UNIVERSIDADE ESTADUAL DE CAMPINAS

Faculdade de Engenharia Química

RAFAEL RIBEIRO SENCIO

RECONFIGURABLE CONTROL OF PROCESSES SUBJECTED TO ACTUATOR FAULTS: A TWO-LAYER MPC-BASED APPROACH

CONTROLE RECONFIGURÁVEL DE PROCESSOS SUJEITOS A FALHAS EM ATUADORES: UMA ABORDAGEM BASEADA NO MPC EM DUAS CAMADAS

CAMPINAS

2017

RAFAEL RIBEIRO SENCIO

RECONFIGURABLE CONTROL OF PROCESSES SUBJECTED TO ACTUATOR FAULTS: A TWO-LAYER MPC-BASED APPROACH

CONTROLE RECONFIGURÁVEL DE PROCESSOS SUJEITOS A FALHAS EM ATUADORES: UMA ABORDAGEM BASEADA NO MPC EM DUAS CAMADAS

Dissertation presented to the School of Chemical Engineering of the Universiy of Campinas in partial fulfillment of the requirements for the degree of Master in Chemical Engineering.

Dissertação apresentada à Faculdade de Engenharia Química da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Química.

Supervisor/Orientador: Prof. Dr. Flávio Vasconcelos da Silva
Co-supervisor/Coorientador: Prof. Dr. Thiago Vaz da Costa

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO DEFENDIDA PELO ALUNO RAFAEL RIBEIRO SENCIO E ORIENTADA PELO PROF. DR. FLÁVIO VASCONCELOS DA SILVA.

CAMPINAS
2017

Agência(s) de fomento e nº(s) de processo(s): CNPq, 130952/2015-0

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Área de Engenharia e Arquitetura
Luciana Pietrosanto Milla - CRB 8/8129

Se88r

Sencio, Rafael Ribeiro, 1991-
Reconfigurable control of processes subjected to actuator faults : a two-layer MPC-based approach / Rafael Ribeiro Sencio. – Campinas, SP : [s.n.], 2017.

Orientador: Flávio Vasconcelos da Silva.
Coorientador: Thiago Vaz da Costa.
Dissertação (mestrado) – Universidade Estadual de Campinas, Faculdade de Engenharia Química.

1. Controle de processos químicos. 2. Controle preditivo. 3. Falha de sistema (Engenharia). 4. Automação. I. Silva, Flávio Vasconcelos da,1971-. II. Costa, Thiago Vaz da,1982-. III. Universidade Estadual de Campinas. Faculdade de Engenharia Química. IV. Título.

Informações para Biblioteca Digital

Título em outro idioma: Controle reconfigurável de processos sujeitos a falhas em atuadores : uma abordagem baseada no MPC em duas camadas
Palavras-chave em inglês:
Chemical process control
Predictive control
System failure (Engineering)
Automation
Área de concentração: Sistemas de Processos Químicos e Informática
Titulação: Mestre em Engenharia Química
Banca examinadora:
Flávio Vasconcelos da Silva [Orientador]
Ely Carneiro de Paiva
Maurício Bezerra de Souza Júnior
Data de defesa: 20-02-2017
Programa de Pós-Graduação: Engenharia Química

Dissertação de Mestrado defendida por Rafael Ribeiro Sencio e aprovada em 20 de fevereiro de 2017 pela Comissão Examinadora constituída pelos doutores:

Prof. Dr. Flávio Vasconcelos da Silva
Faculdade de Engenharia Química
Universidade Estadual de Campinas

Prof. Dr. Ely Carneiro de Paiva
Faculdade de Engenharia Mecânica
Universidade Estadual de Campinas

Prof. Dr. Maurício Bezerra de Souza Júnior
Escola de Química
Universidade Federal do Rio de Janeiro

A ata de defesa com as respectivas assinaturas dos membros da Comissão Examinadora encontra-se no processo de vida acadêmica do aluno.

Aos meus pais,
meus eternos professores.

E a Orlando Sencio,
In memoriam.

"You are young and life is long and there is time to kill today
And then one day you find ten years have got behind you.
No one told you when to run, you missed the starting gun."
– Pink Floyd, Time

"It is far better to grasp the universe as it really is than to persist in delusion,
however satisfying and reassuring."
– Carl Sagan, The Demon-Haunted World: Science as a Candle in the Dark

Resumo

Plantas industriais modernas estão suscetíveis a falhas em equipamentos de processo e em instrumentos e componentes da malha de controle. Tais eventos anormais podem acarretar danos a equipamentos, degradação do desempenho do processo e até cenários extremos como a parada da planta e acidentes graves. Em vista disso, o emprego de sistemas de controle tolerante a falhas visa a elevar o grau de confiabilidade e segurança do processo por meio do tratamento e mitigação de eventos anormais, evitando que evoluam para situações críticas. Nesse sentido, este trabalho tem como objetivo desenvolver uma técnica de controle reconfigurável tolerante a falhas para processos sujeitos a falhas em atuadores. A presente proposta é baseada em abordagens por atuadores virtuais e ocultação da falha. Essas técnicas consistem no recálculo das ações de controle e na ocultação da falha do ponto de vista do controlador nominal, permitindo que o mesmo seja mantido após a reconfiguração da malha de controle. Na presente proposta, o atuador virtual é baseado na estrutura do controlador preditivo em duas camadas. Uma camada consiste no cálculo de referências para as variáveis de entrada e para o desvio previsto entre o comportamento da planta nominal e com falha. A outra camada, por sua vez, é responsável por conduzir as variáveis de processo para as referências calculadas na etapa anterior. Ambas as camadas são baseadas em problemas de programação quadrática e levam em consideração as restrições do processo, como limites de atuadores e desvios permissíveis em relação ao comportamento nominal da planta. Essa técnica possibilita a consideração de cenários de falhas nos quais não há graus de liberdade suficientes para a manutenção de variáveis controladas em valores desejados. Assim, a estimativa de perturbações permite que novas referências atingíveis sejam calculadas, ainda que haja erros de identificação do modelo pós-falha do processo. Por fim, a estrutura de controle proposta foi aplicada em simulações utilizando um processo de tanques quádruplos, bem como em experimentos conduzidos em uma planta de neutralização de pH.

Palavras-chave: Controle Tolerante a Falhas, Controle Reconfigurável, Controle Preditivo, Atuador Virtual.

Abstract

Modern industrial plants are susceptible to faults in process equipment and in instruments and components of the control loop. Such abnormal events can lead to equipment damage, degradation of process performance and even extreme scenarios such as plant shutdown and serious accidents. Thus, the use of fault-tolerant control systems aims to increase process reliability and safety by treating and mitigating abnormal events, preventing them from evolving to critical situations. In this sense, this work aims to develop a reconfigurable fault tolerant control technique for processes subject to actuator faults. The present proposal is based on the virtual actuator and fault hiding approaches. These techniques consist of recomputing control actions and hiding the fault from the nominal controller perspective, allowing it to be maintained after the control loop reconfiguration. We propose a virtual actuator based on the two-layer model predictive control structure. One layer consists of calculating references for input variables and for the predicted deviation between the nominal and faulty plant behaviors. The other layer, in turn, is responsible for driving process variables to the references calculated in the previous step. Both layers are based on quadratic programming problems and take into account process constraints such as actuator limits and permissible deviations from the nominal plant behavior. This technique allows the consideration of fault scenarios in which there are not enough degrees of freedom for the maintenance of controlled variables in desired values. Thus, disturbance estimation allows the calculation of new achievable references, even though there are identification errors in the post-fault model. Finally, the proposed control structure has been applied to an experimental pH neutralization plant. Finally, the proposed control structure was applied in simulations to a quadruple-tank process as well as in experiments conducted in a pH neutralization plant.

Keywords: Fault Tolerant Control, Reconfigurable Control, Model Predictive Control, Virtual Actuator.

List of Figures

List of Tables

List of Abbreviations and Acronyms

AFTCS	Active Fault Tolerant Control System
DMC	Dynamic Matrix Control
FDD	Fault Detection and Diagnosis
FTC	Fault Tolerant Control
FTCS	Fault Tolerant Control System
PFTCS	Passive Fault Tolerant Control System
MHVA	Moving Horizon Virtual Actuator
MPC	Model Predictive Control
MPHC	Model Predictive Heuristic Control
RC	Reconfigurable Control

List of Symbols

η	slack variables vector
A	state matrix
B	input matrix
B_d	input disturbance matrix
B_m	input matrix of the nominal model
B_p	input matrix of the post-fault model
B_s	faulty input matrix
C	output matrix
C_d	output disturbance matrix
d	disturbance vector
$\hat{d}$	disturbance estimate vector
I	identity matrix
L_d	Kalman filter gain related to the disturbance
L_x	Kalman filter gain related to the state
N	prediction horizon length
N_c	control horizon length
P	terminal penalty matrix
Q	state penalty matrix
Q_t	output deviation target penalty matrix
Q_y	output penalty matrix
R	input penalty matrix
R_t	input target penalty matrix
S	input rate of change penalty matrix
S_t	slack variable penalty matrix

s_t	slack variable penalty vector
u	input vector
u_m	input vector of the nominal model
u_p	input vector of the post-fault model
u_{max}	input upper limit vector
u_{min}	input lower limit vector
u_{sp}	input setpoint vector
u_s	faulty input vector
$\bar{u}$	input target vector
$\bar{u}_p$	input target vector of the post-fault model
x	state vector
x_m	state vector of the nominal model
x_p	state vector of the post-fault model
x_Δ	state deviation vector
$\bar{x}$	state target vector
$\bar{x}_\Delta$	state deviation target vector
$\hat{x}$	state estimate vector
y	output vector
y_m	output vector of the nominal model
y_p	output vector of the post-fault model
y_Δ	output deviation vector
y_{max}	output upper limit vector
y_{min}	output lower limit vector
y_{sp}	output setpoint vector
$\bar{y}_p$	output target of the post-fault model
$\bar{y}_\Delta$	output deviation target vector

Contents

CHAPTER 1

Introduction

Following several global changes, industrial processes have undergone extensive modernization in recent decades. The growth of the world economy and the production of goods and products has resulted in the search for more efficient and economic processes. Thus, with the expansion of production and competition on the market, tasks such as increasing productivity and reducing energy consumption often led to the development of more complex chemical processes. Therefore, in order to attend new performance requirements, it was necessary to increase the degree of automation and the use of advanced control and monitoring systems, which also increased the level of instrumentation in the plants.

Like all technological processes, industrial systems are subject to failure or malfunction of equipment and components of the control loop. In addition, the greater the level of instrumentation employed in monitoring and controlling the process, the greater the number of points susceptible to failures. Malfunctioning actuators, for example, can significantly decrease performance of control systems, while poorly calibrated sensors generate inaccurate measurements and are responsible for deviations from desired operating points (BLANKE et al., 2006). A conventional feedback loop may present poor performance or even instability in the presence of faults in any of the system components. In light of this, it is necessary to apply techniques capable of dealing with abnormal events, thus avoiding the degradation of process behavior and plant shutdown, or the occurrence of damages to equipment and to the environment and even catastrophes.

A control loop capable of automatically dealing with malfunction of its components while maintaining stability and acceptable process performance is classified as a fault tolerant control system (FTCS) (ZHANG; JIANG, 2008). In general, FTCS can be classified as passive (PFTCS) or active (AFTCS), where PFTCS are based on robust control laws that deal with predetermined fault scenarios and AFTCS perform the recon-

figuration of the control loop – also called reconfigurable control (RC) – and require fault detection and diagnosis (FDD) systems.

The development and study in the FTCS area was mostly stimulated by aircraft control systems design, especially after the occurrence of aircraft accidents (PATTON, 1997). For example, during flight 1080 of Delta company in April 1977, there was no indication of an elevator, compromising the control of vertical movement of the aircraft. In this case, the pilot was fortunately able to regain the airplane stability using physical redundancies. Another example is the American Airlines (Flight 191) crash in Chicago in May 1979 that was caused by an engine loss in a wing. However, further investigations concluded that the crash could have been avoided if the alarm system had worked (MONTOYA et al., 1983).

In the context of the chemical process industry, the use of fault-tolerant control systems was also intensified after some incidents such as the Three Mile Island nuclear plant in Pennsylvania, USA, in 1979, as well as the well-known Chernobyl disaster in 1986 (ZHANG; JIANG, 2008). Since then, several methods have been developed and new strategies have been research subjects in the last years (YU; JIANG, 2015; ROTONDO et al., 2015; GAO; DING; CECATI, 2015; YANG; GE; SUN, 2015; TABATABAEIPOUR; STOUSTRUP; BAK, 2015; COSTA, 2014; ZHANG et al., 2014; ROTONDO; NEJJARI; PUIG, 2014). It should be emphasized here that, due to the complexity of FTCS techniques, most of the works treat FDD and RC separately. Thus, many RC strategies assume that an FDD is available to provide post-fault information.

Currently, the applications of FTCS strategies are diverse and can be found not only in process, manufacturing, automotive, aerospace and nuclear industries, but also in electronic systems, communications networks and even software (ZHANG; JIANG, 2008).

1.1 Motivation and objectives

Most reconfigurable control strategies eliminate the nominal controller from the reconfigured control loop, that is, the controller used in non-fault situations is discarded. However, the methods presented in Steffen (2005) and Lunze and Steffen (2006) perform the control reconfiguration by means of a virtual actuator, characterized by a reconfiguration block inserted between the nominal controller and plant actuators. This structure aims to modify the output signal of the nominal controller in order to produce the same control effort previously applied to the nominal plant. This is a less intrusive approach since it maintains the knowledge about the plant that was incorporated in the nominal controller design (RICHTER; LUNZE; SCHLAGE, 2007; RICHTER, 2011).

As an extension to this technique, recent works (COSTA et al., 2013; COSTA, 2014) presented a proposal of a moving horizon-based virtual actuator that aims to minimize the deviation between nominal and faulty plant behaviors. This approach allows the consideration of physical limits of actuators and other process constraints in the

controller design.

However, these studies do not consider fault situations in which plant controllability is compromised and setpoint values become unreachable. Thus, without available redundancies, the remaining degrees of freedom in the system may be insufficient to conduct controlled variables to their desired levels. Such scenario, in the absence of appropriate fault tolerant control techniques, leads to the overload of the remaining actuators, which may lead to saturation in its limits, as well as a significant reduction in the performance of the control loop, not to mention its instability. In addition, even though the process is stabilized, attempting to maintain unreachable setpoints in a fault scenario may produce offsets in the controlled variables. Thus, the steady state configuration of the process variables may not be the best option available from an economic or safety point of view.

In this sense, more interesting operational points can be obtained by recalculating the setpoints taking into account the post-fault model and constraints both in the limits of actuators and in the permissible deviations between nominal and faulty plant behaviors. In addition, conserving controlled variables within acceptable ranges avoids unnecessary plant shutdowns and allows the process to operate under safer conditions until corrective maintenance is performed on defective actuators.

In view of the above, this work aims to present an extension of the virtual actuator based on a moving horizon approach, presented in recent works (COSTA et al., 2013; COSTA, 2014; COSTA et al., 2015). Thus, the present proposal contemplates the recalculation of the references to be followed by the controller and considers restrictions in the deviation between nominal and faulty plant behaviors. For this, a control reconfiguration block is proposed whose structure is composed of three items: an observer of states and disturbances, an optimization layer responsible for the recalculation of references and the virtual actuator based on a moving horizon optimization. This approach is similar to the two-layer predictive controller (MPC) in which an external (static layer) optimization is used in conjunction with the MPC (dynamic layer) (YING; VOORAKARANAM; JOSEPH, 1999). The use of a static layer within the virtual actuator was first proposed in (COSTA et al., 2015).

In addition to calculating new reachable setpoints in a fault scenario, the proposed technique also contemplates situations in which the post-fault model is not perfect, which is a case that has received little attention in the literature, as pointed out by (RICHTER; LUNZE; SCHLAGE, 2007). Thus, differences between the model provided by the FDD system and the true condition of the faulty plant are considered as disturbances, whose estimates are provided by an observer based on a disturbance model.

Finally, for the purpose of testing the proposed technique, simulations were carried out using a quadruple-tank process subjected to actuator faults. In addition, experiments were performed on a experimental pH neutralization plant.

CHAPTER 2

Theory and literature review

In this chapter we introduce the concepts that will serve as the basis for the proposed reconfiguration block described in Chapter 3. First, the formulation of MPC is discussed, exposing the development of equations with focus on a control loop structure composed of a state estimator, an external optimization (static layer) and the moving horizon-based controller (dynamic layer). Later, a brief theoretical foundation on reconfigurable fault tolerant control is presented, in which we describe some recently proposed methods that served as a starting point for this work.

2.1 Model predictive control

2.1.1 Introduction

Model Predictive Control is a term used to designate a class of control algorithms that make use of the process model and perform the optimization of a linear or quadratic objective function subjected to constraints along a prediction horizon of the open loop plant response. Thus, the future trajectory of the manipulated variable is obtained, but only the first control action is implemented in the plant. After updating the measured variables, the trajectory of the control variable is recomputed and the procedure is repeated at each sampling interval (MUSKE; RAWLINGS, 1993). This strategy is called receding horizon or moving horizon control (Figure 1).

MPC is a well-established control strategy in both industry and academia and its early developments and industrial applications date back to the 1970s. Among the first implementations are the Model Predictive Heuristic Control (MPHC) developed by Richalet et al. (1978) and the Dynamic Matrix Control (DMC) developed by Shell engineers Cutler and Ramaker (1980). An extension of the DMC that comprises a quadratic objective

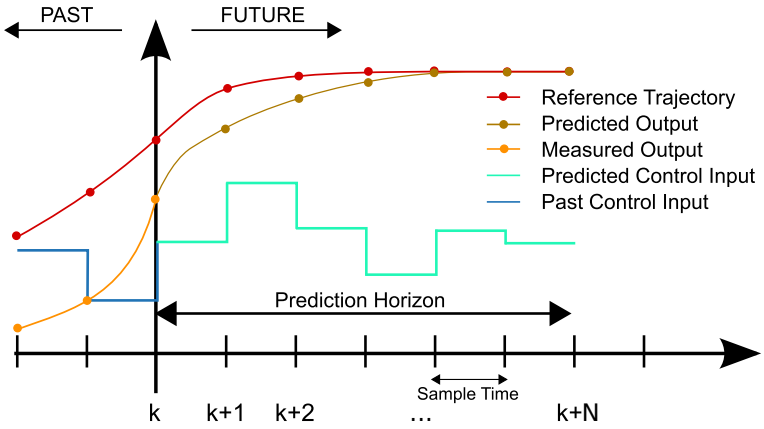

Figure 1 – The moving horizon scheme.

Source: Adapted from the original by Martin Behrendt, via Wikimedia Commons and licensed under the terms of the GFDL (<http://www.gnu.org/copyleft/fdl.html>).

function subjected to constraints was later proposed by Garcia and Morshedi (1986). Qin and Badgwell (2003) presented in detail the historical development of MPC as well as approaches to different products and industrial solutions that use this class of algorithms. In addition, extensive material can be found in the literature, from tutorials (RAWLINGS, 1999; RAWLINGS, 2000; WANG, 2004), to textbooks (CAMACHO; BORDONS, 2007; RAWLINGS; MAYNE, 2009; WANG, 2009; MACIEJOWSKI, 2002).

Among the reasons for the success of MPC in applications in the process industry is the optimization of an objective function that easily incorporates constraints such as limits of actuators and product specification. In this way, the process can operate safely close to the constraints, reaching more interesting operating conditions from the economic point of view. Another great advantage is the possibility of applying MPC in multivariable processes by superposition of linear models that are usually based on transfer functions, space-state models or even convolution models. Normally, these models do not require advanced methods of system identification and can be obtained by analyzing the data collected in tests carried out in the plant (MUSKE; RAWLINGS, 1993).

2.1.2 The MPC formulation

For discrete time-invariant systems, the state space linear model that is assumed to describe the system dynamics is given according to 2.1, in which $x \in \mathbb{R}^{n_x}$ is the vector of states, $u \in \mathbb{R}^{n_u}$ is the vector of inputs and $y \in \mathbb{R}^{n_y}$ is the vector of plant outputs. The system matrices are $A \in \mathbb{R}^{n_x \times n_x}$, $B \in \mathbb{R}^{n_x \times n_u}$ and $C \in \mathbb{R}^{n_y \times n_x}$.

$$x(k+1) = Ax(k) + Bu(k) \tag{2.1a}$$

$$y(k) = Cx(k) \tag{2.1b}$$

Usually, the conventional form of the model predictive controller are given by the following optimization problem:

$$\min_{\Delta U(k)} \quad V \coloneqq \sum_{j=k}^{N} \Big(y(j) - y_{sp}(k)\Big)^T Q_y \Big(y(j) - y_{sp}(k)\Big) + \sum_{j=k}^{N_c-1} \Delta u(j)^T R \Delta u(j) \tag{2.2}$$

subjected to

$$x(j+1) = Ax(j) + Bu(j) \qquad j = k, \ldots, N_c - 1 \tag{2.3}$$

$$u_{min} \leq u(j) \leq u_{max} \qquad j = k, \ldots, N_c - 1 \tag{2.4}$$

$$\Delta u_{min} \leq \Delta u(j) \leq \Delta u_{max} \qquad j = k, \ldots, N_c - 1 \tag{2.5}$$

in which y_{sp} is the current output setpoint, $\Delta u(k) = u(k) - u(k-1)$ is the input change, $\Delta U(k)$ is the vector of future input changes given in 2.6, Q_y and R are weighting matrices assumed to be positive definite, N is the prediction horizon and N_c is the control horizon.

$$\Delta U^T(k) = \Big[\Delta u^T(k) \quad \Delta u^T(k+1) \quad \cdots \quad \Delta u^T(k+N_c-1)\Big]^T \tag{2.6}$$

However, as pointed out by Bitmead, Gevers and Wertz (1990), the controller based on the objective function presented in 2.2 has no guaranteed stability since the prediction of the process response is performed over a finite horizon. In general, there are several methods for obtaining a stable MPC and one of the most popular is the inclusion of a state terminal constraint (MEADOWS et al., 1995). Another approach was proposed in (RAWLINGS; MUSKE, 1993), in which the authors present a controller with infinite horizon and whose stability is guaranteed and independent of the controller tuning parameters. The infinite horizon control problem is defined as follows

$$\min_{x(j),u(j)} \quad V \coloneqq \sum_{j=k}^{\infty} x^T(j) Q x^T(j) + \sum_{j=k}^{\infty} u^T(j) R u(j) \tag{2.7}$$

subjected to

$$x(j+1) = Ax(j) + Bu(j) \tag{2.8}$$

$$u_{min} \leq u(j) \leq u_{max} \tag{2.9}$$

$$-\Delta u_{max} \leq \Delta u(j) \leq \Delta u_{max} \tag{2.10}$$

The solution of this convex optimization problem is intractable since it has an infinite number of decision variables. However, the control objective 2.7 can be written as a finite horizon problem as follows

$$V \coloneqq \sum_{j=k}^{N-1} x^T(j) Q x^T(j) + \sum_{j=k}^{N-1} u^T(j) R u(j) + x^T(k+N) P x(k+N) \tag{2.11}$$

in which $x^T(k+N)Px(k+N)$ denotes the terminal cost and P is a positive semi-definite matrix computed by the Lyapunov equation according to 2.12 for stable systems.

$$A^T P A - P = -Q \tag{2.12}$$

2.1.3 The two-layer MPC

In modern industrial plants, MPC is usually part of a multi-layer hierarchical structure, as shown in Figure 2. The real time optimization (RTO) layer is characterized by an economical optimizer based on rigorous steady state process models. Usually, this optimizer determines new desired operational points in periods of the order of hours. However, these operational points may become unreachable when disturbances are affecting the system. Therefore, in some applications there are local optimizers that solves linear or quadratic optimization problems at the same frequency as the MPC layer. These local optimizers are based on steady state linear models and are responsible for calculating targets to be tracked by the MPC layer (MACIEJOWSKI, 2002).

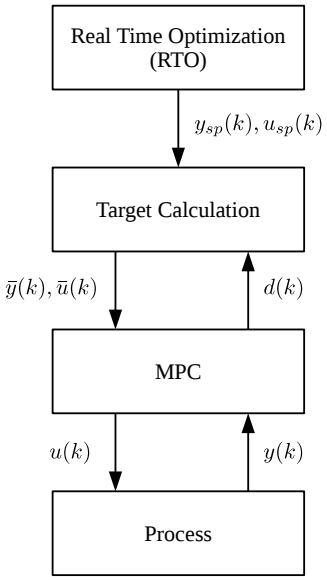

Figure 2 – Hierarchical control structure.

Before formulating the two-layer MPC we shall address a method for adding integral control and also discuss the estimation of states and disturbances.

Disturbance model

The steady state mismatch between the linear model and the true plant can be captured by augmenting the plant model with a disturbance model, which adds integrating modes to the system. The resulting model is given by 2.13 in which $B_d \in \mathbb{R}^{n_x \times n_d}$, $C_d \in \mathbb{R}^{n_y \times n_d}$, $I \in \mathbb{R}^{n_d \times n_d}$ is the identity matrix and n_d is the number of additional disturbances.

$$x(k+1) = Ax(k) + Bu(k) + B_d d(k) \tag{2.13a}$$

$$d(k+1) = d(k) \tag{2.13b}$$

$$y(k) = Cx(k) + C_d d(k) \tag{2.13c}$$

This model can be rewritten as follows

$$\begin{bmatrix} x(k+1) \\ d(k+1) \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} x(k) \\ d(k) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(k) \tag{2.14a}$$

$$y(k) = \begin{bmatrix} C & C_d \end{bmatrix} \begin{bmatrix} x(k) \\ d(k) \end{bmatrix} \tag{2.14b}$$

Muske and Badgwell (2002) and Pannocchia and Rawlings (2003) have shown conditions for the model to be detectable, which is an indispensable requirement for the existence of an asymptotically stable observer. Such condition is related to the choice of matrices B_d and C_d according to the following *lemma*.

Lemma 1. *The augmented system 2.14 is detectable if and only if the pair* (C, A) *is detectable, that is, if the matrix* $\begin{bmatrix} I - A \\ C \end{bmatrix} \in \mathbb{R}^{(n_y+n_x)\times n_x}$ *has rank* n_x*, and if 2.15 holds true.*

$$rank \begin{bmatrix} I - A & -B_d \\ C & C_d \end{bmatrix} = n_x + n_d \tag{2.15}$$

In addition, the number of additional disturbances in the augmented system must not exceed the number of system outputs, that is, $n_d \leq n_y$ (see Pannocchia and Rawlings (2003), Corollary 1).

It should be noted that the augmented system 2.14 is not stabilizable since the addition of disturbances introduces unstable modes to the system. Thus, since these modes can not be controlled, the purpose of adding disturbances is removing their influences from the controlled variables (MUSKE; BADGWELL, 2002; PANNOCCHIA; RAWLINGS, 2003).

State estimation

In real applications the states may not be available through plant measurements so they must be estimated. A simple approach is to use a steady state Kalman filter to perform state and disturbance estimation. Here, we shall rewrite the augmented system given in 2.14 according to 2.16, in which $w(k) \in \mathbb{R}^{(n_x+n_d)}$ and $\xi(k) \in \mathbb{R}^{n_y}$ are state and output noises, respectively.

$$\begin{bmatrix} x(k+1) \\ d(k+1) \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} x(k) \\ d(k) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(k) + w(k) \tag{2.16a}$$

$$y(k) = \begin{bmatrix} C & C_d \end{bmatrix} \begin{bmatrix} x(k) \\ d(k) \end{bmatrix} + \xi(k) \tag{2.16b}$$

Also, these noises are assumed to be white, zero-mean, uncorrelated and have known constant covariance matrices Θ_w and Θ_ξ as follows:

$$\begin{aligned} w(k) &\sim (0, \Theta_w(k)) \\ \xi(k) &\sim (0, \Theta_\xi(k)) \\ E\left[w(k)w(i)^T\right] &= \Theta_w \delta(k-i) \\ E\left[\xi(k)\xi(i)^T\right] &= \Theta_\xi \delta(k-i) \\ E\left[\xi(k)w(i)^T\right] &= 0 \end{aligned}$$

in which $\delta(k-i)$ is the Kronecker delta function, with $\delta(k-i) = 1$ if $k = i$ and $\delta(k-i) = 0$ if $k \neq i$.

First, using past information about the states and disturbances, we can predict them at current instant k according to 2.17. The notation $\hat{\psi}(i \,|\, j)$ denotes the estimate of the variable ψ at instant i using information available up to instant j.

$$\begin{bmatrix} \hat{x}(k \,|\, k-1) \\ \hat{d}(k \,|\, k-1) \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{x}(k-1 \,|\, k-1) \\ \hat{d}(k-1 \,|\, k-1) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(k-1) \tag{2.17}$$

Then the estimates can be updated by taking into account the plant measurements $y(k)$ at the current sampling time according to 2.18,

$$\begin{bmatrix} \hat{x}(k \,|\, k) \\ \hat{d}(k \,|\, k) \end{bmatrix} = \begin{bmatrix} \hat{x}(k \,|\, k-1) \\ \hat{d}(k \,|\, k-1) \end{bmatrix} + \begin{bmatrix} L_x \\ L_d \end{bmatrix} \left(y(k) - C\hat{x}(k \,|\, k-1) - C_d \hat{d}(k \,|\, k-1) \right) \tag{2.18}$$

in which $L_x \in \mathbb{R}^{n_x \times n_y}$ and $L_d \in \mathbb{R}^{n_d \times n_y}$ are filter gains related to the states and disturbances, respectively, and are computed according to 2.19 (SIMON, 2006).

$$\tilde{L} = \tilde{A}\Lambda\tilde{C}^T \left(\tilde{C}\Lambda\tilde{C}^T + \Theta_\xi \right) \tag{2.19}$$

in which

$$\tilde{L} = \begin{bmatrix} L_x \\ L_d \end{bmatrix}, \quad \tilde{A} = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix}, \quad \tilde{C} = \begin{bmatrix} C & C_d \end{bmatrix}$$

and Λ is given as the steady state solution of the discrete-time algebraic Riccati equation (DARE) provided that the augmented system is stabilizable (WANG, 2009).

$$\Lambda = \tilde{A}\Lambda\tilde{A}^T + \Theta_w - \tilde{A}\Lambda\tilde{C}^T \left(\tilde{A}\Lambda\tilde{A} + \Theta_\xi \right)^{-1} \tilde{C}\Lambda\tilde{A}^T \tag{2.20}$$

Targets calculation

Given the disturbances estimates, the target calculation is performed to obtain updated values of $\bar{x}$ and $\bar{u}$. This is done by solving a quadratic programming problem with exact soft constraints, which are formulated by adding a penalty l_1/l_2^2 to the objective function. In this way, the solution of the problem reaches the desired values x_{sp} and u_{sp} if

possible, but it is relaxed if the problem is unfeasible. Details and comments regarding the use of exact penalty can be found in Rawlings (1999).

The target calculation problem is given from 2.21 to 2.25, in which it is assumed that the weighting matrices Q_t and R_t are symmetric positive-definite with appropriate dimensions and $\eta \in \mathbb{R}^{n_y}$ is the vector of slack variables.

$$\min_{\bar{x}(k),\bar{u}(k),\eta(k)} \quad V_t \coloneqq \eta^T(k)Q_t\eta(k) + \Big(\bar{u}(k) - u_{sp}(k)\Big)^T R_t\Big(\bar{u}(k) - u_{sp}(k)\Big) + q_t^T\eta(k) \tag{2.21}$$

subjected to

$$\begin{bmatrix} I-A & -B & 0 \\ C & 0 & I \\ C & 0 & -I \end{bmatrix} \begin{bmatrix} \bar{x}(k) \\ \bar{u}(k) \\ \eta(k) \end{bmatrix} \begin{Bmatrix} = \\ \geq \\ \leq \end{Bmatrix} \begin{bmatrix} B_d\hat{d}(k) \\ y_{sp}(k) - C_d\hat{d}(k) \\ y_{sp}(k) - C_d\hat{d}(k) \end{bmatrix} \tag{2.22}$$

$$u_{min} \leq \bar{u}(k) \leq u_{max} \tag{2.23}$$

$$y_{min} \leq C\bar{x}(k) + C_d\hat{d}(k) \leq y_{max} \tag{2.24}$$

$$\eta(k) \geq 0 \tag{2.25}$$

Regulator

After the target calculation, the control actions are obtained as solution of an optimization problem, in which the quadratic objective function of the infinite horizon controller is given by Equation 2.26, where Q, R and S are weighting matrices (MUSKE; RAWLINGS, 1993).

$$\begin{aligned} V \coloneqq &\sum_{j=k}^{\infty}\Big(x(j) - \bar{x}(k)\Big)^T Q\Big(x(j) - \bar{x}(k)\Big) + \sum_{j=k}^{\infty}\Big(u(j) - \bar{u}(k)\Big)^T R\Big(u(j) - \bar{u}(k)\Big) \\ &+ \sum_{j=k}^{\infty}\Delta u(j)^T S\Delta u(j) \end{aligned} \tag{2.26}$$

We can express this objective function considering a finite horizon N so the controller is given as solution of the following optimization problem.

$$\begin{aligned} \min_{U(k)} \quad V \coloneqq &\sum_{j=k}^{N-1}\Big(x(j) - \bar{x}(k)\Big)^T Q\Big(x(j) - \bar{x}(k)\Big) + \sum_{j=k}^{N-1}\Big(u(j) - \bar{u}(k)\Big)^T R\Big(u(j) - \bar{u}(k)\Big) \\ &+ \sum_{j=k}^{N-1}\Delta u(j)^T S\Delta u(j) + \Big(x(k+N) - \bar{x}(k)\Big)^T P\Big(x(k+N) - \bar{x}(k)\Big) \end{aligned} \tag{2.27}$$

subjected to

$$\tilde{x}(j+1) = A\tilde{x}(j) + B\tilde{u}(j) \tag{2.28}$$

$$u_{min} - \bar{u}(k) \leq \tilde{u}(j) \leq u_{max} - \bar{u}(k) \tag{2.29}$$

$$\Delta u_{min} \leq \Delta u(j) \leq \Delta u_{max} \tag{2.30}$$

$$\tilde{x}(j) = \hat{x}(j) - \bar{x}(k), \quad \tilde{u}(j) = u(j) - \bar{u}(k) \tag{2.31}$$

in which $U(k)$ is the vector of future control actions 2.32 and P is the terminal penalty matrix and can be computed with the Lyapunov equation 2.12 for open-loop stable systems.

$$U(k) = \begin{bmatrix} u^T(k) & u^T(k+1) & \cdots & u^T(k+N-1) \end{bmatrix}^T \tag{2.32}$$

Finally, combining the states and disturbances estimation, the target calculation and the regulator (Figure 3), the MPC algorithm can be summarized in the following steps:

1. Obtain states and disturbances estimates $(\hat{x}(k),\ \hat{d}(k))$ trough plant measurements $y(k)$;
2. Solve the static optimization layer in order to determine the steady state targets $(\bar{y}(k),\ \bar{x}(k),\ \bar{u}(k))$;
3. Solve the dynamic layer problem to obtain $\tilde{u}(k)$;
4. Make $u(k) = \tilde{u}(k) + \bar{u}(k)$ and apply to the plant;
5. Repeat this procedure at the next sampling time.

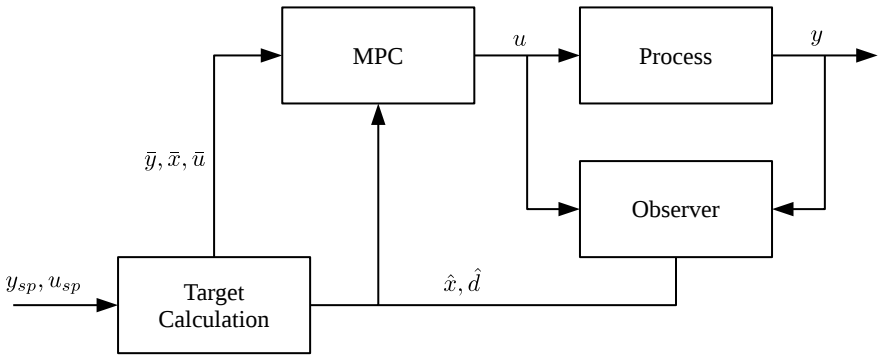

Figure 3 – The MPC control structure.

2.2 Reconfigurable control

An extensive bibliographical review on reconfigurable control is presented in (ZHANG; JIANG, 2008). We will present here only the concepts that served as the basis for the proposal described in Chapter 3.

2.2.1 The fault hiding and virtual actuator approaches

Actuator faults may affect the process controllability, which can lead to degraded performance and even undesirable shutdowns. As already mentioned, there are several

fault-tolerant control techniques, which are mostly based on the removal of the nominal controller from the reconfigured control loop. An exception is the virtual actuator-based approach, being less intrusive since it maintains all the knowledge about the plant that was incorporated in the nominal controller design (RICHTER; LUNZE; SCHLAGE, 2007; RICHTER, 2011).

This approach consists of a block that is placed between the nominal controller and process actuators in order to modify the interface signals, such as the process output y_p and the control action u_m (see Figure 4). Thus, the output signal of the nominal controller is changed so as to cause the same control effort that the faulty actuator had in the nominal plant before the fault occurrence (STEFFEN, 2005; LUNZE; STEFFEN, 2006). In addition, the process performance degradation is hidden from the point of view of the nominal controller by reconstructing the normal behavior of the plant y_m. Next, the development of the strategy is exposed.

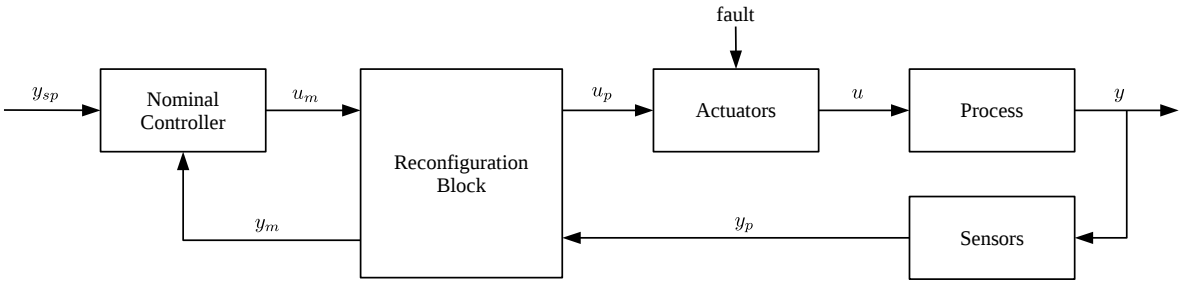

Figure 4 – Reconfiguration block placed between the nominal controller and the faulty plant.

The discrete time state-space model that describes the nominal plant is given in 2.33, in which $x_m \in \mathbb{R}^{n_x}$, $u_m \in \mathbb{R}^{n_u}$ and $y_m \in \mathbb{R}^{n_y}$ are vectors of states, inputs and outputs, respectively. The system matrices are $A \in \mathbb{R}^{n_x \times n_x}$, $B_m \in \mathbb{R}^{n_x \times n_u}$ e $C \in \mathbb{R}^{n_y \times n_x}$.

$$x_m(k+1) = Ax_m(k) + B_m u_m(k) \tag{2.33a}$$

$$y_m(k) = Cx_m(k) \tag{2.33b}$$

Similarly, the faulty plant can be represented by the post-fault model given in 2.34, in which $u_p \in \mathbb{R}^{n_u}$. In cases of actuator faults, the input matrix of the post-fault model differs from its counterpart in the nominal model, so that $B_p \neq B_m$. For instance, the actuator locked at the nominal position can be represented by a null column in the input matrix (RICHTER; LUNZE; SCHLAGE, 2007).

$$x_p(k+1) = Ax_p(k) + B_p u_p(k) \tag{2.34a}$$

$$y_p(k) = Cx_p(k) \tag{2.34b}$$

We can now build the difference model (STEFFEN, 2005) that describes the deviation between nominal and faulty plant behaviors. This model is given in 2.35, in which

$x_\Delta(k) = x_m(k) - x_p(k)$.

$$x_\Delta(k+1) = Ax_\Delta(k) + B_m u_m(k) - B_p u_p(k) \tag{2.35a}$$

$$y_\Delta(k) = Cx_\Delta(k) \tag{2.35b}$$

After the fault occurrence, the post-fault model is provided by the FDD system and the nominal plant output signal can be reconstructed according to 2.36.

$$y_m(k) = y_p(k) + Cx_\Delta(k) \tag{2.36}$$

As proposed in (STEFFEN, 2005; LUNZE; STEFFEN, 2006; RICHTER; LUNZE; SCHLAGE, 2007), the control signal is modified according to 2.37, in which M and N are feedback and feedforward matrices, respectively. Details regarding the computation of these matrices can be found in (STEFFEN, 2005; RICHTER, 2011).

$$u_p(k) = Mx_\Delta(k) + Nu_m(k) \tag{2.37}$$

2.2.2 The moving horizon-based virtual actuator

A moving horizon-based strategy as an extension of the virtual actuator approach was first presented by (COSTA et al., 2013) and detailed in his PhD thesis (COSTA, 2014). This proposal is based on the solution of a quadratic programming problem to obtain control actions in order to stabilize the deviation between faulty and nominal plant behaviors. Also, this technique allows the insertion of process constraints in the optimization problem, which allows to consider physical limits of actuators and process variables.

Problem formulation

The formulation of the moving horizon-based virtual actuator is similar to the MPC algorithm that was presented in Section 2.1.

Using the difference model 2.35 and applying it along a prediction horizon N_v, we obtain the Equation 2.38

$$X_\Delta(k+1) = Fx_\Delta(k) + \Phi_m U_m(k) - \Phi_p U_p(k) \tag{2.38}$$

with

$$X_\Delta(k+1) = \begin{bmatrix} x_\Delta(k+1) \\ x_\Delta(k+2) \\ \vdots \\ x_\Delta(k+N_v) \end{bmatrix}, \quad F = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{N_v} \end{bmatrix},$$

$$\Phi_m = \begin{bmatrix} B_m & 0 & \dots & 0 \\ AB_m & B_m & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N_v-1}B_m & A^{N_v-2}B_m & \dots & B_m \end{bmatrix}, \quad \Phi_p = \begin{bmatrix} B_p & 0 & \dots & 0 \\ AB_p & B_p & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N_v-1}B_p & A^{N_v-2}B_p & \dots & B_p \end{bmatrix},$$

$$U_m(k) = \begin{bmatrix} u_m(k) \\ u_m(k+1) \\ \vdots \\ u_m(k+N_v-1) \end{bmatrix}, \quad U_p(k) = \begin{bmatrix} u_p(k) \\ u_p(k+1) \\ \vdots \\ u_p(k+N_v-1) \end{bmatrix}.$$

The trajectory U_p of future control actions is obtained by solving an optimization problem as given from 2.39 to 2.41, in which $\bar{Q}_v$ and $\bar{R}_v$ are diagonal block matrices formed by the penalty matrices Q_v and R_v, respectively.

$$\min_{U_p} \quad V \coloneqq X_\Delta^T \bar{Q}_v X_\Delta + U_p^T \bar{R}_v U_p \tag{2.39}$$

subjected to

$$X_\Delta(k+1) = Fx_\Delta(k) + \Phi_m U_m(k) - \Phi_p U_p(k) \tag{2.40}$$

$$\Delta U \in \mathcal{U} \tag{2.41}$$

It is worth mentioning that this technique uses the nominal control trajectory, which is easily obtained when there is a well defined control law to be applied to the plant model. However, Costa et al. (2013) also discussed the possibility of keeping constant the nominal controller output in U_m. This is necessary when the control law equation is not available or is too complex. It is understood that the maintenance of the future control signal as constant represents an additive disturbance along the prediction horizon. However, simulation results presented in Costa (2014) have demonstrated that this simplification does not compromise process performance and fault accommodation.

An extension to the moving horizon approach for virtual actuators was presented in COSTA et al. (2015), whose proposal is based on the use of an external optimization layer responsible for calculating the steady-state deviation between the nominal and faulty plant behaviors. In addition, the calculation of the control trajectory U_p is modified in order to incorporate the tracking of steady state references $\bar{x}_\Delta$ and $\bar{u}_p$, computed according to the following optimization problem.

$$\min_{\bar{x}_p, \bar{u}_p} \quad V_t \coloneqq \left(\bar{x}_m - \bar{x}_p\right)^T Q_s \left(\bar{x}_m - \bar{x}_p\right) \tag{2.42}$$

subjected to

$$\left(I - A\right) \bar{x}_p - B_p \bar{u}_p = 0 \tag{2.43}$$

$$\bar{u}_p \in \mathcal{U} \tag{2.44}$$

As pointed out in COSTA et al. (2015), if the process redundancies perform enough control effort to maintain the null steady-state deviation, complete fault accommodation can be achieved. However, more severe fault scenarios may lead to a non-zero steady-state deviation in some variables.

CHAPTER 3

The two-layer moving horizon virtual actuator

3.1 The proposed reconfiguration block

As already mentioned, Costa et al. (2013) proposed a moving horizon-based virtual actuator. Similar to the MPC algorithm, this approach aims to minimize the deviation x_Δ between the nominal and faulty plant behavior over a prediction horizon. This is accomplished by solving a quadratic programming problem (QP), in which the constraints of the nominal plant, as actuator limits and product specification, can be taken into account.

However, in cases of actuator loss with no available redundancies, the controllability of the plant is affected and it may become impossible to drive x_Δ to the origin, since there are not enough degrees of freedom in the system. In this sense, since we have the post-fault model, it is possible to calculate achievable references to be traced by the optimization problem in a manner similar to that presented in Section 2.1. On the other hand, if the post-fault model is not perfect due to misidentified faults, the calculation of references can result in erroneous values in the steady state that may lead to offset. This can also occur when using a linear model-based approach in nonlinear processes, which is often the case of real applications.

Thus, in order to deal with the aforementioned situations, the proposed reconfiguration block makes use of a structure typical of many MPC applications (see Section 2.1). This approach consists of two layers: an optimization layer responsible for computing steady state targets for the faulty plant and a second one that computes control actions in order to track the targets. The way these steps are connected is shown in Figure 5, in which the reconfiguration block is inside the red dashed rectangle.

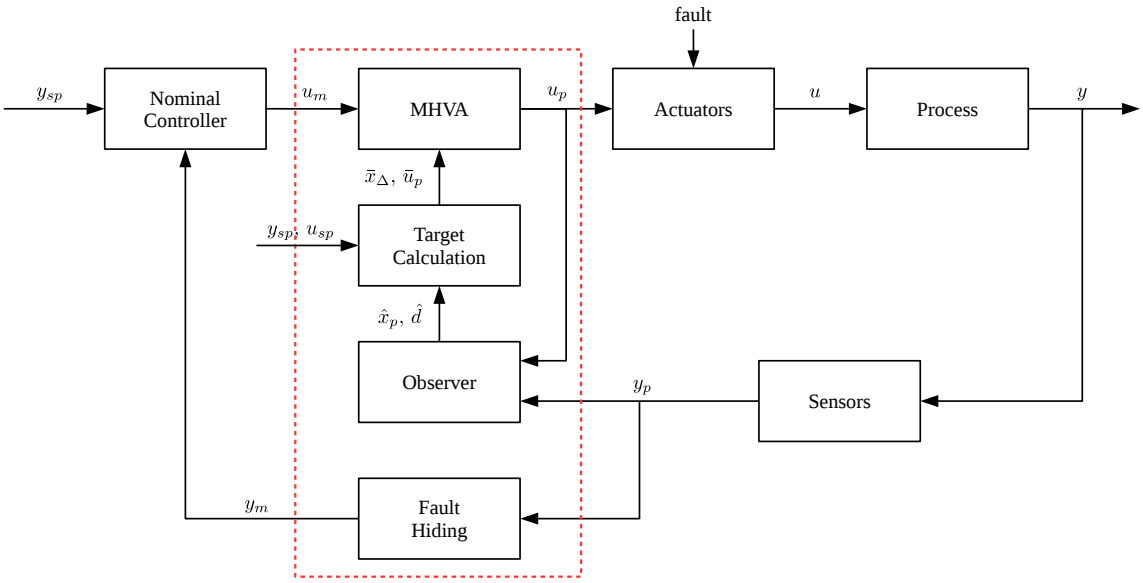

Figure 5 – The proposed reconfiguration block structure.

3.1.1 Target calculation

From the difference model described in 2.35, Equation 3.1 must hold in the steady state supposing the system is undisturbed.

$$\bar{x}_\Delta(k+1) = A\bar{x}_\Delta(k) + B_m u_m(k) - B_p\bar{u}_p(k) \tag{3.1a}$$

$$\bar{y}_\Delta(k) = C\bar{x}_\Delta(k) \tag{3.1b}$$

in which $\bar{x}_\Delta \in \mathbb{R}^{n_x}$, $\bar{u}_p \in \mathbb{R}^{n_u}$ and $\bar{y}_\Delta \in \mathbb{R}^{n_y}$ represent steady states, inputs and outputs of the faulty plant, respectively.

When the process is subjected to faults, the origin of the system may not be reachable, causing the controlled variables to present steady state offset. Then, new steady states can be computed by solving the following optimization problem.

$$\min_{\bar{x}_\Delta(k),\bar{u}_p(k),\eta(k)} \quad V_t \coloneqq \bar{y}_\Delta^T(k)Q_t\bar{y}_\Delta(k) + \left(\bar{u}_p(k) - u_{sp}(k)\right)^T R_t \left(\bar{u}_p(k) - u_{sp}(k)\right) \tag{3.2}$$

$$+ \eta^T(k)S_t\eta(k) \tag{3.3}$$

subjected to:

$$\left(I - A\right)\bar{x}_\Delta(k) + B_p\bar{u}_p(k) = B_m u_m(k) \tag{3.4}$$

$$\eta(k) + \bar{y}_{\Delta min} \leq C\bar{x}_\Delta(k) \leq \bar{y}_{\Delta max} + \eta(k) \tag{3.5}$$

$$y_{min} \leq C\bar{x}_\Delta(k) \leq y_{max} \tag{3.6}$$

$$u_{min} \leq \bar{u}_p(k) \leq u_{max} \tag{3.7}$$

However, when the model used by the reconfiguration block does not represent the true faulty plant, the targets can be sub-optimal and even unreachable. In fact, this is

known to happen in real control applications in which the true plant is often nonlinear. Besides plant nonlinearities, the post-fault model can also be wrong if the fault magnitude is misidentified by the FDD system.

In this sense, we can remove the steady state offset by including disturbances estimates in the targets calculation. To do so, we build a disturbance model by augmenting the post-fault model with disturbances. From the post-fault model 2.34, we add terms that represent disturbances affecting the system or the difference between model and plant due to model mismatch, obtaining the model shown in 3.8, in which $B_d \in \mathbb{R}^{n_x \times n_d}$, $C_d \in \mathbb{R}^{n_y \times n_d}$ and n_d is the number of disturbances.

$$x_p(k+1) = Ax_p(k) + B_p u_p(k) + B_s u_s(k) + B_d d(k) \tag{3.8a}$$

$$y_p(k) = Cx_p(k) + C_d d(k) \tag{3.8b}$$

Then, using the nominal plant model, we can obtain the new difference model making $x_\Delta(k) = x_m(k) - x_p(k)$:

$$x_\Delta(k+1) = Ax_\Delta(k) + B_m u_m(k) - B_p u_p(k) - B_s u_s(k) - B_d d(k) \tag{3.9a}$$

$$y_\Delta(k) = Cx_\Delta(k) - C_d d(k) \tag{3.9b}$$

Therefore, the targets calculation can be reformulated as in the following optimization problem.

$$\min_{\bar{x}_\Delta(k), \bar{u}_p(k), \eta(k)} \quad V_t \coloneqq \bar{y}_\Delta^T(k) Q \bar{y}_\Delta(k) + \left(\bar{u}_p(k) - u_{sp}(k)\right)^T R \left(\bar{u}_p(k) - u_{sp}(k)\right) \tag{3.10}$$

$$+ \eta^T(k) S \eta(k) \tag{3.11}$$

subjected to:

$$(I - A)\,\bar{x}_\Delta(k) + B_p \bar{u}_p(k) = B_m u_m(k) - B_s u_s(k) - B_d d(k) \tag{3.12}$$

$$\eta(k) + \bar{y}_{\Delta min} + C_d d(k) \leq C\bar{x}_\Delta(k) \leq \bar{y}_{\Delta max} + C_d d(k) + \eta(k) \tag{3.13}$$

$$y_{min} \leq C\bar{x}_\Delta(k) \leq y_{max} \tag{3.14}$$

$$u_{min} \leq \bar{u}_p(k) \leq u_{max} \tag{3.15}$$

3.1.2 Moving Horizon Virtual Actuator

Thus, we formulate the moving horizon-based problem responsible for tracking the targets $\bar{x}_\Delta$, $\bar{u}_p$ and $\bar{y}_\Delta$. So, the objective function can be written according to 3.16.

$$\begin{aligned} V_v \coloneqq & \sum_{j=k}^{\infty} \Big(x_\Delta(j) - \bar{x}_\Delta(k)\Big)^T Q_v \Big(x_\Delta(j) - \bar{x}_\Delta(k)\Big) \\ & + \sum_{j=k}^{\infty} \Big(u_p(j) - \bar{u}_p(k)\Big)^T R_v \Big(u_p(j) - \bar{u}_p(k)\Big) \\ & + \sum_{j=k}^{\infty} \Delta u_p^T(j) S_v \Delta u_p(j) \end{aligned} \tag{3.16}$$

Problem solution

By applying the Equation 3.9 along a prediction horizon N_v, we obtain 3.17.

$$X_\Delta(k+1) = \Phi x_\Delta(k) + \Gamma_m U_m(k) + \Gamma_p U_p(k) + \Gamma_s U_s(k) + \Gamma_d U_d(k) \tag{3.17}$$

with

$$X_\Delta(k+1) = \begin{bmatrix} x_\Delta(k+1) \\ x_\Delta(k+2) \\ \vdots \\ x_\Delta(k+N_v) \end{bmatrix}, \quad \Phi = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{N_v} \end{bmatrix}, \quad U_m(k) = \begin{bmatrix} u_m(k) \\ u_m(k) \\ \vdots \\ u_m(k) \end{bmatrix},$$

$$U_p(k) = \begin{bmatrix} u_p(k+1) \\ u_p(k+2) \\ \vdots \\ u_p(k+N_v-1) \end{bmatrix}, \quad U_s(k) = \begin{bmatrix} u_s(k) \\ u_s(k) \\ \vdots \\ u_s(k) \end{bmatrix}, \quad U_d(k) = \begin{bmatrix} d(k) \\ d(k) \\ \vdots \\ d(k) \end{bmatrix},$$

$$\Gamma_m = \begin{bmatrix} B_m & 0 & \dots & 0 \\ AB_m & B_m & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N_v-1}B_m & A^{N_v-2}B_m & \dots & B_m \end{bmatrix}, \quad \Gamma_p = \begin{bmatrix} B_p & 0 & \cdots & 0 \\ AB_p & B_p & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N_v-1}B_p & A^{N_v-2}B_p & \cdots & B_p \end{bmatrix},$$

$$\Gamma_s = \begin{bmatrix} B_s & 0 & \cdots & 0 \\ AB_s & B_s & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N_v-1}B_s & A^{N_v-2}B_s & \cdots & B_s \end{bmatrix}, \quad \Gamma_d = \begin{bmatrix} B_d & 0 & \cdots & 0 \\ AB_d & B_d & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N_v-1}B_d & A^{N_v-2}B_d & \cdots & B_d \end{bmatrix}.$$

Since the targets are considered constant along the prediction horizon, we can write 3.17 using steady state values according to 3.18.

$$\bar{X}_\Delta(k) = \Phi \bar{x}_\Delta(k) + \Gamma_m U_m(k) + \Gamma_p \bar{U}_p(k) + \Gamma_s U_s(k) + \Gamma_d U_d(k) \tag{3.18}$$

with

$$\bar{X}_\Delta(k) = \begin{bmatrix} \bar{x}_\Delta(k) \\ \bar{x}_\Delta(k) \\ \vdots \\ \bar{x}_\Delta(k) \end{bmatrix}, \quad \bar{U}_p(k) = \begin{bmatrix} \bar{u}_p(k) \\ \bar{u}_p(k) \\ \vdots \\ \bar{u}_p(k) \end{bmatrix}.$$

Then we can simplify the problem by changing the system origin so we define $\tilde{x}_\Delta = x_\Delta - \bar{x}_\Delta$ and $\tilde{u}_p = u_p - \bar{u}_p$ and subtract 3.18 from 3.17, obtaining 3.19.

$$\tilde{X}_\Delta(k+1) = \Phi \tilde{x}_\Delta(k) + \Gamma_p \tilde{U}_p(k) \tag{3.19}$$

So the value of the control objective is given as follows

$$\begin{aligned} V_v(k) = \tilde{x}_\Delta^T(k)Q\tilde{x}_\Delta(k) + &\begin{bmatrix} \tilde{x}_\Delta(k+1) \\ \tilde{x}_\Delta(k+2) \\ \vdots \\ \tilde{x}_\Delta(k+N_v) \end{bmatrix}^T \begin{bmatrix} Q & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & Q & 0 \\ 0 & \cdots & 0 & P \end{bmatrix} \begin{bmatrix} \tilde{x}_\Delta(k+1) \\ \tilde{x}_\Delta(k+2) \\ \vdots \\ \tilde{x}_\Delta(k+N_v) \end{bmatrix} \\ + &\begin{bmatrix} \tilde{u}_p(k) \\ \tilde{u}_p(k+1) \\ \vdots \\ \tilde{u}_p(k+N_v-1) \end{bmatrix}^T \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & R & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & R \end{bmatrix} \begin{bmatrix} \tilde{u}_p(k) \\ \tilde{u}_p(k+1) \\ \vdots \\ \tilde{u}_p(k+N_v-1) \end{bmatrix} \\ + &\begin{bmatrix} \Delta\tilde{u}_p(k) \\ \Delta\tilde{u}_p(k+1) \\ \vdots \\ \Delta\tilde{u}_p(k+N_v-1) \end{bmatrix}^T \begin{bmatrix} S & 0 & \cdots & 0 \\ 0 & S & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & S \end{bmatrix} \begin{bmatrix} \Delta\tilde{u}_p(k) \\ \Delta\tilde{u}_p(k+1) \\ \vdots \\ \Delta\tilde{u}_p(k+N_v-1) \end{bmatrix} \end{aligned} \tag{3.20}$$

Then 3.20 can be rewritten as 3.21 using the prediction given by 3.19.

$$\begin{aligned} V_v(k) = \tilde{x}_\Delta^T(k)Q\tilde{x}_\Delta(k) + \tilde{X}_\Delta^T(k+1)\bar{Q}\tilde{X}_\Delta(k+1) + \tilde{U}_p^T(k)\bar{R}\tilde{U}_p(k) \\ + \Delta\tilde{U}_p^T(k)\bar{S}\Delta\tilde{U}_p(k) \end{aligned} \tag{3.21}$$

Given the previous control action $\tilde{u}_p(k-1)$, we can write $\tilde{U}_p(k)$ in terms of $\Delta\tilde{U}_p(k)$ according to 3.22, in which I_u is the identity matrix with dimensions $n_u \times n_u$.

$$\tilde{U}_p(k) = \overbrace{\begin{bmatrix} I_u \\ I_u \\ \vdots \\ I_u \end{bmatrix}}^{M_1} \tilde{u}_p(k-1) + \overbrace{\begin{bmatrix} I_u & 0 & \cdots & 0 \\ I_u & I_u & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ I_u & I_u & \cdots & I_u \end{bmatrix}}^{M_2} \Delta\tilde{U}_p(k) \tag{3.22}$$

Then 3.20 can be expanded as follows

$$\begin{aligned} V_v(k) = \tilde{x}_\Delta^T(k)Q\tilde{x}_\Delta(k) + \tilde{x}_\Delta^T(k)\Phi^TQ\Phi\tilde{x}_\Delta(k) + \tilde{u}_p^T(k-1)M_1^T\Gamma_p^T\bar{Q}\Gamma_pM_1\tilde{u}_p(k-1) \\ + \Delta\tilde{U}_p^T(k)M_2^T\Gamma_p^T\bar{Q}\Gamma_pM_2\Delta\tilde{U}_p(k) + \tilde{u}_p^T(k-1)M_1^T\bar{R}M_1\tilde{u}_p(k-1) \\ + \Delta\tilde{U}_p^T(k)M_2^T\bar{R}M_2\Delta\tilde{U}_p(k) + 2\tilde{x}_\Delta^T(k)\Phi^T\bar{Q}\Gamma_pM_1\tilde{u}_p(k-1) \\ + 2\tilde{x}_\Delta^T(k)\Phi^T\bar{Q}\Gamma_pM_2\Delta\tilde{U}_p(k) + 2\tilde{u}_p^T(k-1)M_1^T\bar{R}M_2\Delta\tilde{U}_p(k) \\ + \Delta\tilde{U}_p^T(k)\bar{S}\Delta\tilde{U}_p(k) \end{aligned} \tag{3.23}$$

Now we can write the control objective in the standard form of a quadratic programming problem

$$V_v(k) = \frac{1}{2}\Delta\tilde{U}_p^T(k)H\Delta\tilde{U}_p(k) + g^T\Delta\tilde{U}_p(k) + c \tag{3.24}$$

with

$$
\begin{aligned}
H &= 2\left(M_2^T \bar{R} M_2 + M_2^T \Gamma_p^T \bar{Q} \Gamma_p M_2 + \bar{S}\right) \\
g &= 2\left(M_2^T \Gamma_p^T \bar{Q} \Phi \tilde{x}_\Delta(k) + M_2^T \Gamma_p^T \bar{Q} \Gamma_p M_1 \tilde{u}_p(k-1) + M_2^T \bar{R} M_1 \tilde{u}_p(k-1)\right) \\
c &= \tilde{x}_\Delta^T(k) Q \tilde{x}_\Delta(k) + \tilde{x}_\Delta^T(k) \Phi^T \bar{Q} \Phi \tilde{x}_\Delta(k) + \tilde{u}_p^T(k-1) M_1^T \Gamma_p^T \bar{Q} \Gamma_p M_1 \tilde{u}_p(k-1) \\
&\quad + \tilde{u}_p^T(k-1) M_1^T \bar{R} M_1 \tilde{u}_p(k-1) + 2\tilde{x}_\Delta^T(k) \Phi^T \bar{Q} \Gamma_p M_1 \tilde{u}_p(k-1)
\end{aligned}
$$

3.2 Illustrative example

The following illustrative example aims to show the advantage of incorporating the zone control strategy in target calculation problem in order to obtain operating points that attend control objectives. In the next chapter we will cover the case in which there is model mismatch as well as wrong identification of the post-fault model. Although the following application example is based on simulations, in the next chapter we will present experimental results of a pH neutralization plant.

3.2.1 The quadruple-tank process

The quadruple-tank process (JOHANSSON, 2000) is a laboratory process that was designed to illustrate various concepts in control design of multivariable processes. The plant consists of four water tanks and two pumps. A schematic diagram is depicted in Figure 6. The system has adjustable zero, which allows to operate with either minimum or nonminimum phase conditions by simply changing valves setting. The goal is to control both lower tanks by manipulating the speed of the pumps. Therefore, the system has two inputs v_1 and v_2 that are input voltages to pump 1 and 2, respectively; and two outputs[1] y_1 and y_2, which are the level of tank 1 and 2, respectively.

The mathematical model can be obtained from mass balances and Bernoulli's law as follows

$$\frac{dh_1}{dt} = -\frac{a_1}{A_1}\sqrt{2gh_1} + \frac{a_3}{A_1}\sqrt{2gh_3} + \frac{\gamma_1 k_1}{A_1} v_1 \tag{3.25}$$

$$\frac{dh_2}{dt} = -\frac{a_2}{A_2}\sqrt{2gh_2} + \frac{a_4}{A_2}\sqrt{2gh_4} + \frac{\gamma_2 k_2}{A_2} v_2 \tag{3.26}$$

$$\frac{dh_3}{dt} = -\frac{a_3}{A_3}\sqrt{2gh_3} + \frac{(1-\gamma_2)\, k_2}{A_3} v_2 \tag{3.27}$$

$$\frac{dh_4}{dt} = -\frac{a_4}{A_4}\sqrt{2gh_4} + \frac{(1-\gamma_1)\, k_1}{A_4} v_1 \tag{3.28}$$

with variables described in Table 1.

[1] The outputs of the original plant are voltages provided by level measurement devices.

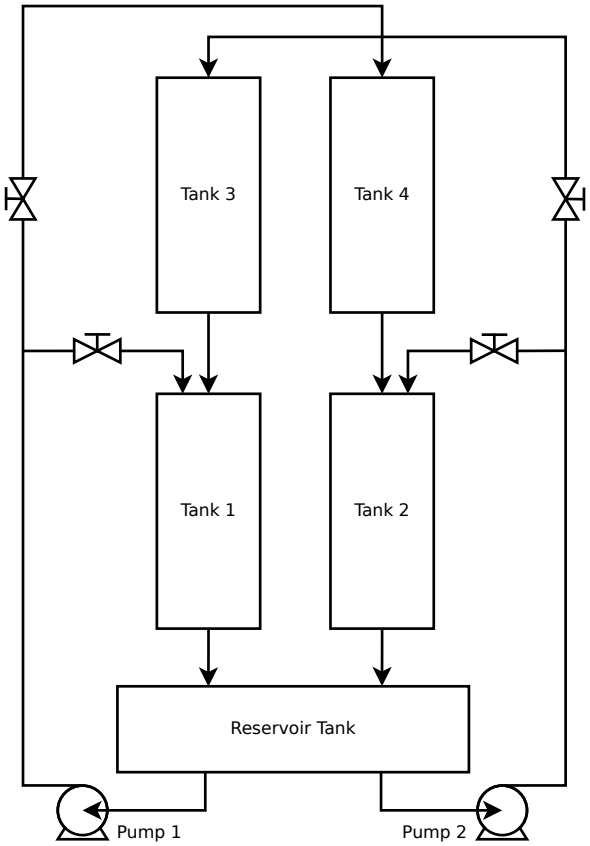

Figure 6 – Schematic diagram of the quadruple-tank process.

Table 1 – Variables of the quadruple-tank process model.

Symbol	Significance
h_i	water level of tank i
a_i	cross-section area of outlet pipe of tank i
A_i	cross-section area of tank i
γ_1	ratio of water diverting to tanks 1 and 4
γ_2	ratio of water diverting to tanks 2 and 3
k_i	gain of pump i
g	acceleration of gravity

Parameters values of the laboratory plant used by Johansson (2000) are given in Table 2.

A linear state-space model that describes the plant around a given operating point is obtained according to Equations 3.29 and 3.30.

$$\frac{dx(t)}{dt} = \begin{bmatrix} -\frac{1}{\tau_1} & 0 & \frac{A_3}{A_1\tau_3} & 0 \\ 0 & -\frac{1}{\tau_2} & 0 & \frac{A_2}{A_4\tau_4} \\ 0 & 0 & -\frac{1}{\tau_3} & 0 \\ 0 & 0 & 0 & -\frac{1}{\tau_4} \end{bmatrix} x(t) + \begin{bmatrix} \frac{\gamma_1 k_1}{A_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{A_2} \\ 0 & \frac{(1-\gamma_2)k_2}{A_3} \\ \frac{(1-\gamma_1)k_1}{A_4} & 0 \end{bmatrix} u(t) \tag{3.29}$$

Table 2 – Parameters of the quadruple-tank process.

Parameter	Value	Units
A_1, A_3	28	cm^2
A_2, A_4	32	cm^2
a_1, a_3	0.071	cm^2
a_2, a_4	0.057	cm^2
k_1	3.33	cm^3/Vs
k_2	3.35	cm^3/Vs
g	981	cm/s^2

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x(t) \tag{3.30}$$

in which $x(t) = h(t) - h^0$ and $u(t) = v(t) - v^0$ are deviations from the operating points and the time constants are defined as

$$\tau_i = \frac{A_i}{a_i}\sqrt{\frac{2h_i^0}{g}}, \qquad i \in \{1, 2, 3, 4\}.$$

Linear model matrices are computed by considering an operating point given in (JOHANSSON, 2000), in which the valve settings correspond to a minimum phase system. The operating point is given in Table 3.

Table 3 – Operating point of the quadruple-tank process.

Variable	Value	Units
h_1^0	12.4	cm
h_2^0	12.7	cm
h_3^0	1.8	cm
h_4^0	1.4	cm
v_1^0	3.0	V
v_2^0	3.0	V
γ_1	0.7	-
γ_2	0.6	-

The discretization of the linear system is performed using the Scilab function *dscr* with a sampling period $T_s = 1\,\text{s}$. The following discrete-time system is obtained:

$$x(k+1) = \begin{bmatrix} 0.9842 & 0 & 0.0407 & 0 \\ 0 & 0.9890 & 0 & 0.0326 \\ 0 & 0 & 0.9590 & 0 \\ 0 & 0 & 0 & 0.9672 \end{bmatrix} x(k) + \begin{bmatrix} 0.0826 & 0.0010 \\ 0 & 0.0625 \\ 0.0005 & 0.0469 \\ 0.0307 & 0 \end{bmatrix} u(k) \tag{3.31}$$

$$y(k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x(k) \tag{3.32}$$

As proposed in (JOHANSSON, 2000), decentralized PI controllers are applied with the following output-input pairing: $y_1 - u_1$, $y_2 - u_2$. The control law is given in 3.33 in which ℓ denotes the closed-loop, $e(k) = y(k) - y_{sp}(k)$ where y_{sp} is the setpoint and K_c and τ_i are tuning parameters given in Table 4.

$$u_\ell(k) = u_\ell(k-1) + K_{c_\ell}\left(e_\ell(k) - e_\ell(k-1) + \frac{T_s}{\tau_{i_\ell}}e_\ell(k)\right), \qquad l \in \{1, 2\} \tag{3.33}$$

Table 4 – Tuning parameters of the PI controllers applied to the quadruple-tank process.

Closed-loop	K_c (V/cm)	τ_i (s)
$y_1 - u_1$	3.0	30
$y_2 - u_2$	2.7	40

As already mentioned, we aim to control the levels of the lower tanks. However, modern industrial plants may have complex control loops with multiple objectives and priority rules. For example, variables with higher economic value or that involve safety hazards may have control priority over the less important ones. Such priorities should exist especially in situations of lack of degrees of freedom which may occur during fault scenarios with actuator losses.

In order to illustrate these situations, we assume that y_1 is an important variable so the control of y_1 is prioritized over y_2. Also, acceptable control zones instead of setpoints will be considered during fault scenarios. Then, y_1 and y_2 should lie inside a zone around their setpoint values. These limits are summarized in terms of $y_{\Delta_{min}}$ and $y_{\Delta_{max}}$ in Table 5.

Table 5 – Control zones during fault scenarios.

Zone limit	Value (cm)
$y_{\Delta_{min}}$	$-[1,\ 2]^T$
$y_{\Delta_{max}}$	$[1,\ 2]^T$

Process constraints given in Table 6 were considered for the purpose of performing the simulations. Outputs constraints were computed by taking into account the operating point (Table 3) as well as tank dimensions given in (JOHANSSON, 2000).

3.2.2 Simulation results

In the following simulations[2], we change the setpoint at time $t = 1$ min from $[y_{sp_1},\ y_{sp_2}]^T = [0,\ 0]^T$ to $[y_{sp_1},\ y_{sp_2}]^T = [4,\ 0.5]^T$. All the results are shown in deviation variables.

[2] Simulations were carried out using Scilab, a free and open-source software for numerical computation (<http://www.scilab.org>).

Table 6 – Process constraints in deviation variables.

Constraint	Value
u_{max}	$[4,\ 4]^T$ V
u_{min}	$-[1,\ 1]^T$ V
Δu_{max}	$[0.5,\ 0.5]^T$ V
Δu_{min}	$-[0.5,\ 0.5]^T$ V
y_{max}	$[7.6,\ 7.3]^T$ cm
y_{min}	$-[12.4,\ 12.7]^T$ cm

Normal operation

The process response in normal operating situation is shown in Figure 7. In addition, Figure 8 shows the setpoint and final value of controlled variables, as well as their feasible space respecting the plant constraints. The process stabilized at the setpoint $[y_{sp1},\ y_{sp2}]^T = [4,\ 0.5]^T$, as expected in fault-free situations.

Scenario 1a

In Figure 9 a fault is inserted at time $t = 5\,\text{min}$, when the input u_1 remains locked in $u_1 = 0.5$. The control of y_2 is not compromised, since the input u_2 remains functional. However, y_1 stabilized with an offset error and the output of the controller reached the upper limit of u_1. We can observe in Figure 10 the new feasible region after the fault occurs, which is represented by just a line. It is also observed that controlled variable y_2 remained at its setpoint value, while the zone control of y_1 was violated.

Scenario 1b

The same fault scenario was simulated with control reconfiguration after the fault occurrence. The penalty matrices used in the two-layer MHVA are given in Table 7.

Table 7 – Penalty matrices of the two-layer MHVA in the Scenario 1a.

Parameter	Value
Q_t	$diag\,([20,\ 10,\ 0,\ 0])$
R_t	$diag\,([1,\ 1]) \times 10^{-1}$
S_t	$diag\,([0,\ 0])$
Q_v	$diag\,([5,\ 1,\ 0,\ 0])$
R_v	$diag\,([1,\ 1]) \times 10^{-1}$
S_v	$diag\,([1,\ 1])$

Note that we have chosen Q_t so the distance between the target and the setpoint of y_1 is more penalized in comparison to y_2. In addition, we chose S null in this example to disregard the zone control. The process response is presented in Figure 11, in which we

observed that y_1 stabilized closer to its control range, while y_2 remained within its range. Figure 12 illustrates the approximation of y_1 towards its range. However, there is a region where both variables are within their specified ranges. This region would be preferred in a fault scenario according to the assumptions regarding control objectives.

Scenario 1c

Thus, a new simulation was performed activating the control zones by choosing nonzero S. The tuning parameters os this scenario are given in Table 8. Note that this value of S allows the control zone of y_2 to be violated before the y_1 one. The process response (Figure 13) was similar to the previous case, but stabilized within the range specified for both controlled variables. Figure 14 illustrates the steady state reached after the fault occurrence. In this way, the use of nonzero penalty of slack variables increases the attraction of targets to the control zone.

Table 8 – Penalty matrices of the two-layer MHVA in the Scenario 1b.

Parameter	Value
Q_t	$diag\left([20,\ 10,\ 0,\ 0]\right)$
R_t	$diag\left([1,\ 1]\right) \times 10^{-1}$
S_t	$diag\left([10^5,\ 1]\right) \times 10^{10}$
Q_v	$diag\left([5,\ 1,\ 0,\ 0]\right)$
R_v	$diag\left([1,\ 1]\right) \times 10^{-1}$
S_v	$diag\left([1,\ 1]\right)$

Scenario 2a

In this scenario, a fault is inserted into the plant so that u_1 remains locked at the nominal equilibrium point, i.e. $u_1 = 0$. The process response without control reconfiguration is presented in Figure 15 and the steady state reached after the fault is shown in Figure 16. Similar to the Scenario 1a, the nominal controller output concerning u_1 saturated at its upper limit, while the control of y_2 was not affected, since the actuator referring to u_2 still remained functional.

Scenario 2b

Figure 17 shows the process response when the same fault of the previous scenario is inserted into the plant and the control reconfiguration is performed with the tuning parameters given in Table 7. The fault was hidden and the nominal controller output signals remained the same as the fault-free scenario. However, it is interesting to note that the system reached a steady state that violates the control zones of both controlled variables, as shown in Figure 18. This occurs because, without the penalty of

the zones violation, the targets are computed only by taking into consideration the penalty of distances from setpoints.

Scenario 2c

Now we change the tuning parameters according to Table 8 and obtain the results shown in Figure 19. In this case, the system stabilizes in a more interesting steady state according to the control objectives (Figure 20). This time, y_1 is held at the edge of its control zone, while only the control zone of y_2 is violated.

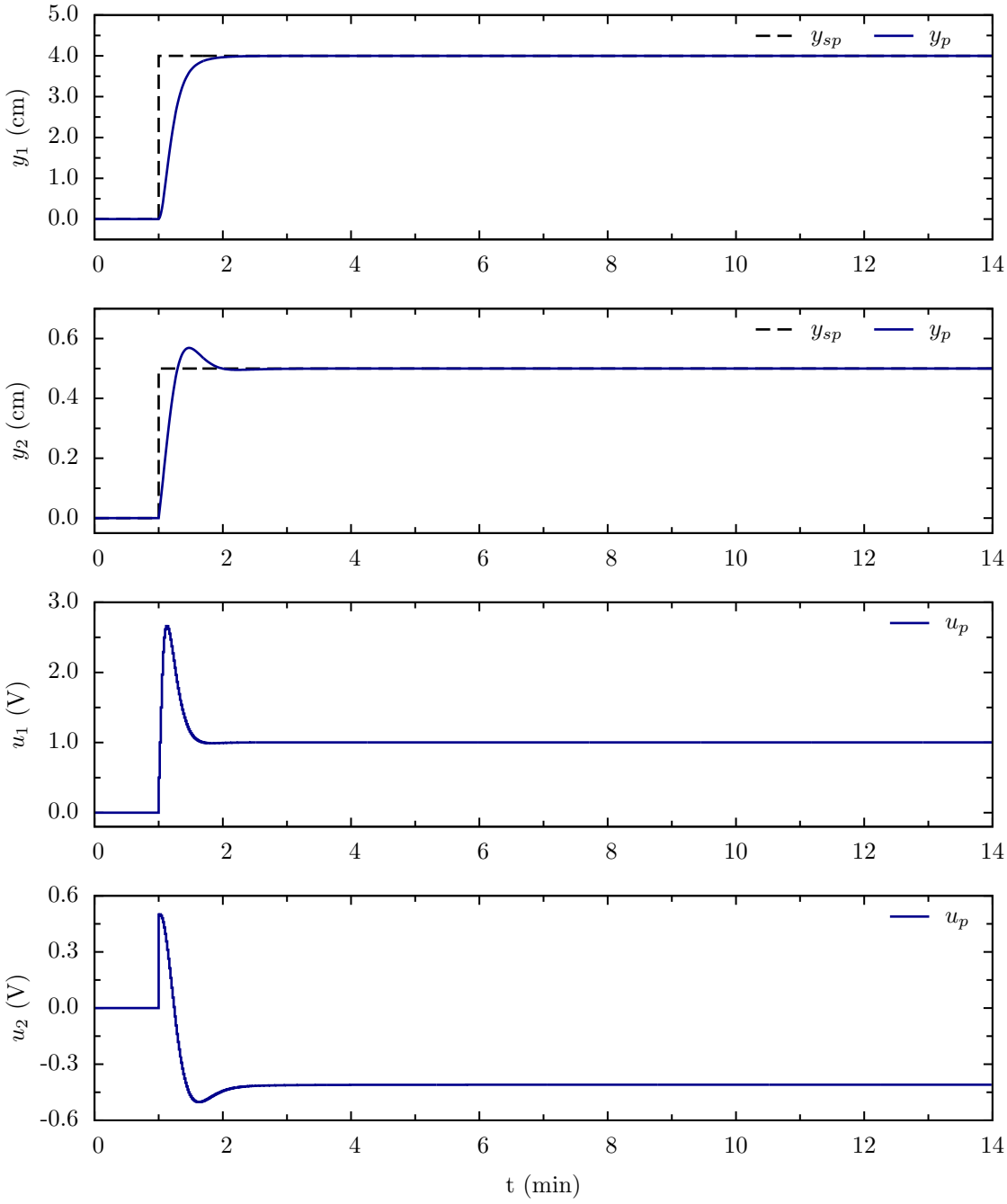

Figure 7 – Process response after a setpoint change during normal operation.

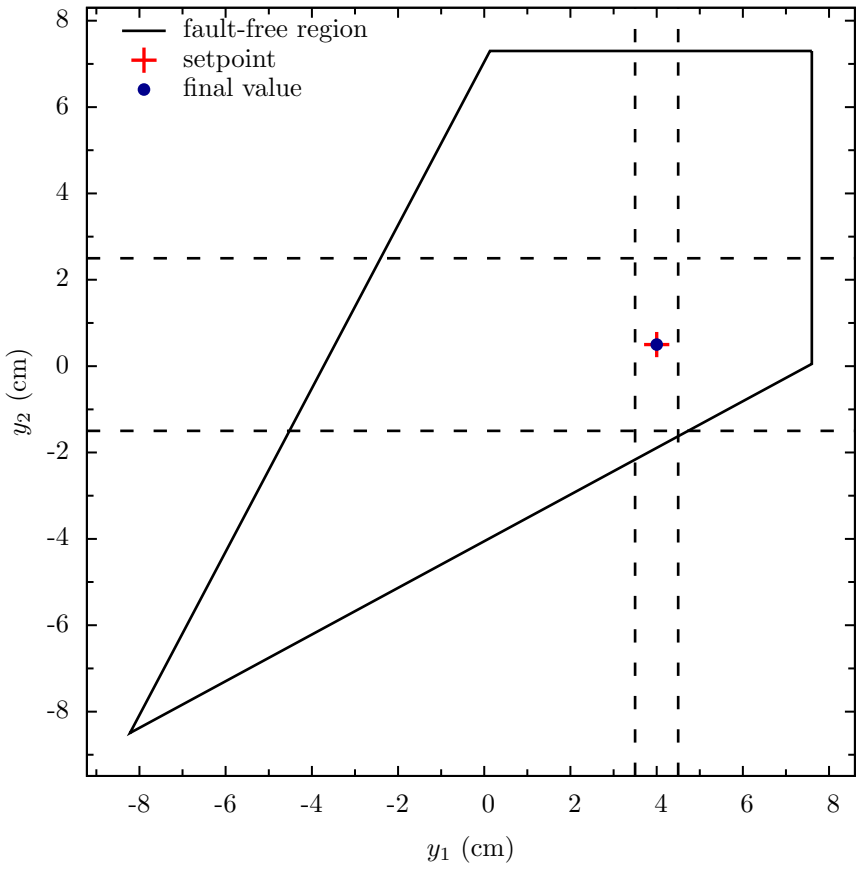

Figure 8 – Output space, setpoint and final value during normal operation.

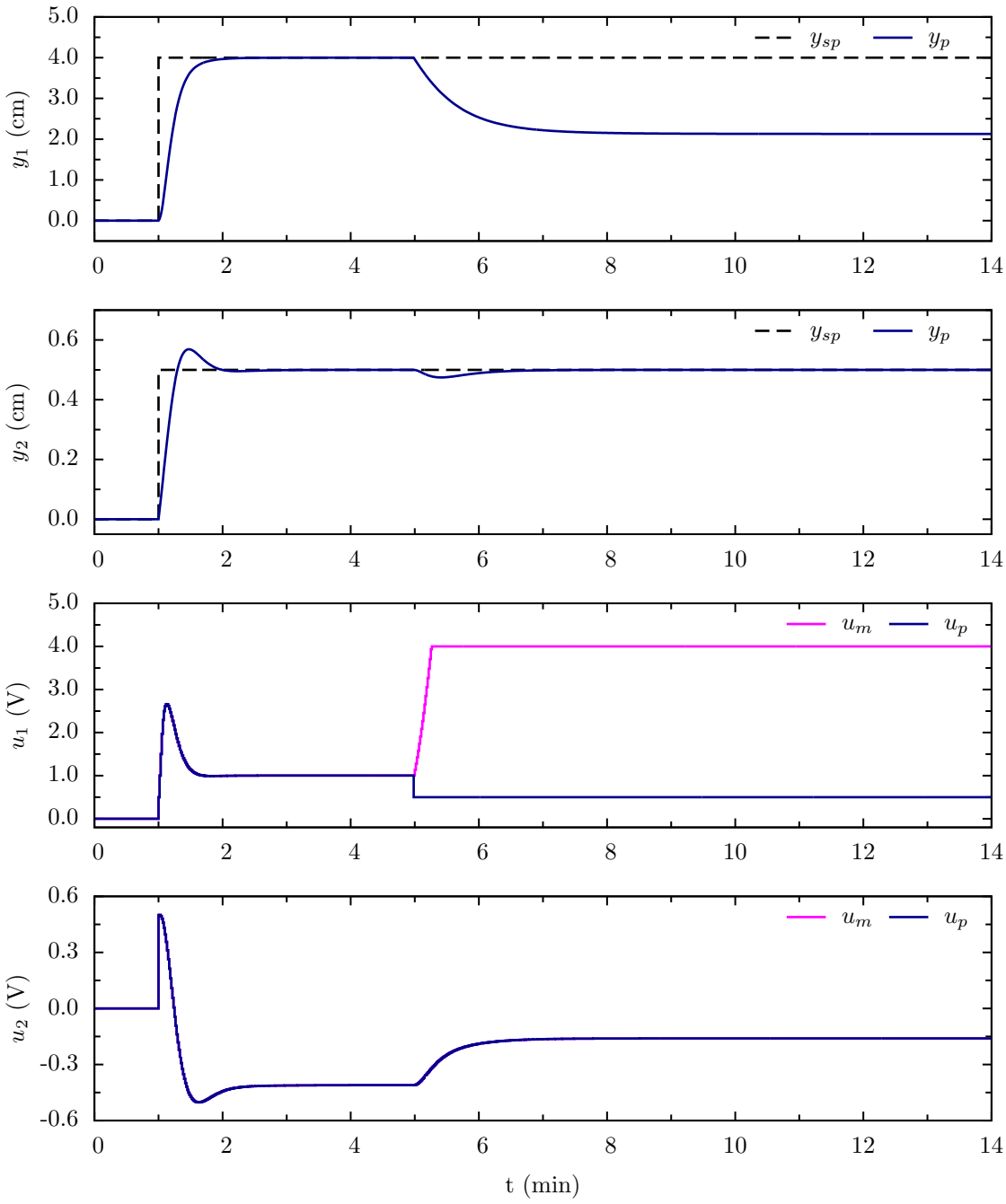

Figure 9 – Scenario 1a: the actuator related to u_1 became stuck at 0.5 V.

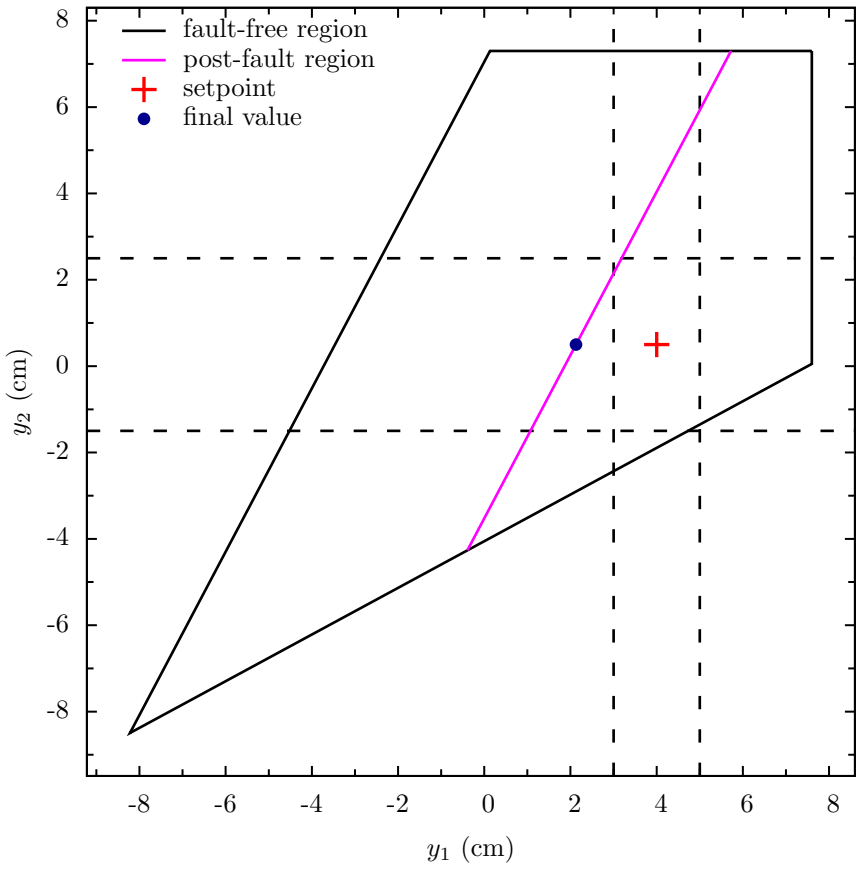

Figure 10 – Scenario 1a: Output space, setpoint and final value.

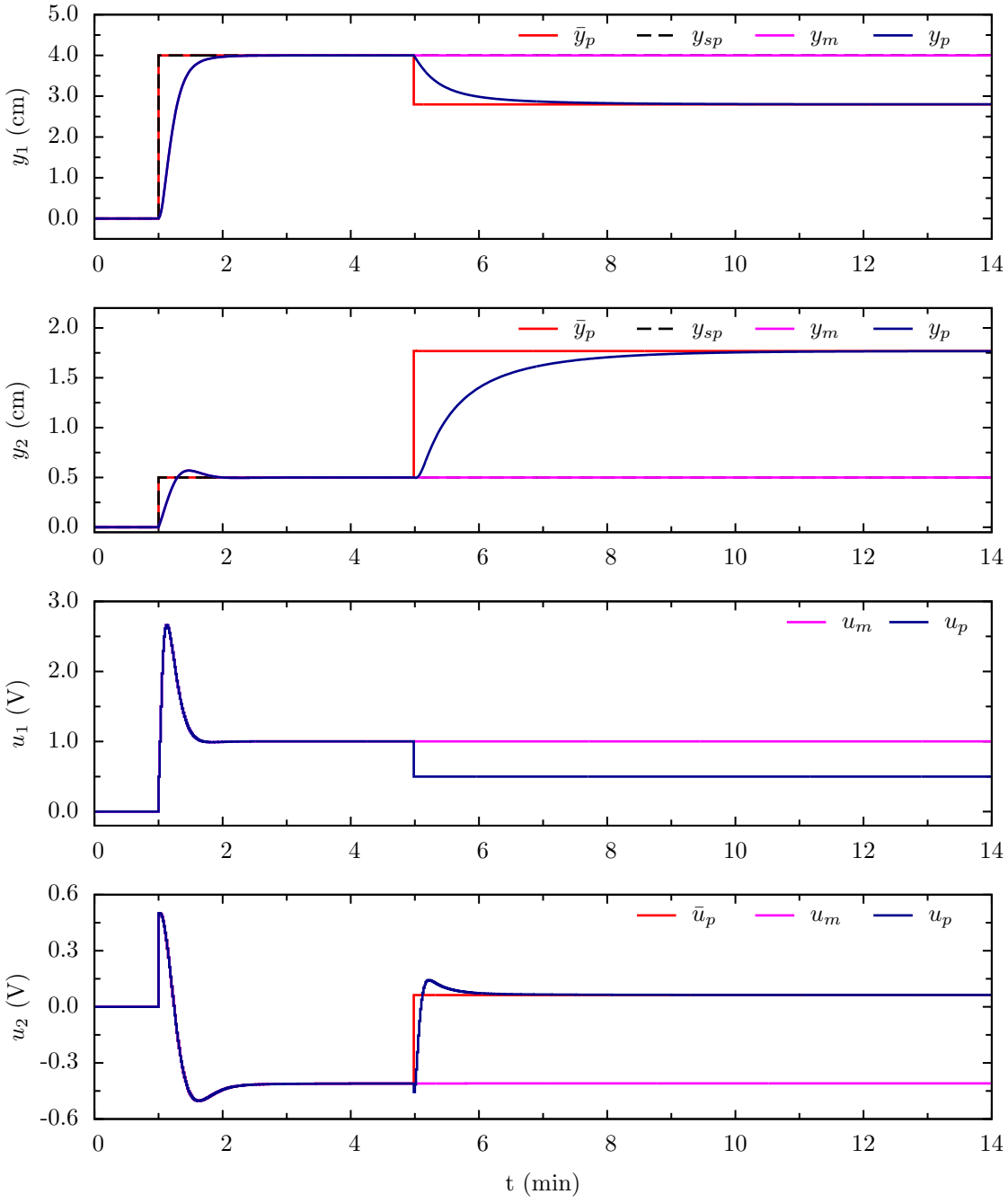

Figure 11 – Scenario 1b: control reconfiguration without control zones when the actuator related to u_1 became stuck at 0.5 V.

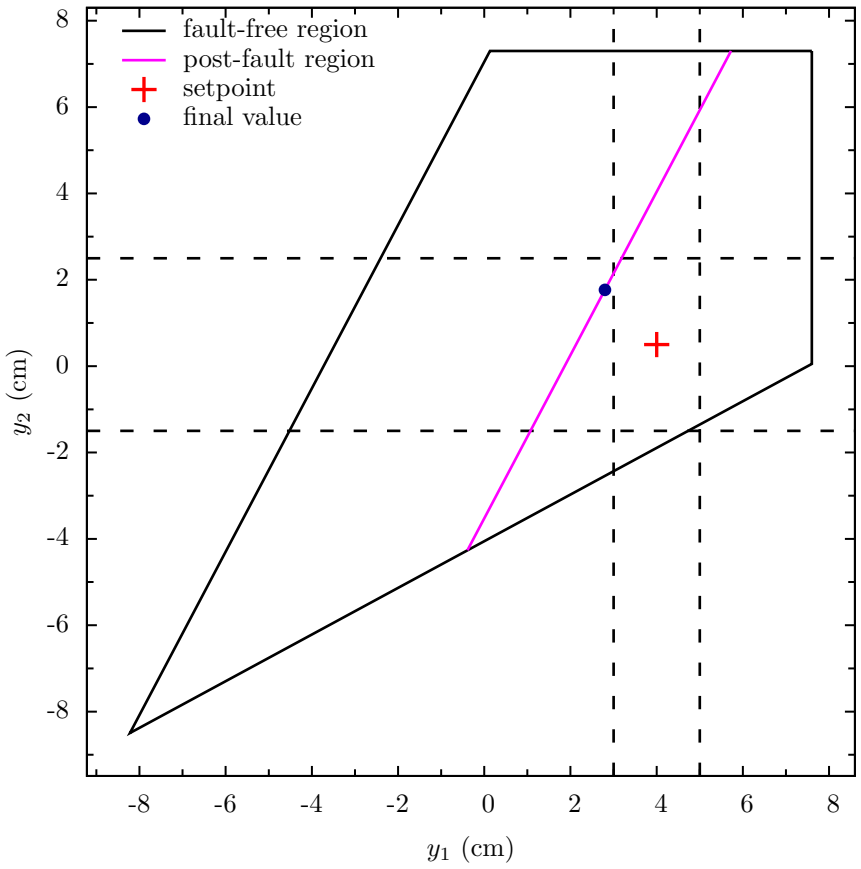

Figure 12 – Scenario 1b: Output space, setpoint and final value.

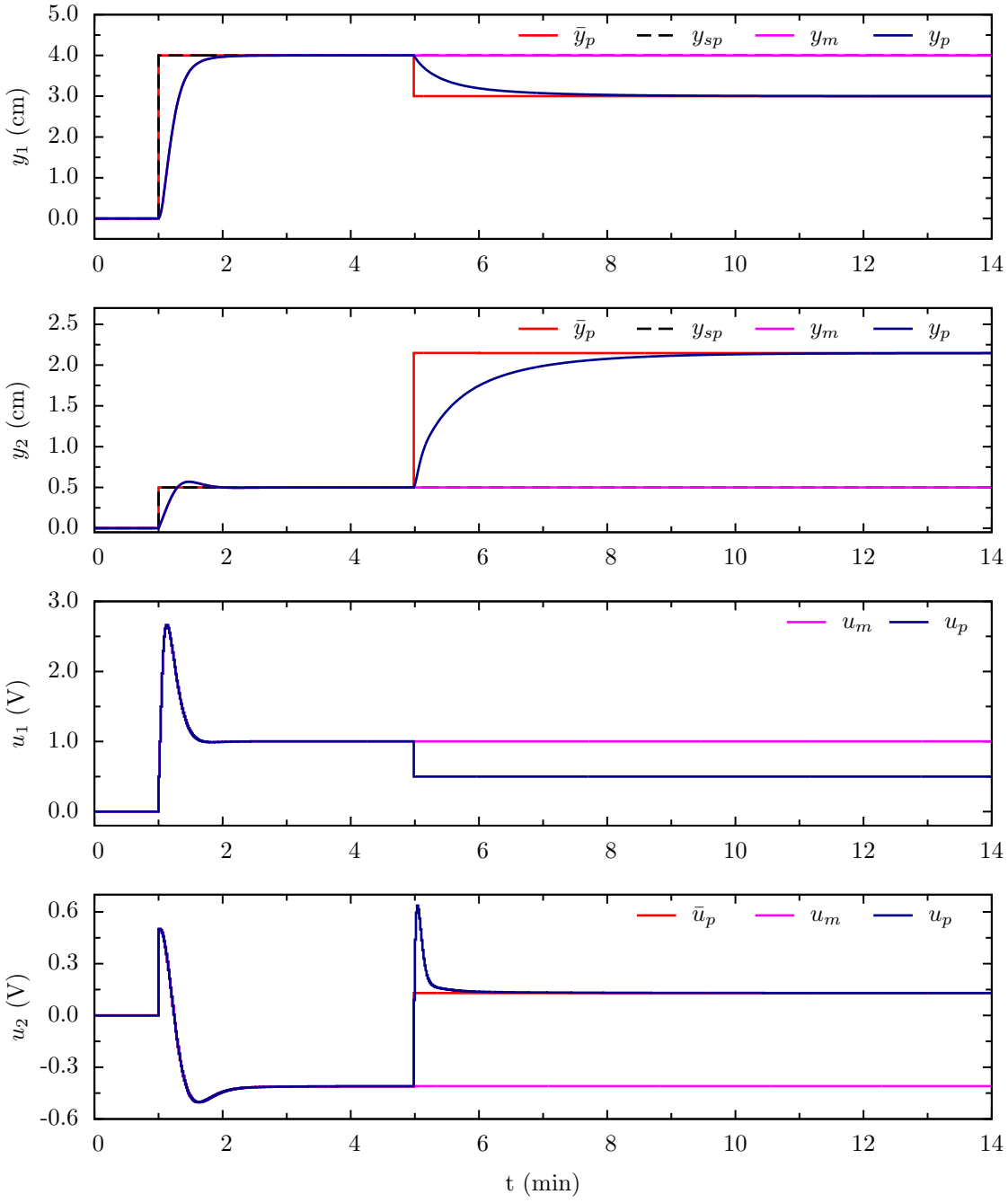

Figure 13 – Scenario 1c: control reconfiguration with control zones when the actuator related to u_1 became stuck at 0.5 V.

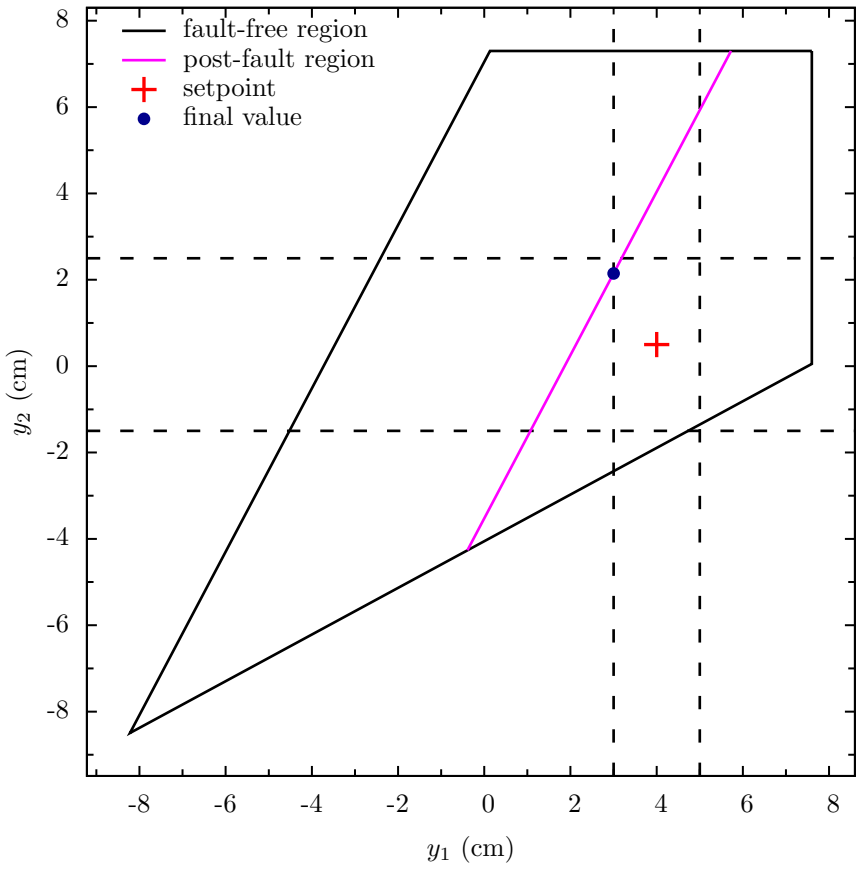

Figure 14 – Scenario 1c: Output space, setpoint and final value.

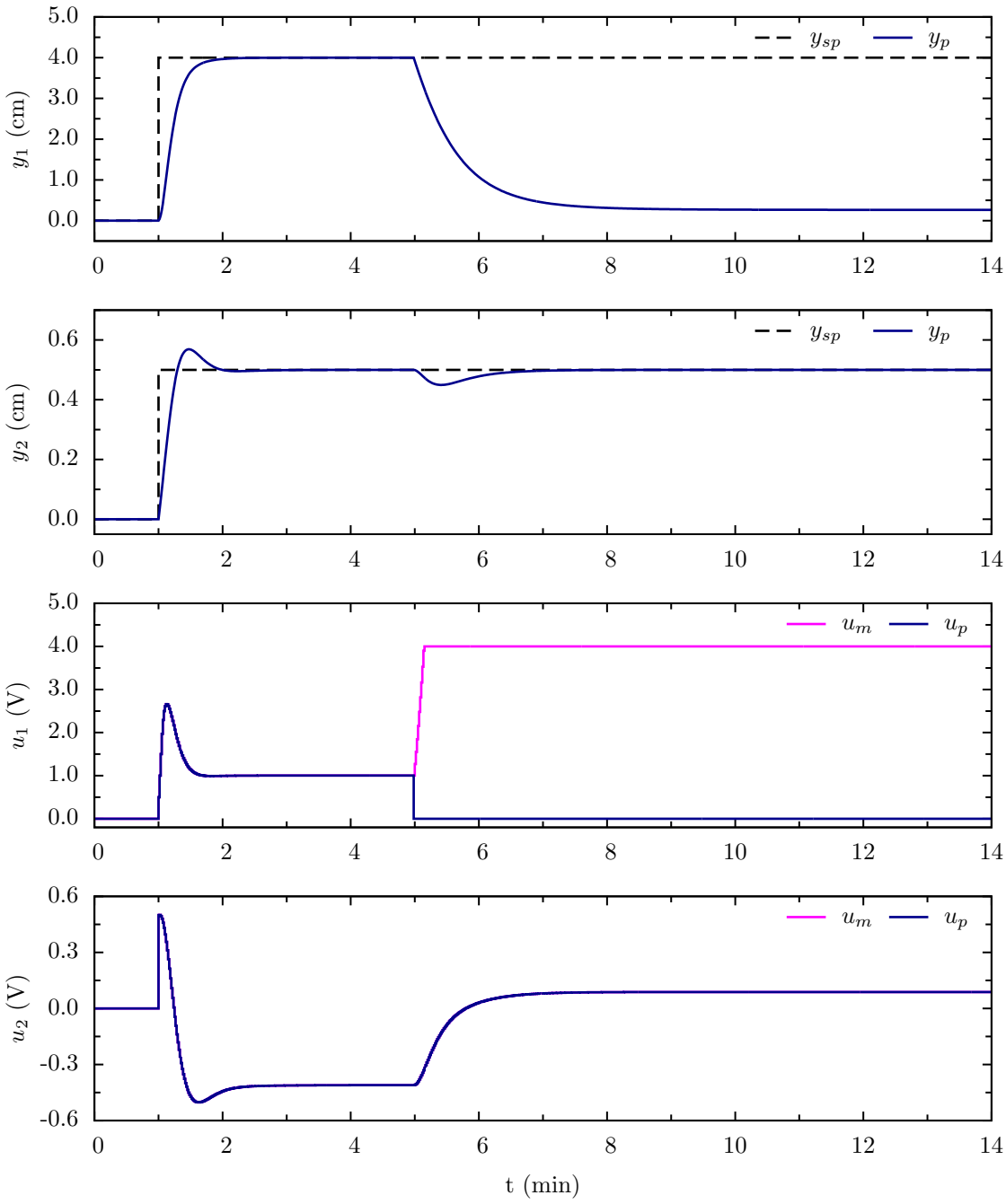

Figure 15 – Scenario 2a: the actuator related to u_1 became stuck at 0 V.

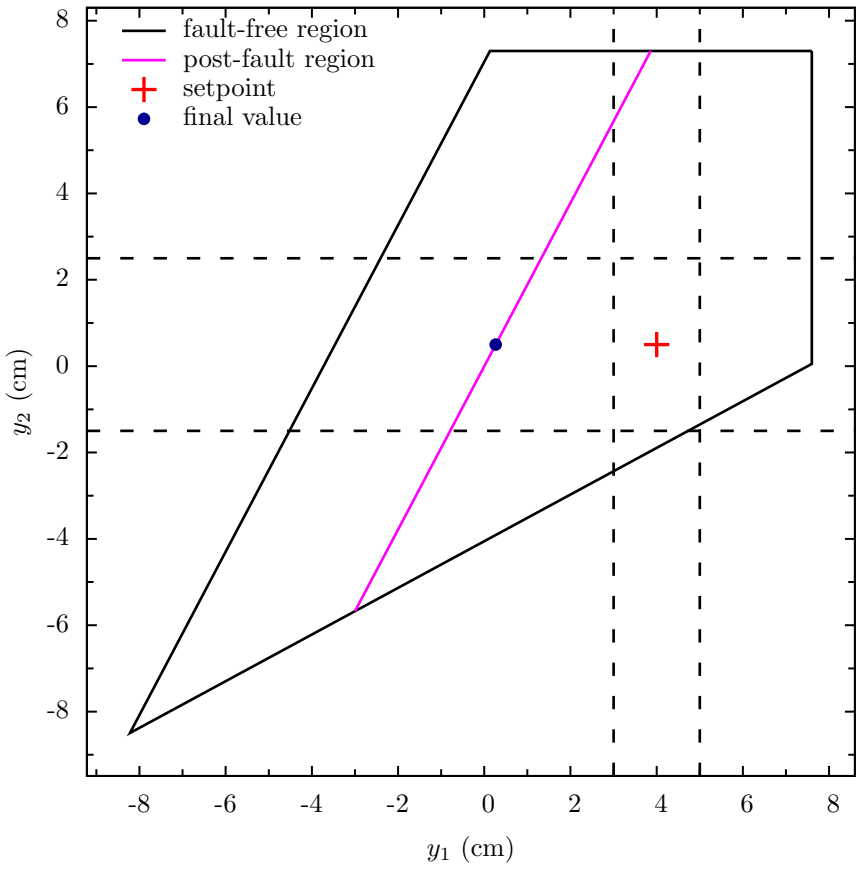

Figure 16 – Scenario 2a: Output space, setpoint and final value.

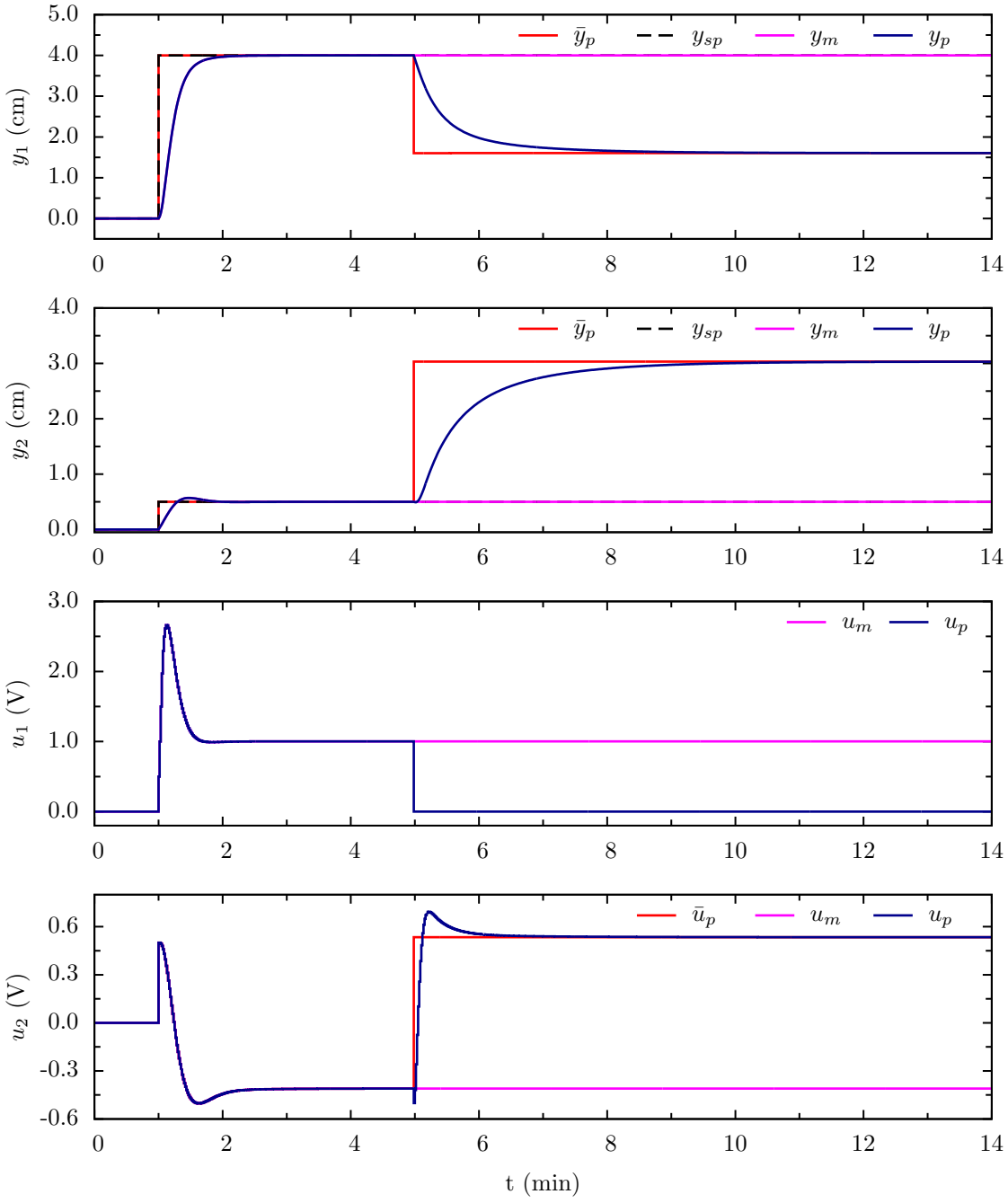

Figure 17 – Scenario 2b: control reconfiguration without control zones when the actuator related to u_1 became stuck at 0 V.

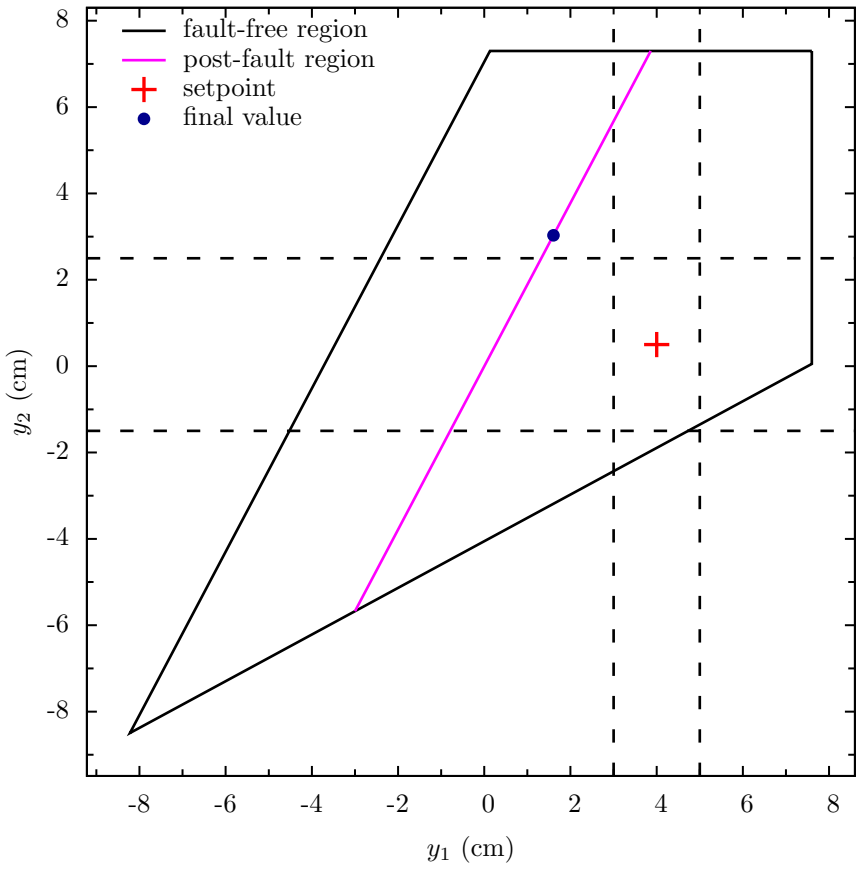

Figure 18 – Scenario 2b: Output space, setpoint and final value.

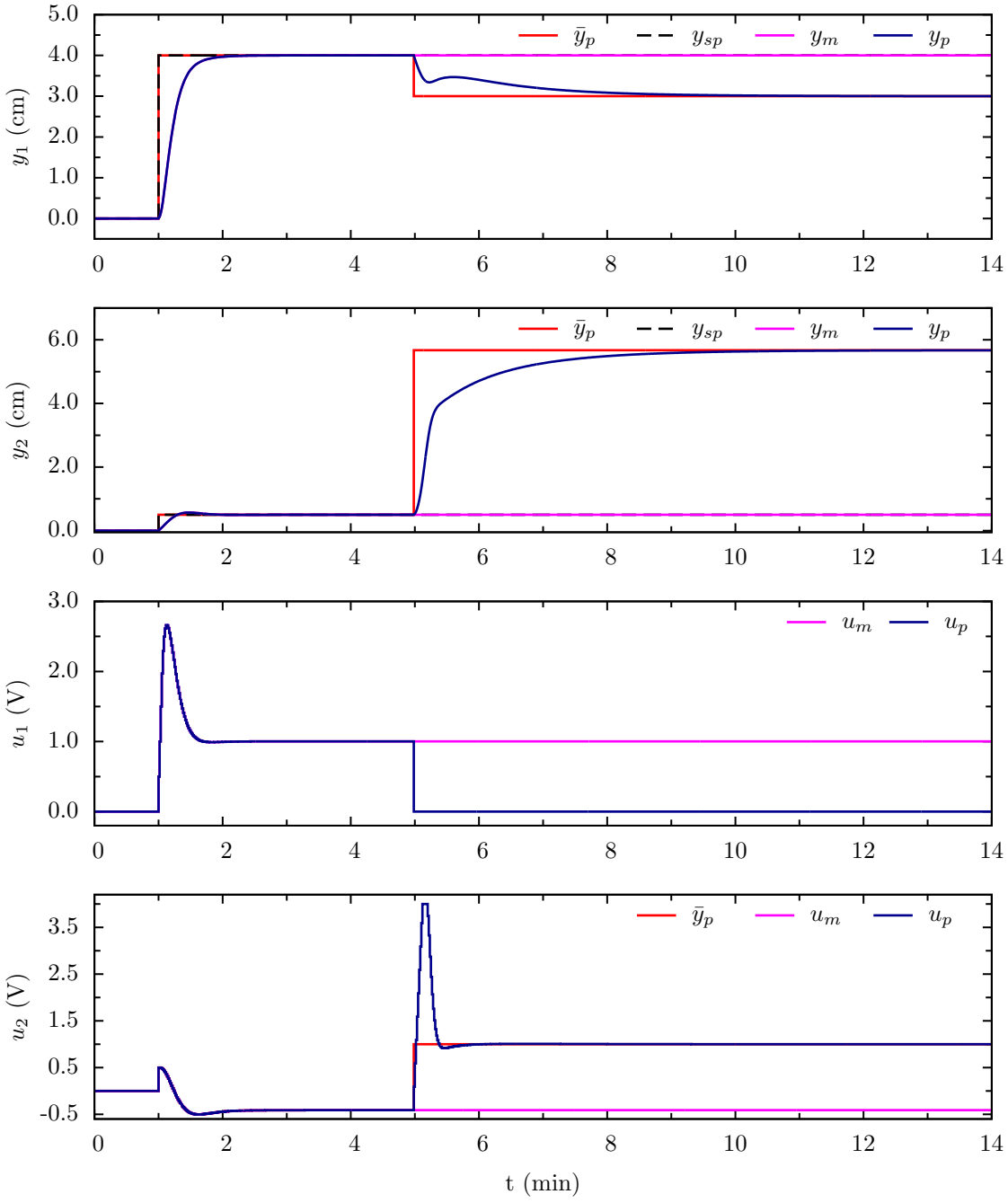

Figure 19 – Scenario 2c: control reconfiguration with control zones when the actuator related to u_1 became stuck at 0 V.

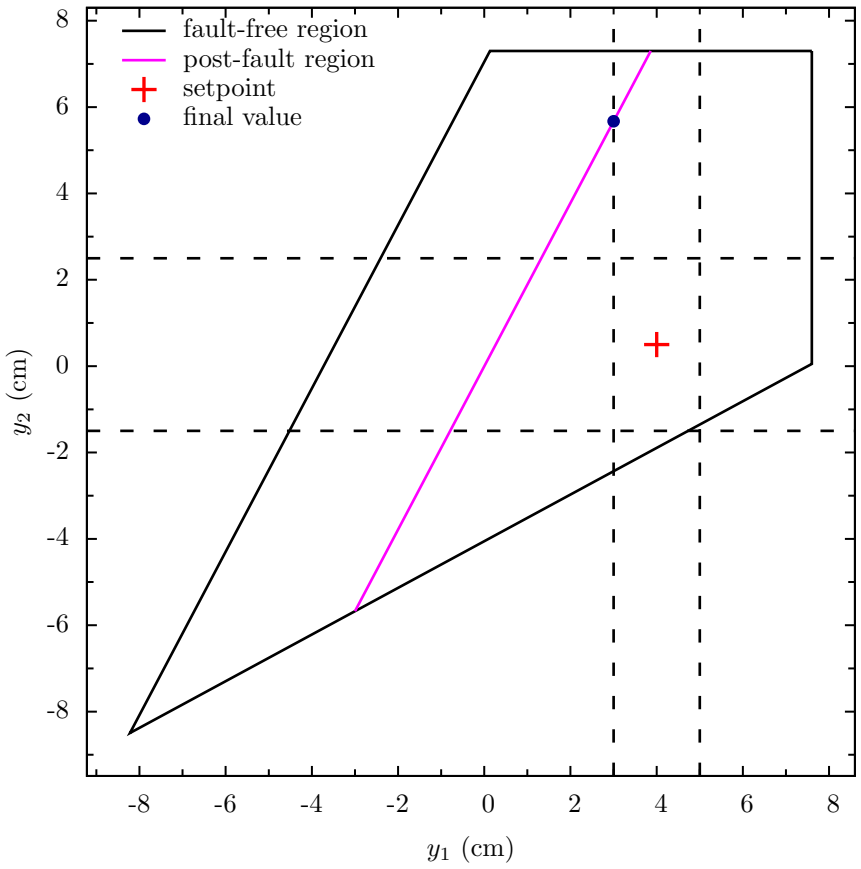

Figure 20 – Scenario 2c: Output space, setpoint and final value.

CHAPTER 4

Application to an experimental pH neutralization plant

4.1 The pH neutralization experiment

The use of pH control is widely found in many chemical and biotechnological processes with applications ranging from the maintenance of pH within optimal ranges of operation in fermentation processes to neutralization of industrial effluents that are later destined to water bodies. In addition to its applicability, pH neutralization processes have also been employed in the study and development of new control strategies due to their high static nonlinearity, which represents a challenge from the point of view of controllers design (HERMANSSON; SYAFIIE, 2015).

Thus, the ease of experimental design of the neutralization process as well as the possibility of studying advanced control structures made possible the construction of the experimental pH neutralization plant (Figure 21) located in the Laboratory of Process Control and Automation (LCAP) of the School of Chemical Engineering at the University of Campinas. The experimental set-up was constructed and initially operated by Costa (2010) for his master dissertation in which he implemented a GPC controller (Generalized Process Control) based on a network of local linear models (COSTA 2015a). After, the plant was modified as described in Costa's PhD thesis Costa (2014), allowing the application of the moving horizon virtual actuator. Thus this plant has been used for the experimental evaluation of the proposed reconfiguration block presented in this work.

Figure 21 – The experimental pH neutralization plant.

4.1.1 Experimental apparatus description

The instrumentation diagram[1] of the experimental pH neutralization process is shown in Figure 22. The reactor is fed by four streams: two of nitric acid (HNO_3) solution, one of sodium hydroxide (NaOH) solution and one of sodium bicarbonate ($NaHCO_3$) solution. Concentration values of these streams were defined based on Costa (2014) and are shown in Table 9.

Table 9 – Feed streams concentrations.

Chemical specie	Name	Stream	Concentration (mol/L)
HNO_3	nitric acid #1	1	0.0028
NaOH	sodium hydroxide	2	0.0057
HNO_3	nitric acid #2	3	0.0034
$NaHCO_3$	sodium bicarbonate	4	0.03

In this plant, acid and base solutions are stored in tanks with 100 L capacity whose outputs are connected to gear pumps magnetically coupled to motors that are actuated by standard $4-20$ mA analog signal. The volumetric flow in these streams are measured by positive-displacement flow sensors that send pulse signals to transmitters that, in turn, convert to standard $4-20$ mA signal and send it to a Programmable Logic Controller (PLC). PI (Proportional-Integral) controllers embedded in the PLC are used to control the flow of these streams. The sodium bicarbonate buffer solution is stored in

[1] Further information regarding brands and models of the equipment installed in the plant can be found in Costa's thesis (COSTA, 2014), in Portuguese.

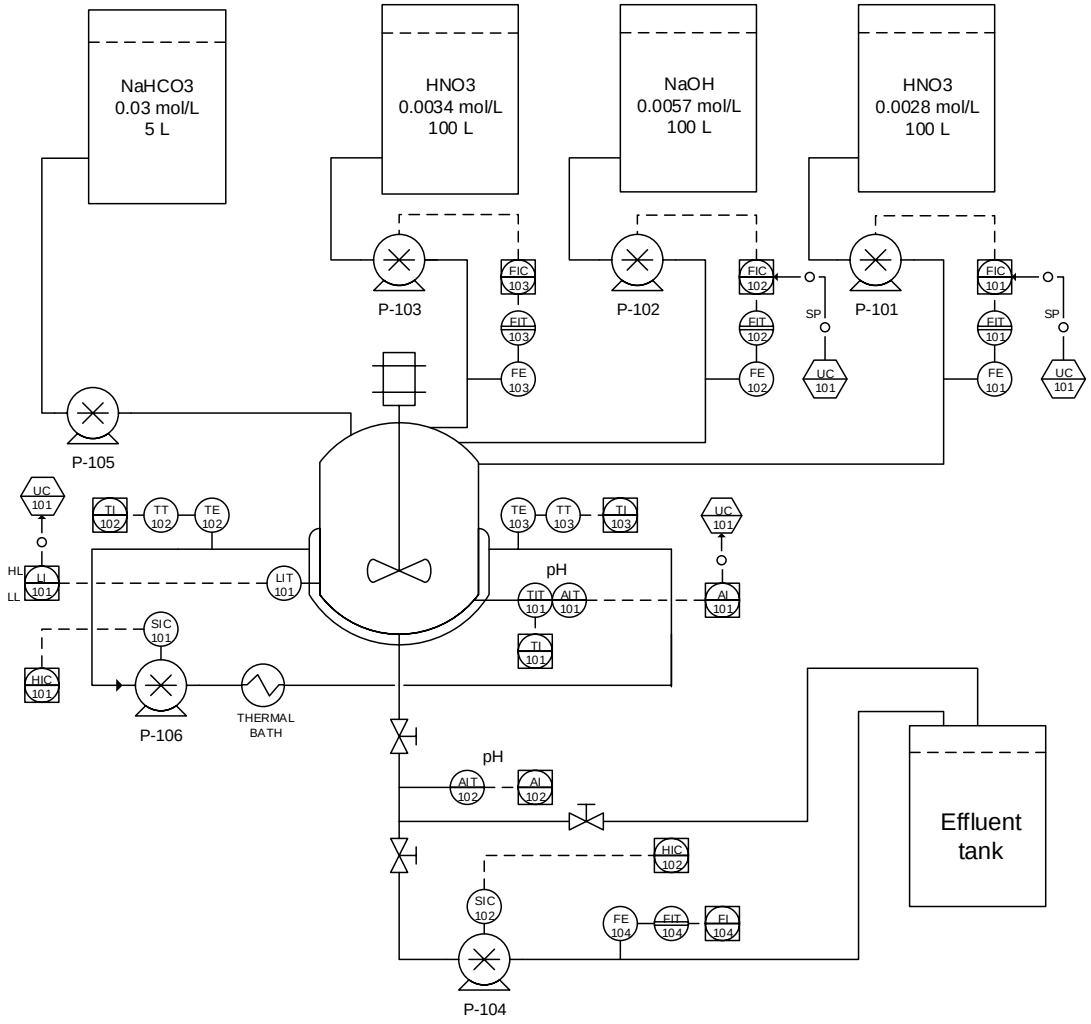

Figure 22 – Instrumentation diagram of the pH neutralization plant.

Source: Reproduced from Costa (2014) with author's permission.

a 5 L tank and is sent to the reactor through a peristaltic pump whose flow can be set manually by means of a potentiometer.

The reactor consists of a jacketed cylindrical vessel with nominal capacity of 6 L, height of 30 cm and base area of 201.06 cm^2. A mechanical stirrer is used for mixing the reaction medium into which is immersed an industrial pH electrode connected to a pH transmitter and indicator. The level of the reaction medium is inferred by measuring the hydrostatic pressure of the liquid by means of a pressure sensor located at the reactor base.

The reactor effluent is collected in a 200 L storage tank where it is treated before disposal. The liquid flow in the outlet stream can be carried out either by gravity or by means of a positive displacement pump whose motor is connected to a variable-frequency drive. A manual valve allows the operator to select the desired output flow setting. It should be noted that level and pH control loops are highly coupled in the case of output flow by gravity. On the other hand, the use of a pump in the outlet stream adds an

integrating mode to the system dynamics. In this work the output flow by gravity was used.

Also, there is a cooling system installed in the plant consisting of a thermal bath and a positive displacement pump that circulates the utility fluid (propylene glycol solution) through the reactor jacket. However, this cooling system was not used in this work.

4.1.2 Plant model

In order to obtain a phenomenological model for the pH neutralization system, we use theoretical concepts presented in (GUSTAFSSON; WALLER, 1983; GUSTAFSSON et al., 1995; HENSON et al., 1994).

In general, acid-base systems models consist of mass balances for hydrogen and hydroxide ions as well as for ampholytes and weak acids and bases. Since acid-base reactions between diluted components usually have high reaction rates, we can consider the system is approximately at equilibrium so additional algebraic state equations may be used (GUSTAFSSON; WALLER, 1983).

Reactions between strong acid and base as well as carbonates ions are present in the system reactor described above. Considering the equilibrium is approached, we may write the following dissociation reactions:

$$\mathrm{H_2O} \rightleftharpoons \mathrm{H^+ + OH^-} \tag{4.1}$$

$$\mathrm{H_2CO_3} \rightleftharpoons \mathrm{H^+ + HCO_3^-} \tag{4.2}$$

$$\mathrm{HCO_3^-} \rightleftharpoons \mathrm{H^+ + CO_3^{2-}} \tag{4.3}$$

The ionic product of water is given in 4.4 and equilibrium relations between chemical species in 4.2 and 4.3 are given by 4.5 and 4.6, respectively.

$$K_w = [\mathrm{H^+}][\mathrm{OH^-}] \tag{4.4}$$

$$K_{a1} = \frac{[\mathrm{HCO_3^-}][\mathrm{H^+}]}{[\mathrm{H_2CO_3}]} \tag{4.5}$$

$$K_{a2} = \frac{[\mathrm{CO_3^{2-}}][\mathrm{H^+}]}{[\mathrm{HCO_3^-}]} \tag{4.6}$$

The electroneutrality condition and the mass balance for the weak acid and its conjugates species are described by the reaction invariants 4.7 and 4.8, respectively, for each reactor inlet i.

$$W_{ai} = [\mathrm{H^+}]_i - [\mathrm{OH^-}]_i - [\mathrm{HCO_3^-}]_i - 2[\mathrm{CO_3^-}]_i \tag{4.7}$$

$$W_{bi} = [\mathrm{H_2CO_3}]_i + [\mathrm{HCO_3^-}]_i + [\mathrm{CO_3^{2-}}]_i \tag{4.8}$$

Then, mass balance equations can be written as follows

$$A_b h \frac{dW_a}{dt} = q_1 (W_{a1} - W_a) + q_2 (W_{a2} - W_a) + q_3 (W_{a3} - W_a) + q_4 (W_{a4} - W_a) \quad (4.9)$$

$$A_b h \frac{dW_b}{dt} = q_1 (W_{b1} - W_b) + q_2 (W_{b2} - W_b) + q_3 (W_{b3} - W_b) + q_4 (W_{b4} - W_b) \quad (4.10)$$

$$A_b \frac{dh}{dt} = q_1 + q_2 + q_3 + q_4 - q \quad (4.11)$$

in which A_b denotes the reactor base area, h is the reactor level, q_i denotes the inlet i flow rate and q is the outlet flow rate.

Finally, from 4.7 and 4.8 we can obtain an implicit nonlinear equation that relates the reaction invariants W_a and W_b to the pH of the medium according to 4.12.

$$W_a + 10^{pH-14} - 10^{-pH} + W_b \frac{1 + 2 \times 10^{pH-pK_{a2}}}{1 + 10^{pK_{a1}-pH} + 10^{pH-pK_{a2}}} = 0 \quad (4.12)$$

with

$$pH = -\log\left([\mathrm{H}^+]\right) \quad (4.13)$$

$$pK_{ai} = -\log\left(K_{ai}\right), \quad i \in \{1, 2\} \quad (4.14)$$

Following the steps described in (COSTA, 2014) and considering the operating point given in Table 10, we obtain the discrete-time linear model given in 4.15 and 4.16.

Table 10 – Operating point of the pH neutralization plant.

Variable	Value	Units
W_a^0	-4.6627×10^{-4}	mol/L
W_b^0	6.5059×10^{-4}	mol/L
h^0	19.57	cm
pH^0	6.75	-
q_1^0	29	L/h
q_2^0	29	L/h
q_3^0	29	L/h
q_4^0	1.93	L/h

$$x(k+1) = \begin{bmatrix} 0.8820 & 0 & 0 \\ 0 & 0.8820 & 0 \\ 0 & 0 & 0.9470 \end{bmatrix} x(k) + \begin{bmatrix} 1.5878 & -2.5102 & 1.8008 \\ -0.3108 & -0.3108 & -0.3108 \\ 0.0968 & 0.0968 & 0.0968 \end{bmatrix} u(k) \quad (4.15)$$

$$y(k) = \begin{bmatrix} 0 & 0 & 1 \\ -0.0328 & -0.0235 & 0 \end{bmatrix} x(k) \quad (4.16)$$

with

$$x(k) = \begin{bmatrix} W_a(k) - W_a^0 \\ W_b(k) - W_a^0 \\ h(k) - h^0 \end{bmatrix}, \quad u(k) = \begin{bmatrix} q_1(k) - q_1^0 \\ q_2(k) - q_2^0 \\ q_3(k) - q_3^0 \end{bmatrix}, \quad y(k) = \begin{bmatrix} h(k) - h^0 \\ pH(k) - pH^0 \end{bmatrix}$$

4.2 Experimental results

4.2.1 Normal operation

In the plant start-up, all the inlet flow rates were set up at their nominal steady state values. Then, the controller was turned on when reactor level reached about 70%. Initially, the setpoint for the level and pH were 20 cm and 8, respectively. At time 10 min the pH setpoint was changed to 8.8. Figure 23 shows the process response. Although the process was not at the steady state in the beginning of the experiment, the controlled variables were driven to their setpoint values before the pH setpoint change. After the pH setpoint change the reactor level was not disturbed since the control loops are decoupled.

4.2.2 Actuator fault scenarios

Scenario 1a

An actuator fault was inserted at time $t = 15$ min in which the actuator related to the acid #1 flow rate became stuck. This was performed by setting q_1 to 32 L/h and disregarding the nominal controller output for this variable. The results have shown steady state error in both pH and level as expected since only one actuator was functioning after the fault occurrence (Figure 24). It should be noticed that the system control resulted in steady state error in both variables because it was operating with decoupled control loops. Otherwise, in a coupled loops situation, we would expect an offset only in the reactor level.

Scenario 1b

In this scenario, the same fault described in Scenario 1a was inserted in the experiment. However, the control loop was reconfigured immediately after the fault occurrence. The results are shown in Figure 25. The offset was removed only in the pH, but it was increased in the level. This occurs because the system does not have sufficient degrees of freedom to drive both variables to their setpoints. In this case, there is a tradeoff between removing the offset of one or other controlled variable. Then, supposing that the pH is a more important variable and must be as close as possible to its setpoint, it was prioritized over the reactor level. In addition, the target level was computed to suit the new steady state.

Also, the fault was hidden from nominal controller perspective. Then the nominal plant output signal y_m was driven to the setpoint in both controlled variables and the output of the nominal controller did not saturate at the limit of q_2.

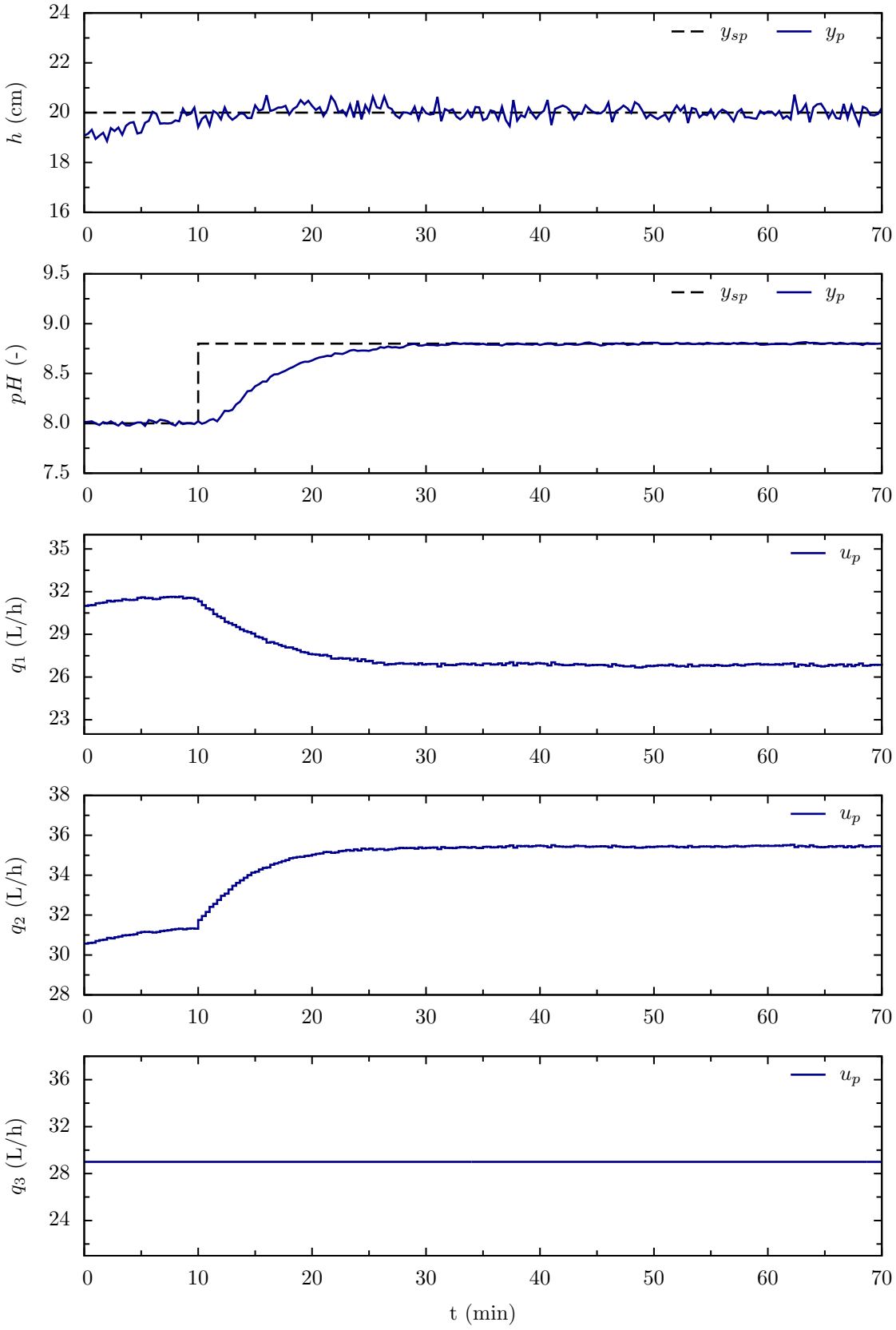

Figure 23 – Process response after a pH setpoint change during normal operation.

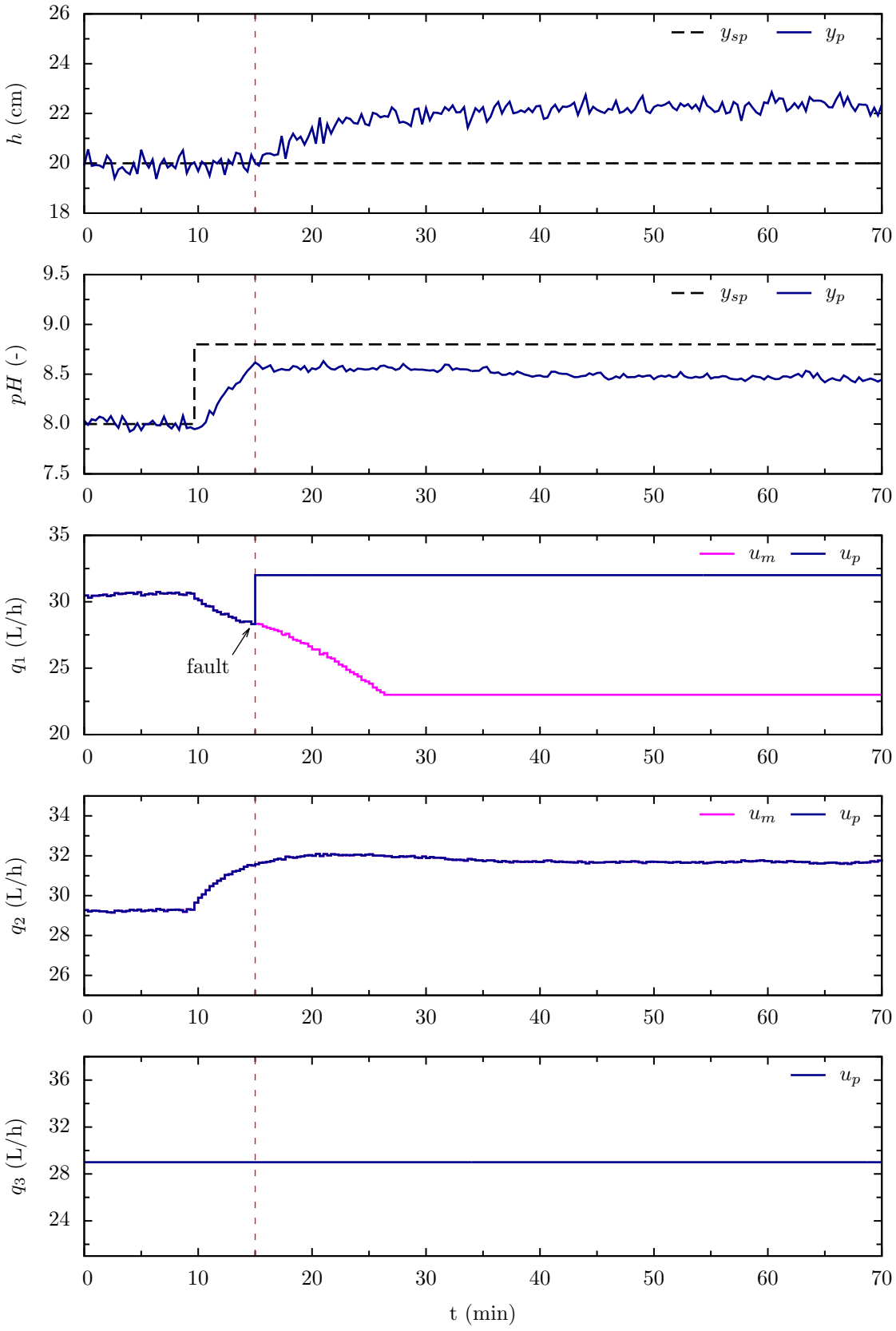

Figure 24 – Scenario 1a: the actuator related to q_1 became stuck at 32 L/h.

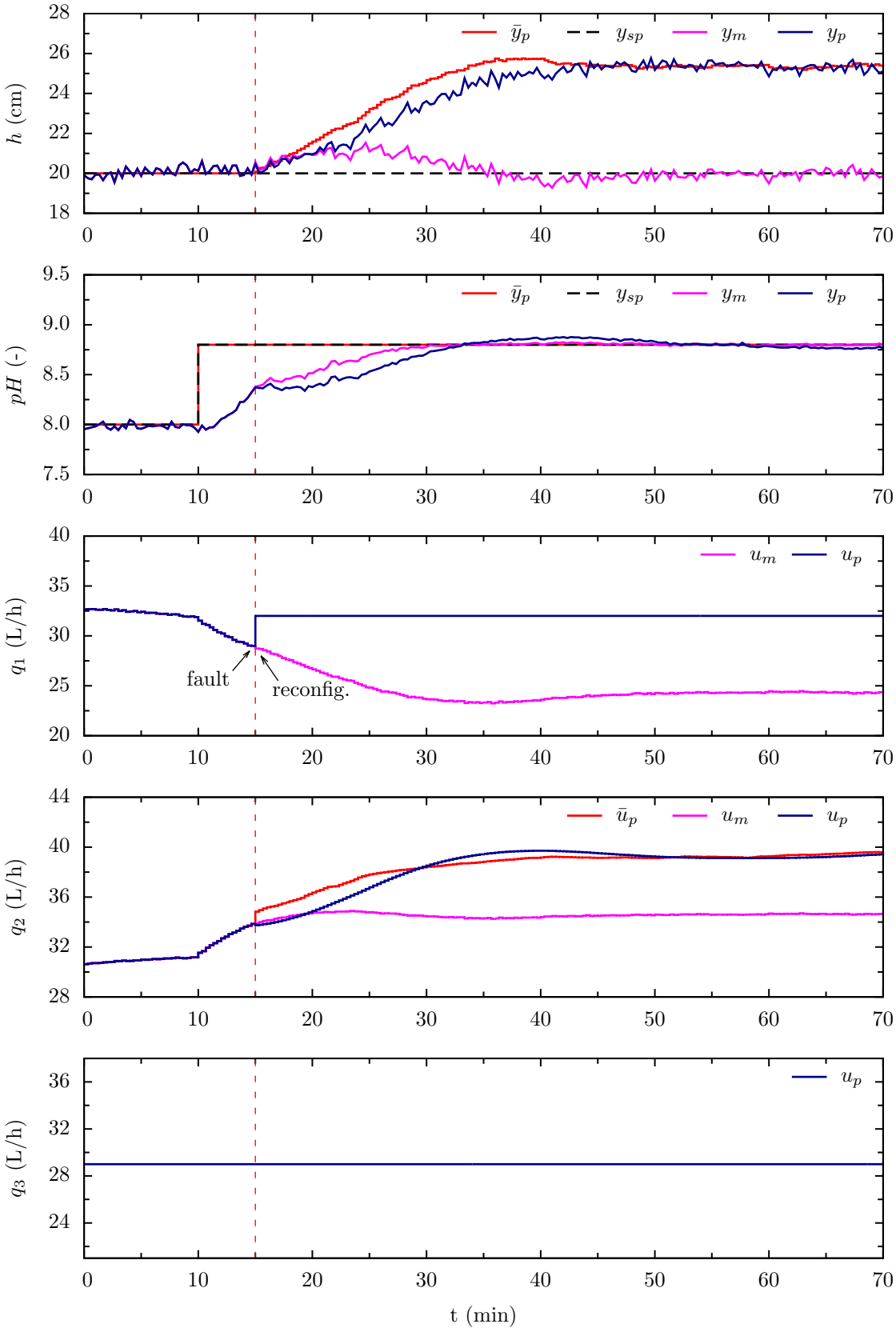

Figure 25 – Scenario 1b: the actuator related to q_1 became stuck at 32 L/h and control reconfiguration occurred immediately.

Scenario 1c

Again, the system is subjected to the same fault described in scenario 1a, but in this case the control reconfiguration is performed 3 minutes after the fault occurrence. In fact, the delay between the fault occurrence and control reconfiguration is expected in real situations in which fault detection is only performed when online indexes are out of stipulated thresholds or when process dynamics begins to be compromised depending on the detection method used. In addition, the diagnostic step can take several sampling times to determine a reliable post-fault model.

In this case, the results shows the pH value tended to present offset after the occurrence of fault, but this trend was interrupted after the control reconfiguration and the offset was removed. The targets for the level were calculated according to the new steady state. Since the fault hiding depends on the availability of the post-fault model, it only occurred after the control reconfiguration. Even though, the nominal plant output signals were driven to their setpoints and the nominal controller output did not saturate in q_2 lower limit.

Since the same fault occurred in scenarios 1b and 1c, it was expected that the steady state deviation of the reactor level from its setpoint would be similar in both cases. However, the use of tap water to prepare the solutions can generate situations with different levels of buffering agents in each experiment, leading to distinct steady states.

Scenario 2a

In this scenario, a fault was inserted into the base solution stream at time $t = 15$ minutes, blocking the actuator at its upper limit. Then, the flow rate of this stream was maintained at 40 L/h, disregarding the output signal from the nominal controller for this actuator. Figure 27 shows the process response a pH setpoint change from 8 to 8.8 and after the occurrence of fault in a case where the control loop is not reconfigured and remains with the actuation of the nominal controller.

As a result of the fault in the system both level and pH presented steady state deviation from the desired setpoint. If the control loops were not decoupled, only the pH would present offset since it was the manipulated variable used to control the pH that was blocked. The nominal controller output related to q_2 has saturated at its lower limit. This is because this signal has no effect on the plant which causes the pH error to remain and then makes controller to indefinitely increase the signal until it reaches saturation.

Scenario 2b

The same fault of the previous scenario was inserted in the system, but in this case the control reconfiguration was performed 3 minutes after the fault occurrence. Figure 28 shows the data collected from the plant. The system had the same tendency to present

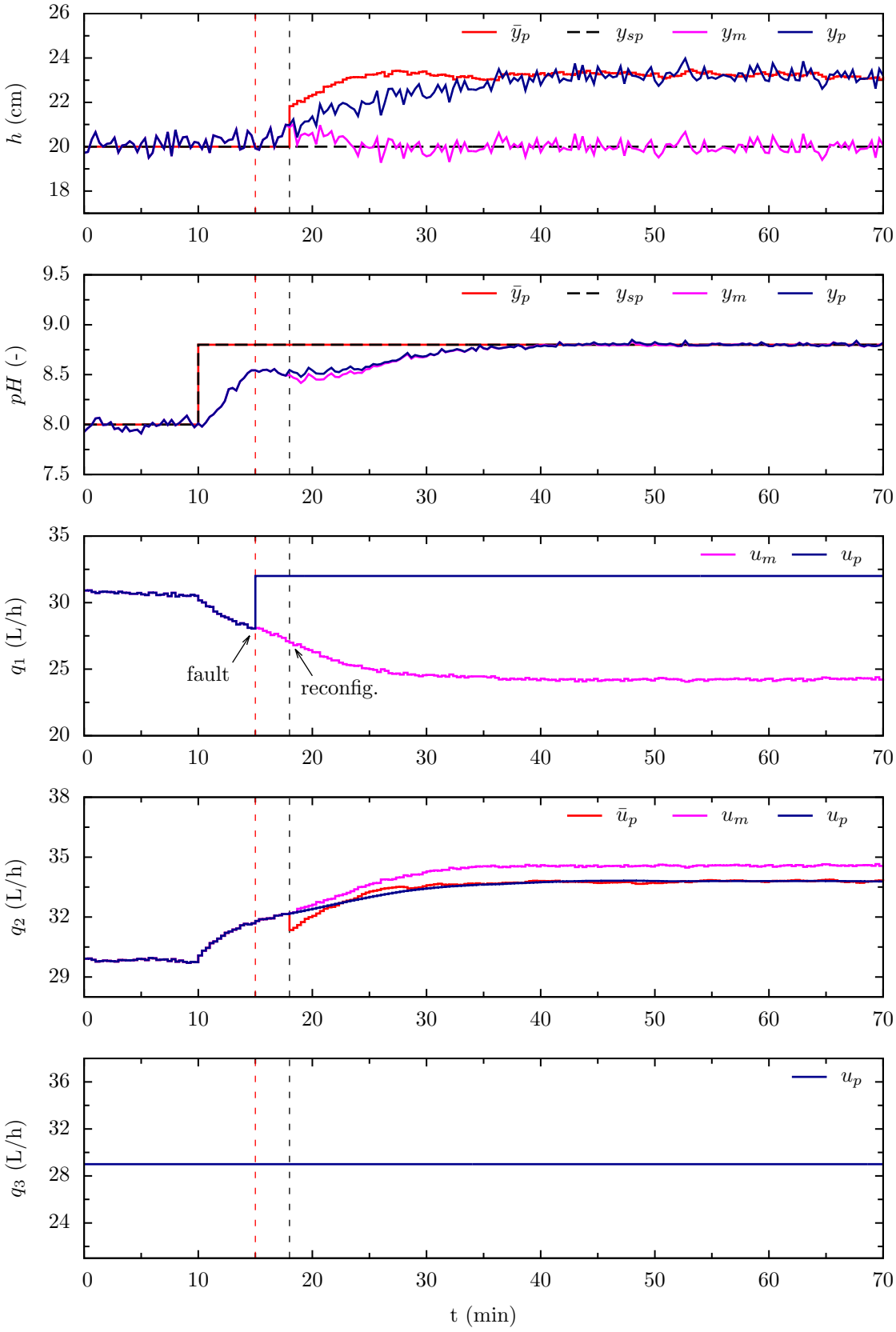

Figure 26 – Scenario 1c: the actuator related to q_1 became stuck at 32 L/h and control reconfiguration was performed 3 minutes after the fault occurrence.

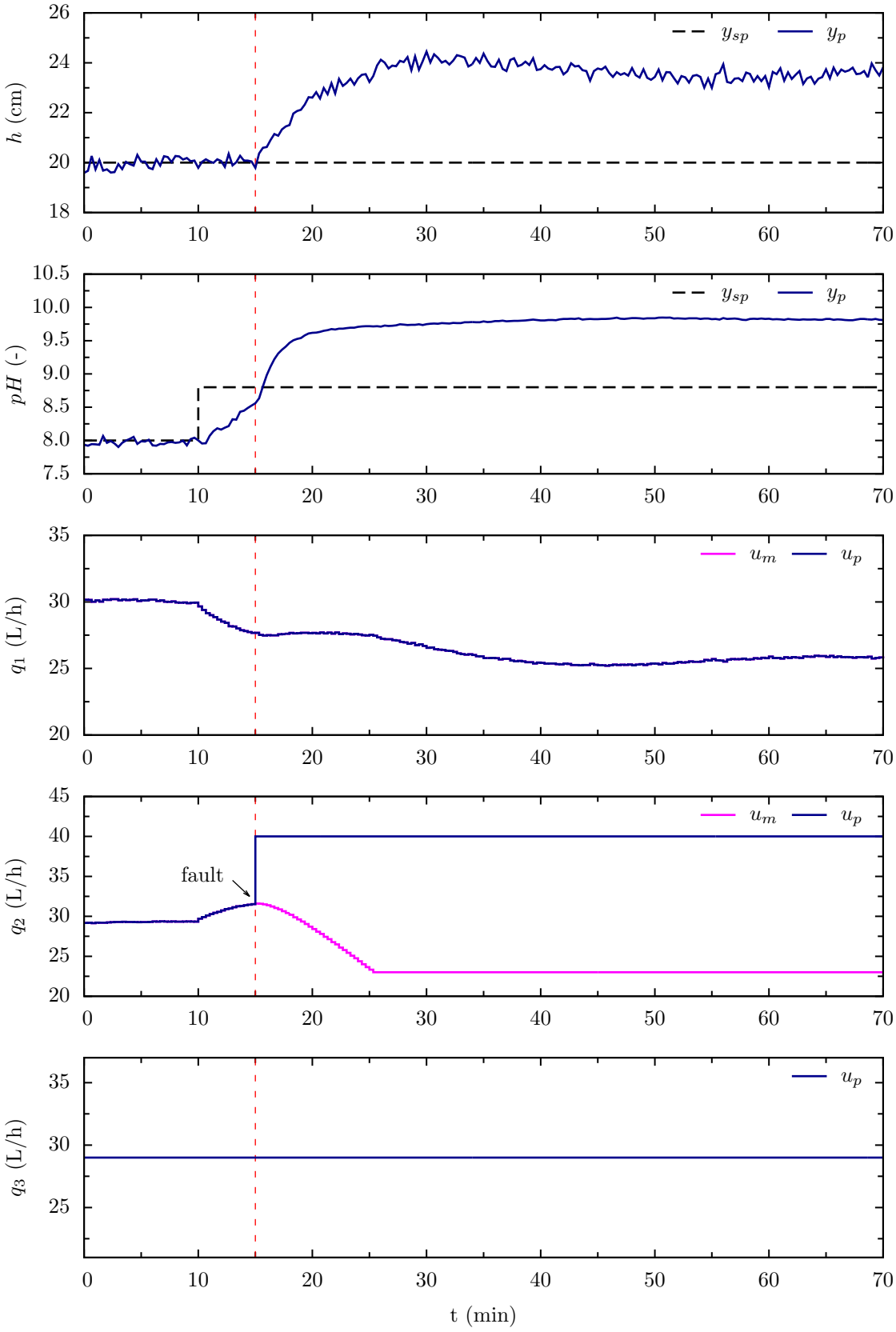

Figure 27 – Scenario 2a: the actuator related to q_2 became stuck at its upper limit of 40 L/h.

offset, however new targets were calculated after the control reconfiguration and the pH stabilized around 9 instead of 10, as in the previous scenario.

It is interesting to note that, although the control reconfiguration has occurred, both controlled variables remained with steady state error from the setpoint. This happened because it was not possible to transfer all the deviation to the level, since its target reached its upper limit of security, that is, 26 cm to prevent the reactor from overflowing. Thus, the pH was prioritized to a certain extent and, for safety reasons, the acid stream flow rate could not be increased to bring the pH back to 8.8.

Just after the control reconfiguration at time $t = 18$ min, the pH target was initially computed around 10, but it was continuously updated as new estimates of disturbances are made. This occurs because between $t = 15$ min and $t = 18$ min the estimate of states and disturbances were based on a wrong model and this estimated values were used as the initial condition for estimating states and disturbance based on the post-fault model from $t = 18$ min.

Scenario 2c

It is intended to evaluate now the case in which the fault diagnosis system fails in detecting the right position of the stuck actuator, thus generating a wrong post-fault model. Thus, the same fault as the previous scenario is inserted, but an identification error is assumed so the actuator became stuck at 29 L/h. The results for this scenario are shown in Figure 29.

The target of the level initially tends to present a negative deviation from the setpoint, but it was updated until reaching the maximum allowed limit of 26 cm. This is explained by the fact that the observer makes estimates of states and disturbances using the nominal model until $t = 18$ min and, from that moment on, it uses the post-fault model with the wrong stuck position. In fact, if the true stuck position of q_2 were 29 L/h, a lower acid flow rate would be required to reduce the pH to its setpoint of 8.8, causing the level to stabilize at a lower value.

Another interesting thing to note is that the pH target differed from the setpoint only when the target of the level reached its upper limit with the estimate of disturbances. In addition, the steady state pH deviation was similar to the previous scenario, as expected.

Finally, the output signals of the nominal plant were driven to their setpoints and the fault was hidden.

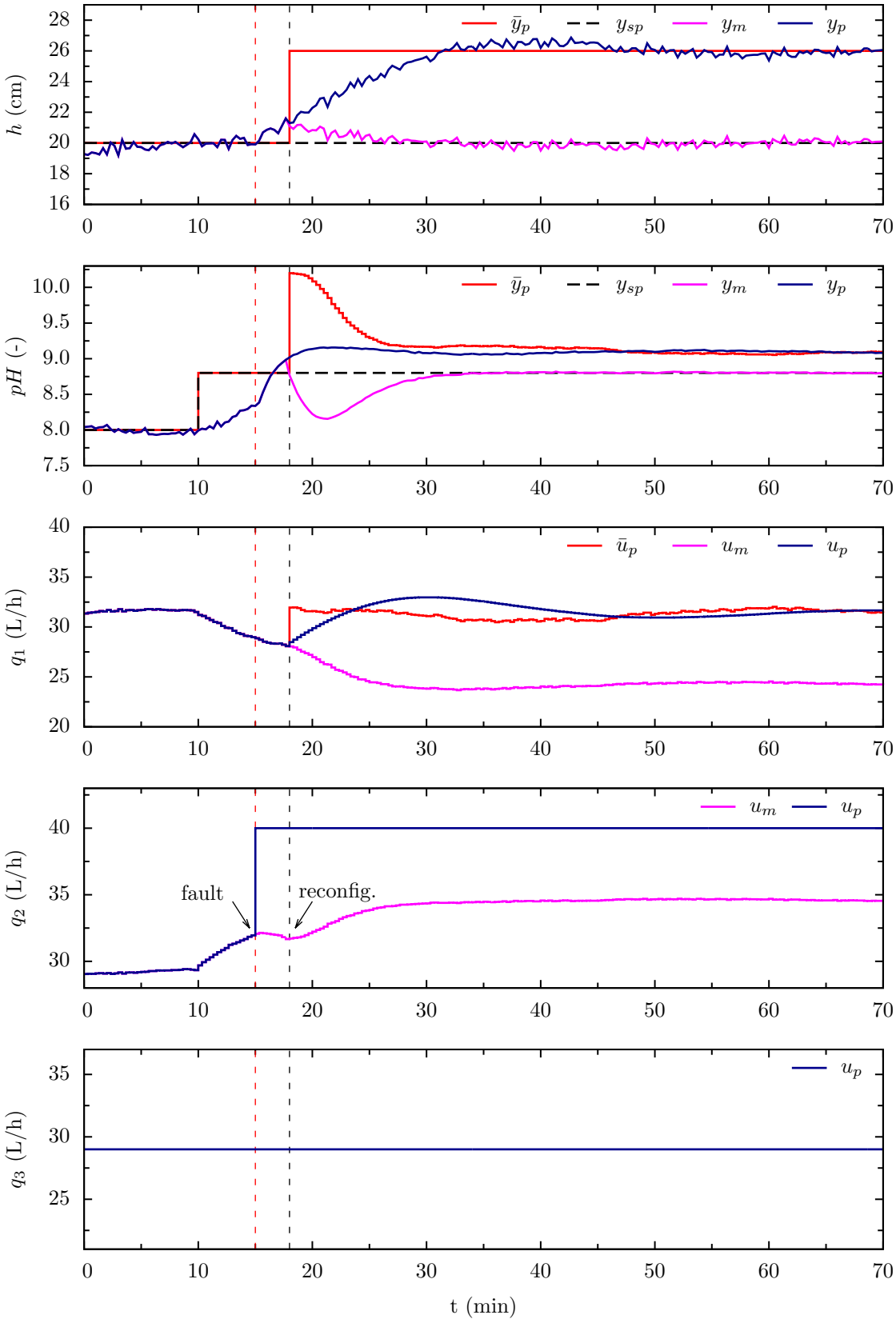

Figure 28 – Scenario 2b: the actuator related to q_2 became stuck at 40 L/h and control reconfiguration was performed 3 minutes after the fault occurrence.

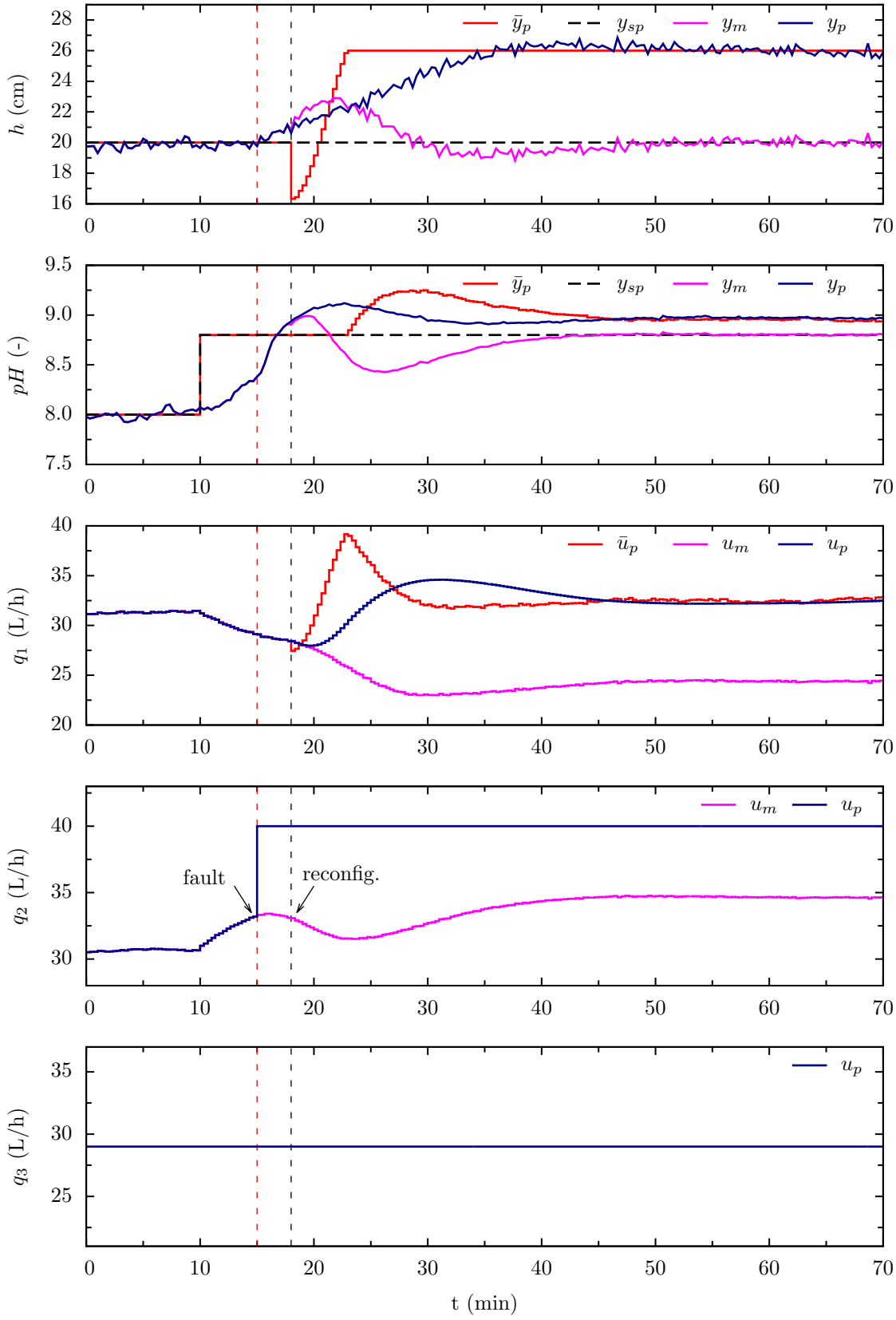

Figure 29 – Scenario 2c: the actuator related to q_2 became stuck at 40 L/h, control reconfiguration was performed 3 minutes after the fault occurrence and the blocked position was misdetected at 29 L/h.

CHAPTER 5

Conclusions and future research directions

A moving horizon virtual actuator with target tracking was proposed as an extension of recent works (COSTA et al., 2013; COSTA, 2014; COSTA et al., 2015). This approach is based on the two-layer MPC framework used in most industrial applications and that comprises basically three elements: a state and disturbance estimator, an external optimizer to perform target calculation and the moving horizon-based controller itself.

In addition, the proposal contemplates constraints in the deviation between nominal and faulty plant behaviors. It can be argued that one way to prioritize the reference of a controlled variable so that it approaches its setpoint value is the configuration of the weights in the Q_t matrix. However, the use of constraints in y_Δ allows to meet process requirements, such as security criteria or product specifications. In addition, the use of an augmented model in the estimation of states and disturbances allows the steady state targets to not be compromised by modeling errors, especially in the post-fault model provided by the FDD system.

In general, cases of actuator loss without the availability of redundancies in the system represent a greater challenge in terms of fault accommodation. As mentioned, insufficient degrees of freedom in the system can affect the controllability of the plant, making setpoints unreachable in some cases. Therefore, in order to investigate these situations, we considered scenarios in which there are no redundancies to be triggered after the occurrence of faults.

The proposed reconfiguration block was applied to a quadruple-tank process. Simulation results demonstrated the advantage of using the zone control in fault situations, allowing the stabilization of the process in more interesting operating points in comparison to the targets calculation without the approach by zones. In fact, it has been shown that

simply penalizing the distance between the setpoints of controlled variables can lead to a steady state condition that violates the specified ranges.

In addition, experiments were conducted in a pH neutralization plant. In this step, we aimed to demonstrate situations in which there were errors in the fault model, by the plant nonlinear dynamics as well as by the wrong identification of the actuators stuck position. The results demonstrated the fault hiding from the point of view of the nominal controller, allowing to drive the reconstructed signals y_m to the setpoints. In addition, the targets were continuously updated according to new disturbance estimates.

Future work may contemplate a combined design of observer and disturbance model (PANNOCCHIA; BEMPORAD, 2007) before the reconfiguration of the control loop as well as the extension of the virtual actuator using the multi-model approach for robust MPC, as proposed by Badgwell (1997), which allows to consider uncertainties in the process model, improving control performance.

References

BADGWELL, T. A. Robust model predictive control of stable linear systems. *International Journal of Control*, Taylor & Francis Group, v. 68, n. 4, p. 797–818, 1997.

BITMEAD, R. R.; GEVERS, M.; WERTZ, V. Adaptive optimal control : the thinking man's GPC. *Prentice Hall international series in systems and control engineering*, v. 29, n. 3, p. xi, 244 p., 1990.

BLANKE, M. et al. *Diagnosis and Fault-Tolerant Control*. 2nd. ed. Berlin Heidelberg: Springer-Verlag, 2006. 695 p.

CAMACHO, E. F.; BORDONS, C. *Model Predictive Control*. London: Springer-Verlag, 2007. (Advanced Textbooks in Control and Signal Processing).

COSTA, T. V. *Estudo e Implementação de Estruturas de Controle Reconfigurável aplicado a Processos Químicos*. Tese (PhD Thesis) — University of Campinas, 2014.

COSTA, T. V. et al. Control reconfiguration of chemical processes subjected to actuator faults: A moving horizon approach. In: *Proceedings of The 14th IASTED International Symposium on Intelligent Systems and Control (ISC 2013)*. Marina del Rey, USA: ACTA Press, 2013. p. 377–384.

COSTA, T. V. et al. Controle Tolerante a Falhas baseado em Atuadores Virtuais com Horizonte Móvel. In: *Proceedings of the XX Brazilian Congress of Chemical Engineering*. Florianópolis: Blucher Chemical Engineering Proceedings, 2015. p. 11254–11261.

CUTLER, C. R.; RAMAKER, B. L. Dynamic matrix control – a computer control algorithm. In: *Joint Automatic Control Conference*. [S.l.: s.n.], 1980. v. 17.

GAO, Z.; DING, S. X.; CECATI, C. Real-Time Fault Diagnosis and Fault-Tolerant Control. *IEEE Transactions on Industrial Electronics*, v. 62, n. 6, p. 3752–3756, 2015.

GARCIA, C. E.; MORSHEDI, A. Quadratic Programming Solution of Dynamic Matrix Control (QDMC). *Chemical Engineering Communications*, Taylor & Francis Group, v. 46, n. 1-3, p. 73–87, 1986.

GUSTAFSSON, T. K. et al. Modeling of pH for control. *Industrial and Engineering Chemistry Research*, v. 34, n. 3, p. 820–827, 1995.

GUSTAFSSON, T. K.; WALLER, K. V. Dynamic modeling and reaction invariant control of pH. *Chemical Engineering Science*, v. 38, n. 3, p. 389–398, 1983.

HENSON, M. a. et al. Adaptive nonlinear control of a ph neutralization process. *IEEE Transactions on Control Systems Technology*, v. 2, n. 3, p. 169–182, 1994.

HERMANSSON, A.; SYAFIIE, S. Model predictive control of pH neutralization processes: A review. *Control Engineering Practice*, v. 45, p. 98–109, 2015.

JOHANSSON, K. The quadruple-tank process: a multivariable laboratory process with an adjustable zero. *IEEE Transactions on Control Systems Technology*, v. 8, n. 3, p. 456–465, 2000.

LUNZE, J.; STEFFEN, T. Control Reconfiguration After Actuator Failures Using Disturbance Decoupling Methods. *IEEE Transactions on Automatic Control*, v. 51, n. 10, p. 1590–1601, 2006.

MACIEJOWSKI, J. *Predictive control: with constraints*. London: Pearson Education, 2002. 331 p.

MEADOWS, E. S. et al. Receding horizon control and discontinuous state feedback stabilization. *International Journal of Control*, Taylor & Francis Group, v. 62, n. 5, p. 1217–1229, 1995.

MONTOYA, R. J. et al. Restructurable Controls. In: *Proceedings of a workshop held at NASA Langley Research Center*. Hampton, Virginia: NASA Conference Publication, 1983.

MUSKE, K. R.; BADGWELL, T. A. Disturbance modeling for offset-free linear model predictive control. *Journal of Process Control*, v. 12, n. 5, p. 617–632, 2002.

MUSKE, K. R.; RAWLINGS, J. B. Model predictive control with linear models. *AIChE Journal*, v. 39, n. 2, p. 262–287, 1993.

PANNOCCHIA, G.; BEMPORAD, A. Combined design of disturbance model and observer for offset-free model predictive control. *IEEE Transactions on Automatic Control*, v. 52, n. 6, p. 1048–1053, 2007.

PANNOCCHIA, G.; RAWLINGS, J. B. Disturbance models for offset-free model-predictive control. *AIChE Journal*, v. 49, n. 2, p. 426–437, 2003.

PATTON, R. Fault-tolerant control systems: The 1997 situation. In: *Proceedings of the IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes*. [S.l.]: IFAC, 1997. p. 1033–1054.

QIN, S.; BADGWELL, T. A. A survey of industrial model predictive control technology. *Control Engineering Practice*, v. 11, n. 7, p. 733–764, 2003.

RAWLINGS, J. Tutorial: model predictive control technology. In: *Proceedings of the 1999 American Control Conference*. San Diego, California, USA: IEEE, 1999. v. 1, p. 662–676.

RAWLINGS, J. Tutorial overview of model predictive control. *IEEE Control Systems Magazine*, v. 20, n. 3, p. 38–52, 2000.

RAWLINGS, J.; MUSKE, K. The stability of constrained receding horizon control. *IEEE Transactions on Automatic Control*, IEEE, v. 38, n. 10, p. 1512–1516, 1993.

RAWLINGS, J. B.; MAYNE, D. Q. *Model Predictive Control: Theory and Design*. Madison, WI: Nob Hill Publishing, LCC, 2009.

RICHALET, J. et al. Model predictive heuristic control. *Automatica*, v. 14, n. 5, p. 413–428, 1978.

RICHTER, J.; LUNZE, J.; SCHLAGE, T. Control reconfiguration of a thermofluid process by means of a virtual actuator. *IET Control Theory & Applications*, IET, v. 1, n. 6, p. 1606–1620, 2007.

RICHTER, J. H. *Reconfigurable Control of Nonlinear Dynamical Systems: A Fault-Hiding Approach*. Berlin: Springer, 2011.

ROTONDO, D.; NEJJARI, F.; PUIG, V. A virtual actuator and sensor approach for fault tolerant control of LPV systems. *Journal of Process Control*, v. 24, n. 3, p. 203–222, 2014.

ROTONDO, D. et al. A virtual actuator approach for the fault tolerant control of unstable linear systems subject to actuator saturation and fault isolation delay. *Annual Reviews in Control*, v. 39, p. 68–80, 2015.

SIMON, D. *Optimal state estimation: Kalman, H-infinity, and nonlinear approaches*. Hoboken, New Jersey: John Wiley & Sons, 2006.

STEFFEN, T. *Control Reconfiguration of Dynamical Systems: Linear Approaches and Structural Tests*. Berlin Heidelberg: Springer-Verlag, 2005. v. 320. 183–290 p.

TABATABAEIPOUR, S. M.; STOUSTRUP, J.; BAK, T. Fault-tolerant control of discrete-time LPV systems using virtual actuators and sensors. *International Journal of Robust and Nonlinear Control*, v. 25, n. 5, p. 707–734, 2015.

WANG, L. A Tutorial on Model Predictive Control: Using a Linear Velocity-Form Model. *Developments in Chemical Engineering and Mineral Processing*, v. 12, n. 5-6, p. 573–614, 2004.

WANG, L. *Model Predictive Control System Design and Implementation Using MATLAB*. London: Springer London, 2009. (Advances in Industrial Control).

YANG, Q.; GE, S. S.; SUN, Y. Adaptive actuator fault tolerant control for uncertain nonlinear systems with multiple actuators. *Automatica*, v. 60, p. 92–99, 2015.

YING, C. M.; VOORAKARANAM, S.; JOSEPH, B. Performance and Stability Analysis of LP-MPC and QP-MPC Cascade Control Systems. *AIChE Journal*, v. 45, n. 7, p. 1521–1534, 1999.

YU, X.; JIANG, J. A survey of fault-tolerant controllers based on safety-related issues. *Annual Reviews in Control*, v. 39, p. 46–57, 2015.

ZHANG, R. et al. State space model predictive fault-tolerant control for batch processes with partial actuator failure. *Journal of Process Control*, v. 24, n. 5, p. 613–620, 2014.

ZHANG, Y.; JIANG, J. Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control*, v. 32, n. 2, p. 229–252, 2008.