical Engineering*, Volume 86, pp. 62-74.

Papasavvas, A., de Avila Ferreira, T., Marchetti, A. & Bonvin, D., 2017. *Real-time optimization via modifier adaptation using partial models*. Toulouse (France).

Pitzer, K. S., 1955. The Volumetric and Thermodynamic Properties of Fluids. *Journal of American Chemical Society*, 77(13), pp. 3427-3440.

Polat, M. & Chande, S., 2000. First-order flotation kinetics models and methods for estimation of the true distribution of flotation rate constants. *Int J Miner Process*, Volume 58, pp. 145-166.

Porru, M., Baratti, R. & Álvarez, J., 2015. *Energy saving through control in an industrial multicomponent distillation column*. Whistler, British Columbia, Canada.

Roberts, P. D., 1979. An algorithm for steady-state system optimization and parameter estimation. *International Journal of Systems Science*, Volume 10, pp. 719-734.

Roberts, P. D., 1995. A review of integrated system optimisation and parameter estimation (ISOPE). Coping with model-reality differences in industrial process optimisation. *Computers in Industry*, Volume 26, pp. 281-290.

Rodríguez-Blanco, T., Sarabia, D. & de Prada, C., 2016. *Modifier-adaptation approach to deal with structural and parametric uncertainty*. Trondheim, Norway, IFAC Symposium Dycops, pp: 851-856.

Rodríguez-Blanco, T., Sarabia, D. & de Prada, C., 2017. Modifier Adaptation methodology based on transient and static measurements for RTO to cope with structural uncertainty. *Computers and Chemical Engineering*, <http://dx.doi.org/10.1016/j.compchemeng.2017.07.001>.

Rodríguez-Blanco, T., Sarabia, D., Navia, D. & de Prada, C., 2015. *Modifier-adaptation methodology for RTO applied to distillation columns*. Whistler, British Columbia, Canada.

Rodríguez-Blanco, T., Sarabia, D., Navia, D. & de Prada, C., 2017. *Efficient Nested Modifier Adaptation for RTO using Lagrangian functions*. Barcelona, Spain.

Rodríguez, T. et al., 2013. *Modelling and monitoring of natural gas networks*. Paris (France).

Rueda, A., 2015. *Modelado, simulación, reducción de modelos y control predictivo en columnas de destilación*. Universidad de Valladolid.

Serralunga, F. J., Mussati, M. C. & Aguirre, P. A., 2013. Model adaptation for real-time optimization in energy systems. *Industrial & Engineering Chemistry Research*, Volume 52, pp. 16795-16810.

Shapiro, A. & Philpott, A., 2007. A tutorial on stochastic programming. Manuscript: Available at www2.isye.gatech.edu/ashapiro/publications.html, 17.

Skogestad, S., 2000. Self-optimizing control: The missing link between steady-state optimization and control. *Computers and Chemical Engineering*, Volume 24, pp. 569-575.

Soyster, A., 1973. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, Volume 21, p. 1154–1157.

Tatjewski, P., 2002. *Iterative Optimizing Set-Point Control. The basic principle redesigned*. Barcelona, Spain.

Vahidi, A., Stefanopoulou, A. & Peng, H., 2005. Recursive least squares with forgetting for online estimation of vehicle mass and road grade: Theory and experiments. *Vehicle System Dynamics*. Volume 43, pp. 31-55.

Wenzel, S., Gao, W. & Engell, S., 2015. *Handling disturbances in modifier adaptation with quadratic approximation*. Garmisch-Partenkirchen, Germany.

Williams, T. J. & Otto, R. E., 1960. A generalized chemical processing model for the investigation of computer control. *AIEE Trans*, Volume 79, pp. 458-473.

Woldeyohannes, A. D. & Majid, M. A. A., 2011. Simulation model for natural gas transmission pipeline network system. *Simulation Modelling Practice and Theory*, Volume 19(1), pp. 196-212.

Yip, W. S. & Marlin, T. E., 2002. Multiple data sets for model updating in real-time operations optimization. *Computers and Chemical Engineering*, Volume 26, pp. 1345-1362.

Zhang, J., Mao, Z., Jia, R. & He, D., 2015. Real-time optimization based on a serial hybrid model for gold cyanidation leaching process. *Minerals Engineering*, Volume 70, p. 250–263.

Zhang, Y. & Roberts., P., 1990. On-line steady-state optimisation of nonlinear constrained processes with slow dynamics. *Transactions of the Institute of Measurement and Control*, Volume 12(5), pp. 251-261.

