Simultaneous Design and Control of Chemical Plants:

A Robust Modelling Approach

by

Luis Alberto Ricardez Sandoval

A thesis

presented to the University of Waterloo

in fulfillment of the

thesis requirement for the degree of

Doctor of Philosophy

in

Chemical Engineering

Waterloo, Ontario, Canada, 2008

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

This research work presents a new methodology for the simultaneous design and control of chemical processes. One of the most computationally demanding tasks in the integration of process control and process design is the search for worst case scenarios that result in maximal output variability or in process variables being at their constraint limits. The key idea in the current work is to find these worst scenarios by using tools borrowed from robust control theory. To apply these tools, the closed-loop dynamic behaviour of the process to be designed is represented as a robust model. Accordingly, the process is mathematically described by a nominal linear model with uncertain model parameters that vary within identified ranges of values. These robust models, obtained from closed-loop identification, are used in the present method to test the robust stability of the process and to estimate bounds on the worst deviations in process variables in response to external disturbances.

The first approach proposed to integrate process design and process control made use of robust tools that are based on the Quadratic Lyapunov Function (QLF). These tests require the identification of an uncertain state space model that is used to evaluate the process asymptotic stability and to estimate a bound (γ) on the random-mean squares (RMS) gain of the model output variability. This last bound is used to assess the worst-case process variability and to evaluate bounds on the deviations in process variables that are to be kept within constraints. Then, these robustness tests are embedded within an optimization problem that seeks for the optimal design and controller tuning parameters that minimize a user-specified cost function. Since the value of γ is a bound on one standard deviation of the model output variability, larger multiples of this value, e.g. 2γ, 3γ, were used to provide more realistic bounds on the worst deviations in process variables. This methodology (γ-based) was applied to the simultaneous design and control of a mixing tank process. Although this approach resulted in conservative designs, it posed a nonlinear constrained optimization problem that required less computational effort than that required by a Dynamic Programming approach which had been the main method previously reported in the literature.

While the γ-based robust performance criterion provides a random-mean squares measure of the variability, it does not provide information on the worst possible deviation. In order to search for the worst deviation, the present work proposed a new robust variability measure based on the Structured Singular Value (SSV) analysis, also known as the μ-analysis. The calculation of this measure also returns the critical time-dependent profile in the disturbance that generates the maximum model output error. This robust measure is based on robust finite impulse response (FIR) closed-loop models that are directly identified from simulations of the full nonlinear dynamic model of the process. As in the γ-based approach, the simultaneous design and control of the mixing tank problem was considered using this new μ-based methodology. Comparisons between the γ-based and the μ-based strategies were discussed. Also, the computational time required to assess the worst-case process variability by the proposed μ-based method was compared to that required by a Dynamic Programming approach. Similarly, the expected computational burden required by this new μ-based robust variability measure to estimate the worst-case variability for large-scale processes was assessed. The results show that this new robust variability tool is computationally efficient and it can be potentially implemented to achieve the simultaneous design and control of chemical plants.

Finally, the Structured Singular Value-based (μ-based) methodology was used to perform the simultaneous design and control of the Tennessee Eastman (TE) process. Although this chemical process has been widely studied in the Process Systems Engineering (PSE) area, the integration of design and control of this process has not been previously studied. The problem is challenging since it is open-loop unstable and exhibits a highly nonlinear dynamic behaviour. To assess the contributions of different sections of the TE plant to the overall costs, two optimization scenarios were considered. The first scenario considered only the reactor's section of the TE process whereas the second scenario analyzed the complete TE plant.

To study the interactions between design and control in the reactor's section of the plant, the effect of different parameters on the resulting design and control schemes were analyzed. For this scenario, an alternative calculation of the variability was considered whereby this variability was obtained from numerical simulations of the worst disturbance instead

of using the analytical μ-based bound. Comparisons between the analytical bound based strategy and the simulation based strategy were discussed. Additionally, a comparison of the computational effort required by the present solution strategy and that required by a Dynamic Programming based approach was conducted.

Subsequently, the topic of parameter uncertainty was investigated. Specifically, uncertainty in the reaction rate coefficient was considered in the analysis of the TE problem. Accordingly, the optimization problem was expanded to account for a set of different values of the reaction rate constant. Due to the complexity associated with the second scenario, the effect of uncertainty in the reaction constant was only studied for the first scenario corresponding to the optimization of the reactor section.

The results obtained from this research project show that Dynamic Programming requires a CPU time that is almost two orders of magnitude larger than that required by the methodology proposed here. Likewise, the consideration of uncertainty in a physical parameter within the analysis, such as the reaction rate constant in the Tennessee Eastman problem, was shown to dramatically increase the computational load when compared to the case in which there is no process parametric uncertainty in the analysis.

In general, the integration of design and control within the analysis resulted in a plant that is more economically attractive than that specified by solely optimizing the controllers but leaving the design of the different units fixed. This result is particularly relevant for this research work since it justifies the need for conducting simultaneous process design and control of chemical processes. Although the application of the robust tools resulted in conservative designs, the method has been shown to be an efficient computational tool for simultaneous design and control of chemical plants.

Acknowledgements

I would like to thank the Mexican National Council for Science and Technology (CONACYT) for providing the financial support to this research work. Also, I would like to acknowledge NSERC from Canada for providing financial support to present this research work in international conferences.

My sincere gratitude to my supervisors Prof. Hector Budman and Peter Douglas. I could have not achieved this goal without your guidance and support. It has been a privilege to work with you. Muchas Gracias!

I would also like to acknowledge the members of the committee, Prof. Chris Swartz, David Fuller, Ali Elkamel, Bill Anderson and Eric Croiset. Many thanks for your valuable comments and suggestions and for improving the quality of this research work.

I would also like to express my gratitude to Chemical Engineering Department at the University of Waterloo for all the projects, sessionals and teaching assistants that were assigned to me. I really enjoyed my time at the University of Waterloo. Similarly, I would like to thank Pat Anderson, Liz Bevan, Siva Ganeshlingam, Dennis Herman, Ravindra Singh for their support during this journey. My regards to all of them.

I would also like to thank the Engineering Computing Department at the University of Waterloo for its financial support. In particular, my deepest thanks to Steve Carr for his patience, guidance and support, it was a pleasure to work on the consultant office.

I would like to express my sincere appreciation to all my friends, Fernando, Claudia, Jose Luis, Isaias, Malena, Joel, Cinthia, Fermin, Laura, Jose, Monica, Gerardo, Lorena, Norberto, Alicia, Agustin, Lila, Alejandro, Yamili, Jose Rafael, Jorge, Estanislao, Jazmin, Rosendo, Rafael, Maritza, Julio, Lulu, Fernando, Rosalva, Deborah, Will, Penny, Ian, Reza and Meg. I will always remember all the times we spent together and you certainly did this PhD a pleasant journey with your friendship. Thank you all!

I wish to acknowledge my parents for their love, care and support. The distance has not been an obstacle to love you more than I loved you when I left Mexico four years ago. Also, I would like to thank all the members of my extended family: Tere, Ruben, Angeles, Heberto, Maria de la Paz, Juan Manuel, Manuel, Manuelito and Nydia. Los queremos mucho!

Finally, my deepest and heartfelt gratitude goes to my wife Nelly, none of this work could have been done if you did not push me to pursue my dreams and goals. Thanks for letting this to happen and for being next to me always! I love you with all my heart. My love and gratitude to my little boy, Emilio Alberto, and my little girl, Christelle Andrea, thanks for you smile, love and patience. I love you with all my heart and I always will. I have really enjoyed this journey and I will always be if you are with me. Thanks for being part of me life!

Table of contents

List of Tables

List of Figures

Nomenclature

List of English symbols

A	A state space matrix
a	Model parameter
B	B state space matrix
b	Model parameter
C	C state space matrix
C_r	Crest factor
c	Constant parameter
D	D state space matrix
$\mathit{Đ}$	Scaling matrix in structured singular value analysis
d	Design variables
E	Expected value in least squares estimation
e	Independent random variables with mean zero
F	Linear matrix inequality representation
f	Nonlinear function
f	Number of full blocks in the perturbation matrix
F	Linear fractional transformation
f_0	Cut-off frequency
f_s	Sampling frequency
G	Physical process
$\check{G}$	Nominal process model
H	Input variables
h_{iq}	Nominal impulse response coefficients
I	Identity matrix
k	Bound on the μ-based worst-case output variability
L	Perturbation block
L_A	Additive uncertainty
L_E	Inverse multiplicative output uncertainty
L_I	Multiplicative input uncertainty
L_O	Multiplicative output uncertainty
M	Interconnection matrix in the SSV analysis
m	Full block dimensions in the perturbation matrix
N	Process settling time

o	Adjustable parameter
P	Independent Lyapunov matrix
p	Input vector in the linear fractional model structure
$\mathbf{P}_N$	Covariance matrix of the parameter estimates in least squares estimation
q	Output vector in the linear fractional model structure
Q	Scaling matrix in structured singular value analysis
r	Repeated block dimensions in the perturbation matrix
R	Covariance matrix of the unknown model parameters
s	Number of repeated blocks in the perturbation matrix
T	Sampling period
t	time
t_f	Time horizon
t_{set}	System's settling time
u	Input vector
V	Lyapunov function
w	Output variables
x	System's states
y	Output vector
Z	Data set

List of Greek symbols

α	Real scalar parameter
β	Real scalar parameter
γ	Bound on the random mean squared (RMS) gain
δ	Scalar weight
Δ	Normalized perturbation block
$\boldsymbol{\Delta}$	Perturbation matrix in the SSV analysis
δ_c	Complex scalar column vector
δh_{iq}	Uncertainty in the impulse response coefficients
$\delta\upsilon$	Maximum change in a disturbance variable
ε	Stopping criterion
ϕ	Worst-case process variability function
H	Bound on the γ-based model output variability
$\boldsymbol{\eta}$	Vector of decision variables
θ	Model parameter uncertainty
ϑ	Linear scalar function
$\boldsymbol{\iota}$	Unmeasured variables
$\boldsymbol{\kappa}$	Adjustable variables
$\boldsymbol{\lambda}$	Controller tuning parameters
μ	Structured Singular Value problem
$\boldsymbol{\nu}$	Covariance matrix of the added noise in least squares estimation
$\boldsymbol{\xi}$	Prediction error
ρ	Spectral radius
ς	Adjustable parameter
σ	Singular value
τ_{min}	System's smallest time constant
$\boldsymbol{\upsilon}$	Disturbance variables
$\boldsymbol{\varphi}$	Input-output data matrix in least squares estimation
χ	Eigenvalue
$\boldsymbol{\psi}$	Unknown parameter vector in prediction error estimation
Ω	Subset of all the possible trajectories in an uncertain model
$\boldsymbol{\omega}$	Process parametric uncertainty
ϖ	Adjustable parameter

Nomenclature for the mixing tank process, case study in chapters 3 and 4.

e_T	Feedback error for the temperature measurement
e_V	Feedback error for the volume measurement
F_c	Cold feed flow rate
F_h	Hot feed flow rate
Kc_T	Proportional tuning parameter, temperature controller
Kc_V	Proportional tuning parameter, volume controller
T	mixing tank's temperature
T_c	Temperature of the cold stream
T_h	Temperature of the hot stream
V	mixing tank's hold up or volume
V_d	mixing tank's design volume
z	Actuator's position of the outlet valve
ζ_T	State variable of the temperature controller
ζ_V	State variable of the volume controller
τ_T	Time integral tuning parameter, temperature controller
τ_V	Time integral tuning parameter, volume controller

Nomenclature for the Tennessee Eastman problem, case study in chapter 5.

A1	Pre-exponential term of the second reaction rate
A2	Activation energy coefficient of the second reaction rate
C_{BM}	Process unit's bare module cost
cp_G	Cost of producing product G
cp_H	Cost of producing product H
D	Diameter of the process units
F_p	Production index
G	Product G's mass flow rate
G*	Product G's specification in mass flow rate
$\breve{G}$	Product G's set point
H	Product H's mass flow rate
H*	Product H's specification in mass flow rate
$\breve{H}$	Product H's set point
L	Length of the process units
$\breve{L}_f$	Separator's liquid level set point
$\breve{L}_r$	Reactor's liquid level set point
$\breve{L}_s$	Stripper's liquid level set point
P	Pressure
P_A, P_B and P_E	Partial pressure of components A, B and E in the reactor.
$\breve{P}_r$	Reactor's pressure set point
r	Return on investment
R	Universal gas constant
r_i	Signal ratio for stream *i*
$\breve{T}_r$	Reactor's temperature set point
xm_G	Mass fraction of component G in the product's outlet
xm_H	Mass fraction of component H in the product's outlet
δA1	Uncertainty in the pre-exponential term of the second reaction rate
υ_A	Component A's mole fraction in stream 4
υ_B	Component B's mole fraction in stream 4
φ^G	Variability function for product G
φ^H	Variability function for product H

List of abbreviations and acronyms

CC	Capital Cost
CF	Cost Function
DP	Dynamic Programming
EVP	Eigen Value Problem in Linear Matrix Inequality
FIR	Finite Impulse Response
FOH	First Order Hold
FOPDT	First Order Plus Dead Time
FP	Feasibility problem in Linear Matrix Inequality
GBD	Generalized Benders Decomposition
GEVP	Generalized Eigen Value Problem in Linear Matrix Inequality
IMC	Internal Model Control
ISE	Integral Square Error
LMI	Linear Matrix Inequality
MIMO	Multiple-Input Multiple-Output
MINLP	Mixed Integer Nonlinear Programming
MPC	Model Predictive Control
OP	Operating Cost
PRBNS	Pseudo Random Binary Noise Signal
QLF	Quadratic Lyapunov Function
RMS	Random Mean Squared
RP	Robust Performance
RS	Robust Stability
SDV	Singular Value Decomposition
SISO	Single-Input Single-Output
SQP	Sequential Quadratic Programming
SS	State Space
SSV	Structured Singular Value
TE	Tennessee Eastman
VC	Variability Cost

1. Introduction

Chemical processes are nonlinear dynamic systems continuously affected by external disturbances and parametric uncertainties leading to process variability. Although control algorithms can be used to reduce variability, process design decisions may also have a large impact on the closed-loop dynamic behaviour, e.g. a larger valve would provide a wider range for the controller action such that it would help to reduce process variability and keep the process on specification. Chemical plants have been traditionally designed based on steady-state calculations and process synthesis heuristics that takes process controllability into account only after the process design variables have been determined. Thus, the process controllability and flexibility aspects have been generally treated independently. This sequential design approach is often inadequate since process controllability and flexibility may largely affect each other.

Based on the above, it has been proposed to simultaneously optimize the design and control strategy of chemical processes. Thus, the system's dynamic closed-loop performance is analyzed together with the system's design and control related degrees of freedom to determine the system's optimal operating state and the process units' capacity. This problem is non-trivial since it involves addressing trade-offs between conflicting design and control objectives. The problem has different aspects such as control structure selection, process synthesis analysis, process parametric uncertainty and the worst-case scenario assessment. At present there is no unified framework to address all these aspects at once. Instead, several methodologies that focus one part of the problem while ignoring the others have been reported in the literature. The latest reported methodologies have formulated this problem within a dynamic programming framework. Although this approach may cover many aspects of the problem, the computational effort required by this method is significant and it is the main barrier to its implementation for large problems.

This work presents a new approach to integrate design and control that alleviates some of the intensive computational burden required by dynamic programming based approaches. The core of the method is based on the application of robust control tools to estimate bounds on the stability, the dynamic operability, the controllability, the resiliency and the feasibility of

the process to de designed. The application of these tools results in a nonlinear optimization problem that is significantly easier to solve than the dynamic programming based formulations. A key idea in this work is to propose a formulation that provides a feasible asymptotically stable design, while meeting the proposed process's production and operational constraints in the face of *any* magnitude bounded external perturbations and process parametric uncertainty.

The methodology is based on a cost function that is defined in terms of process variables that can be physically measured. The capital and operating cost functions are used to determine the process steady-state economics. The capital cost is directly related to the process unit capacities whereas the operating cost is related to utility and process streams costs. In addition to these costs, the dynamic deviations of the process variables with respect to a nominal operating condition are also assigned a cost that is added as an additional term into the cost function. Different approaches have been used to measure the dynamic deviations in the process variables referred heretofore as the process variability. The controllability index-based methodologies have omitted the process variability related cost within the problem's cost function but have considered a controllability related constraint within the analysis. Similarly, the dynamic programming-based approaches have traditionally used an index performance weighted cost function that does not have a clear assignable economic cost. In contrast with previous work in this research, a worst-case process variability function is defined that assigns an explicit dollar value to the process variability. This variability function, specified in terms on the available process measurements, is estimated based on robust control performance tools that have been previously proposed in the literature for the design of robust controllers.

The proposed methodology has been tested with different case studies. Initially, a simple case study that involves a mixing tank process was used to analyze the advantages and disadvantages of the proposed approach. Then, to demonstrate the applicability of the proposed strategy to large scale processes, the methodology was applied to the simultaneous design and control of the Tennessee Eastman process.

Finally, several studies were conducted to determine the computational burden required by the proposed methodology. These studies included a comparison between the CPU times required by the proposed methodology versus that required by a dynamic programming-based approach.

1.1 Research objectives and novelties

Based on the above discussion, the research objective proposed in this project is to develop a methodology that performs the simultaneous design and control of chemical processes under the effect of magnitude-bounded external perturbations and process parametric uncertainties. The novelty in this new methodology is the application of robust control tools to test the process asymptotic stability and to estimate bounds on the process dynamic performance. To apply the robustness tests, the closed-loop process dynamic behaviour must be represented as a robust (uncertain) model. Thus, a key preliminary step in this work is to develop a systematic procedure to obtain uncertain model descriptions that are accurate representations of the closed-loop transient behaviour.

The use of robust control tools permits the calculation of analytical bounds on the process variability thus bypassing the need for dynamic optimizations. Therefore, the resulting methodology formulation is expected to require less computational effort than those approaches that formulates the design and control problem as a dynamic optimization problem. The reduced computational load expected with the present methodology is especially critical for solving large-scale chemical processes.

On the other hand, the application of a robust modelling approach may potentially lead to conservative designs. Accordingly, studies were conducted in this research work to assess the conservatism of the results. These studies included comparisons between the resulting design and control schemes obtained by the proposed strategy and the solutions found by dynamic programming-based methodologies.

Finally, there are a number of controller algorithms that can be considered for the control of process variables such as PID, MPC (Model Predictive Control) and IMC (Internal Model Control). Accordingly, an additional important objective of the present work was to

formulate a methodology that will be able to accommodate any one of these controller algorithms into the resulting optimization problems.

1.2 Research approach

The proposed general methodology consists in applying a robust stability test along with the calculation of robust performance bounds to simultaneously optimize the design and control of dynamic systems facing external magnitude-bounded disturbances and process parametric uncertainty. The stability test is used to ensure that the process final design is asymptotically stable in the presence of any magnitude-bounded disturbance and process parametric uncertainty. Similarly, the robust performance tests were used to establish bounds on the process operational constraints, the production constraints and the worst-case process variability function, used to determine the process variability cost.

The present work used two different criteria to formulate the robust stability and performance tests. The first approach made use of the concept of a Quadratic Lyapunov Function (QLF) to test the process asymptotic stability and to estimate bounds on the process constraints and the worst-case process variability function. These robustness tests were formulated assuming that an uncertain state-space model that describes the closed-loop process dynamic behaviour can be identified from the full nonlinear dynamic model of the process. Thus, the full closed-loop nonlinear dynamic process model is represented here as a nominal linear state space model with uncertain model parameters that are known to be within identified ranges of values. The combination of the nominal linear model and the uncertainty description is referred to as a robust (uncertain) state space model. The procedure to obtain such uncertain model as well as the method to determine the bounds on the state-space model parameters are presented in this work. Then a QLF-based robust performance test is used to estimate a bound on the random-mean squares (RMS) gain of a robust model (γ-approach), i.e. a bound on the model output error's variance.

Since the worst case scenario is related to a maximal possible deviation rather than to a mean squares deviation as calculated by the γ-approach mentioned above, an alternative second robust performance metric was sought. Accordingly, a robust performance test based on the Structured Singular Value problem (μ-approach) was used to calculate a bound of the

maximal deviation to replace the RMS-based bounds discussed above. To apply this test, the closed-loop model of the process was represented by the interconnection of a matrix representing the identified nominal linear model and the magnitudes of the disturbances and model uncertainties and a perturbation matrix that takes into account all the sources of uncertainties in the system. The bounds obtained from the application of this robust performance test were also used to evaluate the problem's constraints. The proposed calculation of the μ-based robust performance test is based on a robust finite impulse response (FIR) model of the closed-loop process dynamic behaviour. Each of the impulse response coefficients of this model is defined by a nominal value and corresponding lower and upper bounds to represent the uncertainty in the model parameter. The optimization problem to be solved for the calculation of the bound on the largest input/output variability is convex, i.e. a global maximum is always obtained. With similar robust performance tests it is also possible to calculate the critical disturbance realizations that produce the largest deviation in the robust FIR model output errors. Then, the maximal deviations in the process variables corresponding to the calculated critical disturbance can be obtained either from the μ-based analytical bound calculation or from simulations of the critical disturbance with the full nonlinear dynamic model of the process. It is expected that the later calculation method will produce less conservative designs but at the expense of larger computational time.

To ensure closed-loop operability, a robust stability test based on the SSV problem was also developed and integrated into the optimization problem. Accordingly, when using the μ-based formulation both stability and performance are addressed with the same robust FIR model ensuring consistency in the calculations.

1.3 Research work contributions

The current PhD research thesis has resulted in the following contributions:

1. The development of a basic simultaneous process design and control methodology that applies robust control tools to test the process asymptotic stability and to estimate bounds on the worst-case process variability and the problems' feasibility constraints.

2. The development of a novel methodology to tackle the simultaneous design and control for large-scale processes. This methodology has been shown to significantly reduce the computational burden as compared to dynamic optimization based methodologies.

3. The development of a novel μ-based robust performance test that estimates a bound on the largest model output error's variability. A by-product of this test is the critical disturbance profile that produces the largest calculated output variability.

4. Insight regarding the advantages or disadvantages of using different robust control tools to attain the simultaneous design and control of chemical processes.

5. Insight regarding the design of the Tennessee Eastman process used as an example of an integrated chemical plant.

1.4 Outline of the work

The thesis is organized in 6 chapters as follows:

Chapter 1 presents an overview of the research work that was performed. The objectives, novelties and the contributions of this project are discussed in this chapter.

Chapter 2 provides a literature review of the subjects that are relevant for this project. First the current available simultaneous design and control methodologies that have been reported in the literature are summarized. Subsequently, a review is included on general robust control related concepts and calculations including uncertain model structures, Linear Matrix Inequalities (LMI's), Quadratic Lyapunov Functions (QLF) and on the Structured Singular Value problem (μ-analysis). An overview of the current techniques available for closed-loop identification is presented at the end of this chapter.

In chapter 3, a methodology based on a QLF (Quadratic Lyapunov Function) concept is introduced for the simultaneous design and control of chemical processes. The key idea in this strategy is to represent the process dynamic behaviour as a robust state-space models that can be used to test the process stability and to estimate bounds on the input/output RMS gain. The procedure to identify robust state-space models from simulations, the robust stability

criterion, and the use of the robust RMS test (γ-approach) to formulate the worst-case process variability function and to evaluate the process constraints are explicitly presented in this chapter. Then, the methodology for simultaneous optimization of design and control based on the robust bounds discussed above is presented. To test the proposed methodology, the simultaneous design and control of a mixing tank process is considered in this chapter. Comparisons between the results obtained from the proposed methodology and those obtained by a dynamic programming-based methodology are also shown followed by a study conducted to determine the computational burden of the two compared techniques.

Chapter 4 discusses a simultaneous design and control methodology that applies a SSV-based robust performance criterion. This novel robustness test estimates a bound on the largest model output error's variability. Thus, this μ-based performance criterion can be used instead of the γ-based performance criterion within the methodology's formulation presented in chapter 3. The formulation of this new robust performance test, its application to a simple robust state-space model and the reformulation of the methodology's mathematical formulation in terms of this new performance criterion are presented in this chapter. The reformulated methodology was tested on the simultaneous design and control of the mixing tank process, previously introduced in chapter 3. The results obtained by this new methodology's formulation and those obtained by the QLF-based methodology and by a dynamic programming-based methodology are discussed followed by a comparison of the computational requirements for the two techniques.

Chapter 5 extends the methodology presented in chapter 4 to the design and control of large-scale processes. The formulation of more parsimonious μ-based robust stability test and additional simplifications made to reduce the methodology's computational load are initially discussed. Then, the methodology was tested on the Tennessee Eastman process. The process flow-sheet philosophy and the control structure used to regulate this plant's dynamic behaviour are explained in this chapter. The methodology was first applied to the simultaneous design and control of the reactor section of the plant. The effect of adding disturbances, the use of different initial guesses on the methodology's optimization problem, the use of the critical disturbance vector to attain the design, comparisons with an optimal control problem and the effect of process parametric uncertainty were also analyzed and

discussed. Subsequently, a comparison of the CPU time required by the present method to that required by a dynamic programming-approach methodology is presented. The methodology was then applied to study the effects of multiple disturbances when the complete plant was considered within the analysis. The results obtained from this study as well as comparisons to those obtained when only the reactor section was considered in the analysis are presented at the end of this chapter.

Finally, chapter 6 presents the conclusions obtained from the present research work and outlines the areas for future research work.

2. Literature Review

The aim of this work is to develop a new methodology that performs the integration of design and control for dynamic systems under parametric uncertainty and external disturbances. The novelty in this approach consists in the application of robust stability and variability measures to find the process optimal design. To estimate the aforementioned measures, linear models with uncertain model parameters are needed. Accordingly, mathematical tools and methods traditionally used for robust control design and system identification are extensively used in this work and consequently they will be reviewed in this chapter.

This chapter initially presents a discussion on the previous approaches proposed to perform the simultaneous design and control of dynamical systems and then it revises the concepts and mathematical methods that are relevant for this work. The chapter is organized as follows: Section 2.1 presents a review on the current approaches that has been proposed to formulate the integration of design and control problem. In Section 2.2, the concepts and formulations from robust control theory that are relevant for this study are introduced; model uncertainty descriptions and the concepts of Linear Matrix Inequalities (LMI's), Lyapunov functions, and Singular Structured Value (SSV) are reviewed on this section. Section 2.3 presents a brief review on systems identification. The least-squares method, the design of suitable input signals for identification and other important guidelines for experimental design are also discussed in this section.

2.1 Integration of process design and control

Chemical processes are inherent dynamic systems that are continuously subjected to the effect of perturbations. These perturbations may be of two types: unmeasured disturbances entering the process and unmeasured changes in process parameters or operating conditions over time. The later type of perturbation will be referred to heretofore as *process parameter uncertainty*. Designers of chemical processes must take into account these process perturbations and uncertainties in order to provide realistic designs of the plant.

In the past, process design has been performed using an approach that involves a set of sequential steps. The earlier steps of the design are devoted to process synthesis analysis and to determine the process operating conditions that optimize the process steady-state economics. Both process synthesis and operating conditions calculation steps are performed at steady-state assuming that a controller can be designed and implemented into the process to keep it on target and within its feasible region. Following these design steps, process design parameters, i.e. process unit's capacities, valve sizes and process operating conditions are specified. The last stage of the design in this sequential approach typically consists of a controllability analysis and the design of a regulatory algorithm that would keep the process on specification.

The sequential design strategy outlined above have been often found to be unreliable since the assumptions made on the earlier stages may not be completely satisfied at the later stages of the dynamic control design. Often, the process closed-loop performance may not be satisfactory in the presence of external disturbances since the controller's range of operation is limited by design specifications such as control valve sizes that were fixed at the earlier stages of the design. Also, there is no guarantee that process operating conditions that optimize a steady-state based cost-function will still be optimal when there is process variability due to disturbances and parametric uncertainties affecting the system. In extreme situations, process design decisions made on the earlier design stages may result in a process that is very difficult or impossible to control. For instance, the control actions required to reject disturbances may drive the process to an undesirable operating condition that could put at risk the plant's safety, e.g. thermal runaway in a catalytic reactor. Shiskey[1] presents examples that describe uncontrollable systems that were designed with the traditional sequential approach.

Following the discussion above, it can be concluded that the interaction between design specifications and the process dynamic characteristics have a profound effect on the process' optimal dynamic performance. Thus, while the process steady-state economics is used to calculate design parameters for an idealized plant that is not subject to dynamic perturbations, these design parameters determine the process dynamic characteristics and performance. Dynamic performance of processes has been quantified in the past by different

concepts. For example, the process ability to move between different operating conditions has been referred to as switchability whereas the ability to reject disturbances has been referred to as process resiliency. On the other hand, the controllability concept has been used to describe and quantify the ability of a process to keep variables at target and to meet products specifications by reducing variability.

To remain competitive, low cost integrated plants with tighter process constraints and stringent environmental regulations need to be designed. To achieve this goal, the interactions between process dynamic effects and design specifications may be analyzed together in the earlier stages of the process design. The area of design that combines steady-state design and dynamic control considerations into one optimization step has been referred to in the literature as *integration of design and control*. Thus, when referring to *design* and *control,* the *design* word refers to decisions regarding flowsheet synthesis and nominal operating conditions based on steady-state models whereas the word *control* refers to the design of controllers that result in optimal closed loop dynamic performance.

The task of performing both design and control simultaneously as per the above definitions is not straightforward since optimizing both may require a trade-off between conflicting objectives. For example, while process units with small hold up may be calculated based on steady-state capital cost considerations, larger unit's capacities will be often required for proper disturbance rejection. Similarly, steady-state design could be used to specify the narrowest range of operability in the process operating conditions to minimize the system's operating cost whereas corrective actions performed by the control algorithm may require a wider range of operability in the process variables to achieve the desired process dynamic performance. Luyben[2] discusses typical chemical processes in which inherent conflicts between these two objectives occur. Moreover, process flexibility, i.e. the ability to remain feasible in the presence of external perturbations or process uncertainties, must be guaranteed for both the process steady-state and transient behaviour. Thus, a dynamic flexibility analysis should be conducted within the design to define the process feasible region that can accommodate the effects of process parameter uncertainties and any possible time-dependent unmeasured disturbance profile. To this regard, one of the most challenging tasks in the integration of both activities is the definition of the critical time-dependent profile in the

perturbations along with the process parameter uncertainty's steady-state value that generates the largest deviation in the system. This condition, to be referred heretofore as the *worst-case scenario*, is not easily identifiable and is computationally demanding. In addition, a stability analysis should also be taken into account within the design optimization problem to ensure stability both at steady state and during transients.

To guarantee that the process meets a suitable dynamic performance criterion, a controllability analysis is needed. Such controllability analysis may consider the selection of a suitable control structure for the process under study as well as the specification of the tuning parameters for the selected control algorithm. Although the dynamic behaviour may be used as an indicator of performance, it is clearly desirable to correlate the system closed-loop dynamics with an economic cost. However, the definition of a controllability cost function is not simple because often the relationship between controllability and process economics is not obvious and is problem dependent.

Due to the complexity associated with this problem, there are not general approaches in the literature that address the integration of design and control. Instead, several methodologies have been proposed to solve partial aspects of the integration of design and control problem. The methods reported in the literature vary in their level of complexity and in the assumptions and simplification made to solve the resulting optimization problem. In general, most of the methodologies focus on some aspects of the problem, e.g. process flexibility and controllability, while ignoring others, e.g. process stability. Perkins[3] provides a comprehensive review of the first methods that were proposed to solve this problem whereas Seferlis and Georgiadis[4] reviews in detail the recent contributions and new techniques that have emerged in the integration of design and control field.

Several classifications can be made to sort the current methodologies that have been published in the literature. In the present work, the methods are classified based on the method used to assess the process dynamic performance. Accordingly, the approaches taken to address the simultaneous design and control of chemical processes can be classified as follows: 1) Controllability index approach, 2) Dynamic programming approach and 3)

Robust approach. The following subsections present each of these approaches and outline the contributions that have been done in that area.

2.1.1 Controllability index-based approach

The basic idea of this approach is to made use of a controllability index to quantify the process closed-loop dynamic performance. Studies based on this approach have usually used an economic cost function that is defined as the combination of the process steady-state economics and an economic cost correlated with a controllability index. To measure the process economics at steady-state, the traditional capital and operating cost functions have been used. Controllability indicators such as the relative gain array[5], the condition number[6], the disturbance condition number[7] or the integral error criterion [8] have been used to measure the process dynamic performance. The studies that use this approach can be considered as the first attempts to solve the integration of design and control problem. The majority of these studies have used steady-state models or linear dynamic models to represent the process dynamic behaviour.

Lenhoff and Morari[9] proposed a Lagrangian function-based bounding technique to evaluate process synthesis, process design and control in a heat-integrated distillation columns. In this method, the process dynamic performance was measured using the weighted mean squared error criterion, obtained from simulations of the system under different scenarios. Palazoglu and Arkun[10] proposed a multi-objective function-based methodology that applies a Singular Value Decomposition (SDV) technique. The method measures the process closed-loop dynamic performance to improve the dynamic operability of the system by considering the process design aspects. A key feature in this work is the inclusion of process uncertainty within the problem formulation; however, the SDV-based robust controllability indices are only considered as inequality constraints within the formulation and do not appear in the economic cost function to be optimized. The same authors extended their previous work to incorporate process operability uncertainty within the design and control problem[11]. The revised method formulates the problem as a two-stage optimization: the lower level optimization solves the feasibility and operability problem for a given robust index while the upper level optimization seeks for the design parameters that optimize the cost function.

Since the robust dynamic performance appears as inequality constraint in the inner optimization problem, the design and control problem is solved for a user-defined robust performance index value. Thus, this method generates alternative designs for each robust index performance value tested. Luyben and Floudas[12] proposed a formulation based on a mixed-integer nonlinear optimization programming (MINLP) to integrate process synthesis, process design and control within one single optimization problem. The method defines a cost function based on the process steady-state economics and considers different open-loop controllability measures, such as relative gain array and the condition number, as constraints within the optimization problem formulation. A pareto-optimal analysis is used to show the advantages/disadvantages of different design alternatives. In this study, only steady-state models are used to describe the process behaviour. Alhammadi and Romagnoli[13] proposed an step-by-step integrated framework that incorporates economical, environmental, heat integration and controllability aspects of the process to be designed. Although the method uses the full process dynamic model to analyze heat integration, only steady-state controllability indicators are used to measure the process dynamic performance. Also, process resiliency is explicitly not considered in the problem formulation. The methodology was applied to a heat exchanger network (HEN) in which steady-state controllability measures such as relative gain array and condition number were employed to measure the process dynamic performance. Brengel and Seider[14] developed a coordinated design optimization strategy to address the process design and control problem by simultaneously solving a process design optimization problem and a Model Predictive Control (MPC) nonlinear optimization problem. In this work, the integral of the squared error (ISE) was used as controllability measure. This ISE is estimated from the simulation of the complete process dynamic model under different scenarios. Also, the mathematical framework of this method was developed by considering an MPC controller as the regulatory algorithm. Accordingly, this strategy offers two key advantages: 1) it uses the complete process dynamic model, and 2) model-based controllers such as MPC can be defined within the design strategy to control the system.

Although the methodologies described above provided an approximation of the simultaneous design and control problem, these strategies are limited by the fact that the economic cost resulting from product variability under closed-loop dynamic conditions is not strictly

considered in the process economic function. In most of the cases, the controllability indicators were set as constraints within the problem formulation or even considered in the cost function by making use of weighted functions that were selected ad hoc and are not based on their relative effect on the overall costs. Apart from Brengel and Seider[14], these methodologies represent the process dynamic behaviour as steady-state models or linear dynamic models which limits its applicability to those systems that exhibit approximate linear behaviour. This assumption of linearity implies that the range of process operability for which the analysis is valid is limited to a small neighbourhood of a nominal steady state. Thus, due to the problem formulation, i.e. the use of multi-objective cost function, and to the linearity assumptions, the application of the above strategies to simultaneously design and control chemical processes generally results in a suboptimal design.

2.1.2 Dynamic programming approach

Due to the limitations of the controllability index-based approach, a set of methodologies based on dynamic programming have been proposed. Instead of using steady-state or matrix measures as process variability indicators, this approach is based on the evaluation of a single economic index objective function that measures the process overall closed loop performance. To evaluate this objective function, the process dynamic behaviour is usually analyzed for a pre-specified time horizon. In contrast with the controllability index based method, this approach uses a comprehensive process nonlinear dynamic model within a dynamic optimization problem. One of the key advantages of this approach is that process uncertainty and external time-dependent perturbations can be rigorously accounted for within the problem formulation. Thus, it is possible to identify the scenario that, for a given set of nominal operating conditions, produces the largest process output error. Then, the process design and control problem is solved based on the predicted worst-case scenario.

The popularity of this approach has increased due to advances in dynamic programming algorithms and the increasing computing power available to researchers and practitioners in this area. Perkins and Walsh[15] proposed a dynamic worst-case based design method that considers process design and control structure selection. The method takes into account process uncertainty and a pre-specified time-dependent profile in the disturbances. The

process dynamic performance is quantified based on the response in process variables due to changes in the disturbances and to given process parameter uncertainty values. The control structure selection problem is posed as a mixed-integer optimal control problem. Due to a-priori specification of the disturbance profile, alternative designs are often obtained from this method. Mohideen et al.[16] introduced the basis of a conceptual design framework to simultaneously design and control dynamic systems in the presence of process parameter uncertainty and disturbances. The problem is mathematically formulated as a mixed-integer stochastic optimization problem. This optimization is solved by an iterative decomposition procedure that is composed of the solution of a multi-period design and control problem that gives a set of design and control tuning parameters, followed by the solution of a dynamic feasibility problem[17] which specifies critical realizations of the uncertain parameters and the disturbance. These critical profiles are then used in the multi-period problem to define a new set of design and control parameters. Integer decisions appear in the problem to account for the control structure selection problem. The concept of matrix measures were presented and incorporated within the proposed framework[18] to account for process stability in the presence of uncertainty realizations and time-dependent disturbances. The algorithm framework proposed by Mohideen et al. is complex and computationally demanding even when a small number of process units are considered in the design. To circumvent this shortcoming, Bansal et al.[19] proposed a different solution procedure based on a variant-2 of the generalized Benders Decomposition (v2-GBD) technique for MINLP. This method was applied to simultaneously design a distillation column. In this case study, process parameter uncertainty and external disturbances changes are assumed to be a mix of slow and fast sinusoidal variations with bounded frequency and amplitude. This method has also been applied to simultaneously design and control a double effect distillation column[20], a high purity industrial distillation system[21], and a multi-component mixed-integer distillation column model[22]. In addition, Sakizlis et al.[23] incorporated the concept of parametric programming within the conceptual design framework proposed by Bansal et al.[19]. The parametric programming algorithm proposed in this work can be used to incorporate advanced control techniques within the design and control problem. The method systematically maps the optimal control actions developed by the model-based controller in the space of the state measurements. Thus, a simple explicit state feedback controller is derived that moves off-line

the embedded on-line control optimization problem. Consequently, the simultaneous design and control methodology proposed in this work allows the use of advanced on-line optimization controllers such as MPC. Similarly, Schweiger and Floudas[24] proposed a systematic approach that considers a steady-state process economic function combined with a controllability related index, i.e. ISE, embedded within a multi-objective mixed-integer optimal control optimization problem. Due to the complexity associated with the formal solution of the problem, the ε-constrained method is used to avoid the formal solution of a multi-objective problem, i.e. a controllability criterion have to be defined a-priori and added as an inequality constraint within the optimization problem. In addition, the optimal control problem is parameterized to reduce the problem to a finite dimension. This method results in a suboptimal design since the combination of integer flow-sheet decisions with pareto-optimal solutions generates a large set of alternative designs that can differ drastically one from another. Bahri et al.[25] used the idea that a system should be moved away from the optimal steady-state operating condition in order to satisfy dynamic constraints. This concept is referred to in the literature as the *back-off approach*. The distance between the optimal steady state operating point and the operating point at which all the process constraints are satisfied is used in this work as a flexibility measure. The resulting design is obtained from the iterative solution of a two-stage optimization problem: in the high level optimization stage a dynamic MINLP algorithm is used to calculate design and control parameters whereas in the lower level optimization stage a dynamic NLP algorithm is used to find critical disturbance time profiles that may lead to constraints violations. The integer decisions that appear in the upper level optimization problem are used for flow-sheet synthesis and control structure selection. Kookos and Perkins[26] applied the back-off approach to perform design and control structure selection for dynamic systems under the effect of uncertainty and fast disturbances given as sinusoidal variations with bounded frequency and amplitude. The algorithm proposes the iterative solution of two nested optimization problems: the outer loop consists of a mixed-integer steady-state optimization problem that includes an inequality constraint that determines the steady-state operation to a dynamic performance bound. The solution of this outer loop optimization generates steady-state design parameters and an optimal control structure along with the corresponding controller's tuning parameters. Then, in the inner loop, the solution of an optimal control problem that takes into account the

process dynamics is performed and a corresponding new dynamic performance bound is obtained. This iterative procedure continues until no further improvements can be made in the outer optimization problem, i.e. the inequality constraint based on the steady-state and the dynamic performance index cannot be satisfied. Swartz[27] developed a systematic framework that considers the application of the control Q-parameterization within the design and control problem. The methodology is posed as a multi-period optimization problem because it considers the case of bounded parametric uncertainty and pre-specified time-dependent disturbances. Although the Q-parameterization is applicable to linear control problems only, the proposed mathematical description also imposes path constraints that account for the process nonlinear dynamic behaviour.

Although the dynamic programming-based methodologies outlined above solves the integration of design and control problem with realistic scenarios and on the basis of rigorous nonlinear dynamic models, the complexity associated with the optimization problem and their corresponding implementation within the proposed iterative solution procedures are key drawbacks of these methodologies. Furthermore, the computational burden associated with the solution of these problems is enormous, even when only a small number of process units are considered in the analysis. Thus, the curse of dimensionality associated with dynamic programming-based methods make these methods impractical for tackling real industrial problems. Although new approaches have emerged in the recent years to alleviate the computational burden involved in dynamic programming[28], no application of these approaches to the design and control problem have been reported in the literature. With regards to the cost functions used in the studies reported in this section, they have generally used a set of weighted terms that represent variables such as the system operability or the environmental impact of the plant, that do not have a clear assignable economic cost as in the case of capital and operating costs. In addition, apart from Mohideen et al.[18] and Sakizlis et al.[23] the studies reviewed in this section do not explicitly consider system stability, i.e. the controller tuning parameters, obtained from the solution of the proposed optimization problem, could potentially result in a closed loop unstable process.

2.1.3 Robust approach

Due to the difficulties associated to the dynamic programming approach, new approaches have been proposed in recent years to deal with the simultaneous design and control of dynamic systems. Instead of using the full nonlinear process model, these methods estimate bounds on the process flexibility, stability and controllability problem. Thus, the computationally demanding task of solving a dynamic optimization problem is reduced to a nonlinear programming optimization problem.

The bounds on these methods are obtained by using mathematical tools commonly used in the control design area. Chawankul et al.[29] proposed a single economic cost function that accounts for the process steady-state economics and a variability cost term, obtained from robust control measures. The key idea is to approximate the full nonlinear dynamic model by a nominal linear model with model uncertainty that accounts for the difference between the nominal linear model and the full nonlinear model behaviour. The linear nominal models are First Order plus Time Delay (FOPDT). The robust models complemented with model uncertainty are used to estimate bounds over the process variability and stability. This robust control theory-based method has also been applied to the simultaneous design and control of a binary distillation column using MPC as the control algorithm[30]. The main drawback of this methodology is that it is limited to sinusoidal disturbances at one given frequency. Gerhard et al.[31] have explored the space of the uncertain parameters to build critical boundaries over the region process stability and feasibility. The method, based on constructive nonlinear dynamics[32], establishes robust measures based on a minimal distance between the uncertain parameter space region and the critical boundaries. Since this method uses the complete nonlinear process model, it requires the iterative solution of a nonlinear constrained optimization problem combined with the numerical integration of the process model. This formulation has been applied to simultaneously design and control the reaction section of the hydrodealkylation of toluene[33]. Although the method requires a priori knowledge of the parametric uncertainty dynamics, exogenous disturbances have not been considered yet within the analysis. However, this technique may have potential to treat more complex and realistic scenarios.

From the computational point of view, the above formulations require less effort than dynamic programming-based descriptions. On the other hand the use of bounds as flexibility and controllability measures may result in suboptimal designs. Also, the fact that disturbances are not strictly treated as time-varying perturbations imposes restrictions on the resulting design.

2.2 Robust control theory

The approach to integration of control and design taken in the current thesis is based on mathematical tools that have been commonly used for the design of robust controllers. Therefore, the current section will briefly review the robust control techniques that will be used later in this work.

Control algorithms are designed to guarantee system's stability and performance. Often, the controller is tuned based on the process model and closed-loop simulations. Since the process model rarely captures *all* the variations that may possibly occur during the system's operation, the mathematical model of the process is only a simplified version of the true process behaviour. Consequently, there is no guarantee that the control law developed for this system would provide the desired stability and performance on the real system. If model inaccuracies, usually referred to as *model uncertainty* or *plant/model mismatch*, are considerable, then the controller actions can be such that it may drive the system to instability. Thus, it is desirable to design a control algorithm that is *insensitive* to model uncertainty effects, that is, the control algorithm must be *robust*. To guarantee process stability and a given level of process variability in the presence of model uncertainty, a robust stability (RS) and robust performance (RP) analysis are performed to test the controller's ability to remain stable and provide good performance in the presence of the different sources of uncertainties that arise on a physical system.

To perform such analyses, the process model must consider process uncertainties in their structure. Thus, the mathematical representation of the physical models varies according to the form in which the uncertainties appear in this model and to the way these uncertainties are quantified within the model. There are two main analytical tools that can be used to measure RS and RP. On one hand, the RS and RP problem can be formulated as convex

optimization problems that require the evaluation of Lyapunov functions-based Linear Matrix Inequalities (LMI's). This computationally efficient tool is suitable for those models that can be represented as uncertain state space models. On the other hand, there are physical models that are best described by the combination of a nominal model and a perturbation block that accounts for all the sources of uncertainties in the system. For this model representation, i.e. linear time-invariant model supplemented with linear-fractional uncertainty, it is more convenient and efficient to use Structured Singular Value (SSV) as the synthesis tool to study RS and RP.

The above concepts constitute the basic tools used in this work to attain the simultaneous design and control of dynamic systems in the presence of process uncertainties and external perturbations. Therefore, the rest of this section is dedicated to review the aforementioned concepts. Section 2.2.1 reviews the different structures that can be used to represent uncertain model. In Section 2.2.2, the LMI's theory and the LMI's standard problems are introduced. Then, Section 2.2.3 revises the background related to Lyapunov theory. In Section 2.2.4, the concept of Structured Singular Value (SSV) and its potential use to measure process variability is discussed.

2.2.1 Model Uncertainty Representation

Uncertain dynamical models are fundamental in robust control theory. These models are based on parameters which actual or true values are not exactly known but they lay within a known range of values. Traditionally, robust controllers are designed based on a linear dynamic model, which describes the process dynamics around a nominal steady-state operating point, complemented with uncertainty in model parameters, which accounts for the differences between the nominal linear model and the actual process behaviour. These models are usually referred to as *robust models* or *uncertain models*.

Uncertainty can be classified into two classes:

Parametric Uncertainty: it refers to the uncertainty that is directly related to the system's physical parameters, i.e. *structured uncertainty*. This class of uncertainty can be mathematically represented as follows:

$$a_i = \bar{a}_i + \theta_i \tag{2.1}$$

where $\bar{a}_i$ and θ_i denote the parameter's nominal value and the corresponding uncertainty region for the parameter a_i. In general, $\bar{a}_i$ and θ_i define a range of values that the parameter a_i can take at any time during the systems dynamic operation. In chemical processes, the uncertain parameter θ_i is commonly described in terms of two extreme values, that is,

$$\theta_i = [\theta_{i,\max}, \theta_{i,\min}] \tag{2.2}$$

where $\theta_{i,max}$ and $\theta_{i,min}$ respectively represent the upper and lower bounds that delimits the uncertainty region for each uncertain parameter, a_i. If $\boldsymbol{\theta}=[\theta_1,\theta_2,\ldots,\theta_n]$ represents the vector of all the individual sources of structured uncertainty around a given system, then each combination of θ_i's define a vertex of an hyperrectangle, usually called the *parameter box*, that encloses all the parameter space uncertain region $\Re^n$. To illustrate the above, consider a system with two uncertain parameters defined as in (2.1) and (2.2). As it is shown in Figure 2.1, each vertex, (black circles) represent particular combinations between the lower and upper limits of each of the uncertain parameters, e.g. ($\theta_{1,min}$, $\theta_{2,max}$) Thus, the set of all the vertexes in Figure 2.1 generate a rectangle (shaded area enclosed by dashed lines) which delimits the uncertain parameter space feasible region, i.e. parameter box in $\Re^2$.

This type of uncertainty arises due to imperfect knowledge of process physical parameters and model parameters that are in error due to process nonlinearities following changes in the operating conditions. Although there are other parametric uncertainty descriptions available in the literature, e.g. bounds on the rate of variation and polytopic parameter space representations[34], the uncertain parameter representation given in (2.2) is relevant for this work since the robust process model used to test RS and RP will be defined in terms of this description.

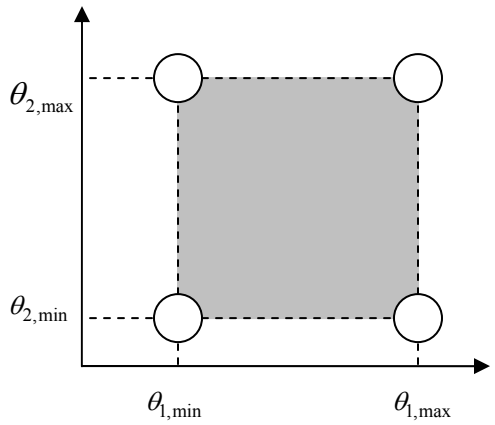

Figure 2.1 Parameter box for two uncertain parameters.

Dynamic Uncertainty: This type of uncertainty description is used when the system's dynamic behaviour is so complex that the individual sources of uncertainty cannot be easily identified. Then, all the sources of uncertainty that generate this behaviour are lumped into one single perturbation block within the model structure. Accordingly, this type of uncertainty description is also referred to as *unstructured uncertainty*. Dynamic uncertainty is often associated with the fact that the process behaviour at high frequencies is not accurately measured or the system's response to changes in the operating conditions is highly nonlinear. In principle, any mathematical model of a physical system contains is prone to this type of model uncertainty since models are only an approximated representation of the true physical system.

This source of uncertainty represents the physical process G as a nominal process model ($\breve{G}$) complemented with a perturbation block (L) that accounts for the dynamic uncertainty. Depending on the locations of the unstructured uncertainty in the closed-loop system, four single perturbations are used to quantify this class of uncertainty: additive uncertainty (L_A), multiplicative output uncertainty (L_O), multiplicative input uncertainty (L_I), and inverse multiplicative output uncertainty (L_E). Their corresponding mathematical descriptions are as follows:

$$
\begin{aligned}
G &= \breve{G} + L_A \\
G &= \breve{G}\left(I + L_O\right) \\
G &= \breve{G}\left(I + L_I\right) \\
G &= \breve{G}\left(I - L_E\right)^{-1}
\end{aligned}
\tag{2.3}
$$

Figure 2.2 illustrates the interconnections between each of these perturbations blocks and the closed-loop nominal model for each one of the uncertainty descriptions. The magnitude of the perturbation block in (2.3) may be defined in terms of bounds obtained from different matrix measures. For example a perturbation block may be estimated as follows[35]:

$$
L_i = \delta_i \Delta_i, \quad \sigma_{\max}\left(\Delta_i\right) \leq 1 \tag{2.4}
$$

where Δ_i is a normalized perturbation block bounded in terms of its singular values, i.e., the maximum singular value of Δ_i has to be less than or equal to the unity, and δ_i is a scalar weight on Δ_i.

One of the limitations associated with the description shown in (2.3) is that the bounds used to represent uncertainty in the model may be considerably larger than the real uncertainty. Hence, models that consider in its structure this uncertainty description produce controllers with significant conservatism.

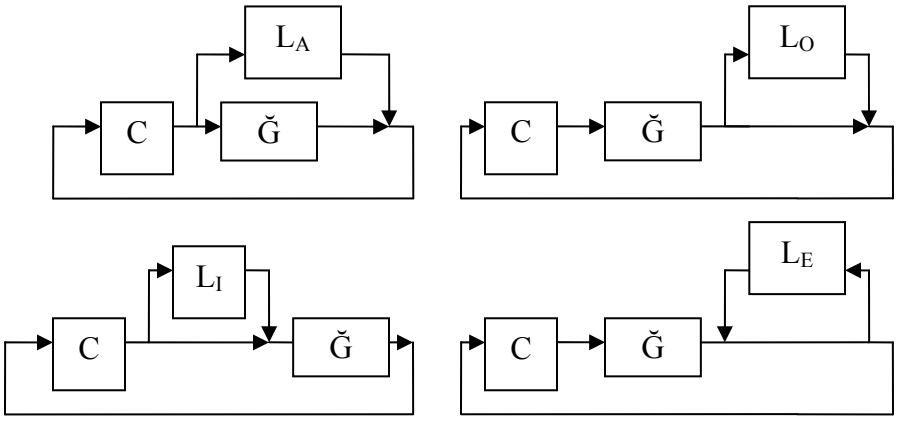

Figure 2.2 Unstructured uncertainty description.

In general, robust controllers are designed with process models that consider a combination of parametric and dynamic uncertainty within its structure. The selection of the model's mathematical representation is thus determined by the uncertainty descriptions and the synthesis tool to be used to design the control algorithm. Most of the uncertain models used in robust control theory are based on linear differential inclusions[36] given as follows:

$$\dot{\mathbf{x}} \in \Omega\mathbf{x}, \quad \mathbf{x}(0) = \mathbf{x_0} \tag{2.5}$$

where $\mathbf{\Omega}$ is a subset in $\Re^{nxn}$ and x is the system's state. Every trajectory in (2.5) satisfies:

$$\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x}, \quad \mathbf{x}(0) = \mathbf{x_0} \tag{2.6}$$

where $\mathbf{A}(t) \in \mathbf{\Omega}$ contains the uncertain or time-varying coefficients of the uncertain linear system. It should be noticed that the uncertainty associated with $\mathbf{A}(t)$ defines an infinite number of solutions of equation (2.6). Thus, a relaxed version of (2.6) based on the Relaxation Theorem[36] has been proposed for studying these systems as follows:

$$\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x}, \quad \mathbf{A}(t) \in \mathbf{CoA}(t) \tag{2.7}$$

The above confirms that $\mathbf{A}(t)$ is a convex set for every trajectory x and t if $\mathbf{CoA}(t) \supseteq \mathbf{A}(t)$. This means that every trajectory in (2.6) is also a trajectory of the relaxed equations' set (2.7). This important approximation of the original problem is relevant for model uncertainty representation since it allows the nonlinear or time varying model behaviour to be described by a finite set of linear models. The above results can be expanded to consider continuous state space models as follows:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(t)\mathbf{x} + \mathbf{B}(t)\mathbf{u}, \quad \mathbf{x}(0) = \mathbf{x_0} \\ \mathbf{y} &= \mathbf{C}(t)\mathbf{x} + \mathbf{D}(t)\mathbf{u} \end{aligned} \tag{2.8}$$

with the set $\mathbf{\Omega}$ defined as follows:

$$\begin{bmatrix} \mathbf{A}(t) & \mathbf{B}(t) \\ \mathbf{C}(t) & \mathbf{D}(t) \end{bmatrix} \in \mathbf{\Omega} \tag{2.9}$$

The above definition constitutes the general representation used in this work to represent an uncertain state space dynamic model. This definition can then be used to formulate different model representations.

A special case of (2.8) is when the state space model is a linear time-invariant model, i.e. the coefficients of the state space matrices do not depend affinely on any uncertain or time-varying parameter. In this case, the set $\mathbf{\Omega}$ is made of one single matrix as follows:

$$\mathbf{\Omega} = \left\{ \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \right\} \tag{2.10}$$

On the other hand, if the model coefficients are defined as bounded parameters like (2.2), then the set $\mathbf{\Omega}$ can be described as a polytope, i.e. a convex hull formed with the extreme values of the model's coefficients. This can be mathematically described as follows[36]:

$$\begin{bmatrix} \mathbf{A}(t) & \mathbf{B}(t) \\ \mathbf{C}(t) & \mathbf{D}(t) \end{bmatrix} \in \mathbf{Co}\left\{ \begin{bmatrix} \mathbf{A}_1 & \mathbf{B}_1 \\ \mathbf{C}_1 & \mathbf{D}_1 \end{bmatrix}, \ldots, \begin{bmatrix} \mathbf{A}_j & \mathbf{B}_j \\ \mathbf{C}_j & \mathbf{D}_j \end{bmatrix}, \ldots, \begin{bmatrix} \mathbf{A}_n & \mathbf{B}_n \\ \mathbf{C}_n & \mathbf{D}_n \end{bmatrix} \right\} \tag{2.11}$$

where the j^{th} vertex of the polytope represents an LTI system formed with a particular combination of the extreme values of the state space matrices coefficients. Thus, the uncertain model shown in (2.8) is described as a finite set of LTI systems. This class of model representation is widely used to analyze RS and RP with Linear Matrix Inequalities (LMI's).

More generally, uncertain model representation with both parametric and dynamic uncertainty can be defined in terms of a norm-bound differential inclusion. In this case, the general model description presented in (2.8) is modified as follows:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}_p\mathbf{p} + \mathbf{B}_u\mathbf{u}, \quad \mathbf{x}(0) = \mathbf{x_0} \\ \mathbf{q} &= \mathbf{C}_q\mathbf{x} + \mathbf{D}_{qp}\mathbf{p} + \mathbf{D}_{qu}\mathbf{u} \\ \mathbf{y} &= \mathbf{C}_y\mathbf{x} + \mathbf{D}_{yp}\mathbf{p} + \mathbf{D}_{yu}\mathbf{u} \\ \mathbf{p} &= \mathbf{\Delta}(t)\mathbf{q}, \quad \left\|\mathbf{\Delta}(t)\right\| \le 1 \end{aligned} \tag{2.12}$$

where $\Delta : \Re_+ \to \Re^{nqxnq}$, with $\|\mathbf{\Delta}(t)\| \le 1$ for all time t and represents a normalized perturbation block as defined in (2.4). Figure 2.3 shows a schematic representation of the system (2.12). As shown, this system is described by an LTI nominal system **M**, with input **u**, output **y** and a time-varying feedback matrix **Δ** with input **p** and output **q**, respectively. This model representation is also referred to as a *linear fractional model*. For this system, the set **Ω** is defined as follows[36]:

$$\mathbf{\Omega} = \left\{ \breve{\mathbf{A}} + \breve{\mathbf{B}}\mathbf{\Delta}\left(\mathbf{I} - \mathbf{D}_{pq}\mathbf{\Delta}\right)^{-1}\breve{\mathbf{C}} \;\middle|\; \|\mathbf{\Delta}(t)\| \le 1 \right\}$$
$$\breve{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_u \\ \mathbf{C}_y & \mathbf{D}_{yu} \end{bmatrix}, \quad \breve{\mathbf{B}} = \begin{bmatrix} \mathbf{B}_p \\ \mathbf{D}_{yp} \end{bmatrix}, \quad \breve{\mathbf{C}} = \begin{bmatrix} \mathbf{C}_q \\ \mathbf{D}_{qu} \end{bmatrix}^T \tag{2.13}$$

Thus, **Ω** denotes a convex set with the image of the unit ball under a linear fractional mapping.

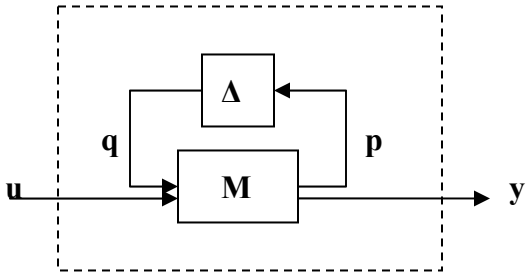

Figure 2.3 Linear Fractional Model

This model representation is more generic since the structure of the uncertain block **Δ** can consider multiple sources of uncertainty located in different sections in the closed loop system (see Figure 2.2) where each element of the uncertain block can be associated with a particular source of uncertainty in the system. Due to the flexibility of this representation, different uncertain model representations can be posed as linear fractional models, e.g. an uncertain polytope state space model can be reformulated as a linear fractional model. As it is shown on Section 2.2.4, linear fractional models are essential to study RS and RP with the Structured Singular Value (SSV) based analysis.

2.2.2 Linear Matrix Inequalities (LMI's)

Linear matrix inequalities play a key role within the robust control theory framework; they have emerged as a computationally efficient tool to address a wide variety of matrix variables problems that appear in systems and control theory. Instead of applying the classical approach to solve the matrix variable problem, i.e. analytical or frequency domain-based methods, the problem is reformulated as convex (or quasiconvex) optimization problems that are subject to a finite set of matrix inequalities that are linear (or affine) with respect to the original set of matrix variables. Due to the recent developments in interior-point optimization methods, convex optimization problems involving LMI's are now tractable and can be efficiently solved with a relatively high accuracy[34].

There are several factors that make LMI's a powerful computational tool:

- Many complex control system problems that involve several matrix variables with different structures can be formulated in terms of a linear matrix variable problem.

- The LMI formulation can be exactly solved by the use of convex optimization algorithms, referred to as *LMI-solvers*.

- LMI techniques are an alternative solution to those problems with multiple constraints and objectives for which the classical analytical methods either fail or have numerical difficulties in finding a solution.

Therefore, LMI's formulations have been used to treat a wide variety of robust control problems. The most typical LMI applications in control design problems includes the following: robust H_∞ control [37-38], robust H_2 control[39], gain-scheduled controller design[40], robust model predictive control[41], and robust stability and performance of linear time-invariant systems[42]. A more comprehensive review that presents the control-theoretic problems that can be solved with LMI techniques is presented by Boyd et al.[36].

The general form of a linear matrix inequality is given as:

$$\mathbf{F}(x) := \mathbf{F_0} + \sum_{i=1}^{m} \mathbf{F}_i x_i > \mathbf{0} \tag{2.14}$$

where $x \in \Re^m$ are the design variables and $\mathbf{F}_i = \mathbf{F}_i^T \in \Re^{nxn}$ are symmetric matrices that are given. The inequality sign '>' indicates that the smallest eigenvalue of $\mathbf{F}(x)$ is positive, i.e. $\mathbf{F}(x)$ is positive definite. This LMI is convex on x, that is, since $\mathbf{F}(x_1)>\mathbf{0}$ and $\mathbf{F}(x_2)>\mathbf{0}$, then it follows that $\mathbf{F}\left(\left(x_1 + x_2\right)/2\right) > \mathbf{0}$; thus the feasible set in (2.14) is a convex subset of $\Re^m$. Although (2.14) does not have a formal analytical solution, it can be solved numerically with the guarantee of finding a solution set, when one exists, that is convex in the design variables.

The numerical solution of an LMI is a binary decision problem, i.e. the answer to an LMI problem is either yes or no. For example, the formulation in (2.14) can be used to determine if a given system is asymptotically stable, the solution in this case is either yes or no. In other cases, the LMI problem is formulated in terms of a real positive parameter α, and it is desired to measure a property of the system due to changes in this parameter, i.e. the problem is formulated so as to find the largest value of α for which the LMI problem has a feasible solution. Typically, this requires an iterative solution on α and therefore, the binary decision problem has to be solved as many times as the number of iterations required for convergence. On the other hand, most of the control systems problems are formulated as $\mathbf{F}(x)<\mathbf{0}$ and $\mathbf{F}(x)<\mathbf{G}(x)$. These constraints are only special cases of the general representation shown in (2.14) since they can be rewritten as $-\mathbf{F}(x)>\mathbf{0}$ and $\mathbf{G}(x)-\mathbf{F}(x)>\mathbf{0}$, respectively.

The three basic LMI problems that are frequently solved for control systems design are the following:

LMI feasibility problem (FP).– The goal in this problem is to find a set of values in the design variables, x^{sol}, such that they satisfy a given linear matrix inequality problem. For example, given the LMI formulation in (2.14), the corresponding feasibility problem can be defined as follows: find the feasible set in the matrix variables values such that $\mathbf{F}(x^{sol})>\mathbf{0}$ holds. An alternative solution to this problem is to find that x^{sol} is an empty set which corresponds to the case that the proposed LMI problem is infeasible. It should be noted that

the feasible solution set is not being optimized, i.e. this problem only specifies if the LMI is stable or not but it does not provide measures of how far (or close) the LMI formulation is from instability or stability.

LMI eigenvalue problem (EVP).– This problem deals with the minimization of the maximum eigenvalue of a matrix (χ) that depends affinely on a variable and that is subject to LMI's constraints as follows:

$$\begin{aligned} &\min_{x} \chi \\ &s.t. \\ &\quad \chi\mathbf{I} - \mathbf{A}(x) > \mathbf{0} \\ &\quad \mathbf{B}(x) > \mathbf{0} \end{aligned} \tag{2.15}$$

where **A** and **B** are symmetric matrices that depend affinely on the design variable x. This problem can be reformulated in different equivalent forms[36]; e.g., it can be rewritten in terms of a linear scalar function ϑ that depends affinely on x and that is subject to LMI constraints as follows:

$$\begin{aligned} &\min_{x} \vartheta(x) \\ &s.t. \\ &\quad \mathbf{F}(x) > \mathbf{0} \end{aligned} \tag{2.16}$$

The above formulation is also known as the *linear objective minimization problem*. In this type of problem, an objective function is to be minimized (or maximized) such that it satisfies a given set of LMI's. Due to its flexibility, this generic problem is often used in control theory to design controllers which formulation is based on LMI's. In most of the cases, the function ϑ measures a particular property of the system of interest.

LMI generalized eigenvalue problem (GEVP).- This is a general formulation of the previous problem. In this case, the eigenvalues are subject to a pair of matrices that affinely depend on the design variables x. Thus, the GEVP can be posed as follows:

$$
\begin{aligned}
&\min_{x} \chi \\
&s.t. \\
&\quad \chi \mathbf{A}(x) - \mathbf{B}(x) > \mathbf{0} \\
&\quad \mathbf{A}(x) > \mathbf{0} \\
&\quad \mathbf{B}(x) > \mathbf{0} \\
&\quad \mathbf{C}(x) > \mathbf{0}
\end{aligned}
\tag{2.17}
$$

where **A**, **B** and **C** have the same properties as the matrix **A** in (2.15). By inspection of (2.17), it is noticed that the constraints are convex and the objective function is quasiconvex. Accordingly, the GEVP solution requires quasiconvex programming. This means that for a feasible x_1, x_2 and $0 \le \theta \le 1$, the following inequality holds[36]:

$$
\begin{aligned}
\chi_{\max}\left(A\left(\theta x_1 + (1-\theta)x_2\right)\right), B\left(\theta x_1 + (1-\theta)x_2\right) \le \\
\max\left\{\chi_{\max}\left(\mathbf{A}(x_1), \mathbf{B}(x_1)\right), \chi_{\max}\left(\mathbf{A}(x_2), \mathbf{B}(x_2)\right)\right\}
\end{aligned}
\tag{2.18}
$$

where $\chi_{\max}(\mathbf{A}(x),\mathbf{B}(x))$ represents the largest generalized eigenvalue of the term $\chi\mathbf{A}(x)$-$\mathbf{B}(x)$ with $\mathbf{B}>\mathbf{0}$. Although the GEVP is not convex, the algorithms used to solve this problem are computationally efficient and similar to those used for both the feasibility and the eigenvalue problems presented above.

A special class of problems that use the LMI-framework are Lyapunov method-based problems. These methods, widely used in control theory to establish the stability and the performance of a given control system, are reviewed in the following section.

2.2.3 Lyapunov functions for robust stability and robust performance.

In 1892, Lypunov[43] published the first theoretical work that discuss the basic notions of stability in nonlinear systems. In this work, Lyapuvov proposed two approaches to study the stability of a given control system. The first approach is based on the idea that a nonlinear system may behave as a linear system around an infinitesimal nominal steady-state operating point. This approach referred to as the linearization method, examines the local stability of a nonlinear system around an equilibrium point by approximating the nonlinear model into a linear model. Because all physical systems exhibit nonlinear behaviour, the linearization method is only helpful for establishing stability of a system in an infinitesimal

neighbourhood of an operation point for which the behaviour is approximately linear. Consequently, finite bounds in the nominal operating conditions for which local stability is valid cannot be explicitly calculated by using the linearization concept and they highly depend on the physical system's degree of nonlinearity. Moreover, most of the physical systems are subject to perturbations which may drive the system outward the nominal operating region, in which case the concept of local stability is not valid any longer. These restrictions on the linearization method motivated the necessity to propose a second approach that does not require an assumption of linearity and it is referred to as Lyapunov's direct method.

Lyapunov's direct method determines the stability properties of a nonlinear system by constructing a scalar "energy-like" function (Lyapunov function) for the system and examining how this function develops in the time domain. Unlike the linearization method, the direct method is not restricted to local motion, that is, it determines the stability of the physical system *globally*. The direct method is based on a fundamental physical observation[44]:

> *"If the total energy of a mechanical (or electrical) system is continuously dissipated, then the system, whether linear or nonlinear, must eventually settle down to an equilibrium point."*

To demonstrate the previous statement, the following autonomous system can be considered:

$$\dot{\mathbf{x}} = f(x) \tag{2.19}$$

To study global stability on a physical system, the direct method requires the definition of a Lyapunov function, $\mathbf{V}(x)$, such that it is positive definite with continuous partial derivatives, and it decreases along every nonzero state trajectory of (2.19). Then, the equilibrium at the origin of the system (2.19) is globally asymptotically stable if[44]:

- $\mathbf{V}(x)$ *is positive definite.*
- $\dot{\mathbf{V}}(x)$ *is negative definite.*

- $\mathbf{V}(x) \to \infty \quad as \quad \|\mathbf{x}\| \to \infty$

The fundamental limitation with this approach is that there is not a formal procedure to obtain $\mathbf{V}(x)$ for a particular system. Thus different structures can be used as Lyapunov functions to test the global stability and this may lead to different results according to the particular choice of these functions. To circumvent this shortcoming, general Lyapunov functions have proposed for different type of systems. The most common function used as a candidate to test the global stability and performance in linear systems is a *Quadratic function*. Given $\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x}$, a quadratic Lyapunov function (QLF) that is related to the states of the system is defined as follows[36]:

$$\mathbf{V}(x) = \mathbf{x}(t)^T \mathbf{P}\mathbf{x}(t) \tag{2.20}$$

where **P** is a given symmetric positive definite matrix that is also known as the *Lyapunov matrix*. This matrix is basis-independent, that is, it does not depend on the system's parameters.

The definition shown in (2.20) is fundamental for this work since closed-loop process stability and process variability are measured in terms of this quadratic function. Therefore, the robust stability and robust performance criteria are defined as convex optimization problems that are subject to LMIs constraints, i.e. FP and EVP. These particular tests are presented in more detail in chapter 3.

2.2.4 Structured Singular Value (SSV) analysis

LMI's is an efficient tool to analyze models with a reduced number of structured uncertainties since the computational burden associated with these problems grows exponentially as the number of uncertain parameters considered in the model structure increases. To deal with this issue, the uncertainties that appear on different sections of the system are lumped together in order to avoid an increase in the dimensionality of the problem. This is generally done by increasing the uncertain model coefficient's range of values considered for analysis. This remedy partially alleviates the computational burden at the expense of having a model with uncertainty descriptions that are larger than the true

uncertainty. Consequently, the RS and RP measures obtained with these models are expected to be conservative. These facts restrict the use of LMI's as a robust analysis tool.

The Structured Singular Value (SSV)[45] based approach strictly addresses RS and RP in MIMO processes with several sources of uncertainties, e.g. structured or unstructured uncertainty, appearing in different sections of the closed loop system. This matrix function, denoted by SSV or μ, is based on the generalization of the singular value and the spectral radius for constant matrices. This technique requires the system's dynamic behaviour to be represented as an uncertain linear fractional model, see Section 2.2.1. Considering the linear fractional model description given in Figure 2.3 the matrix $\mathbf{M} \in \mathbf{C}^{nxn}$ represents all the LTI models which parameters are perfectly known. This matrix is generally formulated by combining the process linear model, the controller and the scalar weights or bounds on the uncertain model parameters. The input variable, **u**, accounts for all the possible exogenous perturbations that may affect the system, e.g. disturbances, set points and measurement noise, whereas the variable *y*, groups all the system's outputs. The perturbation block $\mathbf{\Delta} \in \mathbf{C}^{nxn}$ considers in its structure the interconnections between the different sources of uncertainties and the nominal plant **M**. The structure of **Δ** is problem specific and depends on the number of blocks to be considered in the system, their types and dimensions. The type blocks can be either repeated scalar blocks, i.e. structured uncertainty, or full blocks, i.e. unstructured uncertainty. The generic definition of the matrix **Δ** is as follows[46]:

$$\Delta = \left\{ diag\left[\delta_1 \mathbf{I}_{r1}, \ldots, \delta_s \mathbf{I}_{rs}, \Delta_1, \ldots, \Delta_f\right] : \delta_i \in \mathbf{C}, \Delta_j \in \mathbf{C}^{m_j x m_j} \right\} \tag{2.21}$$

Where the subscripts *s* and *f* are nonnegative integers that specify the number of repeated scalar and full blocks, respectively. Similarly, the subscripts $r_1,\ldots,r_s$ and $m_1,\ldots,m_f$ specify the dimensions of each block, e.g. the first repeated scalar block is of dimensions $r_1 x r_1$ whereas the j^{th} full block is of dimensions $m_j x m_j$, respectively. For consistency,

$$\sum_{i=1}^{s} r_i + \sum_{j=1}^{f} m_j = n \tag{2.22}$$

Each of the blocks in (2.21) is often bounded as in equation (2.4):

$$\mathbf{\Delta} = \{\Delta \in \mathbf{\Delta} : \sigma_{\max}(\Delta) \leq 1\} \tag{2.23}$$

Based on the above definitions, the structured singular value problem, $\mu_{\mathbf{\Delta}}(\mathbf{M})$, is defined as follows[45]:

$$\mu_{\mathbf{\Delta}}(\mathbf{M}) := \frac{1}{\min\{\sigma_{\max}(\Delta) : \Delta \in \mathbf{\Delta}, \det(\mathbf{I} - \mathbf{M\Delta}) = 0\}} \tag{2.24}$$

if no Δ exists such that it makes $\mathbf{I}$-$\mathbf{M\Delta}$ *singular, then* $\mu_{\Delta}(\mathbf{M}) := 0$.

Thus, $\mu_{\mathbf{\Delta}}(\mathbf{M})$ can be viewed as a measure of the smallest structure $\mathbf{\Delta}$ that causes instability in the system. From (2.24) it follows that:

$$\mu_{\mathbf{\Delta}}(\mathbf{M}) := \max_{\Delta \in \mathbf{\Delta}} \rho(\mathbf{M\Delta}) \tag{2.25}$$

where ρ stands for the spectral radius. Thus, $\mu_{\mathbf{\Delta}}(\mathbf{M})$ is a generalization of the spectral radius , $\rho(\mathbf{M})$, and the maximum singular value, $\sigma_{max}(\mathbf{M})$.

Although definition (2.24) represents a formal mathematical definition of the structured singular value function, this definition is not used explicitly for the estimation of μ since it poses a complex optimization problem. To circumvent this issue, upper bounds for the SSV are generally sought. For example, for the case where the uncertain block is defined as a single repeated scalar block, i.e. $\mathbf{\Delta} = \{\delta \mathbf{I} : \delta \in \mathbf{C}\}$, then $\mu_{\mathbf{\Delta}}(\mathbf{M}) = \rho(\mathbf{M})$. On the other hand, if the perturbation block is described by a full block, i.e. $\mathbf{\Delta} = \mathbf{C}^{nxn}$, then $\mu_{\mathbf{\Delta}}(\mathbf{M}) = \sigma_{max}(\mathbf{M})$. Then, for the general representation given in (2.21) and the μ definition (2.24) it follows that:

$$\rho(M) \leq \mu_{\mathbf{\Delta}}(\mathbf{M}) \leq \sigma_{\max}(\mathbf{M}) \tag{2.26}$$

Although these limits do not represent tight descriptions for μ, they can be reformulated by considering transformations on the nominal plant $\mathbf{M}$ that are insensitive to $\mu_{\mathbf{\Delta}}(\mathbf{M})$ but do affect ρ and σ_{max}. To do this, the following scaling matrices are defined:

$$\mathbf{Q} = \{Q \in \mathbf{\Delta} : QQ^* = \mathbf{I}_n\}$$
$$\mathbf{Đ} = \begin{Bmatrix} diag[\mathbf{Đ}_1, \ldots, \mathbf{Đ}_s, \mathbf{d}_1\mathbf{I}_{m1}, \ldots, \mathbf{d}_{f-1}\mathbf{I}_{mf-1}, \mathbf{I}_{mf}] \\ \mathbf{Đ}_i \in \mathbf{C}^{r_i x r_i}, \mathbf{Đ}_i = \mathbf{Đ}_i^* > \mathbf{0}, d_j \in \Re, d_j > 0 \end{Bmatrix} \quad (2.27)$$

If for any $Q \in \mathbf{Q}$, $Đ \in \mathbf{Đ}$, and $\Delta \in \mathbf{\Delta}$, then $Q^* \in \mathbf{Q}$, $Q\Delta \in \mathbf{\Delta}$, $\Delta Q \in \mathbf{\Delta}$, and the following equalities hold:

$$\begin{gathered} \sigma_{\max}(\Delta Q) = \sigma_{\max}(Q\Delta) = \sigma_{\max}(\mathbf{\Delta}) \\ Đ\Delta = \Delta Đ \end{gathered} \quad (2.28)$$

Therefore, for all $Q \in \mathbf{Q}$ and $Đ \in \mathbf{Đ}$, it can be stated that:

$$\mu_{\mathbf{\Delta}}(\mathbf{MQ}) = \mu_{\mathbf{\Delta}}(\mathbf{QM}) = \mu_{\mathbf{\Delta}}(\mathbf{Q}) = \mu_{\mathbf{\Delta}}(\mathbf{ĐMĐ}^{-1}) \quad (2.29)$$

Consequently, inequality (2.26) can be reformulated as follows[45]:

$$\max_{Q \in \mathbf{Q}} \rho(\mathbf{QM}) \le \mu_{\mathbf{\Delta}}(\mathbf{M}) = \max_{\Delta \in \mathbf{\Delta}} \rho(\mathbf{\Delta M}) \le \inf_{D \in \mathbf{D}} \sigma_{\max}(\mathbf{ĐMĐ}^{-1}) \quad (2.30)$$

where the above upper and lower limits represent tighter bounds on μ than those presented in (2.26). Although the lower bound is always equal to the function μ[45], its corresponding optimization problem is not convex, i.e., it is subject to local solutions and a global solution may not be found. On the other hand, the upper bound can be reformulated as follows[47]:

$$\begin{aligned} &\mu_{\mathbf{\Delta}}(\mathbf{M}) < \beta \Leftarrow \exists \mathbf{Đ} > \mathbf{0} : \sigma_{\max}(\mathbf{ĐMĐ}^{-1}) < \beta \\ &\Leftrightarrow \exists \mathbf{Đ} > \mathbf{0} : \mathbf{Đ}^{-1}\mathbf{M}^*\mathbf{Đ}^2\mathbf{MĐ}^{-1} - \beta^2\mathbf{I} < \mathbf{0} \\ &\Leftrightarrow \exists \mathbf{Đ} > \mathbf{0} : \mathbf{M}^*\mathbf{Đ}^2\mathbf{M} - \beta^2\mathbf{Đ}^2 < \mathbf{0} \end{aligned} \quad (2.31)$$

where β^2 is an scalar and $\mathbf{M}^*$ is the conjugate of $\mathbf{M}$. The matrix $\mathbf{Đ}^2$ can be redefined as follows:

$$\mathbf{P} = \{\mathbf{Đ}^2 \in \mathbf{P} : \mathbf{P} = \mathbf{P}^T, \mathbf{P} > 0\} \quad (2.32)$$

Consequently, the last inequality in (2.31) is equivalent to:

$$\mathbf{M}^*\mathbf{PM} - \beta^2\mathbf{P} < \mathbf{0} \quad (2.33)$$

This last expression is an LMI which eigenvalues depend affinely on the pair of matrices **P** which can be viewed as a Lyapunov function weighting matrix, and **M** is based, as explained above, on the nominal linear model of the plant. Thus, the calculation of the SSV can be formulated according to (2.33) as a Generalized Eigenvalue Problem (GEVP) that was reviewed in the previous subsection. Thus, since the upper bound in (2.30) is redefined as a convex LMI, the global minimum to this problem can be found. The drawback of this calculation is that the upper bound is not always equal to μ though it usually provides a good approximation to its actual value. For uncertain blocks satisfying $2s+f \leq 3$, the upper bound is always equal to μ, whereas for blocks that satisfies $2s+f > 3$, there are matrices for which the function μ is less than the infimum (inf). More details about the relationship between μ and its upper bound are given by Packard and Doyle[48].

The most widespread use of the function μ is for testing the robustness of a given system. The robust stability test expressed in terms of μ can be formulated as follows[49]:

Consider the linear fractional model shown in Figure 2.3. Assume that the nominal model $\mathbf{M} \subset \mathbf{C}^{nxn}$ is stable and that the perturbation block **Δ** has been defined as in (2.21). Then the system in Figure 2.3 is stable for all perturbations $\mathbf{\Delta} \subset \mathbf{C}^{n_p x n_q}$ if and only if:

$$\mu_{\mathbf{\Delta}}(\mathbf{M}) < 1 \tag{2.34}$$

where the superscripts n_q and n_q denotes the number of inputs and outputs between the nominal plant **M** and the uncertain block **Δ**, respectively.

The μ-based robust performance test is defined in terms of the worst-case random mean squares (RMS) gain from the input **u** to the output **y** in the presence of uncertainties **Δ**. To derive such a test, an additional uncertain block has been added to the system presented in Figure 2.3; this structure has been interconnected with the system's inputs and outputs as it is shown in Figure 2.4. The new uncertain block ($\mathbf{\Delta_p}$) takes into account the external perturbations affecting the system's outputs. As it is shown in Figure 2.4, the two uncertain blocks can be combined to form a single uncertain block. Assume that the nominal plant can be partitioned as follows:

$$M = \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix} \tag{2.35}$$

then, the transfer matrix from input **u** to output **y** in Figure 2.4 can be defined as follows:

$$\mathbf{y} = F(\mathbf{M}, \boldsymbol{\Delta}_u)\mathbf{u} \tag{2.36}$$

where $F(\mathbf{M}, \boldsymbol{\Delta}_u)$ is a linear fractional model defined as:

$$F(\mathbf{M}, \boldsymbol{\Delta}_u) = \mathbf{M}_{22} + \mathbf{M}_{21}\boldsymbol{\Delta}_u(\mathbf{I} - \mathbf{M}_{11}\boldsymbol{\Delta}_u)^{-1}\mathbf{M}_{12} \tag{2.37}$$

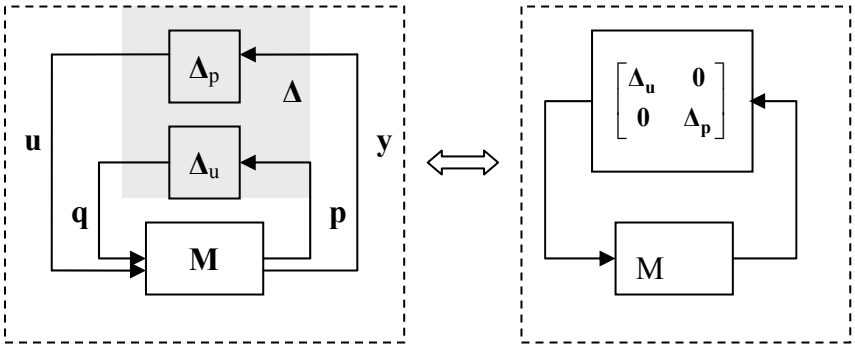

Figure 2.4 Block diagram structure for robust performance.

Based on the above definitions, the robust performance condition can now be stated as follows:

$$\left\| F(\mathbf{M}, \boldsymbol{\Delta}_u) \right\|_\infty = \sup \sigma_{\max}\left(F(\mathbf{M}, \boldsymbol{\Delta}_u)\right) < 1 \tag{2.38}$$

This criteria can also be expressed in terms of the function μ as follows[49]:

The nominally stable system **M** subjected to the block diagonal uncertainty, $\boldsymbol{\Delta}_\mathbf{u}$ satisfies the robust performance condition $\left\| F(\mathbf{M}, \boldsymbol{\Delta}_u) \right\|_\infty < 1$ if and only if:

$$\sup \mu_{\boldsymbol{\Delta}_p}(\mathbf{M}) < 1 \tag{2.39}$$

This is a very useful property in structured singular value analysis since it shows that the robust performance test is equivalent to the robust stability test given by equation (2.34) with an augmented perturbation block $\mathbf{\Delta}_p$. Accordingly, the robust performance criteria (2.39) can also be stated as follows:

$$\mu_{\mathbf{\Delta}=diag\{\mathbf{\Delta}_u,\mathbf{\Delta}_p\}}(\mathbf{M}) < 1 \tag{2.40}$$

In general, the performance uncertain block is defined as a complex full block. With regards to computation of the function μ in (2.39), upper and lower bounds for this problem are estimated using the definitions given in (2.33) and (2.30), respectively. A more comprehensive review on the definition and properties of the function μ can be found elsewhere[45-47].

The SSV (μ) method is fundamental for the present work since it is used as the computational tool to evaluate a new robust performance criterion. This performance measure can be also used to determine the time-dependent critical profile in the external perturbations such that will cause the largest process output deviation with respect to a nominal operating point. This worst-case disturbance analysis is formulated as a linear fractional model with the general interconnection structure shown in Figure 2.4. The formal mathematical description of this robust performance criterion is presented in chapter 4.

2.3 Model identification

The robust control tools reviewed in the previous section require that the process will be modelled by robust models given by a nominal linear model supplemented with a suitable uncertainty description. Thus, a key component of the present work is the identification of these nominal linear models, and their corresponding uncertainty descriptions, from simulation results produced with the full nonlinear model of the process under study.

This section presents a brief review on systems identification concepts that has been used in this work to model the dynamic behaviour of the system under the effect of external perturbations and parametric uncertainties.

To be able to estimate robust stability and robust performance measures, it is necessary first to build an uncertain dynamic model that takes into account the process transient behaviour in the presence of uncertainties and external perturbations. Model uncertainty is usually described in terms of linear dynamic models complemented with uncertain model coefficients. Linear models are estimated under the assumption that the system behaves linearly around an infinitesimal neighbourhood of a nominal steady-state operating point. Although this description has limitations, they have been widely used in robust control theory because its structure is simple and there are well-known methods for the identification of models under this assumption. There are two basic approaches that can be used to obtain linear dynamic models:

1.-**Linearization. -** This approach estimates analytically or numerically a linear dynamic model from the system's rigorous mathematical mechanistic model. A key advantage associated with this method is that the model coefficients have a physical meaning. Thus, insight regarding the process transient behaviour can be easily obtained from this type of models. On the other hand, this method may be suitable for simple process models but it is generally impractical and computationally demanding for process models that involves many equations with many model parameters. When computed numerically, this method requires the evaluation of the Jacobian matrix which is computationally expensive for large-scale systems.

2.-**Identification. -** The second approach consists of performing a set of experiments (simulations) on the full nonlinear dynamic process model. These experiments usually involve an informative input signal that is used to simulate the process transient behaviour of a signal of interest, i.e. output signal. The input/output values recorded from the experiment are then used by an identification technique to infer a linear dynamic model. This method is relatively easier to implement and it offers the advantage that complex process dynamic behaviour can be characterized by simple low-order linear models. For example, the transient behaviour of a distillation column can be characterized by a set of simple FOPDT models. Furthermore, systems identification for linear dynamic models is a mature area that provides reliable solutions to the empirical modelling problem.

The ARMA (Auto-regressive Moving Average) discrete model structure, the most commonly used model structure in system identification, is given as follows:

$$y(t)+a_1y(t-1)+\ldots+a_ny(t-n)=b_1u(t-1)+\ldots+b_my(t-m) \tag{2.41}$$

where y is a signal of interest and u is the input signal, $a_1,\ldots,a_n$, and $b_1,\ldots,b_m$, represent the model parameters, and the subscripts n and m denote the order of the polynomial for the input and output, respectively. It should be noticed that $n>m$ has to be satisfied for physical causality.

This basic linear dynamic model description can be transformed to other alternative model structures. For example, taking the Laplace Transformon both sides in (2.41) yields[8]:

$$G(q)=\frac{b_1q^{-1}+\ldots+b_mq^{-m}}{1+a_1q^{-1}+\ldots+a_nq^{-n}} \tag{2.42}$$

where q^{-1} is the unit delay operator and $G(q)$ represents the transfer function model that relates the signal y to the input u, i.e. $y(q)=G(q)u(q)$, respectively. The LTI in (2.41) can also be represented as a state space model:

$$\begin{aligned}\mathbf{x}_{k+1}&=\mathbf{A}\mathbf{x}_k+\mathbf{B}\mathbf{u}_k\\ \mathbf{y}_k&=\mathbf{C}\mathbf{x}_k+\mathbf{D}\mathbf{u}_k\end{aligned} \tag{2.43}$$

where $\mathbf{x}=[x_1,\ldots,x_n]^T$ represents the system's state space vector. The state space matrices may take different representations for the same linear dynamic model. Assuming that $m=n-1$ in (2.41), then a state space realization of (2.43) can be described as follows:

$$\mathbf{A}=\begin{bmatrix}-a_1 & 1 & 0 & \ldots & 0\\ -a_2 & & 1 & & \\ \vdots & & & \ddots & \\ \vdots & & & & 1\\ -a_n & 0 & \ldots & \ldots & 0\end{bmatrix},\quad \mathbf{B}=\begin{bmatrix}b_1\\ b_2\\ \vdots\\ \vdots\\ b_n\end{bmatrix},\quad \mathbf{C}=\begin{bmatrix}1\\ 0\\ \vdots\\ \vdots\\ 0\end{bmatrix}^T,\quad \mathbf{D}=0 \tag{2.44}$$

The above state space matrices description is known as the observer canonical form of a state space model[50].

The above linear dynamic model's representations are considered parametric models since the system's behaviour is represented by a fixed number of model's coefficients. Alternatively, a linear system can also be described in terms of its response due to specific input signals. Thus, the LTI model in (2.41) can also be represented in the form of an impulse response (FIR) model:

$$y(kT) = \int_0^\infty g(\tau)u(kT-\tau)d\tau \tag{2.45}$$

where k is the sampling instant and T is the sampling period, these two elements specifies the time t in the discrete domain, i.e. $t=kT$, with $k=1,2,\ldots\infty$ and T equal to a constant value. After algebraic manipulations, it can be shown that (2.45) is equivalent to the following convolution model:

$$y(t) = \left[\sum_{k=1}^{\infty} g(k)q^{-k}\right]u(t) \tag{2.46}$$

where the function $g(k)$ represents the impulse-response coefficients or equivalently the output's response at time interval k to a unit pulse in the input signal u.

For the purpose of regression, the model in (2.41) can be written in the following compact form:

$$y(t) = \varphi^T(t)\boldsymbol{\psi} \tag{2.47}$$

where $\boldsymbol{\psi}$ represents the model's coefficients and the function $\varphi(t)$, known as thew regression vector, has the following structure:

$$\varphi(t) = \left[-y(t-1),\ldots,-y(t-n),\quad u(t-1),\ldots,u(t-m)\right]^T \tag{2.48}$$

To determine the model coefficients' values, it is necessary first to define a cost function such that it minimizes the differences between the actual linear system's response and the predicted linear model as per the following objective function:

$$V(\boldsymbol{\psi}, Z^N) = \frac{1}{N}\sum_{t=1}^{N} \ell(\xi)$$
$$\xi = y(t) - \hat{y}(t), \quad 1 \le t \le N \tag{2.49}$$

where Z^N is the input/output data set that is used for model identification and has been recorded from the identification experiment, ξ represents the errors between the actual system, *y(t)*, and the predicted model, $\hat{y}(t)$ and $\ell(\xi)$ is a criterion used to reduce the errors in the predicted model. The choice of $\ell(\xi)$ is user-defined and it specifies the *prediction-error identification method* that is used to perform the model-identification. Many descriptions for this criterion has been proposed[51] where the most common and practical definition is to consider $\ell(\xi)$ of a quadratic norm as follows:

$$\ell(\xi) = \frac{1}{2}\xi^2 \tag{2.50}$$

Using this choice, the cost function can be defined as follows:

$$V(\boldsymbol{\psi}, Z^N) = \frac{1}{N}\sum_{t=1}^{N} \frac{1}{2}\left[y(t) - \varphi^T(t)\boldsymbol{\psi}\right]^2 \tag{2.51}$$

This last expression is referred to as the *least-squares method*. A key feature in this criterion is that the description for the function $\ell(\xi)$ is quadratic with respect to the linear model coefficients ψ and consequently (2.51) can be solved analytically as follows[51]:

$$\hat{\psi} = \arg\min V(\boldsymbol{\psi}, Z^N) = R(N)^{-1} f(N)$$
$$R(N) = \frac{1}{N}\sum_{t=1}^{N} \varphi(t)\varphi^T(t) \tag{2.52}$$
$$f(N) = \frac{1}{N}\sum_{t=1}^{N} \varphi(t) y(t)$$

where $\hat{\psi}$ represents the model coefficients' estimates. The expression $\arg\min$ means the minimizing argument, i.e. the value in the set $\boldsymbol{\psi}$ that minimizes the cost function. By inspecting (2.52), one can imply that the least-squares estimates $\hat{\psi}$ are obtained by the solution of a single algebraic equation that only requires information about the input/output

data arranged in the regression vector $\varphi(t)$. Let assume that the true physical system is defined as follows:

$$y(t) = \varphi^T(t)\psi_0 + e(t) \tag{2.53}$$

where ψ_0 represents the true values of the linear system's coefficients and $e(t)$ is defined as a set of independent random variables with mean zero values, i.e. $e(t)$ has white-noise properties. The error in the parameter estimates can be defined as follows:

$$\tilde{\psi} = \hat{\psi} - \psi_0 = R(N)^{-1}\sum_{t=1}^{N}\varphi(t)e(t) \tag{2.54}$$

Thus, to obtain an unbiased least-squares estimation, i.e. $\tilde{\psi} \to 0$, it is required that[51]:

$$E\tilde{\psi} = R(N)^{-1}\sum_{t=1}^{N}\varphi(t)Ee(t) = 0 \tag{2.55}$$

where the term E denotes a mathematical expectation. Based on the above definition, the covariance matrix of the error associated with the parameter estimates is defined as follows:

$$P_N = E\tilde{\psi}\tilde{\psi}^T = R(N)^{-1}v \tag{2.56}$$

where the function v is the covariance matrix of the measurement noise $e(t)$ and represents the noise variance associated with the data set Z^N. From the above expression it is clear that the errors in the parameter estimates only depend on the system's input properties $R(N)$ and the noise level v. The covariance matrix of R(N) is defined as follows:

$$\overline{R} = \lim_{N\to\infty}\frac{1}{N}R(N) \tag{2.57}$$

Given that $\overline{R}$ is not singular; the covariance matrix of the parameter estimates is approximately given by[51]:

$$P_N = \frac{v}{N}\overline{R}^{-1} \tag{2.58}$$

thus, $P_N \to 0$ as $N \to \infty$, and therefore $\hat{\psi}$ converge to ψ_0. The following conclusions can be drawn from (2.58):

- For $\hat{\psi} \to \psi_o$, $\overline{R}$ must be non-singular.
- The parameters approach to their true values at a rate $1/\sqrt{N}$.
- The covariance does not depend on input and noise signal's shapes, only on their corresponding variance properties.
- The covariance matrix is sensitive to the signal to noise ratio.
- For the least-squares to be consistent,

$$f* = E\varphi(t)e(t) = 0 \tag{2.59}$$

The condition in (2.59) is valid only if *e(t)* is white noise, i.e. $Ee(t) = 0$. This means that the actual values of the noise are independent from their past values; thus, the noise variance is represented as an identity matrix of proper dimensions.

From equation (2.56), it is desired to have $\overline{R}$ such that $\overline{R}^{-1} \to 0$; thus, a key factor that determines the quality in the parameter estimation is related to the input *u* used to generate the data set Z^N. Therefore, input-signal selection and design play a key role in systems identification. The input signals are selected based on factors such as the system's constraints, the identification method, and the class of model that is desired to obtain. A measure of the quality of the input signal is given by the crest factor function[51]:

$$C_r^2 = \frac{\max_t u^2(t)}{\lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} u^2(t)} \tag{2.60}$$

In principle, it is desired to have an input signal that have a small crest factor, which is achieved for binary, symmetric signals: $u(t) = \pm\overline{u}$. The smallest crest factor value is the

unity which corresponds to the maximum input power. In general, there are four test signals that are widely used in parameter estimation:

Filtered white noise.- This signal is generated by passing a white noise signal through a stable linear filter. A key advantage with this input is that it can be designed such that its power spectrum is bounded; i.e. it has specific frequency content. This is done by a proper design of the filter's parameters, i.e. the filter's time constant. On the other hand, this signal is expected to have large crest factors due to its asymmetry.

Random Binary Signal (RBN).- It is defined as a random signal that it can only take two extreme values. This signal's crest factor is thus ideal, *Cr=1,* at the expense of having a wider power spectrum range, i.e its frequency content may not be concentrated within the bandwidth relevant for the system to be identifed.

Multiple Sinusoids Signal.- The signal is described as a sum of sinusoids:

$$u(t) = \sum_{i=1}^{o} a_i \cos(\varpi_i t + \varsigma_i) \tag{2.61}$$

where o, a_i, ς_i and ϖ_i are adjustable parameters that can generate in principle a sinusoid signal with a very specific frequency content. As in the filtered white-noise case, the crest factor for this signal can be large. For specific amplitude, a_i, and assuming that all the sinusoids are in phase, the squared amplitude is (oa^2) and thus $C_r = \sqrt{2o}$; i.e. the crest factor is large when o is large.

Pseudo Random Binary Signal (PRBS).- It is a two state deterministic signal with white noise like properties. This signal is characterized by the maximum length sequence, Q, and its sample period or switching time (p). These parameters determines the input signal's period, i.e. $R = Qp$. To make good use of this signal's properties, the PRBNS has to be repeated R times. This is the reason why this signal is referred to as a *Pseudo* deterministic signal. The maximum length of the sequence (Q) is determined by factors such as the system's settling time. The PRBNS have approximately the same properties as white noise when it is used with a maximum length sequence. Such signals have maximum power for a

given amplitude and minimum crest factor, which are desirable properties for a candidate input signal to be used for identification. This signal is widely used to identify non-parametric models such as finite impulse response (FIR) models or step response models. Due to the form in which this signal is generated, the PRBNS contains equal power at all frequencies. Often, it is desired to design an input signal that excites the system in a particular frequency range. For example, for control purposes, it is often desirable to design controllers based on experiments around the crossover frequency. To circumvent this problem, PRBNS can be filtered by sampling faster, that is, given a PRBNS, a new signal can be constructed by taking *K* samples over each switching time π of the original signal. The new signal will thus be constant for at least *K* samples. This will increase the sampling frequency and reduce the original signal's high frequency content. Typical guidelines for choosing *K*, also known as the clock period, is to let *K* to be 2.5 times the bandwidth to be covered by the signal or to sample 10 times faster than the bandwidth of the system to be identified[51].

Since the data collected for identification is in discrete form an incorrect sample time can lead to design a controller that in principle provide stability and performance but when it is connected to the physical system can generate poor performance or even instability. Thus, the sampling time must be carefully chosen when designing an experiment for model identification. Although there is no systematic way to define the best sampling time for a given system, there exist guiding criteria to choose an appropriate sampling period[52]:

- Choose $T \approx \frac{\tau_{\min}}{3}$, where $\tau_{\min}$ is the system's smallest time constant of interest.

- Choose $f_s = 10 f_0$, where f_s is the sampling frequency and f_0 is the cut-off frequency of the system.

- Choose $\frac{t_{set}}{100} \leq T \leq \frac{t_{set}}{20}$, where t_{set} is the system's settling time.

System identification tools, and particularly the least-squares estimation, are central in this work. As it is shown in the next chapters, robust models, which are used to measure the

system's stability and performance, are obtained based on least-squares identification. Accordingly, the properties associated with the identification of a linear model using a quadratic norm criterion are widely exploited in this work to establish bounds on the predicted model's parameters.

In summary, this chapter has presented the methods, concepts and mathematical theory that are relevant for the current research work. In the first section, the current methodologies available to perform the integration of design and control were discussed and classified based on the key ideas that were used in the different approaches. Three main groups of methodologies were identified: controllability index-based approach, dynamic programming approach and robust approach. The second section presented a review on robust control theory. To design robust controllers, uncertain models are needed to measure robust stability and variability. Consequently the second section of this review chapter presented the different forms that uncertainty descriptions can be formulated and the form in which they are included in the models. Also, Lyapunov function methods were presented since they are central in robust control theory. In addition, two mathematical techniques for solving robust control problems were presented in this section: Linear Matrix Inequalities (LMI's) and Structured Singular Value (μ) analysis. The first technique is suitable for uncertain state space-based models while linear fractional uncertain models are best suitable for the μ analysis framework. The models used to robustly design the dynamic system can be obtained from closed-loop identification. Thus, the last section of this chapter have presented the different model representations, the basic algorithm for least-square identification, which is the most preferred method used for identification, the different types of input signals available for identification, and guidelines for an appropriate sampling time. The next chapter presents the basic methodology formulation that is proposed in this research work to simultaneously design and control dynamic systems.

3. Simultaneous Design and Control of Chemical Processes: A Quadratic Lyapunov Function Approach

The methodologies that are available for performing the integration of design and control have been based on a dynamic programming approach or on robust measures-based approach. The first set of methods searches for the disturbance and uncertainties realizations that generate the largest process variability by using the complete process nonlinear dynamic model. As it was stated in Section 2.2.2, the computational burden associated with the dynamic programming approach is extremely high, even with today's computer hardware which rules out its application to solve large problems. On the other hand, the latter methodologies estimate bounds on the process' stability and performance using a simpler representation of the process nonlinear dynamic behaviour. However, the reported robust approaches considered particular disturbance profiles and consequently, they do not guarantee that the resulting design is valid for other disturbance and uncertainties profiles.

This chapter presents the basic mathematical formulation that has been proposed to circumvent the limitations related to the previous methodologies. The key idea in this method is to represent the system's closed-loop nonlinear behaviour as an uncertain state space model where the model parameter values lie within ranges of values. The proposed methodology applies a robust stability test along with robust performance criteria based on a Quadratic Lyapunov Function (see Section 2.2.3). Thus, Linear Matrix Inequalities (LMI's) are included as constraints within the mathematical formulation. The new methodology was applied to the simultaneous design and control of a mixing tank process. This case study was also used to compare the computational burden required by the present method and that required by a dynamic programming approach.

This chapter is organized as follows: Section 3.1 presents the mathematical formulation proposed to integrate design and control on chemical processes. In Section 3.2, a case study, used to test the new methodology, is presented. The results obtained by this method as well as comparisons with another methodology are also discussed in this section. Section 3.3

presents a comparison, for the tank case study, of the computational times required by the present method and a methodology that uses dynamic optimization.

3.1 Conceptual Mathematical Formulation

This section describes in detail the methodology proposed to integrate process design and process control for systems under uncertainty and disturbances. The next paragraphs define the notation used throughout the chapter.

The variables used in the proposed method are classified as follows: 1-input variables (**h**), 2-output variables (**w**), 3-controller tuning parameters ($\boldsymbol{\lambda}$), 4-design variables (**d**) and 5-constants (**c**). The input variables **h** are partitioned as follows:

$$\mathbf{h} = [\boldsymbol{\kappa}, \boldsymbol{\iota}] \tag{3.1}$$

where $\boldsymbol{\kappa}$ represents the set of variables that can be adjusted to perform on-line control whereas $\boldsymbol{\iota}$ denotes the set of unmeasured perturbations that may affect the process during its normal operation. The set of variables $\boldsymbol{\kappa}$ can be divided as follows:

$$\boldsymbol{\kappa} = [\mathbf{u}, \mathbf{p}] \tag{3.2}$$

where **u** denotes the set of manipulated variables used by the control algorithm to reject the perturbations $\boldsymbol{\iota}$ whereas **p** represents the remaining set of parameters that are kept constant during plant operation and which values are obtained from optimization. To simplify the analysis, the unmeasured perturbation variables $\boldsymbol{\iota}$ are partitioned as follows:

$$\boldsymbol{\iota} = [\boldsymbol{\omega}, \boldsymbol{\upsilon}] \tag{3.3}$$

where $\boldsymbol{\omega}$ denotes a set of unmeasured perturbations that change very infrequently in the time domain, to be referred heretofore as *process parameter uncertainty*. This type of parameter uncertainty is considered to account for variables that remain at a particular steady-state for long periods of time. However, different steady-state values are considered for these variables in the optimization problem. Thus, the variations in time of these parameters are ignored in this analysis since they will occur very infrequently during the process normal

operation. For example, the production rate of a chemical unit may be kept constant for long periods of time but the plant have to be designed for different possible production rates. On the other hand, $\boldsymbol{\upsilon}$ represents the set of unmeasured perturbations whose values change rapidly in the time domain and are defined from heretofore as *disturbances*. The process output variables are classified as follows:

$$\mathbf{w} = \left[\mathbf{w}_{\mathbf{cl}}, \mathbf{w}_{\mathbf{ol}}\right] \tag{3.4}$$

where $\mathbf{w}_{\mathbf{cl}}$ refers to the set of output variables that are controlled in closed-loop by a controller whereas $\mathbf{w}_{\mathbf{ol}}$ are the remaining set of output variables that are not controlled. It is assumed that the variables $\boldsymbol{\kappa}$, $\boldsymbol{\iota}$, $\mathbf{w}$ and $\boldsymbol{\lambda}$ can take on values that fluctuate between a lower and an upper bound.

The simultaneous design and control problem can be conceptualized as follows:

$$\begin{array}{l} \text{Minimize Cost Function} \\ \text{s.t.} \\ \quad \text{Process Model Equations,} \\ \quad \text{Control Algorithm Equations,} \\ \quad \text{Process Stability Constraint,} \\ \quad \text{Process Output Variability Constraint,} \\ \quad \text{Process Feasibility Constraint.} \end{array} \tag{3.5}$$

The constraints in (3.5) have to be satisfied for the process entire period of operation and for any possible realization of the disturbances $\boldsymbol{\upsilon}$ and the process parameter uncertainty $\boldsymbol{\omega}$, respectively.

3.1.1 Cost Function

The cost function (CF) is formulated as the combination of the process capital and operating costs that are calculated based on steady state information, and a variability cost, which takes into account the dynamic behaviour of the process to be designed. This is mathematically expressed as:

$$CF = CC\left(\overline{\mathbf{u}}, \mathbf{p}, \overline{\mathbf{w}}, \mathbf{c}, \mathbf{d}, \phi^{d}\right) + OP\left(\overline{\mathbf{u}}, \mathbf{p}, \overline{\mathbf{w}}, \mathbf{c}, \mathbf{d}, \phi^{d}\right) + VC\left(\overline{\mathbf{u}}, \mathbf{p}, \overline{\mathbf{w}}, \mathbf{c}, \mathbf{d}, \boldsymbol{\lambda}, \phi^{d}\right) \tag{3.6}$$

where the annualized capital cost (CC), the operating cost (OP) and the variability cost (VC) are functions of the nominal steady-state values of the available manipulated variables ($\bar{\mathbf{u}}$) and output variables ($\bar{\mathbf{w}}$), the value of the remain set of parameters that are not used for control (**p**), the design variables (**d**), and a process variability function (ϕ^d), respectively. In addition, the variability cost term is also a function of the controller tuning parameters ($\boldsymbol{\lambda}$).

The plant's capital and operating costs are usually estimated from cost correlations which depend on the design variables of the system. In many chemical processes, the design variables are usually expressed as a function of other process variables, e.g. process manipulated variables, which are continuously changing in time. Therefore, the design variables not only depend on the nominal values of those variables but also on their variability with respect to their specified nominal value, i.e. the design variables also depend on the process dynamics. Consequently, the capital and operating costs functions are also functions of the process variability (ϕ^{d}) as shown in (3.6).

For example, when the design of a distillation column is considered, the number of trays in the column is related to the reflux ratio. Often, the reflux ratio is manipulated for control purposes, i.e. its value changes with time. Thus, the total number of trays will be estimated based on the nominal reflux ratio obtained from steady-state calculations, plus the variability in reflux with respect to the nominal value.

To assess the variability cost, it is necessary to first measure the process variability and to assign a cost to it. The process variability function (ϕ^{d}) is process specific and depends on factors such as the goals to be attained by the design, the process inputs and outputs and the nature of the process itself. The assignment of a specific economic value to the process variability function also depends on such factors. For example, if the goal is to design a process to keep the property of a product on target, then the cost is related to the deviations in this property with respect to the target and the corresponding quality degradation or quality improvement. On the other hand, if the sole objective is to design a system that can successfully accommodate the effect of magnitude-bounded disturbances entering the process, then the variability cost may be directly associated with the size of the unit to be designed. For example, the capital cost of a storage tank will be a function of the design

variable at a nominal value, i.e. the steady-state volume, plus a measure of the variations of this variable with respect to the nominal value, e.g. the changes in volume, occurring due to imperfect control.

3.1.2 Process Model Equations and Control Algorithm

It is assumed in this work that a closed-loop nonlinear dynamic process model is available for simulations. For both case studies, full nonlinear simulation models based on mass and energy balances were available. Similarly, the control structure used to control the system is also assumed to be known a priori.

To develop the simultaneous flexibility and controllability analysis, the dynamic optimization-based methodologies have used the complete nonlinear dynamic model of the system to represent the process behaviour. This requires the numerical solution of a dynamic optimization problem, which may involve, even for chemical processes with a small number of units, rigorous mathematical calculations and an extensive computational time. This fact represents a limitation towards the application of these methodologies to the simultaneous design and control of chemical processes composed of several process units.

A key idea in this work is to represent the closed-loop nonlinear dynamic model of the process as a nominal closed-loop state space model complemented with uncertainty in the model parameters. This implies that each of the elements of the state space matrices is bounded between extreme values. The set of uncertain values are defined as model parameter uncertainty (θ). This type of uncertainty should be distinguished from the process parameter uncertainty ($\boldsymbol{\omega}$) defined above that is related to uncertainty in the process physical parameters. The uncertainty values in the model parameters are used to capture the differences between the nominal linear state space model and the actual process behaviour and they are consequently related to the process nonlinearities due to the changes in the disturbances $\boldsymbol{\upsilon}$. Thus, the robust model is mathematically represented as follows:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\theta_A)\mathbf{x} + \mathbf{B}(\theta_B)\boldsymbol{\upsilon} \\ \mathbf{y} &= \mathbf{C}(\theta_C)\mathbf{x} + \mathbf{D}(\theta_D)\boldsymbol{\upsilon} \end{aligned} \tag{3.7}$$

where the matrix $\mathbf{A}(\theta_A)$ has the following structure:

$$A(\theta_A) = \begin{bmatrix} \overline{a}_{11} \pm \theta_{A(1,1)} & \cdots & \overline{a}_{1m} \pm \theta_{A(1,m)} \\ \vdots & \overline{a}_{ij} \pm \theta_{A(i,j)} & \vdots \\ \overline{a}_{n1} \pm \theta_{A(n,1)} & \cdots & \overline{a}_{nm} \pm \theta_{A(n,m)} \end{bmatrix} \tag{3.8}$$

The remaining state space matrices defined in (3.7) are posed as in (3.8). The uncertain state space model given in (3.7) corresponds to a continuous affine parameter-dependent polytope type model. The formal polytope model description has been given in equation (2.11). The uncertain model parameters are defined as in (2.1), that is:

$$a_{ij} = \left[\overline{a}_{ij} + \theta_{A(i,j)}, \overline{a}_{ij} - \theta_{A(i,j)}\right] \tag{3.9}$$

where it has been assumed that the amplitude of the uncertainty region associated with each element of the state space matrices is symmetric, i.e.

$$\left|\theta_{A(i,j)}\right| = \min\left|\theta_{A(i,j)}\right| = \max\left|\theta_{A(i,j)}\right| \tag{3.10}$$

The model coefficient description shown in (3.9) corresponds to a parameter that ranges between two extreme values. The input in (3.7) is the magnitude-bounded disturbance $\boldsymbol{\upsilon}$ that affects the process during its normal operation; the model output ($\mathbf{y}$) can be either a closed-loop process output variable ($\mathbf{w_{cl}}$) or an open-loop process output variable ($\mathbf{w_{ol}}$).

This type of model, also referred heretofore as *robust model*, can be obtained from identification of the closed-loop process nonlinear dynamic model as follows:

i) Based on the process open-loop time constant estimated from open-loop simulations of the process, an excitation signal is designed to simulate the process closed loop nonlinear transient behaviour. The input to this model is the disturbance $\boldsymbol{\upsilon}$ that affects the process outputs $\mathbf{y}$. Thus, an excitation signal is imposed on the input using the actual lower and upper bounds specified for this variable, i.e., $\boldsymbol{\upsilon}^{\mathbf{l}}$ and $\boldsymbol{\upsilon}^{\mathbf{u}}$.

ii) The process is simulated under closed-loop. The input/output data collected from simulation is used to fit a continuous linear transfer function model applying the least-squares method that has been explained in Section 2.3.

iii) The model is initially identified in transfer function form (see equation 2.42) and then is transformed into a state space canonical form by using the transformation of transfer function to a canonical state space model that has been explained in Section 2.3 and given by equation (2.44). The resulting linear model is complemented with model parameter uncertainty to capture the process nonlinearities. The information obtained from the covariance matrix, estimated from least-squares identification, is used as an approximation to describe the uncertainty region associated with each model parameter.

Although the above procedure applies a least-squares method to generate a robust state space model, the present methodology is not restricted to this identification criterion since there are other methods available for closed-loop identification of continuous state space models[51].

It should be noticed that the only input used to generate the data set for identification is the disturbance set ($\boldsymbol{\upsilon}$). Thus, the robust model specified in (3.7) is only valid within a small neighbourhood of the nominal operating state, specified by nominal values in the manipulated variables to be used by the control algorithms and the process outputs, $\overline{\mathbf{u}}$ and $\overline{\mathbf{w}}$, and for fixed values in the adjustable variables ($\mathbf{p}$), the design variables ($\mathbf{d}$), the controller tuning parameters ($\boldsymbol{\lambda}$) and the process parameter uncertainty ($\boldsymbol{\omega}$).

The uncertain model description given above can be used to test robustness in the system to be designed. As it has been shown in Section 2.2.2, uncertain state space models like (3.7) are suitable for the LMI's framework; therefore, robust measures based on a Quadratic Lyapunov Function (QLF) can be developed to estimate bounds on the process asymptotic stability margins and output variability. This approach is expected to reduce the computational time since long and expensive simulations of the full nonlinear model required by dynamic programming approaches are avoided.

In addition, robust models like (3.7) can also be generated to describe the dynamic behaviour of the manipulated variables ($\mathbf{u}$) with respect to changes in the disturbance variables ($\boldsymbol{\upsilon}$).

Then, using these models that have **u** as an output, bounds over the variability in **u** can be estimated with this model to test constraints on the manipulated variables used by the control algorithms.

3.1.3 Process Stability

To ensure process asymptotic stability, this approach applies a robust stability criterion based on a Quadratic Lyapunov Function, **V**(**x**), which has been defined in Section 2.2.3.

Given a QLF: $\mathbf{V}(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{P}\mathbf{x}$, with **x** being the robust model's states, the uncertain state space model described in (3.7) is asymptotically stable if and only if the following condition is satisfied[42]:

$$\mathbf{A}(\theta_A)^T\mathbf{P} + \mathbf{P}\mathbf{A}(\theta_A) < \mathbf{0} \tag{3.11}$$

where **P** is a symmetric positive definite Lyapunov matrix and $\mathbf{A}(\theta_{\mathrm{A}})$ is the **A** state space matrix given in (3.7), which uncertain parameters (θ_{A}) are bounded between pre-specified values. The objective in this problem is to find a matrix **P** such that satisifies the above inequality for all the possible combination in uncertain state space matrix $\mathbf{A}(\theta_{\mathrm{A}})$. Inequality (3.11) cannot be solved directly since it places an infinite number of inequalities corresponding to all the possible values of the uncertain model parameters. However, due to the convexity of the condition 3.11 with respect to the model parameter uncertainty terms, this condition can be satisfied for all possible parameter values by testing only, the extreme values of each of the elements of the **A** state space matrix**.** These extreme values are used to form a polytope (see equation 2.11), usually referred to as the *uncertain parameter box*. Each vertex of the uncertain parameter box is specified by a particular combination of the extreme values of the parameters θ_{A} of the matrix **A**. Therefore, the stability test can be reduced to a finite set of Linear Matrix Inequalities (LMI's) defined as follows:

$$\begin{aligned} &\mathbf{A}(m_k)^T\mathbf{P} + \mathbf{P}\mathbf{A}(m_k) < \mathbf{0} \text{ for all } m_k \in \mathbf{m} \\ &\mathbf{m} = \left\{(m_1, \ldots, m_k, \ldots m_r) : m_k \in \left\{a_{ij} + \theta_{A(i,j)}, a_{ij} - \theta_{A(i,j)}\right\}\right\} \end{aligned} \tag{3.12}$$

where **m** denotes the set of 2^r vertexes specified by the extreme values of the **A** state space matrix uncertain elements. The set of LMI's given in (3.12) is used in the present methodology to test the system's asymptotic stability. This implies that the resulting design obtained by this approach is guarantee to be asymptotically stable despite any bounded realization in **ω** and **υ**.

3.1.4 Process Output Variability

One of the key problems faced when integrating design and control is the assessment of the critical time-dependent profiles in the perturbation variables that produces the largest variability in the process output. This condition, referred to as the worst-case variability scenario, has been estimated by the most recently reported methodologies from the full numerical solution of dynamic optimization problems which is computationally intensive. Instead, in the present approach, a QLF is used together with the robust model specified in (3.7) to compute a norm-based bound on the process output error's variance.

Assume that there exists a QLF: $\mathbf{V}(\mathbf{x}) = \mathbf{x}^T\mathbf{P}\mathbf{x}$ with $\mathbf{P} > \mathbf{0}, \mathbf{P} = \mathbf{P}^T$ and a parameter $\gamma \geq 0$ such that:

$$\mathbf{V}(\mathbf{x}) \geq \mathbf{0} \text{ for all } \mathbf{x},\ \mathbf{V}(0) = 0 \tag{3.13}$$

$$\dot{\mathbf{V}}(\mathbf{x}) + \mathbf{y}(t)^T\mathbf{y}(t) - \gamma^2\boldsymbol{\upsilon}(t)^T\boldsymbol{\upsilon}(t) < 0 \text{ for all } y, \upsilon \tag{3.14}$$

Inequality (3.13) has to be satisfied for all cases for which there is no external perturbations (**υ**=0), i.e. the free response of the system, whereas inequality (3.14) has to be satisfied for all time *t* along any nonzero closed-loop trajectory of the system (3.7). Integrating (3.14) from 0 to t_f in combination with the fact that inequality (3.13) is satisfied when the disturbances are removed yields:

$$\int_0^{t_f} \mathbf{y}(t)^T\mathbf{y}(t)dt - \gamma^2\int_0^{t_f} \boldsymbol{\upsilon}(t)^T\boldsymbol{\upsilon}(t)dt < \mathbf{0} \tag{3.15}$$

After taking the limit $t_f \to \infty$ inequality (3.15) is rewritten as follows:

$$\gamma > \frac{\|\mathbf{y}\|_2}{\|\boldsymbol{\upsilon}\|_2} \tag{3.16}$$

where $\|\mathbf{y}\|$ and $\|\boldsymbol{\upsilon}\|$ represent the Euclidean norm of the model output errors and the disturbances affecting the process, respectively. The result obtained in (3.16) implies that γ is an upper bound on the 2-norm gain of the model output in the presence of external disturbances and consequently, this norm can be used to measure the system's closed loop variability. In order to compute γ, inequality (3.14) can be rewritten in a matrix quadratic form by expanding the quadratic function term $\mathbf{V}(\mathbf{x})$ and using the robust model given in (3.7). Thus, it can be shown that (3.14) is equivalent to the following set of linear matrix inequalities[36]:

$$\begin{gathered}\boldsymbol{\psi}^T \begin{bmatrix} \mathbf{A}(\theta_A)^T\mathbf{P} + \mathbf{A}(\theta_A)\mathbf{P} & \mathbf{B}(\theta_B)\mathbf{P} & \mathbf{C}(\theta_C)^T \\ \mathbf{B}(\theta_B)^T\mathbf{P} & -\gamma^2 I & \mathbf{D}(\theta_D)^T \\ \mathbf{C}(\theta_C) & \mathbf{D}(\theta_D) & -\mathbf{I} \end{bmatrix} \boldsymbol{\psi} < \mathbf{0} \\ \boldsymbol{\psi}^T = \begin{bmatrix} \mathbf{x}^T & \boldsymbol{\upsilon}^T & \left(\mathbf{C}(\theta_C)\mathbf{x} + \mathbf{D}(\theta_D)\boldsymbol{\upsilon}\right)^T \end{bmatrix}\end{gathered} \tag{3.17}$$

Similarly to the stability test described before, inequality (3.17) involves an infinite set of linear matrix inequalities since each element of the state space matrices is given by an infinite set of values bounded by a lower and an upper bound. To circumvent this condition, the same approach used in the process stability analysis is also applied in the computation of γ, i.e., an uncertain parameter box is created using only the extreme values of the uncertain model parameters of the state space matrices specified in (3.7) and then inequality (3.17) is reduced to the following set of LMI's:

$$\begin{gathered}\begin{bmatrix} \mathbf{A}(s_l)^T\mathbf{P} + \mathbf{A}(s_l)\mathbf{P} & \mathbf{B}(s_l)\mathbf{P} & \mathbf{C}(s_l)^T \\ \mathbf{B}(s_l)^T\mathbf{P} & -\gamma^2 I & \mathbf{D}(s_l)^T \\ \mathbf{C}(s_l) & \mathbf{D}(s_l) & -\mathbf{I} \end{bmatrix} < \mathbf{0} \text{ for all } s_l \in \mathbf{s} \\ \mathbf{s} := \left\{ \left(s_1, \ldots, s_l, \ldots, s_q\right) : s_l \in \left\{ e_{ij} + \theta_{(ij)}, e_{ij} - \theta_{(ij)} \right\} \right\}\end{gathered} \tag{3.18}$$

where $\mathbf{s}$ is the set of the 2^q vertices of the uncertain parameter box that represents all the possible combinations between each of the upper and lower bounds of the elements of the

state matrices given in (3.7). Accordingly, inequality (3.17) is reduced to a set of 2^q linear matrix inequalities. To compute γ, it is required to find the smallest value in this parameter such that (3.18) holds with **P** being a symmetric positive definite matrix. This problem is formulated as follows:

$$\begin{aligned} &\min_{\mathbf{P}} \gamma^2 \\ &s.t. \\ &\begin{bmatrix} \mathbf{A}(s_l)^T\mathbf{P} + \mathbf{A}(s_l)\mathbf{P} & \mathbf{B}(s_l)\mathbf{P} & \mathbf{C}(s_l)^T \\ \mathbf{B}(s_l)^T\mathbf{P} & I & \mathbf{D}(s_l)^T \\ \mathbf{C}(s_l) & \mathbf{D}(s_l) & -\mathbf{I} \end{bmatrix} < \gamma^2 \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \\ &\mathbf{P} > \mathbf{0}, \mathbf{P}^T = \mathbf{P} \end{aligned} \tag{3.19}$$

Problem (3.19) corresponds to a generalized eigenvalue problem (GEVP) or it can also be reformulated as a eigenvalue problem (EVP) where the formal definitions of GEVP and EVP have been given in Section 2.2.2.The solution in (3.19) provide the value of the parameter γ which is an upper bound on the largest input/output root-mean squares (RMS) gain of the robust model given in (3.7). The computation of γ from (3.19) can be easily performed using the current available off-the-shelf LMI-solvers such as the MATLAB's LMI Toolbox [34].

Inequality (3.16) can be expanded to obtain the following formulation:

$$\gamma^2 > \frac{\sum_{i=0}^{N} y_i^2}{t_f \delta \upsilon^2} \tag{3.20}$$

where $\delta\upsilon$ denotes the rate of change on $\boldsymbol{\upsilon}$ and t_f represents a time horizon ($t_f \to \infty$). Inequality (3.20) specifies that γ *provides a bound on the standard deviation of the model output variability*. However, the worst-case scenario calculation requires the specification of a bound over the maximum variability in the model output (**y**). For a normal distribution, one standard deviation indicates that approximately 68.26% of the data lies within this value, i.e. the process output values will remain within the value specified by γ at least 68.26% of the time horizon, t_f. Base on this assumption, higher multiple values of γ may be used to estimate a more realistic bound on the maximum model output error, that is,

$$\mathrm{H} = n\gamma \tag{3.21}$$

where H represents a bound on the process output variability based on the multiples of the parameter γ to be used in the design and n is an integer number referred to as the number of multiples of the variability index to be used within the analysis. The integer parameter n is specified by the user to guarantee that a certain percentage of the output values will lie within the bounds given by H.

The previous analysis specifies a bound on the worst process output error based on the robust model (3.7), which in turn was generated around a nominal operating point and at a fixed value of the process parameter uncertainty **ω**. Thus, one must search for the value in **ω** that drives the model output variable **y** to its largest variation. Accordingly, the QLF-based worst-case scenario formulation is defined as follows:

$$\max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega} \le \boldsymbol{\omega}^u} \phi^d\left(\overline{\mathbf{u}}, \mathbf{p}, \overline{\mathbf{w}}, \mathbf{c}, \mathbf{d}, \boldsymbol{\lambda}, \mathrm{H}^d\right) \tag{3.22}$$

Where H^d denotes an upper bound over the process output variability and is directly related to the process variability ϕ^d. The function ϕ^d is problem specific and it also depends, as indicated in equation 3.22 and as illustrated later in the case study, on the manipulated variables **u**, the adjustable variables **p**, the process outputs **w**, the constant parameters, **c**, the design variables, **d**, and the controller tuning parameters **λ**. To solve (3.22), identification of a closed-loop robust model of the structure given by (3.7) is performed around different process parameter uncertainty values (**ω**) while the remaining process variables (**u**, **p**, **w**, **c**, **d,** and **λ**) are fixed to specific values. Problem (3.22) also requires the computation of (3.21) and (3.19) for each **ω** tested but this calculation is very simple and can be performed rapidly.

The optimization problem posed in (3.22) could be reduced to a single calculation of ϕ^d if the process parameter uncertainty were also considered as an input in the robust model (3.7); i.e., **υ** and **ω** would be considered as the external inputs that generate the process dynamic simulation outputs that are used to perform the closed-loop identification. This approach would take into account the transients in **ω** and would also eliminate the necessity to compute (3.22), but it would potentially lead to the identification of a robust model with

larger uncertainty, that could potentially cause the final design to be more conservative. Thus, although solving for (3.22) requires the systems identification at every **ω** tested it involves a simpler identification problem as compared with the approach that considers **ω** as a disturbance. Moreover the uncertain model parameters of the identified model are expected to lie within a small neighbourhood of the nominal point around which the identification is done due to closed-loop operation.

Based on the knowledge of the maximum process variability, a cost can be assigned to quantify the economic impact of the process variability in the cost function specified in (3.6) as illustrated later in the case study.

3.1.5 Process Feasibility

Process feasibility requires that all the time dependent process variables, e.g. the manipulated variables **u,** must remain within pre-specified bounds in the presence of changes in the perturbation variables, i.e. **υ** and **ω**. To guarantee process feasibility, the present work applies the same approach used to assess the worst-case scenario, i.e. an upper bound on the largest input/output RMS gain is computed based on the closed-loop identification of a robust model with the structure (3.7). The output of this model is the variable that it is desired to keep in between bounds, e.g. the manipulated variables should be kept in between the saturation limits of the actuator. In this case, the model has to be identified from the disturbance variables **υ** to any process variable (g) that is required to remain within bounds. Then, the largest variability in this variable is estimated by solving (3.19) and (3.21) respectively. Thus, the feasibility condition for each process variable is given as follows:

$$\bar{g} \pm \max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}' \le \boldsymbol{\omega}^u} \mathrm{H}^g \le \rho \tag{3.23}$$

where ρ are the input limits and $\bar{g}$ represents the nominal value of the process variable that has to remain within the specified bounds, e.g., the nominal value of the manipulated variable ($\bar{\mathbf{u}}$) used by the controller or the set point of an output variable ($\bar{\mathbf{w}}_{cl}$). It should be noticed that the process parameter uncertainty value **ω** that produces the maximum variability in (3.22) may not be same as in (3.23). Hence, **ω'** in inequality 3.23 represents the process

parameter uncertainty that produces the maximum variation in g which values ranges between a lower and an upper bound, denoted as $\boldsymbol{\omega}^l$ and $\boldsymbol{\omega}^u$, respectively. In addition, the parameter H^g denotes a bound on the variability in g due to changes in $\boldsymbol{\upsilon}$ and is computed from (3.21).

3.1.6 Optimization Problem and Solution Strategy

The cost function given in (3.6), the process stability test (3.12), the worst-case scenario calculation (3.22) and the process feasibility constraint (3.23) are combined together in the present approach to achieve the simultaneous design and control of dynamic systems under uncertainty and disturbances. Therefore, the conceptual simultaneous design and control optimization problem outlined in (3.5) is mathematically expressed as follows:

$$
\begin{aligned}
&\min_{\boldsymbol{\eta}=[\overline{\mathbf{u}},\mathbf{p},\overline{\mathbf{w}}_{cl},\overline{\mathbf{w}}_{ol},\mathbf{d},\boldsymbol{\lambda}]} CF = CC + OP + VC \qquad \ldots \text{Cost Function} \\
&\text{s.t.} \\
&\quad \mathbf{A}(m_k)^T\mathbf{P} + \mathbf{P}\mathbf{A}(m_k) < \mathbf{0} \qquad \ldots \text{Process stability test} \\
&\quad \max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega} \le \boldsymbol{\omega}^u} \phi^d\left(\overline{\mathbf{u}}, \mathbf{p}, \overline{\mathbf{w}}, \mathbf{c}, \mathbf{d}, \boldsymbol{\lambda}, \mathrm{H}^d\right) \quad \ldots \text{Process output variability} \\
&\quad \overline{g} \pm \max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}' \le \boldsymbol{\omega}^u} \mathrm{H}^g \le \rho \qquad \ldots \text{Process feasibility} \\
&\quad \boldsymbol{\eta}^l \le \boldsymbol{\eta} \le \boldsymbol{\eta}^{\mathrm{u}} \\
&\quad \text{Process Model Equations} \\
&\quad \text{Control Algorithm Equations}
\end{aligned}
\tag{3.24}
$$

Although the process model equations and the control algorithms equations do not explicitly appear within the mathematical formulation, they are implicitly considered through the use of the identified robust model. For example, the robust stability criterion is based on the robust model matrix $\mathbf{A}(m_k)$, whereas the process output variability and process feasibility bounds, H^d, and H^g respectively, are obtained by using the robust model as explained above.

The resulting optimization problem stated in (3.24) corresponds to a nonlinear constrained optimization problem. The algorithm proposed to solve this problem can be summarized in the following steps:

Step 0 (Initialization).- Given the process flowsheet, the control algorithm and the pairing between the manipulated variables and the controlled variables, specify initial nominal values for the manipulated variables to be used by the controller ($\overline{\mathbf{u}}$), the output variables that are not being controlled ($\overline{\mathbf{w}}_{ol}$) and for the set point of the output variables that are under control ($\overline{\mathbf{w}}_{cl}$). Also, specify initial values for the controller tuning parameters ($\boldsymbol{\lambda}$) and the remaining set of adjustable variables that are kept at a constant value during the entire period of operation (**p**) and the design variables (**d**). For simplicity, these variables are lumped into the vector $\boldsymbol{\eta}$ and lower and upper bounds are assumed for each of the variables in this vector , i.e., $\boldsymbol{\eta}^l$ and $\boldsymbol{\eta}^u$, respectively. Also, an initial value for the process parameter uncertainty $\boldsymbol{\omega}$ is specified.

At each iteration k:

Identification of a state space model with uncertain model coefficients.- In order to solve problem (3.24), the complete process nonlinear model and the selected control structure are used to identify a robust state space model around a nominal operating condition, specified by the values in $\boldsymbol{\eta}$ and $\boldsymbol{\omega}$, respectively. The input data used to perform the identification process consists of a Pseudo Random Binary Noise Signal (PRBNS) on the disturbance variables ($\boldsymbol{\upsilon}$) which is designed based on the process dominant time constant and the lower and upper bounds specified for this variable. The required properties of the input excitation signal have been specified in Section 2.3. Then, this signal is used to simulate the closed-loop behaviour of the system to be designed. The input/output data collected from this simulation are used to identify a linear transfer function model using the least squares criterion[51] and then, a state space realization is performed on the identified transfer function to obtain a canonical linear state space model. The process nonlinear behaviour in the presence of perturbations $\boldsymbol{\upsilon}$ is captured by using uncertainty terms (θ) with respect to the linear state space nominal model parameters as it is shown in (3.8). The values of θ are obtained from the covariance matrix generated during the linear parameter estimation of the linear transfer function model (see Section 2.3). The application of this procedure generates a robust model such of the structure given in (3.7).

Model Validation.-The use of the elements of the covariance matrix as bounds of the parameter estimates of the nominal state space model is a practical approximation based on the assumption of normal distribution of parameter values. On the other hand, since the residuals between the closed-loop nonlinear system and the corresponding identified linear model are deterministic, the parameters may not be normally distributed. Asymptotic normality is obtained for infinite number of data points since the input signal used for identification is a PRBNS with white noise-like properties[51]. In the current study, although the number of points for identification is large, it is not infinite. Therefore, the parameters may not be normally distributed and the variance may not be an accurate bound. Thus, to test that the identified robust state space model captures the systems closed-loop dynamics and it is sufficiently inclusive to calculate a bound on the variability index, γ, a model validation step may be performed. For each closed-loop identification, this step consists in comparing the index γ calculated from the simulation of the nonlinear system using (3.16) to the analytical bound of γ calculated from (3.19) and (3.21). Since the calculation of γ considers models that are made of all the possible combinations of the uncertain model parameter values, θ, considered in the construction of the robust model given by (3.7), it was expected that the resulting analytical bound is sufficient to bound the γ obtained from the simulation of the nonlinear system. This was fully corroborated in the case study presented in Section 3.2.

If the test is satisfied, the optimization proceeds to the next step. Otherwise, one can increase the uncertainty bounds on each of the robust model parameters by multiplying the covariance matrix estimated by a factor until this test is satisfied. However, this never occurred in the case studies presented in this work.

Step 1 (Worst-case scenario).- Given the process variability function ϕ^d and the nominal operating conditions specified by $\boldsymbol{\eta}$, find the critical value in the process parameter uncertainty $\boldsymbol{\omega}$ that produces the maximum variability in the process (3.22). At this stage, a different robust model such as (3.7) has to be identified for each $\boldsymbol{\omega}$ tested. Once problem (3.22) is solved, a cost can be assigned to the process variability to evaluate the impact of the process variability on the cost function (3.6).

Step 2 (Process stability test).- Given the critical value in $\boldsymbol{\omega}$ obtained from the previous step and with the nominal operating conditions fixed ($\boldsymbol{\eta}$), compute inequality (3.12) to evaluate the process asymptotic closed-loop stability.

Step 3 (Process Feasibility).- Compute problem (3.23) to verify that the process variables of the system to be designed are within the pre-specified bounds. At this stage, the identification of the robust model such as (3.7) has to be generated for each $\boldsymbol{\omega}'$ tested.

Step 4 (Objective function evaluation).- Using the information given in the previous steps, calculate the capital, operating and variability cost of the system to be designed (3.6). If the process constraints specified in (3.24) are satisfied and the difference in the cost function value between the previous and the actual iteration lies within a specific tolerance, then STOP, and optimal solution has been found; otherwise, set $k=k+1$, redefine the values in $\boldsymbol{\eta}$ according to the selected optimization algorithm and return to step 1.

Although the present methodology attempts to optimally design chemical processes by integrating process design and control, there are a series of issues considered in its development that require further clarification. Problem (3.24) assumes that the process flowsheet and the control structure are defined a priori; thus, the process and control synthesis problem are not considered within the mathematical formulation presented above. However, the present approach can be expanded by embedding problem (3.24) into an optimization problem that searches for an optimal control structure and optimal flow sheet by adding integer decisions within the formulation. This would obviously increase the design degrees of freedom at the expense of solving a more complex optimization problem and the optimization problem will then be formulated as a mixed-integer nonlinear programme (MINLP).

As mentioned above, it is assumed that the process variables in the system to be designed follow a normal distribution function. Hence, the value of the integer parameter n in (3.21) can be chosen to capture certain percentage of the variability. Although a normal distribution was assumed, the present methodology is not restricted to this choice and other types of distributions can be used for the assessment of the worst-case scenario and the process feasibility requirement. Regarding the selection of the value for n in (3.21), the parameter γ is

computed based on a state space model with uncertainty in the model parameters. Consequently, the value of γ tends to be conservative and so does the design. As it is shown in chapter 4, a robustness test based on a Singular Structure Value analysis (μ) has been developed to estimate a 1-norm bound on the maximum model output variability, i.e., $\max|y(t)|$.

Furthermore, it is assumed in this approach that the process parameter uncertainty and the disturbance have a uniform probability, i.e., the probability of occurrence of a given value is the same as for all the values considered within the specified bounds. This assumption is made to ensure that, for the process entire period of operation, the final design can successfully reject any time-dependent magnitude bounded perturbations $\boldsymbol{\upsilon}$ when the process is operated around different steady-state values of $\boldsymbol{\omega}$ without exceeding the process constraints specified in (3.24).

The optimization problems solved in the present work were based on decentralized PID control strategies. However, the methodology presented above is not restricted to this type of control strategy and it could be extended for use with model based controllers. If model based controllers are used as the control algorithm, then the complete open-loop process model needs to be identified first and represented as a linear transfer function model. The identification may be performed as per the following steps:

i) Based on the time constant of the open-loop process, a PRBNS is designed for the disturbance which is assumed to be the input into this process. This signal is used to simulate the process open-loop behaviour. The data recorded from this simulation is used to identify a linear transfer function model using a suitable prediction error method[51], e.g. the least-squares criterion.

ii) The identified transfer function model could then be used to either estimate the controller tuning parameters, by using the internal model control method (IMC), or as the internal process model for model predictive control algorithm (MPC).

After the closed-loop model with the model-based controller is identified, the optimization problem is formulated as per the same steps presented above. The calculation of the

variability index γ for a model predictive controller (MPC) has been shown in a work related to the design of gain-scheduled controllers[53]. However, using model-based controllers would be generally more tedious because it requires the identification of the process open-loop model for use as the internal model of the model-based algorithm.

3.2 Case Study: mixing tank Process

The methodology presented in the previous section was applied to the simultaneous design and control of a mixing tank process that was previously studied by Mohideen et al.[16]. This problem was selected as a case study because its dynamics are described by two relatively simple nonlinear equations. However, the integration of design and control of this system has been still identified as a challenging task for dynamic programming-based methods since a bounded disturbance and a process parameter uncertainty is considered within the system's design. Thus, the resulting design parameters are required to keep variables at their set points despite any possible combination of the process parameter uncertainty and the disturbance while maintaining the process operation within the feasible region.

Figure 3.1 presents the mixing tank process flowsheet. The system consists of a hot and a cold stream mixed in a stirred tank. The hot stream flow rate, F_h, and temperature, T_h, are time-varying variables due to variations in the hot feed flow rate, F_h and the hot stream temperature T_h. The flow rate F_h, is assumed, based on the problem presented in the literature, to change very infrequently in time and therefore it is considered in this work as an unmeasured perturbation that remains at a steady state value for long periods of time. Then, as per the definitions given in the previous sections, F_h is assumed to be a process parameter uncertainty (**ω**). On the other hand, the temperature of the hot stream, T_h, is assumed to be an unmeasured perturbation that exhibits fast-varying changes between extreme values; consequently, T_h is considered in the design to be an external time varying disturbance (**υ**). The temperature in the cold stream, T_c, is a constant (**c**) during the entire period of operation. The cold stream flow rate, F_c, and the stem position of the valve located at the outlet, z, are the candidates manipulated variables for this process (**κ**). These variables can be set either to a constant value for the entire period of operation and they will be treated as a design variable **p** or they will be determined by the controller and they will be treated accordingly as

a manipulated variable **u**. The mixing tank's hold up, $V(t)$, and the outlet temperature, $T(t)$, are state variables (**w**) that describe the process behaviour at any time t and it is assumed that they can be measured on-line. The tank's outlet flow rate is assumed to be a nonlinear function of mixing tank's hold up, i.e. $F(V) = z\sqrt{V}$. In addition, the tank is assumed to be well stirred and the density of the fluid is assumed constant for the entire period of operation. Table 3.1 shows the process model and the ranges of values considered for the different variables defining the process.

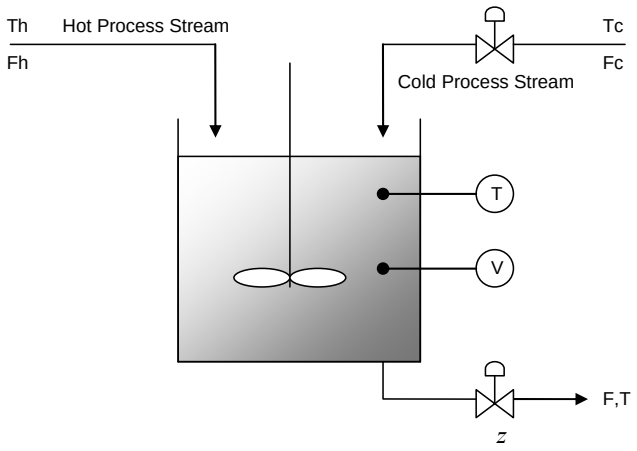

Figure 3.1 mixing tank process

The objective is to minimize the tank's cost such that, the values of the controller tuning parameters (**λ**) and the available manipulated variables (F_c and z) can ensure process stability and feasibility for any time evolutions in the process hot stream (F_h and T_h) at any time t. The tank's total cost is directly proportional to the tank's final size or volume. For this case study, the capital cost is assumed to be equal to the total volume of the tank at steady state plus a variability in volume around that steady state, i.e. the variability in volume accounts for the tank's hold up variations due to the effect of disturbances entering the process and the imperfect rejection of these disturbances by the controller. The nominal or steady state volume depends on factors such as the nominal feed flow rates (F_c) and temperature (T_c) of the cold stream whereas the variability with respect to this nominal volume depends on both, the nominal volume around which the variability is estimated and the controller tuning parameters that determines the process closed loop dynamic performance. The operating

costs for this system are assumed to be zero. Thus, the cost function used to simultaneously design the mixing tank process is given by:

$$V_d = \left(V_{nom} + \mathrm{H}^V\right) \tag{3.25}$$

where V_d denotes the tank's total size, V_{nom} represents the tank's volume at a steady state condition and H^V represents a bound on the tank's hold up variability due to the external disturbances.

Table 3.1 Process model and data for the mixing tank process.

mixing tank process model	$\frac{dV}{dt} = F_h + F_c - z\sqrt{V}$ $V\frac{dT}{dt} = F_h(T_h - T) + F_c(T_c - T)$
Disturbance variable (υ)	350 K≤ T_h≤ 390 K
Process parameter uncertainty (ω)	0.05 m^3/hr≤ F_h≤ 0.15 m^3/hr
State variables (**y**)	$V(t)$≥0.9 m^3, 300 K≤ $T(t)$≤ 370 K
Available manipulated variables (**κ**)	0.0015 m^3/hr≤ F_c≤ 2.0 m^3/hr, 0 ≤ z≤ 1.0
Constant variables (c)	T_c=298 K

Two different scenarios are considered: Scenario I-Temperature control only and Scenario II-Simultaneous control of temperature and volume. These two scenarios are described in the following subsections.

3.2.1 Scenario I-Temperature control only

The first strategy proposed for the design of this process is to use a proportional-integral (PI) controller to maintain the tank's outlet temperature $T(t)$ within the specified bounds by manipulating the flow rate of the cold stream, F_c. This controller will make the necessary adjustments on F_c to reject the perturbations in the hot process stream (F_h and T_h) entering the process. In this case, the tank's hold up, $V(t)$, is not controlled and the stem position, z, remains at a fixed value, to be specified from optimization, for the entire period of operation. Thus, the design is expected to result in a larger volume to the one expected to be obtained for *Scenario II* where both the tank's temperature and volume are being simultaneously controlled.

For the purpose of formulating the closed loop system in state space form, the state space form of a PI controller is used as follows:

$$\begin{aligned} \dot{\xi}_T &= \frac{1}{\tau_{IT}} e_T \\ F_c &= F_{c,nom} + Kc_T\left(e_T + \xi_T\right) \end{aligned} \tag{3.26}$$

where Kc_T and τ_{IT} are the PI controller tuning parameters that form the vector $\boldsymbol{\lambda}$, $F_{c,nom}$ represents the nominal value in F_c ($\overline{\mathbf{u}}$), ξ_T denotes the controller's state and e_T denotes the error between the set point, T_{sp}, and the actual tank's temperature as follows:

$$e_T = T_{sp} - T(t) \tag{3.27}$$

The mixing tank process model equations given in Table 3.1 are combined with (3.26) and (3.27) to simulate the process closed-loop dynamic behaviour. Since the sole objective for this process is to minimize the tank's total cost subject to process constraints, the process variability function (ϕ^d) is equal to the tank's total volume, V_d which is given by equation (3.25). Thus, the worst-case scenario for this process is estimated as follows (see problem 3.22):

$$\max_{F_h^l \le F_h \le F_h^u} \left(V_{nom} + \mathrm{H}^V\right) \tag{3.28}$$

where H^V is estimated from (3.21) and (3.19), respectively. To compute the value of γ^V, it is necessary first to estimate a state space model with model parameter uncertainty (3.7) from the disturbance variable, T_h, to the output variable, $V(t)$. This model is identified from input/output data recorded from the actual closed-loop process model equations following the procedure explained in Section 3.1.6.

To guarantee that the outlet temperature remains within its upper and lower bounds at any time t, the problem in (3.23) is reformulated as follows:

$$\begin{aligned} T_{sp} + \max_{F_h^l \le F_h' \le F_h^u} \mathrm{H}^T - T^u \le 0 \\ T^l - T_{sp} + \max_{F_h^l \le F_h' \le F_h^u} \mathrm{H}^T \le 0 \end{aligned} \tag{3.29}$$

where T^l and T^u represent respectively the lower and upper bounds for the tank's outlet temperature. To calculate H^T, a robust model of the structure of (3.7) has to be identified from T_h to $T(t)$ for every F_h tested by following the procedure detailed in Section 3.1.6. Expressions similar to (3.29) can be derived to test the constraints on the manipulated variable, F_c, and on the tank's minimum hold up. The integer parameter n required to calculate H in problems (3.28)-(3.29) and the process feasibility constraints were set to unity corresponding to one sigma in the normal distributions. Furthermore, to ensure process stability, robust models from T_h to $V(t)$ and $T(t)$ are obtained using the F_h values obtained from the solution of problems (3.28) and (3.29), respectively.

The mathematical formulation used to design the mixing tank when only one PI controller for temperature control is given in (3.30). The decision variables specified *Scenario I*'s formulation are the controller gain, Kc_T, the controller integral time, τ_{IT}, the stem position of the valve at the outlet, z, and the tank's temperature set point, T_{sp}. Although the process model equations and the control algorithm equations are not explicitly shown in (3.30), they appear in the form of robust models, identified from the simulation of the process closed-loop model equations, and used to measure the process stability, variability and feasibility.

To illustrate that the above problem has a feasible search space, a graphical solution was obtained for which the controller's tuning parameters (Kc_T and τ_{IT}), specified as decision variables in (3.30), were set to be constant and equal to -30 K*hr/m^3 and 5 hr respectively. The remaining set of decision variables shown in the above optimization problem, i.e. the stem position (z) and the tank's temperature set point (T_{sp}), were assumed to be the decision variables for this simplified case.

Figure 3.2 shows a contour plot of the cost function values (3.25) obtained for a specific range of values of the decision variables (z and T_{sp}). This Figure also shows the active constraints for the range of values selected for the decision variables. The area enclosed by the active constraints, shown as dashed lines in Figure 3.2, represents the active feasible search space where the optimal solution lies. According to this figure the two active constraints defining the feasible region are the minimum volume constraint and the

maximum cold-feed flow rate constraint, specified in the problem formulation (3.30), respectively.

$$
\begin{aligned}
&\min_{\boldsymbol{\eta}=[Kc_T,\tau_{IT},z,T_{sp}]} \quad V_d && \text{…Cost Function}\\
&s.t.\\
&\quad \mathbf{A}_V(m_k)^T\mathbf{P}+\mathbf{P}\mathbf{A}_V(m_k)<\mathbf{0}\\
&\quad \mathbf{A}_T(m_k)^T\mathbf{P}+\mathbf{P}\mathbf{A}_T(m_k)<\mathbf{0} && \text{…Process stability test}\\
&\quad \max_{F_h^l\le F_h\le F_h^u}\left(V_{nom}+\mathrm{H}^V\right) && \text{… Worst - case scenario evaluation}\\
&\quad T_{sp}+\max_{F_h^l\le F_h^1\le F_h^u}\mathrm{H}^T-T^u\le 0\\
&\quad T^l-T_{sp}+\max_{F_h^l\le F_h^1\le F_h^u}\mathrm{H}^T\le 0\\
&\quad F_{c,nom}+\max_{F_h^l\le F_h^2\le F_h^u}\mathrm{H}^{F_c}-F_c^u\le 0\\
&\quad F_c^l-F_{c,nom}+\max_{F_h^l\le F_h^2\le F_h^u}\mathrm{H}^{F_c}\le 0\\
&\quad V^l-V_{nom}+\max_{F_h^l\le F_h^3\le F_h^u}\mathrm{H}^V\le 0\\
&\quad \boldsymbol{\eta}^{\mathbf{l}}\le\boldsymbol{\eta}\le\boldsymbol{\eta}^{\mathbf{u}} && \text{…Process feasibility constraints}\\
&\quad \dot{V}(t)=F_h+F_c-zV^{1/2}\\
&\quad \dot{T}(t)=\frac{\left[F_h\left(T_h(t)-T(t)\right)+F_c(t)\left(T_c-T(t)\right)\right]}{V(t)} && \text{…Process model equations *}\\
&\quad \dot{\xi}_T=\frac{1}{\tau_{IT}}e_T\\
&\quad F_c=F_{c,nom}+Kc_T\left(e_T+\xi_T\right)\\
&\quad e_T=T_{sp}-T(t) && \text{…Control algorithm equations *}
\end{aligned}
\tag{3.30}
$$

The graphic solution for the simplified case shown in Figure 3.2 is not practical for the case where all variables including the controller tuning parameters are considered or optimization. Accordingly, the full optimization problem shown in (3.30) was solved in MATLAB® using Sequential Quadratic Programming (SQP) as the optimization algorithm[54-55]. The results obtained from this calculation are shown in Table 3.2. The resulting relative large volume obtained for this scenario was somewhat expected since the tank's hold up is not being controlled. Furthermore, the other potential manipulated variable in the process, z, is kept at a constant value for the entire period of operation whereas it could be effectively used for volume control $V(t)$ as shown later for *Scenario II*.

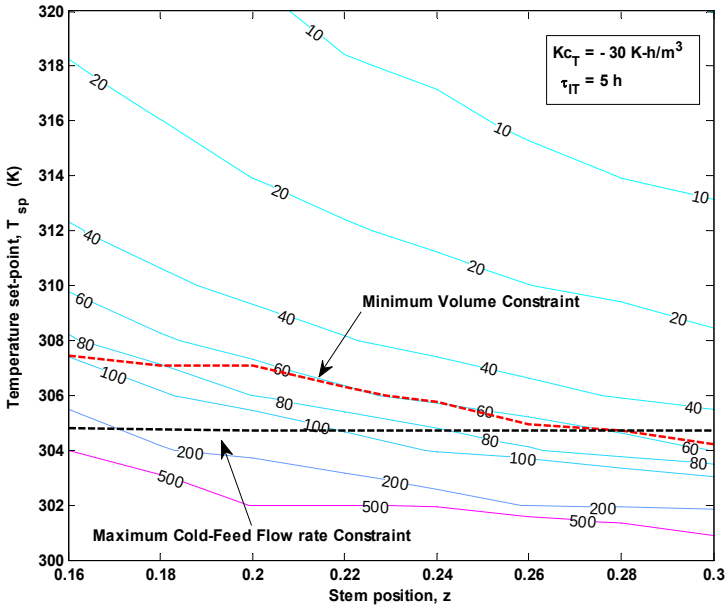

Figure 3.2 Feasibility region, *Scenario I*.

Table 3.2 Mixing tank's design, *Scenario I*, QLF-based methodology

Variables	Solution
$Vd = (V_{nom}+H^V)$	37.70 m3
z	0.3343
Tsp	304.68 K
$Fc_{,nom}$	1.4652 m3/hr
K_{CT}	-37.78
τ_{IT}	5.7225 hr

To test the results obtained by the present methodology, the full nonlinear process model (Table 3.1) and the controller equations (3.26) and (3.27) were used to simulate the mixing tank's design for more than 1 year of operation. The process dynamics were generated using the hot stream temperature profile shown in Figure 3.3 with the hot feed flow rate, F_h, fixed to a steady-state value for the entire period of simulation. For this case study, the design was simulated for each of the hot feed flow rate extreme values, i.e., for the first simulation, F_h was fixed to 0.15 m^3/hr whereas for the last simulation, F_h was fixed to 0.05 m^3/hr, respectively. The results of these simulations are shown in Figure 3.4 through Figure 3.6. Specifically, Figure 3.4 shows that the tank's actual hold up never exceeds the tank's total

size, V_d. Similarly, Figure 3.5 and Figure 3.6 show that the outlet temperature and the cold feed flow rate fluctuate between their extreme limits. Thus, the design obtained by this approach remains stable and within the specified feasible region for many different changes in the flow rates and temperature values of the process hot stream.

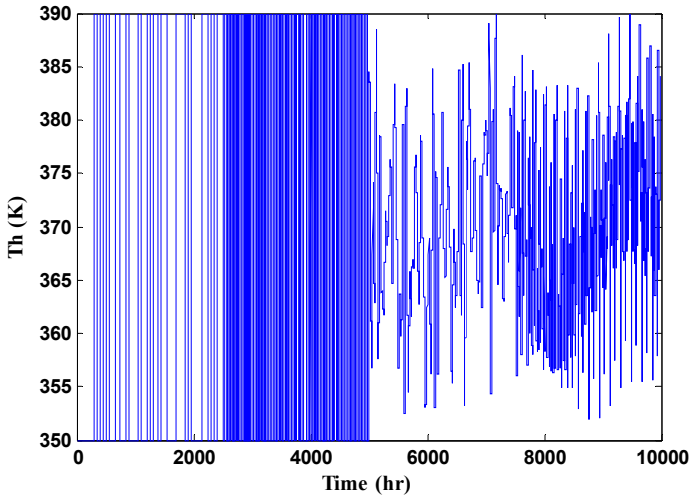

Figure 3.3 Hot stream temperature profile used to simulate the design.

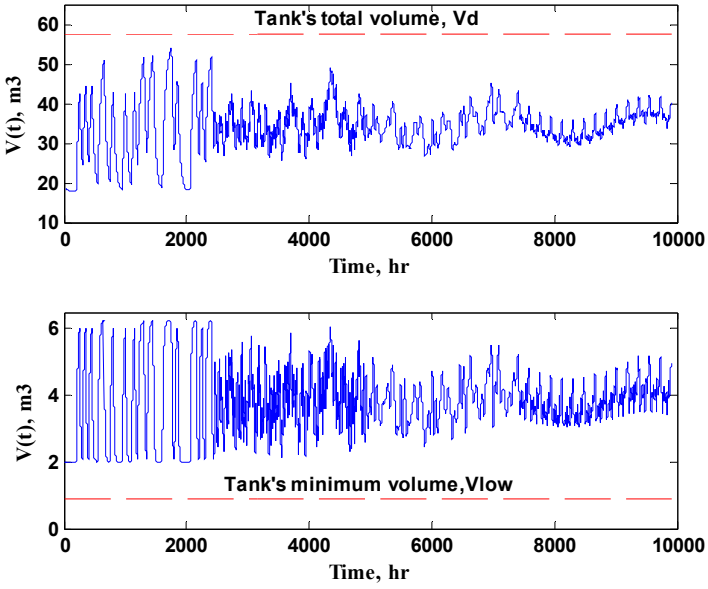

Figure 3.4 Mixing tank's actual volume, *Scenario I.*

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

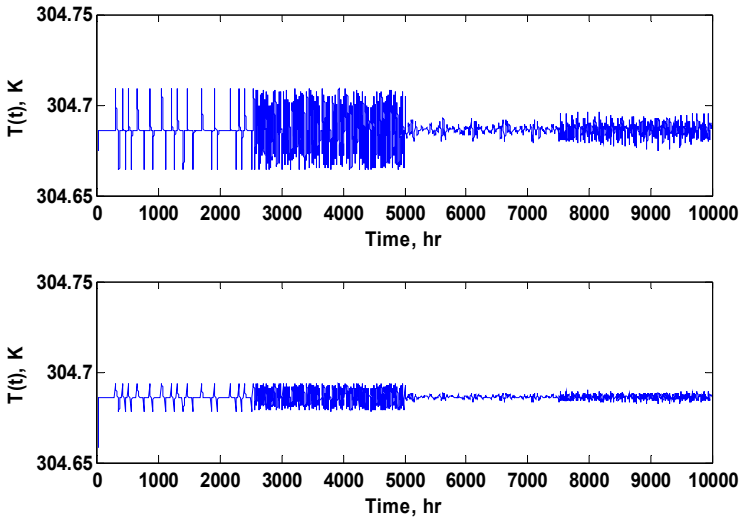

Figure 3.5 Mixing tank's actual temperature, *Scenario I*.

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

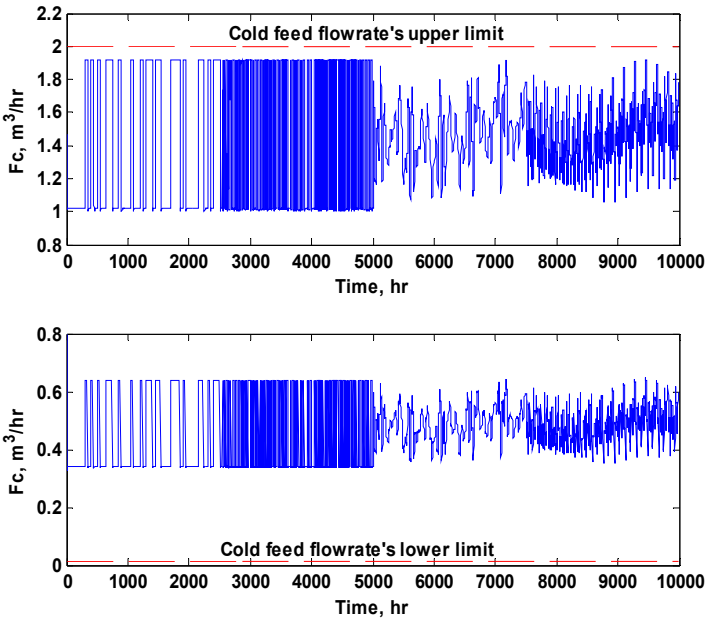

Figure 3.6 Cold feed flow rate's actual values, *Scenario I*.

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

3.2.2 Scenario II-Simultaneous Control of Temperature and Volume

In this scenario the optimization problem is solved using a control strategy whereby both the tank's volume and temperature are under closed-loop control. Thus, an additional PI controller that controls the tank's volume by adjusting the stem position of the valve at the outlet, z, was added to the temperature control used in *Scenario I*.

The equations that describe the performance of the additional PI controller are given by:

$$\begin{aligned} \dot{\xi}_V &= \frac{1}{\tau_{IV}} e_V \\ z &= z_{nom} + Kc_V \left(e_V + \xi_V \right) \end{aligned} \tag{3.31}$$

where Kc_V and τ_{IV} are the PI controller tuning parameters that controls the tank's hold up, z_{nom}, represents the nominal value in z, ξ_V denotes the controller's state and e_V denotes the feedback error between the set point for the tank's volume, V_{sp}, and the actual tank's hold up:

$$e_V = V_{sp} - V(t) \tag{3.32}$$

Thus, the closed-loop dynamics for this system are described by the process model equations shown in Table 3.1 and the controller equations, specified in (3.26), (3.27), (3.31) and (3.32), respectively. The process maximum variability was calculated using (3.28). In this case however, the nominal tank's hold up (V_{nom}) is redefined as the tank's volume set point (V_{sp}) since this output variable is now controlled by a PI controller. Also, the stem position, z, now becomes a manipulated variable (**u**) to control the tank's volume at its desired set point. Accordingly, constraints on the variable z must be included in the analysis to ensure that it remains within bounds as follows,

$$\begin{aligned} z_{nom} + \max_{F_h^l \le F_h' \le F_h^u} \mathrm{H}^z - z^u \le 0 \\ z^l - z_{nom} + \max_{F_h^l \le F_h' \le F_h^u} \mathrm{H}^z \le 0 \end{aligned} \tag{3.33}$$

To compute H^z, a robust model like (3.7) that relates the disturbance T_h to z for every F_h tested, is identified. The process constraints defined for *Scenario I* and inequalities (3.33) represent the feasibility problem specified for *Scenario II*. As in *Scenario I*, the integer

parameter n required to compute the different variabilities (H's) related to the feasibility constraints was initially set to the unity but it was increased later to analyze its effect. In addition, the process stability test is performed using the same approach used in *Scenario I*.

The QLF-based (γ-based) methodology developed in this research work was applied to simultaneously design and control the mixing tank process when both state variables, $V(t)$ and $T(t)$, are under closed-loop control. The mathematical formulation defined for *Scenario II* is given in (3.34). The decision variables specified for this problem are the controllers' gain, Kc_T and Kc_V, the controllers' integral time, τ_{IT} and τ_{IV}, and the set points of the tank's temperature and volume, T_{sp} and V_{sp}, respectively. As in *Scenario I*, the proposed optimization problem posed for *Scenario II* was implemented in MATLAB® using Sequential Quadratic Programming (SQP) as the optimization algorithm[54-55]. The results obtained for this problem are shown in Table 3.3 ($H = \gamma$).

Table 3.3 Mixing tank's design, *Scenario II* and Mohideen et al.[16]'s design

Design Variables	Solution (**H**=γ)	Solution (**H**=2γ)	Solution Mohideeen
V_d (m^3)	0.9253	1.0462	1.0
V_{sp} (m^3)	0.9153	1.0001	--
z_{nom}	0.4740	0.5973	--
T_{sp} (K)	321.81	316.07	360.0
$F_{c,nom}$ (m^3/hr)	0.3035	0.4476	0.035
Kc_T	-21.8402	-27.5806	-0.005
Kc_V	-24.5018	-12.5709	--
τ_{IT} (hr)	2.3702	4.0201	5.0
τ_{IV} (hr)	1.9506	1.9805	--

Since the tank's hold up is maintained at a specified set point by a control algorithm, the variability in the mixing tank process was significantly reduced in comparison with *Scenario I*. Consequently, the tank's total size required to reject the disturbances in the process hot stream is smaller than the design obtained in *Scenario I*.

$$\min_{\boldsymbol{\eta}=[Kc_T,\tau_{IT},Kc_V,\tau_{IV},T_{sp},V_{sp}]} \; V_d \qquad \ldots \text{Cost Function}$$

s.t.

$$\mathbf{A}_V(m_k)^T\mathbf{P}+\mathbf{P}\mathbf{A}_V(m_k)<\mathbf{0}$$

$$\mathbf{A}_T(m_k)^T\mathbf{P}+\mathbf{P}\mathbf{A}_T(m_k)<\mathbf{0} \qquad \ldots \text{Process stability test}$$

$$\max_{F_h^l\le F_h\le F_h^u}\left(V_{sp}+\mathrm{H}^V\right) \qquad \ldots \text{Worst - case scenario evaluation}$$

$$T_{sp}+\max_{F_h^l\le F_h^1\le F_h^u}\mathrm{H}^T-T^u\le 0$$

$$T^l-T_{sp}+\max_{F_h^l\le F_h^1\le F_h^u}\mathrm{H}^T\le 0$$

$$F_{c,nom}+\max_{F_h^l\le F_h^2\le F_h^u}\mathrm{H}^{F_c}-F_c^u\le 0$$

$$F_c^l-F_{c,nom}+\max_{F_h^l\le F_h^2\le F_h^u}\mathrm{H}^{F_c}\le 0$$

$$V^l-V_{nom}+\max_{F_h^l\le F_h^3\le F_h^u}\mathrm{H}^V\le 0$$

$$z_{nom}+\max_{F_h^l\le F_h^4\le F_h^u}\mathrm{H}^z-z^u\le 0$$

$$z^l-z_{nom}+\max_{F_h^l\le F_h^4\le F_h^u}\mathrm{H}^z\le 0$$

$$\boldsymbol{\eta}^l\le\boldsymbol{\eta}\le\boldsymbol{\eta}^u \qquad \ldots \text{Process feasibility constraints}$$

$$\dot{V}(t)=F_h+F_c-zV^{1/2}$$

$$\dot{T}(t)=\frac{\left[F_h\left(T_h(t)-T(t)\right)+F_c(t)\left(T_c-T(t)\right)\right]}{V(t)} \ldots \text{Process model equations *}$$

$$\dot{\xi}_T=\frac{1}{\tau_{IT}}e_T$$

$$F_c=F_{c,nom}+Kc_T\left(e_T+\xi_T\right)$$

$$e_T=T_{sp}-T(t)$$

$$\dot{\xi}_V=\frac{1}{\tau_{IV}}e_V$$

$$z=z_{nom}+Kc_V\left(e_V+\xi_V\right)$$

$$e_V=V_{sp}-V(t) \qquad \ldots \text{Control Algorithm equations *} \qquad (3.34)$$

The design obtained for *Scenario II* was simulated for more than 1 year of operation using the complete process model and the controller equations specified above. As in *Scenario I*, two simulations were carried out: the first one with the hot feed flow rate fixed to its upper value and the last one with the same variable fixed to its lower value. The hot stream temperature profile shown in Figure 3.3 was used for both simulations. Figure 3.7 shows the

simulated tank's hold up. This Figure shows that when F_h is kept constant at its upper value, the actual tank's hold up is very close to the maximum volume obtained by the optimization. This result can be explained by the selection made on the value of the integer parameter n, used to estimate a bound on the different variability's around the mixing tank process (H^V, H^T, H^{Fc}, and H^z). As mentioned earlier in this chapter, this integer number specifies the multiples of the parameter γ to be used in the simultaneous design calculations.

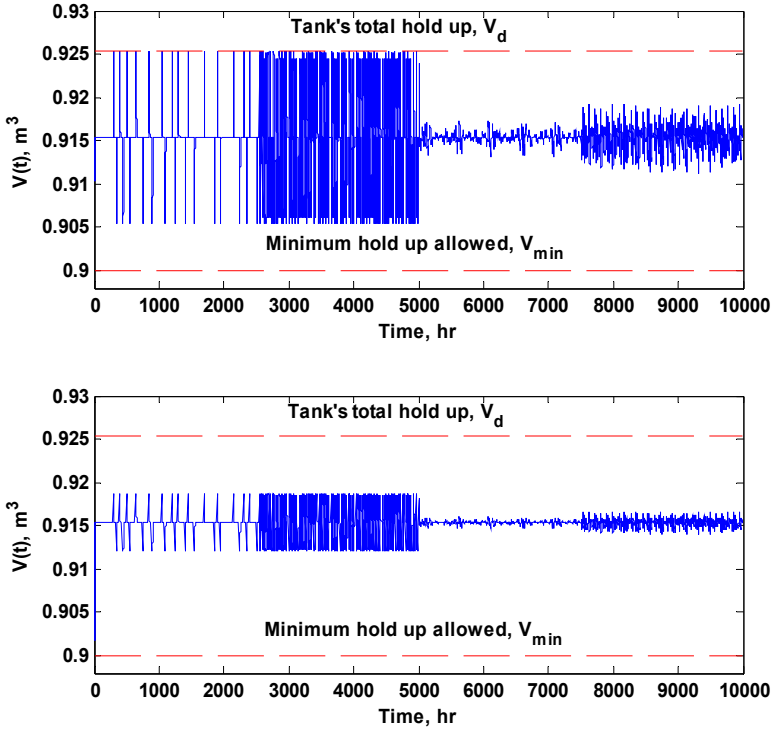

Figure 3.7 Mixing tank's actual volume, *Scenario II* (H=γ).

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

Based on the results shown in Table 3.3 ($H = \gamma$), setting n=1 may not sufficient to cope with the hot process stream variations since in that case the bound is expected to account for only 68.26% of the process variability. Therefore, the simultaneous design calculations for *Scenario II* were redone by setting the process variabilities H^V, H^T, H^{Fc}, and H^z, to be equal to twice the value of the corresponding bounds γ's (n=2). Table 3.3 shows the results obtained following this calculation ($H = 2\gamma$) where as expected, the tank's total size is slightly larger

than the value obtained when the integer number n was set to the unity. The resulting design was simulated using the complete process closed-loop nonlinear model and the hot stream temperature profile shown in Figure 3.3. Figure 3.8 shows that, when $H = 2\gamma$, the tank's actual hold up remains within the limits specified by the resulting design. Thus, the current value of the parameter n ensures that the resulting design obtained for this process can accommodate, within the design's feasible region, the perturbations that may occur in the process hot stream. Figure 3.9 through Figure 3.11 show that the rest of the mixing tank process variables, $T(t)$, F_c, and z, are also kept within the pre-specified limits for the selected period of operation.

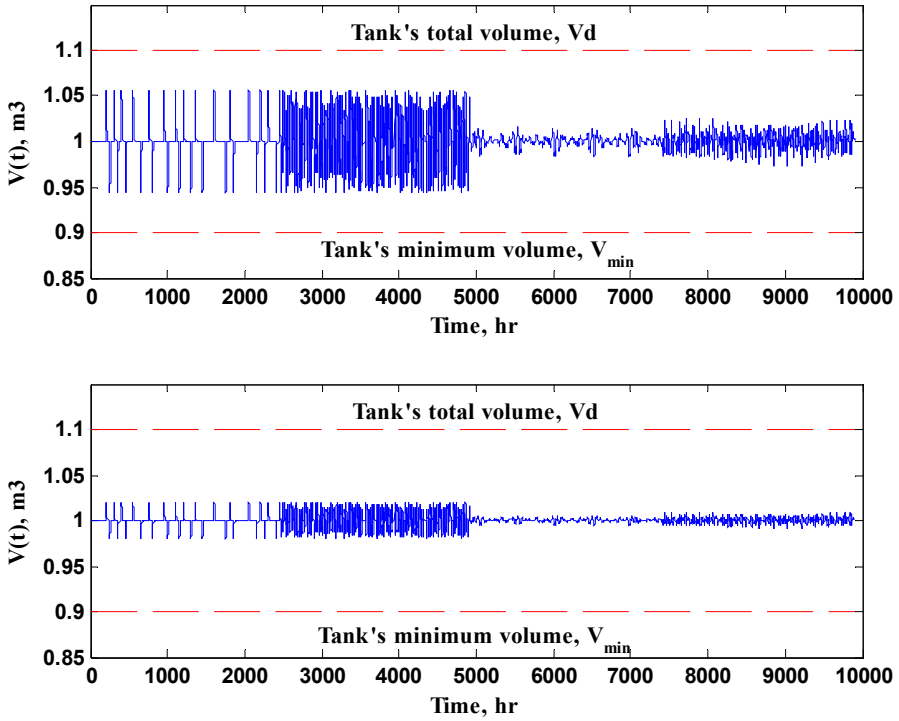

Figure 3.8 Mixing tank's actual volume, *Scenario II* (H=2γ).

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

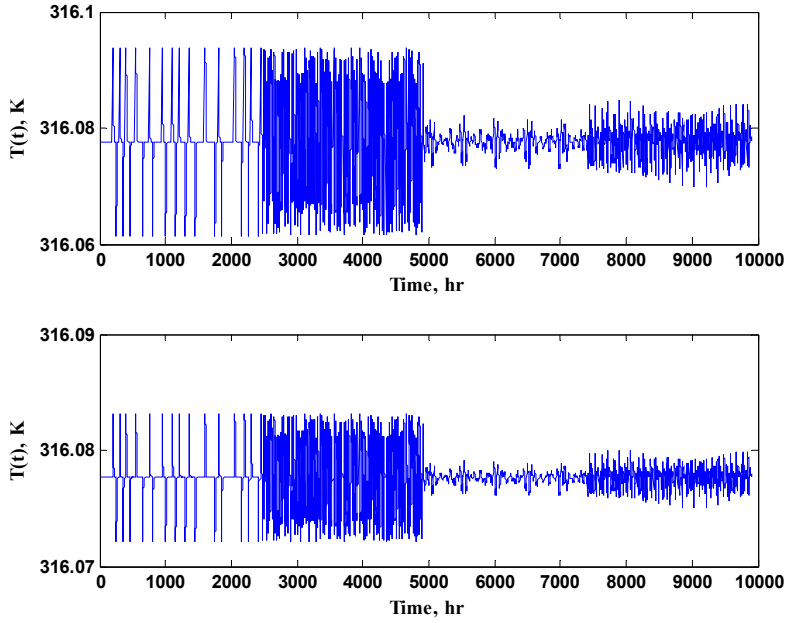

Figure 3.9 mixing tank's actual temperature, *Scenario II* (H=2γ).

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

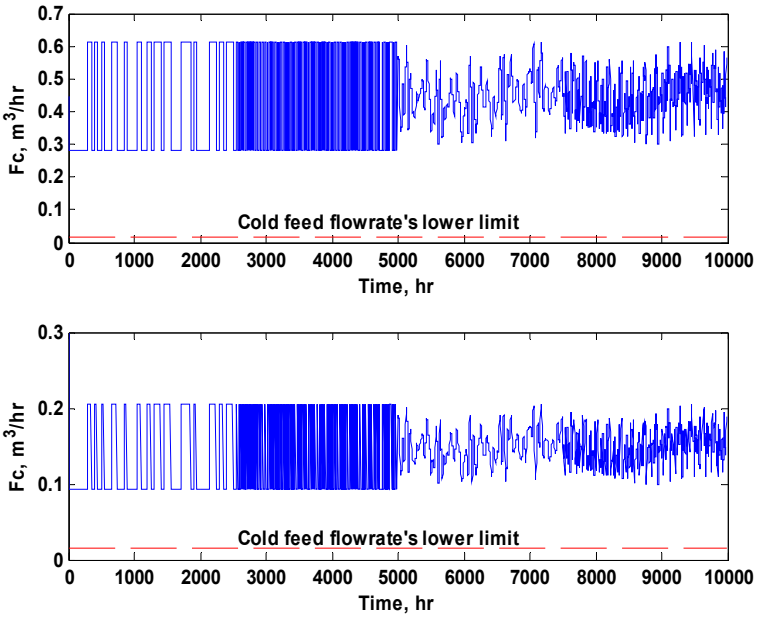

Figure 3.10 Cold feed flow rate's actual values, *Scenario II* (H=2γ).

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

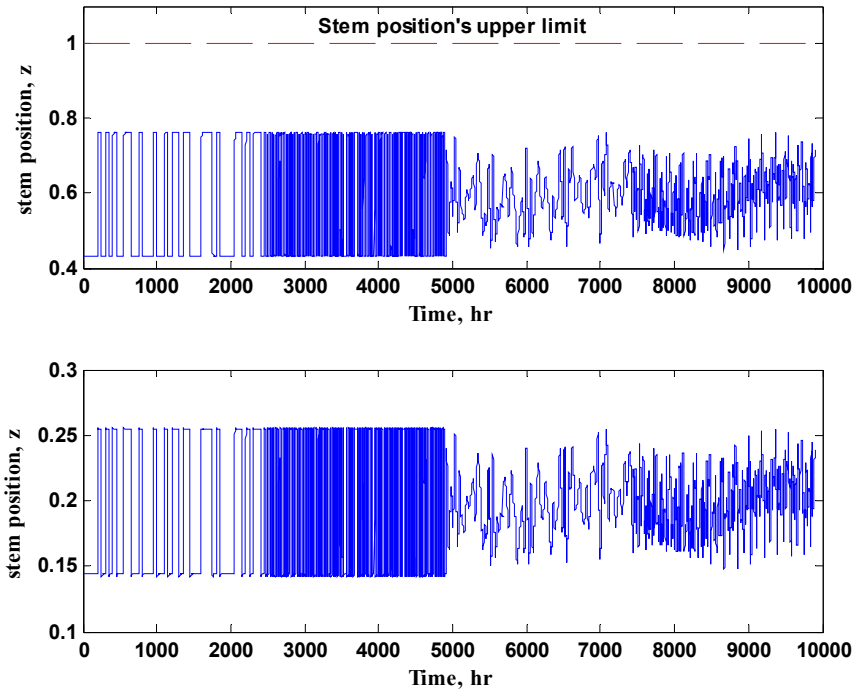

Figure 3.11 Valve constant's actual values, *Scenario II* (H=2γ).

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

Since Mohideen et al.[16] proposed a dynamic programming approach to solve the same mixing tank problem considered above, it was of interest to conduct a comparison between their results to the results by using the robust modelling and design approach proposed in this research work. Table 3.3 shows the results obtained when Mohideen's methodology was applied to the mixing tank process. These authors solved this problem applying a mixed integer stochastic dynamic optimization-based approach. Although their design resulted in a slightly smaller tank and only one PI-controller was used to control $T(t)$ by manipulating F_c, the stem position of the outlet valve, z, was manipulated based on a nonlinear stochastic dynamic optimization to accommodate the changes in F_h for only 30 hours of operation. Since the bounds calculated in the robust approach are applicable to an infinite time horizon, a full fair comparison between Mohideen's approach and the robust model approach proposed in this thesis is not possible. Clearly, the calculation of bounds along an infinite time horizon has a more practical value than a design applicable for a finite time horizon

only. However, it should be pointed out that Mohideen et al.[16] solved the finite time horizon problem to illustrate the analytical steps of their approach rather than promoting the practicality of their design.

The solution proposed in this work also differs from Mohideen's solution from the point of view of the control strategies used to regulate the process. For example, in the second scenario, a PI controller is considered for the mixing tank's volume control by manipulating z whereas Mohideen search for the optimal changes in z based on the assumption that the hot stream flow rate changes are known and are available for the optimization algorithm. Although the volume obtained for *Scenario II* by the proposed robust model based strategy is larger than Mohideen's and the bounds on the process variables were relaxed to account for the steady-state condition, the implementation of the designs obtained by the present approach is very simple and practical since the controller does not require measurements of the hot stream flow rate F_h, that it strictly treated as unmeasured perturbations.

In the present case study, the hot feed flow rate, F_h, was assumed to be an unmeasured perturbation that remains at a steady-state value for long periods of time and consequently it was treated as a process parameter uncertainty ($\boldsymbol{\omega}$). Therefore, its transients were ignored in the present analysis. The dynamics of this perturbation could have been taken into account if F_h would have been considered as a disturbance. However, treating $\boldsymbol{\omega}$ as a disturbance may have resulted in larger uncertainty values and consequently, more conservative designs.

3.3 Expected Computational Burden

One of the key motivations to use a robust model based strategy was to reduce the computational effort that is required for finding an optimal design. Therefore, to assess the computational advantages of the proposed method, it was compared on the basis of computation time with a dynamic programming based method. In this regard, a single calculation of variability has been solved for the mixing tank process using both strategies. Since the variability calculation based on closed-loop identification is the core of the proposed method, it makes sense to compare the present method to other strategies on the basis of this single calculation of variability that has to be repeated many times during the optimization.

For the present method, a variability calculation for the mixing tank process is obtained from the solution of problem (3.28) for a given value of the hot feed flow rate (F_h). The solution of this problem provides an infinite time-horizon bound on the tank's volume variance around a nominal operation point, specified by the nominal value of cold feed flow rate (F_c), the stem position of the valve at the outlet (z), the cold stream temperature (T_c) and the controller's gain and integral time (Kc_V and τ_{IV}). The nominal values for the above variables have been taken from the solution of *Scenario I*, shown in Table 3.2. The process parameter uncertainty in this system (F_h) was set to a specific value, F_h=0.15 m^3/hr. The tank's nominal volume and temperature (V_{nom} and T_{nom}) are calculated from the steady-state solution of the process model equations.

The procedure to estimate the worst variability applying the current methodology has been explained in Section 3.1.6 and is defined specifically for *Scenario I* as follows:

$$
\begin{aligned}
& V_d = \left(V_{nom} + H^V\right) && \ldots \text{Mixing tank's maximum variability} \\
& V_{nom} = \frac{F_h + F_{c,nom}}{z^2} && \\
& T_{sp} = \frac{F_h T_h + F_{c,nom} Tc}{F_h + F_{c,nom}} && \ldots \text{Mixing tank's steady - state model} \\
& \dot{x} = \mathbf{A}(\theta_A)x + \mathbf{B}(\theta_B)T_h && \\
& \begin{bmatrix} V & T \end{bmatrix}^T = \mathbf{C}(\theta_C)x + \mathbf{D}(\theta_D)T_h && \ldots \text{Robust closed - loop model} \\
& \mathrm{H} = n\gamma && \ldots \text{Ouput variability definition} \\
& \min_{\mathbf{P}} \gamma^2 && \ldots \text{Single variability calculation} \\
& s.t. && \\
& \begin{bmatrix} \mathbf{A}(s_l)^T\mathbf{P} + \mathbf{A}(s_l)\mathbf{P} & \mathbf{B}(s_l)\mathbf{P} & \mathbf{C}(s_l)^T \\ \mathbf{B}(s_l)^T\mathbf{P} & I & \mathbf{D}(s_l)^T \\ \mathbf{C}(s_l) & \mathbf{D}(s_l) & -\mathbf{I} \end{bmatrix} < \gamma^2 \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} && \\
& \mathbf{P} > \mathbf{0}, \mathbf{P}^T = \mathbf{P} &&
\end{aligned}
\tag{3.35}
$$

Similarly, the dynamic optimization formulation used to solve same problem is as follows:

$$
\begin{aligned}
&\max_{T_h \in \mathbf{T_h}(t)} V(t) && \text{…Mixing tank's maximum volume at any time } t \\
&s.t. \\
&\quad V(0) = \frac{F_h + F_{c,nom}}{z^2} \\
&\quad T_{sp}(0) = \frac{F_h T_h(0) + F_{c,nom} Tc}{F_h + F_{c,nom}} && \text{……Mixing tank's steady - state model} \\
&\quad \xi(0) = 0 \\
&\quad \dot{V}(t) = F_h + F_c(t) - z\sqrt{V(t)} \\
&\quad \dot{T}(t) = \frac{F_h\left(T_h(t) - T(t)\right) + F_c(t)\left(T_c - T(t)\right)}{V(t)} && \text{…Process model equations} \\
&\quad \dot{\xi}_T = \frac{1}{\tau_{IT}} e_T \\
&\quad F_c = F_{c,nom} + Kc_T\left(e_T + \xi_T\right) \\
&\quad e_T = T_{sp} - T(t) && \text{…Control algorithm equations} \\
&\quad \mathbf{T_h}(t) = \left\{T_h(t), T_h(t)^l \le T_h(t) \le T_h(t)^u\right\} && \text{…Disturbance vector} \\
&\quad t \in \left[0, t_f\right]
\end{aligned}
\tag{3.36}
$$

As it is shown in the above description, the complete closed-loop mixing tank model is used to seek for the profile in the hot stream temperature, T_h, which produces the largest variation in the tank's volume within a finite time horizon, t_f. The decision variable in this problem is a vector composed of the hot stream temperature values at each sampling period *i* that are used as inputs to the process model equations. The problem shown described by the set of equations (3.36) was coded in MATLAB®. Sequential Quadratic Programming (SQP) was used as the optimization method to solve this problem. Similarly, the MATLAB built-in function *ODE23s* was used to integrate the process model differential equations[56]. For each new set of decision variables chosen by the SQP algorithm, the process model differential equations were solved applying the following procedure:

i) Using the first element of the vector of the hot stream temperature values, initial values for the tank's volume and temperature are calculated from the steady-state solution of the process model equations; the initial values of the controller states are set to zero.

ii) At the i^{th} sampling period, the model differential equations are integrated from the previous sampling period i^{th} until the next $(i+1)^{th}$ using a modified version of the Rosenbrock method[56], the i^{th} element of the vector of hot stream temperature values is kept constant for this period of integration. The values of the tank's volume and temperature and the controller states obtained for the $(i+1)^{th}$ sampling period are recorded and used as initial values for the next period of integration. Also, the $(i+1)^{th}$ element of the hot stream temperature values is used and kept constant for the next period of integration.

iii) This procedure is repeated until the finite time horizon t_f is reached. Once completed, the tank's maximum volume variability is sought from the solution of the differential equations at each sampling period i. Since this value represents the problem's objective function to be maximized, it is used by the optimization algorithm to choose a new set of hot stream temperature values.

As in the proposed method, the nominal values for the process variables (F_c, T_c, z, Kc_T and τ_{IT}) and the process parameter uncertainty F_h were also taken from the solution of *Scenario I*, shown in Table 3.2.

To test the computational burden, the problems shown in (3.35) and (3.36) were solved on a Pentium 4, 3.01GHz with 1.5GB of RAM. The ratio of the CPU time (seconds) was approximately *27.42:329.20* in favour of the present methodology, i.e. the proposed methodology is more than 12 times faster than dynamic optimization. Similarly, the problems posed for *Scenario II* were also solved using both strategies. For this case, the CPU time ratio was approximately *18.94:183.12* in favour also of the present approach. These results show that the computational demands for the proposed method are much less than for dynamic programming-based methods. Also, these results can be seen as a potential advantage of the proposed method to simultaneously design and control large-scale processes. To this regard, the formulation proposed for *Scenario I* was used to provide an estimate of the expected computational burden when more than one variability calculation is to be performed with the present method. To obtain these results, four different cases were considered:

i) Worst-variability calculation (3.35).

ii) Worst-variability calculation (3.35) and minimum volume constraint as specified in problem (3.30).

iii) Worst-variability calculation (3.35), minimum volume constraint and maximum temperature constraint as specified in problem (3.30).

iv) Worst-variability calculation (3.35), minimum volume constraint and maximum and minimum temperature constraints as specified in problem (3.30).

In all the cases, the CPU time was recorded. Figure 3.12 shows the CPU time used by the present methodology when the number of variability calculations is increased. As shown, the CPU time is approximately proportional to the number of variables for which the worst variability is calculated. This may be used as an indicator of the expected computational requirement when the present methodology is applied to simultaneously design and control large-scale processes with a large number of variables that should be optimized or kept within limits.

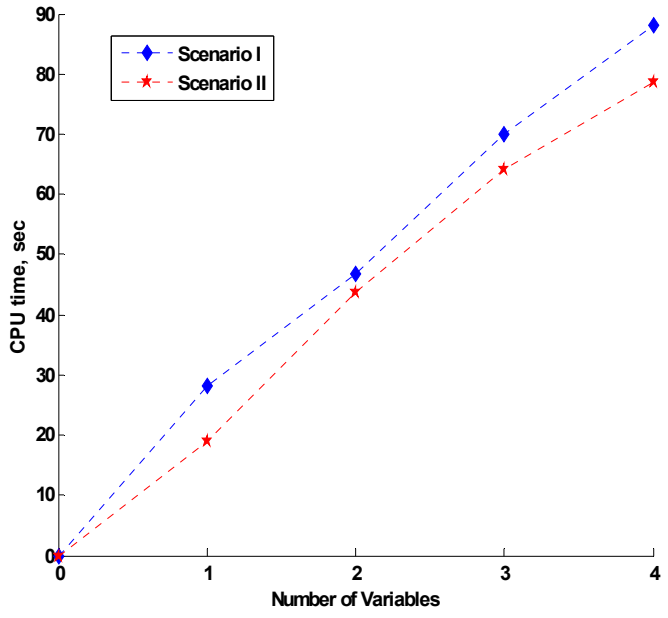

Figure 3.12 CPU time as a function of the number of variables.

In summary, this chapter has presented a Quadratic Lyapunov function-based approach to integrate process design and control for dynamic systems under uncertainty and disturbances. This method (γ-based) is based on the closed loop identification of a robust state space model from the complete process nonlinear model. The uncertain parameters that appear in the robust model capture the process nonlinearities due to the changes in the disturbance variables. The identified robust model is used to evaluate the robustness in the design. The tests determine a bound on the standard deviation of the robust models output. Then, multiples of the calculated bound are used to measure process variability and feasibility. Also, process asymptotic stability is strictly enforced by considering a robust stability constraint within the methodology framework.

The present method was applied to the integration of design and control of a mixing tank problem using two different control strategies. Near violations of constraints were found for the *Scenario II* when only one standard deviation is considered for variability calculations ($H=\gamma$). These violations were eliminated by considering two standard deviations in the calculations of variability ($H=2\gamma$). In the next chapter, a new technique to estimate a bound on the largest output error, instead of the output error's variance considered in this chapter, is presented and incorporated within the methodology framework.

Additionally, an analysis of the computational burden was performed on the proposed method and compared to a dynamic programming-based variability calculation. The results show that the present method is an order of magnitude faster than dynamic programming. This result makes the proposed method computationally attractive and justifies its application to the simultaneous design and control large-scale processes.

4. Worst-case Variability Estimation for the Simultaneous Design and Control of Dynamic Systems: A *SSV* analysis Approach.

Although the methodology presented in the previous chapter is computationally efficient, the robust variability index (γ) used to evaluate process variability and process feasibility only provides *a bound on the process output error's variance*. This means that γ only gives an estimation of the process output variability in average sense and it may not consider the cases when a particular perturbation profile may produce, at a specific time t, a large deviation that cannot be captured by this index. Thus, as explained in the previous chapter, multiples of this index are needed to guarantee that the resulting design satisfies the process constraints and process variability restrictions. Consequently, a set of optimization runs using different multiples of γ may be needed to obtain a design that completely satisfies the process specification and constraints.

This chapter introduces a new robust measure that provides *a bound on the largest output error* with respect to a set point and the critical time-dependent profile in the disturbance variables ($\boldsymbol{\upsilon}$) that produces this deviation from a given normal operation point. This measure, based on a Singular Structured Value (SSV) test, μ-analysis, is used to estimate the worst-case process variability scenario and to test the feasibility constraints considered in the process to be designed. Accordingly, the methodology presented in chapter 3 is reformulated by incorporating this new robustness test in the method's mathematical formulation.

This chapter is organized as follows: Section 4.1 introduces the new robust variability measure that has been proposed to evaluate the worst-case output variability. Based on the new robustness tests, Section 4.2 presents the new mathematical formulation and the iterative procedure proposed to address the integration of design and control problem. In Section 4.3, the new methodology description is applied to the simultaneous design and control of the mixing tank process that was previously studied in Section 3.2. Comparisons between the γ-based methodology, Mohideen's methodology[16] and the μ analysis-based (μ-based) methodology introduced in Section 4.2 are also discussed in this section. In addition, Section

4.4 presents the analysis made on the computational burden required by the μ-based method presented in Section 4.2 and those required by a dynamic programming-based methodology.

4.1 Robust variability measure

A key obstacle for integrating design and control is the search for the time-dependent disturbance profile that produces the largest variability in the system. This condition is the most numerical demanding task to perform within the optimization problem to achieve simultaneous design and control of a system. The goal is to find the critical time-dependent profile in the disturbance variables (**υ**) that produces the largest variability of the output variable, **y**. This can be mathematically expressed as follows:

$$\max_{\upsilon_q \in \boldsymbol{\upsilon}} |y(t)| \tag{4.1}$$

This optimization problem does not have a formal analytical solution. The current integration of design and control methodologies reported in the literature has circumvented this problem by using two approaches:

i) The profile that produces the largest variability is assumed to be known a priori. That is, the disturbance dynamics are specified by a time-dependent function that does not necessarily represent the disturbance's critical transient behaviour. This limits the resulting design since there is no guarantee that other disturbance realizations may produce process constraints violations, e.g. Brengel and seider[14], Swartz[27], Gerhard et al.[33].

ii) The critical disturbance profile is estimated by dynamic programming. In this case, a search in the disturbance profile for a pre-specified time horizon is performed such that it produces the maximum variation in the process output variables. Although this strategy is general and has been the traditional approach followed for the worst-case scenario estimation, it requires an intensive computational effort. Thus, its application to large-scale processes is often limited by this fact, e.g. Mohideen et al.[16], Bansal et al.[20], Sakizlis et al.[23].

In the current research work, a new robustness measure that estimates a bound on the largest deviation of a process variable at any time *t* has been developed. The computation of the largest variability at the end of a batch crystallization process has been calculated in a previous study applying a Mixed SSV approach[57]. The present work expands upon that work by proposing a calculation of the variability for a continuous process over a predetermined time horizon. In principle, the calculation for continuous processes should be done for an infinite time horizon. Instead, a practical approximation consists of limiting the calculation to the output's settling time, *N*.

To evaluate the largest variability in a process variable, the continuous robust model given in (3.7), which describes the relationship between the disturbances and the output variable, is used to generate a discrete robust finite impulse response (FIR) model between these two variables as follows:

$$y(j) = \sum_{q=1}^{m} \sum_{i=0}^{j} \left[h_{iq} \delta \upsilon_q (j-i) + \delta h_{iq} \delta \upsilon_q (j-i) \right]; \quad 0 \le j \le N \tag{4.2}$$

This model, equivalent to the impulse response convolution model shown in (2.45), provides a complete characterization of the system's closed-loop dynamic behaviour. Since more than one disturbance can affect the process, the disturbance $\boldsymbol{\upsilon}$ is defined as follows:

$$\boldsymbol{\upsilon} = \left\{ \upsilon_q(t), \upsilon_q{}^{l}(t) \le \upsilon_q(t) \le \upsilon_q{}^{u}(t) \right\} \tag{4.3}$$

Where $\upsilon_q(t)$ denotes the value of the disturbance q at time t. For simplicity, this variable will be denoted as υ_q. The vector $\boldsymbol{\upsilon}$ has dimensions mx1, with m being the number of disturbances affecting the process. Based on the definition for $\boldsymbol{\upsilon}$, a maximal change ($\delta\upsilon_q$) with respect to the mean value is defined for each disturbance q using the upper and lower bounds specified in (4.3) as follows:

$$\delta \upsilon_q = \frac{\upsilon_q^l - \upsilon_q^u}{2} \tag{4.4}$$

Following (4.2), the nominal values and the uncertainties of the impulse response coefficients relating the output $\mathbf{y}$ to the disturbance υ_q, h_{iq} and δh_{iq} are respectively defined as follows:

$$h_{iq} = \frac{h_{iq}^{up} + h_{iq}^{low}}{2}; \quad \delta h_{iq} = \frac{h_{iq}^{up} - h_{iq}^{low}}{2} \tag{4.5}$$

where h_{iq}^{up} and h_{iq}^{low} represent the upper and lower bounds of the output variable **y** at each time step *i* for each disturbance υ_q.

The procedure to obtain (4.2) from (3.7) is illustrated in Figure 4.1 and is described as follows: A unit pulse is is imposed on each υ_q to simulate the robust state space model (3.7). Since the simulation of the robust state space model would require an infinite number of runs, due to the uncertain state space matrices, the relaxation theorem principle given in equation (2.7) is applied to reduce the uncertain state space model to a finite set of FIR models. This means that only the vertexes of the uncertain parameter box formed with the extreme values of each of the models uncertain parameters shown in (3.7) are used to limit the calculation to a finite set of responses, i.e. the robust model (3.7) is assumed to be a polytopic model. Since the simulation of a continuos state space model to a unit pulse will produce an infinite number of impulse response coefficients, each of the state space models considered in the polytope is discretized using a First Order Hold (FOH) to obtain a finite number of impulse response coefficients. As shown in Figure 4.1, the impulse response coefficients' lower and upper bounds, h_{iq}^{up} and h_{iq}^{low}, are obtained from the search of the minimum and maximum values among the family of impulse response coefficients at each time interval *i*. Then, the impulse response coefficients' nominal and uncertain values are calculated from (4.5).

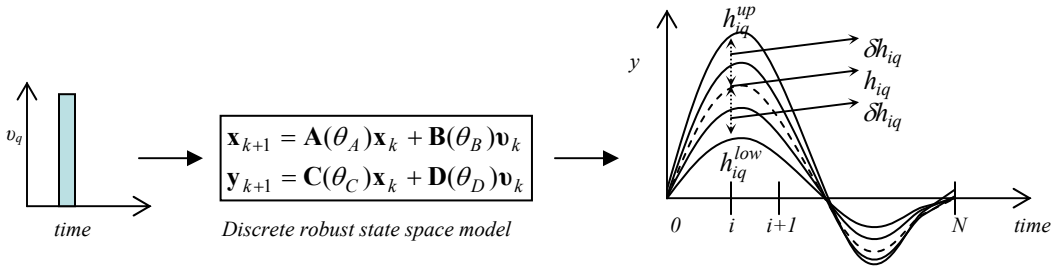

Figure 4.1 Illustrative procedure to obtain a robust FIR model.

Since the discrete robust FIR model (4.2) also represents the process closed-loop nonlinear dynamic behaviour from $\boldsymbol{\upsilon}$ to $\mathbf{y}$, the problem given by equation (4.1) can be reformulated as follows:

$$\max_{\delta\upsilon_q \in \boldsymbol{\delta\upsilon}} \left| \sum_{q=1}^{m} \sum_{i=0}^{j} \left[h_{iq} \delta\upsilon_q (j-i) + \delta h_{iq} \delta\upsilon_q (j-i) \right] \right| \tag{4.6}$$

Following the approach proposed by Nagy and Braatz[57], problem (4.1) can be rewritten in terms of the Mixed SSV (μ) for which upper and lower bounds for the output variable $\mathbf{y}$ can be estimated for a prediction horizon of N intervals. For any real k:

$$\max_{\delta\upsilon_q \in \boldsymbol{\delta\upsilon}} \left| \sum_{q=1}^{m} \sum_{i=0}^{j} \left[h_{iq} \delta\upsilon_q (j-i) + \delta h_{iq} \delta\upsilon_q (j-i) \right] \right| \geq k \Leftrightarrow \mu_{\boldsymbol{\Delta}}(\mathbf{M}) \geq k \tag{4.7}$$

The concept of the SSV analysis (μ) was presented in Section 2.2.4. Similarly, Figure 2.4 shows a graphical interpretation of the μ structure. Following (4.7), the largest output error of $\mathbf{y}$ at any time interval i along a time horizon of N intervals can then be found from the solution of a skewed-μ problem as follows[58]:

$$\max_{\mu_{\boldsymbol{\Delta}}(\mathbf{M}) \geq k} k \tag{4.8}$$

where the perturbation block $\boldsymbol{\Delta}$ used in the μ-analysis has the following structure:

$$\boldsymbol{\Delta} = diag\left(\boldsymbol{\Delta}_{r1}, \boldsymbol{\Delta}_{r2}, \boldsymbol{\delta}_c\right) \tag{4.9}$$

where $\boldsymbol{\Delta}_{r1}$ and $\boldsymbol{\Delta}_{r2}$ are independent real scalar column vectors of length $m\left(N+1+\sum_{j=1}^{N}(N-j+1)\right)$ which structure is defined as follows:

$$\boldsymbol{\Delta}_{r1} = \begin{bmatrix} \alpha_1 \mathbf{I}_{N+1,N+1} \\ \alpha_2 \mathbf{I}_{N,N} \\ \alpha_3 \mathbf{I}_{N-1,N-1} \\ \vdots \\ \alpha_{N+1} \end{bmatrix}; \quad \boldsymbol{\Delta}_{r2} = \begin{bmatrix} \beta_1 \mathbf{I}_{N+1,N+1} \\ \beta_2 \mathbf{I}_{N,N} \\ \beta_3 \mathbf{I}_{N-1,N-1} \\ \vdots \\ \beta_{N+1} \end{bmatrix} \tag{4.10}$$

where the α's and β's are real scalars values obtained from the solution of the μ-analysis defined in the RHS of (4.7). These values define the time-dependent critical disturbance profile. In addition, δ_c in (4.9) is a complex scalar column vector of length N+1. The interconnection matrix **M** used in (4.8) has the following structure:

$$\mathbf{M} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & k\left[\mathbf{W}_1,\ldots,\mathbf{W}_q,\ldots,\mathbf{W}_m\right]^T \\ \mathbf{S} & \mathbf{0} & \mathbf{0} \\ \left[\mathbf{H}_1,\ldots,\mathbf{H}_q,\ldots,\mathbf{H}_m\right] & \left[\mathbf{R}_1,\ldots,\mathbf{R}_q,\ldots,\mathbf{R}_m\right] & \mathbf{0} \end{bmatrix} \tag{4.11}$$

where each of the matrices specified in (4.11) is defined as follows:

$$\mathbf{W}_q = \begin{bmatrix} & & & & \delta\upsilon_q(0)\mathbf{I}_{N+1,N+1} \\ \mathbf{0} & & & & \delta\upsilon_q(1)\mathbf{I}_{N,N} \\ \vdots & \ddots & & & \vdots \\ \mathbf{0} & \ldots & \mathbf{0} & & \delta\upsilon_q(p)\mathbf{I}_{N-p+1,N-p+1} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{0} & \ldots & \mathbf{0} & \ldots\ \mathbf{0} & \delta\upsilon_q(N) \end{bmatrix} \tag{4.12}$$

$$\mathbf{R}_q = \begin{bmatrix} & & & & \delta h_{0,q}\mathbf{I}_{N+1,N+1} \\ \mathbf{0} & & & & \delta h_{1,q}\mathbf{I}_{N,N} \\ \vdots & \ddots & & & \vdots \\ \mathbf{0} & \ldots & \mathbf{0} & & \delta h_{p,q}\mathbf{I}_{N-p+1,N-p+1} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{0} & \ldots & \mathbf{0} & \ldots\ \mathbf{0} & \delta h_{N,q} \end{bmatrix} \tag{4.13}$$

$$\mathbf{H}_q = \begin{bmatrix} diag\left(h_{0,q},h_{1,q},\ldots,h_{j,q},\ldots,h_{N,q}\right) \\ \mathbf{0} \quad diag\left(h_{0,q},\ldots,h_{N-1,q}\right) \\ \vdots \quad \ddots \\ \mathbf{0} \quad \ldots \quad \mathbf{0} \quad diag\left(h_{0,q},\ldots,h_{N-p,q}\right) \\ \vdots \quad \ddots \quad \vdots \quad \ddots \\ \mathbf{0} \quad \ldots \quad \mathbf{0} \quad \ldots \quad \mathbf{0} \quad h_{0,q} \end{bmatrix}^T \tag{4.14}$$

$$\mathbf{S} = diag\left(\mathbf{T},\ldots,\mathbf{T}\right), \quad for \quad q = 1,\ldots m \tag{4.15}$$

$$\mathbf{T}=\begin{bmatrix} & & & & & \mathbf{B} \\ \mathbf{0} & & & & & \mathbf{B}_1 \\ \vdots & \ddots & & & & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & & & \mathbf{B}_p \\ \vdots & \ddots & \vdots & \ddots & & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \dots & \mathbf{0} & \mathbf{B}_{N+1} \end{bmatrix} \tag{4.16}$$

where the elements of **B** are defined as follows:

$$B_{i,j}=\begin{cases} k & if \begin{cases} i=1,2,3,\dots,N+1 \\ j=1,\dots,\sum_{l=1}^{i-1}\left(N+2-l\right)+1,\dots,\sum_{l=1}^{N}\left(N+2-l\right)+1 \end{cases} \\ 0 & otherwise \end{cases} \tag{4.17}$$

From (4.17), **B** has dimensions $\left(N+1\right)\times\left(\sum_{l=1}^{N}\left(N+2-l\right)+1\right)$. The matrix $\mathbf{B}_1$ in (4.16) consists of the first $\left(N-1\right)\times\left(N-1\right)$ rows and columns of matrix **B**; likewise, $\mathbf{B}_p$ in (4.16) consists of the first $\left(N-p\right)\times\left(N-p\right)$ rows and columns of **B**. An example for the construction of the interconnection matrix **M** for a simple SISO problem for a 3 steps-horizon FIR model is given in Appendix A.

The perturbation matrix shown in (4.9) takes into account the uncertainty associated with the maximal change of the disturbance variables ($\mathbf{\Delta}_{r1}$) and the impulse response coefficients ($\mathbf{\Delta}_{r2}$), respectively. The last N rows of the interconnection matrix (4.11) correspond to the different values of $y(j)$ along the time horizon defined between $0 \le j \le N$.

The problem defined in (4.8) is the robust measure test that is used to estimate a bound on the largest process output variability with respect to a nominal steady-state operating condition. The solution to (4.8) given in terms of the scalars α's and β's in (4.10) also provides the worst-disturbance vector **δυ** that produces such extreme deviation, i.e. the critical time-dependent disturbance profile is also obtained from this calculation.

To illustrate the application of the robust variability measure introduced in this section, the following robust state space model is considered:

$$\dot{\mathbf{x}} = \begin{bmatrix} -0.325 \pm 0.05 & 1 \\ -0.890 \pm 0.03 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0.8 \pm 0.01 \\ 0 \end{bmatrix} \boldsymbol{\upsilon}$$
$$\mathbf{y} = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \end{bmatrix} \boldsymbol{\upsilon} \tag{4.18}$$

To evaluate the robust measure defined in (4.8), the model in (3.7) must be transformed into a robust FIR model following the procedure described above, i.e. a unit pulse is used to simulate the robust model (4.18). Since the robust state space model parameters are uncertain, only the extreme values of the uncertain model parameters are used to generate a family of linear impulse response models. Similarly, each of the resulting linear state space models is discretized with a FOH to generate a finite number of impulse response coefficients. Thus, the system shown in (4.18) produces a set of eight linear impulse models shown in Figure 4.2. The impulse response coefficient's nominal values with their corresponding uncertainty bounds, h_{iq} and δh_{iq} respectively, are obtained from the search of the minimum and maximum values among the set of impulse response coefficients at each time interval. Given h_{iq}, δh_{iq} and the maximum change in the input, $\delta \upsilon$, set to unity for this illustrative example, i.e. $|\delta \upsilon| < 1$, the interconnection matrix **M** given in (4.11) and the structure of the perturbation matrix **Δ** shown in (4.9) can be constructed for this problem. Thus, the formulation presented in (4.8) can be applied to estimate a bound on the largest variability for model output **y**. This problem was formulated in MATLAB® using Sequential Quadratic Programming[55-54].

Figure 4.3 shows the critical time-dependent profile in the input variable **υ** that produces the largest variation in the model output, **y**, obtained from the solution of problem (4.8). This profile is not intuitive and a significant computational effort would have been required for finding it from trial and error simulations. To test the bound obtained from (4.8) for this system (k=14.3602), the worst-case input profile is used to simulate the robust state space model given in (4.18). Figure 4.4 shows the results of such simulations. To be consistent with the analysis, the extreme values specified for the uncertain model parameters are used to generate a finite set of system's responses as per the procedure described above. As it is shown in Figure 4.4, the different combinations of the actual system's responses do not exceed the bound that has been estimated by this technique. Since the uncertainty structure considered in this problem only has one decision variable, which it is also the parameter to be

maximized (k), then a graphical method can also be used to find the bound. Figure 4.5 shows the feasible region in which the maximum value for the parameter k is sought. As also shown in Figure 4.5, the constraint that appears in equation (4.8), i.e. $\mu_{\mathbf{\Delta}}(\mathbf{M})\geq k$, delimits the feasible search space for this problem. Figure 4.5 also shows that this problem only has one single solution. The convexity in this problem is due to the particular structure defined for the matrices $\mathbf{M}$ and $\mathbf{\Delta}$ and it does not depend on the uncertain state space model parameters. Thus, a *global maximum* is always obtained from the solution of (4.8). This property of problem (4.8) is relevant since the worst-case variability is usually obtained from dynamic programming that is subject to local solutions.

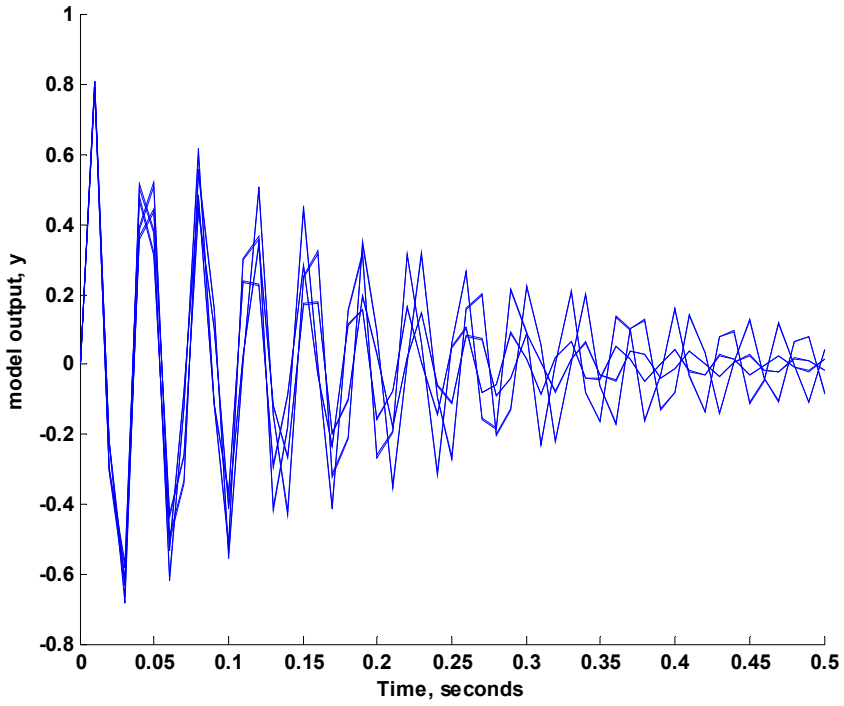

Figure 4.2 Illustrative example: Impulse response model.

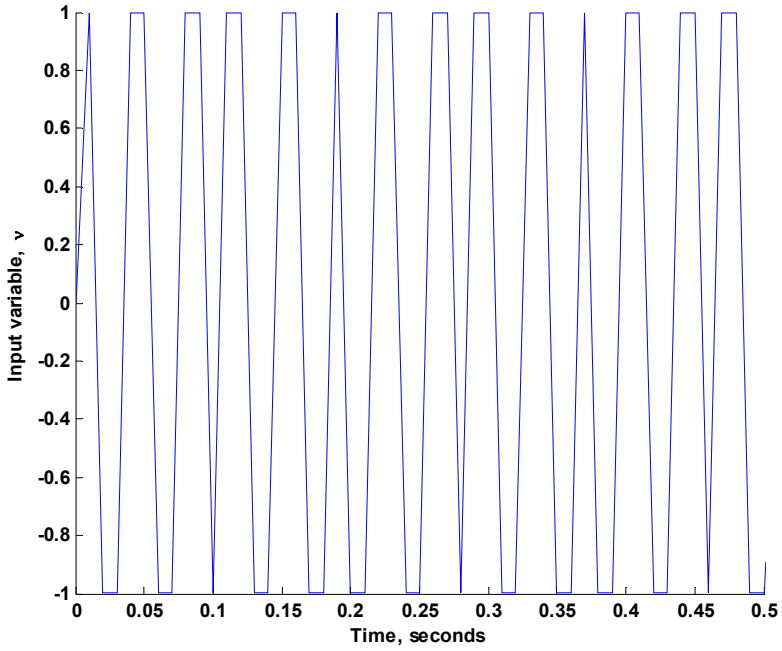

Figure 4.3 Illustrative example: Worst-case disturbance profile.

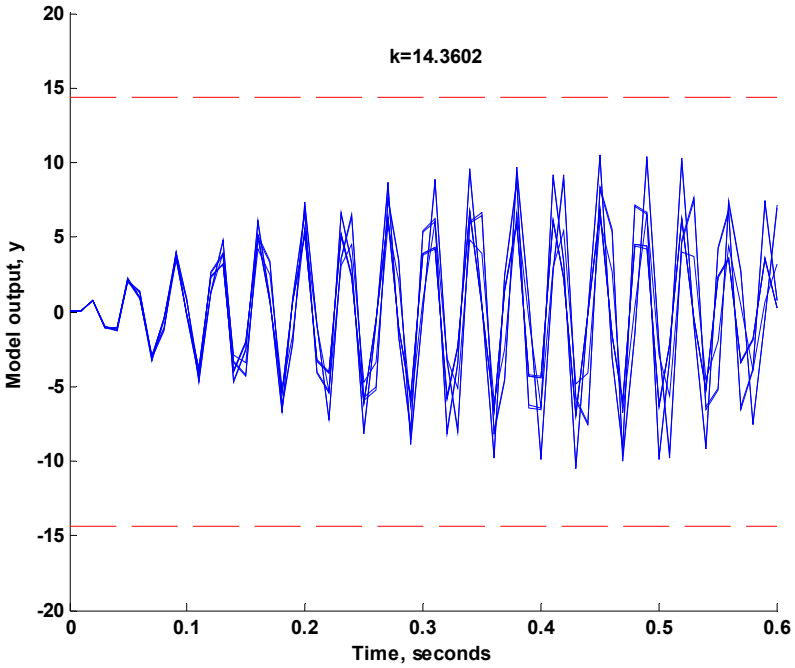

Figure 4.4 Illustrative example: Worst-case process output variability.

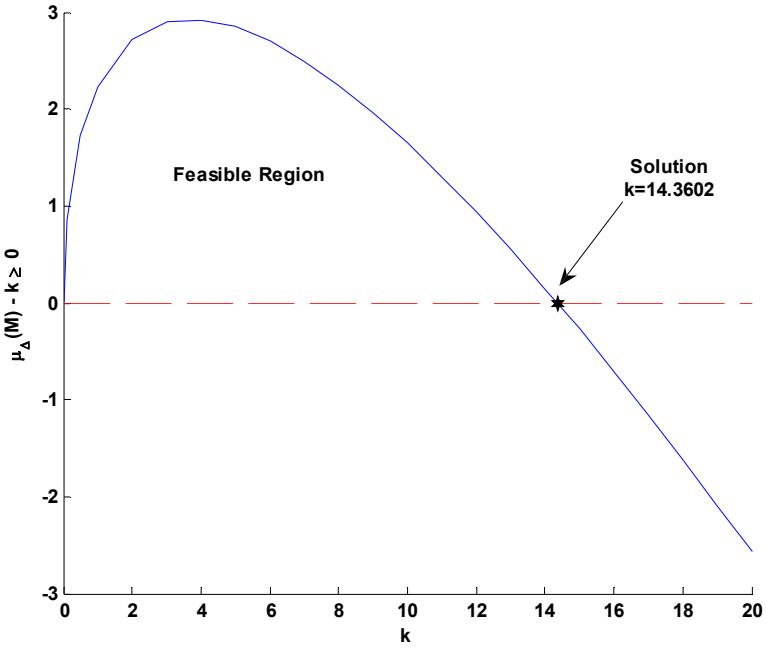

Figure 4.5 Illustrative example: Feasible region.

4.2 Simultaneous design and control methodology upgraded

The robust measure introduced in the previous section can be used to estimate a bound on the process output variability and to test the process feasibility constraints considered in the process to be designed. Thus, the mathematical description presented in (3.24) has to be modified accordingly. The next subsections present the new formulation that has been incorporated within the methodology framework to evaluate the worst-case process variability scenario and to test the process feasibility.

4.2.1 Assessment of the worst-case process variability scenario

The robust variability measure proposed in (4.8) requires the identification of a robust FIR model which can be obtained from a robust state space model of the closed loop system. As it is pointed out in Section 3.1.2, the uncertain model (3.7) is only valid in the vicinity of a specified operating state, defined by a nominal steady-state value in the manipulated variables to be used by the control algorithms, $\overline{\mathbf{u}}$, and the output variables, $\overline{\mathbf{w}}$, and for a fixed value in the adjustable variables (**p**), the design variables (**d**), the controller tuning

parameters ($\boldsymbol{\lambda}$) and the process parameter uncertainty ($\boldsymbol{\omega}$). Thus, the solution to (4.8) only provides a bound on the process output's largest variability in a neighbourhood of the operating state. The vectors $\overline{\mathbf{u}}$, $\overline{\mathbf{w}}$, $\mathbf{p}$, $\mathbf{d}$ and $\boldsymbol{\lambda}$ are considered to be decision variables in the methodology's formulation. On the other hand, $\boldsymbol{\omega}$ is assumed to be an uncertain parameter which true value is unknown but it ranges between extreme values and it remains at a constant value for long periods of time. Therefore, one must search for the critical profile in the disturbance $\boldsymbol{\delta\upsilon}$, which is the input in (3.7) and (4.2), and the corresponding steady-state value in $\boldsymbol{\omega}$ that produces the largest error in $\mathbf{y}$.

The methodology presented in chapter 3 estimates the largest process variability error based on a QLF according to the problem given by equation (3.22). This problem is now reformulated in terms of the new robust variability measure as follows:

$$\begin{aligned} &\max_{\omega^l \le \omega \le \omega^u} \phi^d\left(\overline{\mathbf{u}}, \mathbf{p}, \overline{\mathbf{w}}, \mathbf{c}, \mathbf{d}, \boldsymbol{\lambda}, k^d\right) \\ &s.t. \\ &\quad \max_{\mu_{\Delta^d}\left(\mathbf{M}^d\right) \ge k^d} k^d \end{aligned} \tag{4.19}$$

where the superscript d is only used to indicate that this calculation referrers to the worst-case process variability scenario. The solution to (3.22) provides a bound on the largest process variability (ϕ^d) and the critical time-dependent profiles of $\boldsymbol{\delta\upsilon}$ and $\boldsymbol{\omega}$ that produces such condition. Solving (3.22) requires the identification of a system like (3.7) and (4.2) for every $\boldsymbol{\omega}$ tested. However, the identification process is simple and, most importantly, the identified models are expected to have small model uncertainties since the process is identified in closed-loop and it consequently remains in a small neighbourhood of the nominal operating point.

Once problem (3.22) is being solved, a cost can be assigned to the maximum process variability in order to quantify its economic impact on the operation at an operating state defined by $\overline{\mathbf{u}}$, $\overline{\mathbf{w}}$, $\mathbf{p}$, $\mathbf{d}$ and $\boldsymbol{\lambda}$. This cost, referred to as the process variability cost, appears in the methodology's cost function defined in (3.6).

4.2.2 Process Feasibility

Process feasibility requires that the manipulated variables **u** or the process output variables **w** must remain within specified bounds despite any combination of **υ** and **ω**. The largest variability in **u** or **w** can be obtained by using the approach proposed in the previous section to solve the worst-case process variability scenario. Thus, problems (4.8) and (3.22) can be rewritten in terms of **u** and **w** as follows:

$$\begin{aligned} \overline{u} \pm \max_{\omega^l \le \omega' \le \omega^u} \left(\max_{\mu_{\Delta^u}(\mathbf{M}^u) \ge k^u} k^u \right) \le \rho_1 \\ \overline{w} \pm \max_{\omega^l \le \omega'' \le \omega^u} \left(\max_{\mu_{\Delta^w}(\mathbf{M}^w) \ge k^w} k^w \right) \le \rho_2 \end{aligned} \tag{4.20}$$

where ρ_1 and ρ_2 are the corresponding input limits and $\overline{\mathbf{u}}$ and $\overline{\mathbf{w}}$ are the nominal steady state values of the manipulated variable and the process output variable, respectively. The terms $\overline{\mathbf{u}}$ and $\overline{\mathbf{w}}$ have been considered as decision variables in the methodology's optimization problem. In addition, the matrices $\mathbf{M}^u$, $\mathbf{M}^w$ and $\mathbf{\Delta}^u$, $\mathbf{\Delta}^w$ are the corresponding interconnection and perturbation matrices for each variable and are defined as in (4.9) and (4.11), respectively.

To evaluate (3.23), it is necessary to obtain robust state space models, and consequently robust FIR models, that relate the effect of the disturbance **υ** to the particular **u** and **w** that are to be kept within bounds. The apostrophes used to denote the process parameter uncertainty in equation (3.23) are only used to indicate the fact that critical realizations in **υ** and **ω** are particular for every process variable, i.e., the worst-case profile for $\overline{\mathbf{u}}$ may not be the same as for $\overline{\mathbf{w}}$. Thus, the optimization problems posed in (3.23) must be solved for each process variable that must remain within constraints.

Since the problems posed in (3.23) provides a bound on the worst-case variability in $\overline{\mathbf{u}}$, $\overline{\mathbf{w}}$ or any other process variable that it is desired to keep within bounds, the process feasibility test given in (3.23), which was defined in terms of QLF, is replaced by (3.23) to account for the largest variability on a variable that is subject to constraints.

4.2.3 Mathematical Formulation of the Methodology

Based on the descriptions given above, the basic mathematical formulation presented in (3.24) is modified to account for the new robust worst-case variability measure within the methodology. Accordingly, the process output variability and process feasibility problems specified in (3.22) and (3.23) are replaced within the methodology's formulation by problems (4.19) and (4.20), respectively. Following these modifications the optimization problem proposed to perform the simultaneous design and control of chemical processes under external perturbations and process parameter uncertainty given in (3.24) is reformulated as follows:

$$
\begin{aligned}
&\min_{\boldsymbol{\eta}=[\overline{\mathbf{u}},\mathbf{p},\overline{\mathbf{w}}_{cl},\overline{\mathbf{w}}_{ol},\mathbf{d},\boldsymbol{\lambda}]} CF = CC + OP + VC \qquad \text{…Cost Function}\\
&\text{s.t.}\\
&\quad \mathbf{A}(m_k)^T\mathbf{P} + \mathbf{P}\mathbf{A}(m_k) < \mathbf{0} \qquad \text{…Process stability test}\\
&\quad \max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega} \le \boldsymbol{\omega}^u} \phi^d\left(\overline{\mathbf{u}},\mathbf{p},\overline{\mathbf{w}},\mathbf{c},\mathbf{d},\boldsymbol{\lambda},k^d\right) \quad \text{…Process output variability}\\
&\quad \text{s.t.}\\
&\qquad \max_{\mu_{\boldsymbol{\Delta}^d}(\mathbf{M}^d)\ge k^d} k^d\\
&\qquad \overline{g} \pm \max_{\omega^l \le \omega' \le \omega^u}\left(\max_{\mu_{\boldsymbol{\Delta}^g}(\mathbf{M}^g)\ge k^g} k^g\right) \le \rho\\
&\quad \boldsymbol{\eta}^l \le \boldsymbol{\eta} \le \boldsymbol{\eta}^{\mathrm{u}} \qquad \text{…Process feasibility}\\
&\quad \text{Process Model Equations}\\
&\quad \text{Control Algorithm Equations}
\end{aligned}
\tag{4.21}
$$

The above formulation corresponds to a constrained nonlinear optimization problem. Within this formulation the process stability test remains the same as in (3.24). Similarly, the process model equations and the control algorithm equations are not explicitly shown in (4.21) but they are implicitly considered in the form of robust state space models, which are used for the worst-case process variability calculation and the process stability and feasibility tests. It should be emphasized that although the overall problem is a constrained nonlinear optimization with many possible minima, the calculations of the bounds based on the structured singular value index are convex and have a global minimum.

The iterative procedure proposed to solve (4.21) is similar to that described in Section 3.1.6 for the basic methodology formulation. The only change that must be considered for solving (4.21) is the identification of a robust FIR model like (4.2), needed to estimate the worst-case process variability and to test process feasibility. These models are obtained from the robust state space models (3.7) using the procedure outlined in Section 4.1 and illustrated in Figure 4.1.

Similar to the assumed for the γ-based methodology formulation presented in the previous chapter, the mathematical description specified in (4.21) assumes that the process flowsheet and the control structure have been fixed a priori. This logically limits the degrees of freedom in the system and thus the process optimal design. However, the present approach can be expanded to consider those analyses within the design by embedding (4.21) into a control structure and flowsheet selection algorithm. Although a better design may be obtained, the complexity and the computational time involved in the estimation of the design are increased since integer variables would have to be added to the formulation. This has not been considered in the present work and it is left for future studies.

4.3 Testing the μ-based methodology: mixing tank process revisited

The methodology presented in the last section was applied to the simultaneous design and control of the mixing tank process, which was also used to test the γ–based methodology (see Section 3.2). This process was chosen because its process model is relatively simple and the resulting design parameters obtained by the μ-based method can be compared to those obtained by the γ–based methodology and dynamic programming-based methodology proposed by Mohideen et al.[16].

The mixing tank process flowsheet philosophy has been explained in Section 3.2 and it is schematically represented in Figure 3.1. Likewise, the process model and the operating parameter values for this system are given in Table 3.1.

The design goals proposed for this case study are the same as in Section 3.2, that is, to seek for the nominal steady-state values in the manipulated variables used for control, $\overline{\mathbf{u}}$, and the process outputs, $\overline{\mathbf{w}}$, the values in the design variables (**d**), the manually adjustable variables

(**p**), and the controller tuning parameters (**λ**) that minimize the tank's total cost function. The resulting design must remain asymptotically stable and within constraints in the presence of magnitude bounded changes in the hot stream flow rate, F_h, and temperature, T_h, considered as a process parameter uncertainty (**ω**) and a disturbance (**υ**), respectively.

As in Section 3.2, the two PI-based control schemes that were used to test the γ–based methodology are also used here to test the μ-based methodology. The scenarios are: *Scenario I*-Temperature control and *Scenario II*-Temperature and volume control. The results of the application of the methodology to these scenarios are presented in the following sub-sections.

4.3.1 Scenario I-Temperature control

The first control scheme proposed to the simultaneous design and control of the mixing tank process consists in controlling the tank's temperature using a Proportional-Integral (PI) controller. This controller would make the necessary adjustments in the cold feed flow rate, F_c, to maintain the tank's temperature, $T(t)$, within its pre-specified bounds despite any variations that may occur in the process hot stream, i.e. F_h and T_h. In addition, the tank's volume is not being controlled and the other potential variable that can be used for control purposes, the stem position of the valve at the outlet, z, is assumed to be manually adjusted and accordingly is viewed as a system parameter (**p**).

Cost function.- Since the single design goal is to minimize the tank's total cost, the cost function for this system can be directly estimated from the tank's final size or volume, V_d. Also, since bounded disturbances and process parametric uncertainty are assumed to be affecting the process, the tank's final volume is based on a nominal steady state volume plus the variability of this variable due to the effect of the unmeasured perturbations entering the process. Consequently, the capital cost for this process is defined as the total volume of the tank that is equal to the steady state value plus the maximum variability in volume, that is:

$$V_d = \left(V_{nom} + k^V\right) \tag{4.22}$$

The nominal or steady state volume V_{nom} depends on factors such as the outlet valve's stem position, z, and tank's nominal temperature, $T(t)$, whereas the variability with respect to this

nominal volume (k^V) depends on both, the nominal volume around which the maximum variability is estimated and the controller tuning parameters that determine the process closed loop dynamic performance. V_d denotes the tank's total size and it is considered the design variable (**d**) for this problem. The term V_{nom} represents the tank's nominal volume at a steady state condition; since the design goals do not specify a value for this term, it is considered as an optimization variable within *Scenario-I*'s formulation. The term k^V represents a bound on the tank's volume maximum variability due to the process closed-loop dynamic performance. Due to the design goals specifications outlined above for this particular case study, the capital cost given by (4.22) is also equal to the process variability function, ϕ^d, which maximum can be estimated from the problem defined in (4.19). In other more general cases such as the Tennessee Eastman problem, the capital cost function is different from the process variability function.

Closed-loop Process Model.- The process model equations shown in Table 3.1 and the control algorithm equations given in (3.26) and (3.27) represent the closed-loop nonlinear model equations specified for *Scenario I*. These equations are used to simulate the mixing tank process dynamic behaviour due to changes in the hot stream temperature, T_h, and around a given steady-state value of the hot stream flow rate, F_h. The input/output data collected from this simulation is then used to estimate robust state space models of the type given by (3.7). Then, as explained in the previous section, the robust performance and process feasibility tests require the transformation of the continuous uncertain state space model into a discrete robust FIR model. Since the design goals only specifies an operating range for the tank's temperature, i.e. no set point has been defined for this variable, the tank's temperature set point, T_{sp}, is treated as a decision variable in the optimization problem defined for *Scenario I*.

Process Stability.- The robust stability criterion used by the present method has been given in (3.12). To evaluate the asymptotic stability at a given operating state, robust models from T_h to the state variables, $V(t)$ and $T(t)$, have to be identified. The procedure used to identify such models has been given in Section 3.1.2 and Section 3.1.6, respectively.

Worst-case process variability scenario.- The process variability function for *Scenario I* has been defined in (4.22). To estimate the cost associated with the process closed-loop dynamic performance, it is required to calculate the maximum expected variability in the process. This condition is estimated for *Scenario I* using the robust variability measure introduced in Section 4.1 and that has been incorporated within the worst-case variability formulation as shown in (4.19). Accordingly, the worst-case process variability search for *Scenario I* can be stated as follows:

$$\begin{aligned} &\max_{F_h^l \le F_h \le F_h^u} V_d = \left(V_{nom} + k^V\right) \\ &s.t. \\ &\quad \max_{\mu_{\Delta^V}\left(\mathbf{M}^V\right) \ge k^V} k^V \end{aligned} \tag{4.23}$$

where the term k^V provides an upper bound on the maximum volume variability with respect to a nominal steady state volume denoted by V_{nom}. The evaluation of the inner maximization problem on the parameter k^V in (4.23) requires the identification of a discrete robust FIR model. Thus, robust state space models in the form of (3.7) have to be identified first from T_h to $V(t)$ for each hot feed flow rate (F_h) tested and at a given operating state. Then, this family of linear state space models are simulated and discretized to obtain a discrete robust FIR model (4.2) following the procedure outlined in Section 4.1 and illustrated in Figure 4.1.

Process Feasibility.- To guarantee that the tank's temperature remains within the specified upper and lower pre-specified bounds for the entire period of operation, the same approach considered to determine the largest process variability in the tank's volume is used to test compliance with the process feasibility constraints. Thus, problem (4.20) is rewritten in terms of the temperature in the tank as follows:

$$\begin{aligned} &T_{sp} - T^u + \max_{F_h^l \le F_h' \le F_h^u} \left(\max_{\mu_{\Delta^T}\left(\mathbf{M}^T\right) \ge k^T} k^T \right) \le 0 \\ &T^l - T_{sp} + \max_{F_h^l \le F_h' \le F_h^u} \left(\max_{\mu_{\Delta^T}\left(\mathbf{M}^T\right) \ge k^T} k^T \right) \le 0 \end{aligned} \tag{4.24}$$

where T^l and T^u represents the lower and upper limits defined for the tank's temperature given in Table 3.1.

To solve the maximization problem specified in (4.24), discrete robust FIR models like the ones described by equation (4.2) are needed for each F_h tested at a given operating state. To obtain such models, a robust state space model like (3.7) must be identified first from the process hot stream temperature T_h to the tank's temperature $T(t)$. Then, the uncertain state space model is simulated and discretized following the procedure outlined in Section 4.1 to obtain the robust convolution model description shown in (4.2).Similar expressions to (4.24) are derived to evaluate the rest of the process constraints defined for *Scenario I*, i.e. constraints on the cold feed flow rate (F_c) and the tank's minimal allowed volume. Based on the above descriptions, the overall optimization problem defined in (4.21) is stated for *Scenario I* in (4.25). The decision variables for this problem are the PI controller tuning parameters, Kc_T and τ_{IT}, the temperature set point, T_{sp}, and the outlet valve's stem position, z. The tank's nominal volume and the nominal cold feed flow rate, V_{nom} and $F_{c,nom}$, are obtained from the steady state solution of the process model equations for each new set $\boldsymbol{\eta}$ tested. The optimization problem shown in (4.25) was coded and executed in MATLAB® using Sequential Quadratic Programming[55-54]. The resulting design parameters are shown in 4.1. The results show that tank's total volume necessary to accommodate the constraints and the variability in level is relatively large. The final tank's size was expected since the state variable, $V(t)$, is not under closed-loop control.

Table 4.1 mixing tank's design, *Scenario I*, μ-based methodology.

Variables	Solution
V_d	33.77 m^3
Z	0.3398
T_{sp}	305.25 K
$F_{c,nom}$	1.3390 m^3/hr
Kc_T	-0.6764
τ_{IT}	7.13 hr

$$
\begin{aligned}
&\min_{\boldsymbol{\eta}=\left[Kc_T,\tau_{IT},z,T_{sp}\right]} \quad V_d && \text{…Cost Function}\\
&s.t.\\
&\quad \mathbf{A}_V\left(m_k\right)^T\mathbf{P}+\mathbf{P}\mathbf{A}_V\left(m_k\right)<\mathbf{0}\\
&\quad \mathbf{A}_T\left(m_k\right)^T\mathbf{P}+\mathbf{P}\mathbf{A}_T\left(m_k\right)<\mathbf{0} && \text{…Process stability test}\\
&\quad \max_{F_h^l\le F_h\le F_h^u}\left(V_{nom}+\max_{\mu_{\Delta^V}\left(\mathbf{M}^V\right)\ge k^V}k^V\right) && \text{…Worst - case scenario evaluation}\\
&\quad V^l-V_{nom}+\max_{F_h^l\le F_h\le F_h^u}\left(\max_{\mu_{\Delta^V}\left(\mathbf{M}^V\right)\ge k^V}k^V\right)\le 0\\
&\quad T_{sp}-T^u+\max_{F_h^l\le F_h^1\le F_h^u}\left(\max_{\mu_{\Delta^T}\left(\mathbf{M}^T\right)\ge k^T}k^T\right)\le 0\\
&\quad T^l-T_{sp}+\max_{F_h^l\le F_h^1\le F_h^u}\left(\max_{\mu_{\Delta^T}\left(\mathbf{M}^T\right)\ge k^T}k^T\right)\le 0\\
&\quad F_{c,nom}-F_c^u+\max_{F_h^l\le F_h^2\le F_h^u}\left(\max_{\mu_{\Delta^{F_c}}\left(\mathbf{M}^{F_c}\right)\ge k^{F_c}}k^{F_c}\right)\le 0\\
&\quad F_c^l-F_{c,nom}+\max_{F_h^l\le F_h^2\le F_h^u}\left(\max_{\mu_{\Delta^{F_c}}\left(\mathbf{M}^{F_c}\right)\ge k^{F_c}}k^{F_c}\right)\le 0\\
&\quad \boldsymbol{\eta}^{\mathbf{l}}\le\boldsymbol{\eta}\le\boldsymbol{\eta}^{\mathbf{u}} && \text{…Process feasibility constraints}\\
&\quad \dot{V}(t)=F_h+F_c-zV^{1/2}\\
&\quad \dot{T}(t)=\frac{\left[F_h\left(T_h(t)-T(t)\right)+F_c(t)\left(T_c-T(t)\right)\right]}{V(t)} && \text{…Process model equations *}\\
&\quad \dot{\xi}_T=\frac{1}{\tau_{IT}}e_T\\
&\quad F_c=F_{c,nom}+Kc_T\left(e_T+\xi_T\right)\\
&\quad e_T=T_{sp}-T(t) && \text{…Control algorithm equations *}
\end{aligned}
\tag{4.25}
$$

The resulting design obtained from this formulation can be compared to that obtained from the QLF formulation given in Table 3.2. As it is shown, although the present formulation estimates a bound on the largest error for each process variable considered in the formulation, it results in a smaller tank's size than in the case in which a bound on the variables error's variance is used as a robustness test for variability and feasibility, i.e. using the γ-based methodology presented in the previous chapter. This indicates that the μ-based robust variability measure, introduced in this chapter provides a tighter bound than those obtained by the variability index γ, i.e. it provides a less conservative measure.

It is has been reported in the literature that the γ calculation based on the QLF may produce conservative bounds[59-61]. The problem proposed for the estimation of γ shown in (3.19) must search for a single matrix weighting matrix **P** that satisfies all the LMI constraints that appears due to all the possible combinations between the extreme values of the uncertain state space model parameters. The γ index provides, for all these combinations, a bound according to inequality (3.14) given in chapter 3. According to this inequality, the bound must be satisfied for *all* the possible rate of changes of the Lyapunov function $\mathbf{V}(\mathbf{x})$ represented by the derivative term in that inequality.

On the other hand, the μ-based robust variability measure is not subject to the computation of an additional independent function like the Lyapunov matrix **P** needed in the γ calculation and does not depend on the possible rates of change of the $\mathbf{V}(\mathbf{x})$ with time. Thus, it is expected that this measure provides a tighter bound on the output error's largest variability. To summarize, the addition of the system-independent Lyapunov matrix **P** within the robust variability calculation adds conservatism in the solution of the robust variability problem.

To test the optimization results and the compliance with the process constraints, the resulting design values shown in Table 4.1 were used to simulate the process dynamic behaviour for more than 1 year of operation. As in the γ-based methodology test, the profile shown in Figure 3.3 was also used to represent the disturbance dynamic behaviour (T_h) with the hot feed flow rate (F_h) fixed to each one of its extreme values shown in Table 3.1. The process was simulated using the process closed-loop nonlinear dynamic model equations shown in problem (4.25).Figure 4.6 through Figure 4.8 present the results of this simulation. As shown, the tank's actual volume never exceeds the tank's total size, V_d, obtained by the present method, whereas the tank's temperature, $T(t)$, and the cold feed flow rate, F_c, vary within the limits specified for these variables. Therefore it can be concluded that the resulting design obtained by this approach remains stable and feasible in the presence of any bounded disturbance profile in T_h and when the system is operated around different values of the process parameter uncertainty, F_h.

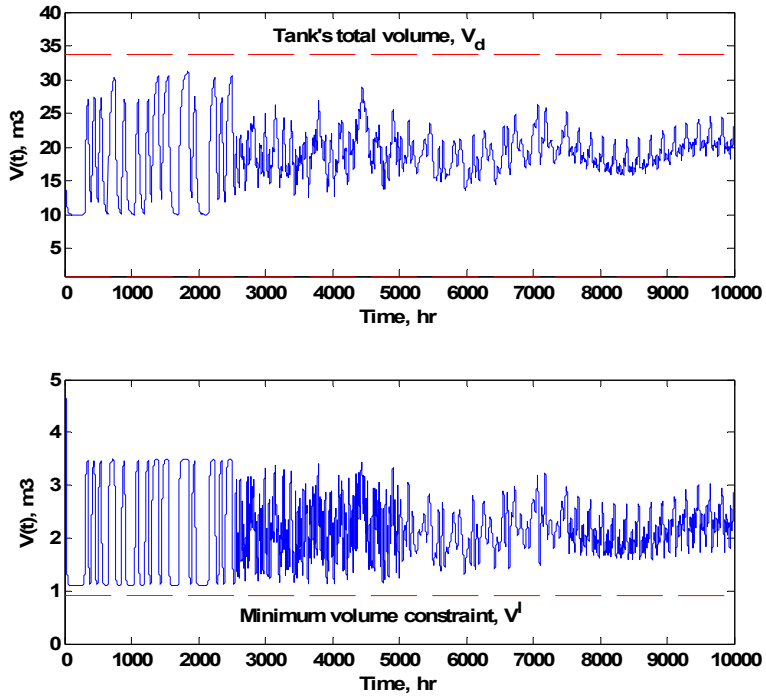

Figure 4.6 Tank's actual volume, *Scenario I*, μ-based methodology.

Top graph Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

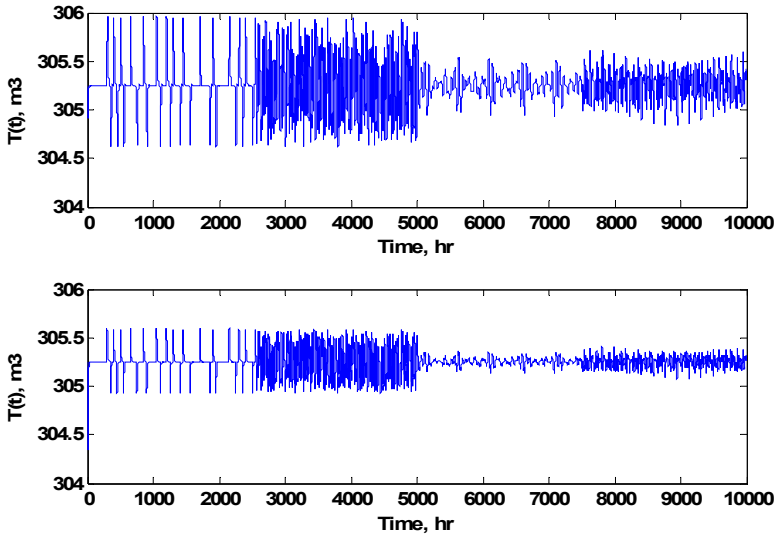

Figure 4.7 Tank's actual temperature, *Scenario I*, μ-based methodology.

Top graph Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

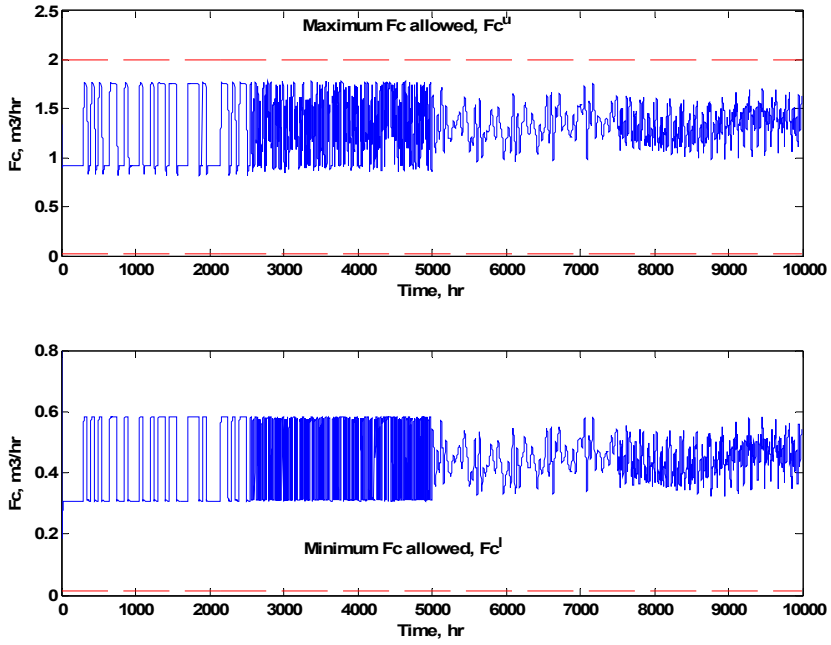

Figure 4.8 Cold Feed Flow rate's actual values, *Scenario I*, μ-based methodology.

Top graph Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

4.3.2 Scenario II-Temperature and Volume control

For this scenario, a second PI controller is incorporated within the process control scheme to regulate the tank's hold up variations in addition to the PI controller used to control temperature in *Scenario I*. The added volume controller makes adjustments in the stem position of the outlet valve, z, to attenuate the effect of the perturbations, T_h and F_h, on the tank's volume, $V(t)$. The closed-loop process model for *Scenario II* is defined by the process model and the control equations for temperature control shown in (4.25) together with the control equations for volume control given by (3.31) and (3.32). In this case however, the term V_{nom} shown in (4.25) is replaced by V_{sp} to indicate that the tank's volume under closed-loop control around a set point of volume to be obtained from the solution of the optimization problem. On the other hand, the outlet valve's stem position, z, becomes a manipulated variable to be used for control. Hence, constraints on this variable must be added to the methodology formulation to ensure that its variations are kept within the limits specified for this variable.

Thereby, *Scenario I*'s formulation given in (4.25) is expanded to account for the controller equations for volume control and the constraints on the outlet valve's stem position. The mathematical formulation defined for *Scenario II* is given in problem (4.26). The decision variables for this problem are the controller tuning parameters, Kc_T, τ_{IT}, Kc_V, τ_{IV}, and the controllers' set points, T_{sp} and V_{sp}.

To evaluate the robustness tests specified in (4.26), robust state space models must be identified between the disturbance variable, T_h, to both controlled variables, i.e. the tank's volume and temperature, $V(t)$ and $T(t)$, and the manipulated variables, i.e. the cold feed flow rate, F_c, and the outlet valve's stem position, z. These models are obtained from the simulation of the process closed-loop nonlinear dynamic model around a nominal operating state and for each F_h value tested by the inner optimization problems specified in (4.26). As in *Scenario I*, the values of $F_{c,nom}$ and z_{nom} must satisfy the process model steady-state equations for each new set of values of $\boldsymbol{\eta}$ tested. Problem (4.26) was also solved in MATLAB® using SQP [54-55]. The results obtained for this optimization problem are shown in Table 4.2. Due to fact that this scenario adds a controller to regulate the variations in the tank's volume, the variability on this variable was as expected largely reduced. Thus, the tank's minimum volume required to successfully reject the process hot stream perturbations is logically much smaller than the tank's volume obtained by *Scenario I*.

Table 4.2 mixing tank's design, *Scenario II*, μ-based methodology

Design Variables	Solution
V_d (m^3)	0.9278
V_{sp} (m^3)	0.9133
T_{sp} (K)	335.68
z_{nom}	0.2998
$F_{c,nom}$ (m^3/hr)	0.1366
Kc_T (K-hr/m^3)	-1.7122
Kc_V (m^3)	-16.3809
τ_{IT} (hr)	8.01
τ_{IV} (hr)	3.99

$$\min_{\boldsymbol{\eta}=\left[Kc_T,\tau_{IT},Kc_V,\tau_{IV},T_{sp},V_{sp}\right]} \quad V_d \qquad \text{…Cost Function}$$

s.t.

$$\mathbf{A}_V\left(m_k\right)^T\mathbf{P}+\mathbf{P}\mathbf{A}_V\left(m_k\right)<\mathbf{0}$$

$$\mathbf{A}_T\left(m_k\right)^T\mathbf{P}+\mathbf{P}\mathbf{A}_T\left(m_k\right)<\mathbf{0} \qquad \text{…Process stability test}$$

$$\max_{F_h^l\le F_h\le F_h^u}\left(V_{sp}+\max_{\mu_{\Delta^V}\left(\mathbf{M}^V\right)\ge k^V}k^V\right) \text{…Worst - case process variability evaluation}$$

$$V^l-V_{sp}+\max_{F_h^l\le F_h\le F_h^u}\left(\max_{\mu_{\Delta^V}\left(\mathbf{M}^V\right)\ge k^V}k^V\right)\le 0$$

$$T_{sp}-T^u+\max_{F_h^l\le F_h^1\le F_h^u}\left(\max_{\mu_{\Delta^T}\left(\mathbf{M}^T\right)\ge k^T}k^T\right)\le 0$$

$$T^l-T_{sp}+\max_{F_h^l\le F_h^1\le F_h^u}\left(\max_{\mu_{\Delta^T}\left(\mathbf{M}^T\right)\ge k^T}k^T\right)\le 0$$

$$F_{c,nom}-F_c^u+\max_{F_h^l\le F_h^2\le F_h^u}\left(\max_{\mu_{\Delta^{F_c}}\left(\mathbf{M}^{F_c}\right)\ge k^{F_c}}k^{F_c}\right)\le 0$$

$$F_c^l-F_{c,nom}+\max_{F_h^l\le F_h^2\le F_h^u}\left(\max_{\mu_{\Delta^{F_c}}\left(\mathbf{M}^{F_c}\right)\ge k^{F_c}}k^{F_c}\right)\le 0$$

$$z_{nom}-z^u+\max_{F_h^l\le F_h^3\le F_h^u}\left(\max_{\mu_{\Delta^z}\left(\mathbf{M}^z\right)\ge k^z}k^z\right)\le 0$$

$$z^l-z_{nom}+\max_{F_h^l\le F_h^3\le F_h^u}\left(\max_{\mu_{\Delta^z}\left(\mathbf{M}^z\right)\ge k^z}k^z\right)\le 0$$

$$\boldsymbol{\eta}^l\le\boldsymbol{\eta}\le\boldsymbol{\eta}^u \qquad \text{…Process feasibility constraints}$$

$$\dot{V}(t)=F_h+F_c-zV^{1/2}$$

$$\dot{T}(t)=\frac{\left[F_h\left(T_h(t)-T(t)\right)+F_c(t)\left(T_c-T(t)\right)\right]}{V(t)} \qquad \text{…Process model equations *}$$

$$\dot{\xi}_T=\frac{1}{\tau_{IT}}e_T$$

$$F_c=F_{c,nom}+Kc_T\left(e_T+\xi_T\right)$$

$$e_T=T_{sp}-T(t)$$

$$\dot{\xi}_V=\frac{1}{\tau_{IV}}e_V$$

$$z=z_{nom}+Kc_V\left(e_V+\xi_V\right)$$

$$e_V=V_{sp}-V(t) \qquad \text{…Control algorithm equations *} \qquad (4.26)$$

To test the results shown in Table 4.2, the design accomplished by the μ-based method was simulated for more than 1 year of operation using the closed-loop process model equations shown in (4.26). As in *Scenario I*, the hot stream temperature variations are represented by the time-dependent profile shown in Figure 3.3. The simulations were carried out assuming that the process hot stream flow rate, F_h, is constant and equal to either one of its extreme values. Figure 4.9 through Figure 4.12 show the results of these simulations. As shown in Figure 4.9, the tank's total volume, V_d, is never exceeded by the tank's actual volume. Also, Figure 4.10 through Figure 3.11 show that process constraints defined in (4.26) are within the pre-specified limits despite the time-varying profile in T_h and for any given fixed value of F_h within the pre-specified range of this variable.

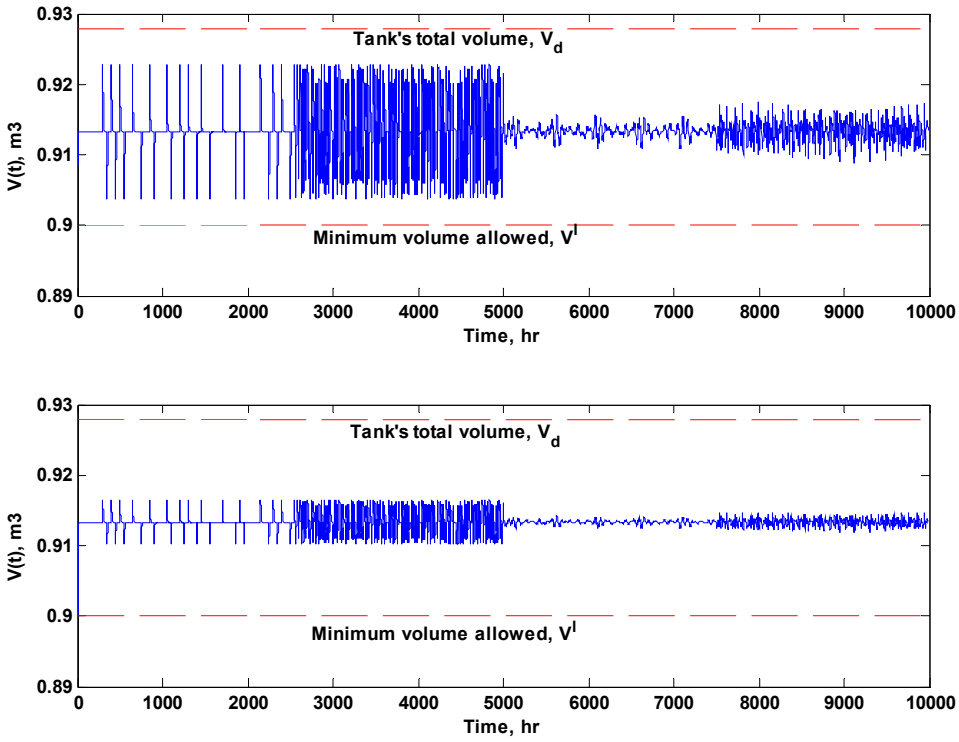

Figure 4.9 Mixing tank's actual volume, *Scenario II*, μ-based methodology.

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

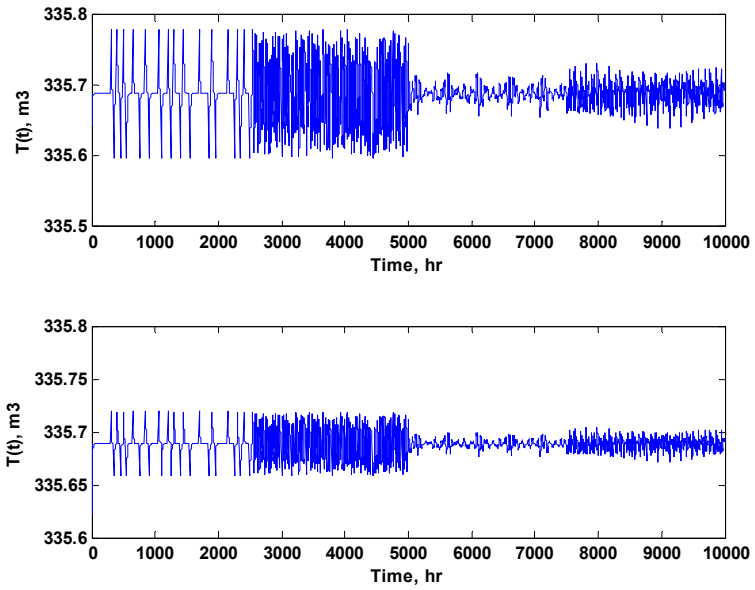

Figure 4.10 Mixing tank's actual temperature, *Scenario II*, μ-based methodology.

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

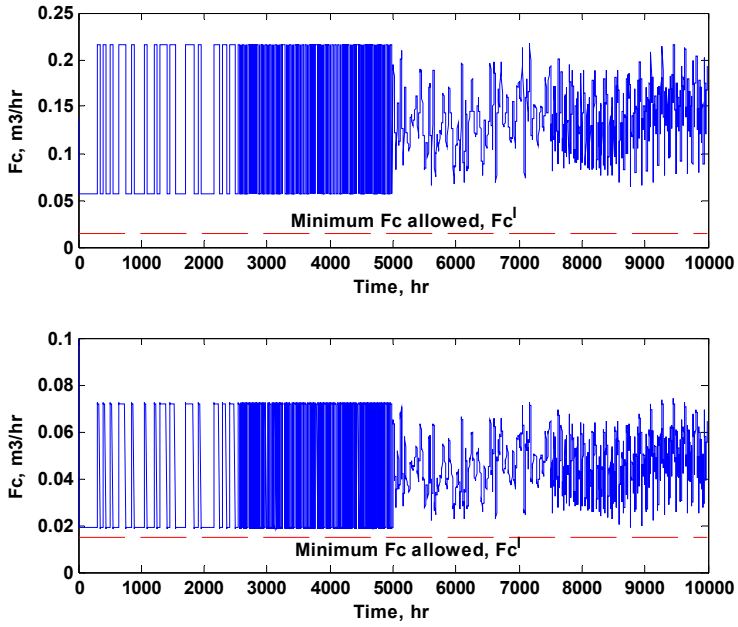

Figure 4.11 Cold feed flow rate's actual values, *Scenario II*, μ-based methodology.

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

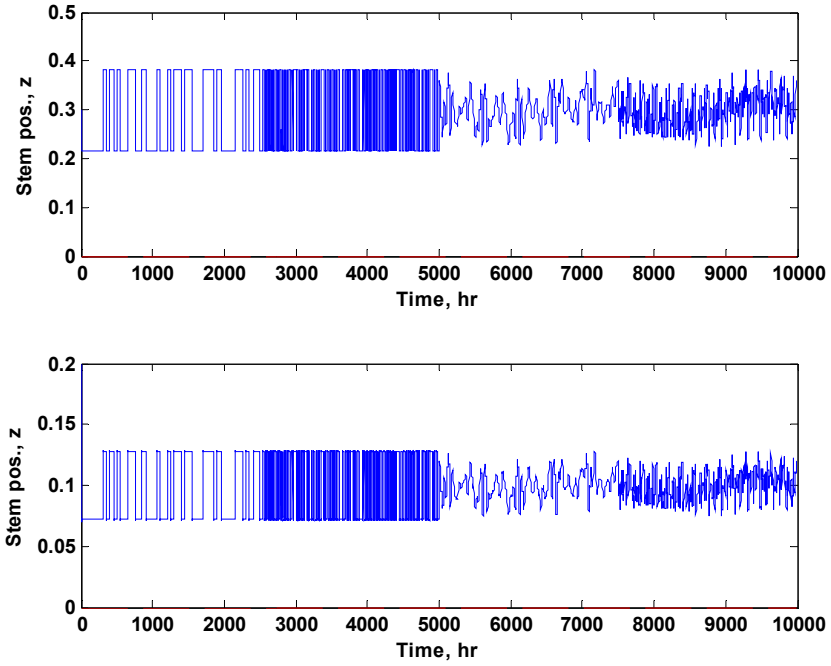

Figure 4.12 Valve constant's actual values, *Scenario II*, μ-based methodology.

Top graph: F_h at 0.15 m^3/hr. Bottom graph: F_h at 0.05 m^3/hr.

Table 3.3 lists the design variables' values obtained when the γ-based methodology was applied to *Scenario II*. This Table shows that for this scenario the design obtained when H= γ is slightly smaller than that obtained by the μ-based method. As it shown in the simulation results shown in Figure 3.7, the design obtained when H= γ does not completely satisfies the tank's hold up constraints. This confirms the fact that the γ-based variability calculation only guarantees a bound on the output error's variance and not on the output error's largest variability. On the other hand, Figure 4.9 shows that the design parameters specified from the μ-based method never violates the tank's hold up at any time *t*. Therefore, the bounds specified from the μ-based robust measure provide more accurate bounds for the tank's hold up maximum variability. Clearly, when twice of the γ values are considered in the γ-based methodology's formulation (H= 2γ) the resulting design is more conservative than that obtained by the μ-based method.

In addition, it is interesting to notice that the tank's total volume attained by the μ-based method is smaller than the design obtained by Mohideen et al.[16] given in Table 3.3. This is partly related to the fact that the bounds were slightly relaxed, to account for the steady-state operation, as compared to Mohidden's work. Thus, it was not possible to satisfy the original bounds proposed by Mohideen with the present method. However, the present method only requires conventional PI control algorithms whereas Mohideen's approach requires on-line optimization and a priori knowledge of the parameter uncertainty profile. Also, the robustness measures implemented in the present methodology guarantees that the attained design is robustly stable and valid for an infinite time horizon in the presence of perturbations in the hot stream whereas Mohideen's problem addressed a problem occurring in a finite time horizon of 30 hours.

4.4 Computational requirements by the modified method

From the optimization point of view, the use of the robust variability bound proposed in this research work avoids the task of solving dynamic optimization problems, as used in the most recent methodologies[4,23]. In this section, the computational effort required for the μ-based methodology to find a solution was compared to the computation required for solving the problem with dynamic programming.

To test the required computational effort, the calculation of the worst process variability for a given process parameter uncertainty value has been solved for the mixing tank process. This variability calculation represents the core of the μ-based method since it has to be repeated several times during the solution of the proposed optimization problem (4.21). Thus, it is logical to compare the present approach to other strategies, such as dynamic programming, on the basis of this single calculation.

The μ-based method assesses the maximum variability in a variable and for a given $\boldsymbol{\omega}$ from problem (4.8). This single calculation was performed to estimate the worst-case process variability for both scenarios and for a given values in the operating states and F_h. For *Scenario I*, the outcome of this calculation returns a bound on the tank's volume maximum variability around a nominal operation point, specified by the tank's temperature set point

(T_{sp}), the stem position of the valve at the outlet, z, the cold stream temperature, T_c, and the controller's gain and integral time, Kc_T and τ_{IT}. The nominal values for the above variables were taken from the solution of *Scenario I* (see Table 4.1). For the purpose of this comparison, the process parameter uncertainty in the mixing tank process, F_h, was set to a specific value, i.e. F_h=0.15 m^3/hr. The tank's nominal volume, V_{nom}, and nominal cold feed flow rate, $F_{c,nom}$, were estimated from the steady-state solution of the process model equations shown in (4.25). The worst-case process variability calculation was coded and solved for both scenarios in MATLAB®.

For comparison, the worst-case process variability calculation was also performed using dynamic programming. In this case, the complete nonlinear closed-loop process model of the mixing tank is used to seek for the profile in the hot stream temperature, T_h, which produces the largest process variability in the tank's volume within a finite time horizon, t_f. The decision variable in this problem is a vector composed of the hot stream temperature values at each time step, i, used as input into the process model equations. This problem has been defined for *Scenario I* in (3.36). This formulation was programmed and executed in MATLAB® using SQP[54-55]. Likewise, the process model differential equations were integrated using the MATLAB built-in function *ODE23s* which applies a modified version of the Rosenbrock method[56]. The iterative procedure followed to solve such problem has been given in Section 3.3. As in the μ-based method, the values used for T_{sp}, T_c, z, Kc_T and τ_{IT} were taken from Table 4.1 whereas F_h was set to 0.15 m^3/hr.

The worst-case process variability problem formulated with the μ-based method and dynamic programming were solved on a Pentium 4, 3.01GHz workstation with 1.5GB of RAM. Figure 4.13 shows the CPU time required by both strategies as a function of the number of sampling instants (i) used in dynamic optimization. The same sampling instants were also used by the proposed method to describe the discrete robust FIR model, i.e., the number of impulse response coefficients (h_{iq} and δh_{iq}) needed to compute (4.8). As shown in Figure 4.13, the CPU time is directly proportional to the number of sampling instants used in both strategies. This is an expected result since more calculations have to be performed at the expense of having a more accurate result.

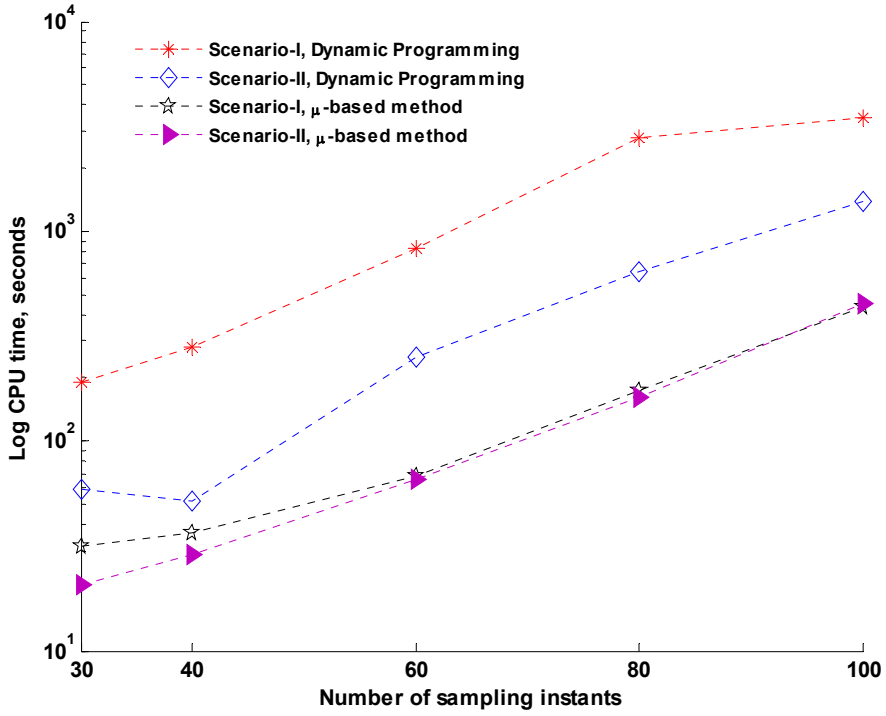

Figure 4.13 CPU time comparisons.

Figure 4.13 also shows that the proposed method estimates the worst-case process variability from 3 to 6 times faster than dynamic optimization for *Scenario I* whereas for *Scenario II* this relation is from 6 to 16 in favour of the μ-based method. The difference in the speed on the calculations between the two approaches can be explained as follows: dynamic programming requires the stepwise numerical integration of the process model equations, i.e., the $(i+1)^{th}$ sampling instant calculations are based on the calculations performed at the i^{th} time step. Thus, the solution of the actual sampling instant depends on the calculations performed at the previous time step. On the other hand, the skewed-μ method used to compute (4.8) treats each input independently ($\mathbf{\Delta}$ in 4.9), that is, for each k tested in (4.8), a bound is obtained for all the inputs in just one single step calculation rather than the recursive approach needed by dynamic programming. The drawback of the μ-based method lies in the fact that the solution to (4.8) would provide a bound rather than the true worst output variability as obtained by dynamic programming. However, the solution to (4.8) would also provide, for a given $\boldsymbol{\omega}$, the critical profile in $\boldsymbol{\upsilon}$ that generate the largest output variability. Consequently, this profile could be used to simulate the process transient behaviour using the complete process

nonlinear model to obtain the actual largest output variability. This approach will be explained in detail in chapter 5, Section 5.3.3 and Section 5.3.4.

Moreover, for this particular problem, i.e. the worst-case process variability estimation, the dynamic programming based method could potentially provide a local solution rather than the global optimum whereas the problem formulated in (4.8) is not subject to local solutions since it has been shown that the μ–based bound is convex[45], i.e., the solution obtained from (4.8) is a global maximum (see the example at the end of Section 4.1).

The above results show that the μ-based approach presented in this work highly reduces the computational effort required to assess the simultaneous design and control problem over the traditional approach that uses dynamic programming. Also, these results show the potential of the present method to tackle the simultaneous design and control for large scale processes. To estimate the increase of the computational effort when dealing with increasingly larger models, the expected computational burden required to assess the worst-case process variability for more than one variable was also analyzed. This was done by measuring the computational time required to estimate the variability for an increasing number of constraints considered in the optimization problems for *Scenario I* (4.25) and *Scenario II* (4.26), respectively. The procedure followed to obtain the nominal values was the same as in the estimation of the tank's volume maximum variability, explained at the beginning of this section. As shown in Figure 4.14, the CPU time is directly proportional to the number of variables included in the calculations. In addition, Figure 4.14 also shows the computational time required by the γ-based methodology. Although the computational times for the γ-based and the μ-based methods are similar, the γ-based methodology becomes more computationally demanding when the number of variables considered in the formulation is increased. Although these results may be case specific, they may be used as an approximate indicator of the expected computational burden requirement when the present methodology is applied to simultaneously design and control a large-scale processes.

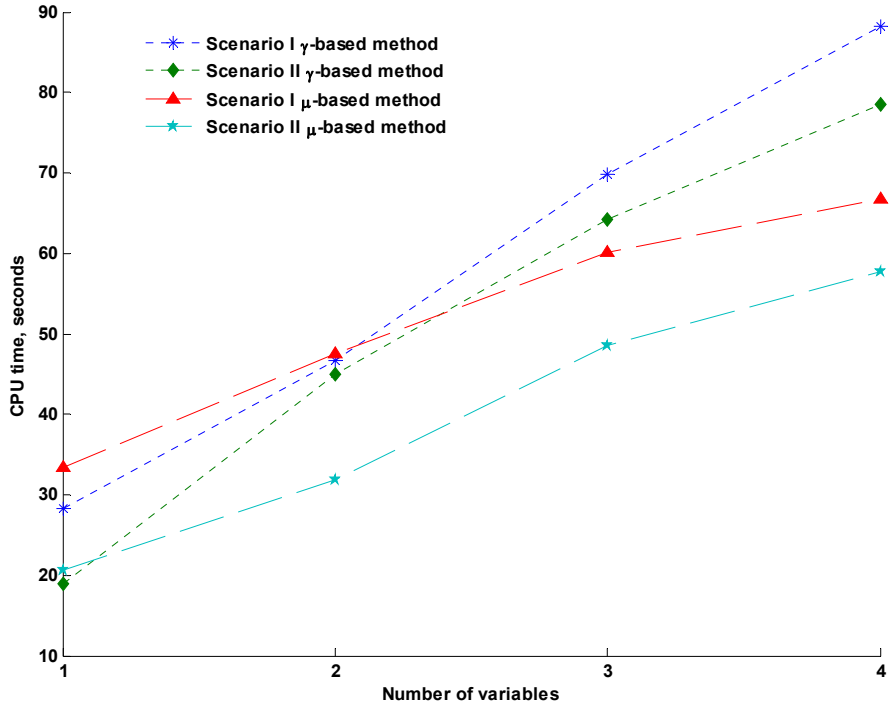

Figure 4.14 CPU time as a function of the number of variables.

In summary, this chapter have presented a new robust measure that can be used to estimate a bound on the maximum variability in a variable. This robust variability test has been used by the current research work to estimate the worst-case process variability scenario and to test process feasibility. This robust measure has replaced the γ-based variability calculations within the methodology formulation. The core of the method is based on the mathematical representation of the process closed loop nonlinear dynamic model as a robust state space model. Then, this model is transformed to a discrete robust FIR model to obtain bounds on the process worst-case variability and to test process feasibility.

The μ-based mathematical formulation was applied to the simultaneous design and control of the mixing tank process that was also used as a case study to evaluate the γ-based methodology described in the previous chapter. The results indicate that the Quadratic Lyapunov Function tends to produce conservative designs for this system.

Although dynamic programming-based methodologies do not generate a conservative design, larger computational times are required to solve this type of optimization problems, even in the case of simple processes such as the mixing tank process. The μ-based method avoids the solution of a dynamic optimization problem by proposing robustness measures to obtain bounds on the process output and constrained variables. Although the resulting design would tend to be slightly conservative due to the adopted robust modelling approach, the proposed μ-based method is based on a nonlinear constrained optimization problem which requires less computational effort than dynamic programming. Consequently, the proposed methodology offers potential for the simultaneous design and control of large scale systems. The next chapter discuss the application of the μ-based methodology to the simultaneous design and control of a large-scale system: the Tennessee Eastman Process.

5. Simultaneous Design and Control Methodology for Large-Scale Processes

One of the key objectives proposed in this research is to develop a robust control tool-based methodology that can be applied to the simultaneous design and control of large-scale processes. The previous chapters presented two robust control-based approaches to the simultaneous design and control of dynamic systems under external perturbations and process parametric uncertainty. The first methodology applies a Quadratic Lyapunov Function-based approach to define robust stability and variability tests (γ-based method) whereas the second methodology estimates the worst-case variability scenario from a Singular Structured Value-based robustness test (μ-based method). Both γ-based and the μ-based method have only been tested on a relatively simple case study involving a mixing tank process under closed-loop control.

This chapter presents a comprehensive methodology that has been proposed to attain the simultaneous design and control of chemical plants that involve several numbers of units. Although the γ-based and μ-based methodologies could be both applied to the design and control of chemical processes, the μ-based methodology was chosen as the basic metric of the design methodology due to the fact that, as discussed in the previous chapter, it provides a tighter bound on the maximal deviations of the process variables. Since reducing computation times is a must when dealing with complex process operations the μ-based methodology was further modified to provide better estimates of bounds and to reduce the speed in the calculations. To test the proposed methodology, a process model that simulates the dynamic behaviour of a chemical plant was required. Among the different chemical process models that are available in the open literature, the Tennessee Eastman (TE) process model[62] was chosen since a full simulator of the process was available and the process involves a number of interconnected units making it an ideal ground for testing the computational burden of the proposed methodology. Although the controllability of this process has been widely studied, the design of the TE plant and the interactions between design and control has not been previously addressed. Thus, it is a suitable candidate to apply the methodology proposed in this research work.

This chapter is organized as follows: Section 5.1 presents the methodology's mathematical formulation that has been proposed to simultaneously design and control chemical plants. Section 5.2 introduces the Tennessee Eastman process and the control strategy that was used to stabilize the process dynamic behaviour. Section 5.3 presents the simultaneous design and control of the reactor section of the TE plant, i.e. only the parameters that directly affect the reactor's operation are considered in the analysis in this section. The effect of external perturbations, the selection of the initial conditions for the optimization problem, comparisons of the proposed method with a dynamic programming approach to the problem, and the effect of process parameter uncertainty are also discussed in this section. In Section 5.4, the complete set of parameters specified for this plant are used to simultaneously design and control the TE process. This scenario was solved under the effect of external perturbations.

5.1 Methodology for large-scale systems

The previous chapter presented a methodology that uses a QLF-based robust test to evaluate the system's asymptotic stability and a μ-based robust performance test to estimate bounds on the worst-case system's variability and process feasibility. The robust stability test requires the closed-loop identification of a robust state space model whereas the worst-case variability and process feasibility tests require the transformation of such model into a robust finite impulse response (FIR) model (see Section 4.1). Thus, the method ends up using two different uncertain model structures to represent the same input/output closed-loop dynamic behaviour. Although this approach could be used to simultaneously design and control chemical processes, there are a series of issues that may be considered when applying this methodology to large-scale systems:

i) The μ-based methodology has proven to be efficient when it was applied to the simultaneous design and control of a mixing tank process (see Section 3.2). In this case study, the closed-loop identification of the robust state space models was simple and fast since the closed-loop process model, as described for *Scenario II*, consisted of 4 ordinary differential equations (ODE's), 2 algebraic equations (AE's) and 7 process model parameters. On the other hand, the chemical plant's dynamic behaviour is usually

represented by a large number of ordinary differential and algebraic nonlinear equations that contain hundreds of process model parameters. Therefore, it is expected that the robust state space model's parameter estimation for large scale systems will be a difficult and computationally demanding step since higher order models will be required.

ii) Although the robust state space model is identified in closed-loop, it is expected that the chemical plant will exhibit a higher degree of nonlinearity. Thus, model parameters with large uncertainty descriptions will obviously produce more conservatism in the resulting design.

iii) As mentioned in Section 4.3.1, the Quadratic Lyapunov Function used in this work generally produced a conservative result. The conservatism was partially due to the presence of the Lyapunov function rate of change in the condition used to calculate a bound on variability. The robust stability test used in the μ-based methodology is based on the same Lyapunov function's rate of change (see Problem 4.21). Thus, the use of this stability test may produce misleading results. That is, due to the conservatism of this test, one may conclude that a given robust state space model is not asymptotically stable while in principle *all* the eigenvalues of the **A** state space matrix have negative real parts. In addition, based on the computational effort to calculate this robust stability test for simple system, it is expected that the Quadratic Lyapunov Function-based robust stability test with higher order models will require a significant amount of computational time.

iv) The use of two different uncertain model structures to represent the same input/output dynamic behaviour adds unnecessary computational load within the methodology. In the case of a low order robust state space model, the transformation to its corresponding discrete robust FIR model is performed rapidly but a significant amount of computational time will be often required to convert higher order robust state space models into the robust FIR models.

v) The μ-based robust variability measure introduced in Section 4.1 requires the repeated computation of a μ problem. The dimensions of the interconnection and perturbations matrices, **M** and **Δ**, given in problem (4.8) determine the computational load required to solve such problem. The sizes of **M** and **Δ** are determined by the number of impulse

response coefficients and the number of external perturbations considered in the analysis. Thus, it is expected that the μ calculation will become computationally demanding when this test is applied to systems that require a large set of impulse response coefficients to characterize the input/output behaviour and that are subject to many external perturbations.

To address these issues, the following subsections present further modifications that were performed to the μ-based methodology to account for the simultaneous design and control of large-scale systems.

5.1.1 Closed-loop identification of a robust FIR model

It is proposed that the closed-loop dynamic behaviour of the chemical process will be directly identified from input and output values as a robust FIR model thus eliminating the need for identifying a closed-loop robust state space model. The Robust FIR model has been defined in Section 4.1 as follows:

$$y(j) = \sum_{q=1}^{m} \sum_{i=0}^{j} \left[h_{iq} \delta \upsilon_q (j-i) + \delta h_{iq} \delta \upsilon_q (j-i) \right]; \quad 0 \le j \le N \tag{5.1}$$

This model will be directly estimated from closed-loop identification as per the following steps:

a) Design a PRBNS for each disturbance (υ_q) considered in the analysis. The PRBNS's maximum length sequence and switching time can be determined from the closed-loop system's settling time and time constant. The amplitude of the PRBNS must be defined in terms of the lower and upper bounds specified for the disturbance, i.e., υ_q^l and υ_q^u. Also, depending on the computational tools available, repeat the PRBNS sequence as many times as possible to improve the quality of the model parameter's estimates.

b) Simulate the closed-loop system's output response (y) using the PRBNS specified for each disturbance.

c) The input/output data recorded from this simulation, i.e. δv_q and y, is used to fit a linear impulse response model applying the least-squares method[51]. The number of model parameters, i.e. impulse response coefficients, needed to correctly describe the system dynamics is determined from the system's closed-loop settling time specified above.

d) The least squares criterion applied for the identification of the linear impulse response model also produce an estimate the covariance matrix of the parameter estimates. The parameter estimates' variance is used by the proposed methodology as an approximation to describe the system's nonlinearities due to the changes in v_q, i.e., the elements of the covariance matrix are assumed to be the equal to the uncertainty bounds for the model parameters. Correspondingly, each impulse response model parameter is described by a nominal value (h_{iq}) complemented with model parameter uncertainty (δh_{iq}).

The above procedure is repeated for each disturbance (v_q) and for each process variable (y) that is considered in the analysis. One of the advantages of using robust the FIR models is that they are output response models that require information of current inputs and outputs and past inputs but do not require information of previous outputs. Thus, the number of parameters for the input is determined by the output's settling time, N, i.e. $h_q = [h_{1q}, h_{2q}, \ldots, h_{Nq}]$ whereas there are no parameters to be estimated for past values of the output (y). Accordingly, the determination of the model order is done systematically according to the settling time and does not require the use of computationally expensive model structure selection algorithms.

In summary, the direct identification of the robust FIR model is expected to reduce the conservatism of the bound estimated from the μ-based robust variability test and will also increase the speed in the methodology's calculations since the model transformation step from state space to FIR form discussed in chapter 4 is eliminated and consequently, the robust FIR model identification is performed fast.

5.1.2 Robust stability test for large-scale systems

As shown in Figure 2.4, the robust performance test for the SSV analysis (equation 2.40) also takes into account the system's input/output stability. This indicates that for a stable input/output system a finite bound on the system's dynamic performance is obtained. Accordingly, very large values of variability obtained from the SSV performance calculation for a finite time horizon, are an indication of system's instability. In principle, the system's asymptotic stability can be tested using the robust stability criterion presented in problem (3.12). However, this requires the identification of a robust state space model at each iteration step which can be computationally expensive for large-scale systems. Instead, the present method uses a practical approximation step to determine whether a current set of decision variables ($\boldsymbol{\eta}$ in problem 4.21) define an operating state in the system that is stable or not. This method consists in performing the robust stability test (3.12) only when the computation of the skewed-mu analysis (problem 4.8) produced a bound on the worst-case output variability that is sufficiently large when compared to the bound estimated with the set of decision variables used in the previous iteration step. From preliminary testing of the variability bounds obtained for stable and unstable systems, it was concluded that the robust stability test must be performed only when the bound estimated with the current set of decision variables ($\boldsymbol{\eta}_k$) is approximately 2 orders of magnitude larger than the bound obtained with the previous set of decision variables ($\boldsymbol{\eta}_{k-1}$). When this occurs, the identification of a robust state space model around that particular operating state ($\boldsymbol{\eta}_k$) must be estimated and the system is tested for stability as per inequality (3.12). The above approach was implemented and tested in the simultaneous design and control of the Tennessee Eastman process.

5.1.3 Reduction of the computational load in the robust variability test

The μ-based robust variability test introduced in Section 4.1 requires the specification of the interconnection and perturbation matrices, $\mathbf{M}$ and $\boldsymbol{\Delta}$, as per equations (4.11) and (4.9) respectively. The last N rows of the matrix $\mathbf{M}$ accounts for the different values of the output y at each sampling instant j. The last row of $\mathbf{M}$ that corresponds to the prediction of the output at time interval N, the settling time of the process, contains the full set of impulse response coefficients and its corresponding uncertainty values (see equations 4.13 and 4.14

respectively) of the robust FIR model. Since the critical disturbance profile for which the maximal variability occurs between time *0* and time interval *N* can be expressed by the combination of the contributions of all the coefficients of the FIR model and their corresponding uncertainty bound, it is possible to simplify the problem formulation. The simplification consists in reformulating the problem by considering only those rows and columns in the interconnection matrix **M** that contain *all* the maximum changes in the disturbance variables ($\delta\upsilon_q$), the nominal values of the impulse response coefficients (h_{iq}) and its corresponding uncertainty values (δh_{iq}). Accordingly, the matrices that define the structure of **M** in (4.11), i.e. $\mathbf{W_q}$ (4.12), $\mathbf{R_q}$ (4.13), $\mathbf{H_q}$ (4.14), **S** (4.15) and **T** (4.16), are reduced as follows:

$$\mathbf{W}_q = \begin{bmatrix} \delta\upsilon_q(0) & \delta\upsilon_q(1) & \dots & \delta\upsilon_q(p) & \dots & \delta\upsilon_q(N) \end{bmatrix} \tag{5.2}$$

$$\mathbf{R}_q = \begin{bmatrix} \delta h_{0,q} & \delta h_{1,q} & \dots & \delta h_{p,q} & \dots & \delta h_{N,q} \end{bmatrix} \tag{5.3}$$

$$\mathbf{H}_q = \begin{bmatrix} h_{0,q} & h_{1,q} & \dots & h_{p,q} & \dots & h_{N,q} \end{bmatrix} \tag{5.4}$$

$$\mathbf{S} = diag(\mathbf{T},\dots,\mathbf{T}), \quad for \quad q = 1,\dots m \tag{5.5}$$

$$\mathbf{T} = kI_{N,N} \tag{5.6}$$

Similarly, the matrices $\mathbf{\Delta}_{r1}$ and $\mathbf{\Delta}_{r2}$ defined in (4.10) and used to specify the perturbation matrix structure $\mathbf{\Delta}$ are reduced as follows:

$$\begin{aligned} \mathbf{\Delta}_{r1} &= \begin{bmatrix} \alpha_1,\dots,\alpha_N \end{bmatrix} \\ \mathbf{\Delta}_{r2} &= \begin{bmatrix} \beta_1,\dots,\beta_N \end{bmatrix} \end{aligned} \tag{5.7}$$

Following this reformulation of the problem, the resulting dimensions of the reduced **M** and $\mathbf{\Delta}$ are $(2N+1)\times(2N+1)$. Due to this simplification, the computational time to solve the μ calculation in (4.8) is significantly reduced. Table 5.1 shows the CPU time required to compute a particular μ problem using the expanded and reduced structure of the matrices **M** and $\mathbf{\Delta}$. The Table shows that the computational demands increases as the number of impulse response (IR) coefficients considered in the analysis are increased. However, the

computational requirements for the expanded **M** and **Δ** μ-based calculation grows exponentially whereas the reduced **M** and **Δ**-based μ calculation grows in a linear fashion. Figure 5.1 shows the CPU time ratio between the reduced and the expanded **M** and **Δ** structures used to solve the μ problem in (4.8). As shown, the ratio decreases exponentially as the number of IR parameters considered in the μ computation is increased. Thus, it is expected that the computational burden associated with the reduced **M** and **Δ**-based μ calculation is less sensitive to the number of IR coefficients considered in this calculation. In addition, Table 5.1 also shows the worst-case output variability bound (k) obtained from (4.8) indicating that both the expanded and reduced M and **Δ** structures provided the same result. Thus, it is expected that the μ-based robust variability test is not affected by the reformulation of the problem.

Table 5.1 CPU time comparison for the μ calculation.

Number of IR Coefficients	Reduced **M** and **Δ**		Expanded **M** and **Δ**	
	CPU time (sec)	k	CPU time (sec)	k
10	0.0469	3.5771	0.5938	3.5771
20	0.0625	4.9068	2.3438	4.9068
30	0.0938	5.9216	25.7656	5.9216
40	0.1001	6.7740	153.7500	6.7740
50	0.1250	7.5177	858.8438	7.5177

5.1.4 Mathematical formulation and algorithm

The modifications discussed in the previous subsections were applied to the μ-based methodology formulation defined in (4.21). Thus, the optimization problem that has been proposed to the simultaneous design and control of chemical plants is presented in (5.8). This formulation is different from (4.21) because the robust stability test is only performed when the worst-case output variability bounds between two subsequent iterations differ by 2 orders of magnitude and the matrices $\mathbf{M}^d$, $\mathbf{\Delta}^d$, $\mathbf{M}^g$, and $\mathbf{\Delta}^g$ are constructed from the vectors and matrices specified in (5.2)-(5.7), respectively. Although the closed-loop process model equations identified at each step of the optimal solution search are not explicitly shown in (5.8), they are implicitly considered in the evaluations of the asymptotic stability, process feasibility and the worst-case process variability.

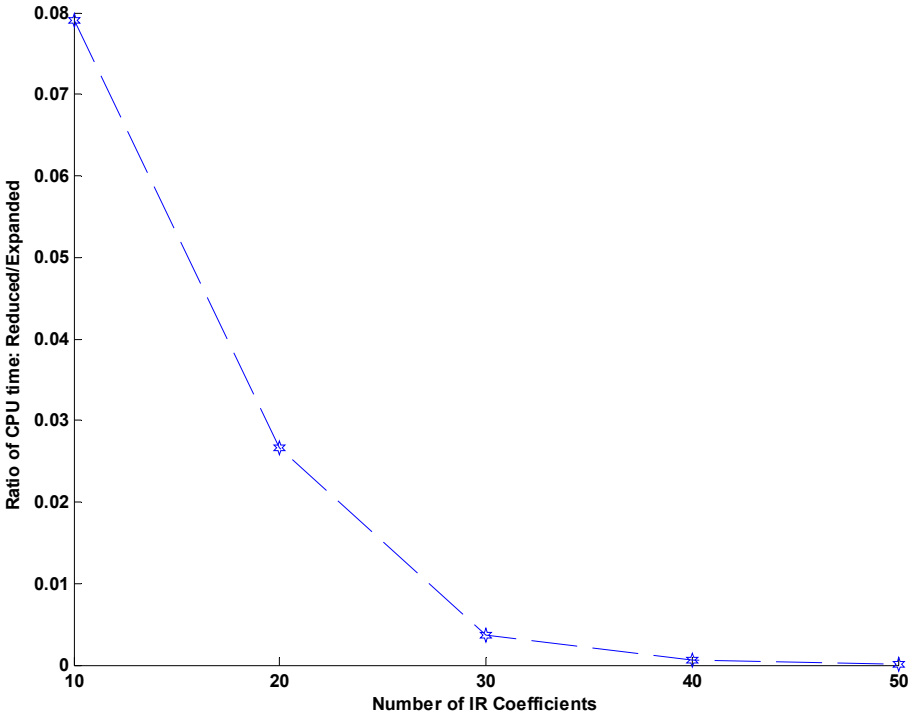

Figure 5.1 CPU time ratio for the μ calculation.

$$\min_{\boldsymbol{\eta}=[\bar{\mathbf{u}},\mathbf{p},\bar{\mathbf{w}}_{cl},\bar{\mathbf{w}}_{ol},\mathbf{d},\boldsymbol{\lambda}]} CF = CC + OP + VC \qquad \ldots \text{Cost Function}$$

s.t.

$$\max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega} \le \boldsymbol{\omega}^u} \phi^d\left(\bar{\mathbf{u}}, \mathbf{p}, \bar{\mathbf{w}}, \mathbf{c}, \mathbf{d}, \boldsymbol{\lambda}, k^d\right) \ldots \text{Worst} - \text{case process variability}$$

s.t.

$$\max_{\mu_{\Delta^d}\left(\mathbf{M}^d\right) \ge k^d} k^d \tag{5.8}$$

$$\max_{\omega^l \le \omega' \le \omega^u} \left(\bar{g} \pm \max_{\mu_{\Delta^g}\left(\mathbf{M}^g\right) \ge k^g} k^g \right) \le \rho$$

$$\boldsymbol{\eta}^l \le \boldsymbol{\eta} \le \boldsymbol{\eta}^u \qquad \ldots \text{R}obust \text{ feasibility test}$$

Process Model Equations

Control Algorithm Equations

The algorithm to solve problem (5.8) is schematically given in Figure 5.2 and summarized by the following steps:

Step 0 (Initialization). Provide initial guesses for the vector $\boldsymbol{\eta}\bullet$ and the process parameter uncertainty $\boldsymbol{\omega}\bullet$. Also, specify the lower and upper bounds for these variables and the disturbance variables, i.e. $\boldsymbol{\eta}^l$, $\boldsymbol{\eta}^u$, $\boldsymbol{\omega}^l$, $\boldsymbol{\omega}^u$, $\boldsymbol{\upsilon}^l$ and $\boldsymbol{\upsilon}^u$, respectively.

At each iteration *k*:

Step 1 (Worst-case process variability scenario). Given the process variability function φ^d and the current values in the optimization variables, $\boldsymbol{\eta}_k$, find the critical realizations in the disturbance variables $\boldsymbol{\upsilon}$ and the nominal values in the process parameter uncertainty $\boldsymbol{\omega}$ that generates the maximum process variability in the system. To solve the worst-case process variability problem defined in (5.8), it is necessary to perform the closed-loop identification of a robust FIR model for each $\boldsymbol{\omega}$ tested. The procedure to obtain such robust model was explained in Section 5.1.1. In this step, the robust FIR model must be identified from the disturbance $\boldsymbol{\upsilon}$ to those process variables that define φ^d, i.e. $\mathbf{y}^{\mathrm{d}}$.

Step 2 (Robust stability test). The bounds estimated in the previous step are compared to those obtained in the previous iteration (*k-1*). If the difference between these two bounds is of 2 orders of magnitude, i.e. $\left|k^d{}_k - k^d{}_{k-1}\right| \geq \varepsilon_1$, then the robust stability test must be performed to guarantee the system's stability. To apply the robust stability test (3.12), a robust state space model must be identified around the current set of decision variables ($\boldsymbol{\eta}_k$) and at the given process parameter uncertainty ($\boldsymbol{\omega}$) obtained from the previous step. The procedure to estimate a robust state space model has been outlined in Section 3.1.2. If the difference between the bounds is less than 2 orders of magnitude, the present analysis considers that the system is stable at the specified operating state defined by $\boldsymbol{\eta}_k$ and $\boldsymbol{\omega}$, i.e. the robust stability test is not performed at this iteration.

Step 3 (Robust feasibility test). Given $\boldsymbol{\eta}_k$, estimate the critical profiles in $\boldsymbol{\upsilon}$ and the steady-state values in $\boldsymbol{\omega}$ that produces the largest variability in the process variables ($\mathbf{g}$) that are to be maintained within pre-selected bounds. To evaluate the robust feasibility test given in (5.8), robust FIR models must be identified from the disturbance variables $\boldsymbol{\upsilon}$ to the process variable $\mathbf{g}$ for each $\boldsymbol{\omega}$ tested.

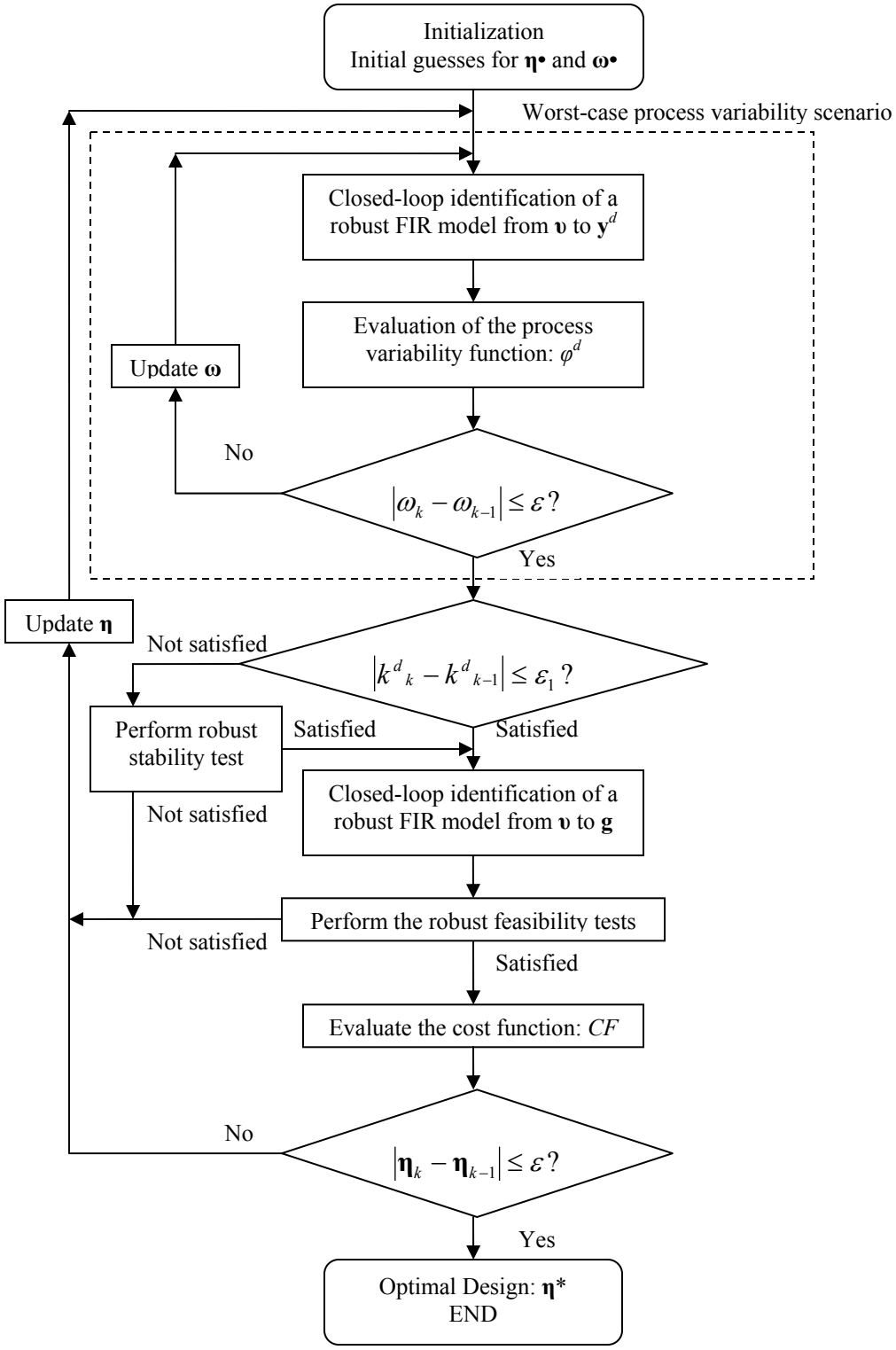

Figure 5.2 Flow sheet diagram of the proposed methodology

Step 4 (Cost function evaluation). The current $\boldsymbol{\eta}_k$ and the value for φ^d obtained from **Step 1** are used to estimate the capital, operating and variability costs of the chemical process. If the constraints specified in (5.8) and the optimization stopping criterion are satisfied, i.e. $\left|\boldsymbol{\eta}_k - \boldsymbol{\eta}_{k-1}\right| \le \varepsilon$, then **STOP**, an optimal design has been found and the values of the optimization variable in the k^{th} iteration becomes the solution for this problem, i.e. $\boldsymbol{\eta}_k = \boldsymbol{\eta}^*$. Otherwise proceed to the next step.

Step 5 (Reset). Set $k = k + 1$, redefine the values in $\boldsymbol{\eta}_k$ according to the optimization algorithm that is used to solve problem (5.8) and go back to **Step 1**.

5.2 The Tennessee Eastman Process

The Tennessee Eastman (TE) process[62] is an industrial challenging problem that has been widely used to test topics related to process systems engineering. The process involves five major process units: a reactor, a product condenser, a flash separator, a recycle compressor and a product stripper. The process consists of four gaseous components (A, C, D and E) that react, in the presence of one inert component (B), to form two liquid products (G and H) and one by-product (F). The reactions that occur in this process are:

$$\begin{aligned} A(g) + C(g) + D(g) &\rightarrow G(liq) \quad \text{Product} \\ A(g) + C(g) + E(g) &\rightarrow H(liq) \quad \text{Product} \\ A(g) + E(g) &\rightarrow F(liq) \quad \text{By-product} \\ 3D(g) &\rightarrow 2F(liq) \quad \text{By-product} \end{aligned} \tag{5.9}$$

The four reactions are defined as irreversible exothermic reactions. The reactions rates are temperature-dependent and are described by Arrhenius-like functions. The first reaction specified in (5.9) has the highest activation energy in the system and thus it is more sensitivity to temperature changes. In addition, the reactions are approximately first order with respect to the reactant concentrations.

Based on the market demands and capacity limitations, the production targets specified for this process are given by the products' mass ratios and flow rates. Accordingly, several modes of operation are defined for this process summarized in Table 5.2. As shown in this

Table, the modes of operations are determined by the desired mix in the products G and H. Mode 1 of operation is referred to as the base case since it is the most common one. The production rates for modes 4, 5 and 6 were not specified by Downs and Vogel[62].

Table 5.2 Modes of operation of the TE plant

Mode	G/H mass ratio	Production rate G*/H* (kg/hr)
1	50/50	7038/7038
2	10/90	1408/12669
3	90/10	10000/1111
4	50/50	Maximum production rate
5	10/90	Maximum production rate
6	90/10	Maximum production rate

The TE plant's process flow sheet is shown in Figure 5.3. Pure reactants A, D and E are fed to the reactor unit together with a mixture of gaseous components that comes from the recycle stream (stream 8 in Figure 5.3). The components react inside the reactor to form the liquid products G and H. The gas phase reactions are catalyzed by a non-volatile catalyst dissolved in the liquid phase. The reactor has an internal cooling bundle for removing the heat of reaction. The products leave the reactor, along with the unreacted components, in a vapour phase. The reactor's outlet stream is passed through a partial condenser. The products from the condenser are a vapour mixture of products and unreacted gaseous reactants. The volatile and non-volatile gas mixture components are separated in a flash unit. The noncondensed components are recycled back to the reactor's feed stream through a centrifugal compressor. Additionally, a purge stream (stream 9 in Figure 5.3) is used to remove the excess of the inert component (B) and the by-product (F) from the system. The liquid collected in the separator is pumped to a stripper column to separate the remaining unreacted components from the products. To achieve separation, the solvent used in this process unit is a mixture of reactant A and C, which are fed to the process through the stripper base, indicated as stream C in Figure 5.3. The vapour stream from the stripper is mixed with the recycle stream. The liquid stream at the bottom of the striper is reheated using steam to refine the liquid products. The products G and H exit the stripper (stream 11 in Figure 5.3) and are separated in a downstream refining section that has not been included in the TE problem.

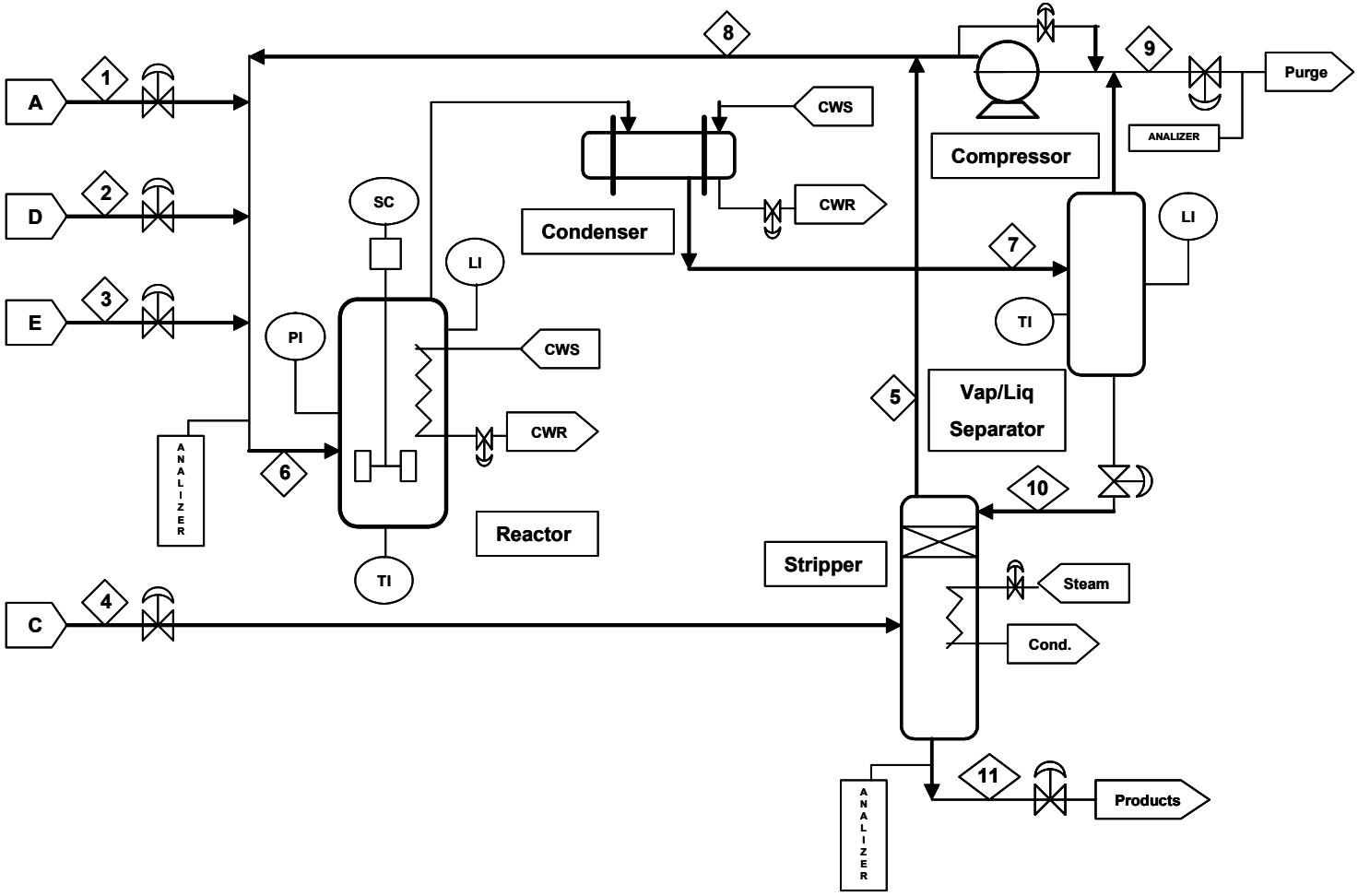

Figure 5.3 The Tennessee Eastman process

Downs and Vogel[62] specified for this problem 41 available measurements and 12 manipulated variables that can be potentially used for control. Also, six process operational constraints were specified for this problem. These constraints are primarily used to protect the process units, i.e. reactor, flash separator and stripper, from an undesired event, e.g. the gas in the reactor exceeding its maximum allowed working pressure. Likewise, the substances' physical properties and a steady-state analysis of the base case mode of operation (mode 1) were also provided for this problem. Appendix B lists the process measurements, the manipulated variables, the process constraints and the physical properties specified for this system. Furthermore, products variability restrictions were also specified for this process. According to Downs and Vogel[62] flow rate changes in the products stream (stream 11 in Figure 5.3) greater than ±5% with significant frequency content in the range 8-16 h^{-1} and fluctuations exceeding ±5 mol% in the product G in the products stream with significant frequency content in the range 6-10h^{-1} are particularly harmful for this process.

In order to provide a tool that can be used to test a wide variety of process systems engineering problems, the Downs and Vogel[62] made available a mechanistic model-based code that simulates the open-loop transient operation of this process. The process model equations in the TE code are formulated in the following state space form:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= f\left(\mathbf{x}, \mathbf{u}, \boldsymbol{\upsilon}, t\right) \\ \mathbf{y}(t) &= h(\mathbf{x}, t, \varepsilon) \end{aligned} \tag{5.10}$$

where the vector **x** represent the process states variables, **y** is the vector that describes the available measurement outputs and **u** are the available manipulated variables. The vectors **x**, **y** and **u** are of lengths 50, 41 and 12, respectively. Following (5.10), the vector *f* is an implicit nonlinear function that models the mass and energy balances in the reactor, the recycle stream, the flash unit and the stripper column. Similarly, *h* is also an implicit nonlinear function that relates the process states to the available measured outputs. To provide more realistic plant behaviour, the output measurements are corrupted by Gaussian measurement noise (ε) included in the function *h*. For control purposes, a set of 20 process disturbances (**υ**) were also included in the TE process model. Each disturbance affects the process operation in a different form. In addition, a process operating cost function was also provided for the

basic operating mode. Appendix B shows the TE code provided by Downs and Vogel [62] to simulate the TE plant's dynamic behaviour. Additionally, Appendix B includes the disturbance set and the operating costs specified for the TE process. Further details about this process can be found in Downs and Vogel[62].

The present research work chose the Tennessee Eastman process as a case study to test the simultaneous design and control methodology presented in Section 5.1. This process was selected for the following reasons:

i) It is a benchmark problem that has been widely studied by the process systems engineering community.

ii) Although more than 100 publications have used this process as a case study, the integration of process design and control has not been addressed for this process.

iii) The TE process is open-loop unstable, that is, open-loop simulations of this plant reach it shutdown limits within an hour. Therefore, a regulatory control strategy has to be used for stability reason, i.e. open loop operation is not possible.

iv) The process exhibits a complex dynamic behaviour with a high degree of interaction between the process variables. Similarly, the plant contains a recycle stream that introduces natural positive feedback into the process leading to further control challenges. For example, this effect may produce non-minimum phase behaviour that can seriously affect the process stability.

Due to these facts, the Tennessee Eastman process is a suitable and challenging candidate to test the proposed methodology for the simultaneous design and control of large-scale systems.

The methodology's formulation presented in (5.8) requires the specification of the process flowsheet and a suitable control structure that stabilizes the process. For simplicity the process flowsheet synthesis and control structure selection problem have not been considered in the present analysis. Thus the TE process flowsheet originally proposed by Downs and Vogel[62] is assumed for the analysis. Since the TE process is a challenging process control

problem, several control strategies have been reported in the literature for this particular process[63-71]. Each strategy present different control objectives and control configurations. To assess the integration of design and control of the TE process the present research work chose the decentralized control strategy proposed by Ricker[63]. This control strategy was chosen because it can attenuate almost all the disturbances expected for the TE process. Also, a Simulink® code provided by Ricker[63] is available to simulate the plant's closed-loop transient behaviour. Although Ricker's control structure was developed for the plant's six modes of operation, only the base case operating mode was considered in the present analysis. The production targets specified for the base case are given in Table 5.2.

Figure 5.4 shows the decentralized control strategy proposed by Ricker[63]. As shown, this control configuration requires 17 PI controllers and 2 PI override controllers (not shown in Figure 5.4) to stabilize the plant. This control configuration was designed based on process heuristics. Similarly, the PI controllers' tuning parameters were specified based on the insight gained from the plant's dynamic simulations.

The plant's overall dynamic performance is determined by a production rate control mechanism developed for this process. This method uses a ratio control-like configuration on the inlet and outlet streams' flow rate to regulate the product's production rate. Figure 5.5 shows the ratio control structure designed for this control strategy. The production index (*Fp*) is given in percentage form where at the base case 100 % corresponds to a production rate of 23 m^3/hr. The set point for the flow controllers are obtained from:

$$y_i^{set\ point} = r_i Fp \tag{5.11}$$

where r_i is the current signal ratio for stream *i* obtained from the controllers' output signal according to the configuration shown in Figure 5.4. The control configurations for the liquid level control on the reactor, the flash separator and the stripper column, temperature control on the reactor and separator, the control of the chemical inventories and the product composition control were designed based on process heuristics combined with the production rate ratio control mechanism shown in Figure 5.5. Further details about this control configuration are given in Ricker[63].

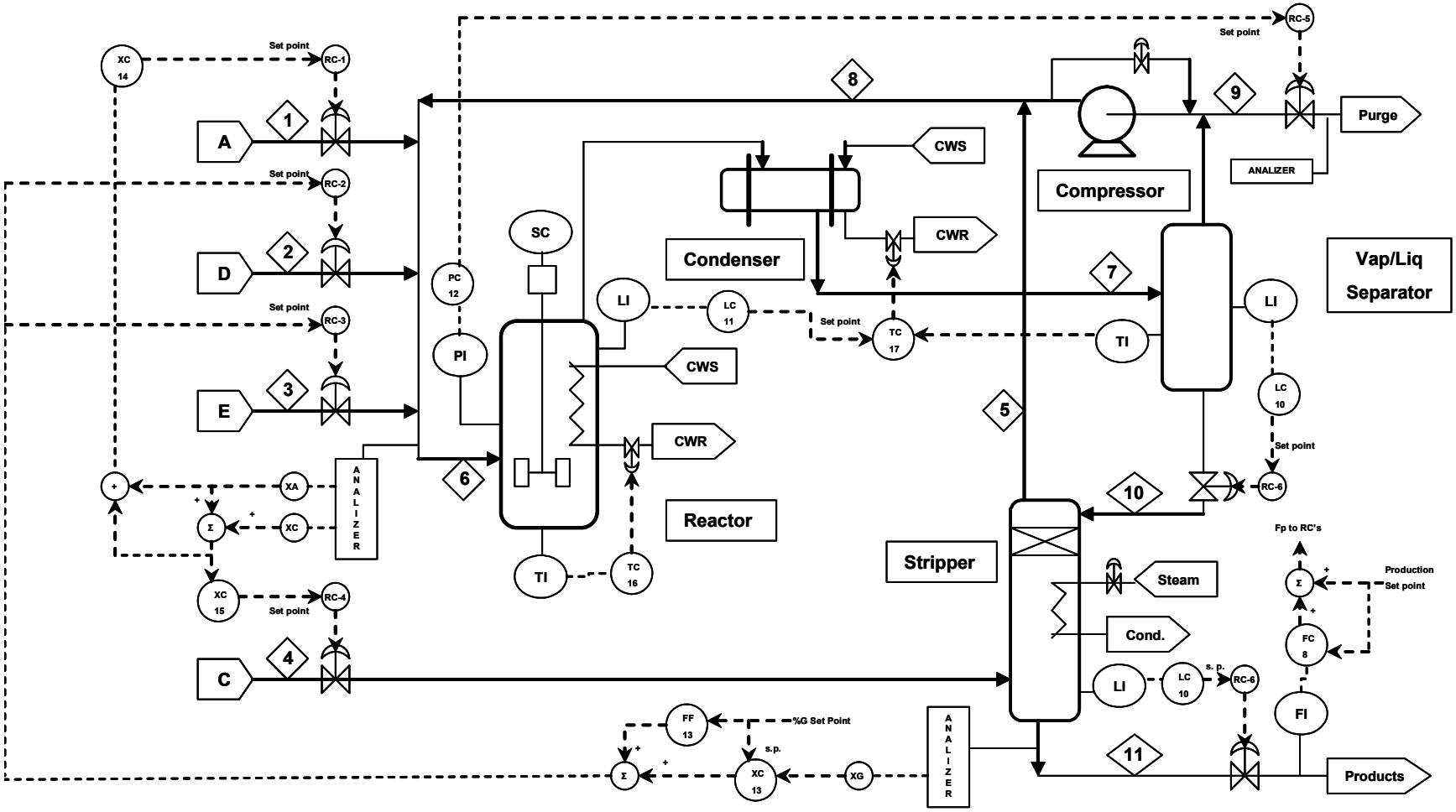

Figure 5.4 Decentralized control structure proposed by Ricker[63].

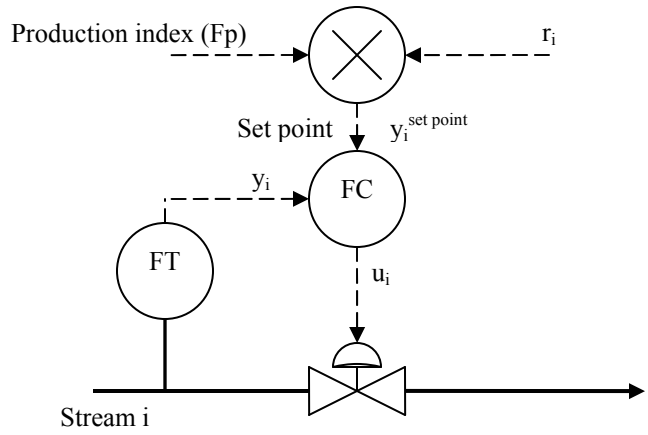

Figure 5.5 Ratio control configuration

Table 5.3 lists the PI control loops characteristics including the controller tuning parameters values and the set point values specified for the base case mode of operation[63]. As shown in Table 5.3, the plant has nine adjustable set points that can be potentially used for optimization. Likewise, the PI controller constants together with the parameters that specify the equipment sizes, i.e. reactor, separator and stripper capacities, can be also used for optimization. Thus, there are 46 variables that can be potentially specified as decision variables for the simultaneous design and control of the TE problem. In addition, the present control configuration assumes the agitator speed to be 100% and the compressor recycle valve and the stripper steam valve to 0% of opening. Thus, the present research work considered those variables as constants parameters.

The objective defined for this problem is to find the values in the set points, the controllers tuning parameters and the equipment size's parameters that simultaneously minimize the annualized cost of the TE process at the base case mode of operation. The resulting design must ensure process stability and process feasibility at any time *t* in the presence of magnitude-bounded disturbances and at any given steady-state value in the process parameter uncertainty.

Base on the above, two scenarios were proposed to study the simultaneous design and control of the TE process. The first scenario considers only the design of the reactor whereas the second scenario takes into account the design of the reactor, the flash separator and the

stripper column. The details of each scenario and the resulting designs obtained by the present method are discussed in the following Sections.

Table 5.3 Control loops characteristics for the strategy shown in Figure 5.4

Loop	Controlled variable	Manipulated variable	Set point (Base case value)	Controllers' parameters	
				Kc	τ_I(min)
1	A feed rate (stream 1)	A feed flow (stream 1)	r_1xFp	0.01	0.001
2	D feed rate (stream 2)	D feed flow (stream2)	r_2xFp	$1.6x10^{-6}$	0.001
3	E feed rate (stream 3)	E feed flow (stream 3)	r_3xFp	$1.8x10^{-6}$	0.001
4	C feed rate (stream 4)	C feed flow (stream 4)	r_4xFp	0.003	0.001
5	Purge rate (stream 9)	Purge valve (stream 9)	r_5xFp	0.01	0.001
6	Sep. Liq. rate (str. 10)	Sep. pot Liq. flow (stream 10)	r_6xFp	$4x10^{-4}$	0.001
7	Strip. Liq. rate (str. 11)	Strip. Liq. flow (stream 11)	r_7xFp	$4x10^{-4}$	0.001
8	Production rate	Production index (Fp)	Production set point (22.89 m^3/hr)	3.2	120
9	Stripper liquid level	Ratio in loop 7 (r_7)	Strip. level set point (50 %)	$-2x10^{-4}$	200
10	Separator liquid level	Ratio in loop 6 (r_6)	Sep. level set point (50 %)	$-1x10^{-3}$	200
11	Reactor liquid level	Setpoint of loop 17 (l_{17})	Reac. level set point (65 %)	0.8	60
12	Reactor pressure	Ratio in loop 5 (r_5)	Reactor pressure (2800 kPa)	$-1x10^{-4}$	20
13	Mol % G in stream 11	Ratio in loop 2 (r_2) and 3 (r_3)	Mol % G set point (53.83 %)	-0.4	100
14	y_A	Ratio in loop 1 (r_1)	y_A set point (63.1373 %)	$2x10^{-4}$	60
15	y_{AC}	Ratio in loop 4 (r_4)	y_{AC} set point (51 %)	$3x10^{-4}$	120
16	Reactor temperature	Reactor coolant valve	Reac. temp. set point (122.9 °C)	-8.0	7.5
17	Separator temperature	Condenser coolant valve	l_{17}	-4	15
18*	Maximum react. Press.	Production index (Fp)	Production set point	$2.0x10^{-6}$	0.001
19*	Reactor level override	Recycle valve	Reac. level set point	$1.0x10^{-6}$	$1.0x10^{-5}$

** Override loops not shown in Figure 5.4*

5.3 Simultaneous design and control of the TE process: Scenario I-Reactor's design only

In this scenario, only design changes in the reactor section of the TE process were considered in the analysis. Thus, the only equipment that is to be designed in this scenario is the reactor's capacity. This process unit was chosen because the reactions that produce products G and H take place in this unit. Besides, the TE process is open-loop unstable because of the reactions' sensitivity to temperature. Thus, the smallest change in the reactor' temperature causes an imbalance in the reaction that produces G (see 5.9) leading to quick changes in composition of the components in the reactor's gas phase. This causes a sudden increment in the reactor's pressure which destabilizes other sections of the plant and that eventually may result in the plant's shut down. Therefore, it is expected that the reactor's operation play a key role in the overall operation and have a significant influence on the plant's stability and performance.

Accordingly, only the controller tuning parameters that are related to the reactor were considered in the analysis: the reactor pressure, level and temperature controllers listed as loops 12, 11and 16 in Table 5.3, respectively and the control that regulates the purge losses denoted as loop 5 in Table 5.3. The rest of the controllers tuning parameters were set to the values specified in Table 5.3. Similarly, the set points for the reactor pressure, level and temperature controllers and the set point for the production rate controller were defined as optimization variables within the analysis. The rest of the set points were set to their defaults values shown in Table 5.3. In summary, 13 decision variables were specified for this scenario and are collectively denoted as the vector $\boldsymbol{\eta}$.

TE process cost function.-

For the present study, the sizes of the reactor, the flash unit and the stripper column were used to specify the TE process capital cost. The capacities specified by Downs and Vogel[62] for each process unit are: 1,300 ft^3 for the reactor, 3,500 ft^3 for the flash unit and 156.5 ft^3 for the stripper column, respectively. Since Downs and Vogel did not provide any additional information regarding the equipment sizes, the present work assumed that the three process units are vertical cylindrical vessels made of carbon steel with a length/diameter ratio of 4.

Hence, the bare-module cost, in 1982 USD, for each process unit can be expressed as follows[72]:

$$C_{BM,unit} = 5946 D^{2.1}\left(19.82 - 12.55 \ln P + 2.36(\ln P)^2\right) \tag{5.12}$$

where D is the vessel's diameter in meters and P is the pressure in bars. Assuming that the plant was built in 1992, the TE process annualized capital cost function is defined as follows:

$$CC = r\left(C_{BM,reac} + C_{BM,sep} + C_{BM,strip}\right)(360/315) \tag{5.13}$$

where r is the desired return on investment. This parameter was set to 20 %/yr which is a typical value for this factor[72]. Since the present scenario only consider redesign of the reactor, the bare module costs for the flash unit and the stripper column were fixed to a constant value according to their capacity specifications. Thus, the present scenario only requires the changes in plant's capital cost due to changes in the reactor capacity.

The operating cost function for the plant's base case mode of operation was defined as follows[62]:

$$\begin{aligned} OP = &\left(purge\ costs\right)\left(purge\ rate\right) + \left(product\ stream\ costs\right)\left(product\ rate\right) \\ &+ \left(compressor\ costs\right)\left(compressor\ work\right) + \left(steam\ costs\right)\left(steam\ rate\right) \end{aligned} \tag{5.14}$$

where OP represents the TE process operating costs in $\$/h$. The molar costs for each component, which are used to estimate the purge costs and the product stream costs, the compressor costs and the steam costs in the stripper section are given in Appendix B, Table B.6. The purge rate, production rate, steam rate and compressor work are obtained from the steady-sate solution of the TE process model equations. Appendix B shows the procedure to obtain the operating cost at the base case using the data provided by Downs and Vogel[62].

The capital and operating costs given in (5.13) and (5.14) determine the plant's steady-state costs. To assign an economic value to the process dynamic performance, the present work specified a process variability function (φ^d in 4.21) for the process. The TE process dynamic performance is measured in terms of the base case production target values given in Table 5.2. The process variability function for each product is defined as the deviations in the

products' mass flow rates with respect to their target values. To ensure that the products meet the production mass flow rate demands (G* and H*) at all time, the nominal values of the products mass flow rate (Ğ and $\breve{H}$) have to be specified above its targets values. Thus, the fluctuations in the products' mass flow rates due to external perturbations and parametric uncertainties are expected to be above the targets at all times. To measure the products variability with respect to their nominal values, bounds on the largest deviation in the products mass flow rates (k^G and k^H) are estimated using (4.8) and the interconnection and perturbation matrices structures (**M** and **Δ**) presented in Section 5.1.3. Figure 5.6 illustrate the fluctuations in the process variability for one of the products and the corresponding target that is imposed as a lower bound in the calculations. As shown by this Figure, the process variability function for each product has a physical meaning. That is, the variability in product G (φ^G) is specified as the difference between the ideal case flow rate where the plant is operating at steady-state and at the products' target values (G*) and the true products' flow rates obtained when external perturbations and parametric uncertainties are affecting the process (Ğ, k^G). Thus, the variability function for products G and H are defined as follows:

$$\begin{aligned} \varphi^G &= \max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega} \le \boldsymbol{\omega}^u} \left(\breve{G} + \max_{\mu_{\boldsymbol{\Delta}^G}(\mathbf{M}^G) \ge k^G} k^G - G^* \right) \\ \varphi^H &= \max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}_1 \le \boldsymbol{\omega}^u} \left(\breve{H} + \max_{\mu_{\boldsymbol{\Delta}^H}(\mathbf{M}^H) \ge k^H} k^H - H^* \right) \end{aligned} \tag{5.15}$$

where **ω** represents the process parameter uncertainty and the matrices $\mathbf{M}^G$, $\mathbf{M}^H$, $\boldsymbol{\Delta}^G$ and $\boldsymbol{\Delta}^H$ are the corresponding interconnection and perturbations matrices used in the μ analysis to estimate bounds on the products' worst-case variability (k^G and k^H). These matrices are formulated according to the expressions developed in Section 5.1.3.

The functions shown in (5.15) represent the process variability function used in the present research work to measure the TE process dynamic performance. These functions are used to calculate the additional amount of products' mass flow rate (kg/h) that must be produced to remain within the production specs. Then, a cost is assigned to this *excess* in product's mass flow rate.

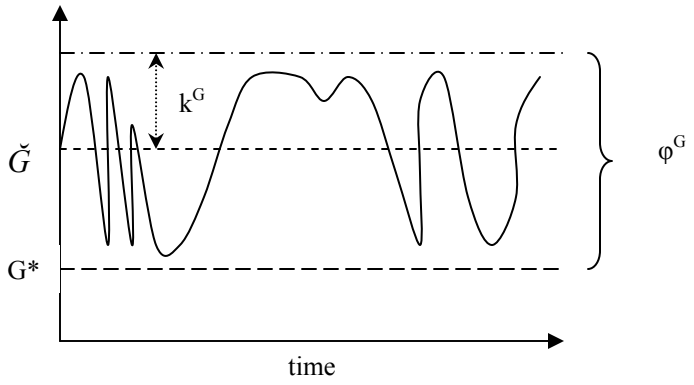

Figure 5.6 Process variability for product G

Accordingly, the variability cost function for the TE process is defined as follows:

$$VC = cp_G \varphi^G + cp_H \varphi^H \tag{5.16}$$

where VC is the process variability cost expressed in $/h. The terms cp_G and cp_H are the costs of producing products G and H, respectively. The values of cp_G and cp_H are given in Appendix B, Table B.6. In addition, φ^G and φ^H represent the products variability estimated from (5.15).

Based on the above descriptions, the TE process annualized cost function is defined as follows:

$$CF_{TE} = CC + 8760(OP + VC) \tag{5.17}$$

where CC, OP and VC are the capital, operating and variability cost functions that has been defined in (5.13), (5.14) and (5.16), respectively. The numerical coefficient in (5.17) is a conversion factor used to represent the TE cost function in dollars per year.

Robust closed-loop TE process model.-

The key idea in the present research work is to represent the process closed-loop nonlinear dynamic behaviour as a nominal linear model complemented with uncertain model parameters, i.e. a robust closed-loop model. As it was mentioned in Section 5.1.1, the closed-

loop process model for a large-scale system is represented by a discrete robust FIR model (4.2). The TE process' robust FIR models were identified following the procedure explained in Section 5.1.1. The closed-loop TE process model provided by Ricker[63] was used to simulate the plant's dynamic behaviour around a nominal operating state specified by values in the scenario's decision variables (**η**). The data collected from the simulations was used to identify the robust FIR models. These models are used by the present methodology to test process stability and to estimate bounds on the worst-case process variability and process feasibility. To accomplish these calculations a set of robust FIR process models must be identified from each of the disturbances considered in the analysis to the specific variable that is to be tested for process stability, worst variability or process feasibility. The closed-loop TE process model consist of 41 available measurements, 9 set points, from which 6 are also process available measurements, and 12 manipulated variables, out of which 3 were fixed to a constant value by Ricker's control strategy, i.e., agitator speed, recycle valve and stripper's steam valve. Thus, a total of 53 robust FIR models have to be identified for each new set of values in **η** tested by the optimization algorithm. For accurate identification of the robust FIR models, all the available process measurements were assumed to be continuous.

TE process stability.-

The stability of the TE process was evaluated according to the procedure outlined in Section 5.1.2 and Section 5.1.4.

TE process worst-case variability.-

The expressions φ^G and φ^H in (5.15) represent the worst-case process variability functions specified for the TE process. To estimate φ^G and φ^H, it is necessary to define robust FIR models from each disturbance (υ) to the products' mass flow rates (G and H). If n_υ represents the number of disturbances considered in the analysis, then $2\times n_\upsilon$ robust FIR models of the type given in (4.2) have to be identified when the system is operated around the current decision variables values (**η**) and for each process parameter uncertainty values (**ω**) that is tested by the optimization. The robust FIR nominal and uncertain model parameters, i.e. h_{iq} and δh_{iq}, in conjunction with the deviations in the selected disturbances

($\delta\upsilon_q$) are then used to define the interconnection and perturbation matrices $\mathbf{M}^G$, $\mathbf{M}^H$, $\mathbf{\Delta}^G$ and $\mathbf{\Delta}^H$ specified in (5.15). The structure of the matrices **M** and **Δ** was defined in (4.11) and (5.2)-(5.6) and (4.9) and (5.7), respectively. The solutions of the inner optimization problems described in (5.15) return bounds over the largest deviation in the G and H's mass flow rate (k^G and k^H) with respect to $\breve{G}$ and $\breve{H}$ and at a certain value of the parametric uncertainty **ω**. Moreover, the nominal values for the products' mass flow rates ($\breve{G}$ and $\breve{H}$) are determined from the current values in the vectors **η**, **ω** and the steady state solution of the TE process model equations in (5.13). In addition, the production rate targets for the base case mode of operation are given in Table 5.2.

The evaluation in (5.15) returns a *bound* on the products' mass flow rate (kg/h). Also, the functions specified in (5.15) return the critical time-dependent profile in **υ**, obtained from the μ analysis, and the nominal steady-state value in **ω** that produces the maximum variability in G and H. These critical profiles in **υ** and **ω** can then be used to simulate the TE process model to estimate the *actual* maximum variability in the products' mass flow rate. This approach, where the worst calculated disturbance is used as input to the actual model equations to compute the maximal output variability, will be compared in Section 5.3.3 to the results obtained by using the bounds on the output variability based on the μ approach.

TE process feasibility.-

As shown in (5.11), the present methodology applies the formulation given in (4.20) to ensure that the TE process constraints are satisfied. The interconnection and perturbation matrices shown in (4.20) were defined in (4.9) and (4.11) and are constructed from the matrices specified in (5.2)-(5.7). As in the worst-case process variability scenario, the nominal values of the process variables to be evaluated are obtained from the current values specified for **η** and **ω** combined with the TE process model's steady-state solution.

Table 5.2 specifies that the base case operation mode requires a 50:50 mass ratio between products G and H. Thus, mass ratio constraints for the process products are included in the problem formulation as follows:

$$
\begin{aligned}
& xm_G - 1.02xm_H \le 0 \\
& xm_G - 0.98xm_H \ge 0 \\
& xm_G + xm_H - 0.98 \ge 0
\end{aligned}
\tag{5.18}
$$

where xm_G and xm_H are the component's mass fractions in the products stream (stream 11 in Figure 5.4). The values for xm_G and xm_H are determined from the current decision variables values ($\boldsymbol{\eta}$) and the steady-state solution of the mass and energy balances equations specified for the TE process. As shown in (5.18), the products' mass ratio constraints were relaxed by allowing a 2 percent deviation from their targets values. This was done to enlarge the problem's feasible search space.

According to Figure 5.6, the process variability function (5.15) was defined as the difference between the maximum deviations in the products' flow rates with respect to a nominal value and their corresponding base case target values. The outcome in (5.15) provides the amount in *kg*/*h* that must be produced in excess to satisfy the production specs but it does not guarantee that the products' mass flow rates variability may violate the productions target values. Thus, constraints on the minimum products' mass flow rate variability were included in the analysis. These constraints were formulated using the robust feasibility test description given in (5.8), that is:

$$
\begin{aligned}
& \max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega} \le \boldsymbol{\omega}^u} \left(\breve{G} - \max_{\mu_{\Delta^G}\left(\mathbf{M}^G\right) \ge k^G} k^G - G^* \right) \ge 0 \\
& \max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}_1 \le \boldsymbol{\omega}^u} \left(\breve{H} - \max_{\mu_{\Delta^H}\left(\mathbf{M}^H\right) \ge k^H} k^H - H^* \right) \ge 0
\end{aligned}
\tag{5.19}
$$

These inequalities ensure that the resulting design meets the production targets at all times in the presence of disturbances and parametric uncertainty. It should be noted that the worst-case profiles found for $\boldsymbol{\upsilon}$ and $\boldsymbol{\omega}$ from the process variability function evaluation in (5.15) are the same used for the constraints posed in (5.19). Therefore, these constraints are evaluated using only the final values obtained from (5.15) for k^G, k^H, $\breve{G}$ and $\breve{H}$, i.e. it is not necessary to solve the maximizations of k^G and k^H within the parenthesis in equation (5.19).

In Section 5.2, products variability constraints for the TE process were specified. To account for these fluctuations in the products stream, the following constraints have been added to the problem formulation:

$$\begin{aligned}
&\max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}_2 \le \boldsymbol{\omega}^u} \left(-0.05\left(\breve{G} + \breve{H}\right) + \max_{\mu_{\Delta G}\left(\mathbf{M}^G\right) \ge k^G} k^G + \max_{\mu_{\Delta H}\left(\mathbf{M}^H\right) \ge k^H} k^H \right) \le 0 \\
&\max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}_3 \le \boldsymbol{\omega}^u} \left(-0.05\breve{x}_G + \max_{\mu_{\Delta^{x_G}}\left(\mathbf{M}^{x_G}\right) \ge k^{x_G}} k^{x_G} \right) \le 0
\end{aligned} \tag{5.20}$$

The first constraint ensures that the mass flow rate variability in the products stream is ±5% of the nominal production mass flow rate ($\breve{G}$ and $\breve{H}$) whereas the second constraint indicates that the product G's mol fraction at the outlet (stream 11 in Figure 5.4) cannot deviate ±5 mol% G of its set point value ($\breve{x}_G$). To evaluate the quality constraint in G, robust FIR models have to be identified from the disturbance to the G's mole fraction (x_G) in the products stream. As shown in Table 5.3, the mole fraction of G has been specified as a set point by the decentralized control structure. Thus, x_G may be a decision variable which value may be obtained from optimization. However, x_G was not considered as a decision variable for this scenario and thus its set point value was set to the value shown in Table 5.3.

In addition to the products variability constraints, the TE problem also considers six process operability constraints that protect the three major process units from exceptional critical events. The pressure, temperature and liquid level operability ranges for the reactor, separator and stripper column are shown in Table B.3, Appendix B. Since the reactor is the only equipment designed in this scenario, robust feasibility tests formulated as shown in (5.8) were included in the problem formulation to account for the reactor's process operational constraints as follows:

$$
\begin{aligned}
&\max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}_4 \le \boldsymbol{\omega}^u} \left(\breve{P}_r + \max_{\mu_{\boldsymbol{\Delta}^{P_r}}(\mathbf{M}^{P_r}) \ge k^{P_r}} k^{P_r} - 2895 \right) \le 0 \\
&\max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}_5 \le \boldsymbol{\omega}^u} \left(\breve{L}_r + \max_{\mu_{\boldsymbol{\Delta}^{L_r}}(\mathbf{M}^{L_r}) \ge k^{L_r}} k^{L_r} - 100 \right) \le 0 \\
&\max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}_5 \le \boldsymbol{\omega}^u} \left(30 - \breve{L}_r + \max_{\mu_{\boldsymbol{\Delta}^{L_r}}(\mathbf{M}^{L_r}) \ge k^{L_r}} k^{L_r} \right) \le 0 \\
&\max_{\boldsymbol{\omega}^l \le \boldsymbol{\omega}_6 \le \boldsymbol{\omega}^u} \left(\breve{T}_r + \max_{\mu_{\boldsymbol{\Delta}^{T_r}}(\mathbf{M}^{T_r}) \ge k^{T_r}} k^{T_r} - 150 \right) \le 0
\end{aligned}
\tag{5.21}
$$

The variables $\breve{P}_r$, $\breve{L}_r$ and $\breve{T}_r$ are the reactor pressure, liquid level and temperature set points, respectively. As in (5.18), the minimum liquid level allowed in the reactor was reduced from 50% to 30% to enlarge the problem's feasible region. To evaluate (5.21), robust FIR models from each disturbance (υ) to P_r, L_r and T_r, are required. The procedure to obtain such models was given in Section 5.1.1. Moreover, the values for $\breve{P}_r$, $\breve{L}_r$ and $\breve{T}_r$ are determined from optimization, i.e. they are included in the decision variables vector **η** specified for this scenario.

The constraints shown in (5.18)-(5.21) represent the process feasibility constraints that were considered in the present scenario. It should be noted from (5.19)-(5.21) that constraints with the same process parameter uncertainty subscript ($\boldsymbol{\omega}_j$) can be evaluated simultaneously, i.e. there is no need to re-calculate the bounds (k) for each constraint since they all have a common worst-case scenario as defined by the same values of **υ** and **ω**. On the other hand, each problem with a different $\boldsymbol{\omega}_j$ must be solved independently.

Optimization problem for Scenario I.-

The mathematical formulation presented in (5.8) was applied to perform the simultaneous design and control of the TE process when only changes in the reactor's capacity are considered. Thus, the annualized cost function (5.17), the worst-case process variability (5.15) and the process feasibility constraints (5.18)-(5.21) specified above for the TE problem are combined into one optimization problem as follows:

$$\min_{\boldsymbol{\eta}} \quad CF_{TE} \qquad \text{…Cost Function (5.17)}$$

s.t.

$$\varphi^{G} = \max_{\boldsymbol{\omega}^{l} \le \boldsymbol{\omega} \le \boldsymbol{\omega}^{u}} \left(\breve{G} + \max_{\mu_{\Delta^{G}}(\mathbf{M}^{G}) \ge k^{G}} k^{G} - G^{*} \right)$$

$$\varphi^{H} = \max_{\boldsymbol{\omega}^{l} \le \boldsymbol{\omega}_{1} \le \boldsymbol{\omega}^{u}} \left(\breve{H} + \max_{\mu_{\Delta^{H}}(\mathbf{M}^{H}) \ge k^{H}} k^{H} - H^{*} \right) \text{…Worst - case proc. var. (5.15)}$$

Products' mass ratio constraints (5.18)

Products' mass flow rates constraints (5.19) (5.22)

Products' variability constraints (5.20)

Reactor operational constraints (5.21)

$\boldsymbol{\eta}^{l} \le \boldsymbol{\eta} \le \boldsymbol{\eta}^{u}$

TE Process model equations (5.10)

17 PI controller equations (Table 5.3)

The vector of decision variables $\boldsymbol{\eta}$ is composed of the reactor's design capacity (ft^3), the PI controller tuning parameters (Kc and τ_I) of loops 5, 11, 12, 16 and the set points for the loops 8, 11, 12 and 16, listed in Table 5.3 respectively. The TE process model equations together with the 17 PI controller equations describe the closed-loop TE process model, represented here by closed-loop robust FIR models. These robust models are used to test process variability and process constraints using the robust variability tests presented above. Although is not explicitly shown in (5.22), the bounds obtained for the worst-case scenario between subsequent iterations are used to determine if it is necessary to evaluate the system's stability at the current iteration point according to the procedure outlined in Section 5.1.2. The procedure followed to solve (5.22) was presented in Section 5.1.4. Also, the problem was solved using the MATLAB's Optimization Toolbox solvers available for nonlinear constrained optimization problems[55]. The present work used the function *'fmincon'* to solve the optimization problem (5.22). This function uses the Sequential Quadratic Programming (SQP) as the solution algorithm. This function implements a BFGS-based Quasi-Newton approximation for the Hessian estimation of the problem's Lagrangian function[54]. The Hessian and the first derivative of the Lagrangian function are used to pose a quadratic programming sub-problem for each new set of decision variables ($\boldsymbol{\eta}$ in 5.22) tested. This quadratic problem is solved using an active set strategy[73] that returns the search direction that

is expected to minimize the problem's Lagrangian function. The new search direction vector is relaxed by a step length parameter determined from a merit function[74]. The new estimates in the decision variables set is obtained from a line search using the current decision variables values, the current set direction vector and its corresponding step length parameter. Further details about the implementation of the SQP optimization strategy in MATLAB can be found in elsewhere[54-55].

The formulation posed for Scenario I in (5.22) represents a challenging computationally intensive optimization problem. Depending on the values specified for the vector **η**, one complete cost function evaluation that includes the process constraints evaluation and the stability check requires a CPU time of approximately 12,600 seconds. In order to provide an insight with respect to the design and control trade offs for this plant, the present work simplified the formulation presented in (5.22) by relaxing the process parameter uncertainty definition specified for this scenario. Also, the current research work analyzed the TE plant's simultaneous design and control under different assumptions and using different solution strategies. Therefore, the following subsections discuss the simplifications made to Scenario I's formulation shown in (5.22) and present the different design and control schemes obtained for each one of the scenarios studied in this research work.

5.3.1 Effect of external perturbations

The design and controller tuning parameters obtained from the application of the current methodology to a large-scale process are only valid for the set of bounded disturbances variables and bounded process parameter uncertainty specified in the problem. Since the problem was significantly more complex than the examples solved in the previous chapters, a first step for the study of the TE plant consisted of studying the effect of adding disturbances into the design and control problem and assuming that *all* the process parameters' true values are perfectly known. Thus, the process parameter uncertainty (**ω**) was not included in this first phase of the analysis. This last assumption simplifies Scenario I's problem specified in (5.22) because all the maximization problems specified for **ω** are removed from the formulation. Accordingly, the worst-case process variability function defined in (5.22) is specified for the present analysis as follows:

$$\varphi^G = \breve{G} + \max_{\mu_{\Delta^G}\left(\mathbf{M}^G\right) \geq k^G} k^G - G^*$$
$$\varphi^H = \breve{H} + \max_{\mu_{\Delta^H}\left(\mathbf{M}^H\right) \geq k^H} k^H - H^* \tag{5.23}$$

The remaining of the functions that include the variable **ω** in (5.22) are redefined as in (5.23).This simplification clearly reduces the necessary calculations. For instance, the present analysis approximately required only 420 seconds of CPU time to perform one complete function evaluation whereas the formulation in (5.22) required 3.5 hours of CPU time to perform the same calculation. Thus, the removal of **ω** from the analysis greatly reduces the computational burden associated with Scenario I's calculations.

5.3.1.1 Scenario I-A: Changes in the A feed composition, stream 4

As it was pointed out in Section 5.2, a set of 20 process disturbances have been specified for the TE process. The present work selected as a disturbance for this process random variations in the A, B, and C feed composition in stream 4 (see Figure 5.4), listed as idv-8 in Table B.5, Appendix B. This disturbance was selected because it is considered as one of the most difficult disturbance to reject by using the decentralized control strategy[63]. Table 5.4 shows the disturbance specifications used for the present analysis. Although the components' rates of change appear to be small, they have a significant effect on the products' flow rate variability as it will be shown later on this subsection. To study the disturbances effects only on the simultaneous design and control of the TE process, Scenario I's optimization problem (5.22) was initially solved considering only changes in the A and C feed composition of stream 4 while maintaining the B composition constant and equal to its nominal value. To avoid confusion, this problem is referred from heretofore as Scenario I-A.

Table 5.4 Disturbance specifications

Component	Nominal value	Lower bound ($\boldsymbol{\upsilon}^l$)	Upper bound ($\boldsymbol{\upsilon}^u$)	Rate of change ($\boldsymbol{\delta\upsilon}$)
A	0.485	0.475	0.495	±0.01
B	0.005	0.002	0.008	±0.003
C	0.510	0.480	0.540	±0.03

To satisfy stream 4's mole fraction balance, the sum of the mole fractions of the components A, B and C in this stream has to be equal to the unity. Since component B was assumed constant, specifying one component's mole fraction also specifies the other. Thus, the present analysis assumed a magnitude bounded random variation in component's A mole fraction as the sole external perturbation affecting this process. Clearly, the changes in the C mol fraction composition due to the random variations in the A mol fraction are also included in the analysis since they are also affecting the plant's dynamic behaviour.

Therefore, the original Scenario I's optimization problem presented in (5.22) was reduced, following the assumptions presented above, to the formulation shown in (5.24).

Since only one disturbance is included in the analysis, i.e. n_{υ}=1, 2 robust FIR models have to be identified between υ_A to the products' mass flow rate variability G and H to estimate the worst-case process variability in terms of the variability in the resulting production rates. Similarly, 4 robust FIR models are required to evaluate the process constraints specified for Scenario I-A. Consequently, a total of 6 robust FIR models have to be obtained for the complete evaluation of Scenario I-A's optimization problem specified in (5.24). These models are obtained following the procedure outlined in Section 5.1.1. As discussed earlier, the robust FIR models are only valid around an operating state determined by the steady-state solution of the TE process model equations and by the decision variables vector $\boldsymbol{\eta}$, which values are updated by the optimization algorithm. Scenario I-A was programmed in MATLAB® and solved using the built-in function *fmincon* available in the MATLAB's Optimization toolbox[55]. The optimization was initialized using the current reactor's design capacity (1,300 ft^3) and the values given in Table 5.3 for the remaining decision variables specified in $\boldsymbol{\eta}$. In addition, the upper and lower values specified for Scenario I-A's decision variables ($\boldsymbol{\eta}^u$ and $\boldsymbol{\eta}^l$) are given in Table 5.7. These values were obtained from the simulation of the closed-loop TE process.

$$
\begin{aligned}
&\min_{\boldsymbol{\eta}} \quad CF_{TE} \\
&\text{s.t.} \\
&\varphi^{G} = \breve{G} + \max_{\mu_{\Delta^G}\left(\mathbf{M}^{G}\right)\geq k^{G}} k^{G} - G^{*} \\
&\varphi^{H} = \breve{H} + \max_{\mu_{\Delta^H}\left(\mathbf{M}^{H}\right)\geq k^{H}} k^{H} - H^{*} \\
&xm_{G} - 1.02xm_{H} \leq 0 \\
&0.98xm_{H} - xm_{G} \leq 0 \\
&0.98 - xm_{G} - xm_{H} \leq 0 \\
&G^{*} - \breve{G} + \max_{\mu_{\Delta^G}\left(\mathbf{M}^{G}\right)\geq k^{G}} k^{G} \leq 0 \\
&H^{*} - \breve{H} + \max_{\mu_{\Delta^H}\left(\mathbf{M}^{H}\right)\geq k^{H}} k^{H} \leq 0 \\
&-0.05\left(\breve{G} + \breve{H}\right) + \max_{\mu_{\Delta^G}\left(\mathbf{M}^{G}\right)\geq k^{G}} k^{G} + \max_{\mu_{\Delta^H}\left(\mathbf{M}^{H}\right)\geq k^{H}} k^{H} \leq 0 \\
&-0.05\breve{x}_{G} + \max_{\mu_{\Delta^{x_G}}\left(\mathbf{M}^{x_G}\right)\geq k^{x_G}} k^{x_G} \leq 0 \\
&\breve{P}_{r} + \max_{\mu_{\Delta^{P_r}}\left(\mathbf{M}^{P_r}\right)\geq k^{P_r}} k^{P_r} - 2895 \leq 0 \\
&\breve{L}_{r} + \max_{\mu_{\Delta^{L_r}}\left(\mathbf{M}^{L_r}\right)\geq k^{L_r}} k^{L_r} - 100 \leq 0 \\
&30 - \breve{L}_{r} + \max_{\mu_{\Delta^{L_r}}\left(\mathbf{M}^{L_r}\right)\geq k^{L_r}} k^{L_r} \leq 0 \\
&\breve{T}_{r} + \max_{\mu_{\Delta^{T_r}}\left(\mathbf{M}^{T_r}\right)\geq k^{T_r}} k^{T_r} - 150 \leq 0 \\
&\boldsymbol{\eta}^{l} \leq \boldsymbol{\eta} \leq \boldsymbol{\eta}^{u} \\
&\text{TE Process model equations (5.13)} \\
&\text{17 PI controller equations (Table 5.3)}
\end{aligned}
\tag{5.24}
$$

Table 5.5 summarizes the decision variables values and the plant's costs obtained from the optimization of Scenario I-A's formulation. As shown in this Table, the resulting reactor's design capacity is larger than the reactor capacity currently specified for the TE process. The resulting design shows that the operating costs are significantly larger than the annualized capital costs. This result still matches the situation for the current design of the process. For example, the TE plant specified by Ricker's design[63] has a capital and an operating cost of 0.087 MM$/yr and 0.999 MM$/yr, respectively. Although the present scenario specified a plant that have a higher annualized capital cost, the plant's steady-state costs, i.e. capital are operating costs, were reduced by 2.2 % if compared to the costs specified by Ricker's design.

Therefore, scenario I-A's design and control scheme presents a more economically attractive design than that specified by Ricker. Furthermore, the results obtained for this scenario show that TE process economics is predominantly dominated by the operational costs since they represent 60% of the plant's annualized costs specified for Scenario I-A. This idea will be corroborated with the results obtained from the scenarios proposed in the next subsections.

Table 5.5 Design parameters, Scenario I-A

Variables	Specification
Reactor's design capacity (ft^3)	2972.40
Reactor's pressure set point (KPa)	2600.00
Reactor's liquid level set point (%)	63.20
Reactor's temperature set point (°C)	117.00
Production set point (m^3/hr)	23.11
Costs breakdowns (MM$/yr)	
Annualized Capital Cost	0.1066
Operating Cost	0.9554
Variability Cost	0.6193
Plant's Annualized Cost (MM$/yr)	1.6813

To validate Scenario I-A's design, the final decision variables values were used to simulate the TE process dynamic behaviour. The input to this simulation was the A feed composition in stream 4 (υ_A), that is the disturbance considered in this scenario. The time-dependent profile used to describe the disturbance behaviour is shown in Figure 5.7 and corresponds to the critical profile that produces the maximum variability in product H. The critical profile in υ_A was obtained from the evaluation of the worst-case process variability for product H at the optimal solution, defined in (5.23) for Scenario I-A. This profile corresponds to the situation where the minimum mass flow rate constraint for product H specified in the problem's formulation is active at the optimal solution. Thus, it is expected that changes in this process variable may produce constraint violations.

As shown in Figure 5.8, Scenario I-A's optimal design parameters satisfies the products specifications even when the process is subject to the critical time profile in υ_A that produces the maximum deviation in product H's mass flow rate with respect to its set point. As shown

in this Figure, the actual product H's variability is near the minimum products limit, i.e. the smallest difference between the product's actual value and its bound is about 0.6% of the product specification (7038 kg/hr). This corroborates the idea that the use of robust FIR models provides tight descriptions of the TE process dynamic behaviour resulting in only a slightly conservative design. This Figure also illustrates that the allowed products' mass flow rate variability (red dashed lines) is satisfied for the worst-case disturbance profile. Similarly, Figure 5.9 and Figure 5.10 show that the resulting design parameters also satisfy the products variability constraints and the reactor operational constraints included in Scenario I-A's optimization problem. Therefore, the simulations have corroborated that the simultaneous design and control methodology applied to the reactor section of the TE process generates a design and control scheme that is robustly stable and feasible in the presence of critical disturbance realizations.

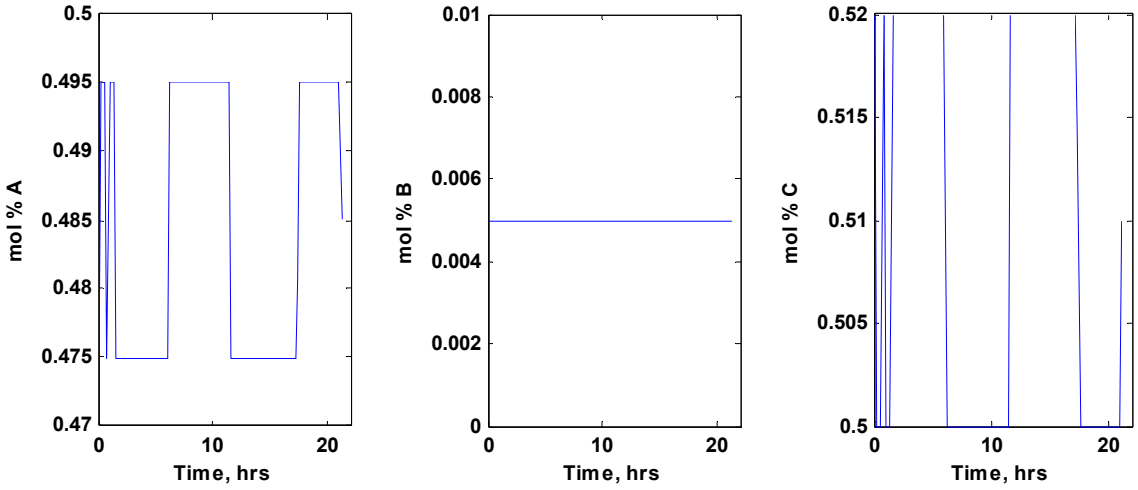

Figure 5.7 Critical disturbance profile, Scenario I-A

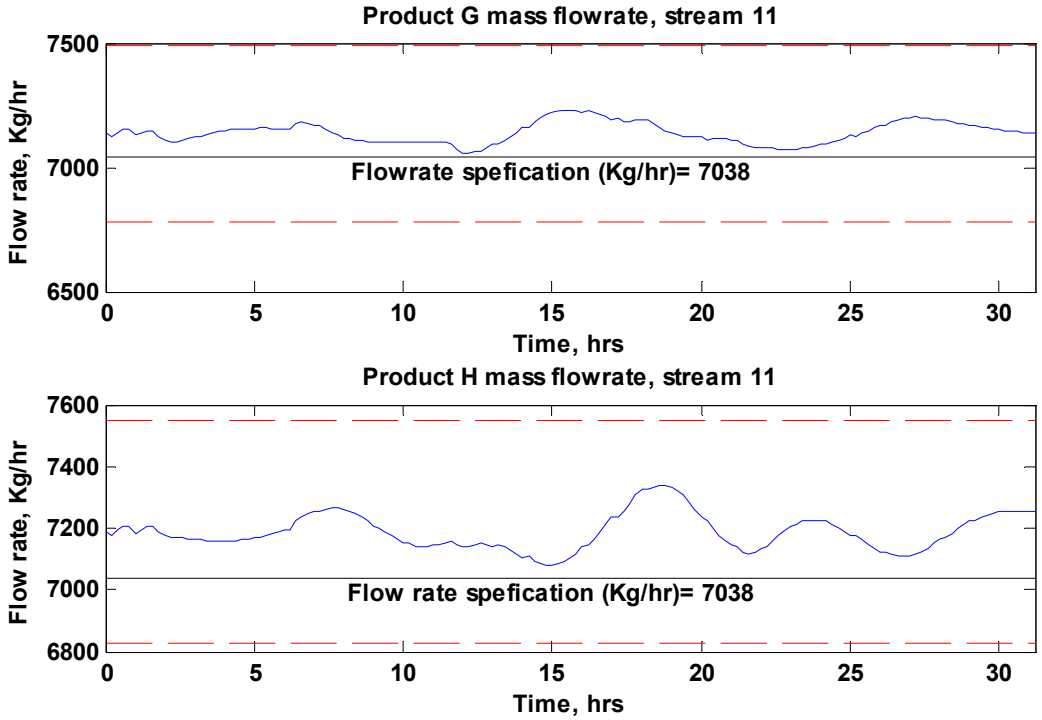

Figure 5.8 Products mass flow rates, Scenario I-A

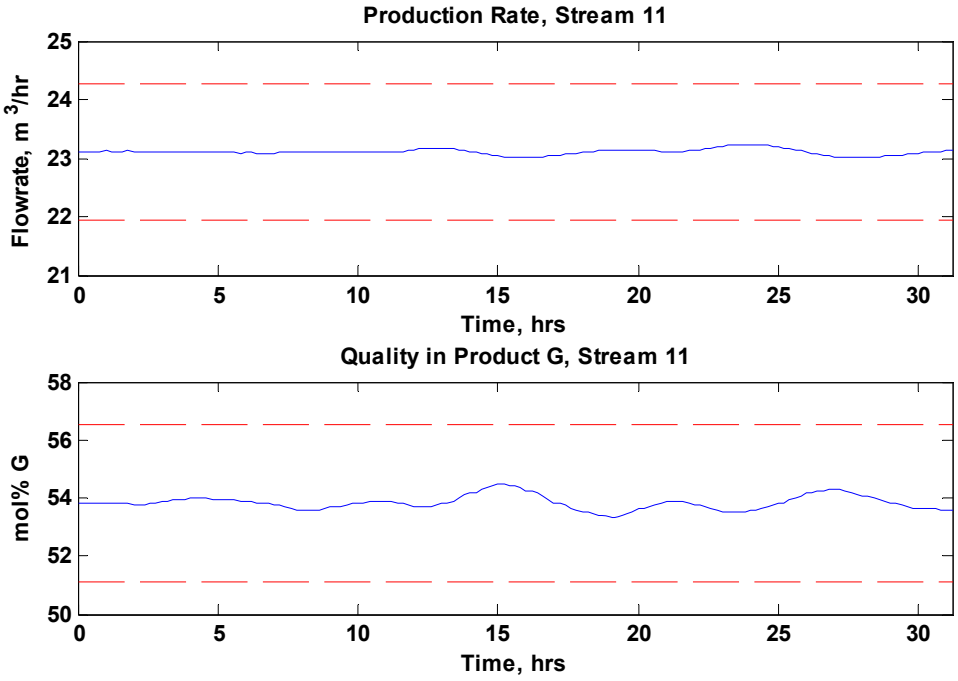

Figure 5.9 Products variability constraints, Scenario I-A

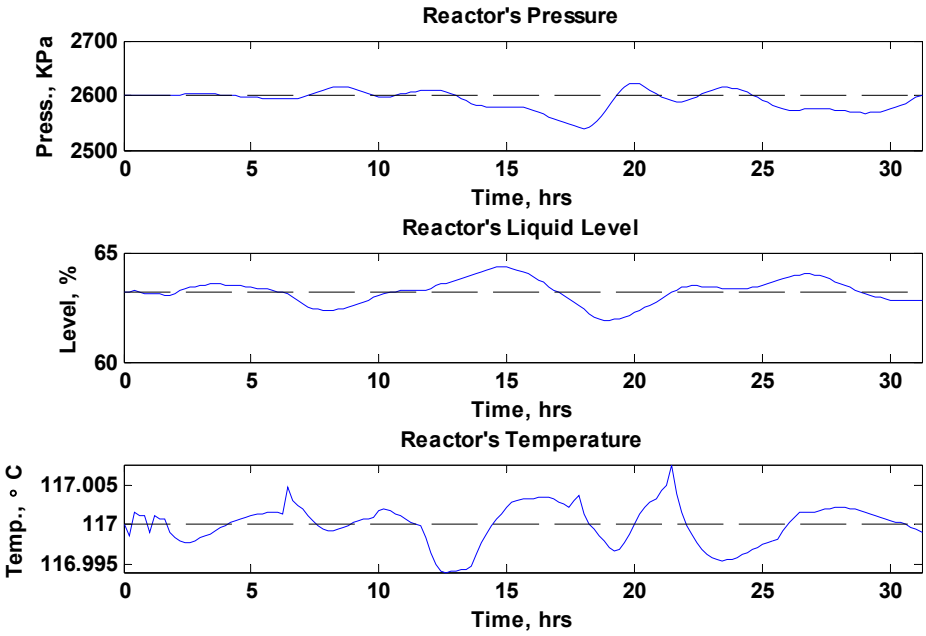

Figure 5.10 Reactor operational constraints, Scenario I-A

5.3.1.2 Scenario I-AB: Changes in the A and B feed composition, stream 4

As discussed in the previous section the mol fraction of component B remained constant during Scenario I-A's analysis. To study the effect of additional disturbances on the simultaneous design and control problem, this second scenario considers the simultaneous design and control of the TE process reactor by considering changes in both A and B feed composition in stream 4. This disturbance is also assumed to be a random magnitude bounded variation in the component B's mole fraction (υ_B). The nominal mol fraction value and the upper and lower mole fraction limits specified for component B are given in Table 5.4. As in the previous scenario perfect knowledge of the process parameters' true values was assumed and the same set ($\boldsymbol{\eta}$) of values used for Scenario I-A was used in the current scenario. This new scenario is referred heretofore as Scenario I-AB.

Since the present scenario only considers the addition of a second disturbance, the formulation specified for Scenario I-AB is the same as that proposed for Scenario I-A given in (5.24). In this case however, there are two disturbances affecting the process, i.e. $n_{\upsilon}=2$ in (5.24). Thus, 4 robust FIR models have to be identified from the disturbance set (υ_A and υ_B)

to the products' mass flow rate variability G and H to estimate the worst-case process variability in the products and 8 robust FIR models are required to evaluate the process constraints. Accordingly, a total 12 closed loop robust models have to be identified during each step of the optimization search to completely evaluate (5.24) when two disturbances are included in the study. Thus, the addition of disturbances in the analysis does not change the problem's formulation but it increases the optimization problem's curse of dimensionality.

As in Scenario I-A, the problem formulated for Scenario I-AB was coded in MATLAB® and solved using Sequential Quadratic Programming-based function *fmincon* available in the MATLAB's Optimization toolbox[55]. The initial guesses for Scenario I-AB's optimization problem were the same as for Scenario I-A, i.e. the reactor's design capacity was initially set to 1,300 ft^3 whereas the controller tuning parameters values for loops 5, 11, 12 and 16 and the set point values for loops 8, 11, 12 and 17 were set to the values shown in Table 5.3. A summary of the resulting design parameters obtained from the optimization of Scenario I-AB's formulation is given in Table 5.6. The results show that the plant's costs increased. This result was expected since the addition of disturbances within the analysis impact the process economics. In this particular case study, it is necessary to increase the plant's annualized cost by approximately 12% to account for the B composition's variability in stream 4.

The results obtained for this scenario corroborate the idea that the operating costs dominate the TE process economics. Particularly, the purge losses accounts for approximately 65% of the process operational costs (see equation B.1, Appendix B). The novelty of the current optimization reside on the result that the reactor's capacity should be further increased to allow for a larger storage of the systems' reactants. Consequently, the operational costs can be reduced since the purge losses are diminished. According to Table 5.6, Scenario I-AB's solution specified a reactor's size that is approximately 130% larger than the current reactor's design capacity. Although this represents an increase in the plant's capital costs, Scenario I-AB's solution set the plant's steady-state cost to 0.684 MM$/yr that is approximately 37% more economical than Ricker's design (1.086 MM$/yr). Therefore, it is logical to increase the reactor's size, and thus the corresponding capital costs, since this increase results in a reduction, by approximately 42%, of the process operational costs.

Table 5.6 Design parameters, Scenario I-AB

Variables	Specification
Reactor's design capacity (ft^3)	3000.00
Reactor's pressure set point (KPa)	2711.00
Reactor's liquid level set point (%)	47.02
Reactor's temperature set point (°C)	117.16
Production set point (m^3/hr)	23.22
Costs breakdowns (MM$/yr)	
Annualized Capital Cost	0.1069
Operating Cost	0.5767
Variability Cost	1.2271
Plant's Annualized Cost (MM$/yr)	1.9107

As in Scenario I-A, the reactor size obtained by the present scenario is larger than the current design specification. Also, the reactor specified for Scenario I-AB is slightly larger than that obtained from Scenario I-A. On the other hand, the specified reactor pressure set point is approximately 4% higher than the set point that was specified by Scenario I-A for this process variable while the reactor liquid level set point obtained for Scenario I-AB is approximately 26% below the set point specified from Scenario I-A's optimization. These results indicate that the omission of disturbances that significantly affects the process economics can result in design and control schemes that in practice may not completely satisfy the production goals and process constraints.

The solution obtained from Scenario I-AB's optimization problem was tested with closed-loop TE process model's simulations. The inputs to the process models were the disturbances selected for this analysis, i.e. the A and B feed compositions (υ_A and υ_B) in stream 4. The critical profiles imposed on both inputs corresponded to the situation where the minimum mass flow rate constraint for product G is active at the optimal solution. Thus, it was reasonable to test the TE process using these disturbance realizations since they were expected to drive the system to its feasibility limits. Figure 5.11 shows the disturbance profiles used to validate Scenario I-AB's design. It should be noticed that the disturbance

realizations obtained for Scenario I-AB (Figure 5.11) are different than those obtained for Scenario I-A (Figure 5.7).

Figure 5.12 illustrate the products mass flow rate variability when the TE process is simulated with stream 4's feed composition profiles shown in Figure 5.11. As shown in the Figure, product G's mass flow rate is very close to its production targets but it always remained above that specification. The profile for the product G's mass flow rate shown in Figure 5.12 is expected to be the worst-case deviation for this product. Likewise, Figure 5.13 illustrates that the variability in the production rate and the quality in product G remain within the operating ranges specified in Scenario I-AB's optimization problem. In addition, Figure 5.14 shows that the reactor operates within the operational constraints specified for this process unit in Scenario I-AB's formulation. Thus the simulations corroborated that the design parameters' values obtained from Scenario I-AB's optimization achieve the process production targets while keeping the process operational conditions within their pre-specified constraints.

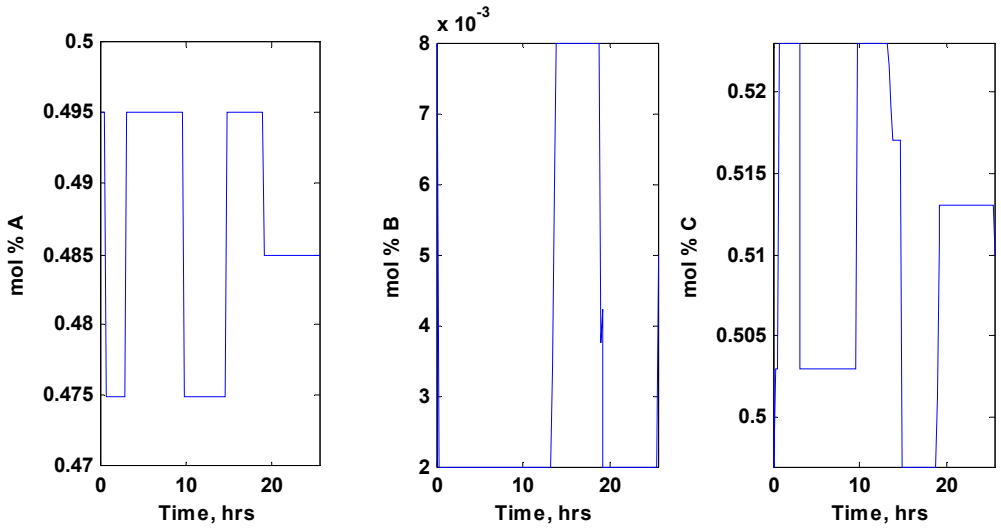

Figure 5.11 Disturbance profiles, Scenario I-AB

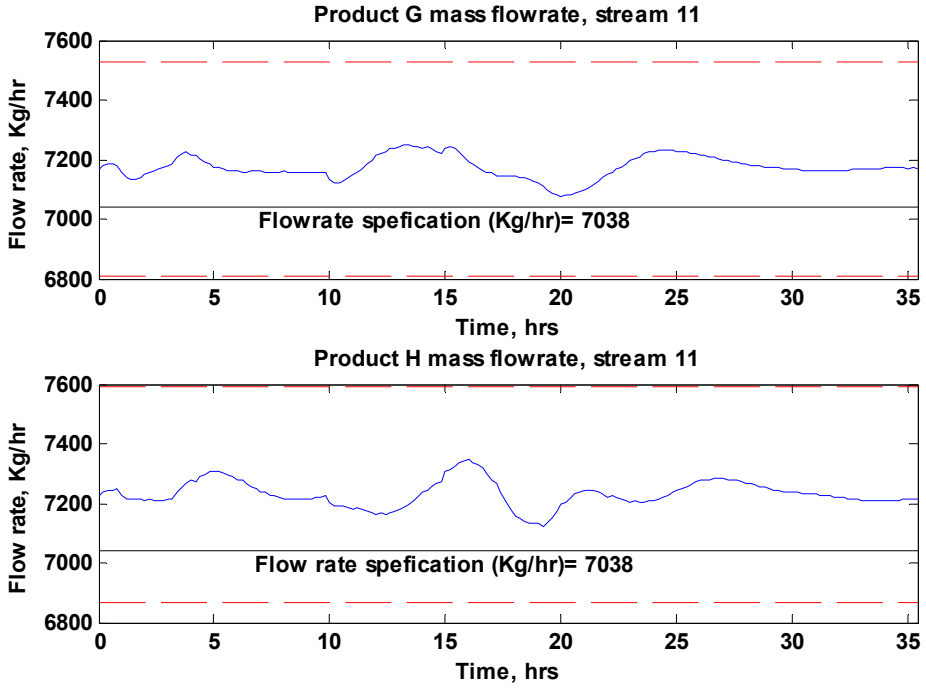

Figure 5.12 Products mass flow rates, Scenario I-AB

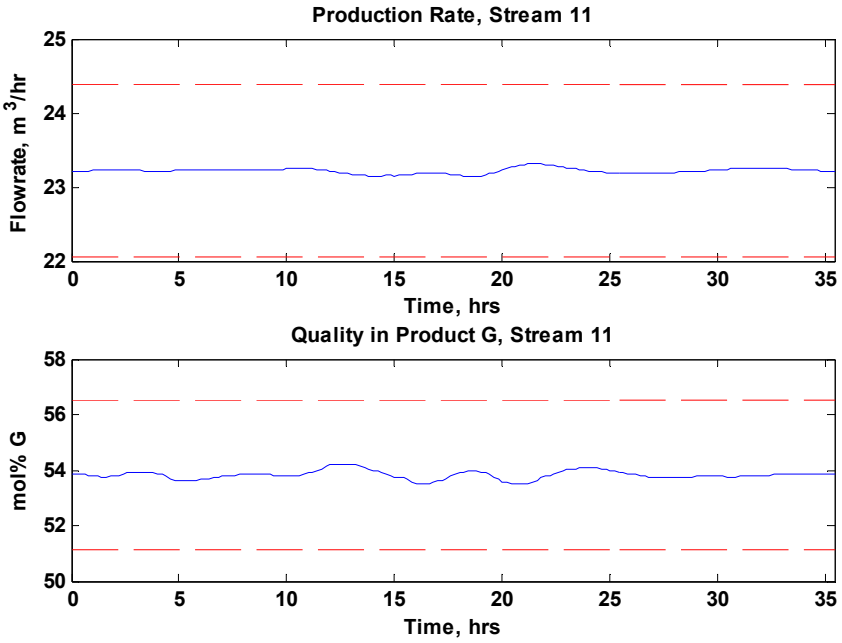

Figure 5.13 Products variability constraints, Scenario I-AB

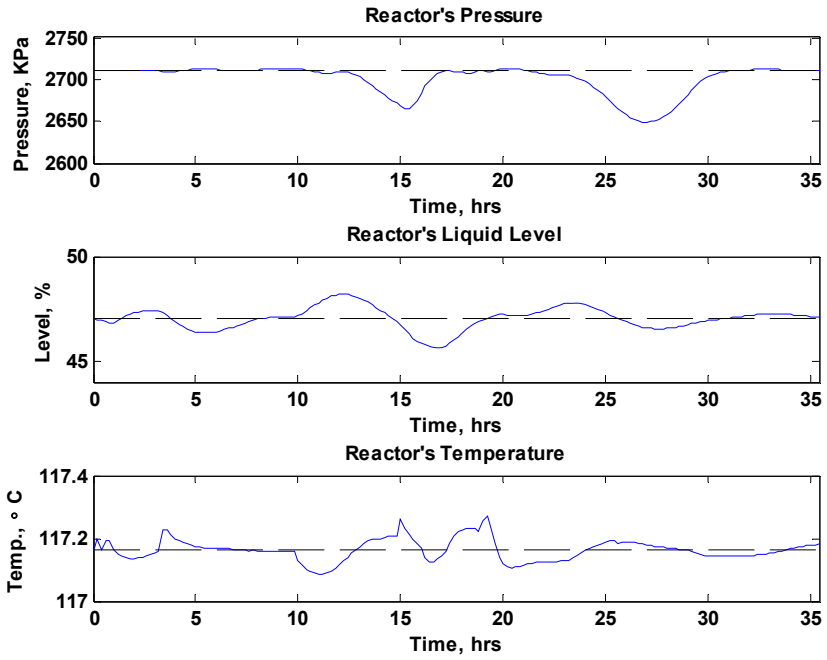

Figure 5.14 Reactor operational constraints, Scenario I-AB

5.3.2 Effect of the initial conditions

As discussed in the previous sections, the formulation shown in (5.24) was solved using the MATLAB's function *fmincon* that uses SQP as the iterative solution method. Since the formulation posed in (5.24) is a nonlinear constrained optimization problem and the problem's functions are not strictly quadratic, it is expected that the solutions obtained for this problem are local solutions only. Moreover, the SQP algorithm is a gradient-based method that highly depends on the initial guess for non-convex problems. Thus, different initial conditions may lead to different solutions.

Based on the above discussion, this section presents the design and control schemes obtained when Scenario I-AB's optimization problem was initialized from different starting points. To conduct this test the optimization problem (5.24) was solved using different initial values for the decision variable vector ($\boldsymbol{\eta}$). Table 5.7 shows the decision variables specified for Scenario I-AB and the set of initial values used to solve Scenario I-AB's mathematical formulation. The first set of values (η_1•) corresponds to the values specified by Ricker's decentralized

control strategy [63]. In the second set of initial values (η_2•), the reactor's design capacity initial value and the reactor and production rate set points values were set to their corresponding upper values ($\boldsymbol{\eta}^u$ in 5.24) whereas the controller tuning parameter values were defined as in η_1•. The third set of initial values (η_3•) fixed all the decision variable values to their upper values. Similarly, the forth starting point vector (η_4•) set the reactor's design capacity and the reactor and production rate set points to their lower values ($\boldsymbol{\eta}^l$ in 5.24) whereas the controller tuning parameters' initial values were defined as in η_1•. The fifth set of initial conditions (η_5•) specified the decision variable vector to their lower values. The results obtained from the previous tests stated that different combinations in the reactor's design capacity and reactor and production set points may produce a similar plant cost. Thus, the bounds and the worst-case variability observed from the previous tests on these decision variables were used to define the sixth and the seventh vector of decision variables values (η_6• and η_7•). In addition, the values specified for $\boldsymbol{\eta}^l$ and $\boldsymbol{\eta}^u$ were obtained from simulations of the TE process' closed-loop behaviour.

Table 5.7 Initial starting points, Scenario I-AB

Decision variables ($\boldsymbol{\eta}$)	$\boldsymbol{\eta}_1$•	$\boldsymbol{\eta}_2$•	$\boldsymbol{\eta}_3$• ($\boldsymbol{\eta}^u$)	$\boldsymbol{\eta}_4$•	$\boldsymbol{\eta}_5$• ($\boldsymbol{\eta}^l$)	$\boldsymbol{\eta}_6$•	$\boldsymbol{\eta}_7$•
Reactor's design capacity (ft^3)	1300.0	3000.0	3000.0	1000.0	1000.0	3000.0	1000.0
Reactor's pressure set point (KPa)	2800.0	2900.0	2900.0	2600.0	2600.0	2884.0	2600.0
Reactor's liquid level set point (%)	65.0	70.0	70.0	38.0	38.0	40.0	70.0
Reactor's temperature set point (°C)	122.9	125.0	125.0	117.0	117.0	118.0	125.0
Production set point (m^3/hr)	22.89	24.0	24.0	21.0	21.0	23.1	21.0
Kc, purge valve (loop 5)	0.01	0.01	0.10	0.01	0.005	0.01	0.01
τ_I, purge valve (loop 5)	0.001	0.001	0.001	0.001	$5x10^{-4}$	0.001	0.001
Kc, reactor liquid level(loop 11)	0.8	0.8	1.5	0.8	0.6	0.8	0.8
τ_I, reactor liquid level (loop 11)	60.0	60.0	80.0	60.0	60.0	60.0	60.0
Kc, reactor pressure (loop 12)	$-1x10^{-4}$	$-1x10^{-4}$	$-5x10^{-5}$	$-1x10^{-4}$	$-1x10^{-3}$	$-1x10^{-4}$	$-1x10^{-4}$
τ_I, reactor pressure (loop 12)	20.0	20.0	60.0	20.0	15.0	20.0	20.0
Kc, reactor temperature (loop 16)	-8.0	-8.0	-1.0	-8.0	-13.0	-8.0	-8.0
τ_I, reactor temperature (loop 16)	7.5	7.5	15.0	7.5	5.0	7.5	7.5

Table 5.8 presents the final decision variables values obtained by solving Scenario I-AB's optimization problem with the set of initial points specified in Table 5.7. As shown in Table

5.8, each initial guess lead to a different optimal solution. However, the results show that some consistent trends and correlations can be found between the different optimal solutions.

Table 5.8 Effect of the starting point on the solution, Scenario I-AB

	$\boldsymbol{\eta}_1$*	$\boldsymbol{\eta}_2$*	$\boldsymbol{\eta}_3$*	$\boldsymbol{\eta}_4$*	$\boldsymbol{\eta}_5$*	$\boldsymbol{\eta}_6$*	$\boldsymbol{\eta}_7$*
Decision variables ($\boldsymbol{\eta}$)							
Reactor's design capacity (ft^3)	3000.0	2286.4	3000.0	2414.1	2919.4	2828.4	1903.9
Reactor's pressure set point (KPa)	2711.0	2892.6	2797.9	2859.4	2600.0	2885.2	2854.7
Reactor's liquid level set point (%)	47.0	65.4	70.0	53.5	58.9	59.1	56.0
Reactor's temperature set point (°C)	117.2	121.4	123.9	120.3	120.9	123.9	122.1
Production set point (m^3/hr)	23.2	23.6	23.6	23.3	23.4	23.5	23.4
Kc, purge valve (loop 5)	3.45E-02	9.89E-02	3.92E-02	4.58E-02	2.88E-02	9.57E-02	7.83E-02
τ_I, purge valve (loop 5)	9.75E-04	3.55E-03	9.11E-03	7.48E-03	1.92E-03	9.49E-03	6.58E-04
Kc, reactor liquid level (loop 11)	1.44	0.63	0.60	1.49	1.48	1.48	1.42
τ_I, reactor liquid level (loop 11)	77.33	67.63	75.57	64.11	61.25	78.55	65.35
Kc, reactor pressure (loop 12)	-2.22E-04	-9.31E-04	-8.87E-04	-8.07E-04	-6.65E-04	-6.71E-04	-1.16E-04
τ_I, reactor pressure (loop 12)	54.03	33.30	15.00	59.60	27.27	56.06	37.95
Kc, reactor temperature (loop 16)	-1.50	-12.89	-7.24	-7.43	-18.18	-8.76	-4.14
τ_I, reactor temperature (loop 16)	15.00	14.76	9.03	7.17	14.13	13.38	1.67

Table 5.8 continues

	η_1*	η_2*	η_3*	η_4*	η_5*	η_6*	η_7*
Constraints[a] evaluated at η*							
Constraint 1	-0.0142	-0.0138	-0.0146	-0.0142	-0.0145	-0.0146	-0.0132
Constraint 2	-0.0058	-0.0062	-0.0055	-0.0059	-0.0056	-0.0055	-0.0068
Constraint 3	-0.0158	-0.0154	-0.0159	-0.0126	-0.0137	-0.0125	-0.0195
Constraint 4	-0.0036	-64.1930	-28.3100	-0.3665	-3.2143	-0.0124	-26.6240
Constraint 5	-0.5575	-24.7380	-43.4590	-27.4110	-34.7430	-49.5570	-52.3469
Constraint 6	-401.66	-327.49	-347.59	-429.68	-431.97	-403.94	-396.75
Constraint 7	-4.39	-4.29	-4.46	-4.43	-4.46	-4.52	-4.31
Constraint 8	-178.73	-1.07	-43.92	-30.19	-292.22	-7.06	-30.71
Constraint 9	-51.00	-30.26	-25.84	-44.88	-39.73	-39.34	-42.37
Constraint 10	-15.04	-31.05	-35.84	-21.91	-27.44	-27.53	-24.30
Constraint 11	-32.70	-28.62	-26.05	-29.64	-29.07	-26.04	-27.89
Cost breakdowns (MM$/yr)							
Annualized Capital Cost	0.1069	0.0988	0.1069	0.1003	0.1060	0.1050	0.0943
Operating Cost	0.5767	0.6475	0.6816	0.6162	0.6655	0.6616	0.7108
Variability Cost	1.2271	1.7162	1.5981	1.1714	1.1837	1.3222	1.3025
Plant's Annualized Cost (MM$/yr)	1.9107	2.4624	2.3867	1.8878	1.9553	2.0888	2.1076

a The constraints are listed according to order in which they appear in problem (5.24).

From the steady-state point of view, it was consistently found that the reactor should operate at a high pressure since the products are produced at a higher reaction rate and the purge losses are diminished. This same effect is observed when the liquid level in the reactor is set to a lower value. Therefore, two different combinations of the liquid level in the reactor and the reactor's pressure can in principle satisfy the same production specs. Similarly, setting the nominal reactor temperature to a lower value improves the selectivity of the reactions that produce the products G and H over the reactions that produce the by-product F (see equations 5.9). Moreover, the reactor's design capacity determines the amount of available raw

material that can be stored on this unit to produce the products. Therefore, it is desired to have a reactor with a large capacity that operates at steady-state at the highest possible working pressure with the minimum liquid volume and at the lowest possible working temperature. However, the process operational constraints do not allow the system to operate at this ideal steady-state condition. Moreover, the disturbances affecting the system generate process variability.

Consequently, combinations (trade-offs) between the reactor's design capacity (process design), the reactor's nominal operating point (process operability) and the controllers' tuning parameters (process control) are sought that minimize the plant's cost function and satisfies the production goals and the variability and steady-state constraints specified for this problem.

Based on the above, it was expected that although each starting point will lead to a different solution due to the non-convexity of the problem, the solutions will have a similar annualized cost. However, the results in Table 5.8 show a discrepancy in the plant's costs of about 23%. This variation proves that Scenario I-AB's optimization problem has local minima. Apart from the steady-state constraints (constraints 1-3 in Table 5.8), the variability constraints for which a bound is determined, which are equivalent to the path constraints imposed in a dynamic optimization framework, play a key role in defining the best solution to this particular problem.

From Table 5.8, the solution given resulting from the initial guess $\boldsymbol{\eta}_4$* resulted in the minimal annualized costs but the solutions corresponding to the initial starting points $\boldsymbol{\eta}_1$*, $\boldsymbol{\eta}_5$* and $\boldsymbol{\eta}_6$* resulted in very similar small plant's annualized costs. The constraint that is active at the solution is the same for initial guesses $\boldsymbol{\eta}_1$*, $\boldsymbol{\eta}_4$*,$\boldsymbol{\eta}_5$* and $\boldsymbol{\eta}_6$* , and it corresponds to the minimum allowed mass flow rate for product G (constraint 4 in Table 5.8). This constraint directly determines the plant's variability cost (see 5.15 and 5.16) and thus it directly affects the overall plant's annualized cost. Thus, it is expected that this constraint causes the cost function to be at its minimum value and it cannot be further minimized because this would result in a constraint violation. Consequently, the solutions for which an active constraint is directly involved in the process cost function are more likely to be near-optimal solutions.

One can notice from Table 5.8 that the solutions that have the highest plant's costs are those for which the active constraint are more related to the reactor's operation rather than to the process cost function.

According to Table 5.8, the solution that resulted in the lowest annualized cost was $\boldsymbol{\eta}_4$*. To validate the design, simulations of the closed-loop system were performed using as inputs the critical profiles in υ_A and υ_B (disturbances) that produced the largest variability in product G, which is the active constraint at the solution. Figure 5.15 shows the disturbance profile used to simulate the resulting design. It should be noted that these profiles are somehow similar to those obtained for $\boldsymbol{\eta}_1$* shown in Figure 5.11. As shown in Table 5.8, the solution $\boldsymbol{\eta}_1$* specified an annualized cost only 1.20% higher than that obtained from $\boldsymbol{\eta}_4$*. However, the resulting design parameters obtained from $\boldsymbol{\eta}_1$* and $\boldsymbol{\eta}_4$* are somewhat different. This supports the fact that different combinations between the reactor's design capacity (process design), the reactor's nominal operating point in terms of pressure, temperature, and liquid level (process operability) and the controller tuning parameters (process control) may result in a similar annualized cost for this process. Furthermore, Figure 5.16, Figure 5.17 and Figure 5.18 show that $\boldsymbol{\eta}_4$*'s corresponding design parameters satisfy the production goals, the process variability constraints and the reactor's operational constraints specified for this problem. Thus, for any magnitude bounded perturbations υ_A and υ_B, the design and control scheme specified by $\boldsymbol{\eta}_4$* is asymptotically stable, keeps the process within their corresponding operational constraints and satisfies at all times the base case production targets specified for the TE process. The rest of the solution sets shown in Table 5.8 were validated in the same fashion.

5.3.3 An alternative solution strategy, Scenario I-ABsim

As discussed earlier, the robust performance calculation given by (4.8) provides both a bound on the output's largest variability and the corresponding profile in the perturbation variables that generates such maximal deviation. Scenario I-A and Scenario I-AB have used the bound (k) obtained from this calculation to represent the output's variability. Although the resulting design parameters obtained for both scenarios satisfy the design and control goals, the use of an analytical bound on the output variability may usually leads to conservative results.

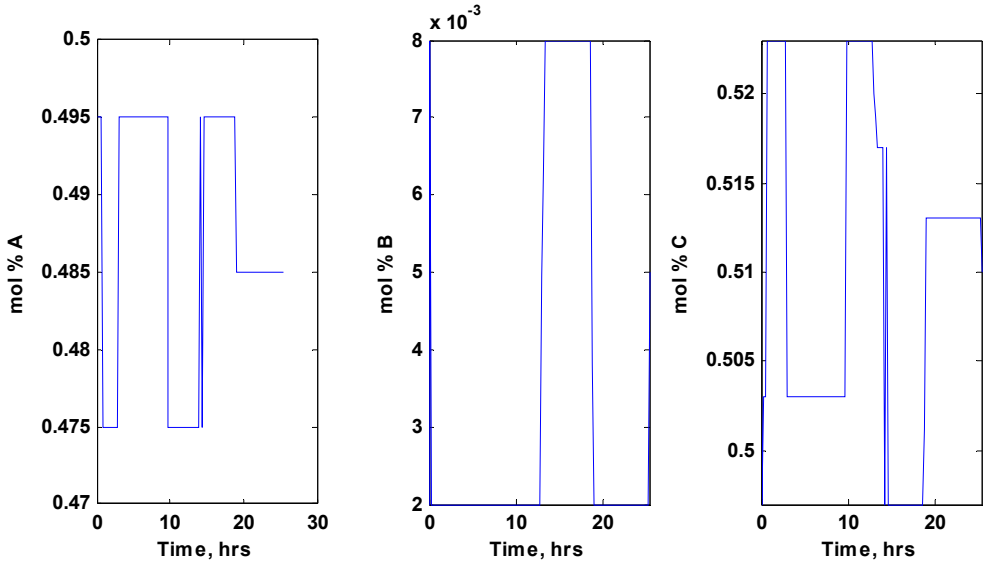

Figure 5.15 Disturbance profiles, Scenario I-AB (η_4*)

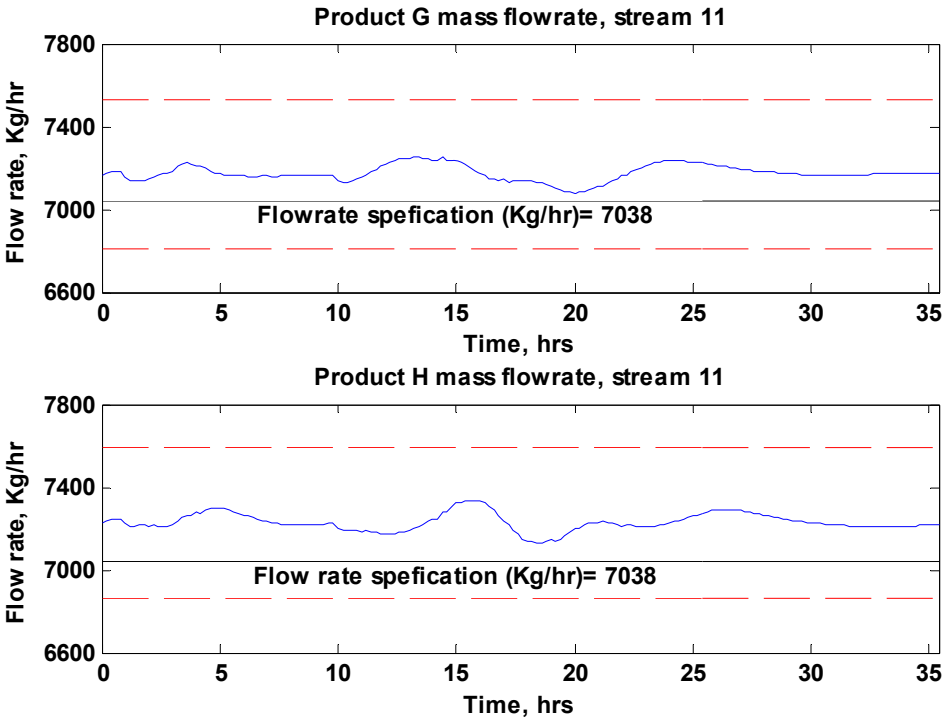

Figure 5.16 Products mass flow rates, Scenario I-AB (η_4*)

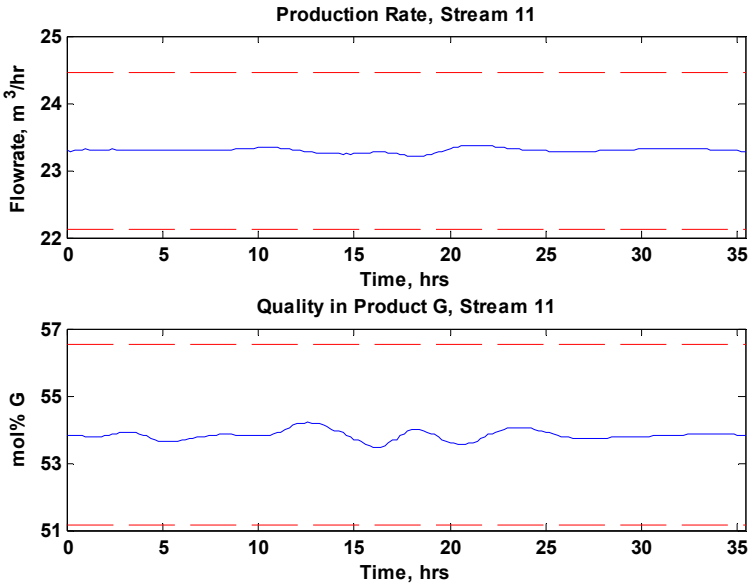

Figure 5.17 Products variability constraints, Scenario I-AB (η_4*)

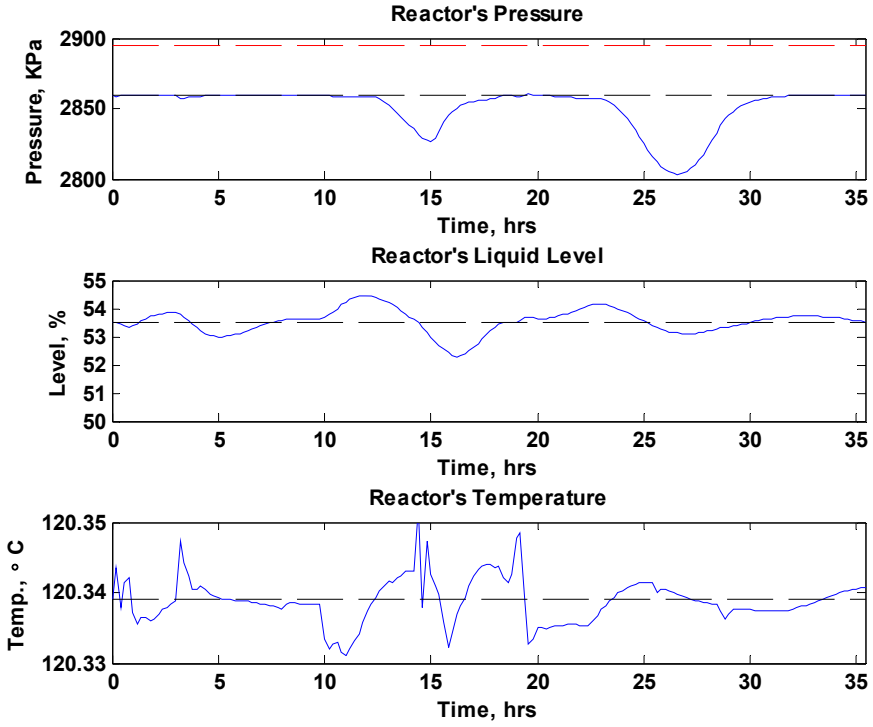

Figure 5.18 Reactor operational constraints, Scenario I-AB (η_4*)

To reduce the conservatism in the solution, the critical profile in the disturbance that generates the output's maximum variability can be used to simulate the closed-loop TE process. In this way an alternative solution to the robust variability calculation is to look for the output's actual largest variability with respect to a nominal value from numerical simulations instead of using the analytical bound on the variability. Although this alternative approach increases the computational load since a simulation have to be run for each robust variability constraint specified in the problem it is expected that it will reduce the conservatism in the solution.

Therefore, Scenario I-AB's problem stated in (5.24) and with the modifications explained in Section 5.3.1.2 was solved using the approach outlined above. For this particular problem, it was necessary to solve six robust performance functions of the type given by (4.8) to evaluate the robust variability constraints shown in (5.24). Thus, for each new set of decision variables values ($\boldsymbol{\eta}$) tested by the optimization algorithm, six critical profiles in υ_A and υ_B are obtained from (4.8) and are used to simulate the TE process. Then, the maximum output variability that corresponds to each critical profile is obtained directly from these numerical simulations. For clarity, this new problem will be referred heretofore as Scenario I-ABsim. This problem was executed in MATLAB® using the Sequential Quadratic Programming-based function *fmincon* that is available in the MATLAB's Optimization toolbox[55]. For comparison purposes, the initial guess that provided the plant's minimum annualized cost for Scenario I-AB ($\boldsymbol{\eta}_4$• in Table 5.7) was used to initialize Scenario I-ABsim's optimization problem.

Table 5.9 summarizes the solution obtained from this problem. As shown, the plant's annualized cost is approximately 10% less than the cost obtained for the design and control scheme specified by $\boldsymbol{\eta}_4$*. It should be noted from Table 5.7 and Table 5.9 that the major cost savings came from variability cost. These results were expected since Scenario I-AB represented the output's variability by an analytical bound, which is potentially conservative, whereas Scenario I-ABsim takes the actual maximum variability values as the output's variability. Thus, the variability cost function was estimated in Scenario I-ABsim considering the actual deviations in the products G and H mass flow rates respectively. Similarly, the constraints stated in Scenario I-ABsim's optimization problem are evaluated more accurately

by using numerical simulation results. Although Scenario I-ABsim's solution specified a larger reactor capacity and a higher steady-state normal operating points for the reactor liquid level and temperature, which resulted in an increase of the capital and operating costs, this design and control scheme resulted in a significantly smaller product's mass flow rate variability at the outlet as compared to the result based on the analytical bound, which significantly reduced the process variability cost. The major disadvantage in using simulations to calculate the variability is that the computation times increase significantly as shown in the following section.

Table 5.9 Alternative solution strategy, Scenario I-ABsim

Decision variables ($\boldsymbol{\eta}$)	**Specification**
Reactor's design capacity (ft^3)	2972.8
Reactor's pressure set point (KPa)	2856.2
Reactor's liquid level set point (%)	70.0
Reactor's temperature set point (°C)	124.92
Production set point (m^3/hr)	23.4
Kc, purge valve (loop 5)	6.86E-02
τ_I, purge valve (loop 5)	5.00E-04
Kc, reactor liquid level(loop 11)	1.13
τ_I, reactor liquid level (loop 11)	78.60
Kc, reactor pressure (loop 12)	-8.21E-04
τ_I, reactor pressure (loop 12)	16.34
Kc, reactor temperature (loop 16)	-19.38
τ_I, reactor temperature (loop 16)	4.15
Cost breakdowns (MM$/yr)	
Annualized Capital Cost	0.1066
Operating Cost	0.6966
Variability Cost	0.8945
Plant's Annualized Cost (MM$/yr)	1.6977

The design parameters obtained by this solution strategy was validated by simulating the TE process with the specifications shown in Table 5.9. Figure 5.19 shows the disturbance profiles that were used to simulate the design. These realizations correspond to the critical profiles in the mol % of A and B in stream 4 that produced the largest variability in product

G's mass flow rate at the solution. Although any of the critical profiles that were estimated at the solution could have been used to validate the present design, the constraint related to the product G's mass flow rate was chosen as the validation scenario since this constraint is active at the solution. This is corroborated in Figure 5.20 where the actual maximum deviation in product G's mass flow rate is equivalent to the minimum allowed product's flow rate. The other set of critical profiles calculated at the solution point were also tested in the same fashion but no constraint violations were observed.

5.3.4 Comparison to a Dynamic Programming approach

One of the goals pursued by the present methodology consists of reducing the computational effort associated with the solution of the simultaneous design and control of chemical processes. To estimate the computational advantages of the proposed approach, the present research work reformulated the problem presented in the previous subsection as a dynamic optimization problem. Accordingly, the μ-analysis problems that were used to estimate the worst-case process variability and the process feasibility constraints in (5.24) are replaced by dynamic optimization problems. Each dynamic programming problem searches for the critical values in the vector of disturbances $\boldsymbol{\upsilon}_A$ and $\boldsymbol{\upsilon}_B$ that produces the maximum deviation of a given output with respect to a nominal value specified by the vector $\boldsymbol{\eta}$. The disturbance vectors $\boldsymbol{\upsilon}_A$ and $\boldsymbol{\upsilon}_B$ consists as before of the values of the mole fraction of components A and B in stream 4, i.e. υ_A and υ_B, at each sampling period i from time=0 to a pre-specified final time t_f. Thus, for each set of values in vector $\boldsymbol{\eta}$ and for each realization in the disturbance vectors $\boldsymbol{\upsilon}_A$ and $\boldsymbol{\upsilon}_B$ tested by the optimization algorithm, the TE process model equations and the controller algorithm equations specified for this process are used to simulate the closed-loop dynamic behaviour of this process. Then, the maximum or largest variability in one particular output is sought from the simulation results.

Based on the above descriptions, Scenario I-AB's optimization problem shown in (5.24) was reformulated as a dynamic optimization problem. This formulation is given in (5.25) and will be referred heretofore as Scenario I-DP.

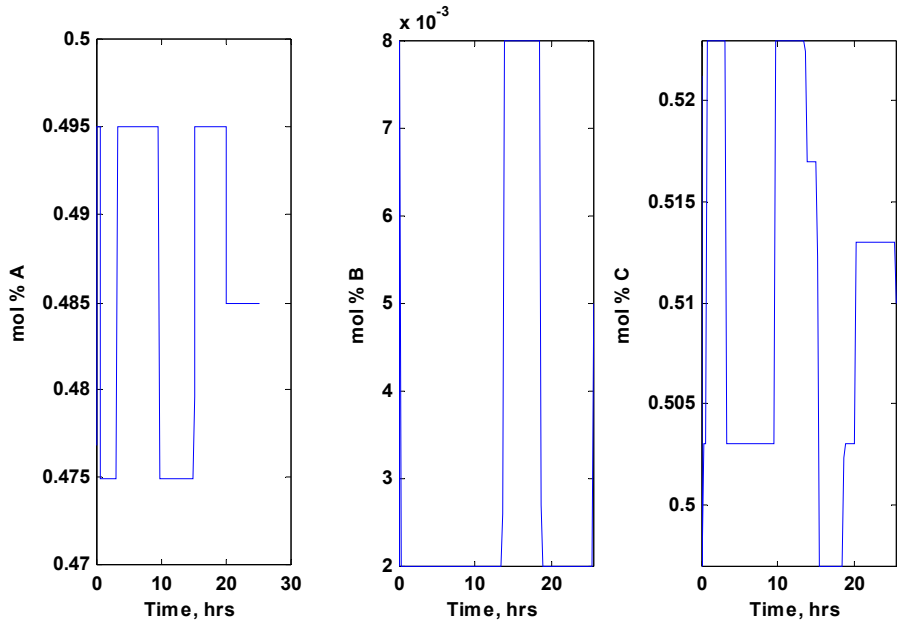

Figure 5.19 Disturbance profiles, Scenario I-ABsim

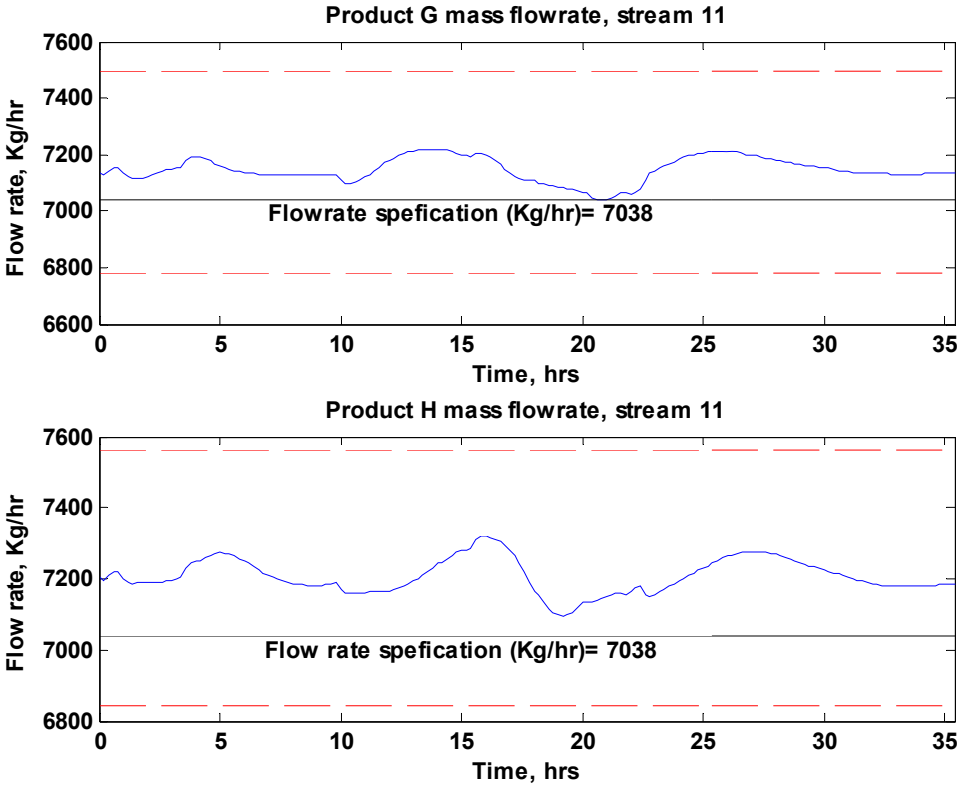

Figure 5.20 Products mass flow rates, Scenario I-ABsim

$$
\begin{aligned}
&\min_{\boldsymbol{\eta}} \quad CF_{TE} \\
&\text{s.t.} \\
&\quad \text{TE Process model equations (5.13)} \\
&\quad \text{17 PI controller equations (Table 5.3)} \\
&\quad \varphi^{G} = \breve{G} + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} G(t) - G^{*} \\
&\quad \varphi^{H} = \breve{H} + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} H(t) - H^{*} \\
&\quad xm_G - 1.02xm_H \le 0 \\
&\quad 0.98xm_H - xm_G \le 0 \\
&\quad 0.98 - xm_G - xm_H \le 0 \\
&\quad G^{*} - \breve{G} + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} G(t) \le 0 \\
&\quad H^{*} - \breve{H} + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} H(t) \le 0 \\
&\quad -0.05\left(\breve{G} + \breve{H}\right) + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} G(t) + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} H(t) \le 0 \\
&\quad -0.05\breve{x}_G + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} x_G(t) \le 0 \\
&\quad \breve{P}_r + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} P_r(t) - 2895 \le 0 \\
&\quad \breve{L}_r + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} L_r(t) - 100 \le 0 \\
&\quad 30 - \breve{L}_r + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} L_r(t) \le 0 \\
&\quad \breve{T}_r + \max_{\substack{\upsilon_A \in \boldsymbol{\upsilon}_{\mathbf{A}}(t) \\ \upsilon_B \in \boldsymbol{\upsilon}_{\mathbf{B}}(t)}} T_r(t) - 150 \le 0 \\
&\quad \boldsymbol{\upsilon}_{\mathbf{A}}(t) = \left\{\upsilon_A(t), \upsilon_A(t)^{l} \le \upsilon_A(t) \le \upsilon_A(t)^{u}\right\} \\
&\quad \boldsymbol{\upsilon}_{\mathbf{B}}(t) = \left\{\upsilon_B(t), \upsilon_B(t)^{l} \le \upsilon_B(t) \le \upsilon_B(t)^{u}\right\} \\
&\quad \boldsymbol{\eta}^{l} \le \boldsymbol{\eta} \le \boldsymbol{\eta}^{u} \\
&\quad t \in \left[0, t_f\right]
\end{aligned}
\tag{5.25}
$$

As shown, the decision variables for each dynamic programming (DP) problem are the values at each sampling period *i* of the disturbance vectors selected for this problem, i.e. $\boldsymbol{\upsilon}_{\boldsymbol{A}}$ and $\boldsymbol{\upsilon}_{\boldsymbol{B}}$. It should be noted that the outer optimization problem is a nonlinear optimization problem since the vector of decision variables $\boldsymbol{\eta}$ is time independent. Therefore, the problem

shown in (5.25) can be solved in MATLAB®. Sequential Quadratic Programming (SQP) was used as the optimization method to solve this problem. Each of the dynamic optimization problems posed in (5.25) was solved in MATLAB applying the following procedure:

1. Given a set of values for the vector $\boldsymbol{\eta}$ and the critical realizations in the disturbance vectors $\boldsymbol{\upsilon}_{\mathbf{A}}$ and $\boldsymbol{\upsilon}_{\mathbf{B}}$, simulate the closed-loop dynamic behaviour of the TE process using the process model equations and the controller algorithm equations specified for this system.

2. The maximum or largest variability in one particular output is sought from the simulation results. Since this value represents the problem's objective function to be maximized, it is used by the optimization algorithm to choose a new set of critical values for the disturbance vectors. Go back to step 1.

3. This procedure is repeated until a stopping criterion is met.

It should be noticed that in this approach, for computation reasons, the stability of the control loops is not explicitly assessed. Since the models are nonlinear such stability checks will require the formulation of multiple Lyapunov-stability calculations that will result in prohibitive computation times.

Scenario I-DP was solved using a sampling period (i) of 0.2 hrs and a finite time horizon (t_f) of 25 hours. Thus, each disturbance vector ($\boldsymbol{\upsilon}_{\mathbf{A}}$ and $\boldsymbol{\upsilon}_{\mathbf{B}}$) is composed of 125 elements that are the decision variables for each of the DP problems posed in (5.25). The optimization problem was initialized using the same starting point that resulted in the lowest annual cost in Scenario I-AB, i.e. η_4• in Table 5.7.

To illustrate the computational loads required by the solution methods proposed in this work and the dynamic programming approach, the CPU time required to evaluate the cost function and the constraints imposed on each of these optimization problems was recorded. Figure 5.21 shows the computational time required to compute the first 100 function evaluations of Scenario I-AB, Scenario I-ABsim and Scenario I-DP, respectively. The Figure shows that the dynamic programming approach requires a CPU execution time of almost 2 orders of magnitude larger as compared to the CPU execution times required by the solution strategies

proposed in this work. On average, Scenario I-AB and Scenario I-ABsim required a CPU time of 157 and 296 seconds respectively to completely evaluate problem (5.24) whereas Scenario I-DP needed 12,710 seconds to perform the same calculation. These results in a CPU time ratio of 1:2:81 in favour of the scenarios that applied the methodology proposed in this research work showing that the computational effort dramatically increased when dynamic programming is applied to solve a simultaneous design and control problem. These results agreed with the results shown in Section 3.3 and Section 4.4 in which the solution of a worst-case variability problem via dynamic programming required a larger computational time than that needed by the approach proposed in this research work.

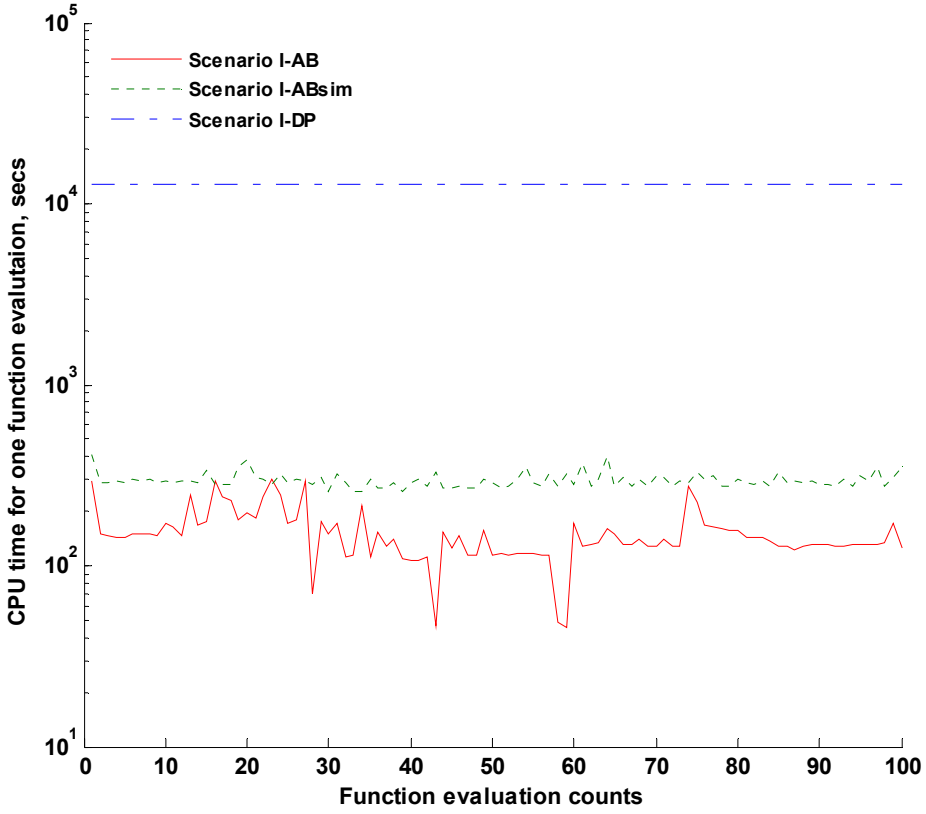

Figure 5.21 Computational burden required by the solution strategies: CPU time for one function evaluation as a function of the number of the completed function evaluations

The CPU time required by Scenario I-DP to compute the first 100 function evaluations was approximately 1.2710×10^6 seconds, i.e. 15 days approximately. Due to time limitations, the final optimization results for Scenario I-DP were not available at the time that this thesis was written. However, it is expected that the use of a dynamic programming approach will be less conservative and consequently it will produce a design with lower annual costs than those obtained for Scenario I-AB and Scenario I-ABsim. However, each one of the dynamic programming problems posed in (5.25) to calculate variability could potentially provide a local solution rather than the global optimum. On the other hand the robust variability problems formulated in (5.24) were shown in Section 4.1 to be convex and consequently result in a global optimum.

5.3.5 Comparison to Ricker's design parameters and optimal control problem

Table 5.3 presented the design parameters that Ricker defined for the TE plant's decentralized control structure[63]. Since the design goals specified by Ricker and the simultaneous design and control goals proposed in the present analysis are different, a complete comparison between the two approaches is not available. On one hand, Ricker's strategy fixed the sizes of the TE process units to its current values, i.e. it only studied the controllability and resiliency of the plant. Similarly, the specifications for the controller tuning parameters and the process unit's nominal operating set points given in Table 5.3 were obtained only from simulations and process heuristics. Moreover, production goals specified in Table 5.2 for each operation mode are only satisfied in an *average sense*, i.e. the design parameters specified by Ricker's strategy meet the production goals at steady-state only. This means that when disturbances are affecting the process, the products specifications at the outlet may not be satisfied at all times. On the other hand, the present work considered changes in the size of the reactor, determined the design and tuning control parameters from a systematic method and imposed constraints on the problem to satisfy at all time the base case production targets in the presence of disturbances.

Figure 5.22 shows the products' mass flow rate variability when the critical profiles shown in Figure 5.19 were used as inputs to simulate the TE process with the design parameters specified by Ricker given in Table 5.3. This Figure also shows the plant's costs specified by

Ricker's design[63]. Although Ricker's plant is 6.6% more economical than the plant specified with Scenario I-ABsim (see Table 5.9), both products do not satisfy the minimum products' mass flow rate specification. As shown in Figure 5.22, products G and H are below their production targets 53% and 52% of the time respectively. Thus, Ricker's design cannot guarantee a minimum production specification. However, it should be noted that Ricker specified the design parameters to satisfy his own design goals that were different than those proposed in this case study. Although the present methodology was defined to consider the simultaneous design and control of a plant, it can also be adapted to solve an optimal control problem by means of setting the equipment's capacities to a fixed value. For comparison purposes, the present research work solved an optimal control problem for the TE plant designed by Ricker assuming that the reactor's capacity was fixed to its original value equal to 1,300ft^3.

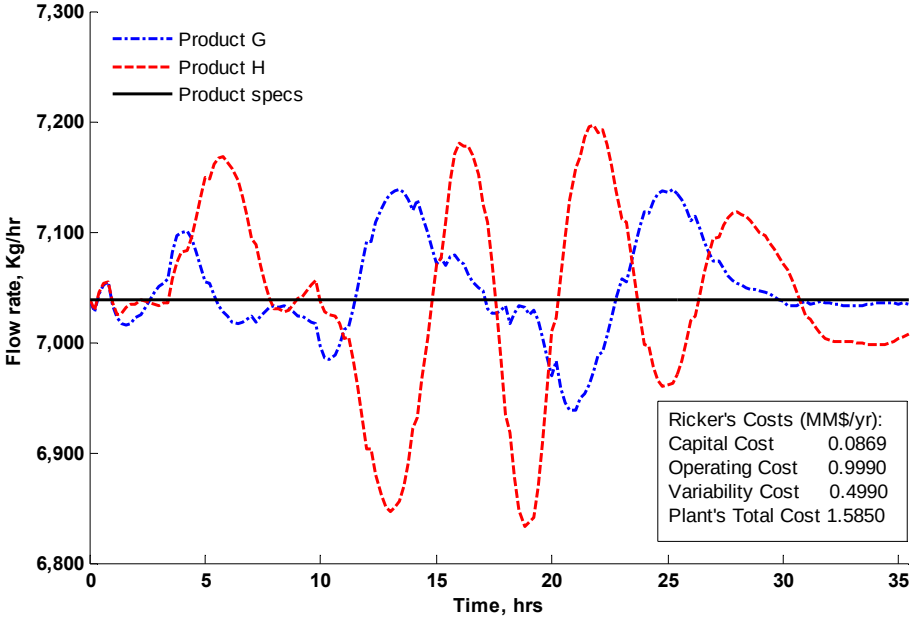

Figure 5.22 Products' variability, Ricker's design parameters.

Based on the above, the alternative solution strategy presented in the Section 5.3.3 was applied to solve this optimcal control problem. The only change that has to be made with respect to Scenario I-ABsim problem is that the reactor design capacity, specified as an optimization variable, is defined as a fixed parameter for the optimal control problem. Thus,

the vector $\boldsymbol{\eta}$ in (5.24) is composed of 12 of decision variables for this problem instead of 13 that were specified for Scenario I-ABsim and the optimization will only determine the values in the reactor's operability parameters (process operability) and in the controller tuning parameters (process control) that minimize the TE cost function shown in (5.24). As in the previous scenarios, this problem was solved using the MATLAB® tools. In addition, this optimization problem was initialized with the specifications given in Table 5.3, which correspond to the design parameter values specified by Ricker's strategy.

Table 5.10 presents the solution obtained from the optimal control problem, with the reactor's capacity fixed to 1,300 ft^3. The process operability and controllability parameters presented in Table 5.10 defined a plant with an annual cost that is approximately 11.54% higher than annual cost of the plant obtained by Scenario I-ABsim shown in Table 5.9. Thus, a more economically attractive design was obtained when both process design and process control, were performed simultaneously as in scenario I-ABsim. This highlights the fact that the interactions between design and control must be taken into account at the earlier stages of the design since they significantly affect the plant's economics. For this particular problem, the simultaneous design and control problem (Scenario I-ABsim) defined a plant which annual cost is $221,300 less than that obtained from the optimal control problem only. The major savings resulting from the simultaneous design and control approach comes from the operating costs, i.e. $178,000/yr. These results support the idea presented in Section 5.3.2 that different combinations or trade-offs between the reactor's design capacity and the process operability and controllability parameters can lead to significant costs savings.

Figure 5.23 shows the critical profiles in the mol% of components A, B and C in stream 4 that were used as inputs to simulate the TE plant using the results shown in Table 5.10. These profiles correspond to the realizations that produced the largest variability in product H, which is the active constraint at the solution point. As shown in Figure 5.24 the profile is bounding the minimum product H's mass flow rate variability constraint. The rest of the process constraints specified for the optimal control problem were also satisfied at all time, e.g. reactor's operability constraints.

Table 5.10 Solution, Optimal control problem

Decision variables (**η**)	**Specification**
Reactor's pressure set point (KPa)	2893.83
Reactor's liquid level set point (%)	38.16
Reactor's temperature set point (°C)	125.0
Production set point (m^3/hr)	23.32
Kc, purge valve (loop 5)	1.76E-02
τ_I, purge valve (loop 5)	7.74E-04
Kc, reactor liquid level(loop 11)	1.5
τ_I, reactor liquid level (loop 11)	75.63
Kc, reactor pressure (loop 12)	-1E-03
τ_I, reactor pressure (loop 12)	59.52
Kc, reactor temperature (loop 16)	-8.42
τ_I, reactor temperature (loop 16)	15
Cost breakdowns (MM$/yr)	
Annualized Capital Cost	0.0869
Operating Cost	0.8742
Variability Cost	0.9578
Plant's Annualized Cost (MM$/yr)	1.9190

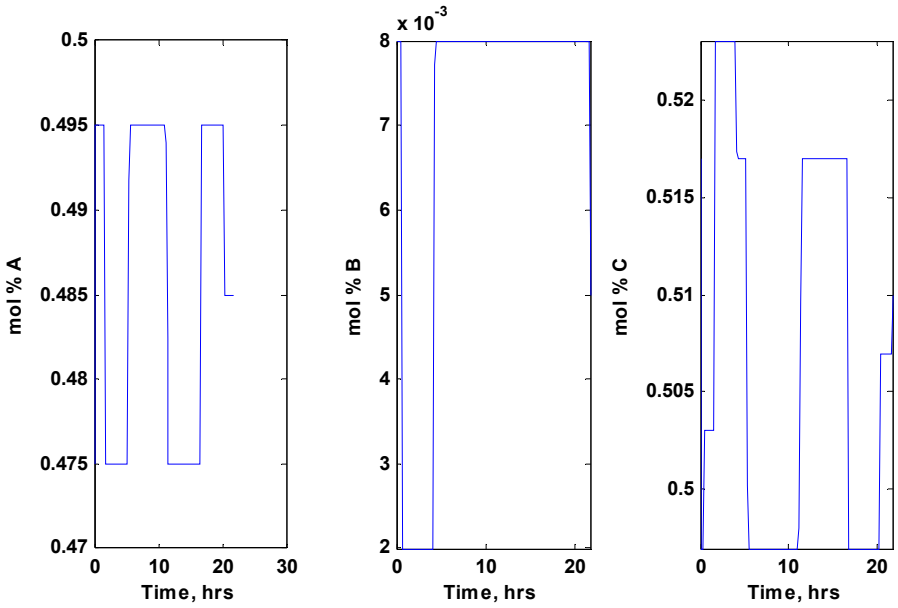

Figure 5.23 Disturbance profiles, optimal control problem

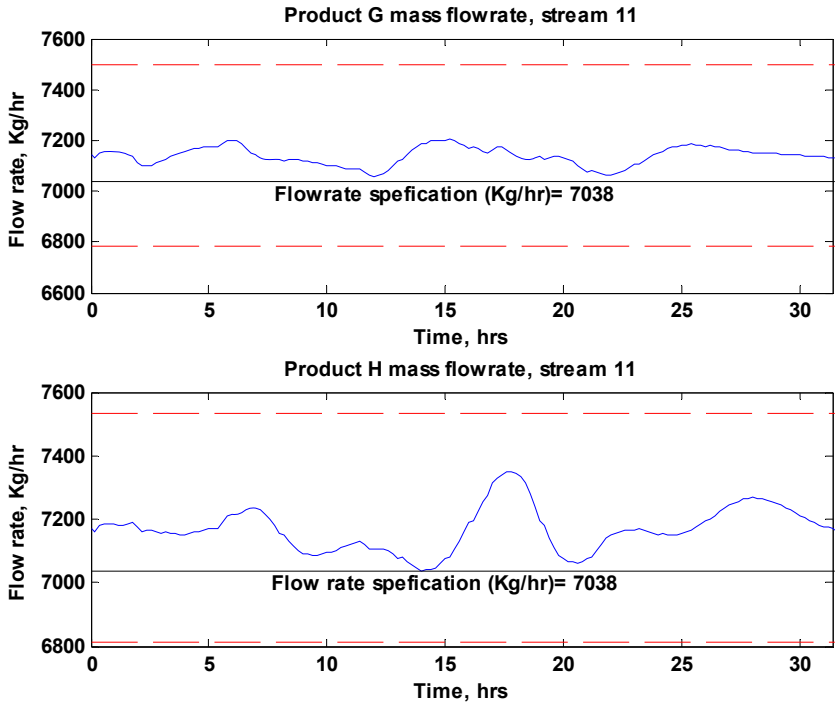

Figure 5.24 Products mass flow rates, optimal control problem

5.3.6 Effect of process parameter uncertainty

One of the key simplifications made on Scenario I-A, Scenario I-AB and Scenario I-ABsim is that the process parameters' true values were assumed to be known a-priori. Accordingly the formulation of these optimization problems did not include in the analysis uncertain process parameters ($\boldsymbol{\omega}$). As it was mentioned at the beginning of Section 5.3, the addition of process parameter uncertainty in the simultaneous design and control problem dramatically increase the computational load. This is because the robust variability function (4.8) must be solved for each $\boldsymbol{\omega}$ tested and for each process variable that is evaluated with this function. The computational time associated with this calculation can be significantly reduced if the process parameter uncertainty ($\boldsymbol{\omega}$) is treated as a disturbance ($\boldsymbol{\upsilon}$) with low and high frequency contents. However, treating $\boldsymbol{\omega}$ as an additional disturbance may result in additional conservatism since the transients in $\boldsymbol{\omega}$ are ignored when it is considered as a parametric uncertainty whereas they will be have to be taken into account if $\boldsymbol{\omega}$ were treated as a disturbance. An alternative approximate solution to this problem is to treat $\boldsymbol{\omega}$ as a *discrete parameter*, i.e., the uncertain process parameter can only be assigned a set of finite number

of values defined within the bounds specified for this parameter. However, this approach cannot guarantee that a value in this parameter that differs from the specific values considered in the set, may produce a larger deviation than any of the values in the set. This will be case if the plant's cost function is highly nonlinear to changes in this parameter.

The second approach explained above was used to solve Scenario I discussed earlier in this chapter. Although Downs and Vogel[62] specified a set of disturbances for this process, they did not specify a process parametric uncertainty as defined in this work. Therefore, the present work specifically investigated the effect of uncertainty in the pre-exponential term of the second reaction specified in (5.9). According to the TE process model provided by Downs and Vogel[62], the reaction rate equation associated with the second reaction was defined as follows:

$$rr(2) = \beta \exp\left(A1 - \frac{A2}{RT_{\mathrm{Re}\,act}} \right) P_A^{1.1544} P_B^{0.3735} P_E \tag{5.26}$$

where β is a constant parameter equal to the unity, R is the universal gas constant, P_A, P_B and P_E are the partial pressures of components A, B and E in the reactor and T_{React} is the reactor's temperature, respectively. Similarly, $A2$ is the activation energy coefficient set to 20,000 and $A1$ is the pre-exponential term, assumed to be an uncertain parameter for this problem, which is defined as follows:

$$A1 = A1_{nom} \pm \delta_{A1} \tag{5.27}$$

where $A1_{nom}$ is the term's nominal value equal to 3.00094014 and δ_{A1} is the uncertainty associated with this parameter. To investigate the effect of this uncertainty, the current analysis assumed the following set of values:

$$\delta_{A1} = \left[-0.4, -0.2, 0, 0.2, 0.4 \right] \tag{5.28}$$

Accordingly, the coefficient $A1$ can only take five values according to the descriptions given above for this parameter. The values specified for δ_{A1} were obtained from simulation of the TE process using the design parameter specifications shown in Table 5.3. Based on the

simulations, a change of ± 33% was observed in the second reaction rate with respect to its nominal value (δ_{A1}=0) .Similarly, the plant's cost function (5.17) varied from 16% to 22% for this set of uncertain values. Thus, it is expected that the set of values specified in (5.28) will significantly affect the plant's design.

The disturbances considered for this problem were the same as in Scenario I-AB, i.e., random magnitude-bounded variations in the mole fraction of components A and B in stream 4, i.e. υ_A and υ_B, respectively. The nominal mol fraction values and the upper and lower mole fraction limits specified for each component are given in Table 5.4.

Based on the above descriptions, Scenario I's problem (5.22) was reformulated to attain the simultaneous design and control of the reactor section of the TE process under the effect of a discrete process parameter uncertainty and external perturbations. This problem, referred heretofore as Scenario I-ABω, is presented in (5.29). The roman number subscript in *A1* in this formulation is used to denote that the critical disturbance profile in υ_A and υ_B that produce the worst-case variability for each robust variability problem posed in (5.29) may be different from each other. This formulation considers that two disturbances are affecting the process, i.e. n_{υ} is set to 2 in (5.29). To estimate the worst-case process variability problem posed in (5.29), 4 robust FIR models between the disturbance set (υ_A and υ_B) and the products' mass flow rate variability G and H must be identified at each discrete uncertain value $A1_j$ tested by the optimization. This same procedure was applied to evaluate the process constraints considered in Scenario I-ABω's problem.

Due to the computational burden associated with this problem, the approach used to solve Scenario I-AB was also applied to solve this problem. That is, the largest variability on each output of interest, posed as a robust performance problem in (5.29), is represented as a bound (k) obtained from the μ-analysis calculations. In this case, five bounds were obtained for each output of interest. Each bound is related to the largest variability in the output corresponding to a given value of the set of values assumed to represent the uncertain process parameter *A1*, i.e. $A1_0$, $A1_1$,…, $A1_5$. Then, the largest bound out of the five the bounds calculated for each output of interested is used to evaluate this output's maximal variability.

$$\min_{\boldsymbol{\eta}} \quad CF_{TE}$$

s.t.

$$\varphi^{G} = \max_{A1\in[A1_0,\ldots,A1_5]}\left(\breve{G} + \max_{\mu_{\Delta^G}(\mathbf{M}^G)\geq k^G} k^{G} - G^{*}\right)$$

$$\varphi^{H} = \max_{A1_I\in[A1_0,\ldots,A1_5]}\left(\breve{H} + \max_{\mu_{\Delta^H}(\mathbf{M}^H)\geq k^H} k^{H} - H^{*}\right)$$

$$xm_G - 1.02xm_H \leq 0$$

$$xm_G - 0.98xm_H \geq 0$$

$$xm_G + xm_H - 0.98 \geq 0$$

$$\max_{A1\in[A1_0,\ldots,A1_5]}\left(\breve{G} - \max_{\mu_{\Delta^G}(\mathbf{M}^G)\geq k^G} k^{G} - G^{*}\right)\geq 0$$

$$\max_{A1_I\in[A1_0,\ldots,A1_5]}\left(\breve{H} - \max_{\mu_{\Delta^H}(\mathbf{M}^H)\geq k^H} k^{H} - H^{*}\right)\geq 0$$

$$\max_{A1_{II}\in[A1_0,\ldots,A1_5]}\left(-0.05\left(\breve{G}+\breve{H}\right) + \max_{\mu_{\Delta^G}(\mathbf{M}^G)\geq k^G} k^{G} + \max_{\mu_{\Delta^H}(\mathbf{M}^H)\geq k^H} k^{H}\right)\leq 0$$

$$\max_{A1_{III}\in[A1_0,\ldots,A1_5]}\left(-0.05\breve{x}_G + \max_{\mu_{\Delta^{x_G}}(\mathbf{M}^{x_G})\geq k^{x_G}} k^{x_G}\right)\leq 0$$

$$\max_{A1_{IV}\in[A1_0,\ldots,A1_5]}\left(\breve{P}_r + \max_{\mu_{\Delta^{P_r}}(\mathbf{M}^{P_r})\geq k^{P_r}} k^{P_r} - 2895\right)\leq 0$$

$$\max_{A1_V\in[A1_0,\ldots,A1_5]}\left(\breve{L}_r + \max_{\mu_{\Delta^{L_r}}(\mathbf{M}^{L_r})\geq k^{L_r}} k^{L_r} - 100\right)\leq 0$$

$$\max_{A1_V\in[A1_0,\ldots,A1_5]}\left(30 - \breve{L}_r + \max_{\mu_{\Delta^{L_r}}(\mathbf{M}^{L_r})\geq k^{L_r}} k^{L_r}\right)\leq 0$$

$$\max_{A1_{VI}\in[A1_0,\ldots,A1_5]}\left(\breve{T}_r + \max_{\mu_{\Delta^{T_r}}(\mathbf{M}^{T_r})\geq k^{T_r}} k^{T_r} - 150\right)\leq 0$$

$$\boldsymbol{\eta}^{l} \leq \boldsymbol{\eta} \leq \boldsymbol{\eta}^{u}$$

TE Process model equations (5.13)

17 PI controller equations (Table 5.3) (5.29)

According to (5.29), six robust performance functions must be estimated. Hence, the TE process must be simulated for each discrete value specified for the parameter *A1* and for each output of interest. Thus, for each new set of decision variables values ($\boldsymbol{\eta}$) tested by the optimization algorithm, 30 simulations of the TE process are required to evaluate the robust variability tests posed in (5.29) as compared to only 6 simulations of the TE process that

were required to test the same scenario when process parameter uncertainty was ignored., i.e. Scenario I-AB. Accordingly, the consideration of one single uncertain process parameter in the analysis significantly increases the problem's curse of dimensionality.Scenario I-AB's optimization problem was executed in MATLAB® using the Sequential Quadratic Programming-based function *fmincon* that is available in the MATLAB's Optimization toolbox[55]. Problem (5.29) was initialized using the solution that provided the lowest annualized cost for Scenario I-AB, i.e. $\boldsymbol{\eta}_4$* in Table 5.8.

In order to illustrate the curse of dimensionality when one uncertain process parameter is considered in the present analysis, the CPU time needed to compute one function evaluation of problem (5.29) was recorded and compared to that needed for Scenario I-AB. Figure 5.25 shows the execution time of one complete function evaluation after different number of function evaluation counts up to the first 100 function evaluations. It is evident from this Figure that the addition of an uncertain parameter within the analysis increases the computational times by at least one order of magnitude. On average, Scenario I-AB required 157 seconds to completely evaluate problem (5.24) whereas Scenario I-ABω needed 5,152 seconds to carry out the same calculation. This results in an average ratio of 1:32 in favour of the scenario that did not consider parametric uncertainty, i.e. Scenario I-AB. Thus, it is expected that adding a second uncertain parameter within the analysis will result in a ratio of about 1:64 with respect to the Scenario I-AB. Although Scenario I-ABω requires a significant amount of CPU time, it is expected that solving the same problem using a dynamic programming approach will result in even larger computational times.

Table 5.11 shows the resulting design parameters obtained from the solution of problem (5.29). The design specified from this Scenario is approximately 42% more expensive than the design that did not consider parametric uncertainty in the formulation, i.e. $\boldsymbol{\eta}_4$* in Table 5.8. Thus, a design that considers uncertainty in the value of the pre-exponential term of the second reaction, *A1* in (5.26), results in an increase of the plant's annual cost by 1.344 MM$/yr approximately. It should be noted from Table 5.11 that the uncertain process parameter value that produces the largest variability in the outputs according to problem (5.29) was not the same for all the constraints. This reinforces the need for solving each robust variability test independently.

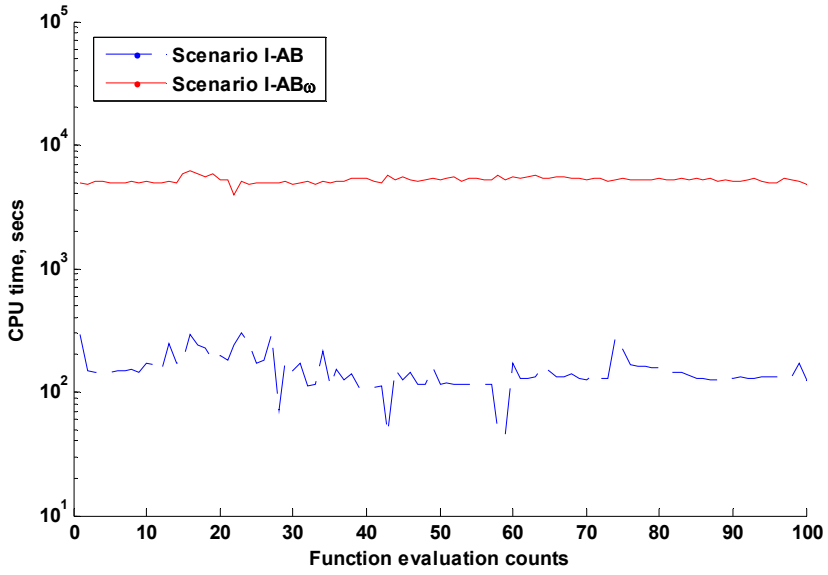

Figure 5.25 Comparison of CPU times with and without parameter uncertainty

The design obtained from Scenario I-ABω was validated from simulation of the TE process using as inputs the value in the uncertain process parameter (*A1*) and the critical disturbance profiles (υ_A and υ_B) that produced the largest variability in the quality of component G at the product's stream, i.e. constraint 7 in Table 5.11. As in previous scenarios, this constraint was found to be active at the solution point. Figure 5.26 shows the critical disturbance profiles in components A, B and C, which combined with an uncertain parameter value of -0.4, produces the largest variability in the quality of product G at the products stream, i.e. stream 11 in Figure 5.4. Similarly, Figure 5.27 shows the results of simulating the design with these critical realizations in *A1* and υ_A and υ_B. Although Figure 5.27 shows that the solution specifies a conservative design, the resulting design parameters satisfy at all time the quality specifications and the minimum production targets specified for this problem. Likewise, the rest of the process constraints posed in problem (5.29) were also satisfied for the critical realizations in *A1* and υ_A and υ_B shown in Figure 5.26.

Table 5.11 Design parameters, Scenario I-ABω

	Specification
Decision variables (**η***)	
Reactor's design capacity (ft^3)	3000.0
Reactor's pressure set point (KPa)	2600.0
Reactor's liquid level set point (%)	47.4
Reactor's temperature set point (°C)	120.2
Production set point (m^3/hr)	23.98
Kc, purge valve (loop 5)	1.00E-01
τ_I, purge valve (loop 5)	5.00E-04
Kc, reactor liquid level(loop 11)	0.60
τ_I, reactor liquid level (loop 11)	77.85
Kc, reactor pressure (loop 12)	-2.18E-04
τ_I, reactor pressure (loop 12)	60.00
Kc, reactor temperature (loop 16)	-12.64
τ_I, reactor temperature (loop 16)	15.00
Constraints[a] / *A1*@ **η***	
Constraint 1	-0.0170/-0.4
Constraint 2	-0.0062/0.4
Constraint 3	-0.0113/ 0.4
Constraint 4	-170.99/0.4
Constraint 5	-83.695/-0.4
Constraint 6	-300.15/-0.4
Constraint 7	-4.16/-0.4
Constraint 8	-287.22/-0.4
Constraint 9	-47.41/-0.4
Constraint 10	-12.15/-0.4
Constraint 11	-29.78/-0.4
Cost breakdowns (MM$/yr)	
Annualized Capital Cost	0.1070
Operating Cost	0.8320
Variability Cost	2.2930
Plant's Annualized Cost (MM$/yr)	3.2320

a The constraints are listed according to order they appear in (5.29).

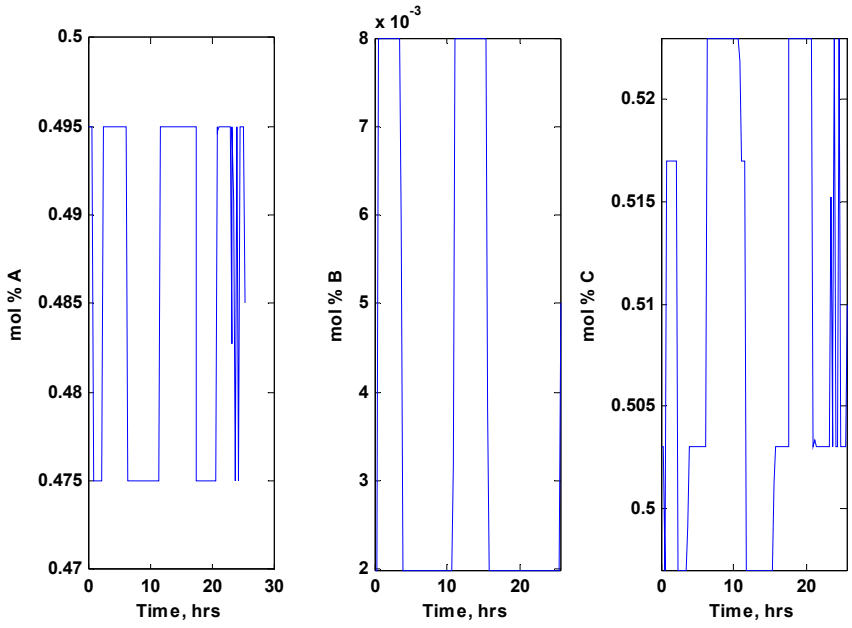

Figure 5.26 Disturbance profiles at *A1*=-0.4, Scenario I-ABω

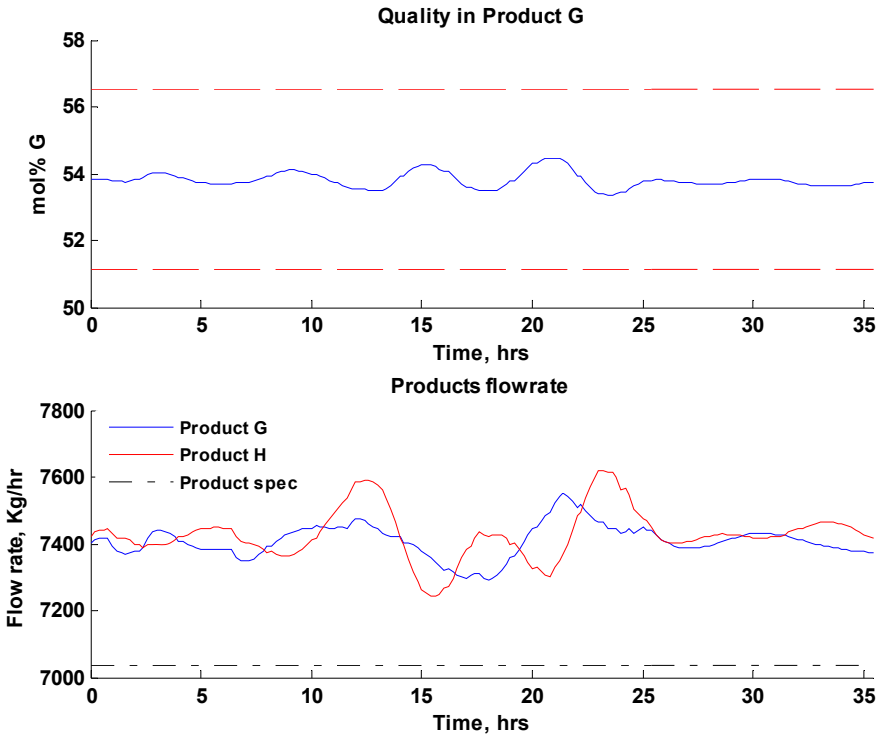

Figure 5.27 Products' specifications in stream 11, Scenario I-ABω

5.4 Simultaneous Design and Control of the Tennessee Eastman Plant

The analysis made for the previous scenarios only considered changes in the design, operability and controllability of the reactor section of the TE process. Although the results satisfied the production and quality targets specified for the base case operation of this process, the scenarios did not include other sections of the plant that may also impact the plant's annual costs. The results obtained from the previous scenarios showed that the plant's annual cost is dominated by the process variability cost. According to the process variability cost formulation given in (5.16), the process variability is directly related to the variations of the product's G and H at the product's stream, i.e. stream 11 in Figure 5.4. However, the previous scenarios considered a set of decision variables ($\boldsymbol{\eta}$) that were only related to the reactor's operation. Thus, none of the elements in $\boldsymbol{\eta}$ was directly related to the process variability that is a direct function of the variability in the products. In the previous analysis, the design capacities of the flash unit and the stripper column remained fixed at their nominal values.

In contrast to the case studies discussed in the previous sections, the study presented in this section proposed the simultaneous design and control of the entire TE plant under the effect of external perturbations only. For simplicity, the current scenario assumed that *all* the process parameters' true values were known a priori, i.e. parametric uncertainty ($\boldsymbol{\omega}$) was not explicitly considered in the analysis. This scenario corresponds to Scenario I-A and Scenario I-AB presented in Section 5.3.1.1 and Section 5.3.1.2, respectively. The optimization problem that describes both scenarios is given in (5.24). Therefore, this problem was used as the base formulation to attain the simultaneous design and control of the entire TE plant.

The vector of decision variables ($\boldsymbol{\eta}$) specified for this problem were as follows: 1) the design capacity of the reactor, the vapour/liquid separator and the stripper column, 2) the nine adjustable set points specified in Table 5.3, i.e. set points for loops 8-16, and 3) the 17 PI controllers tuning parameters specified from the decentralized control strategy proposed by Ricker. Thus, the proposed optimization problem searches for a solution vector $\boldsymbol{\eta}$ composed of 46 decision variables.

The cost function defined in (5.17) was also used by the present analysis to estimate the plant's annual costs. In this case however, the capital cost function (5.13) is a direct function of the design capacity of the three process units considered in the analysis, i.e. the reactor, the flash separator and the stripper column.

The process stability test and the worst-case process variability function specified for the previous scenarios remained the same for this problem. In addition to the process feasibility constraints posed in problem (5.24), the present analysis also considered four additional process operational constraints that are related to the maximum and minimum liquid level allowed in the flash separator and the stripper column, respectively. These constraints were mathematically accounted for by the following inequalities:

$$\begin{aligned}
&\breve{L}_f + \max_{\mu_{\Delta^{L_f}}\left(\mathbf{M}^{L_f}\right)\geq k^{L_f}} k^{L_f} - 100 \leq 0 \\
&30 - \breve{L}_f + \max_{\mu_{\Delta^{L_f}}\left(\mathbf{M}^{L_f}\right)\geq k^{L_f}} k^{L_f} \leq 0 \\
&\breve{L}_s + \max_{\mu_{\Delta^{L_s}}\left(\mathbf{M}^{L_s}\right)\geq k^{L_s}} k^{L_s} - 100 \leq 0 \\
&30 - \breve{L}_s + \max_{\mu_{\Delta^{L_s}}\left(\mathbf{M}^{L_s}\right)\geq k^{L_s}} k^{L_s} \leq 0
\end{aligned} \tag{5.30}$$

where $\breve{L}_f$ and $\breve{L}_s$ denote the flash separator and the stripper column liquid level set points, respectively. These variables were considered as decision variables for this problem and were obtained from the optimization. As in the reactor's liquid level constraints, the minimum liquid levels allowed in both the flash and the stripper were decreased from the value of 50% suggested by Downs and Vogel[62] in the original problem to 30% in order to enlarge the problem's feasible search space. The procedure to evaluate (5.30) was the same used to evaluate (5.21), i.e. discrete robust FIR models from each disturbance (υ) to L_f and L_s are required. The procedure to obtain such models was explained in Section 5.1.1.

Based on the above descriptions, the corresponding optimization problem was formulated as follows:

$$
\begin{aligned}
&\min_{\boldsymbol{\eta}} \quad CF_{TE} \\
&\text{s.t.} \\
&\varphi^{G} = \breve{G} + \max_{\mu_{\Delta^G}\left(\mathbf{M}^{G}\right)\geq k^{G}} k^{G} - G^{*} \\
&\varphi^{H} = \breve{H} + \max_{\mu_{\Delta^H}\left(\mathbf{M}^{H}\right)\geq k^{H}} k^{H} - H^{*} \\
&xm_{G} - 1.02xm_{H} \leq 0 \\
&0.98xm_{H} - xm_{G} \leq 0 \\
&0.98 - xm_{G} - xm_{H} \leq 0 \\
&G^{*} - \breve{G} + \max_{\mu_{\Delta^G}\left(\mathbf{M}^{G}\right)\geq k^{G}} k^{G} \leq 0 \\
&H^{*} - \breve{H} + \max_{\mu_{\Delta^H}\left(\mathbf{M}^{H}\right)\geq k^{H}} k^{H} \leq 0 \\
&-0.05\left(\breve{G} + \breve{H}\right) + \max_{\mu_{\Delta^G}\left(\mathbf{M}^{G}\right)\geq k^{G}} k^{G} + \max_{\mu_{\Delta^H}\left(\mathbf{M}^{H}\right)\geq k^{H}} k^{H} \leq 0 \\
&-0.05\breve{x}_{G} + \max_{\mu_{\Delta^{x_G}}\left(\mathbf{M}^{x_G}\right)\geq k^{x_G}} k^{x_G} \leq 0 \\
&\breve{P}_{r} + \max_{\mu_{\Delta^{P_r}}\left(\mathbf{M}^{P_r}\right)\geq k^{P_r}} k^{P_r} - 2895 \leq 0 \\
&\breve{L}_{r} + \max_{\mu_{\Delta^{L_r}}\left(\mathbf{M}^{L_r}\right)\geq k^{L_r}} k^{L_r} - 100 \leq 0 \\
&30 - \breve{L}_{r} + \max_{\mu_{\Delta^{L_r}}\left(\mathbf{M}^{L_r}\right)\geq k^{L_r}} k^{L_r} \leq 0 \\
&\breve{T}_{r} + \max_{\mu_{\Delta^{T_r}}\left(\mathbf{M}^{T_r}\right)\geq k^{T_r}} k^{T_r} - 150 \leq 0 \\
&\breve{L}_{f} + \max_{\mu_{\Delta^{L_f}}\left(\mathbf{M}^{L_f}\right)\geq k^{L_f}} k^{L_f} - 100 \leq 0 \\
&30 - \breve{L}_{f} + \max_{\mu_{\Delta^{L_f}}\left(\mathbf{M}^{L_f}\right)\geq k^{L_f}} k^{L_f} \leq 0 \\
&\breve{L}_{s} + \max_{\mu_{\Delta^{L_s}}\left(\mathbf{M}^{L_s}\right)\geq k^{L_s}} k^{L_s} - 100 \leq 0 \\
&30 - \breve{L}_{s} + \max_{\mu_{\Delta^{L_s}}\left(\mathbf{M}^{L_s}\right)\geq k^{L_s}} k^{L_s} \leq 0 \\
&\boldsymbol{\eta}^{l} \leq \boldsymbol{\eta} \leq \boldsymbol{\eta}^{u} \\
&\text{TE Process model equations (5.13)} \\
&\text{17 PI controller equations (Table 5.3)}
\end{aligned}
\tag{5.31}
$$

As in the previous scenario, the TE process model equations together with the 17 PI controller algorithm equations were used to describe the closed-loop TE process model. Closed-loop robust FIR models identified from simulations were used as before to test the

process stability, estimate the process variability and evaluate the process constraints as per the equations shown in (5.31).

It should be noticed that the structure of the problem given by (5.34) is similar to that proposed to solve Scenario I-A and Scenario I-AB that only considered changes in the reactor section, i.e. problem (5.24). Therefore, this same optimization problem can also be used to solve the same two scenarios for the entire TE plant. These scenarios were redefined as follows:

1. **Scenario II-A**: The external perturbation affecting the TE process was assumed to be a magnitude- bounded random variation in the mole fraction of component A of stream 4, see Figure 5.4. The composition of B in stream 4 was assumed constant and equal to its nominal value whereas the composition of C varied according to the stream's mole fraction balance. Since this scenario considered only one disturbance, the parameter n_v in (5.31) was correspondingly set to the unity.

2. **Scenario II-AB**: The external perturbations affecting the TE process were assumed to be magnitude-bounded random variations in the mole fractions of components A and B in stream 4, see Figure 5.4. As in Scenario I-A, the variations of C in stream 4 were determined from the stream's mole fraction balance. In this case the parameter n_v in (5.31) was set to 2 since two external perturbations were assumed to be affecting the process.

Further details about the description of each scenario were given in Section 5.3.1.1 and Section 5.3.1.2, respectively. Similarly, the disturbance's nominal values and upper and lower bounds used in the current scenario were given in Table 5.4.

Both Scenario II-A and Scenario II-AB were solved in MATLAB® using the built-in function *fmincon* that performs the optimization of nonlinear constrained problems applying the Sequential Quadratic Programming (SQP) algorithm[55]. Scenario II-A's optimization problem was initialized using the solution obtained from Scenario I-A shown in Table 5.5. Similarly, the solution that provided with the lowest annualized cost for Scenario I-AB, i.e. $\boldsymbol{\eta}_4$* in Table 5.8, was used as the initial guess for Scenario II-AB's optimization problem.

The decision variables' upper and lower bounds ($\boldsymbol{\eta}^u$ and $\boldsymbol{\eta}^l$) were estimated from simulations of the closed-loop TE process.

The results obtained for each scenario are summarized in Table 5.12. Scenario II-A defined a plant which annual costs are 4.8 % less than that specified by Scenario I-A. On the other hand, the design and control scheme defined by Scenario II-AB specified a plant which annual costs differ only by 0.5% with respect to those obtained by Scenario I-AB. Although both scenarios that considered the entire TE plant produced a more economical plant's design than those obtained for Scenario I-A and Scenario I-AB, the annualized cost savings were not significant, i.e. \$81,300/yr and \$9,800/yr for Scenario II-A and Scenario II-AB, respectively. These results suggest that the TE cost function is more sensitive to changes in the parameters related to the reactor's operation, i.e. the vector $\boldsymbol{\eta}$ specified for Scenario I-A and Scenario II-AB, than those related to the operation of the stripper and separator, respectively. Therefore, it can be concluded that the reactor's design parameters dominate the TE process's economics.

From the computational point of view, the addition of more constraints and decision variables within the analysis increase the problem's curse of dimensionality. To evaluate the computational load associated with these scenarios, the CPU time required to compute one function evaluation was recorded and compared to that obtained for Scenario I-A and Scenario I-AB, respectively. Figure 5.28 shows the CPU time required by each scenario to compute one entire function evaluations as a function of the completed function evaluations. As shown in Figure 5.28, the problems related to the design and control of the complete TE plant required a slightly larger computational time that those that considered only the reactor section of the TE plant. On average, Scenario I-A and Scenario II-A required a CPU time of approximately 45 seconds and 69 seconds, respectively. Similarly, Scenario I-AB and Scenario II-AB required an average CPU time of about 157 seconds and 241 seconds, respectively. That is, the simultaneous design and control of the entire TE process requires approximately 36% more computational time than that needed when only the reactor section is considered for optimization.

Table 5.12 Results synopsis, Scenario II-A and Scenario II-AB

Decision variables (**η***)	**Scenario II-A**	**Scenario II-AB**
Process unit's design capacity (ft^3)		
Reactor (Current design:1300 ft^3)	3000.00	2759.15
Flash separator (Current design:3500 ft^3)	3351.12	3879.31
Stripper column (Current design:156.5 ft^3)	193.14	100.44
Adjustable set points		
Reactor's pressure set point (KPa)	2652.92	2858.75
Reactor's liquid level set point (%)	51.20	64.04
Reactor's temperature set point (°C)	119.60	122.17
Separator's liquid level set point (%)	60.00	52.75
Stripper's liquid level set point (%)	55.90	58.52
y_A set point (%)	37.20	32.64
y_{AC} set point (%)	16.47	13.80
Mol % G set point (% Mol G)	53.88	53.80
Production set point (m^3/hr)	23.31	23.31
Cost breakdowns (MM$/yr)		
Annualized Capital Cost	0.1070	0.1040
Operating Cost	0.6660	0.6450
Variability Cost	0.8270	1.1280
Plant's Annualized Cost (MM$/yr)	1.6000	1.8780

a The constraints are listed according to order they appear in (5.31).

The design and control schemes found for Scenario II-A and Scenario II-AB were validated by simulating the TE process with the resulting design parameters. The disturbance profiles used to validate Scenario II-A's solution are shown in Figure 5.29. These realizations correspond to the critical disturbance profiles in the mole fraction of A, B and C in stream 4 that produced the largest variability in the product G's quality. Although any of the critical profiles that were estimated at the solution can be used to validate the Scenario II-A, the constraint related to the product G's quality was active at the solution and consequently it was chosen to illustrate the validity of the results. Figure 5.30 shows that Scenario II-A's resulting design parameters kept the products quality and mass flow rates above their target specifications.

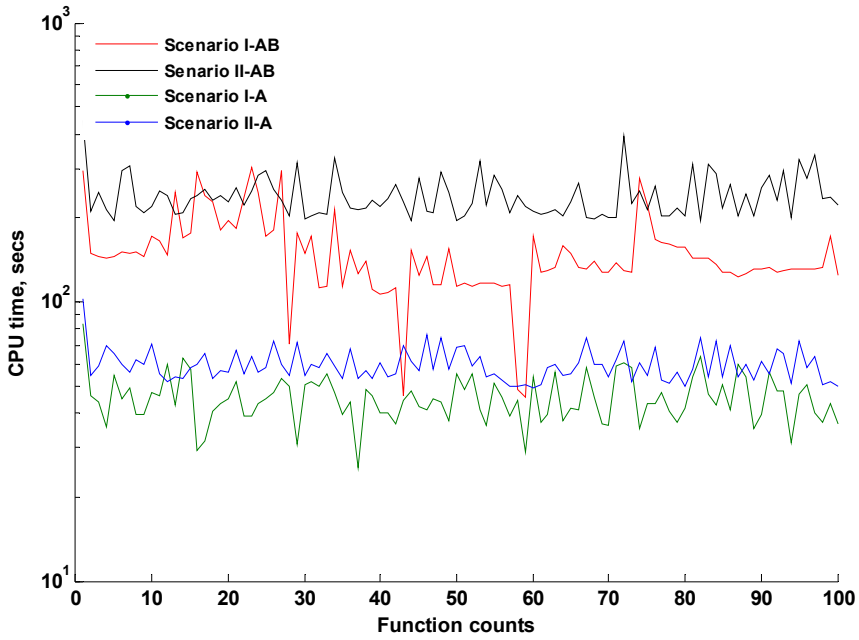

Figure 5.28 Comparison of CPU times for different scenarios

Similarly, Scenario II-AB's design was also validated from simulation. Figure 5.31 shows the disturbance profiles that were used to simulate the TE process design specified by this scenario. These profiles correspond to the critical vector that produced the maximum deviation in Product H's flow rate, which was the constraint that was found to be active at the solution for this scenario. Figure 5.32 shows that the actual maximum deviation in product H's flow rate of 7038 kg/hr causes this production rate to get closer to its product specification. This Figure also shows that the selected disturbance profiles produced large variations in the product's flow rates. The other set of critical profiles calculated at the solution point for Scenario II-A and Scenario II-AB were also simulated to show that the process operational constraints and the products specifications always remained within their target values.

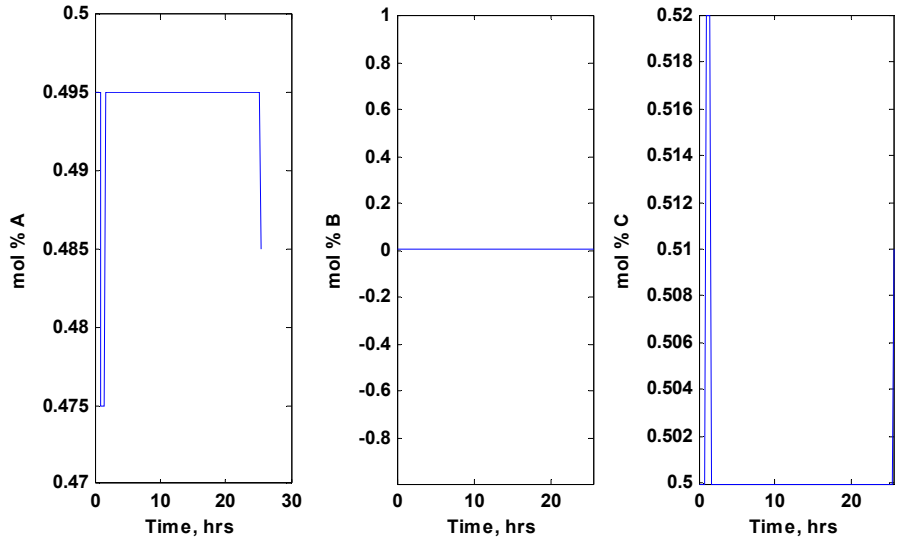

Figure 5.29 Disturbance profiles, Scenario II-A

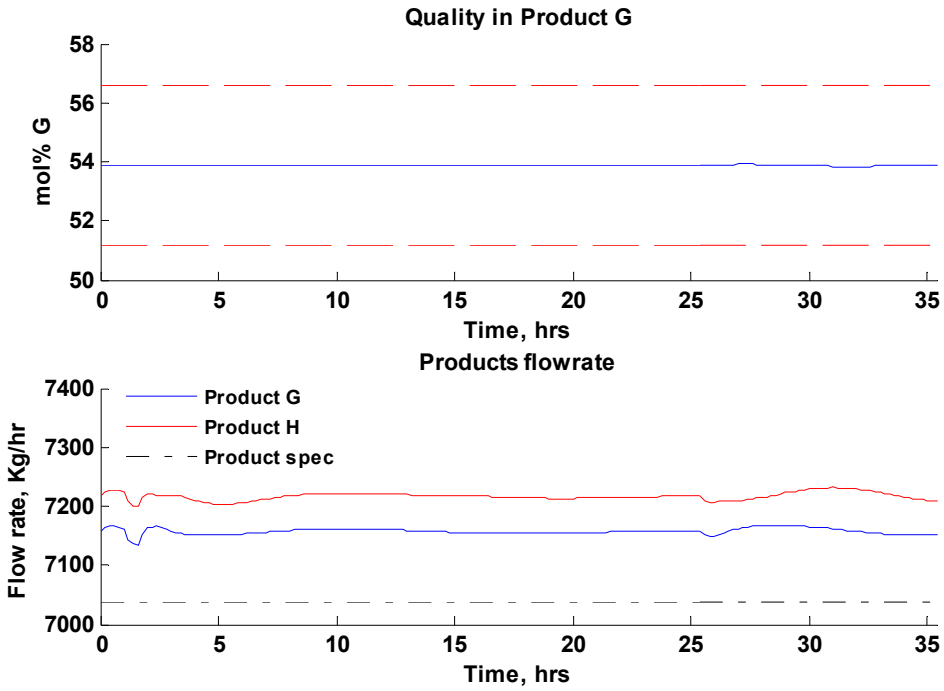

Figure 5.30 Products' specifications in stream 11, Scenario II-A

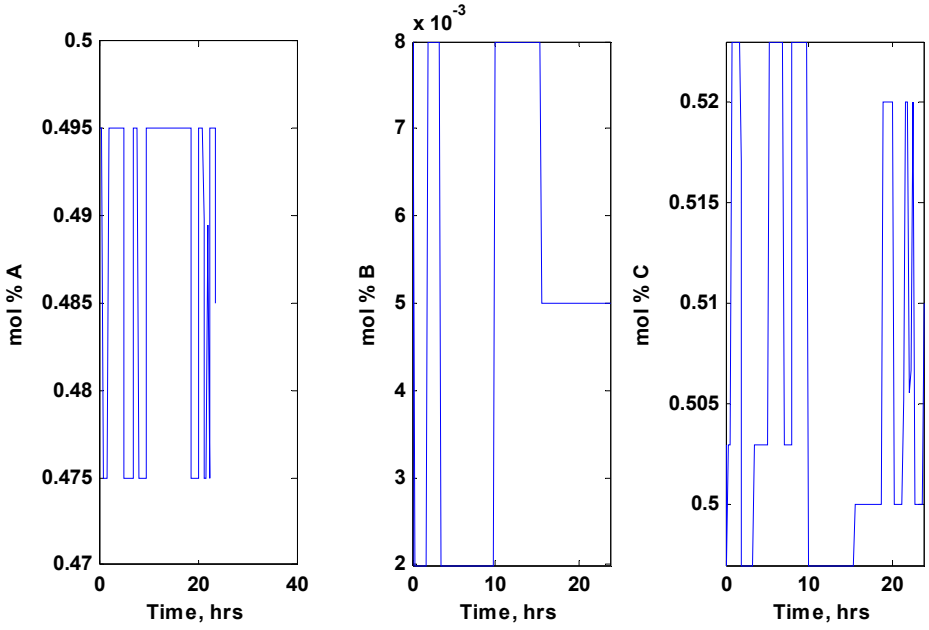

Figure 5.31 Disturbance profiles, Scenario II-AB

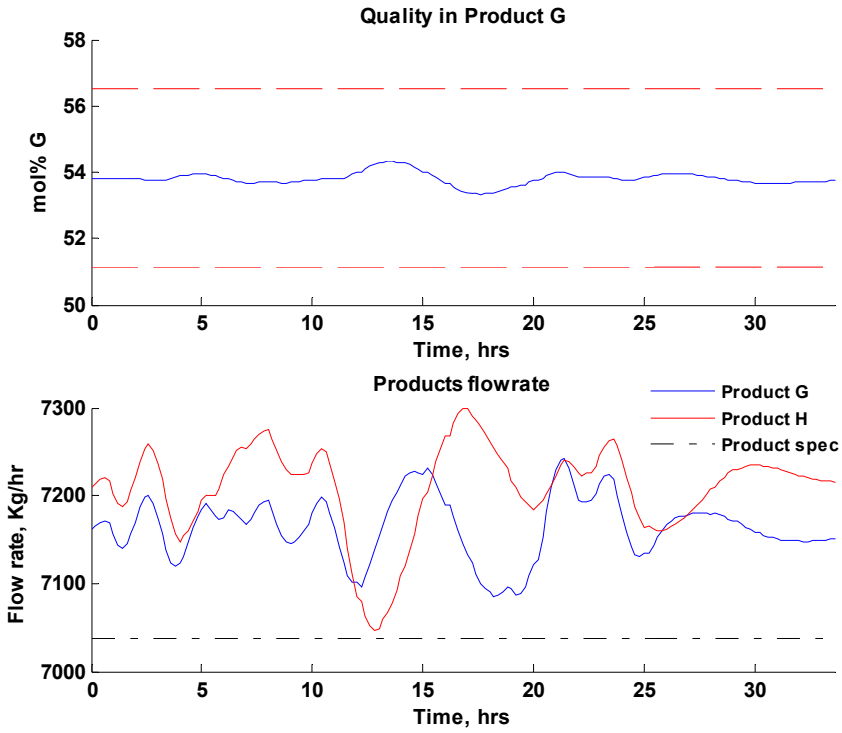

Figure 5.32 Products' specifications in stream 11, Scenario II-AB

In summary, this chapter presented a novel approach to the simultaneous design and control of chemical plants. The key idea in the method is to represent the process closed-loop dynamic behaviour between each disturbance and each process variable as a robust finite impulse response (FIR) model. This robust model, obtained from closed-loop identification, is suitable to test robust performance using a Structured Singular Value (μ analysis) approach. The methodology was tested using the Tennessee Eastman (TE) process that has been used before as a benchmark process by the process systems engineering community to test wide variety of control related problems. Although the controllability of this plant has been widely studied, the interactions between design and control have not been previously addressed for this system. To apply the methodology, the decentralized control strategy proposed by Ricker[63] was used to stabilize and control the plant's transient behaviour. Two scenarios were proposed to attain the simultaneous design and control of the TE process. The first scenario considered only the reactor section of the plant whereas the second scenario analyzed all the sections of the plant. The effect of external perturbations and initial guesses on the solution, comparisons to alternative solution strategies, and the effect of parametric uncertainty were studied for the first scenario. The second scenario was used to test the effect of additional disturbances by comparing the optimal design for one or two external perturbations. Due to the robust modelling approach taken by the present methodology, the resulting design and control schemes tends to specify a plant that is conservative. Although dynamic programming (DP) approach would provide a less conservative design than those obtained by the present approach, it was shown the DP approach requires a much higher computational effort than this needed by the present methodology. Therefore, the present methodology is an effective and practical tool that can be used to solve the simultaneous design and control of large-scale processes.

6. Conclusions and Future Work

This research work presented a new methodology that performs the simultaneous design and control of chemical processes that are subject to external perturbations and process parametric uncertainty. The novelty in this approach is that the nonlinear dynamic behaviour of the system was represented here by a linear dynamic model with uncertain model parameters whereby each of the model parameters is bounded by two extreme values. These uncertain models, referred also to as *robust* due to their common use for designing robust controllers, are obtained from the identification of the closed-loop system around an operating state defined by actual values of the system's closed-loop degrees of freedom. The uncertainties in the model parameters capture the system's nonlinearities due to changes in the disturbances at a given uncertain process parameters' steady-state values. To analyze the flexibility, controllability, resiliency and feasibility of the system to be designed, the present approach used tools from robust control theory to test the system's asymptotic stability and to estimate bounds on the worst-case process variability and on process variables that are to be maintained between constraints, respectively. The inputs to the robustness tests are the robust model's uncertain parameters and the pre-specified bounds on the disturbances and the uncertain process parameters that are considered in the analysis. The robustness tests used to evaluate robust stability, worst-case process variability and robust performance were embedded within an optimization problem that search for the values in the system's closed-loop degrees of freedom that minimizes the plant's predefined cost function. The formulation based on the robustness tests results in a nonlinear constrained optimization problem that is more computationally efficient than those methodologies that formulate the problem in a dynamic programming framework.

The next sections discuss the concluding remarks on the subjects that are relevant for this work and outline the areas of opportunities for future research on the topic.

6.1 Closed-loop identification of a robust (uncertain) model

The identification of robust models is central in this research work. The key idea in this work is to represent the relationship between a disturbance (input) and a process variable of

interest (output) as a linear dynamic model structure with uncertain model parameters, i.e. the model parameter values ranges between two extreme values. The linear nominal model can be obtained from any of the available closed-loop identification methods, e.g. least-squares criterion, whereas the uncertainty associated with each model parameter is determined from the covariance matrix obtained from the closed-loop identification.

Although the process models studied in this work are MIMO models, i.e. the mixing tank process and the TE process, the present research work identified individual SISO robust models for each input/output relationship. The reasons for individually identifying SISO models instead on one MIMO model were: i- the direct identification of a single MIMO model for the complete process model resulted in larger computational times than the identification of many SISO models and ii- the direct identification of a single model MIMO model produced a stable nominal linear model but with large model parameter uncertainty descriptions whereby one or more of the models within the family of models described by the robust model were unstable. On the other hand, the SISO models were fast to compute and resulted in smaller uncertainty regions and better robust stability properties than the ones obtained for the overall MIMO model. Thus, it was concluded that the identification of many SISO models is preferable to the identification of one MIMO models due to both computational efficiency and stability considerations.

The present work used both uncertain state space (SS) models and robust finite impulse response (FIR) models to represent the closed-loop process dynamics between an input and output of interest. The state space models were used to calculate RMS-based bounds whereas the robust FIR models were used to compute bounds based on the Structured Singular Value μ. Generally it was found that the structure of the state space models is more compact and requires less uncertain model parameters than a robust FIR model. However, whereas the model order selection for the robust SS model is trivial for simple process models such as the mixing tank process it may be challenging for cases studies such as the TE process. Although the model order selection is performed off-line, this identification step is tedious and time consuming since it requires the testing of different model structures for each input/output relationship that is considered in the analysis. Also, to use the μ-based robust tools developed in this work, the robust SS model needs to be transformed into a robust FIR model, which

adds computational load within the analysis. Moreover, the use of high order robust SS models was found to increase significantly the computational time.

On the other hand, the robust FIR model is a non-parametric model usually composed of a significantly large number of model parameters, i.e. the number of model parameters depends on the operating state at which the model was identified. Although the identification step is more computationally demanding than that generally required for low-order robust SS model identification, the model order selection for the robust FIR model is trivial since it is specified by the PRBNS properties used to simulate the full nonlinear model describing the process. Thus, the model order selection step can be avoided. Furthermore, the application of the μ-based robust tools is performed faster when the FIR model is identified directly from simulations rather than through an intermediate identification of a SS model. Based on the above, it was concluded that the use of uncertain SS models should be restricted to those case studies that are described by a relatively small number of ODE's, e.g. mixing tank process, whereas robust FIR models are more suitable to represent input-output relations for large processes such as the TE process.

Another important aspect of the identification step is the quantification of the uncertainty in the model parameters. This work used the parameters' variance to describe the uncertainty in the model coefficients. This is solely a practical approximation since the elements of the covariance matrix are estimated from the deterministic errors between the nominal model parameters and the actual input/output values used in the identification. The parameters' variances are expected to converge to their true values as the size of the data set used for identification becomes infinite. Thus, very long simulations of the closed-loop process model should be conducted to improve the accuracy of the parameters' variances. To ensure that the uncertainty in model coefficients was sufficient to describe the system, a model validation step was implemented in chapter 3 whereby at each step of the optimization the calculated maximal deviation is checked versus the maximal deviation obtained from the simulation of the nonlinear model. Although this step prolongs the optimization, it provides some confidence in the uncertainty bounds and in the resulting maximal deviation calculations. One key advantage in the use of uncertain models in the current work is that their

identification is performed in closed-loop resulting in smaller uncertainty descriptions, as compared to open loop models, due to the corrective effect of feedback control.

The signal used for the identification of the robust model was also found to be important for the analysis. The design of the PRBNS used to simulate the process closed-loop dynamic behaviour is not trivial. A highly nonlinear model simulated with a PRBNS that was poorly designed often resulted in robust FIR models that did not capture the true process dynamic behaviour. Consequently, the application of the robustness tests on these model parameters was found to produce inaccurate stability and input/output variability results. Therefore, tests must be performed off-line to design a PRBNS that captures the process basic characteristics such as its time constant and settling time. The dominant time constant of the process is used to define the PRBNS's switching time while the process settling time determines the maximum length of the sequence and the period of the input signal (see Section 2.3). For the mixing tank problem, the design of the PRBNS was relatively simple since this process exhibit a low degree of nonlinearity. In this case, the mixing tank's time constant and settling time were obtained from a single step test. On the other hand, the design of the PRBNS for the TE plant was challenging since this was found to be a highly nonlinear open-loop unstable process. The PRBNS switching time was selected according to the third sample time guideline given in Section 2.3. Thus, the switching time was set to 0.2 hr. which lies between 1/100 and 1/20 of the process settling time. Similarly, the PRBNS period was selected according to the process settling time and the PRBNS switching time previously selected. Although other PRBNS designs were tested, the one specified above produced uncertain FIR models that were good approximations of the TE closed-loop dynamic behaviour. Moreover, since the PRBNS is a deterministic signal with white-noise like properties, multiple repeats of the PRBNS sequence must be used to simulate the TE closed-loop process for a successful identification of robust FIR models.

The degree of nonlinearity of the process model to be designed was also found to be critical for accurate identification of uncertain models. The application of the least-squares method to a data that was generated from the simulation of a stable though highly nonlinear process model may result in the identification of an unstable uncertain model. That is, although the least-squares nominal parameter estimates corresponded to a stable linear dynamic model,

certain combinations between the extreme values of the uncertain model parameters define a linear state space model which eigenvalues lies on the right hand side of the imaginary axis. Thus, even though the resulting robust model is stable, the discrepancies between the identified uncertain model and the true input/output dynamic behaviour may be significant. This was the situation when uncertain SS models were used for the identification of a large-scale process such as the TE process. Therefore, the use of the uncertain SS models is also limited by the process model's degree of nonlinearity. On the other hand, unstable robust FIR models were not obtained for the TE problem but the differences between the model and the actual input/output behaviour were considerable. This consequently added more conservatism into the robust stability and robust performance tests. However, the results obtained for the TE process shows that the conservatism in the design and control schemes found by the present methodology are within a range (5-10%) that were deemed acceptable in view of the computational efficiency as compared to other methods such as dynamic programming.

6.2 Robust stability and performance tools

The present research work used two different approaches to calculate bounds on the stability, dynamic flexibility and feasibility of the system to be designed. The first approach considered in this work used a Quadratic Lyapunov Function (QLF) to establish bounds on the input/output's asymptotic stability and random mean square (γ) variability. The second approach specified a Structured Singular Value problem (μ-analysis) to estimate bounds on the input/output's stability and worst-case variability. Although the QLF approach and μ-analysis techniques have been widely used in control systems applications, they have not been previously considered for analyzing the system's dynamic flexibility and feasibility.

To apply the robustness tests, the input/output process behaviour of the closed loop system under consideration must be represented as an uncertain dynamic model. Each approach requires a different uncertain model representation, that is, the γ-based criteria require an uncertain SS model while the μ-based tests require a robust FIR model, respectively.

According to the results obtained for the mixing tank process shown in chapters 3 and 4, the use of γ-based criteria as a performance measure resulted in a more conservative design than

the design and control scheme obtained from the μ-based strategy. The relative conservatism of the γ-based approach is mostly due to the fact that this variability test must account for *all* the possible rate of changes in the QLF ($\mathbf{V}(\mathbf{x})$) as per the derivative term in inequality (3.14) that depend on the uncertainty of the SS model parameters. Thus, although the γ-based criterion can only provide a bound on the model output error's variance which is theoretically smaller than the maximal possible deviation obtained with the μ-based approach, the γ-based approach is ultimately more conservative due to the mathematical formulation of the test.

Due to the limitations presented by the γ-based performance criterion, the present work proposed the use of alternative robust stability and performance tools based on μ where the robust performance tool provided bounds on the model output's worst-case variability. This test only depends on the uncertain FIR model parameter descriptions and the input's amplitudes. Thus, this worst-case output error bound can be used to test the system's feasibility and the worst-case process variability. This μ-based solution strategy was also applied to the simultaneous design and control of the TE process. The results obtained with the μ-based analysis shows that the current annual costs of the TE plant can be reduced by 11% if the plant is redesigned and retuned according to the design and control strategy specified by the proposed analysis.

An additional useful by-product of the μ-based analysis was the calculation of the input's combinations of the disturbances and parametric uncertainties that will result in the largest model output error. This generated the option to calculate the maximal output deviation from simulations of the full nonlinear dynamic model with the critical input profiles and used within the optimization the simulated maximal deviations instead of the μ-based analytical bounds. According to the results shown in Section 5.3, the use of the simulated bounds of the TE process model with the critical disturbance resulted in a design that has a 10% lower cost than that obtained by using the μ-based analytical bounds (see Section 5.3.2 and Section 5.3.3). However, the estimation of the maximum actual output error from simulation required twice the computational time than that required for the estimation of the worst-case output bound only. This is because the simulation based approach required an additional simulation of the TE process using the critical disturbance profile for each output tested.

As it was shown in Section 4.1, the optimization problem based the μ-based performance test is convex with respect to the parameter that specifies the worst-case output bound, i.e. k in problem (4.8). Thus, a global optimal solution is obtained from this problem. This property is advantageous since the estimation of the worst-case output variability have been traditionally estimated with a dynamic programming optimization, which not only requires an intensive computational effort as shown in Section 5.3.4, but it is also subject to local solutions.

6.3 Methodology's formulation

Although different robustness tests were used during the present research work, the basic structure of the optimization problem remained the same. That is, the system's asymptotic stability was tested using a specific robust stability criterion whereas bounds on the process operational constraints and the worst-case process variability function were evaluated from the robust performance tests. The robustness tests used in this work are based on the QLF criteria (γ-based) or the Structured Singular Value approach (μ-based).

The present work defined a cost function that consisted of capital, operating and process variability related costs. The first two cost terms represent the process steady-state economics and can be easily estimated from the process data and numerical correlations available in the literature. On the other hand, the definition of variability cost function is difficult to obtain since it is problem specific and there is no a direct relationship between the process dynamic performance and the process economics. Based on the goals to attain by the design, this work defined for the case studies variability cost functions that have a physical and related economic meaning. That is, a dollar value was actually assigned to the variability that reflected for the Tennessee Eastman problem for example an actual loss in production. The definition of a physical meaning-based variability cost function is relevant since it provides an economic measure of the benefit of feedback control.The use of this variability cost function is relatively new since most of the previous design and control methodologies that consider a variability cost within their formulation have used physically meaningless weighted functions to translate this process measure into a dollar value. The variability costs proposed in the current work are somewhat related to the back-off concept used in the simultaneous design and control literature by other researchers[4,26]. The key advantage in the

current approach is that the bounds can be computed much faster as compared to the back-off calculations conducted in the literature with dynamic programming based algorithms

The formulation of the optimization problem proposed in this work to perform the simultaneous design and control of a chemical processes corresponds to a nonlinear constrained optimization problem. Although different algorithms have been reported in the literature to address the solution of this class of problems, the present work applied Sequential Quadratic Programming (SQP) to solve all the scenarios considered in the case studies. This algorithm was selected because it is the traditional method used to solve nonlinear constrained optimization problems and the MATLAB's optimization toolbox has made available the built-in function *fmincon* that uses an SQP method. The advantages or disadvantages of using different optimization methods versus SQP were not analyzed in this project since this topic was outside the scope of this work.

The results obtained from the case studies shows that the present methodology provides with a conservative design if compared to those obtained from a dynamic programming approach. The conservatism comes from the representation of the process closed-loop dynamic behaviour as uncertain models, which are used by the robustness tests to determine the process stability and to estimate bounds on the problem's cost function and constraints. Therefore, the degree of conservatism in the final design and control scheme depends on the accuracy of the uncertain model. On the other hand, the evaluation of the worst-case scenario and the process constraints with a dynamic programming approach requires a CPU time that is on average one to two orders of magnitude larger that than required by the present methodology. The results shows that the application of a dynamic programming methodology to a large-scale system such as the TE process results in extremely large computational times in the order of months. Moreover, most of the available dynamic optimization based methods are also subject to obtain local solutions whereas the calculation of the bounds used in this work is convex. Consequently, the methodology proposed in this work is a computationally efficient tool that can be realistically applied to the simultaneous design and control of large-scale systems.

6.4 Future work

The methodology proposed in this work assumed that the process flowsheet and the control structure are defined a priori. That is, the process synthesis and control structure selection problem are not addressed by the present method. Thus a clear extension of the present study will be to include the problems of control structure and flowsheet selection within the overall optimization problem. Obviously, the computational load is expected to increase significantly since addressing this problem will require the use of integer decision variables. However, the inclusion of the flowsheet and control structure selection within the optimization is expected to provide a more economically attractive design than that obtained by the present method.

Due to the robust modelling approach adopted in this work, the present methodology resulted in conservative designs. The conservatism in the solution depends on the differences between the uncertain model and the actual process dynamic behaviour, i.e. tighter uncertain model descriptions will result in less conservative designs. One key aspect that highly determines the *conservatism* in the uncertain model is the system's degree of nonlinearity. Since the uncertain model structures used in this work are assumed to be linear dynamic models with uncertain model coefficients, it is expected that the use of uncertain nonlinear model structures are more likely to provide a better process dynamic behaviour description that may result in less conservatism designs. Thus, the future use of uncertain nonlinear dynamic model structures such as uncertain state-affine models or uncertain Volterra series models to capture the transient behaviour of highly nonlinear systems may produce better designs. Although these models are expected to provide better process dynamic behaviour descriptions their identification may require an intensive computational effort.

Although each of the robustness tests considered within the methodology's formulation is convex, i.e. a global solution is obtained for each of these tests; the overall optimization problem is non-convex. Thus, the SQP method may only provide with a local solution. As it was shown in Section 5.3.2, different starting points resulted in different design and control solutions. Therefore, the present work suggests the use of global optimization methods that can provide for a better search of the variable space.

The proposed formulation considered parameters that remain at a given value for large periods of time but their true value are not known referred to as process parameters uncertainties. To account for this source of process uncertainty, each robustness test was embedded within an optimization problem that searches for the value in the uncertain process parameter that produces the largest output variability. The computation of this optimization problem is performed relatively fast for simple processes such as the mixing tank process but it becomes a major computational bottleneck when performed on large-scale systems such as the TE process. In Section 5.3.6, the TE process was designed for a finite set of values of the process parametric uncertainty. The computational time used to perform the formal optimization problem with respect to the process parametric uncertainty required on average more than double the time needed to solve the optimization with respect to a finite number of values of the process parametric uncertainty. Also, considering a finite set of parameter values does not ensure that the design will be able to accommodate values that were not included in the selected set. The computational load associated with this problem could be alleviated if the uncertain parameter is considered as a disturbance.

Based on the above discussion, it is recommended to refine the solution strategy when uncertain process parameters are considered in the methodology's formulation. One approach that can be followed is to identify robust models that are valid only for a smaller range of values in the uncertain process parameter. In this way the uncertain parameter will be treated as a disturbance within the formulation but instead of defining a single uncertain model for the parameter's entire range of variation, *local* uncertain models that cover subsets of the parameter's range of values will be identified. The method to partition the uncertain process parameter's range of values will depend on the system's degree of nonlinearity, the computational resources available and the desired accuracy in the solution. Although this approach is still an approximation to the original problem's formulation, it is expected that it will provide a less conservative result than that obtained from a single uncertain model. On the other hand, the computational burden will be increased since additional closed-identifications, and therefore more simulations, will be required.

Constraints on the process manipulated variables, e.g. valves, are evaluated using the same procedure as for the process operational constraints, i.e. bounds on the manipulated

variables' variability are estimated using robust performance tests. In the presence of disturbances saturation of the valves may occur. Consequently, the application of the robust performance tests on this valve will produce an unrealistic bound, i.e. values that are exceeding the valve's physical limits, e.g. 0-100 % of opening. Consequently, the use of robust performance tests to formulate manipulated variable constraints may produce either overly conservative designs or optimization problems that in principle have a feasible solution but they cannot satisfy this constraint. Robust performance tests based on a random-mean-squares (RMS) gain criterion (γ-based) might be of use since it estimates a bound on the standard deviation of the output (valve)'s variability. However, this test will also produce unrealistic bounds if the valve's operation is constantly reaching its saturation limits.

Different approaches may be further analyzed to formulate the constraints on the valves. One approach is to estimate, from the μ-based robust performance test, the critical disturbance profile and obtain from simulation the valve's maximum variability. Although this approach adds computational load within the analysis, it is a more realistic formulation to test the valve's range of operability. A second approach that can be considered is to consider within the analysis the design of the valve. Although this may produce an overly large valve design, the use of bounds to test the valve's constraints guarantees that this equipment will not saturate for the disturbances and process parametric uncertainty's amplitudes considered in the analysis. A third approach is to formulate the constraints on the valve as a dynamic optimization problem. This strategy, somehow similar to the first approach proposed above, is expected to require a significant amount of computational time. Consequently, future studies must be conducted to determine the advantages or disadvantages of applying one approach or the other.

Since the robust bounds calculated in this work are based on norms of linear operators, constraints in the manipulated variables have not been explicitly considered since hard constraints can be only represented by nonlinear operators. Thus, the proposed work is potentially conservative regarding the input constraints since it is required in the optimization that the maximal deviations in manipulated variables will be smaller than the constraints at all times. In principle it could be possible to obtain less conservative designs by allowing nonlinear controllers such as a constrained MPC or a PID with anti-windup to operate at the

input constraint for long periods of time. Future studies should assess whether the bounds proposed in this work could be used to deal with hard constrained controllers.

Finally, the methodology presented in this work can be applied to design and control chemical processes that require the use of advanced control techniques such as model-based controllers. Since the case studies analyzed here considered only control structures composed of feedback controllers, e.g. PI controllers, the present work recommends the application of the proposed methodology to cases in which the system is regulated by an advanced control technique such as MPC and IMC.

Bibliography

[1] Shiskey, F. G. *Uncontrollable Processes and what to do with them;* AIChE Annual Meeting; Los Angeles, USA, **1982**.

[2] Luyben, W. L. In *The need for simultaneous design education;* Seferlis, P., Georgiadis, M. C., Eds.; The Integration of process design and process control; Elsevier: Amsterdam, The Netherlands, **2004**; 17, pp 10-41.

[3] Perkins, J. D. In *Interactions between process design and process control;* Perkins, J. D., Ed.; Pergamon Press: London, UK, **1992**.

[4] Seferlis, P.; Georgiadis, M. C. In *The Integration of Process Design and Control;* Seferlis, P., Georgiadis, M. C., Eds.; Elsevier, Ltd: Amsterdam, The Netherlands, **2004**.

[5] Bristol, E. H. *IEEE Transactions on Automatic Control* **1966***, AC-11*, 133-134.

[6] Nguyen, T. C.; Barton, G. W.; Perkins, J. D.; Johnston, R. D. *AICHE J.* **1988***, 34*, 1200-1206.

[7] Skogestad, S.; Morari, M. *Ind Eng Chem Res* **1987***, 26*, 2029-2035.

[8] Seborg, D. E.; Edgar, T. F.; Mellichamp, D. A. In *Process Dynamics and Control;* John Wiley & Sons, Inc.: USA, **2004**.

[9] Lenhoff, A. M.; Morari, M. *Chemical Engineering Science* **1982***, 37*, 245-258.

[10] Palazoglu, A.; Arkun, Y. *Comput. Chem. Eng.* **1986***, 10*, 567-75.

[11] Palazoglu, A.; Arkun, Y. *Comput. Chem. Eng.* **1987***, 11*, 205-216.

[12] Luyben, M. L.; Floudas, C. A. *Comput. Chem. Eng.* **1994***, 18*, 933-969.

[13] Alhammadi, H. Y.; Romagnoli, J. A. In *Chapter B4 Process design and operation Incorporating environmental, profitability, heat integration and controllability considerations;* Panos Seferlis and Michael C. Georgiadis, Ed.; Computer Aided Chemical Engineering, Elsevier: **2004**; 17, pp 264-305.

[14] Brengel, D. D.; Seider, W. D. *Comput. Chem. Eng.* **1992***, 16*, 861-886.

[15] Perkins, J. D.; Walsh, S. P. K. *Comput. Chem. Eng.* **1996***, 20*, 315-323.

[16] Mohideen, M. J.; Perkins, J. D.; Pistikopoulos, E. N. *AICHE J.* **1996***, 42*, 2251-2272.

[17] Dimitriadis, V. D.; Pistikopoulos, E. N. *Ind Eng Chem Res* **1995***, 34*, 4451.

[18] Mohideen, M. J.; Perkins, J. D.; Pistisopoulos, E. N. *J. Process Control* **1997***, 7*, 371-85.

[19] Bansal, V.; Perkins, J. D.; Pistikopoulos, E. N.; Ross, R.; van Schijndel, J. M. G. *Computers and Chemical Engineering* **2000***, 24*, 261-266.

[20] Bansal, V. 1.; Ross, R. 1.; Perkins, J. D. 1.; Pistikopoulos, E. N. 1. In *In The interactions of design and control: double-effect distillation;* Dynamics and Control of Process Systems 1998 (2 Volume Set); Elsevier: Corfu, Greece, **2000**; 10, pp 219-27.

[21] Ross, R. 1.; Perkins, J. D.; Pistikopoulos, E. N.; Koot, G. L. M.; van Schijndel, J. M. G. *Comput. Chem. Eng.* **2001***, 25*, 141-50.

[22] Bansal, V.; Perkins, J. D.; Pistikopoulos, E. N. *Industrial and Engineering Chemistry Research* **2002***, 41*, 760-778.

[23] Sakizlis, V.; Perkins, J. D.; Pistikopoulos, E. N. *Computers and Chemical Engineering* **2004***, 28*, 2069-2086.

[24] Schweiger, C. A.; Floudas, C. A. In *In Interaction of design and control: optimization with dynamic models;* Proceedings of Conference on Optimal Control Theory, Algorithms and Applications; Kluwer Academic Publishers: Gainesville, FL, USA, **1998**; pp 388-435.

[25] Bahri, P. A.; Bandoni, J. A.; Romagnoli, J. A. *AICHE J.* **1997***, 43*, 997-1015.

[26] Kookos, I. K.; Perkins, J. D. *Industrial and Engineering Chemistry Research* **2001***, 40*, 4079-4088.

[27] Swartz, C. L. E. In *Chapter B3 The use of controller parametrization in the integration of design and control;* Panos Seferlis and Michael C. Georgiadis, Ed.; Computer Aided Chemical Engineering, Elsevier: **2004**; 17, pp 239-263.

[28] Lee, J. H.; Lee, J. M. *Computers and Chemical Engineering* **2006***, 30*, 1603-1618.

[29] Chawankul, N.; Budman, H.; Douglas, P. L. *Computers and Chemical Engineering* **2005***, 29*, 261-271.

[30] Chawankul, N.; Ricardez Sandoval, L. A.; Budman, H.; Douglas, P. L. *Can. J. Chem. Eng.* **2007***, 85*, 433-446.

[31] Gerhard, J.; Marquardt, W.; Monnigmann, M. *SIAM J. Appl. Dyn. Syst.* **Accepted for Publication**.

[32] Gerhard, J. 1.; Monnigmann, M. 1.; Marquardt, W. 1. In *Constructive nonlinear dynamics - foundations and application to robust nonlinear control;* Control and Observer Design for Nonlinear Finite and Infinite Dimensional Systems; Springer-Verlag: **2005**; pp 165-82.

[33] Monnigmann, M.; Marquardt, W. *Industrial and Engineering Chemistry Research* **2005***, 44*, 2737-2753.

[34] Gahinet, P.; Nemirovshki, A.; Laub, A. J.; Chilali, M. In *LMI Control Toolbox;* The Mathworks: Chichester, England, **1995**.

[35] Morari, M.; Zafiriou, E. In *Robust process control;* Prentice Hall, Inc.: Englewood Cliffs, NJ, **1989**.

[36] Boyd, S.; El Ghaoui, L.; Feron, E.; Balakrishnan, V. In *Linear Matrix Inequalities in Systems and Control Theory;* SIAM Books: Philadelphia, 1994.

[37] Gahinet, P.; Apkarian, P. *International Journal of Robust and Nonlinear Control* **1994***, 4*, 421-448.

[38] Iwasaki, T.; Skelton, R. E. *Automatica* **1994***, 30*, 1307-1317.

[39] Sato, T.; Liu, K. *Systems and Control Letters* **1999***, 36*, 295-305.

[40] Apkarian, P.; Gahinet, P.; Becker, G. In *In Self-scheduled H infinity control of linear parameter-varying systems;* Part 1 (of 3); American Automatic Control Council, Green Valley, AZ, USA: Baltimore, MD, USA, **1994**; 1, pp 856-860.

[41] Kothare, M. V.; Balakrishnan, V.; Morari, M. *Automatica* **1996***, 32*, 1361-1379.

[42] Boyd, S.; Yang, Q. *Int J Control* **1989***, 49*, 2215-2240.

[43] Lyapunov, A. M. In *The General Problem of Motion Stability, in Russian.* Princeton University Press (1949), Ed.; Russia, **1892**.

[44] Slotine, J. E.; Li, W. In *Applied Nonlinear Control;* Pearson Education Inc.: Englewood Cliffs, New Jersey, **1991**.

[45] Doyle, J. *IEE Proceedings, Part D: Control Theory and Applications* **1982***, 129*, 242-250.

[46] Zhou, K.; Doyle, J. C.; Glover, K. In *Robust and optimal control;* Prentice Hall: Upper Saddle River, NJ, **1996**.

[47] Doyle, J. 1.; Packard, A.; Zhou, K. In *In Review of LFTs, LMIs, and &mu*; Proceedings of the 30th IEEE Conference on Decision and Control; Brighton, UK, **1991**; pp 1227-32.

[48] Packard, A.; Doyle, J. *Automatica* **1993***, 29*, 71-109.

[49] Doyle, J.; Packard, A. In *In uncertain multivariable systems from a state space perspective.* Proceedings of the 1987 American Control Conference: Minneapolis, USA, **1987**; pp 2147-2152.

[50] Kailath, T. In *Linear Systems;* Prentice Hall: Englewood Cliffs, N.J., **1980**.

[51] Ljung, L. In *System Identification: Theory for the user 2/E;* Prentice Hall PTR: New Jersey, USA, **1999**.

[52] Zhu, Y. In *Multivariable System Identification;* Elsevier Science, Ltd: Kidlington, Oxford, UK, **2001**.

[53] Gao, J.; Budman, H. M. *J. Process Control* **2005***, 15*, 807-817.

[54] Nocedal, J.; Wright, S. J. In *Numerical Optimization;* Springer: United States of America, **2006**.

[55] *Optimization Toolbox, For Use with MATLAB;* The MathWorks, Inc.: Natick, MA, **2006**.

[56] Shampine, L. F.; Reichelt, M. W. *SIAM Journal on Scientific Computing* **1997***, 18*, 1.

[57] Nagy, Z. K.; Braatz, R. D. *IEEE Trans. Control Syst. Technol.* **2003***, 11*, 694-704.

[58] Fan, M. K. H.; Tits, A. L.; Doyle, J. C. *IEEE Transactions on Automatic Control* **1991***, 36*, 25-38.

[59] Feron, E.; Apkarian, P.; Gahinet, P. *IEEE Transactions on Automatic Control* **1996***, 41*, 1041-1046.

[60] Siljak, D. D. *IEEE Transactions on Automatic Control* **1989***, 34*, 674-688.

[61] Zhou, S.; Lam, J.; Zheng, W. X. *IEEE Trans. Fuzzy Syst.* **2007***, 15*, 188-199.

[62] Downs, J. J.; Vogel, E. F. *Comput. Chem. Eng.* **1993***, 17*, 245-255.

[63] Ricker, N. L. *J. Process Control* **1996***, 6*, 205-221.

[64] Banerjee, A.; Arkun, Y. *Comput. Chem. Eng.* **1995***, 19*, 453-480.

[65] Larsson, T.; Hestetun, K.; Hovland, E.; Skogestad, S. *Industrial and Engineering Chemistry Research* **2001***, 40*, 4889-4901.

[66] Luyben, W. L. *Ind Eng Chem Res* **1996***, 35*, 3280-3289.

[67] Lyman, P. R.; Georgakis, C. *Comput. Chem. Eng.* **1995***, 19*, 321-331.

[68] McAvoy, T. J.; Ye, N. *Comput. Chem. Eng.* **1994***, 18*, 383-413.

[69] Sriniwas, G. R.; Arkun, Y. *J. Process Control* **1997***, 7*, 387-400.

[70] Vinson, D. R.; Georgakis, C.; Fossy, J. In *In Studies in plant-wide controllability using the Tennessee Eastman challenge problem, the case for multivariable control;* Part 1 (of 6); Seattle, WA, USA, **1995**; 1, pp 250-254.

[71] Wu, K.; Yu, C. *Ind Eng Chem Res* **1997***, 36*, 2239-2251.

[72] Seider, W. D.; Seader, J. D.; Lewin, D. R. In *Product and process design principles;* John Wiley and Sons, Inc.: USA, **2004**.

[73] Gill, P. E.; Murray, M.; Wright, M. H. In *Numerical Linear Algebra and Optimization, Vol. 1;* Addison Wesley: Redwood City, CA, **1991**.

[74] Han, S. P. *J. Optimization Theory and Applications* **1977***, 22*, 297.

Appendix A: Construction of the interconnection matrix (M) for a simple SISO problem.

Consider the case that an output variable **y** is affected by a single disturbance υ. Also, assume that **y** can be accurately described by the following robust FIR model (see equation 4.2):

$$\begin{aligned}
y(0) &= h_0\delta\upsilon(0) + \delta h_0\delta\upsilon(0) \\
y(1) &= h_0\delta\upsilon(1) + h_1\delta\upsilon(0) + \delta h_0\delta\upsilon(1) + \delta h_1\delta\upsilon(0) \\
y(2) &= h_0\delta\upsilon(2) + h_1\delta\upsilon(1) + h_2\delta\upsilon(0) + \delta h_0\delta\upsilon(2) + \delta h_1\delta\upsilon(1) + \delta h_2\delta\upsilon(0) \\
y(3) &= h_0\delta\upsilon(3) + h_1\delta\upsilon(2) + h_2\delta\upsilon(1) + h_3\delta\upsilon(0) + \delta h_0\delta\upsilon(3) + \delta h_1\delta\upsilon(2) + \delta h_2\delta\upsilon(1) + \delta h_3\delta\upsilon(0)
\end{aligned} \tag{A.1}$$

where the nominal and uncertain impulse response coefficients (h_i and δh_i) are obtained from (4.5) according to the procedure described in Section 4.1. The maximal change of the disturbance $\delta\upsilon(i)$ is assumed to have the same lower and upper bounds at each sampling instant, that is,

$$\delta\upsilon^l \le \delta\upsilon(i) \le \delta\upsilon^u \tag{A.2}$$

Thus,

$$\delta\upsilon(i) = \frac{\delta\upsilon^l - \delta\upsilon^u}{2} \tag{A.3}$$

Based on above descriptions, the perturbation matrix $\boldsymbol{\Delta}$ is defined as a 24 x 24 diagonal matrix whereas the interconnection matrix **M**, of similar dimensions as $\boldsymbol{\Delta}$, can be specified according to the matrix definitions given in Section 4.1, that is,

$$\mathbf{W}_q = \begin{bmatrix} k\delta\upsilon(0) & 0 & 0 & 0 \\ 0 & k\delta\upsilon(0) & 0 & 0 \\ 0 & 0 & k\delta\upsilon(0) & 0 \\ 0 & 0 & 0 & k\delta\upsilon(0) \\ 0 & k\delta\upsilon(1) & 0 & 0 \\ 0 & 0 & k\delta\upsilon(1) & 0 \\ 0 & 0 & 0 & k\delta\upsilon(1) \\ 0 & 0 & k\delta\upsilon(2) & 0 \\ 0 & 0 & 0 & k\delta\upsilon(2) \\ 0 & 0 & 0 & k\delta\upsilon(3) \end{bmatrix} \tag{A.4}$$

$$\mathbf{R}_q = \begin{bmatrix} \delta h_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \delta h_0 & 0 & 0 & \delta h_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \delta h_0 & 0 & 0 & \delta h_1 & 0 & \delta h_2 & 0 & 0 \\ 0 & 0 & 0 & \delta h_0 & 0 & 0 & \delta h_1 & 0 & \delta h_2 & \delta h_3 \end{bmatrix} \tag{A.5}$$

$$\mathbf{H}_q = \begin{bmatrix} h_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & h_1 & 0 & 0 & h_0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & h_2 & 0 & 0 & h_1 & 0 & h_0 & 0 & 0 \\ 0 & 0 & 0 & h_3 & 0 & 0 & h_2 & 0 & h_1 & h_0 \end{bmatrix} \tag{A.6}$$

$$\mathbf{T} = \begin{bmatrix} k & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & k & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & k & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & k \\ 0 & k & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & k & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & k & 0 \\ 0 & 0 & k & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & k & 0 & 0 & 0 \\ 0 & 0 & 0 & k & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{A.7}$$

Appendix B: Tennessee Eastman process information

The following information was taken from Downs and Vogel[62].

Table B.1 Available process measurements

Variable number	Variable name	Base case value	Units
xmeas(1)	A feed (stream 1)	0.25052	kscmh
xmeas(2)	D feed (stream 2)	3664.0	kg/h
xmeas(3)	E feed (stream 3)	4509.3	kg/h
xmeas(4)	A and C feed (stream 4)	9.3477	kscmh
xmeas(5)	Recycle flow (stream 8)	26.902	kscmh
xmeas(6)	Reactor feed rate (stream 6)	42.339	kscmh
xmeas(7)	Reactor pressure	2705.0	kPa gauge
xmeas(8)	Reactor level	75.0	%
xmeas(9)	Reactor temperature	120.4	°C
xmeas(10)	Purge rate (stream 9)	0.33712	kscmh
xmeas(11)	Product separator temperature	80.109	°C
xmeas(12)	Product separator level	50.0	%
xmeas(13)	Product separator pressure	2633.7	kPa gauge
xmeas(14)	Product separator underflow	25.16	m^3/h
xmeas(15)	Stripper level	50.0	%
xmeas(16)	Stripper pressure	3102.2	kPa gauge
xmeas(17)	Stripper underflow (stream 10)	22.949	m^3/h
xmeas(18)	Stripper temperature	65.731	°C
xmeas(19)	Stripper steam flow	230.31	kg/h
xmeas(20)	Compressor work	341.43	kW
xmeas(21)	Reactor cooling water outlet temperature	94.599	°C
xmeas(22)	Separator cooling water outlet temperature	77.297	°C
xmeas(23)	Component A, reactor feed (stream 6)	32.188	mol %
xmeas(24)	Component B, reactor feed (stream 6)	8.8933	mol %

Table B.1 Continues

Variable number	Variable name	Base case value	Units
xmeas(25)	Component C, reactor feed (stream 6)	26.383	mol %
xmeas(26)	Component D, reactor feed (stream 6)	6.8820	mol %
xmeas(27)	Component E, reactor feed (stream 6)	18.776	mol %
xmeas(28)	Component F, reactor feed (stream 6)	1.6567	mol %
xmeas(29)	Component A, purge gas (stream 9)	32.958	mol %
xmeas(30)	Component B, purge gas (stream 9)	13.823	mol %
xmeas(31)	Component C, purge gas (stream 9)	23.978	mol %
xmeas(32)	Component D, purge gas (stream 9)	1.2565	mol %
xmeas(33)	Component E, purge gas (stream 9)	18.579	mol %
xmeas(34)	Component F, purge gas (stream 9)	2.2633	mol %
xmeas(35)	Component G, purge gas (stream 9)	4.8436	mol %
xmeas(36)	Component H, purge gas (stream 9)	2.2986	mol %
xmeas(37)	Component D, products (stream 11)	0.01787	mol %
xmeas(38)	Component E, products (stream 11)	0.83570	mol %
xmeas(39)	Component F, products (stream 11)	0.09858	mol %
xmeas(40)	Component G, products (stream 11)	53.724	mol %
xmeas(41)	Component H, products (stream 11)	43.828	mol %

Table B.2 Process manipulated variables

Variable number	Variable name	Base case Value (%)	Low limit	High limit	Units
xmv(1)	D feed flow (stream 2)	63.053	0	5811	kg/h
xmv(2)	E feed (stream 3)	53.980	0	8354	kg/h
xmv(3)	A feed (stream 1)	24.644	0	1.017	kscmh
xmv(4)	A and C feed (stream 4)	61.302	0	15.25	kscmh
xmv(5)	Compressor recycle valve	22.210	0	100	%
xmv(6)	Purge valve (stream 9)	40.064	0	100	%
xmv(7)	Separator pot liquid flow (stream 10)	38.100	0	65.71	m^3/h
xmv(8)	Stripper liquid product flow (stream 11)	46.534	0	49.10	m^3/h
xmv(9)	Stripper steam valve	47.446	0	100	%
xmv(10)	Reactor cooling water flow	41.106	0	227.1	m^3/h
xmv(11)	Condenser cooling water flow	18.114	0	272.6	m^3/h
xmv(12)	Agitator speed	50.000	150	250	rpm

Each manipulated variable is specified by setting the corresponding xmv variable to a value between 0 and 100

Table B.3 Process operating constraints

Process variable	Normal operating limits		Shut down limits	
	Low limit	High limit	Low limit	High limit
Reactor pressure	none	2895 kPa	none	3000 kPa
Reactor level	50% (11.8 m^3)	100% (21.3 m^3)	2.0 m^3	24.0 m^3
Reactor temperature	none	150°C	none	175°C
Separator level	30% (3.3 m^3)	100% (9.0 m^3)	1.0 m^3	12.0 m^3
Stripper base level	30% (3.5 m^3)	100% (6.6 m^3)	1.0 m^3	8.0 m^3

Table B.4 Components' physical properties (at 100°C)

Comp.	Molecular weight	Liquid density (kg/m^3)	Liquid heat capacity (kJ/kg-°C)	Vapour heat capacity (kJ/kg-°C)	Heat of vaporization (kJ/kg)	Vapour pressure constants		
						A	B	C
A	2.0	--	--	14.6	--	--	--	--
B	25.4	--	--	2.04	--	--	--	--
C	28.0	--	--	1.05	--	--	--	--
D	32.0	299	7.66	1.85	202	20.81	-1444.0	259
E	46.0	365	4.17	1.87	372	21.24	-2114.0	266
F	48.0	328	4.45	2.02	372	21.24	-2114.0	266
G	62.0	612	2.55	0.712	523	21.32	-2748.0	233
H	76.0	617	2.45	0.628	486	22.10	-3318.0	250

The vapour pressure of each component is estimated from the Antonie equation.

Table B.5 Process disturbances

Variable number	Process variable	Type
idv(1)	A/C feed ratio, B composition constant (stream 4)	Step
idv(2)	B composition, A/C ratio constant (stream 4)	Step
idv (3)	D feed temperature (stream 2)	Step
idv (4)	Reactor cooling water inlet temperature	Step
idv (5)	Condenser cooling water inlet temperature	Step
idv (6)	A feed loss (stream 1)	Step
idv (7)	C header pressure loos-reduced availability (stream 4)	Step
idv (8)	A,B,C feed composition (stream 4)	Random variation
idv (9)	D feed temperature (stream 2)	Random variation
idv (10)	C feed temperature (stream 4)	Random variation

Table B.5 Continues

Variable number	Process variable	Type
idv (11)	Reactor cooling water inlet temperature	Random variation
idv (12)	Condenser cooling water inlet temperature	Random variation
idv (13)	Reaction kinetics	Slow drift
idv (14)	Reactor cooling water valve	Sticking
idv (15)	Condenser cooling water valve	Sticking
idv (16)	Unknown	Unknown
idv (17)	Unknown	Unknown
idv (18)	Unknown	Unknown
idv (19)	Unknown	Unknown
idv (20)	Unknown	Unknown

Table B.6 Process operating breakdown costs

Component costs	Cost ($/kgmol)
A	2.206
C	6.177
D	22.06
E	14.56
F	17.89
G	30.44
H	22.94
Compressor costs	$ 0.0536/kW-h
Stripper steam costs	$0.0318/kg

Based on the process operating cost function given in (5.14) and the information given in Table B.1 and Table B.6, the total operating costs at the base case for the Tennessee Eastman process are as follows:

Table B.7 Purge losses

Component costs	Mole fraction	Molar Costs	
A	0.329580	2.206	0.7271
C	0.239780	6.177	1.4811
D	0.012565	22.06	0.2773
E	0.185790	14.56	2.7051
F	0.022633	17.89	0.4049
G	0.048436	30.44	1.4745
H	0.022986	22.94	0.5274
		Costs per kgmol of purge	7.5973

Table B.8 Raw materials in the product's stream (stream 11)

Component costs	Mole fraction	Molar Costs	
D	0.00018	22.06	0.0040
E	0.00836	14.56	0.1217
F	0.00099	17.89	0.0177
		Costs per kgmol of product	0.1434

$$OP_{base\ case} = 7.5973\frac{\$}{kgmol}\left(15.099 kgmol/h\right) + 0.1434\frac{\$}{kgmol}\left(211.37 kgmol/h\right) + 0.0536\frac{\$}{kW-h}\left(341.4 kW\right) + 0.0318\frac{\$}{kg}\left(230.3 kg/h\right) = 170.6\$/h \quad (0.1)$$

Tennessee Eastman Code provided by Downs and Vogel[62]

```
C
C            Tennessee Eastman Process Control Test Problem
C
C               James J. Downs and Ernest F. Vogel
C
C             Process and Control Systems Engineering
C                  Tennessee Eastman Company
C                      P.O. Box 511
C                   Kingsport,TN  37662
C
C  Reference:
C    "A Plant-Wide Industrial Process Control Problem"
C    Presented at the AIChE 1990 Annual Meeting
C    Industrial Challenge Problems in Process Control,Paper #24a
C    Chicago,Illinois,November 14,1990
C
C  Subroutines:
C
C    TEFUNC - Function evaluator to be called by integrator
C    TEINIT - Initialization
C    TESUBi - Utility subroutines, i=1,2,..,8
C
C
C  The process simulation has 50 states (NN=50).  If the user wishes to
C  integrate additional states, NN must be increased accordingly in the
C  calling program.  The additional states should be appended to the end
C  of the YY vector, e.g. YY(51),...  The additional derivatives should
C  be appended to the end of the YP vector, e.g. YP(51),...  To initialize
C  the new states and to calculate derivatives for them, we suggest
C  creating new function evaluator and initialization routines as follows.
C
C        C-------------------------------------------------
C        C
C             SUBROUTINE FUNC(NN,TIME,YY,YP)
C        C
C             INTEGER NN
C             DOUBLE PRECISION TIME, YY(NN), YP(NN)
C        C
C        C  Call the function evaluator for the process
C        C
C             CALL TEFUNC(NN,TIME,YY,YP)
C        C
C        C  Calculate derivatives for additional states
C        C
C             YP(51) = ....
C             YP(52) = ....
C               .
C               .
C               .
C             YP(NN) = ....
C        C
C             RETURN
C             END
C        CC         C-------------------------------------------------
C        C
C             SUBROUTINE INIT(NN,TIME,YY,YP)
C        C
C             INTEGER NN
C             DOUBLE PRECISION TIME, YY(NN), YP(NN)
C        C
C        C  Call the initialization for the process
C        C
C             CALL TEINIT(NN,TIME,YY,YP)
C        C
C        C  Initialize additional states
C        C
```

```
C             YY(51) = ....
C             YY(52) = ....
C               .
C               .
C               .
C             YY(NN) = ....
C       C
C             RETURN
C             END
C       C
C       C-------------------------------------------------
C
C  Differences between the code and its description in the paper:
C
C  1.  Subroutine TEINIT has TIME in the argument list.  TEINIT sets TIME
C     to zero.
C
C  2.  There are 8 utility subroutines (TESUBi) rather than 5.
C
C  3.  Process disturbances 14 through 20 do NOT need to be used in
C     conjunction with another disturbance as stated in the paper.  All
C     disturbances can be used alone or in any combination.
C
C
C  Manipulated Variables
C
C    XMV(1)     A Feed Flow (stream 1)
C    XMV(2)     D Feed Flow (stream 2)
C    XMV(3)     E Feed Flow (stream 3)
C    XMV(4)     A and C Feed Flow (stream 4)
C    XMV(5)     Compressor Recycle Valve
C    XMV(6)     Purge Valve (stream 9)
C    XMV(7)     Separator Pot Liquid Flow (stream 10)
C    XMV(8)     Stripper Liquid Product Flow (stream 11)
C    XMV(9)     Stripper Steam Valve
C    XMV(10)    Reactor Cooling Water Flow
C    XMV(11)    Condenser Cooling Water Flow
C    XMV(12)    Agitator Speed
C
C  Continuous Process Measurements
C
C    XMEAS(1)   A Feed  (stream 1)                kscmh
C    XMEAS(2)   D Feed  (stream 2)                kg/hr
C    XMEAS(3)   E Feed  (stream 3)                kg/hr
C    XMEAS(4)   A and C Feed  (stream 4)          kscmh
C    XMEAS(5)   Recycle Flow  (stream 8)          kscmh
C    XMEAS(6)   Reactor Feed Rate  (stream 6)     kscmh
C    XMEAS(7)   Reactor Pressure                  kPa gauge
C    XMEAS(8)   Reactor Level                     %
C    XMEAS(9)   Reactor Temperature               Deg C
C    XMEAS(10)  Purge Rate (stream 9)             kscmh
C    XMEAS(11)  Product Sep Temp                  Deg C
C    XMEAS(12)  Product Sep Level                 %
C    XMEAS(13)  Prod Sep Pressure                 kPa gauge
C    XMEAS(14)  Prod Sep Underflow (stream 10)    m3/hr
C    XMEAS(15)  Stripper Level                    %
C    XMEAS(16)  Stripper Pressure                 kPa gauge
C    XMEAS(17)  Stripper Underflow (stream 11)    m3/hr
C    XMEAS(18)  Stripper Temperature              Deg C
C    XMEAS(19)  Stripper Steam Flow               kg/hr
C    XMEAS(20)  Compressor Work                   kW
C    XMEAS(21)  Reactor Cooling Water Outlet Temp Deg C
C    XMEAS(22)  Separator Cooling Water Outlet Temp  Deg C
C
C  Sampled Process Measurements
C
C    Reactor Feed Analysis (Stream 6)
C       Sampling Frequency = 0.1 hr
C       Dead Time = 0.1 hr
C       Mole %
```

```
C   XMEAS(23)   Component A
C   XMEAS(24)   Component B
C   XMEAS(25)   Component C
C   XMEAS(26)   Component D
C   XMEAS(27)   Component E
C   XMEAS(28)   Component F
C
C   Purge Gas Analysis (Stream 9)
C      Sampling Frequency = 0.1 hr
C      Dead Time = 0.1 hr
C      Mole %
C   XMEAS(29)   Component A
C   XMEAS(30)   Component B
C   XMEAS(31)   Component C
C   XMEAS(32)   Component D
C   XMEAS(33)   Component E
C   XMEAS(34)   Component F
C   XMEAS(35)   Component G
C   XMEAS(36)   Component H
C
C   Product Analysis (Stream 11)
C      Sampling Frequency = 0.25 hr
C      Dead Time = 0.25 hr
C      Mole %
C   XMEAS(37)   Component D
C   XMEAS(38)   Component E
C   XMEAS(39)   Component F
C   XMEAS(40)   Component G
C   XMEAS(41)   Component H
C
C  Process Disturbances
C
C   IDV(1)   A/C Feed Ratio, B Composition Constant (Stream 4)         Step
C   IDV(2)   B Composition, A/C Ratio Constant (Stream 4)              Step
C   IDV(3)   D Feed Temperature (Stream 2)                          Step
C   IDV(4)   Reactor Cooling Water Inlet Temperature                Step
C   IDV(5)   Condenser Cooling Water Inlet Temperature              Step
C   IDV(6)   A Feed Loss (Stream 1)                              Step
C   IDV(7)   C Header Pressure Loss - Reduced Availability (Stream 4)   Step
C   IDV(8)   A, B, C Feed Composition (Stream 4)           Random Variation
C   IDV(9)   D Feed Temperature (Stream 2)                 Random Variation
C   IDV(10)  C Feed Temperature (Stream 4)                 Random Variation
C   IDV(11)  Reactor Cooling Water Inlet Temperature       Random Variation
C   IDV(12)  Condenser Cooling Water Inlet Temperature     Random Variation
C   IDV(13)  Reaction Kinetics                          Slow Drift
C   IDV(14)  Reactor Cooling Water Valve                   Sticking
C   IDV(15)  Condenser Cooling Water Valve                 Sticking
C   IDV(16)  Unknown
C   IDV(17)  Unknown
C   IDV(18)  Unknown
C   IDV(19)  Unknown
C   IDV(20)  Unknown
C
C
C=======================================================================
C

      SUBROUTINE TEFUNC(NN,TIME,YY,YP)
C
C     Function Evaluator
C
C       Inputs:
C
C         NN   = Number of differential equations
C         Time = Current time(hrs)
C         YY   = Current state values
C
C       Outputs:
C
C         YP   = Current derivative values
```

```
C
      DOUBLE PRECISION YY(NN),YP(NN)
            include 'TEcommon.inc'
      DOUBLE PRECISION
     .UCLR,UCVR,UTLR,UTVR,
     .XLR,XVR,ETR,ESR,
     .TCR,TKR,DLR,
     .VLR,VVR,VTR,
     .PTR,PPR,
     .CRXR,RR,RH,
     .FWR,TWR,QUR,HWR,UAR,
     .UCLS,UCVS,UTLS,UTVS,
     .XLS,XVS,ETS,ESS,
     .TCS,TKS,DLS,
     .VLS,VVS,VTS,
     .PTS,PPS,
     .FWS,TWS,QUS,HWS,
     .UCLC,UTLC,XLC,
     .ETC,ESC,TCC,DLC,
     .VLC,VTC,QUC,
     .UCVV,UTVV,XVV,
     .ETV,ESV,TCV,TKV,
     .VTV,PTV,
     .VCV,VRNG,VTAU,
     .FTM,
     .FCM,XST,XMWS,
     .HST,TST,SFR,
     .CPFLMX,CPPRMX,CPDH,
     .TCWR,TCWS,
     .HTR,AGSP,
     .XDEL,XNS,
     .TGAS,TPROD,VST
      INTEGER
     .IVST
      COMMON/TEPROC/
     .UCLR(8),UCVR(8),UTLR,UTVR,
     .XLR(8),XVR(8),ETR,ESR,
     .TCR,TKR,DLR,
     .VLR,VVR,VTR,
     .PTR,PPR(8),
     .CRXR(8),RR(4),RH,
     .FWR,TWR,QUR,HWR,UAR,
     .UCLS(8),UCVS(8),UTLS,UTVS,
     .XLS(8),XVS(8),ETS,ESS,
     .TCS,TKS,DLS,
     .VLS,VVS,VTS,
     .PTS,PPS(8),
     .FWS,TWS,QUS,HWS,
     .UCLC(8),UTLC,XLC(8),
     .ETC,ESC,TCC,DLC,
     .VLC,VTC,QUC,
     .UCVV(8),UTVV,XVV(8),
     .ETV,ESV,TCV,TKV,
     .VTV,PTV,
     .VCV(12),VRNG(12),VTAU(12),
     .FTM(13),
     .FCM(8,13),XST(8,13),XMWS(13),
     .HST(13),TST(13),SFR(8),
     .CPFLMX,CPPRMX,CPDH,
     .TCWR,TCWS,
     .HTR(3),AGSP,
     .XDEL(41),XNS(41),
     .TGAS,TPROD,VST(12),IVST(12)
      INTEGER IDVWLK
      DOUBLE PRECISION
     .ADIST,
     .BDIST,
     .CDIST,
     .DDIST,
     .TLAST,
```

```
     .TNEXT,
     .HSPAN,
     .HZERO,
     .SSPAN,
     .SZERO,
     .SPSPAN
      COMMON/WLK/
     .ADIST(12),
     .BDIST(12),
     .CDIST(12),
     .DDIST(12),
     .TLAST(12),
     .TNEXT(12),
     .HSPAN(12),
     .HZERO(12),
     .SSPAN(12),
     .SZERO(12),
     .SPSPAN(12),
     .IDVWLK(12)
      DOUBLE PRECISION
     .AVP,BVP,CVP,
     .AH,BH,CH,
     .AG,BG,CG,
     .AV,
     .AD,BD,CD,
     .XMW
      COMMON/CONST/
     .AVP(8),BVP(8),CVP(8),
     .AH(8),BH(8),CH(8),
     .AG(8),BG(8),CG(8),
     .AV(8),
     .AD(8),BD(8),CD(8),
     .XMW(8)
      INTEGER NN,I
      DOUBLE PRECISION RG,
     .VPR,
     .FIN(8),
     .TIME,
     .FLMS,
     .DLP,
     .PR,
     .FLCOEF,
     .UAS,
     .UAC,
     .VOVRL,
     .UARLEV,
     .VPOS(12),
     .XMNS,
     .XCMP(41),
     .TMPFAC,
     .R1F,
     .R2F,
     .HWLK,
     .SWLK,
     .SPWLK,
     .TESUB7,
     .TESUB8
      DO 500 I=1,20
      IF(IDV(I).GT.0)THEN
      IDV(I)=1
      ELSE
      IDV(I)=0
      ENDIF
  500  CONTINUE
      IDVWLK(1)=IDV(8)
      IDVWLK(2)=IDV(8)
      IDVWLK(3)=IDV(9)
      IDVWLK(4)=IDV(10)
      IDVWLK(5)=IDV(11)
      IDVWLK(6)=IDV(12)
```

```
      IDVWLK(7)=IDV(13)
      IDVWLK(8)=IDV(13)
      IDVWLK(9)=IDV(16)
      IDVWLK(10)=IDV(17)
      IDVWLK(11)=IDV(18)
      IDVWLK(12)=IDV(20)
      DO 900 I=1,9
      IF(TIME.GE.TNEXT(I))THEN
      HWLK=TNEXT(I)-TLAST(I)
      SWLK=ADIST(I)+HWLK*(BDIST(I)+HWLK*
     .(CDIST(I)+HWLK*DDIST(I)))
      SPWLK=BDIST(I)+HWLK*
     .(2.D0*CDIST(I)+3.D0*HWLK*DDIST(I))
      TLAST(I)=TNEXT(I)
      CALL TESUB5(SWLK,SPWLK,ADIST(I),BDIST(I),CDIST(I),
     .DDIST(I),TLAST(I),TNEXT(I),HSPAN(I),HZERO(I),
     .SSPAN(I),SZERO(I),SPSPAN(I),IDVWLK(I))
      ENDIF
  900 CONTINUE
      DO 910 I=10,12
      IF(TIME.GE.TNEXT(I))THEN
      HWLK=TNEXT(I)-TLAST(I)
      SWLK=ADIST(I)+HWLK*(BDIST(I)+HWLK*
     .(CDIST(I)+HWLK*DDIST(I)))
      SPWLK=BDIST(I)+HWLK*
     .(2.D0*CDIST(I)+3.D0*HWLK*DDIST(I))
      TLAST(I)=TNEXT(I)
      IF(SWLK.GT.0.1D0)THEN
      ADIST(I)=SWLK
      BDIST(I)=SPWLK
      CDIST(I)=-(3.D0*SWLK+0.2D0*SPWLK)/0.01D0
      DDIST(I)=(2.D0*SWLK+0.1D0*SPWLK)/0.001D0
      TNEXT(I)=TLAST(I)+0.1D0
      ELSE
      ISD=-1
      HWLK=HSPAN(I)*TESUB7(ISD)+HZERO(I)
      ADIST(I)=0.D0
      BDIST(I)=0.D0
      CDIST(I)=DBLE(IDVWLK(I))/HWLK**2
      DDIST(I)=0.D0
      TNEXT(I)=TLAST(I)+HWLK
      ENDIF
      ENDIF
  910 CONTINUE
      IF(TIME.EQ.0.D0)THEN
      DO 950 I=1,12
      ADIST(I)=SZERO(I)
      BDIST(I)=0.D0
      CDIST(I)=0.D0
      DDIST(I)=0.D0
      TLAST(I)=0.0D0
      TNEXT(I)=0.1D0
  950 CONTINUE
      END IF
      XST(1,4)=TESUB8(1,TIME)-IDV(1)*0.03D0
     .-IDV(2)*2.43719D-3
      XST(2,4)=TESUB8(2,TIME)+IDV(2)*0.005D0
      XST(3,4)=1.D0-XST(1,4)-XST(2,4)
      TST(1)=TESUB8(3,TIME)+IDV(3)*5.D0
      TST(4)=TESUB8(4,TIME)
      TCWR=TESUB8(5,TIME)+IDV(4)*5.D0
      TCWS=TESUB8(6,TIME)+IDV(5)*5.D0
      R1F=TESUB8(7,TIME)
      R2F=TESUB8(8,TIME)
      DO 1010 I=1,3
      UCVR(I)=YY(I)
      UCVS(I)=YY(I+9)
      UCLR(I)=0.0
      UCLS(I)=0.0
 1010 CONTINUE
```

```
      DO 1020 I=4,8
      UCLR(I)=YY(I)
      UCLS(I)=YY(I+9)
 1020 CONTINUE
      DO 1030 I=1,8
      UCLC(I)=YY(I+18)
      UCVV(I)=YY(I+27)
 1030 CONTINUE
      ETR=YY(9)
      ETS=YY(18)
      ETC=YY(27)
      ETV=YY(36)
      TWR=YY(37)
      TWS=YY(38)
      DO 1035 I=1,12
      VPOS(I)=YY(I+38)
 1035 CONTINUE
      UTLR=0.0
      UTLS=0.0
      UTLC=0.0
      UTVV=0.0
      DO 1040 I=1,8
      UTLR=UTLR+UCLR(I)
      UTLS=UTLS+UCLS(I)
      UTLC=UTLC+UCLC(I)
      UTVV=UTVV+UCVV(I)
 1040 CONTINUE
      DO 1050 I=1,8
      XLR(I)=UCLR(I)/UTLR
      XLS(I)=UCLS(I)/UTLS
      XLC(I)=UCLC(I)/UTLC
      XVV(I)=UCVV(I)/UTVV
 1050 CONTINUE
      ESR=ETR/UTLR
      ESS=ETS/UTLS
      ESC=ETC/UTLC
      ESV=ETV/UTVV
      CALL TESUB2(XLR,TCR,ESR,0)
      TKR=TCR+273.15
      CALL TESUB2(XLS,TCS,ESS,0)
      TKS=TCS+273.15
      CALL TESUB2(XLC,TCC,ESC,0)
      CALL TESUB2(XVV,TCV,ESV,2)
      TKV=TCV+273.15
      CALL TESUB4(XLR,TCR,DLR)
      CALL TESUB4(XLS,TCS,DLS)
      CALL TESUB4(XLC,TCC,DLC)
      VLR=UTLR/DLR
      VLS=UTLS/DLS
      VLC=UTLC/DLC
      VVR=VTR-VLR
      VVS=VTS-VLS
      RG=998.9
      PTR=0.0
      PTS=0.0
      DO 1110 I=1,3
      PPR(I)=UCVR(I)*RG*TKR/VVR
      PTR=PTR+PPR(I)
      PPS(I)=UCVS(I)*RG*TKS/VVS
      PTS=PTS+PPS(I)
 1110 CONTINUE
      DO 1120 I=4,8
      VPR=DEXP(AVP(I)+BVP(I)/(TCR+CVP(I)))
      PPR(I)=VPR*XLR(I)
      PTR=PTR+PPR(I)
      VPR=DEXP(AVP(I)+BVP(I)/(TCS+CVP(I)))
      PPS(I)=VPR*XLS(I)
      PTS=PTS+PPS(I)
 1120 CONTINUE
      PTV=UTVV*RG*TKV/VTV
```

```
      DO 1130 I=1,8
      XVR(I)=PPR(I)/PTR
      XVS(I)=PPS(I)/PTS
 1130 CONTINUE
      UTVR=PTR*VVR/RG/TKR
      UTVS=PTS*VVS/RG/TKS
      DO 1140 I=4,8
      UCVR(I)=UTVR*XVR(I)
      UCVS(I)=UTVS*XVS(I)
 1140 CONTINUE
      RR(1)=DEXP(31.5859536-40000.0/1.987/TKR)*R1F
      RR(2)=DEXP(3.00094014-20000.0/1.987/TKR)*R2F
      RR(3)=DEXP(53.4060443-60000.0/1.987/TKR)
      RR(4)=RR(3)*0.767488334D0
      IF(PPR(1).GT.0.0.AND.PPR(3).GT.0.0)THEN
      R1F=PPR(1)**1.1544
      R2F=PPR(3)**0.3735
      RR(1)=RR(1)*R1F*R2F*PPR(4)
      RR(2)=RR(2)*R1F*R2F*PPR(5)
      ELSE
      RR(1)=0.0
      RR(2)=0.0
      ENDIF
      RR(3)=RR(3)*PPR(1)*PPR(5)
      RR(4)=RR(4)*PPR(1)*PPR(4)
      DO 1200 I=1,4
      RR(I)=RR(I)*VVR
 1200 CONTINUE
      CRXR(1)=-RR(1)-RR(2)-RR(3)
      CRXR(3)=-RR(1)-RR(2)
      CRXR(4)=-RR(1)-1.5D0*RR(4)
      CRXR(5)=-RR(2)-RR(3)
      CRXR(6)=RR(3)+RR(4)
      CRXR(7)=RR(1)
      CRXR(8)=RR(2)
      RH=RR(1)*HTR(1)+RR(2)*HTR(2)
      XMWS(1)=0.0
      XMWS(2)=0.0
      XMWS(6)=0.0
      XMWS(8)=0.0
      XMWS(9)=0.0
      XMWS(10)=0.0
      DO 2010 I=1,8
      XST(I,6)=XVV(I)
      XST(I,8)=XVR(I)
      XST(I,9)=XVS(I)
      XST(I,10)=XVS(I)
      XST(I,11)=XLS(I)
      XST(I,13)=XLC(I)
      XMWS(1)=XMWS(1)+XST(I,1)*XMW(I)
      XMWS(2)=XMWS(2)+XST(I,2)*XMW(I)
      XMWS(6)=XMWS(6)+XST(I,6)*XMW(I)
      XMWS(8)=XMWS(8)+XST(I,8)*XMW(I)
      XMWS(9)=XMWS(9)+XST(I,9)*XMW(I)
      XMWS(10)=XMWS(10)+XST(I,10)*XMW(I)
 2010 CONTINUE
      TST(6)=TCV
      TST(8)=TCR
      TST(9)=TCS
      TST(10)=TCS
      TST(11)=TCS
      TST(13)=TCC
      CALL TESUB1(XST(1,1),TST(1),HST(1),1)
      CALL TESUB1(XST(1,2),TST(2),HST(2),1)
      CALL TESUB1(XST(1,3),TST(3),HST(3),1)
      CALL TESUB1(XST(1,4),TST(4),HST(4),1)
      CALL TESUB1(XST(1,6),TST(6),HST(6),1)
      CALL TESUB1(XST(1,8),TST(8),HST(8),1)
      CALL TESUB1(XST(1,9),TST(9),HST(9),1)
      HST(10)=HST(9)
```

```
      CALL TESUB1(XST(1,11),TST(11),HST(11),0)
      CALL TESUB1(XST(1,13),TST(13),HST(13),0)
      FTM(1)=VPOS(1)*VRNG(1)/100.0
      FTM(2)=VPOS(2)*VRNG(2)/100.0
      FTM(3)=VPOS(3)*(1.D0-IDV(6))*VRNG(3)/100.0
      FTM(4)=VPOS(4)*(1.D0-IDV(7)*0.2D0)
     .*VRNG(4)/100.0+1.D-10
      FTM(11)=VPOS(7)*VRNG(7)/100.0
      FTM(13)=VPOS(8)*VRNG(8)/100.0
      UAC=VPOS(9)*VRNG(9)*(1.D0+TESUB8(9,TIME))/100.0
      FWR=VPOS(10)*VRNG(10)/100.0
      FWS=VPOS(11)*VRNG(11)/100.0
      AGSP=(VPOS(12)+150.0)/100.0
      DLP=PTV-PTR
      IF(DLP.LT.0.0)DLP=0.0
      FLMS=1937.6D0*DSQRT(DLP)
      FTM(6)=FLMS/XMWS(6)
      DLP=PTR-PTS
      IF(DLP.LT.0.0)DLP=0.0
      FLMS=4574.21D0*DSQRT(DLP)
     .*(1.D0-0.25D0*TESUB8(12,TIME))
      FTM(8)=FLMS/XMWS(8)
      DLP=PTS-760.0
      IF(DLP.LT.0.0)DLP=0.0
      FLMS=VPOS(6)*0.151169D0*DSQRT(DLP)
      FTM(10)=FLMS/XMWS(10)
      PR=PTV/PTS
      IF(PR.LT.1.0)PR=1.0
      IF(PR.GT.CPPRMX)PR=CPPRMX
      FLCOEF=CPFLMX/1.197D0
      FLMS=CPFLMX+FLCOEF*(1.0-PR**3)
      CPDH=FLMS*(TCS+273.15D0)*1.8D-6*1.9872D0
     .*(PTV-PTS)/(XMWS(9)*PTS)
      DLP=PTV-PTS
      IF(DLP.LT.0.0)DLP=0.0
      FLMS=FLMS-VPOS(5)*53.349D0*DSQRT(DLP)
      IF(FLMS.LT.1.D-3)FLMS=1.D-3
      FTM(9)=FLMS/XMWS(9)
      HST(9)=HST(9)+CPDH/FTM(9)
      DO 5020 I=1,8
      FCM(I,1)=XST(I,1)*FTM(1)
      FCM(I,2)=XST(I,2)*FTM(2)
      FCM(I,3)=XST(I,3)*FTM(3)
      FCM(I,4)=XST(I,4)*FTM(4)
      FCM(I,6)=XST(I,6)*FTM(6)
      FCM(I,8)=XST(I,8)*FTM(8)
      FCM(I,9)=XST(I,9)*FTM(9)
      FCM(I,10)=XST(I,10)*FTM(10)
      FCM(I,11)=XST(I,11)*FTM(11)
      FCM(I,13)=XST(I,13)*FTM(13)
 5020 CONTINUE
      IF(FTM(11).GT.0.1)THEN
      IF(TCC.GT.170.)THEN
      TMPFAC=TCC-120.262
      ELSEIF(TCC.LT.5.292)THEN
      TMPFAC=0.1
      ELSE
      TMPFAC=363.744/(177.-TCC)-2.22579488
      ENDIF
      VOVRL=FTM(4)/FTM(11)*TMPFAC
      SFR(4)=8.5010*VOVRL/(1.0+8.5010*VOVRL)
      SFR(5)=11.402*VOVRL/(1.0+11.402*VOVRL)
      SFR(6)=11.795*VOVRL/(1.0+11.795*VOVRL)
      SFR(7)=0.0480*VOVRL/(1.0+0.0480*VOVRL)
      SFR(8)=0.0242*VOVRL/(1.0+0.0242*VOVRL)
      ELSE
      SFR(4)=0.9999
      SFR(5)=0.999
      SFR(6)=0.999
      SFR(7)=0.99
```

```
      SFR(8)=0.98
      END IF
      DO 6010 I=1,8
      FIN(I)=0.0
      FIN(I)=FIN(I)+FCM(I,4)
      FIN(I)=FIN(I)+FCM(I,11)
 6010 CONTINUE
      FTM(5)=0.0
      FTM(12)=0.0
      DO 6020 I=1,8
      FCM(I,5)=SFR(I)*FIN(I)
      FCM(I,12)=FIN(I)-FCM(I,5)
      FTM(5)=FTM(5)+FCM(I,5)
      FTM(12)=FTM(12)+FCM(I,12)
 6020 CONTINUE
      DO 6030 I=1,8
      XST(I,5)=FCM(I,5)/FTM(5)
      XST(I,12)=FCM(I,12)/FTM(12)
 6030 CONTINUE
      TST(5)=TCC
      TST(12)=TCC
      CALL TESUB1(XST(1,5),TST(5),HST(5),1)
      CALL TESUB1(XST(1,12),TST(12),HST(12),0)
      FTM(7)=FTM(6)
      HST(7)=HST(6)
      TST(7)=TST(6)
      DO 6130 I=1,8
      XST(I,7)=XST(I,6)
      FCM(I,7)=FCM(I,6)
 6130 CONTINUE
      IF(VLR/7.8.GT.50.0)THEN
      UARLEV=1.0
      ELSEIF(VLR/7.8.LT.10.0)THEN
      UARLEV=0.0
      ELSE
      UARLEV=0.025*VLR/7.8-0.25
      ENDIF
      UAR=UARLEV*(-0.5*AGSP**2
     .+2.75*AGSP-2.5)*855490.D-6
      QUR=UAR*(TWR-TCR)
     .*(1.D0-0.35D0*TESUB8(10,TIME))
      UAS=0.404655*(1.0-1.0/(1.0+(FTM(8)/3528.73)**4))
      QUS=UAS*(TWS-TST(8))
     .*(1.D0-0.25D0*TESUB8(11,TIME))
      QUC=0.D0
      IF(TCC.LT.100.)QUC=UAC*(100.0-TCC)
      XMEAS(1)=FTM(3)*0.359/35.3145
      XMEAS(2)=FTM(1)*XMWS(1)*0.454
      XMEAS(3)=FTM(2)*XMWS(2)*0.454
      XMEAS(4)=FTM(4)*0.359/35.3145
      XMEAS(5)=FTM(9)*0.359/35.3145
      XMEAS(6)=FTM(6)*0.359/35.3145
      XMEAS(7)=(PTR-760.0)/760.0*101.325
      XMEAS(8)=(VLR-84.6)/666.7*100.0
      XMEAS(9)=TCR
      XMEAS(10)=FTM(10)*0.359/35.3145
      XMEAS(11)=TCS
      XMEAS(12)=(VLS-27.5)/290.0*100.0
      XMEAS(13)=(PTS-760.0)/760.0*101.325
      XMEAS(14)=FTM(11)/DLS/35.3145
      XMEAS(15)=(VLC-78.25)/VTC*100.0
      XMEAS(16)=(PTV-760.0)/760.0*101.325
      XMEAS(17)=FTM(13)/DLC/35.3145
      XMEAS(18)=TCC
      XMEAS(19)=QUC*1.04D3*0.454
      XMEAS(20)=CPDH*0.0003927D6
      XMEAS(20)=CPDH*0.29307D3
      XMEAS(21)=TWR
      XMEAS(22)=TWS
      ISD=0
```

```
      IF(XMEAS(7).GT.3000.0)ISD=1
      IF(VLR/35.3145.GT.24.0)ISD=1
      IF(VLR/35.3145.LT.2.0)ISD=1
      IF(XMEAS(9).GT.175.0)ISD=1
      IF(VLS/35.3145.GT.12.0)ISD=1
      IF(VLS/35.3145.LT.1.0)ISD=1
      IF(VLC/35.3145.GT.8.0)ISD=1
      IF(VLC/35.3145.LT.1.0)ISD=1
      IF(TIME.GT.0.0.AND.ISD.EQ.0)THEN
      DO 6500 I=1,22
      CALL TESUB6(XNS(I),XMNS)
      XMEAS(I)=XMEAS(I)+XMNS
 6500 CONTINUE
      ENDIF
      XCMP(23)=XST(1,7)*100.0
      XCMP(24)=XST(2,7)*100.0
      XCMP(25)=XST(3,7)*100.0
      XCMP(26)=XST(4,7)*100.0
      XCMP(27)=XST(5,7)*100.0
      XCMP(28)=XST(6,7)*100.0
      XCMP(29)=XST(1,10)*100.0
      XCMP(30)=XST(2,10)*100.0
      XCMP(31)=XST(3,10)*100.0
      XCMP(32)=XST(4,10)*100.0
      XCMP(33)=XST(5,10)*100.0
      XCMP(34)=XST(6,10)*100.0
      XCMP(35)=XST(7,10)*100.0
      XCMP(36)=XST(8,10)*100.0
      XCMP(37)=XST(4,13)*100.0
      XCMP(38)=XST(5,13)*100.0
      XCMP(39)=XST(6,13)*100.0
      XCMP(40)=XST(7,13)*100.0
      XCMP(41)=XST(8,13)*100.0
      IF(TIME.EQ.0.D0)THEN
      DO 7010 I=23,41
      XDEL(I)=XCMP(I)
      XMEAS(I)=XCMP(I)
 7010 CONTINUE
      TGAS=0.1
      TPROD=0.25
      ENDIF
      IF(TIME.GE.TGAS)THEN
      DO 7020 I=23,36
      XMEAS(I)=XDEL(I)
      CALL TESUB6(XNS(I),XMNS)
      XMEAS(I)=XMEAS(I)+XMNS
      XDEL(I)=XCMP(I)
 7020 CONTINUE
      TGAS=TGAS+0.1
      ENDIF
      IF(TIME.GE.TPROD)THEN
      DO 7030 I=37,41
      XMEAS(I)=XDEL(I)
      CALL TESUB6(XNS(I),XMNS)
      XMEAS(I)=XMEAS(I)+XMNS
      XDEL(I)=XCMP(I)
 7030 CONTINUE
      TPROD=TPROD+0.25
      ENDIF
      DO 9010 I=1,8
      YP(I)=FCM(I,7)-FCM(I,8)+CRXR(I)
      YP(I+9)=FCM(I,8)-FCM(I,9)-
     .FCM(I,10)-FCM(I,11)
      YP(I+18)=FCM(I,12)-FCM(I,13)
      YP(I+27)=FCM(I,1)+FCM(I,2)+
     .FCM(I,3)+FCM(I,5)+
     .FCM(I,9)-FCM(I,6)
 9010 CONTINUE
      YP(9)=HST(7)*FTM(7)-
     .HST(8)*FTM(8)+RH+QUR
```

```
c                       Here is the "correct" version of the separator energy balance:

c             YP(18)=HST(8)*FTM(8)-
c    .(HST(9)*FTM(9)-cpdh)-
c    .HST(10)*FTM(10)-
c    .HST(11)*FTM(11)+
c    .QUS

c                       Here is the original version

      YP(18)=HST(8)*FTM(8)-
     .HST(9)*FTM(9)-
     .HST(10)*FTM(10)-
     .HST(11)*FTM(11)+
     .QUS

      YP(27)=HST(4)*FTM(4)+
     .HST(11)*FTM(11)-
     .HST(5)*FTM(5)-
     .HST(13)*FTM(13)+
     .QUC
      YP(36)=HST(1)*FTM(1)+
     .HST(2)*FTM(2)+
     .HST(3)*FTM(3)+
     .HST(5)*FTM(5)+
     .HST(9)*FTM(9)-
     .HST(6)*FTM(6)
      YP(37)=(FWR*500.53*
     .(TCWR-TWR)-QUR*1.D6/1.8)/HWR
      YP(38)=(FWS*500.53*
     .(TCWS-TWS)-QUS*1.D6/1.8)/HWS
      IVST(10)=IDV(14)
      IVST(11)=IDV(15)
      IVST(5)=IDV(19)
      IVST(7)=IDV(19)
      IVST(8)=IDV(19)
      IVST(9)=IDV(19)
      DO 9020 I=1,12
      IF(TIME.EQ.0.D0 .OR.
     .DABS(VCV(I)-XMV(I)).GT.VST(I)*IVST(I))
     .VCV(I)=XMV(I)
      IF(VCV(I).LT.0.0)VCV(I)=0.0
      IF(VCV(I).GT.100.0)VCV(I)=100.0
      YP(I+38)=(VCV(I)-VPOS(I))/VTAU(I)
 9020 CONTINUE
      IF(TIME.GT.0.0 .AND. ISD.NE.0)THEN
      DO 9030 I=1,NN
      YP(I)=0.0
 9030 CONTINUE
      ENDIF
      RETURN
      END
C
C=======================================================================
C
      SUBROUTINE TEINIT(NN,TIME,YY,YP)
C
C     Initialization
C
C       Inputs:
C
C         NN   = Number of differential equations
C
C       Outputs:
C
C         Time = Current time(hrs)
C         YY   = Current state values
C         YP   = Current derivative values
C
```

```
      DOUBLE PRECISION XMEAS,XMV
      COMMON/PV/XMEAS(41),XMV(12)
      INTEGER IDV
      COMMON/DVEC/IDV(20)
      DOUBLE PRECISION G
      COMMON/RANDSD/G
      DOUBLE PRECISION
     .UCLR,UCVR,UTLR,UTVR,
     .XLR,XVR,ETR,ESR,
     .TCR,TKR,DLR,
     .VLR,VVR,VTR,
     .PTR,PPR,
     .CRXR,RR,RH,
     .FWR,TWR,QUR,HWR,UAR,
     .UCLS,UCVS,UTLS,UTVS,
     .XLS,XVS,ETS,ESS,
     .TCS,TKS,DLS,
     .VLS,VVS,VTS,
     .PTS,PPS,
     .FWS,TWS,QUS,HWS,
     .UCLC,UTLC,XLC,
     .ETC,ESC,TCC,DLC,
     .VLC,VTC,QUC,
     .UCVV,UTVV,XVV,
     .ETV,ESV,TCV,TKV,
     .VTV,PTV,
     .VCV,VRNG,VTAU,
     .FTM,
     .FCM,XST,XMWS,
     .HST,TST,SFR,
     .CPFLMX,CPPRMX,CPDH,
     .TCWR,TCWS,
     .HTR,AGSP,
     .XDEL,XNS,
     .TGAS,TPROD,VST
      INTEGER
     .IVST
      COMMON/TEPROC/
     .UCLR(8),UCVR(8),UTLR,UTVR,
     .XLR(8),XVR(8),ETR,ESR,
     .TCR,TKR,DLR,
     .VLR,VVR,VTR,
     .PTR,PPR(8),
     .CRXR(8),RR(4),RH,
     .FWR,TWR,QUR,HWR,UAR,
     .UCLS(8),UCVS(8),UTLS,UTVS,
     .XLS(8),XVS(8),ETS,ESS,
     .TCS,TKS,DLS,
     .VLS,VVS,VTS,
     .PTS,PPS(8),
     .FWS,TWS,QUS,HWS,
     .UCLC(8),UTLC,XLC(8),
     .ETC,ESC,TCC,DLC,
     .VLC,VTC,QUC,
     .UCVV(8),UTVV,XVV(8),
     .ETV,ESV,TCV,TKV,
     .VTV,PTV,
     .VCV(12),VRNG(12),VTAU(12),
     .FTM(13),
     .FCM(8,13),XST(8,13),XMWS(13),
     .HST(13),TST(13),SFR(8),
     .CPFLMX,CPPRMX,CPDH,
     .TCWR,TCWS,
     .HTR(3),AGSP,
     .XDEL(41),XNS(41),
     .TGAS,TPROD,VST(12),IVST(12)
      INTEGER IDVWLK
      DOUBLE PRECISION
     .ADIST,
     .BDIST,
```

```
     .CDIST,
     .DDIST,
     .TLAST,
     .TNEXT,
     .HSPAN,
     .HZERO,
     .SSPAN,
     .SZERO,
     .SPSPAN
      COMMON/WLK/
     .ADIST(12),
     .BDIST(12),
     .CDIST(12),
     .DDIST(12),
     .TLAST(12),
     .TNEXT(12),
     .HSPAN(12),
     .HZERO(12),
     .SSPAN(12),
     .SZERO(12),
     .SPSPAN(12),
     .IDVWLK(12)
      DOUBLE PRECISION
     .AVP,BVP,CVP,
     .AH,BH,CH,
     .AG,BG,CG,
     .AV,
     .AD,BD,CD,
     .XMW
      COMMON/CONST/
     .AVP(8),BVP(8),CVP(8),
     .AH(8),BH(8),CH(8),
     .AG(8),BG(8),CG(8),
     .AV(8),
     .AD(8),BD(8),CD(8),
     .XMW(8)
      INTEGER I,NN
      DOUBLE PRECISION YY(NN),
     .YP(NN),
     .TIME
      XMW(1)=2.0
      XMW(2)=25.4
      XMW(3)=28.0
      XMW(4)=32.0
      XMW(5)=46.0
      XMW(6)=48.0
      XMW(7)=62.0
      XMW(8)=76.0
      AVP(1)=0.0
      AVP(2)=0.0
      AVP(3)=0.0
      AVP(4)=15.92
      AVP(5)=16.35
      AVP(6)=16.35
      AVP(7)=16.43
      AVP(8)=17.21
      BVP(1)=0.0
      BVP(2)=0.0
      BVP(3)=0.0
      BVP(4)=-1444.0
      BVP(5)=-2114.0
      BVP(6)=-2114.0
      BVP(7)=-2748.0
      BVP(8)=-3318.0
      CVP(1)=0.0
      CVP(2)=0.0
      CVP(3)=0.0
      CVP(4)=259.0
      CVP(5)=265.5
      CVP(6)=265.5
```

CVP(7)=232.9
CVP(8)=249.6
AD(1)=1.0
AD(2)=1.0
AD(3)=1.0
AD(4)=23.3
AD(5)=33.9
AD(6)=32.8
AD(7)=49.9
AD(8)=50.5
BD(1)=0.0
BD(2)=0.0
BD(3)=0.0
BD(4)=-0.0700
BD(5)=-0.0957
BD(6)=-0.0995
BD(7)=-0.0191
BD(8)=-0.0541
CD(1)=0.0
CD(2)=0.0
CD(3)=0.0
CD(4)=-0.0002
CD(5)=-0.000152
CD(6)=-0.000233
CD(7)=-0.000425
CD(8)=-0.000150
AH(1)=1.0D-6
AH(2)=1.0D-6
AH(3)=1.0D-6
AH(4)=0.960D-6
AH(5)=0.573D-6
AH(6)=0.652D-6
AH(7)=0.515D-6
AH(8)=0.471D-6
BH(1)=0.0
BH(2)=0.0
BH(3)=0.0
BH(4)=8.70D-9
BH(5)=2.41D-9
BH(6)=2.18D-9
BH(7)=5.65D-10
BH(8)=8.70D-10
CH(1)=0.0
CH(2)=0.0
CH(3)=0.0
CH(4)=4.81D-11
CH(5)=1.82D-11
CH(6)=1.94D-11
CH(7)=3.82D-12
CH(8)=2.62D-12
AV(1)=1.0D-6
AV(2)=1.0D-6
AV(3)=1.0D-6
AV(4)=86.7D-6
AV(5)=160.D-6
AV(6)=160.D-6
AV(7)=225.D-6
AV(8)=209.D-6
AG(1)=3.411D-6
AG(2)=0.3799D-6
AG(3)=0.2491D-6
AG(4)=0.3567D-6
AG(5)=0.3463D-6
AG(6)=0.3930D-6
AG(7)=0.170D-6
AG(8)=0.150D-6
BG(1)=7.18D-10
BG(2)=1.08D-9
BG(3)=1.36D-11
BG(4)=8.51D-10

```
      BG(5)=8.96D-10
      BG(6)=1.02D-9
      BG(7)=0.D0
      BG(8)=0.D0
      CG(1)=6.0D-13
      CG(2)=-3.98D-13
      CG(3)=-3.93D-14
      CG(4)=-3.12D-13
      CG(5)=-3.27D-13
      CG(6)=-3.12D-13
      CG(7)=0.D0
      CG(8)=0.D0
      YY(1)=10.40491389
      YY(2)=4.363996017
      YY(3)=7.570059737
      YY(4)=0.4230042431
      YY(5)=24.15513437
      YY(6)=2.942597645
      YY(7)=154.3770655
      YY(8)=159.1865960
      YY(9)=2.808522723
      YY(10)=63.75581199
      YY(11)=26.74026066
      YY(12)=46.38532432
      YY(13)=0.2464521543
      YY(14)=15.20484404
      YY(15)=1.852266172
      YY(16)=52.44639459
      YY(17)=41.20394008
      YY(18)=0.5699317760
      YY(19)=0.4306056376
      YY(20)=7.9906200783D-03
      YY(21)=0.9056036089
      YY(22)=1.6054258216D-02
      YY(23)=0.7509759687
      YY(24)=8.8582855955D-02
      YY(25)=48.27726193
      YY(26)=39.38459028
      YY(27)=0.3755297257
      YY(28)=107.7562698
      YY(29)=29.77250546
      YY(30)=88.32481135
      YY(31)=23.03929507
      YY(32)=62.85848794
      YY(33)=5.546318688
      YY(34)=11.92244772
      YY(35)=5.555448243
      YY(36)=0.9218489762
      YY(37)=94.59927549
      YY(38)=77.29698353
      YY(39)=63.05263039
      YY(40)=53.97970677
      YY(41)=24.64355755
      YY(42)=61.30192144
      YY(43)=22.21000000
      YY(44)=40.06374673
      YY(45)=38.10034370
      YY(46)=46.53415582
      YY(47)=47.44573456
      YY(48)=41.10581288
      YY(49)=18.11349055
      YY(50)=50.00000000
      DO 200 I=1,12
      XMV(I)=YY(I+38)
      VCV(I)=XMV(I)
      VST(I)=2.0D0
      IVST(I)=0
  200 CONTINUE
      VRNG(1)=400.00
      VRNG(2)=400.00
```

VRNG(3)=100.00
VRNG(4)=1500.00
VRNG(7)=1500.00
VRNG(8)=1000.00
VRNG(9)=0.03
VRNG(10)=1000.
VRNG(11)=1200.0
VTR=1300.0
VTS=3500.0
VTC=156.5
VTV=5000.0
HTR(1)=0.06899381054D0
HTR(2)=0.05D0
HWR=7060.
HWS=11138.
SFR(1)=0.99500
SFR(2)=0.99100
SFR(3)=0.99000
SFR(4)=0.91600
SFR(5)=0.93600
SFR(6)=0.93800
SFR(7)=5.80000D-02
SFR(8)=3.01000D-02
XST(1,1)=0.0
XST(2,1)=0.0001
XST(3,1)=0.0
XST(4,1)=0.9999
XST(5,1)=0.0
XST(6,1)=0.0
XST(7,1)=0.0
XST(8,1)=0.0
TST(1)=45.
XST(1,2)=0.0
XST(2,2)=0.0
XST(3,2)=0.0
XST(4,2)=0.0
XST(5,2)=0.9999
XST(6,2)=0.0001
XST(7,2)=0.0
XST(8,2)=0.0
TST(2)=45.
XST(1,3)=0.9999
XST(2,3)=0.0001
XST(3,3)=0.0
XST(4,3)=0.0
XST(5,3)=0.0
XST(6,3)=0.0
XST(7,3)=0.0
XST(8,3)=0.0
TST(3)=45.
XST(1,4)=0.4850
XST(2,4)=0.0050
XST(3,4)=0.5100
XST(4,4)=0.0
XST(5,4)=0.0
XST(6,4)=0.0
XST(7,4)=0.0
XST(8,4)=0.0
TST(4)=45.
CPFLMX=280275.
CPPRMX=1.3
VTAU(1)=8.
VTAU(2)=8.
VTAU(3)=6.
VTAU(4)=9.
VTAU(5)=7.
VTAU(6)=5.
VTAU(7)=5.
VTAU(8)=5.
VTAU(9)=120.

```
      VTAU(10)=5.
      VTAU(11)=5.
      VTAU(12)=5.
      DO 300 I=1,12
      VTAU(I)=VTAU(I)/3600.
  300 CONTINUE
      G=1431655765.D0
      XNS(1)=0.0012D0
      XNS(2)=18.000D0
      XNS(3)=22.000D0
      XNS(4)=0.0500D0
      XNS(5)=0.2000D0
      XNS(6)=0.2100D0
      XNS(7)=0.3000D0
      XNS(8)=0.5000D0
      XNS(9)=0.0100D0
      XNS(10)=0.0017D0
      XNS(11)=0.0100D0
      XNS(12)=1.0000D0
      XNS(13)=0.3000D0
      XNS(14)=0.1250D0
      XNS(15)=1.0000D0
      XNS(16)=0.3000D0
      XNS(17)=0.1150D0
      XNS(18)=0.0100D0
      XNS(19)=1.1500D0
      XNS(20)=0.2000D0
      XNS(21)=0.0100D0
      XNS(22)=0.0100D0
      XNS(23)=0.250D0
      XNS(24)=0.100D0
      XNS(25)=0.250D0
      XNS(26)=0.100D0
      XNS(27)=0.250D0
      XNS(28)=0.025D0
      XNS(29)=0.250D0
      XNS(30)=0.100D0
      XNS(31)=0.250D0
      XNS(32)=0.100D0
      XNS(33)=0.250D0
      XNS(34)=0.025D0
      XNS(35)=0.050D0
      XNS(36)=0.050D0
      XNS(37)=0.010D0
      XNS(38)=0.010D0
      XNS(39)=0.010D0
      XNS(40)=0.500D0
      XNS(41)=0.500D0
      DO 500 I=1,20
      IDV(I)=0
  500 CONTINUE
      HSPAN(1)=0.2D0
      HZERO(1)=0.5D0
      SSPAN(1)=0.03D0
      SZERO(1)=0.485D0
      SPSPAN(1)=0.D0
      HSPAN(2)=0.7D0
      HZERO(2)=1.0D0
      SSPAN(2)=.003D0
      SZERO(2)=.005D0
      SPSPAN(2)=0.D0
      HSPAN(3)=0.25D0
      HZERO(3)=0.5D0
      SSPAN(3)=10.D0
      SZERO(3)=45.D0
      SPSPAN(3)=0.D0
      HSPAN(4)=0.7D0
      HZERO(4)=1.0D0
      SSPAN(4)=10.D0
      SZERO(4)=45.D0
```

```
      SPSPAN(4)=0.D0
      HSPAN(5)=0.15D0
      HZERO(5)=0.25D0
      SSPAN(5)=10.D0
      SZERO(5)=35.D0
      SPSPAN(5)=0.D0
      HSPAN(6)=0.15D0
      HZERO(6)=0.25D0
      SSPAN(6)=10.D0
      SZERO(6)=40.D0
      SPSPAN(6)=0.D0
      HSPAN(7)=1.D0
      HZERO(7)=2.D0
      SSPAN(7)=0.25D0
      SZERO(7)=1.0D0
      SPSPAN(7)=0.D0
      HSPAN(8)=1.D0
      HZERO(8)=2.D0
      SSPAN(8)=0.25D0
      SZERO(8)=1.0D0
      SPSPAN(8)=0.D0
      HSPAN(9)=0.4D0
      HZERO(9)=0.5D0
      SSPAN(9)=0.25D0
      SZERO(9)=0.0D0
      SPSPAN(9)=0.D0
      HSPAN(10)=1.5D0
      HZERO(10)=2.0D0
      SSPAN(10)=0.0D0
      SZERO(10)=0.0D0
      SPSPAN(10)=0.D0
      HSPAN(11)=2.0D0
      HZERO(11)=3.0D0
      SSPAN(11)=0.0D0
      SZERO(11)=0.0D0
      SPSPAN(11)=0.D0
      HSPAN(12)=1.5D0
      HZERO(12)=2.0D0
      SSPAN(12)=0.0D0
      SZERO(12)=0.0D0
      SPSPAN(12)=0.D0
      DO 550 I=1,12
      TLAST(I)=0.D0
      TNEXT(I)=0.1D0
      ADIST(I)=SZERO(I)
      BDIST(I)=0.D0
      CDIST(I)=0.D0
      DDIST(I)=0.D0
  550 CONTINUE
      TIME=0.0
      CALL TEFUNC(NN,TIME,YY,YP)
      RETURN
      END
C
C=======================================================================
C
      SUBROUTINE TESUB1(Z,T,H,ITY)
      DOUBLE PRECISION
     .AVP,BVP,CVP,
     .AH,BH,CH,
     .AG,BG,CG,
     .AV,
     .AD,BD,CD,
     .XMW
      COMMON/CONST/
     .AVP(8),BVP(8),CVP(8),
     .AH(8),BH(8),CH(8),
     .AG(8),BG(8),CG(8),
     .AV(8),
     .AD(8),BD(8),CD(8),
```

```
     .XMW(8)
      DOUBLE PRECISION Z(8),R,T,H,HI
      INTEGER ITY,I
      IF(ITY.EQ.0)THEN
      H=0.0D0
      DO 100 I=1,8
      HI=T*(AH(I)+BH(I)*T/2.D0+CH(I)*T**2/3.D0)
      HI=1.8D0*HI
      H=H+Z(I)*XMW(I)*HI
  100 CONTINUE
      ELSE
      H=0.0D0
      DO 200 I=1,8
      HI=T*(AG(I)+BG(I)*T/2.D0+
     .CG(I)*T**2/3.D0)
      HI=1.8D0*HI
      HI=HI+AV(I)
      H=H+Z(I)*XMW(I)*HI
  200 CONTINUE
      END IF
      IF(ITY.EQ.2)THEN
      R=3.57696D0/1.D6
      H=H-R*(T+273.15)
      ENDIF
      RETURN
      END
      SUBROUTINE TESUB2(Z,T,H,ITY)
      DOUBLE PRECISION
     .AVP,BVP,CVP,
     .AH,BH,CH,
     .AG,BG,CG,
     .AV,
     .AD,BD,CD,
     .XMW
      COMMON/CONST/
     .AVP(8),BVP(8),CVP(8),
     .AH(8),BH(8),CH(8),
     .AG(8),BG(8),CG(8),
     .AV(8),
     .AD(8),BD(8),CD(8),
     .XMW(8)
      INTEGER ITY,J
      DOUBLE PRECISION Z(8),T,H,TIN,HTEST,ERR,DH,DT
      TIN=T
      DO 250 J=1,100
      CALL TESUB1(Z,T,HTEST,ITY)
      ERR=HTEST-H
      CALL TESUB3(Z,T,DH,ITY)
      DT=-ERR/DH
      T=T+DT
  250 IF(DABS(DT).LT.1.D-12)GO TO 300
      T=TIN
  300 RETURN
      END
      SUBROUTINE TESUB3(Z,T,DH,ITY)
      DOUBLE PRECISION
     .AVP,BVP,CVP,
     .AH,BH,CH,
     .AG,BG,CG,
     .AV,
     .AD,BD,CD,
     .XMW
      COMMON/CONST/
     .AVP(8),BVP(8),CVP(8),
     .AH(8),BH(8),CH(8),
     .AG(8),BG(8),CG(8),
     .AV(8),
     .AD(8),BD(8),CD(8),
     .XMW(8)
      INTEGER ITY,I
```

```
      DOUBLE PRECISION Z(8),R,T,DH,DHI
      IF(ITY.EQ.0)THEN
      DH=0.0D0
      DO 100 I=1,8
      DHI=AH(I)+BH(I)*T+CH(I)*T**2
      DHI=1.8D0*DHI
      DH=DH+Z(I)*XMW(I)*DHI
  100 CONTINUE
      ELSE
      DH=0.0D0
      DO 200 I=1,8
      DHI=AG(I)+BG(I)*T+CG(I)*T**2
      DHI=1.8D0*DHI
      DH=DH+Z(I)*XMW(I)*DHI
  200 CONTINUE
      END IF
      IF(ITY.EQ.2)THEN
      R=3.57696D0/1.D6
      DH=DH-R
      ENDIF
      RETURN
      END
      SUBROUTINE TESUB4(X,T,R)
      DOUBLE PRECISION
     .AVP,BVP,CVP,
     .AH,BH,CH,
     .AG,BG,CG,
     .AV,
     .AD,BD,CD,
     .XMW
      COMMON/CONST/
     .AVP(8),BVP(8),CVP(8),
     .AH(8),BH(8),CH(8),
     .AG(8),BG(8),CG(8),
     .AV(8),
     .AD(8),BD(8),CD(8),
     .XMW(8)
      DOUBLE PRECISION V,R,X(8),T
      INTEGER I
      V=0.0
      DO 10 I=1,8
      V=V+X(I)*XMW(I)/
     .(AD(I)+(BD(I)+CD(I)*T)*T)
   10 CONTINUE
      R=1.0/V
      RETURN
      END
      SUBROUTINE TESUB5(S,SP,ADIST,BDIST,CDIST,DDIST,TLAST,
     .TNEXT,HSPAN,HZERO,SSPAN,SZERO,SPSPAN,IDVFLAG)
      DOUBLE PRECISION
     .S,
     .SP,
     .H,
     .S1,
     .S1P,
     .ADIST,
     .BDIST,
     .CDIST,
     .DDIST,
     .TLAST,
     .TNEXT,
     .HSPAN,
     .HZERO,
     .SSPAN,
     .SZERO,
     .SPSPAN,
     .TESUB7
      INTEGER  I,IDVFLAG
      I=-1
      H=HSPAN*TESUB7(I)+HZERO
```

```
S1=SSPAN*TESUB7(I)*IDVFLAG+SZERO
S1P=SPSPAN*TESUB7(I)*IDVFLAG
ADIST=S
BDIST=SP
CDIST=(3.D0*(S1-S)-H*(S1P+2.D0*SP))/H**2
DDIST=(2.D0*(S-S1)+H*(S1P+SP))/H**3
TNEXT=TLAST+H
RETURN
END
SUBROUTINE TESUB6(STD,X)
INTEGER I
DOUBLE PRECISION STD,X,TESUB7
X=0.D0
      DO I=1,12
            X=X+TESUB7(I)
      end do
      X=(X-6.D0)*STD
      RETURN
END
DOUBLE PRECISION FUNCTION TESUB7(I)
INTEGER I
DOUBLE PRECISION G,DMOD
COMMON/RANDSD/G
G=DMOD(G*9228907.D0,4294967296.D0)
IF(I.GE.0)TESUB7=G/4294967296.D0
IF(I.LT.0)TESUB7=2.D0*G/4294967296.D0-1.D0
RETURN
END
DOUBLE PRECISION FUNCTION TESUB8(I,T)
INTEGER  I
DOUBLE PRECISION  H,T
INTEGER IDVWLK
DOUBLE PRECISION
.ADIST,
.BDIST,
.CDIST,
.DDIST,
.TLAST,
.TNEXT,
.HSPAN,
.HZERO,
.SSPAN,
.SZERO,
.SPSPAN
COMMON/WLK/
.ADIST(12),
.BDIST(12),
.CDIST(12),
.DDIST(12),
.TLAST(12),
.TNEXT(12),
.HSPAN(12),
.HZERO(12),
.SSPAN(12),
.SZERO(12),
.SPSPAN(12),
.IDVWLK(12)
H=T-TLAST(I)
TESUB8=ADIST(I)+H*(BDIST(I)+H*(CDIST(I)+H*DDIST(I)))
RETURN
END
```

Appendix C: Basic MATLAB codes

This appendix summarizes the MATLAB codes used in this work.

Key MATLAB codes for the mixing tank process presented in chapter 3

```
clc
clear all
format long
warning off all
%%%%%%% MINIMIZATION OF THE VOLUME %%%%%%%%%%
%%%%% Introduding the initial values for Kc,z and Tsp %%%%%%%%%%
load x0 %loads the vectors x0,lb and ub.
gamma=1; % Multiples of gamma are to be used in the optimization...
par=[0;x0(3);370;0.1;0;0.15;0;x0(4);gamma;298;0.015;2.0;0.9];
options=optimset('largeScale','off','Display','iter','TolX',1e-1,...
   'TolFun',1e-1,'TolCon',1e-2,'DiffMinChange',0.01,'DiffMaxChange',1,...
   'MaxFunEvals',300);
 [xsol,Vd,exitflag,output,lambda,grad,hessian]=fmincon(@(xsol)objfun1...
    (xsol,par),x0,[],[],[],[],lb,ub,@(xsol)cofun(xsol,par),options);

function f=objfun1(xsol,par)

%%%%%%% OBJECTIVE FUNTION FOR THE MIXING TANK: Vd %%%%%%%%%%
OPT=optimset('largeScale','off','MaxTime',720000,'Maxiter',1000,...
   'MaxFunEvals',1000,'Display','iter','TolFun',1e-3,'TolX',1e-3,...
   'DiffMaxChange',0.1,'DiffMinChange',1e-3);

[Fhopt Vd]=fmincon(@(Fhopt)Id(Fhopt,xsol,par,1),0.15,[],[],[],[],0.05,0.15,[],OPT);
f=-Vd;

function fval=Id(xsol,Fh,par,flag)


par(6)=Fh;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
samper=timecon(xsol,par,par(5),1); % 1 for Volume, otherwise it's temperature
[tout inputs ynon Fc]=equation(xsol,par,2*samper,2);
in=inputs-par(3);
out=ynon(:,1)-par(1);
data=iddata(out,in,par(4));
[tmin,gamma]=lmitest(data);
if flag==1
   fval=-(par(1)+gamma*20*par(9));
else
   fval=gamma;
end

function samper=timecon(xsol,par,flag,TV)
[tout inputs ynon Fc]=equation(xsol,par,1000,1);
Tinit=(par(6)*350+xsol(3)*par(10))/(xsol(3)+par(6));
if TV==1
   out=ynon(:,1)-par(1);
else
   out=ynon(:,2)-Tinit;
end
if TV==1
   kp=out(end)-out(1);
   kp_t=63.2*kp/100;
   index=find(out>=kp_t);
```

```
    samper=tout(index(3));
else
    ymax=max(out);
    y_settling=3*ymax/100;
    init=find(out==ymax);
    out_n=out(init:end);
    index=find(out_n<=y_settling);
    taop=0.25*tout(index(3));
    index=find(tout>=taop);
    samper=tout(index(1));
end

function [c,ceq]=cofun(xsol,par)

OPT=optimset('largeScale','off','MaxTime',720000,'Maxiter',1000,...
    'MaxFunEvals',1000,'Display','iter','TolFun',1e-3,'TolX',1e-3,...
    'DiffMaxChange',0.1,'DiffMinChange',1e-3);

%%% MINIMUM VOLUME CONSTRAIT %%%%
[Fhopt gminV]=fmincon(@(Fhopt)Id(Fhopt,xsol,par,2),0.05,[],[],[],[],...
    0.05,0.15,[],OPT);
par(6)=Fhopt;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
Vlim=gminV*20*par(9)-par(1)+par(13);
%%% TEMPERATURE CONSTRAINT %%%%
[Fhopt gammaT]=fmincon(@(Fhopt)IdT(Fhopt,xsol,par),0.15,[],[],[],[],...
    0.05,0.15,[],OPT);
par(6)=Fhopt;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
upcon=xsol(3)+gammaT*20*par(9)-370;
lowcon=300-xsol(3)+gammaT*20*par(9);
%%% FC CONSTRAINT %%%
[Fhopt gfcmax]=fmincon(@(Fhopt)IdFc(Fhopt,xsol,par,1),0.15,[],[],[],[],...
    0.05,0.15,[],OPT);
par(6)=Fhopt;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
fcmax=par(7)+20*gfcmax*par(9)-par(12);

[Fhopt gfcmin]=fmincon(@(Fhopt)IdFc(Fhopt,xsol,par,2),0.15,[],[],[],[],...
    0.05,0.15,[],OPT);
par(6)=Fhopt;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
fcmin=par(7)+20*gfcmin*par(9)-par(12);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
c=[upcon;lowcon;Vlim;fcmin;fcmax];
ceq=[];

function f=IdT(xsol,Fh,par)
par(6)=Fh;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
samper=timecon(xsol,par,par(5),0); % 1 for Volume, tepm otherwise
[tout inputs ynon Fc]=equation(xsol,par,samper,2);
in=inputs-par(3);
out=ynon(:,2)-par(2);
data=iddata(out,in,par(4));
[tmin,gammaT]=lmitest(data);
f=-gammaT;
```

```
function f=IdFc(xsol,Fh,par,flag)

par(6)=Fh;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
samper=timecon(xsol,par,par(5),0); % 1 for Volume, otherwise it's temperature
[tout inputs ynon Fc]=equation(xsol,par,samper,2);
in=inputs-par(3);
out=ynon(:,2)-par(2);
data=iddata(out,in,par(4));
[tmin,gammaFc]=lmiperf(xsol,data,par);
if flag==1
   f=-gammaFc;
else
   f=gammaFc;
end

function [tmin gamma]=lmitest(data)
lm=arx(data,[2 2 1]);
lm=d2c(lm); %lm=pem(data,'sspar','can','dist','none','Ts',0,'init','zero','D',[]);
kk=length(lm.da);
da=[lm.da(2:end)' zeros(kk-1,kk-2)];
db=lm.db(2:end)';
dc=[];
dd=[];
[boxa mata mata0]=parambox(lm.a,da);
%%%%% Robust stability test, Lyapunov function: A'*P+P*A<0 %%%%%%%%%
[ma na]=size(mata);
[ka la]=size(boxa);
rowa=ma/ka;
aux=zeros(rowa,na);
setlmis([])
W=lmivar(1,[rowa 1]);
for ia=1:la
   a=0;
   for r=1:ka
      for j=1:rowa
         for o=1:na
            if mata(j+a,o)==1
               aux(j,o)=boxa(r,ia);
            end
         end
      end
      a=a+rowa;
   end
   A=mata0+aux;
   lmiterm([ia 1 1 W],1,A,'s');
end
lmiterm([la+1 1 1 0],1);
lmiterm([-la-1 1 1 W],1,1);
lmistab=getlmis;
[tmin xfeasp]=feasp(lmistab);
%%%%%%%%%%%%%%%%%%  PERFORMANCE TEST   %%%%%%%%%%%%%%
[boxb matb matb0]=parambox(lm.b(:,1),db);
[boxc matc matc0]=parambox(lm.c,dc);
[boxd matd matd0]=parambox(lm.d(:,1),dd);
[mb nb]=size(matb);
[kb lb]=size(boxb);
[mc nc]=size(matc);
[kc lc]=size(boxc);
[md nd]=size(matd);
[kd ld]=size(boxd);
aux=zeros(rowa,na);
rowb=mb/kb;
bux=zeros(rowb,nb);
```

```
rowc=mc/kc;
cux=zeros(rowc,nc);
rowd=md/kd;
dux=zeros(rowd,nd);
setlmis([])
X=lmivar(1,[rowa 1]);
Y=lmivar(1,[1 1]);
Z=lmivar(1,[rowd 1]);
lmiterm([-1 1 1 X],1,1);
lmiterm([-2 1 1 0],1);
lmiterm([-3 1 1 0],1);
z=4;
%%%%introducing the First SS matrix
for ia=1:la
   a=0;
   for r=1:ka
      for j=1:rowa
         for o=1:na
            if mata(j+a,o)==1
               aux(j,o)=boxa(r,ia);
            end
         end
      end
      a=a+rowa;
   end
   A=mata0+aux;
   %%%%introducing the second SS matrix
   for ib=1:lb
      a=0;
      for r=1:kb
         for j=1:rowb
            for o=1:nb
               if matb(j+a,o)==1
                  bux(j,o)=boxb(r,ib);
               end
            end
         end
         a=a+rowb;
      end
      B=matb0+bux;
      %%%%introducing the third SS matrix
      for ic=1:lc
         a=0;
         for r=1:kc
            for j=1:rowc
               for o=1:nc
                  if matc(j+a,o)==1
                     cux(j,o)=boxc(r,ic);
                  end
               end
            end
            a=a+rowc;
         end
         C=matc0+cux;
         %%%%introducing the fourth SS matrix
         for id=1:ld
            a=0;
            for r=1:kd
               for j=1:rowd
                  for o=1:nd
                     if matd(j+a,o)==1
                        dux(j,o)=boxd(r,id);
                     end
                  end
               end
               a=a+rowd;
            end
            D=matd0+dux;
            %%here is where we need to introduce the lmis%%%%%
```

```
            lmiterm([z 1 1 X],1,A,'s');
            lmiterm([z 1 2 X],1,B);
            lmiterm([z 1 3 0],C');
            lmiterm([z 2 2 0],0);
            lmiterm([z 2 3 0],D');
            lmiterm([z 3 3 0],0);
            lmiterm([-z 1 1 0],0);
            lmiterm([-z 2 2 Y],1,1);
            lmiterm([-z 3 3 Z],1,1);

            z=z+1;
         end
      end
   end
end

lmiterm([z 1 1 Y],1,1);
lmiterm([-z 1 1 0],1);
z=z+1;
lmiterm([z 1 1 Z],1,1);
lmiterm([-z 1 1 0],1);
lmiperf=getlmis;
[gamma xopt]=gevp(lmiperf,2);
```

Key MATLAB codes for the mixing tank process chapter 4

```
clc
clear all
format short e
warning off all
global kt0 kv0 kfc0
diary ('initialpoints.txt')
%%%%%%% MINIMIZATION OF THE VOLUME %%%%%%%%%%%
load x0 %loads the vectors x0,lb and ub.
   %%%%%%%%%%%%%Initial Conditions for the Tank %%%%%%%%%%%%%%%%
   Th=370;
   sampint=0.1;
   Fh=0.15;
   Tc=298.0;
   fcmin=0.015;
   fcmax=2.0;
   vmin=0.90;
   Fc0=-(Fh*(Th-x0(3)))/(Tc-x0(3));
   Vnon=((Fc0+Fh)/x0(2))^2;
   flag=0.0; % 1 plots in objfun and confun
   kv0=26.0; %Initial value for k0 in skewmu for V
   kt0=0.82; %Initial value for k0 in skewmu for T
   kfc0=0.82; %Initial value for k0 in skewmu for Fc
   par=[Vnon;x0(3);Th;sampint;flag;Fh;Fc0;x0(4);1.0;Tc;fcmin;fcmax;vmin];
   options=optimset('largeScale','off','Display','iter','TolX',1e-1,...
      'TolFun',1,'TolCon',1e-3,'DiffMinChange',1e-1,'DiffMaxChange',5e-1,'MaxFunEvals',300);
[xsol,Vd,exitflag,output,lambda,grad,hessian]=fmincon(@(xsol)objfun(xsol,par,lb,ub),x0,[],[],[],[],lb,ub,@(xsol)…
cofun(xsol,par,lb,ub),options);
diary off

function f=objfun(xsol,par)
%%%%%%% OBJECTIVE FUNTION FOR THE MIXING TANK: Vd %%%%%%%%%%%
OPT=optimset('largeScale','off','MaxTime',720000,'Maxiter',1000,...
   'MaxFunEvals',1000,'Display','iter','TolFun',1e-3,'TolX',1e-3,...
   'DiffMaxChange',0.1,'DiffMinChange',1e-3);

[Fhopt Vd]=fmincon(@(Fhopt)Id(Fhopt,xsol,par,1),0.15,[],[],[],[],0.05,0.15,[],OPT);
f=-Vd;

function fval=Id(xsol,Fh,par,flag)
par(6)=Fh;
```

```
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
samper=timecon(xsol,par,par(5),1); % 1 for Volume, otherwise it's temperature
[tout inputs ynon Fc]=equation(xsol,par,2*samper,2);
in=inputs-par(3);
out=ynon(:,1)-par(1);
data=iddata(out,in,par(4));
m=arx(data,[2 1 1]);
[y,msd]=simul(m,in);
[kv WC]=volume(xsol,msd,par,samper);
if flag==1
   fval=par(1)+kv;
else
   fval=par(1)-kv;
end

function [c,ceq]=cofun(xsol,par,lb,ub)

OPT=optimset('largeScale','off','MaxTime',720000,'Maxiter',1000,...
   'MaxFunEvals',1000,'Display','iter','TolFun',1e-3,'TolX',1e-3,...
   'DiffMaxChange',0.1,'DiffMinChange',1e-3);

%%% MINIMUM VOLUME CONSTRAIT %%%%
[Fhopt kminV]=fmincon(@(Fhopt)Id(Fhopt,xsol,par,2),0.05,[],[],[],[],...
   0.05,0.15,[],OPT);
par(6)=Fhopt;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
Vlim=par(13)-kminV;

%%% TEMPERATURE CONSTRAINT %%%%
[Fhopt kT]=fmincon(@(Fhopt)IdT(Fhopt,xsol,par),0.15,[],[],[],[],...
   0.05,0.15,[],OPT);
par(6)=Fhopt;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
upcon=xsol(3)+kT-370;
lowcon=300-xsol(3)+kT;

%%% FC CONSTRAINT %%%
[Fhopt kFc]=fmincon(@(Fhopt)IdFc(Fhopt,xsol,par,1),0.15,[],[],[],[],...
   0.05,0.15,[],OPT);
par(6)=Fhopt;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
fcmax=par(7)+kfc-par(12);

[Fhopt kFc]=fmincon(@(Fhopt)IdFc(Fhopt,xsol,par,2),0.15,[],[],[],[],...
   0.05,0.15,[],OPT);
par(6)=Fhopt;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
fcmin=par(11)-par(7)+kFc;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
c=[upcon;lowcon;Vlim;fcmin;fcmax];
ceq=[];

function f=IdT(xsol,Fh,par)

par(6)=Fh;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
samper=timecon(xsol,par,par(5),0); % 1 for Volume, tepm otherwise
[tout inputs ynon Fc]=equation(xsol,par,samper,2);
```

```
in=inputs-par(3);
out=ynon(:,2)-par(2);
data=iddata(out,in,par(4));
m=arx(data,[2 2 1]);
[y,msd]=simul(m,in);
[kT WC]=temperature(xsol,msd,par,samper);
f=-kT;




function f=IdFc(xsol,Fh,par,flag)

par(6)=Fh;
par(2)=xsol(3);
par(7)=-par(6)*(par(3)-xsol(3))/(par(10)-xsol(3));
par(1)=((par(6)+par(7))/xsol(2))^2;
samper=timecon(xsol,par,par(5),0); % 1 for Volume, otherwise it's temperature
[tout inputs ynon Fc]=equation(xsol,par,samper,2);
in=inputs-par(3);
out=ynon(:,2)-par(2);
data=iddata(out,in,par(4));
m=arx(data,[2 2 1]);
[y,msd]=simul(m,in);
[kFc WCFc]=mv(xsol,msd,par,samper);
if flag==1
   f=-kFc;
else
   f=kFc;
end

function [kv WC]=volume(xsol,msd,par,samper)

global kv0
datpt=30;
dt=par(4);
datimp=[];
msd=modgen(data,par,xsol);
for i=1:length(msd)
   y=[];
   mt=d2c(msd{i});
   M=zpk(tf([mt.b],[mt.a]));
   if i==1
      [y,time]=impulse(M);
      my=max(y)*0.03;
      index=find(y>my);
      Ts=time(index(end));
      y=[];
      pt=Ts/(dt*datpt);
   end
   [sr,time]=step(M,0:dt*pt:Ts);
   for n=2:length(sr)
      y(n)=sr(n)-sr(n-1);
   end
   if par(5)==1
      hold on
      figure(1)
      plot(time,y,'g--')
   end
   datimp=[datimp,y'];
end
if par(5)==1
   hold off
end
hii=[];
for j=1:length(time)
   hup=max(datimp(j,:));
   hlow=min(datimp(j,:));
```

```
   hmean=(hup+hlow)/2;
   hun=(hup-hlow)/2;
   hii=[hii;hmean,hun];
end
[kv WC]=skewmod(kv0,hii);
kv0=kv;

function [kfc WC]=mv(xsol,msd,par,samper)

global kfc0
Kc=xsol(1);
Tao=xsol(4);
c=tf([Kc*Tao Kc],[Tao 0]);
dt=par(4);
datpt=30;
datimp=[];
for i=1:length(msd)
   y=[];
   mt=d2c(msd{i});
   g=zpk(tf([mt.b],[mt.a]));
   s=zpk('s');
   pole=g.p{:};
   M=(g.k*s/((s-pole(1))*(s-pole(2))))*(-c);
   if i==1
      [y,time]=impulse(M,samper*6);
      my=min(y)*0.03;
      index=find(y>my);
      Ts=time(index(end));
      y=[];
      pt=Ts/(dt*datpt);
   end
   [sr,time]=step(M,0:dt*pt:Ts);
   for n=2:length(sr)
      y(n)=sr(n)-sr(n-1);
   end
   if par(5)==1
      hold on
      figure(1)
      plot(time,y,'g--')
   end
   datimp=[datimp,y'];
end
if par(5)==1
   hold off
end
hii=[];
for j=1:length(time)
   hup=max(datimp(j,:));
   hlow=min(datimp(j,:));
   hmean=(hup+hlow)/2;
   hun=(hup-hlow)/2;
   hii=[hii;hmean,hun];
end
[kfc WC]=skewmod(kfc0,hii);
kfc0=kfc;

function [kt WC]=temperature(xsol,msd,par,samper)

global kt0
dt=par(4);
datpt=30;
datimp=[];
for i=1:length(msd)
   y=[];
   mt=d2c(msd{i});
   M=zpk(tf([mt.b],[mt.a]));
   if i==1
      [y,time]=impulse(M);
      my=min(y)*0.03;
      index=find(y<my);
```

```
        Ts=time(index(end));
        y=[];
        pt=Ts/(dt*datpt);
    end
    [sr,time]=step(M,0:dt*pt:Ts);
    for n=2:length(sr)
        y(n)=sr(n)-sr(n-1);
    end
    if par(5)==1
        hold on
        figure(1)
        plot(time,y,'g--')
    end
    datimp=[datimp,y'];
end
if par(5)==1
    hold off
end
hii=[];
for j=1:length(time)
    hup=max(datimp(j,:));
    hlow=min(datimp(j,:));
    hmean=(hup+hlow)/2;
    hun=(hup-hlow)/2;
    hii=[hii;hmean,hun];
end
[kt WC]=skewmod(kt0,hii);
kt0=kt;

function [k WC2]=skewmod(k0,hi)
lb2=1e-8;
up2=10000;
Th=0.5;%it's assumed that Thnon=370
w2=0.5*(Th-(-Th))*ones(length(hi(:,1)),1);
hi(1,1)=1e-5;
hi(1,2)=1e-5*0.1;
H=[hi,w2];
crmul=3;
options=optimset('largeScale','off','display','off','maxFunEvals',300);
[k f exitflag output]=fmincon(@(k) 1/k,k0,[],[],[],[],lb2,up2,...
    @(k)muk(k,H,1.0,crmul),options);
[C,delta]=muk(k,H,0.0,crmul);
Bounds=C(1:2);
bigrowp=C(3:end);
[s1 s2]=size(H);
if crmul<=2
    m=1;
    P=1;
    for j=1:2
        rowp(m)=bigrowp(P);
        K=s1+P;
        rowp(m+1)=bigrowp(K);
        J=m+1;
        n=s1-2;
        for i=1:n
            K=K+s1-i;
            rowp(J+i)=bigrowp(K);
        end
        P=K+1;
        m=J+i+1;
    end
    r=length(rowp);
    d1=length(bigrowp(end-s1+1:end));
    rowp(r+1:r+d1)=bigrowp(end-s1+1:end);
else
    rowp=bigrowp(1:s1);
end
if crmul==1 | crmul==3
    WC2=k*Th*rowp(1:s1);
    %WC2=k*Th*rowp(s1+1:end-s1);
```

```
else
   %WC1=k*Fh*rowp(1:s1);
   WC2=k*Th*rowp(1:s1);
end


function [c,ceq]=muk(k,H,flag,crmul)

[s1 s2]=size(H);

N1=[];
delta=[];
%delta1=[];
p=s1;
N2=[];
N3=[];

for i=1:s1
   J=zeros(s1-i+1,s1);
   P=zeros(s1,s1-i+1);
   Q=zeros(s1,s1-i+1);
   P(i:end,1:end)=diag(H(1:s1-i+1,1));
   Q(i:end,1:end)=diag(H(i,2)*ones(s1-i+1,1));
   J(1:end,i:end)=k*diag(H(i,3)*ones(s1-i+1,1));
   N1=[N1;J];
   N2=[N2,P];
   N3=[N3,Q];
   delta=[delta;-p 0];
   p=p-1;
end

p=1;
N4=[];
for i=1:s1
   Na=[];
   for j=1:s1-i+1
      R=zeros(s1-i+1,s1-j+1);
      R(j,p)=k;
      Na=[Na,R];
   end
   if p==1
      [m1 m2]=size(Na);
      N4=[N4;Na];
   else
      [n1 n2]=size(Na);
      aux=zeros(n1,m2-n2);
      Na=[Na,aux];
      N4=[N4;Na];
   end
   p=p+1;
end

if crmul==3
   delta=[-1*ones(s1*2,1),ones(s1*2,1);ones(1,2)];

   for i=0:length(H)-1
      H1(i+1,1)=H(end-i,1);
      H2(i+1,1)=H(end-i,2);
      H3(i+1,1)=H(end-i,3);
   end

   M=[zeros(s1,2*s1),k*H3;diag(k*ones(s1,1)),...
     zeros(s1,s1),zeros(s1,1);H1',H2',1e-4];
 %pause
end

if crmul==2 %crossmultiplication is considered in M
   delta=[delta;delta;ones(s1,2)]; % including delta performance

   [m1 m2]=size(N4);
```

```
   M=[zeros(m1,m2*2),N1;...
      N4,zeros(m1,m2),zeros(size(N1));...
      N2,N3,diag(1e-4*ones(1,s1))];
end

if crmul==1 %crossmultiplication is NOT considered in M

   delta=[delta1;ones(s1,2)]; % including delta performance

   [m1 m2]=size(N2);

   M=[zeros(m2,m2),N1;...
      N2,diag(1e-4*ones(1,s1))];
   %pause

      %M=[zeros(m2,m2*2),N1;...
      %zeros(m2,m2*2),N5;...
      %N2,N6,diag(1e-4*ones(1,s1))];
end

if flag==1
   bnds=mussv(M,delta);
   c=k-bnds(2);
   ceq=[];
else
   [bnds,muinfo]=mussv(M,delta);
   [Vdelta,Vsigma,Vlmi]=mussvextract(muinfo);
   rowp=diag(Vdelta);
   c=[bnds,rowp'];
   ceq=Vdelta;
end
```

Key MATLAB codes for the TE problem presented chapter 5

```
clc
clear all
format short e
warning off all
t0=cputime;
global k0 ITER I XN C
%diary ('iterations.txt')
%%Initial guesses and bounds:
load xinit %loads the initial guesses and bounds for each potential dec. variable
decvar=1:13; % vector that defines which varaibles to be used as dec. variables
for i=1:length(decvar)
   x0(i,1)=(X0(decvar(i),1)-X0(decvar(i),3))./abs(X0(decvar(i),3)...
      -X0(decvar(i),2)); %Initial conditions normalized
end
k0=1000*ones(6,1);
max_iter=10000; %maximum number of function evalutions allowed
ITER=zeros(28,max_iter); % Variable used to save the values at each iteration
XN=zeros(length(decvar),max_iter); % Variable used to save the decvar values at each iteration
I=1; % Index used in ITER
id=0; %defines if the WC profile is plotted
options=optimset('largeScale','off','Display','iter','TolX',1e-2,...
   'TolFun',1e-2,'TolCon',1e-1,'MaxFunEvals',max_iter);
[xsol Cost exitflag output]=fmincon(@(xsol)CostFunc(xsol,X0,id),x0,...
   [],[],[],[],zeros(length(x0),1),ones(length(x0),1),@constr,options);
% diary off
function h_cost=CostFunc(xsol,X0,id)
global XN ITER I k0 C
t0=cputime;
F=7038;
Cost=0.22;
ndist=2;
```

```
M3=[0 0];
M4=M3;
par_unval=0.0;
idv8=0;
idv13=0;
Mode_1_Init
Ts_save=0.2;
amp=[0.01 0.003 0.25 0.25];
amp=amp(1:ndist);
input=[5 6 13 16];
input=input(1:length(amp));
tss=100;
l=length(xsol);
xsol=X0(1:l,3)+xsol.*abs(X0(1:l,3)-X0(1:l,2));
XN(:,I)=xsol;
x=[xsol;X0(l+1:end,1)];
le=[127 127];
Mi=idinput([le(1) 1 2],'prbs',[0 1],[-1 1]);
Mi1=idinput([le(2) 1 4],'prbs',[0 1],[-1 1]);
i=[100 200];
tss=100;
M1=[0 0;i(1)+Ts_save*[0:length(Mi)-1]' amp(1)*Mi;i(1)+Ts_save*length(Mi) 0];
M2=[0 0;i(2)+Ts_save*[0:length(Mi1)-1]' amp(2)*Mi1;i(2)+Ts_save*length(Mi1) 0];
tend=i(2)+Ts_save*length(Mi1);
try
   clear functions
   sim ('TE_II.mdl',[],simset('SrcWorkspace','current'))
   ss=find(tout==tss);
   %%the next test is to ensure that the system reaches a feasible ss
   pres_dif=abs(simout(ss-1,7)-x(9));
   temp_dif=abs(simout(ss-1,9)-x(11));
   if pres_dif>0.001*x(9) || temp_dif>0.01*x(11)
      disp('Warning SS-not reached')
      fff %if the above is true then catch becomes active
   end
   Fg=(4.541*62/100)*simout(:,40).*xmv(:,8); %flow rate of G in stream 11
   Fh=(4.541*76/100)*simout(:,41).*xmv(:,8); % flow rate of H in stream 11
   output=[Fg Fh simout(:,7:9) simout(:,40)]; %xmv(:,6) xmv(:,10)];
   no_output=min(size(output));
   ss_inf=zeros(no_output,1);
   k=zeros(no_output,1);
   in_wc=zeros(length(amp),1);
   for j=1:no_output
      H=[];
      for n=1:length(amp)
         %n
         switch n
            case 1
               i=100;
               start_sq=1+i/Ts_save;
               end_sq=(i+Ts_save*(length(Mi)))/Ts_save;
               ss_inf(j)=output(start_sq-1,j);
               Op_cost=OpCost(start_sq-1);
            case 2
               i=200;
               start_sq=1+i/Ts_save;
               end_sq=(i+Ts_save*(length(Mi1)))/Ts_save;
         end
         u=simout(start_sq:end_sq,input(n))-simout(start_sq-1,input(n));
         yout=output(start_sq:end_sq,j)-output(start_sq-1,j);
         data=iddata(yout,u/amp(n),Ts_save);
         model=arx(data,[0 le(n) 0]);
         my=max(abs(model.b))*0.05;
         index=find(abs(model.b)>my);
         hi=model.b(1:index(end));
         hun=model.db(1:index(end));
         H=[H;hi' hun' ones(length(hi),1)];
         in_wc(n)=length(hi);
      end
      if k0(j)>10000
```

```
   k0(j)=1000;
end
options=optimset('largeScale','off','display','off',...
   'maxFunEvals',1000,'TolFun',1e-2,'TolX',1e-2,'TolCon',1e-3);
k(j)=fmincon(@(k) -k,k0(j),[],[],[],[],1e-8,Inf,@(k)muk(k,H),options);
if k(j)>10000
   k0(j)=1000;
   k(j)=fmincon(@(k) -k,k0(j),[],[],[],[],1e-8,Inf,@(k)muk(k,H),options);
end
k0(j)=k(j);
if id>=1
   [s1,col]=size(H);
   delta=[-1*ones(s1*2,1),ones(s1*2,1);ones(1,2)]; %Perturbation matrix
   M=[zeros(s1,2*s1),k(j)*H(:,end);diag(k(j)*ones(s1,1)),...
      zeros(s1,s1),zeros(s1,1);H(:,1)',H(:,2)',1e-4]; %Interconection matrix
   [bnds,minf]=mussv(M,delta,'f');
   Vdelta=mussvextract(minf);
   rowp=diag(Vdelta);
   for m=1:length(amp)
      if m==1
         Wc=rowp(1:in_wc(m))*k(j)*amp(m);
      else
         Wc=rowp(sum(in_wc(1:m-1))+1:sum(in_wc(1:m)))*k(j)*amp(m);
      end
      WC=[0 0;[tss:Ts_save:tss+(in_wc(m)*Ts_save-Ts_save)]' Wc;...
         tss+in_wc(m)*Ts_save 0];
      eval(['WC',num2str(m),'=WC;'])
   end
   idv8=0;
   idv13=0;
   M1=WC1;
   M2=WC2;
   tend=max([WC1(end,1) WC2(end,1)])+10;
   try
      clear functions
      sim ('TE_sim.mdl',[],simset('SrcWorkspace','current'))
      Fg_sim=(4.541*62/100)*simout_sim(:,40).*xmv_sim(:,8); %flow rate of G in stream 11
      Fh_sim=(4.541*76/100)*simout_sim(:,41).*xmv_sim(:,8); % flow rate of H in stream 11
      output_sim=[Fg_sim Fh_sim simout_sim(:,7:9) simout_sim(:,40)]; %xmv(:,6) xmv(:,10)];
      st_sim=1+tss/Ts_save;
      yout_sim=output_sim(st_sim:end,:);
      K(1:6,1)=ITER(6:11,id)+output_sim(st_sim-1,:)';
      K(1:6,2)=-ITER(6:11,id)+output_sim(st_sim-1,:)';
      figure(j+4)
      subplot(4,2,1)
      plot(WC1(2:end,1)-tss,WC1(2:end,2)+0.485)
      xlabel('Time, hrs','FontWeight','Bold')
      ylabel('Comp. A, % mol','FontWeight','Bold')
      title('Worst profile Component A, Stream 4','FontWeight', 'Bold')
      set(gca,'Fontweight','Bold')
      set(gcf,'Color',[1 1 1])
      subplot(4,2,2)
      plot(WC2(2:end,1)-tss,WC2(2:end,2)+0.005)
      xlabel('Time, hrs','FontWeight','Bold')
      ylabel('Comp. B, % mol','FontWeight','Bold')
      title('Worst profile Component B, Stream 4','FontWeight', 'Bold')
      set(gca,'Fontweight','Bold')
      set(gcf,'Color',[1 1 1])
      subplot(4,2,3)
      plot(tout_sim(st_sim:end)-tss,yout_sim(:,1),...
         tout_sim(st_sim:end)-tss,F*ones(length(yout_sim),1),'r--')
      xlabel('Time, hrs','FontWeight','Bold')
      ylabel('Comp. G, kg/hr','FontWeight','Bold')
      title('Flowrate Product G, Stream 11', 'FontWeight', 'Bold')
      set(gca,'Fontweight','Bold')
      set(gcf,'Color',[1 1 1])
      subplot(4,2,4)
      plot(tout_sim(st_sim:end)-tss,yout_sim(:,2),...
         tout_sim(st_sim:end)-tss,F*ones(length(yout_sim),1),'r--')
      xlabel('Time, hrs','FontWeight','Bold')
```

```
        ylabel('Comp. H, kg/hr','FontWeight','Bold')
        title('Flowrate Product H, Stream 11', 'FontWeight', 'Bold')
        set(gca,'Fontweight','Bold')
        set(gcf,'Color',[1 1 1])
        subplot(4,2,5)
        plot(tout_sim(st_sim:end)-tss,yout_sim(:,6),...
           tout_sim(st_sim:end)-tss,1.05*output_sim(st_sim-1,6)*...
           ones(length(yout_sim),1),'r--',...
           tout_sim(st_sim:end)-tss,0.95*output_sim(st_sim-1,6)*...
           ones(length(yout_sim),1),'r--')
        xlabel('Time, hrs','FontWeight','Bold')
        ylabel('% mol G','FontWeight','Bold')
        Title('Quality in Product G, Stream 11', 'FontWeight', 'Bold')
        set(gca,'Fontweight','Bold')
        set(gcf,'Color',[1 1 1])
        subplot(4,2,6)
        plot(tout_sim(st_sim:end)-tss,yout_sim(:,3),...
           tout_sim(st_sim:end)-tss,2895*ones(length(yout_sim),1),'r--') %,...
        xlabel('Time, hrs','FontWeight','Bold')
        ylabel('Pressure, KPA','FontWeight','Bold')
        Title('Reactor''s Pressure', 'FontWeight', 'Bold')
        set(gca,'Fontweight','Bold')
        set(gcf,'Color',[1 1 1])
        subplot(4,2,7)
        plot(tout_sim(st_sim:end)-tss,yout_sim(:,4),...
           tout_sim(st_sim:end)-tss,30*ones(length(yout_sim),1),'r--') %,...
        xlabel('Time, hrs','FontWeight','Bold')
        ylabel('Level, %','FontWeight','Bold')
        Title('Reactor''s Level', 'FontWeight', 'Bold')
        set(gca,'Fontweight','Bold')
        set(gcf,'Color',[1 1 1])
        subplot(4,2,8)
        plot(tout_sim(st_sim:end)-tss,yout_sim(:,5),...
           tout_sim(st_sim:end)-tss,150*ones(length(yout_sim),1),'r--') %,...
        xlabel('Time, hrs','FontWeight','Bold')
        ylabel('Temperature, \circ C','FontWeight','Bold')
        Title('Reactor''s Temperature', 'FontWeight', 'Bold')
        set(gca,'Fontweight','Bold')
        set(gcf,'Color',[1 1 1])
      catch
        disp('Warning: catch active in Te_sim')
        error=lasterror;
        disp(error.message)
      end
   end
end

C=zeros(10,1);
C(1)=ss_inf(1)-1.02*ss_inf(2); %Fg<=1.02Fh
C(2)=0.98*ss_inf(2)-ss_inf(1); %Fg>=0.98Fh
C(3)=-0.05*(ss_inf(2)+ss_inf(1))+(k(1)+k(2)); %Fg+Fh<=1.05*F
C(4)=F-ss_inf(1)+k(1); %Fg>=Fg*
C(5)=F-ss_inf(2)+k(2); %Fh>Fh*
C(6)=abs(k(6))-5; %plus/minus 5 mol% G
C(7)=ss_inf(3)+k(3)-2895; % Max Reac Pressure const
C(8)=ss_inf(4)+k(4)-100; % Max Reac Level const
C(9)=30-ss_inf(4)+k(4); % Min Reac Level const
C(10)=ss_inf(5)+k(5)-150; % Max Reac Temp const
if id>=1
 disp(['Fg<=1.02Fh', num2str(C(1))])
 disp(['Fg>=0.98Fh', num2str(C(2))])
 disp(['Fg+Fh<=1.05*F', num2str(C(3))])
 disp(['Fg>=Fg*', num2str(C(4))])
 disp(['Fh>Fh*', num2str(C(5))])
 disp(['plus/minus 5 mol% G', num2str(C(6))])
 disp(['Max Reac Pressure const', num2str(C(7))])
 disp(['Max Reac Level const', num2str(C(7))])
 disp(['Min Reac Level const', num2str(C(8))])
 disp(['Max Reac Temp const', num2str(C(10))])
end
```

```
    CCcost=CapCost(x);
    VarCost=Cost*(ss_inf(1)+k(1)+ss_inf(2)+k(2)-2*F); % Var Cost in $/hr
    h_cost=CCcost+Op_cost+VarCost; %in $/hr
    flag=0;
    if I==1
       [[ss_inf;k0;k;I] [C;h_cost;CCcost;Op_cost;VarCost;0;0;0;0;0]]
    end
catch
    disp('Warning: catch is active')
    error=lasterror;
    disp(error.message)
    XN(:,I)
    h_cost=ITER(2,I-1)*1.3;
    C=abs(ITER(18:27,I-1))*(1.3);
    flag=1;
end
t_iter=cputime-t0;
if I==1
    t_iter
end
if flag==1
    vector=[I;h_cost;zeros(15,1);C;t_iter];
    flag=0;
else
    vector=[I;h_cost;CCcost;Op_cost;VarCost;k;ss_inf;C;t_iter];
end
ITER(1:length(vector),I)=vector;
I=I+1

function CCcost=CapCost(xsol)
% This function estimates the Capital Cost investment
%% Cost Function Specification
D=2*(((xsol(12)+156.5+3500)*0.028381685)/(8*pi))^(1/3); %Reac Diam. in cubic meters
L=4*D; %Assuming Equipment's length is 4 times the diameter
FM=1; % Assuming Carbon steel as the equipment's material.
P=30; % Maximum pressure for the three pieces of equipments (KPA)
Cp=1780*(L^0.87)*(D^1.23);
Cbm=Cp*(2.86+1.694*(FM)*(10.01-7.408*log(P)+1.395*(log(P))^2));
Cbm_1992=Cbm*(360/315);
Ctci=Cbm_1992;
r=0.2; %return on investment, typically r=20 %/yr
CCcost=r*Ctci/(24*365); %in $/hr

function [C,ceq]=constr(xsol)
global C
ceq=[];

% Base case initialization
u0=[63.053, 53.98, 24.644, 61.302, 22.21, 40.064, 38.10, 46.534, 47.446, 41.106, 18.114, 50];

for i=1:12;
    iChar=int2str(i);
    eval(['xmv',iChar,'_0=u0(',iChar,');'])
end

Fp_0=100;

r1_0=0.251/Fp_0;
r2_0=3664/Fp_0;
r3_0=4509/Fp_0;
r4_0=9.35/Fp_0;
r5_0=0.337/Fp_0;
r6_0=25.16/Fp_0;
r7_0=22.95/Fp_0;

Eadj_0=0;
SP17_0=80.1;
Ts_base=0.0005;

function [c,ceq]=muk(k,H)
```

```
% This function estimates mu given a value for the parameter k.
[s1,col]=size(H);
delta=[-1*ones(s1*2,1),ones(s1*2,1);ones(1,2)]; %Perturbation matrix
M=[zeros(s1,2*s1),k*H(:,end);diag(k*ones(s1,1)),...
   zeros(s1,s1),zeros(s1,1);H(:,1)',H(:,2)',1e-4]; %Interconection matrix

bnds=mussv(M,delta,'f');
c=k-bnds(2);
ceq=[];
return
```

Figure C.1 and Figure C.2 present the simulink code used to simulate the TE process.

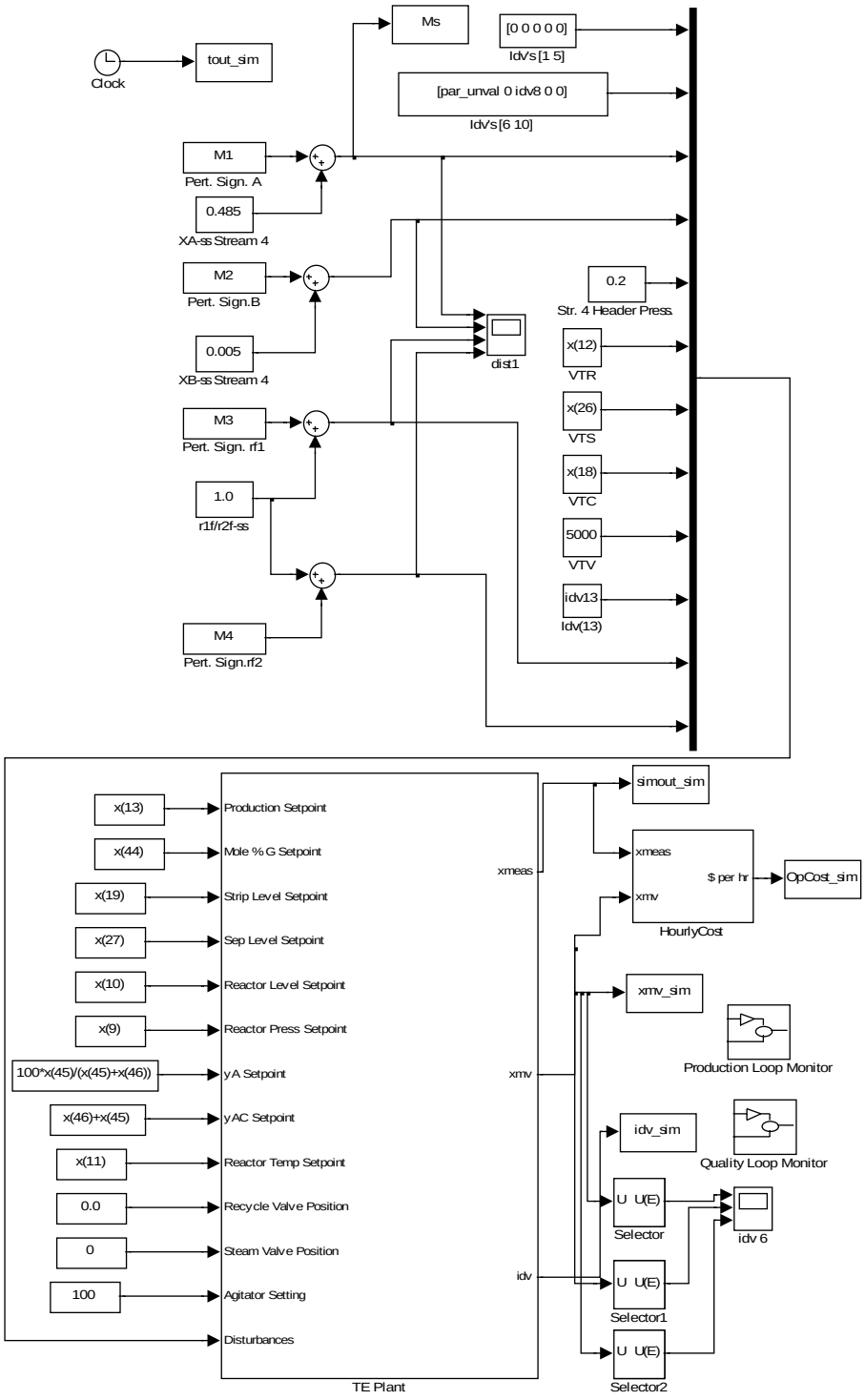

Figure C.1 Main simulink model used to simulate TE process .

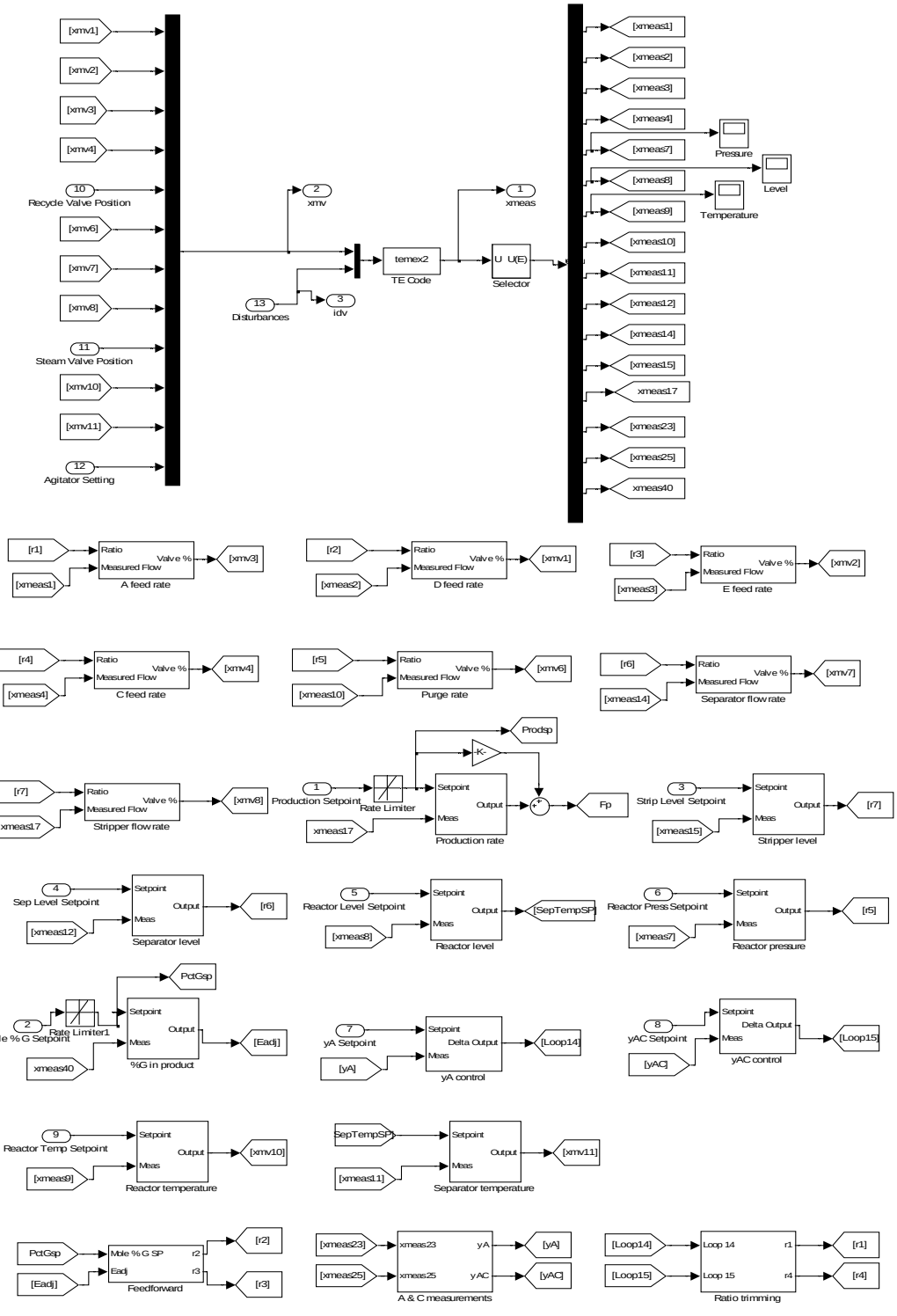

Figure C.2 TE Plant Simulink model[63]