COPPE
Instituto Alberto Luiz Coimbra de
Pós-Graduação e Pesquisa de Engenharia
UFRJ

HEALTH-AWARE CONTROL AND MODEL-BASED PROGNOSTICS OF A SUBSEA OIL AND GAS SEPARATION SYSTEM

Lucas Ferreira Bernardino

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Química, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Química.

Orientadores: Argimiro Resende Secchi
Maurício Bezerra de Souza Jr.

Rio de Janeiro
Agosto de 2019

HEALTH-AWARE CONTROL AND MODEL-BASED PROGNOSTICS OF A SUBSEA OIL AND GAS SEPARATION SYSTEM

Lucas Ferreira Bernardino

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA QUÍMICA.

Examinada por:

Prof. Argimiro Resende Secchi, D.Sc.

Prof. Maurício Bezerra de Souza Júnior, D.Sc.

Prof. Bruno Didier Olivier Capron, D.Sc.

Prof. Eduardo Antônio Barros da Silva, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
AGOSTO DE 2019

Bernardino, Lucas Ferreira

Health-aware control and model-based prognostics of a subsea oil and gas separation system/Lucas Ferreira Bernardino. – Rio de Janeiro: UFRJ/COPPE, 2019.

XX, 96 p.: il.; 29, 7cm.

Orientadores: Argimiro Resende Secchi
Maurício Bezerra de Souza Jr.

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Química, 2019.

Referências Bibliográficas: p. 72 – 77.

1. Controle preditivo. 2. Confiabilidade de equipamentos. 3. Tempo de vida útil restante. 4. Inferência estatística. 5. Processamento submarino. I. Secchi, Argimiro Resende *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Química. III. Título.

Agradecimentos

À minha família, que esteve ao meu lado ao longo do desenvolvimento deste trabalho, mesmo enquanto eu precisava me isolar para me concentrar na pesquisa.

À todos que fizeram parte de minha trajetória desde a graduação: toda a bagagem que vocês me deram continua sendo minha referência. Levo um pedacinho de cada um comigo.

Ao orientador que me acompanhou desde a graduação, professor Argimiro. Neste processo do mestrado, tive a oportunidade de aprender contigo sobre os mais variados temas acadêmicos. E quem te conhece por qualquer dos seus alunos vai saber da sua excelência em ensino e pesquisa. Mas além disso, com a sua orientação aprendi coisas que não cabem num currículo. O senhor é simplesmente inesquecível. Espero conseguir carregar um quê de "Argimiro" pra onde eu for.

Ao professor Maurício, que tive a oportunidade de conhecer no PEQ, e que me propôs esse tema inovador, confiando no meu potencial para desenvolvê-lo. Além das contribuições a este trabalho, sou grato por seu entusiasmo, que me manteve motivado e confiante ao longo dessa trajetória.

Aos amigos da COPPE, que fizeram as dificuldades se tornarem em união, e dessa união surgiram momentos maravilhosos. Em especial, quero agradecer aos xodós da turma de 2018: Danillo, Júlia, Pedro, Renato e Sâmia. Desde aquela cervejinha básica pra desestressar até a troca de figurinhas sobre nossos objetos de pesquisa, vocês foram essenciais.

Aos integrantes do LMSCP (G130) e adjacências: entrei e saí sendo o caçula do grupo, e vocês, como bons irmãos mais velhos, me acolheram e me ensinaram. Muito obrigado pelo ambiente maravilhoso do ponto de vista técnico e emocional. É muito amor envolvido.

Ao Coral da COPPE, comandado por Danielly Souza, agradeço por me transportarem do meu estresse cotidiano para um lugar onde todos se harmonizam e se deixam levar pela música. Agradeço também por manterem viva a minha paixão pelo canto!

A todo o quadro de funcionários do PEQ. Vocês fazem a diferença.

Ao financiamento público da pesquisa brasileira. Em especial, à CAPES, que financiou este trabalho.

Acknowledgements

My time as an exchange student in NTNU (Trondheim) was absolutely incredible, both from the scientific and personal experiences. First and foremost I would like to thank Carol and Rodrigo, my flatmates and fellow academics from UFRJ. From the unofficial orientation on particle filters to some non-Western dramas and unconventional musical tastes, this has been an amazing time that I had with you guys.

I would like to thank Professor Anne Barros, for her supervision during my exchange period. Her contribution to this work was invaluable, and her kindness made our collaboration to be very pleasant.

I am also very grateful for the warm welcoming of the Department of Chemical Engineering of NTNU. The department environment quickly made me feel at home, with the lunchtime conversations, group meetings with cake and breaks for enjoying the sun. Professors Johannes Jäschke and Sigurd Skogestad made sure that I was integrated with the group in the academic activities.

Special thanks go to my Brazilian homies Allyne, Otávio and José, who gave me several pointers on the Norwegian lifestyle so I could fit in more easily.

This work was partly funded by the International Partnerships for Excellence Education, Research and Innovation (INTPART) program during my stay as an exchange student in NTNU (Trondheim).

"O correr da vida embrulha tudo, a vida é assim: esquenta e esfria, aperta e daí afrouxa, sossega e depois desinquieta. O que ela quer da gente é coragem."

(João Guimarães Rosa)

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

CONTROLE CONSCIENTE DA SAÚDE E PROGNÓSTICO BASEADO EM MODELO DE UM SISTEMA DE SEPARAÇÃO SUBMARINA DE ÓLEO E GÁS

Lucas Ferreira Bernardino

Agosto/2019

Orientadores: Argimiro Resende Secchi
Maurício Bezerra de Souza Jr.

Programa: Engenharia Química

A exploração de campos com alto conteúdo de CO_2 é desafiadora, devido ao baixo valor agregado e ao impacto ambiental associado a este componente. Nesse contexto, um uso importante para o CO_2 gerado é sua reinjeção no reservatório, e a separação submarina de CO_2 resulta em um processamento mais eficiente. O prognóstico é uma atividade chave nesse processo, devido à necessidade de minimizar intervenções nos equipamentos. O objetivo principal deste trabalho é investigar aspectos do controle consciente da saúde de um sistema de separação submarina de CO_2. O modelo do sistema considerado, que consiste em um sistema de equações algébrico-diferenciais, foi adaptado da literatura. Todas as simulações foram realizadas utilizando bibliotecas desenvolvidas pelo autor, baseadas nos métodos das bibliotecas Scipy e Assimulo. Um estado estacionário de referência foi obtido para as condições de projeto, e as funções de transferência em tempo contínuo foram identificadas a partir dos dados de resposta ao degrau. Os modelos identificados foram utilizados em um controlador preditivo, e simulações em malha fechada foram realizadas para avaliar a sintonia do controlador. No sentido de prognóstico, um modelo estocástico de degradação de bomba multifásica, sensível às condições de operação do processo, foi proposto, e um filtro de partículas foi implementado para estimação online do estado de degradação e predição do tempo de vida útil. Por fim, um controlador consciente da saúde foi projetado, e foram investigadas algumas dificuldades na combinação dos objetivos de extensão de vida útil e de controle. Os resultados obtidos indicam que o tratamento do problema de controle consciente da saúde sob a teoria de otimização multiobjetivo pode trazer resultados mais satisfatórios, assim como abordar a extensão de vida útil em uma camada de otimização.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

HEALTH-AWARE CONTROL AND MODEL-BASED PROGNOSTICS OF A SUBSEA OIL AND GAS SEPARATION SYSTEM

Lucas Ferreira Bernardino

August/2019

Advisors: Argimiro Resende Secchi
Maurício Bezerra de Souza Jr.

Department: Chemical Engineering

The exploitation of fields with high CO_2 content is challenging, due to the low economic value and the environmental impact associated with this component. In this context, a significant use for the generated CO_2 is its reinjection into the reservoir, and subsea CO_2 separation allows for more efficient processing. Prognostics is a key activity in this process, due to the necessity of minimizing intervention in equipment. The main objective of this work is to investigate the health-aware control of a subsea CO_2 separation system. The considered system model, which consists of a differential-algebraic equation system, was adapted from the literature. All simulations were performed using libraries developed by the author, based on methods from the Scipy and Assimulo libraries. A reference steady state was obtained for the design conditions, and continuous-time transfer functions were identified from step response data. The identified models were used in a predictive controller, and closed-loop simulations were performed to evaluate the controller tuning. In the sense of prognostics, a stochastic model of multiphase pump degradation, sensitive to the process operating condition, was proposed, and a particle filter was implemented to perform online degradation state estimation and remaining useful lifetime prediction. At last, a health-aware controller was designed, and some difficulties in combining reference tracking and lifetime extension objectives were investigated. The obtained results indicate that dealing with the health-aware control problem through the multiobjective optimization theory and addressing the useful lifetime extension in an optimization layer may result in more satisfactory results.

Contents

List of Figures

List of Tables

List of Symbols

$A_{v,b}$	Bottom stream cross-section area, p. 19
$A_{v,in}$	Inlet stream cross-section area, p. 19
$A_{v,t}$	Top stream cross-section area, p. 18
E_c	Heat exchanger cold side energy holdup, as in SOUZA (2018), p. 16
E_h	Heat exchanger hot side energy holdup, as in SOUZA (2018), p. 16
$E_{c,k}$	Heat exchanger cold side k-th stage energy holdup, p. 17
$E_{h,k}$	Heat exchanger hot side k-th stage energy holdup, p. 17
F_c	Heat exchanger cold side molar flow, p. 16
F_h	Heat exchanger hot side molar flow, p. 16
F_{in}	Flash drum inlet molar flow, p. 18
$F_{l,tot}$	Flash drum bottom outlet molar flow, p. 19
F_l	Flash drum liquid outlet molar flow, p. 18
F_v	Flash drum top outlet molar flow, p. 18
F_w	Flash drum aqueous outlet molar flow, p. 18
$H(\cdot)$	Molar enthalpy function, p. 18
H_F	Flash drum inlet molar enthalpy, p. 18
$H_{c,i}$	Heat exchanger cold side inlet molar enthalpy, p. 16
$H_{c,o}$	Heat exchanger cold side outlet molar enthalpy, p. 16
$H_{h,i}$	Heat exchanger hot side inlet molar enthalpy, p. 16

$H_{h,o}$	Heat exchanger hot side outlet molar enthalpy, p. 16
H_l	Flash drum liquid phase molar enthalpy, p. 18
H_v	Flash drum top phase molar enthalpy, p. 18
H_w	Flash drum aqueous phase molar enthalpy, p. 18
$K_{v,b}$	Bottom valve constant, p. 19
$K_{v,in}$	Inlet valve constant, p. 19
$K_{v,t}$	Top valve constant, p. 19
$MM(\cdot)$	Molecular weight function, p. 18
N_c	MPC control horizon, p. 25
N_p	MPC prediction horizon, p. 25
N_i	Flash drum i-th component moles number, p. 14
N_{part}	Number of particles in particle filter, p. 26
N_{tot}	Flash drum total moles number, p. 18
$P_{BL,B}$	Bottoms battery limit pressure, p. 14
$P_{BL,T}$	Top battery limit pressure, p. 14
$P_{c,i}$	Heat exchanger cold side inlet pressure, p. 14
P_d	Multiphase pump discharge pressure, p. 14
P_{flash}	Flash drum pressure, p. 14
$P_{h,i}$	Heat exchanger hot side inlet pressure, p. 14
P_{res}	Reservoir pressure, p. 14
Q_{aqu}	Flash drum heating rate, p. 18
Q_{resf}	Condenser cooling rate, p. 18
T_1	Heat exchanger cold side inlet temperature, p. 14
T_2	Condenser outlet temperature, p. 14
T_{bot}	Bottoms valve outlet temperature, p. 14

$T_{c,k}$	Heat exchanger cold side k-th stage temperature, p. 17
$T_{c,o}$	Heat exchanger cold side outlet temperature, p. 14
T_d	Multiphase pump discharge temperature, p. 14
T_{flash}	Flash drum temperature, p. 14
$T_{h,i}$	Heat exchanger hot side inlet temperature, p. 14
$T_{h,k}$	Heat exchanger hot side k-th stage temperature, p. 17
$T_{h,o}$	Heat exchanger hot side outlet temperature, p. 14
T_{res}	Reservoir temperature, p. 14
UA	Product between overall heat transfer coefficient and heat transfer surface area of the heat exchanger, p. 16
U^e	Flash drum extensive internal energy, p. 14
V_c	Heat exchanger cold side volume, p. 16
V_h	Heat exchanger hot side volume, p. 16
V_{flash}	Flash drum total volume, p. 18
W	Multiphase pump power, p. 18
ΔP_c	Heat exchanger cold side pressure drop, p. 19
ΔP_h	Heat exchanger hot side pressure drop, p. 19
ΔT_{lm}	Heat exchanger logarithmic mean of temperature difference, p. 16
Δf	Multiphase pump degradation rate parameter, p. 20
$\bar{V}(\cdot)$	Molar volume function, p. 18
$\bar{V}_d$	Multiphase pump discharge molar volume, p. 18
$\bar{V}_s$	Multiphase pump suction molar volume, p. 18
$\bar{V}_l$	Flash drum liquid phase molar volume, p. 18
$\bar{V}_v$	Flash drum top phase molar volume, p. 18
$\bar{V}_w$	Flash drum aqueous phase molar volume, p. 18

β_α	Flash drum liquid phase molar fraction, p. 18
β_γ	Flash drum aqueous phase molar fraction, p. 18
β_ν	Flash drum top phase molar fraction, p. 18
$\boldsymbol{\alpha}$	Flash drum liquid phase composition vector, p. 18
$\boldsymbol{\gamma}$	Flash drum aqueous phase composition vector, p. 18
$\boldsymbol{\nu}$	Flash drum top phase composition vector, p. 18
$\boldsymbol{l}$	Flash drum bottom outlet composition vector, p. 18
$\boldsymbol{u}$	Dynamic system input vector, p. 18
$\boldsymbol{x}$	Dynamic system state vector, p. 18
$\boldsymbol{y}$	Dynamic system output vector, p. 19
$\boldsymbol{z_{res}}$	Reservoir composition vector, p. 14
η	Multiphase pump degradation observation, p. 20
λ	Multiphase pump degradation state, p. 20
λ^{lim}	Degradation state threshold value, p. 28
$\rho_m(\cdot)$	Mass density function, p. 19
$\rho_{c,i}$	Heat exchanger cold side inlet molar density, p. 16
$\rho_{c,k}$	Heat exchanger cold side k-th stage molar density, p. 17
$\rho_{c,o}$	Heat exchanger cold side outlet molar density, p. 16
$\rho_{h,i}$	Heat exchanger hot side inlet molar density, p. 16
$\rho_{h,k}$	Heat exchanger hot side k-th stage molar density, p. 17
$\rho_{h,o}$	Heat exchanger hot side outlet molar density, p. 16
h_{flash}	Flash drum liquid volumetric fraction, p. 19
n	Multiphase pump degradation power parameter, p. 20
n_c	Number of components, p. 18
n_{st}	Heat exchanger number of discretized stages, p. 17

q_i MPC i-th controlled variable weight, p. 25

$q_k^{(i)}$ Quantile associated with the k-th evolution of the i-th particle, p. 28

s_i MPC i-th manipulated variable weight, p. 25

w_{HAC} Health-aware control weight, p. 28

$x_{v,b}$ Bottom valve opening, p. 19

$x_{v,t}$ Top valve opening, p. 19

x_v Inlet valve opening, p. 19

List of Abbreviations

AI	Artificial Intelligence, p. 4
AR	Auto-Regressive, p. 4
CBM	Condition-based Maintenance, p. 3
DAE	Differential-Algebraic Equations, p. 11
DMC	Dynamic Matrix Control, p. 10
EKF	Extended Kalman Filter, p. 8
HAC	Health-Aware Control, p. 2
HMM	Hidden Markov Model, p. 5
KF	Kalman Filter, p. 8
LQR	Linear Quadratic Regulator, p. 10
MIMO	multiple-input multiple-output, p. 10
MPC	Model Predictive Control(ler), p. 10
PF	Particle Filter, p. 9
PSE	Process Systems Engineering, p. 1
QP	Quadratic Programming, p. 11
RUL	Remaining Useful Lifetime, p. 2
SIR	Sequential Importance Resampling, p. 9
SIS	Sequential Importance Sampling, p. 9
UKF	Unscented Kalman Filter, p. 8
c.d.f.	cumulative distribution function, p. 28
p.d.f.	probability distribution function, p. 4

Chapter 1

Introduction

1.1 Contextualization

In the context of process systems engineering (PSE), process control and optimization are very important study fields, allowing not only safe and predictable operation, but also maximal improvement of process performance based on metrics of choice. Plant automatization extends control possibilities, and allows operation in places virtually inaccessible to humans, such as subsea environments.

Processes operated in the subsea environment nowadays are mostly related to oil & gas exploitation at deep and ultra-deep waters. Apart from the obvious necessity of reaching those oil reservoirs, subsea technologies are being developed to revitalize mature fields, in which water content is high and production flow rates are dropping (KUCHPIL *et al.*, 2013). More recently, attention has been drawn to subsea CO_2 separation (PASSARELLI, 2017; SOUZA, 2018; SOUZA *et al.*, 2019).

Fields with high CO_2 content represent over 10% of world proven fields. The exploitation of these fields generates major CO_2 quantities, which have low economic value and are an environmental burden, but must be processed in order to recover the main products with required specification. In this context, a significant use for CO_2 is the reinjection into the reservoir, in order to maintain well pressure and thus oil production for longer times (DE MEDEIROS *et al.*, 2018).

Thus, the early separation of CO_2 from oil stream is beneficial, because downstream equipments can be sized to lesser flow rates and produced CO_2 can be readily reinjected into the reservoir. This strategy was addressed by Petrobras. PASSARELLI (2017) proposed a subsea CO_2 separation process that took advantage of mixture thermodynamic properties at well conditions to ensure phase separation and acceptable recoveries. SOUZA (2018) and SOUZA *et al.* (2019) analyzed the feasibility of this process and developed a model inspired on it.

CO_2 reinjection requires the use of a multiphase pump able to promote considerable pressure rise. Therefore, being the equipment under the most operation stress, it is reasonable to assume that process failure is most likely to happen in this equipment, and degradation modeling efforts shall be directed to it.

In the context of subsea processing, reliability analysis and health monitoring play a major role in the sense that process interventions and production stops must be minimized at all costs. Understanding the degradation process of an equipment is fundamental for predicting its maintenance time. Additionally, the comprehension of how the operating point affects degradation patterns is necessary for implementing health-aware process control and optimization.

1.2 Objectives

The general objective of this work is to investigate the main aspects on the development of a Health-Aware Control (HAC) tool, which performs control, prognostics and optimization, applied to a subsea CO_2 separation system model. More specifically, this work aims to assess process control and prognostics, and how they can be combined.

For this goal to be attained the following specific objectives were specified:

- Study the modeling of a subsea CO_2 separation system;
- Identify linear models based on simulated data from the considered system model;
- Apply advanced control techniques to this process, using the identified linear models;
- Propose a degradation model to the multiphase pump, sensitive to operating conditions;
- Design a prognostics tool which predicts the remaining useful lifetime (RUL) of equipment, based on the proposed degradation model;
- Integrate process control and prognostics into an unified automatic framework.

1.3 Document structure

This document is structured as follows. Chapter 2 covers the literature review, highlighting important aspects that support this work. Chapter 3 presents aspects of the models used in this dissertation. Chapter 4 describes the methodology used to obtain the results presented and discussed in Chapter 5. This document is closed with the conclusion and final remarks in Chapter 6.

Chapter 2

Literature review

In this chapter, theoretical aspects and recent developments regarding the main subjects involved in this dissertation are described.

2.1 Prognostics and health monitoring

System reliability is fundamental information in the field of process systems engineering. Casualties are more likely to happen when system parts are deteriorated, and, in this context, reliability analysis aims to quantify rate or probability of casualties, in order to aid in the decision-making process (RAUSAND and HØYLAND, 2004).

Maintenance planning is an example of these decisions. In the early industrial periods, maintenance was made only at the process breakdown ("run-to-failure maintenance"). This behavior often leads to major financial losses, so this kind of maintenance has mostly given way to preventive maintenance, in which parts are repaired or replaced based on events historical data (HENG *et al.*, 2009).

But, as competition in the industries became tighter, the time between repairs became a critical decision variable. If this time interval is too high, chances of a process breakdown rise. If it is too low, maintenance costs become unfeasible. Traditional reliability-based preventive maintenance focused on the equipment populational data, not on specific units with specific operational routines (HENG *et al.*, 2009). To assess these issues, the condition-based maintenance (CBM), a new paradigm of process maintenance, arises: interventions are made only when necessary, and equipment conditions are monitored in order to define if a maintenance is necessary or not (JARDINE *et al.*, 2006).

In the stabilished CBM philosophy, the analyst essentially aims to estimate the remaining useful lifetime (RUL) of equipments. To attain this, health indicators are constructed from available system data, and, with predefined degradation stages and failure thresholds, the indicators are projected into a time horizon (LEI *et al.*, 2018). The RUL of a given equipment is defined as the time instant in which the respective health indicator exceed its failure threshold, and prognostics is defined as the prediction of this RUL given process data (ROEMER *et al.*, 2011).

This definition is well-behaved in deterministic frameworks. But, as the reliability analysis deals with stochastic events, reliability metrics such as RUL are best described by probability distribution functions (p.d.f.) (BANJEVIC, 2009). Therefore, the analyst should be aware that health monitoring information is inherently linked to a confidence level, and inappropriate confidence level selection can lead to false-positive or false-negative results.

There is extensive literature regarding equipment prognostics. LEI *et al.* (2018) published recently a comprehensive review of all technical activities of prognostics, namely: data acquisition; health indicator construction; health stages division and RUL estimation.

Modeling time evolution of health indicators has been done in the literature by different methods. LEI *et al.* (2018) reports that most works accomplish that by using empirical dynamic models, such as auto-regressive (AR) models, with statistical treatment, or artificial intelligence (AI) methods, such as neural networks. Models with physical meaning, which attempt to explain phenomenologically the degradation process, are the minority in this field, which suggests that knowledge about degradation processes is still small.

One of the most used phenomenological models is the Paris law (PARIS and ERDOGAN, 1963; SUN *et al.*, 2014), which describes crack propagation in systems subject to stress. According to it, the crack length growth rate can be modeled by a power law:

$$\frac{da}{dN} = \xi_1 a^{\xi_2} \tag{2.1}$$

where:

- a represents crack length;
- N is the number of loading cycles;
- ξ_1 and ξ_2 are empirical parameters, dependent on the component and the crack geometry.

This model simplicity and generality points again to little knowledge about the physical principles of crack growth. VIRKLER *et al.* (1979) conducted experiments to define the behavior of $\frac{da}{dN}$, which presented considerable variability even in highly controlled environments. For these reasons, data-driven models, whether AI or statistical-based, represent the majority of works in literature, since they are able to use individual past realizations of the health indicator to give information about degradation evolution (HENG *et al.*, 2009; LEI *et al.*, 2018).

Statistical-based prognostics methods are based mainly on empirical knowledge with random coefficients and stochastic process models, fitting measurements to the considered model. Among these models, the most prominent are: Wiener process, Gamma process, Hidden Markov Models (HMMs), and Proportional Hazards (SI *et al.*, 2011).

The Wiener process formulation is extensively used in degradation studies, due to its easy mathematic tractability. Its mathematical representation, in the integral form, is given by (WANG *et al.*, 2014):

$$X(t) = at + \sigma B(t) \tag{2.2}$$

in which X is the health indicator, a and σ are the linear drift and infinitesimal variance parameters respectively, and $B(t)$ represents a standard Brownian motion. It can also be represented in the differential form (given $X(0) = 0$):

$$dX = adt + \sigma dW, \qquad dW \sim \mathcal{N}(0, dt) \tag{2.3}$$

It has been shown that a Wiener process with linear drift presents a RUL p.d.f. corresponding to an inverse Gaussian distribution (CHHIKARA, 1988), and generalizations were made regarding drift behavior (WANG *et al.*, 2014).

Gamma processes are based on the assumption that health indicator increments are independent and have p.d.f. equal to a gamma distribution (VAN NOORTWIJK, 2009). Given a non-decreasing function $v(t)$ with $v(0) = 0$, which gives the shape information, and a constant scale parameter u, the Gamma process has the following simple differential form:

$$dX \sim Gamma(\left.\frac{dv}{dt}\right|_t dt, u) \tag{2.4}$$

The integration of this equation is also simple, due to the properties of Gamma distributions. Given N independent random variables $X_i \sim Gamma(k_i, \theta)$, then the distribution of the sum of these variables is given by:

$$\sum_{i=1}^{N} X_i \sim Gamma(\sum_{i=1}^{N} k_i, \theta) \tag{2.5}$$

As such, integrating Equation 2.4 yields to:

$$X(t) \sim Gamma(v(t), u) \tag{2.6}$$

While Wiener processes describe non-monotonic behaviors, Gamma processes focus on incremental degradation behavior, such as equipment wear (VAN NOORTWIJK, 2009). Nonetheless, both of these formulations need an *a priori* knowledge of the expected behavior as a function of time.

A HMM, essentially, is composed of a discrete number of states, a state transition probability matrix, an observation model and an initial state probability distribution. The transition matrix is responsible for the temporal evolution of the system, and can be interpreted as (BARUAH and CHINNAM, 2005):

$$A_{i,j} = P(X_t = j, |X_{t-1} = i) \tag{2.7}$$

These models predict transition probability with exponential behavior, which may not be accurate in real-life processes, and some variations of HMMs may have an advantage in this matter (DONG and HE, 2007).

The Proportional Hazards Model approach elects covariates among process variables and states that hazard changes proportionally with these covariates (HENG *et al.*, 2009). This approach is mainly empirical, and with low statistical basis, but covariates construction can be made to consider phenomenological and empirical knowledge in the analysis, thus being capable of good description of observations (LEI *et al.*, 2018).

LEI *et al.* (2018) also emphasized that Bayesian filtering algorithms are useful tools for characterizing uncertainty in estimated health status and predicted RUL, being applicable in many prognostics strategies. However, Bayesian filtering needs to be used in tandem with a dynamic model, in order to propagate the current estimated state until system critical state (JOUIN *et al.*, 2016).

Particle filtering applied to prognostics is a recent trend in literature. KOTHAMASU *et al.* (2006) did a prognostics and health monitoring review, and particle filter uses were not reported. Later, JOUIN *et al.* (2016) reported applications of particle filters in prognostics and health monitoring, highlighting its generality in face of non-linear dynamic models and non-Gaussian probability distributions. In this context, some theoretical aspects regarding Bayesian filtering algorithms must be explored.

2.2 Bayesian filtering

Bayesian filtering methods, in the engineering context, are used to estimate states of dynamic systems subject to model and measurement uncertainties. For this, a dynamic model of the system and statistical tools for recursive state estimation are needed. In this context, Bayesian filtering methods use the well-known Bayes theorem to update previous information of system state face to new observations (SIMON, 2006).

For a discrete-time evolution-observation model $(\{X_k\}, \{Y_k\})$, in which X_k is the state vector and Y_k is the observation vector, Bayesian filtering methods are comprised of two steps: time evolution updating, in which current state p.d.f. is projected into the future, generating an *a priori* state p.d.f., and observation updating, in which the *a priori* state p.d.f. is updated to an *a posteriori* state p.d.f. by incorporating observation information (KAIPIO and SOMERSALO, 2005).

The time evolution updating formula can be written as:

$$\pi(x_{k+1}|y_0, \cdots y_k) = \int \pi(x_{k+1}|x_k)\pi(x_k|y_0, \cdots y_k)dx_k \tag{2.8}$$

and the observation updating formula can be written as:

$$\pi(x_{k+1}|y_0, \cdots y_{k+1}) = \frac{\pi(x_{k+1}|y_0, \cdots y_k)\pi(y_{k+1}|x_{k+1})}{\pi(y_{k+1}|y_0, \cdots y_k)} \tag{2.9}$$

in which $\pi(\cdot)$ represents the p.d.f. and:

$$\pi(y_{k+1}|y_0, \cdots y_k) = \int \pi(y_{k+1}|x_{k+1})\pi(x_{k+1}|y_0, \cdots y_k)dx_{k+1} \tag{2.10}$$

In this context, $\pi(x_k|y_0, \cdots y_{k-1})$ represents an *a priori* p.d.f., and $\pi(x_k|y_0, \cdots y_k)$ represents an *a posteriori* p.d.f. with relation to the observation Y_k. Also, $\pi(x_{k+1}|x_k)$ represents the considered dynamic model, known as the Markov transition kernel, and $\pi(y_{k+1}|x_{k+1})$ represents the relationship within the observed vector and the current state, known as the likelihood function. Deterministic processes might also be added to the formulation without loss of validity, to account for process inputs. In this context, they can be treated like parameters, as they do not have associated uncertainties.

Note that, besides the assumption of an evolution–observation model, no restrictions are made on shape of p.d.f.s or process linearity, at the cost of working with the original functions. From these formulas, assuming a linear Gaussian problem, with uncorrelated noise processes, yields to the discrete-time Kalman Filter (KF) (KAIPIO and SOMERSALO, 2005; SIMON, 2006).

The main advantage of the Kalman filter is that updating equations turn into matricial operations. By modeling the process as (SIMON, 2006):

$$\begin{cases} x_{k+1} = F_k x_k + G_k u_k + w_k \\ y_k = H_k x_k + v_k \\ E(w_k w_j^T) = Q_k \delta_{k-j} \\ E(v_k v_j^T) = R_k \delta_{k-j} \\ E(w_k v_j^T) = 0 \end{cases} \tag{2.11}$$

the equations for the discrete-time Kalman filter can be written as:

$$\begin{cases} P_k^- = F_{k-1} P_{k-1}^+ F_{k-1}^T + Q_{k-1} \\ \hat{x}_k^- = F_{k-1} \hat{x}_{k-1}^+ + G_{k-1} u_{k-1} \end{cases} \tag{2.12}$$

for the time evolution update and:

$$\begin{cases} K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \\ P_k^+ = (I - K_k H_k) P_k^- \\ \hat{x}_k^+ = \hat{x}_k^- + K_k (y_k - H_k \hat{x}_k^-) \end{cases} \tag{2.13}$$

for the observation update. In this notation, superscript "$-$" represents *a priori* estimates, superscript "$+$" represents *a posteriori* estimates, and P_k denotes the covariance of the estimate $\hat{x}_k$.

This solution, although simple, relies on strong assumptions regarding process model, so other approaches to the Bayesian filtering problem are needed. The Extended Kalman Filter (EKF) is the closest to the Kalman filter in terms of computational effort. The EKF algorithm is based on local linearization of the dynamic model, which allows for dealing with mildly nonlinear systems, but is still bound to the assumption of a Gaussian distribution (SIMON, 2006).

The Unscented Kalman Filter (UKF) allows for a higher-order approximation for states estimate and covariance based on sampling, using a set of points given by the unscented transformation. Nonlinearity of noises can be incorporated into the formulation by defining an augmented state vector, thus estimating states and noise processes mean and covariance (SIMON, 2006).

The Particle Filter (PF) tackles the Bayesian filtering problem entirely by Monte Carlo sampling, using directly the assumed Markov transition kernel and likelihood function. This algorithm generates a cloud of points, from which state p.d.f.s can be constructed. Some important PF algorithms are the Sequential Importance Sampling (SIS) and Sequential Importance Resampling (SIR) particle filter, which differ on the presence of a resampling step (KAIPIO and SOMERSALO, 2005).

PF algorithms tend to utilize trial distributions, in order to direct state transition calculations to the most important transitions. This strategy is known as importance sampling (SPEEKENBRINK, 2016), which originates the name of this class of filters. These trial distributions are used for state sampling, and these sampled particles are then reweighted using the target distribution. Then, after obtaining the measurement, all particles are reweighted using the likelihood function, as described in Equation 2.9, accruing more knowledge into the particles set.

As particles are reweighted using both the likelihood function and transition kernel, most of them are multiplied by a small probability. This behavior is intensified with the successive multiplications, so that most particles weights tend to zero, and one particle (or a very small group of particles) carries the highest weight, degenerating the distribution which would be reconstructed by the samples. Also, as there is a finite number of particles, there is no guarantee that the surviving particle represents the most representative state among all possible states. This phenomenon is known as weight degeneracy, and it can be observed in most straightforward PF implementations, such as SIS (SPEEKENBRINK, 2016).

To counteract this, a step of resampling is generally used, generating the class of SIR filters. Resampling consists in the obtainment of a representative set of particles, in the sense that it preserves the statistical properties of the population, but with equal weights. This is done generally by replicating particles with high weights at the expense of neglecting the ones with low weights, which can lead to a phenomenon called sample impoverishment. Sample impoverishment is generally important in the construction of smoothing distributions, and there are ways to avoid this issue, see SPEEKENBRINK (2016).

The main advantage of PF over UKF is the possibility of naturally constructing non-Gaussian distributions by sampling. UKF deals with nonlinear distributions by picking representative points in order to correctly estimate states mean and covariance, but the true probability distribution is not directly addressed. In problems such as RUL estimation, it is imperative that p.d.f.s are properly characterized, so that the percentiles are properly estimated and, thus, confidence regions truly reflect available information.

2.3 Model predictive control

In the context of mutivariable control techniques, some control strategies have been developed, with the objective of solving multiple-input multiple-output (MIMO) control problems with significant process interactions. A popular class of control strategies in the academy is the optimal control, which comprehends Linear Quadratic Regulator (LQG) control. Essentially, these optimal control problems are solved offline considering a linear model as representative of the system and do not consider process constraints (SKOGESTAD and POSTLETHWAITE, 2007).

While important in the academy, optimal control techniques were not well-accepted on the industry, mainly due to not being able to deal with constraints and being regarded as impractical by the industrial process control community. Instead, the industrial community started developing Model Predictive Control (MPC) algorithms, such as the Dynamic Matrix Control (DMC), which grew to be the most popular multivariable process control technique (QIN and BADGWELL, 2003).

The main idea behind any MPC algorithm is to solve a constrained optimization problem at each time interval, in which the objective function to be minimized measures the proximity of controlled variables to their reference trajectories, the manipulation effort and the proximity of manipulated variables to their reference trajectories, according to (QIN and BADGWELL, 1997):

$$\min_{\Delta u} \ \mathbf{J} = \sum_{j=1}^{N_p} ||y_{k+j} - y_{k+j}^{sp}||_{Q_j}^2 + \sum_{j=0}^{N_c-1} ||\Delta u_{k+j}||_{S_j}^2 + \sum_{j=0}^{N_c-1} ||u_{k+j} - u_{k+j}^{sp}||_{R_j}^2$$

$$\begin{aligned} \text{subject to:} \quad & y_{k+j}^{min} \leq y_{k+j} \leq y_{k+j}^{max}, \quad j = 1, \cdots, N_p \\ & u_{k+j}^{min} \leq u_{k+j} \leq u_{k+j}^{max}, \quad j = 0, \cdots, N_c - 1 \\ & \Delta u_{k+j}^{min} \leq \Delta u_{k+j} \leq \Delta u_{k+j}^{max}, \quad j = 0, \cdots, N_c - 1 \\ & x_{k+j} = f(x_{k+j-1}, u_{k+j-1}), \quad j = 1, \cdots, N_p \\ & y_{k+j} = g(x_{k+j}, u_{k+j}), \quad j = 1, \cdots, N_p \end{aligned} \tag{2.14}$$

In this approach, N_c control actions are calculated at each time step, and the first control action is implemented. As soon as another measurement is available, the initial condition of the model is updated and a new sequence of control actions is calculated. This strategy is called *receding horizon*, and allows for accounting unmeasured disturbances and modeling errors (BORDONS and CAMACHO, 2007).

The main advantage of MPC in comparison to classical optimal control strategies for linear systems, as already said, is the capability of dealing with process constraints. This comes at the expense of solving an optimization problem at each time interval (SKOGESTAD and POSTLETHWAITE, 2007). While this was a limitant in the development of such techniques, there are now well-established optimization algorithms, optimized routines regarding receding horizon strategies and processors powerful enough to deal with the needed computational effort (BADGWELL and QIN, 2015).

While in Equation 2.14 the model equations $f(\cdot)$ and $g(\cdot)$ are not specified, they play important role in the computational efficiency of the MPC algorithm. If the models are linear, model equality constraints can be incorporated into the objective function and constraints, and the control problem can be rearranged into a quadratic programming (QP) optimization problem (QIN and BADGWELL, 1997). A classical solution of this kind is the QDMC algorithm (GARCIA and MORSHEDI, 1986), in which the model is treated in its step-response form.

Although these simple solutions are available, the models which describe the system can be as complex as differential-algebraic equation (DAE) systems, generating possibly non-convex optimization problems in which finding the optimal solution is not guaranteed and online implementation might not be feasible due to computational effort. The controller designer shall make a compromise between descriptive capability of the model and complexity of the optimization problem.

2.4 Health-aware control (HAC)

The term "health-aware control" is used to define control strategies that consider prognostics into the control structure. Using health status as a key information to operation planning is the main idea of the prognostics field, but using this information into an automatic control framework remains little explored, besides being potentially beneficial to several systems subject to degradation.

The first known appearance of the term “health-aware control” is found on ESCOBET *et al.* (2012), in which the objective was to extend the lifetime of a conveyor belt. To achieve this, a prognostics module dictates velocity set-points to a PID controller. This is an example of a structure that does not rely on advanced control techniques. In spite of this, the majority of the following health-aware control case studies rely on MPC controllers, mostly due to its versatility.

Before this, PEREIRA *et al.* (2010) already implemented what can be considered a health-aware control strategy, without explicitly using the term. In this work, an MPC controller was used, with constraints regarding system degradation, i.e., the system health indicator is not allowed to rise above the failure threshold in the predicted horizon. With this setup, the controller's role was to prevent failure upon the considered operational window, until preventive maintenance is done.

SANCHEZ *et al.* (2015) applied a health-aware control technique to a system of wind turbines, with the objective of minimizing equipment damage and keeping it under normal operation. In this case, an MPC with a damage term added to the controller objective function was used. The authors reported a trade-off between process performance and accrued equipment damage by changing the damage term weight. This was to be expected, if one interprets this formulation as a weighted-sum approach to a multi-objective optimization problem.

GROSSO *et al.* (2016) applied an Economic MPC to a model of the drinking water network of Barcelona. In this work, the same degradation index as PEREIRA *et al.* (2010) was used, also considering system health as a constraint for the MPC. This work was treated in a simplified fashion by SALAZAR *et al.* (2016), in which health was measured by reliability indexes allied to Bayesian Networks. SALAZAR *et al.* (2016) reported that focusing on components reliability may not lead to overall system reliability.

Using a somewhat unusual approach, CHOO *et al.* (2016) studied a smart manufacturing system, modeling subsystems degradation by a Markov Decision Process. States from subsystems were lumped in order to lower the number of states of the system. In a later work, CHOO *et al.* (2017) used this model in a reinforcement learning control framework, a growing concept in process control.

Using a robust MPC approach, VERHEYLEWEGHEN and JÄSCHKE (2017) studied the health-aware control of a subsea compression system subject to degradation at ball bearings. In the work, the Paris law was used to predict the degradation pattern, with an online parameter estimation to correct the model using previous observations. The covariance matrix from this estimation result was used to infer the predicted crack length standard deviation.

The control algorithm used in the work of VERHEYLEWEGHEN and JÄSCHKE (2017) was the scenario-based control (LUCIA *et al.*, 2013), in which uncertainties are explicitly taken into account by considering a set of the uncertain variables realizations. At each calculation of control actions, the controller must find a solution for all system realizations at once, by weighting each realization by its probability of occurring. This probability was obtained from the estimated crack length standard deviation. In this control approach, health condition is considered as a constraint, meaning that the system is prevented from breaking down in the prediction horizon.

Using a hierarchized control framework, POUR *et al.* (2018) studied the control of a pasteurization plant. The control framework comprises an optimization upper layer, in which economic objectives and health condition are dealt, and a regulatory lower layer, which focused on dealing with process dynamics by tracking the reference signals. This work also compared multilayer and single layer health-aware control structures, and the results showed the advantages of a multilayer framework.

VERHEYLEWEGHEN *et al.* (2018) also used a hierarchized control framework, but to study a subsea compression system, in which the compressor is subject to degradation. The hierarchy designed was composed of a regulatory layer with PID controllers, responsible for process stabilization, a supervisory layer, which deals with compression efficiency and surge avoidance, and an economic optimization layer, which considers maximum degradation as a constraint. In this work, health conditions were considered to be measured with perfect accuracy, and a fixed maintenance time was assumed, i.e., the main concern of the higher layer is to prevent system breakdown in the operation horizon.

2.5 Final comments

Based on the health-aware control approaches presented on the literature, a hierarchized approach might be beneficial to system performance. This might be due to the necessity of not only attaining different goals at the same time but to the difference of timescales of the involved phenomena (process dynamics and equipment degradation).

A remarkable characteristic of these works is that none assessed the issue of maintenance decision-making based on prognostic information. This is a vital activity in process management, and can be done based on an estimated RUL p.d.f. considering a confidence level. The classical control constrained by breakdown prevention should also be kept in the formulation, which means that the different control hierarchies shall act in different aspects of health management.

In terms of RUL p.d.f. estimation, particle filter is an adequate tool to account for inherent uncertainties of nonlinear systems in a generic framework. Thus, it will be the tool of choice in this work, allied to the degradation model.

This literature review has covered all theoretical aspects indispensable to this work development, and now we can proceed to the case study modeling.

Chapter 3

Case study modeling

In this chapter, the system model, along with the multiphase pump degradation model, is described.

3.1 CO_2 separation system

The system studied in this work is based on a process conception analyzed by SOUZA (2018). The model used to describe the system was adapted from the original author.

The process diagram considered in this work is reproduced in Figure 3.1, in which some variables of the process model are also indicated.

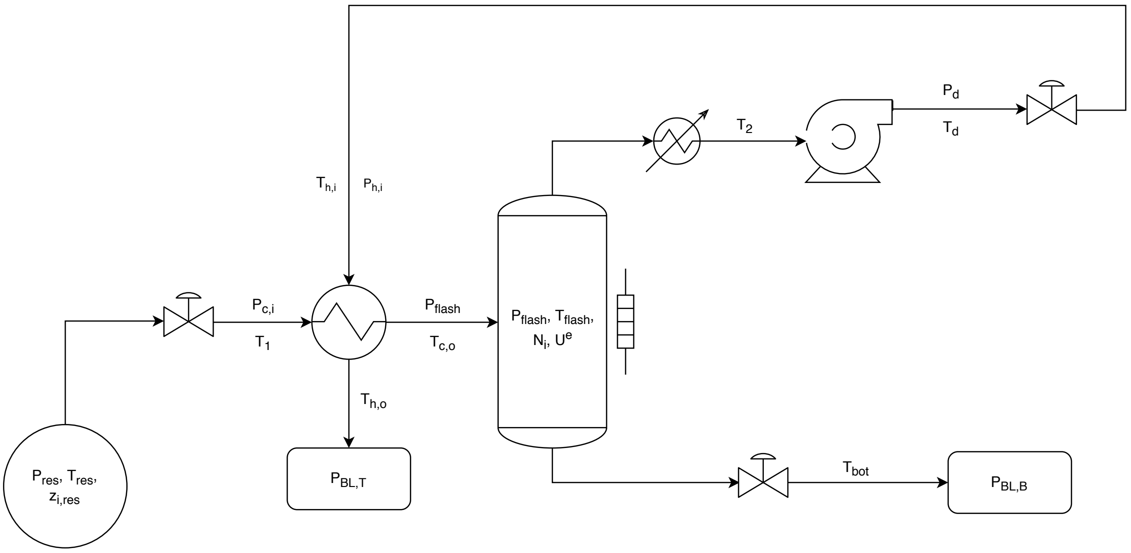

Figure 3.1: Process diagram, emphasizing correspondent variables

Process description can be summarized as follows. The crude oil in reservoir conditions is submitted to heat exchange with a hot stream from the process, which provides part of the heat necessary for separation. This stream is then sent to a separator drum, in which the remaining heat necessary for separation is provided. The CO_2 rich stream, withdrawn at the top outlet, is cooled, elevating its density so that it becomes adequate for pump operation. The pump elevates the fluid pressure, allowing its reinjection, but it also raises significatively the fluid temperature. This temperature elevation enables heat integration between this stream and crude oil extracted from the reservoir. After heat integration, the CO_2 rich stream is sent to reinjection section. The separation drum bottom outlet stream is then sent to topside processing. The interested reader is encouraged to consult SOUZA (2018) and SOUZA *et al.* (2019) for a more detailed description of the considered process.

In order to increase model capability to describe the system, changes were made in the valves and heat exchanger models. These changes are described in the following subsections.

3.1.1 Flow through valve

The model proposed by SOUZA (2018) is described by:

$$F = \frac{A}{MM} K_v x_v \left[\frac{-\displaystyle\int_1^2 \rho_m dP}{\ln \dfrac{\rho_{m,1}}{\rho_{m,2}}} \right]^{1/2} \tag{3.1}$$

in which F represents molar flow, A represents valve cross-section area, MM represents molecular weight, K_v represents valve constant, x_v represents valve opening, ρ_m represents mass density, and 1 and 2 represents inlet and outlet conditions, respectively. This model, while trying to account for compressibility effects, is indeterminate for incompressible fluids. A more traditional approach, followed in this work, is to describe the molar flow through the valve by:

$$F = \frac{A}{MM} K_v x_v \bar{\rho}_m \sqrt{\frac{\Delta P}{\bar{\rho}_m}} \tag{3.2}$$

Comparing this to Equation 3.1, the logarithmic term is omitted and the average density $\bar{\rho}_m$ may be calculated by:

$$\bar{\rho}_m = \frac{-1}{\Delta P} \int_1^2 \rho_m dP \tag{3.3}$$

3.1.2 Heat exchanger

SOUZA (2018) models a counter-current heat exchanger, schematized in Figure 3.2, using an approximate stationary solution, adding an estimate of the thermal holdup of the system, as described in eqs. (3.4a) to (3.4d) Equation 3.4.

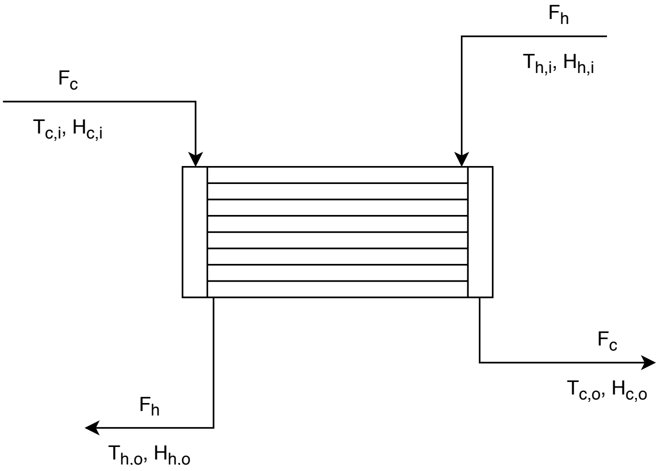

Figure 3.2: Heat exchanger scheme, emphasizing model variables

$$\begin{cases} \dfrac{dE_c}{dt} = F_c(H_{c,i} - H_{c,o}) + UA\Delta T_{lm} & \text{(3.4a)} \\ \dfrac{dE_h}{dt} = F_h(H_{h,i} - H_{h,o}) - UA\Delta T_{lm} & \text{(3.4b)} \\ E_c = V_c \dfrac{\rho_{c,i} + \rho_{c,o}}{2} \dfrac{H_{c,i} + H_{c,o}}{2} & \text{(3.4c)} \\ E_h = V_h \dfrac{\rho_{h,i} + \rho_{h,o}}{2} \dfrac{H_{h,i} + H_{h,o}}{2} & \text{(3.4d)} \end{cases}$$

The stationary solution is valid under the assumption that fluid properties do not change significantly with temperature (INCROPERA, 2006). From this solution arises the logarithmic mean of temperature difference, ΔT_{lm}, defined for a counter-current exchanger as:

$$\Delta T_{lm} = \frac{\Delta T_2 - \Delta T_1}{\ln(\Delta T_2/\Delta T_1)}, \qquad \begin{cases} \Delta T_1 = T_{h,o} - T_{c,i} \\ \Delta T_2 = T_{h,i} - T_{c,o} \end{cases} \tag{3.5}$$

While this solution is valid at process design, some flaws are concerning. The heat transfer rate, modeled as $Q = UA\Delta T_{lm}$, is only valid at the stationary case. In case of temperature change at one of the inlets, for example, this rate would not have precise physical meaning until a new steady state is established. This behavior is further aggravated in a temperature flip, in which outlet and inlet temperature differences have opposite signals, and the logarithmic mean cannot even be computed.

In order to evade these issues, the model proposed in this work consists of discretizing the heat exchanger in theoretical stages, as depicted in Figure 3.3.

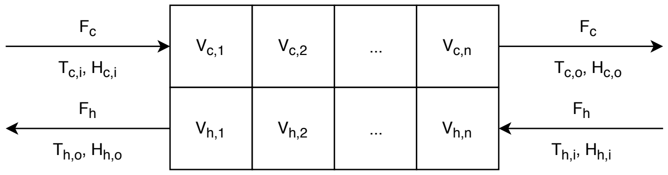

Figure 3.3: Heat exchanger discretization scheme, emphasizing model variables

By performing energy balances at each theoretical stage, the following equations apply:

$$\begin{cases} \dfrac{dE_{c,k}}{dt} = F_c(H_{c,k-1} - H_{c,k}) + UA_k(T_{h,k} - T_{c,k}) & \qquad (3.6a) \\ \dfrac{dE_{h,k}}{dt} = F_h(H_{h,k+1} - H_{h,k}) - UA_k(T_{h,k} - T_{c,k}), \quad k = 1, \cdots, n_{st} & \qquad (3.6b) \\ E_{c,k} = V_{c,k}\rho_{c,k}H_{c,k} & \qquad (3.6c) \\ E_{h,k} = V_{h,k}\rho_{h,k}H_{h,k} & \qquad (3.6d) \end{cases}$$

with n_{st} being the number of theoretical stages after discretization. The boundary conditions, respecting discretization indices, are $T_{c,i} = T_{c,0}$, $T_{c,o} = T_{c,n}$, $T_{h,i} = T_{h,n+1}$ and $T_{h,o} = T_{h,1}$.

By joining these changes with the equations proposed by SOUZA (2018), the complete process model can be obtained. The complete equation system, along with the numerical strategy to solve it, is described in the next subsection.

3.1.3 The equation system

System modeling results in a DAE system. Many algebraic constraints are explicit in some variable, so the resulting system was written in a way that minimizes the number of state variables.

The resulting system is written as:

$$\frac{dE_{c,k}}{dt} = F_{in}(H_{c,k-1} - H_{c,k}) + UA_k(T_{h,k} - T_{c,k}), \qquad k = 1, \cdots, n_{st} \tag{3.7a}$$

$$\frac{dE_{h,k}}{dt} = F_v(H_{h,k+1} - H_{h,k}) - UA_k(T_{h,k} - T_{c,k}), \qquad k = 1, \cdots, n_{st} \tag{3.7b}$$

$$E_{c,k} = V_{c,k}\rho_{c,k}H_{c,k}, \qquad k = 1, \cdots, n_{st} \tag{3.7c}$$

$$E_{h,k} = V_{h,k}\rho_{h,k}H_{h,k}, \qquad k = 1, \cdots, n_{st} \tag{3.7d}$$

$$\frac{dN_i}{dt} = F_{in}z_{i,res} - F_v\nu_i - F_l\alpha_i - F_w\gamma_i, \qquad i = 1, \cdots, n_c \tag{3.7e}$$

$$\frac{dU^e}{dt} = F_{in}H_F - F_vH_v - F_lH_l - F_wH_w + Q_{aqu} \tag{3.7f}$$

$$\frac{U^e + P_{flash}V_{flash}}{N_{tot}} = \beta_\nu H_v + \beta_\alpha H_l + \beta_\gamma H_w \tag{3.7g}$$

$$\frac{V_{flash}}{N_{tot}} = \beta_\nu \bar{V}_v + \beta_\alpha \bar{V}_l + \beta_\gamma \bar{V}_w \tag{3.7h}$$

$$H(T_1, P_{c,i}, \boldsymbol{z_{res}}) = H(T_{res}, P_{res}, \boldsymbol{z_{res}}) \tag{3.7i}$$

$$H(T_d, P_d, \boldsymbol{\nu}) - H(T_2, P_{flash}, \boldsymbol{\nu}) = \int_{P_{flash}}^{P_d} \bar{V}dP \tag{3.7j}$$

$$W = F_v \left\{ MM(\boldsymbol{\nu}) \left[\left(\frac{F_v\bar{V}_d}{A_{v,t}} \right)^2 - \left(\frac{F_v\bar{V}_s}{A_{v,t}} \right)^2 \right] + \int_{P_{flash}}^{P_d} \bar{V}dP \right\} \tag{3.7k}$$

$$H(T_{h,i}, P_{h,i}, \boldsymbol{\nu}) = H(T_d, P_d, \boldsymbol{\nu}) \tag{3.7l}$$

$$Q_{resf} = F_v \left[H(T_2, P_{flash}, \boldsymbol{\nu}) - H(T_{flash}, P_{flash}, \boldsymbol{\nu}) \right] \tag{3.7m}$$

$$H(T_{bot}, P_{BL,B}, \boldsymbol{l}) = H(T_{flash}, P_{flash}, \boldsymbol{l}) \tag{3.7n}$$

with the state vector being $\boldsymbol{x} = [E_c, E_h, T_c, T_h, N, U, T_{flash}, P_{flash}, T_1, P_d, T_2, T_d, T_{h,i}, T_{bot}]$, and input vector being $\boldsymbol{u} = [x_{v,t}, x_{v,b}, x_v, Q_{aqu}, Q_{resf}, W]$.

Other variables presented in the equations are calculated directly as function of states and inputs, according to the following equations:

$$[\beta_\nu, \beta_\alpha, \beta_\gamma, \boldsymbol{\nu}, \boldsymbol{\alpha}, \boldsymbol{\gamma}] = \text{Flash3P}(T_{flash}, P_{flash}, \boldsymbol{z}), \quad z_i = \frac{N_i}{N_{tot}}, \quad N_{tot} = \sum_{i=1}^{n_c} N_i$$

$$l_i = \frac{\beta_\alpha\alpha_i + \beta_\gamma\gamma_i}{\beta_\alpha + \beta_\gamma}, \quad F_l = \frac{\beta_\alpha F_{l,tot}}{\beta_\alpha + \beta_\gamma}, \quad F_w = F_{l,tot} - F_l, \quad H_F = H(T_{c,o}, P_{flash}, \boldsymbol{z_{res}})$$

$$H_v = H(T_{flash}, P_{flash}, \boldsymbol{\nu}), \quad H_l = H(T_{flash}, P_{flash}, \boldsymbol{\alpha}), \quad H_w = H(T_{flash}, P_{flash}, \boldsymbol{\gamma})$$

$$\bar{V}_v = \bar{V}(T_{flash}, P_{flash}, \boldsymbol{\nu}), \quad \bar{V}_l = \bar{V}(T_{flash}, P_{flash}, \boldsymbol{\alpha}), \quad \bar{V}_w = \bar{V}(T_{flash}, P_{flash}, \boldsymbol{\gamma})$$

$$P_{h,i} = \Delta P_h + P_{BL,T}, \quad P_{c,i} = \Delta P_c + P_{flash}$$

$$P_{h,k} = P_{h,i} + \left(\frac{k - \frac{1}{2}}{n_{st}}\right) \Delta P_h, \quad P_{c,k} = P_{c,i} + \left(\frac{n_{st} - k + \frac{1}{2}}{n_{st}}\right) \Delta P_c, \quad k = 1, \cdots, n_{st}$$

$$H_{h,k} = H(T_{h,k}, P_{h,k}, \boldsymbol{\nu}), \quad H_{c,k} = H(T_{c,k}, P_{c,k}, \boldsymbol{z_{res}}), \quad k = 1, \cdots, n_{st}$$

$$\rho_{h,k} = \frac{1}{\bar{V}(T_{h,k}, P_{h,k}, \boldsymbol{\nu})}, \quad \rho_{c,k} = \frac{1}{\bar{V}(T_{c,k}, P_{c,k}, \boldsymbol{z_{res}})}, \quad k = 1, \cdots, n_{st}$$

$$T_{c,0} = T_1, T_{h,n+1} = T_{h,i}, T_{c,o} = T_{c,n}$$

$$\bar{V}_d = \bar{V}(T_d, P_d, \boldsymbol{\nu}), \quad \bar{V}_s = \bar{V}(T_2, P_{flash}, \boldsymbol{\nu}), \quad \rho_m(T, P, \xi) = \frac{MM(\xi)}{\bar{V}(T, P, \xi)}$$

$$F_{in} = \frac{A_{v,in}}{MM(\boldsymbol{z_{res}})} K_{v,in} x_v \left[- \int_{P_{res},T_{res}}^{P_{c,i},T_1} \rho_m dP \right]^{1/2}$$

$$F_v = \frac{A_{v,t}}{MM(\boldsymbol{\nu})} K_{v,t} x_{v,t} \left[- \int_{P_d,T_d}^{P_{h,i},T_{h,i}} \rho_m dP \right]^{1/2}$$

$$F_{l,tot} = \frac{A_{v,b}}{MM(\boldsymbol{l})} K_{v,b} x_{v,b} \left[- \int_{P_{flash},T_{flash}}^{P_{BL,B},T_{bot}} \rho_m dP \right]^{1/2}$$

All integrals were evaluated by the trapezoidal rule, for simplicity. Flash calculations were performed using the "Flash3P" algorithm, available in *CO2Therm* package. Thermodynamic properties (molar enthalpy H and molar volume $\bar{V}$) were evaluated as the weighted sum of the respective properties at each existing phase. Thus, it is required to perform a flash calculation at each thermodynamic property evaluation. Details of the thermodynamic package can be found at SOUZA (2018).

Output variables from the model were selected as $\boldsymbol{y} = [F_{in}, P_{flash}, T_{flash}, T_2, F_v, F_{l,tot}, x_{CO_2}, h_{flash}]$, in which h_{flash} is the flash drum liquid volumetric fraction, calculated by:

$$h_{flash} = \frac{(\beta_\alpha \bar{V}_l + \beta_\gamma \bar{V}_w) N_{tot}}{V_{flash}} \tag{3.8}$$

3.2 Degradation model

To describe multiphase pump wear, a novel discrete-time stochastic model was proposed:

$$\lambda_{k+1} = \lambda_k + \Delta f_k \ \lambda_k^n, \qquad \Delta f_k \sim Gamma(\theta_1 W_k \Delta t, \theta_2) \tag{3.9}$$

with the p.d.f. corresponding to a $Gamma(k, \theta)$ distribution being defined by:

$$f(x|k,\theta) = \frac{x^{k-1}}{\Gamma(k)\theta^k} e^{-\dfrac{x}{\theta}} \tag{3.10}$$

In this model, λ_k represents the degradation state at time k. The proposed degradation model is based on the Paris law for crack growth and Gamma processes. The parameter Δf_k, which represents crack growth rate in Paris law, is modelled as a Gamma-distributed random variable, with its shape parameter proportional to pump power.

Although being based on the Paris law, the state λ aims to describe the overall equipment degradation, lumping all possible degradation processes into one model. With this, it is possible to ally the simplicity and versatility of Paris law with the stochastic nature of Gamma processes.

Also, to simulate noise in the measurements, the observed variable is the equipment wear corrupted by white noise, according to:

$$\eta_k = \lambda_k + \epsilon_k, \qquad \epsilon_k \sim \mathcal{N}(0, \sigma^2) \tag{3.11}$$

Chapter 4

Methodology

All simulations developed in the course of this work were carried out in Python. While numerical packages from the Scipy (JONES *et al.*, 2001) and Assimulo (ANDERSSON *et al.*, 2015) libraries were used, the formulation of control and state estimation problems was made by the author, generating the packages *control_tools* and *PF_tools*. The *control_tools* package features classes for building dynamic models, controllers and state estimators. The *PF_tools* package features the particle filter implementation described in Section 4.6.

4.1 Model implementation

Model implementation was carried out in Python. The *CO2Therm* package developed by SOUZA (2018) was interfaced to Python using the *Boost:Python* library. DAE integration was performed using IDA algorithm from Assimulo package. Sampling time for the simulation was chosen as 1 second.

Equipments were first simulated separately, in order to obtain accurate values for temperatures and densities. Equations used were Equation 3.7i, Equation 3.7j, Equation 3.7l and Equation 3.7n. Valve openings were then determined *a priori*, based on the obtained temperatures and the flash results for reservoir composition and drum conditions, thus solving mass balance equations and drum volume algebraic constraint. For this, entrance valve opening was specified as $x_v = 0.5$.

Then, a reference steady state was determined by solving the remaining system equations, with variables normalized by their initial guess, using the *scipy.optimize.least_squares* algorithm (tolerance $xtol = 5 \times 10^{-6}$), with parameters and fixed variables given by Table 4.1 and Table 4.2, respectively. The free variables were $\boldsymbol{DoF} = [E_c, E_h, T_c, T_h, U^e, T_1, T_d, T_{h,i}, T_{bot}, W, Q_{aqu}, Q_{resf}]$. Heat exchanger volumes were estimated considering flows from SOUZA (2018) and a residence time of 6 seconds on both sides.

To avoid divergences, constraints were imposed on the optimization problem. All variables were allowed to vary 20% of their initial guess.

Table 4.1: Parameters used in the simulation

Parameter	Value	Unit	Description
n_{st}	3	-	Number of heat exchanger theoretical stages
$K_{v,in}$	0.0982	-	Inlet valve constant
$K_{v,t}$	0.0564	-	Top valve constant
$K_{v,b}$	0.1589	-	Bottom valve constant
ΔP_h	0	MPa	Pressure drop of hot fluid (top stream)
ΔP_c	0	MPa	Pressure drop of cold fluid (inlet stream)
UA	0.153	MW/K	Overall heat transfer coefficient
V_{flash}	31.8086	m^3	Flash drum volume
V_c	1.806	m^3	Heat exchanger cold side volume
V_h	0.387	m^3	Heat exchanger hot side volume

Table 4.2: Fixed variables used in the simulation

Parameter		Value	Unit	Description
$\boldsymbol{z_{res}}$	H_2O	0.0100	-	Reservoir composition
	CO_2	0.7500	-	
	CH_4	0.0480	-	
	F_1	0.0543	-	
	F_2	0.0443	-	
	F_3	0.0520	-	
	F_4	0.0413	-	
T_{res}		313.15	K	Reservoir temperature
P_{res}		15	MPa	Reservoir pressure
$P_{BL,B}$		9	MPa	Pressure at bottom-side battery limit
$P_{BL,T}$		59	MPa	Pressure at top-side battery limit
T_{flash}		330.25	K	Flash drum temperature
P_{flash}		10	MPa	Flash drum pressure
P_d		60	MPa	Pump discharge pressure
T_2		303.65	K	Condenser output temperature

To refine the previous solution with relation to the integrator characteristic at the flash drum level, bottom valve opening $x_{v,b}$ was coupled to the output h_{flash} with a proportional controller, aiming to stabilize the system. A closed-loop dynamic simulation was then performed, and the obtained ending value of $x_{v,b}$ is the value corresponding to the true steady state. Controller gain used was -10 (dimensionless), and the system was simulated until 5000 s. Open-loop dynamic simulations were then made to confirm that a steady state was obtained.

In order to implement step disturbances into the system, a regularization function was used. The function used to approximate the Heaviside function (also known as the unit step function) is given by:

$$H_{ap}(t) = \frac{1 + \tanh\left[R\left(2\frac{t}{\Delta t_{set}} - 1\right)\right]}{2} \tag{4.1}$$

in which Δt_{set} is the settling time of the approximate step, and R is its regularization factor. This function was designed to be valued $\dfrac{1 - \tanh R}{2}$ when the approximate step begins ($t = 0$) and valued $\dfrac{1 + \tanh R}{2}$ when the approximate step is established ($t = \Delta t_{set}$). The value of R was set to make $\dfrac{1 - \tanh R}{2} = 10^{-4}$.

4.2 Plant stabilization

In order to stabilize plant dynamic behavior, a PI controller for the pair $x_{v,b}$ x h_{flash} was implemented together with the process model. The transfer function for this input-output pair was estimated from step disturbances of ±1% and ±5% of the steady-state $x_{v,b}$ value. The controller was then tuned using SIMC rules (SKOGESTAD, 2003), due to its superiority in disturbance rejection, with $\tau_c = 10$ seconds. With this, the input vector was changed to $\boldsymbol{u} = [x_{v,t}, h_{sp}, x_v, Q_{aqu}, Q_{resf}, W]$, a new state Ψ representing the integral term was introduced, and $x_{v,b}$ became calculated by the explicit control law:

$$\begin{cases} x_{v,b} = k_c\left[(h_{sp} - h_{flash}) + \dfrac{\Psi}{\tau_I}\right] \\ \dfrac{d\Psi}{dt} = h_{sp} - h_{flash}, \qquad \Psi(0) = \dfrac{x_{v,b}^{steady}\tau_I}{k_c} \end{cases} \tag{4.2}$$

in which $x_{v,b}^{steady}$ is the value for $x_{v,b}$ calculated in the last section and k_c and τ_I are PI controller parameters.

4.3 Model identification

Considering the stabilized plant and its output variables, from the reference steady state, the system was identified using positive and negative steps of 1% and 5% of the nominal input values, except for the input h_{sp}, which was not considered as a manipulated variable for the predictive controller, and the output variable h_{flash}, which is already controlled. From these responses, continuous-time transfer functions were fitted to the data. Parameter estimation was performed using the *scipy.optimize.least_squares* algorithm (tolerance $xtol = 10^{-8}$), with continuous-time transfer function responses being evaluated by the *scipy.signal.lsim* algorithm. Identified transfer functions were of the "*zpk*" form:

$$G(s) = k\frac{\prod\limits_i (s - z_i)}{\prod\limits_i (s - p_i)} \tag{4.3}$$

The number of poles and zeros was determined by inspection of step response behavior, admitting the minimal number which could explain the dynamic behavior.

4.4 Closed loop simulation

The identified model was transformed to a continuous state-space realization, using a diagonal canonical form (OGATA, 2010), which was then discretized (SIMON, 2006) to a sampling time of 10 seconds, generating the following dynamic model:

$$\begin{cases} \boldsymbol{x_k} = \boldsymbol{F}\boldsymbol{x_{k-1}} + \boldsymbol{G}\boldsymbol{u_{k-1}} \\ \boldsymbol{y_k} = \boldsymbol{C}\boldsymbol{x_k} + \boldsymbol{D}\boldsymbol{u_k} \end{cases} \tag{4.4}$$

This discrete model was then used as the internal model to a MPC controller. To compensate for the model-plant mismatch, a Kalman Filter was used to estimate the state bias, $\boldsymbol{w}$, which was used to correct current state estimate and all states in the prediction horizon.

The MPC optimization problem was formulated as:

$$\min_{\boldsymbol{\Delta u_{k+j}}} \quad J = \sum_{i=1}^{N_y}\sum_{j=1}^{N_p} {q_i}^2(y_{i,k+j} - y_{i,k+j}^{sp})^2 + \sum_{i=1}^{N_u}\sum_{j=0}^{N_c-1} {s_i}^2(\Delta u_{i,k+j})^2$$

$$\begin{aligned} \text{subject to:} \quad & \boldsymbol{u_{k+j}^{min}} \leq \boldsymbol{u_{k+j}} \leq \boldsymbol{u_{k+j}^{max}}, \quad j = 0, \cdots, N_c - 1 \\ & -\boldsymbol{\Delta u_{k+j}^{max}} \leq \boldsymbol{\Delta u_{k+j}} \leq \boldsymbol{\Delta u_{k+j}^{max}}, \quad j = 0, \cdots, N_c - 1 \\ & \boldsymbol{x_{k+j}} = \boldsymbol{F x_{k+j-1}} + \boldsymbol{G u_{k+j-1}} + \boldsymbol{w_k}, \quad j = 1, \cdots, N_p \\ & \boldsymbol{y_{k+j}} = \boldsymbol{C x_{k+j}} + \boldsymbol{D u_{k+j}}, \quad j = 1, \cdots, N_p \end{aligned} \tag{4.5}$$

in which $\boldsymbol{w_k}$ represents the model correction at the time step k, and q_i and s_i represent controlled and manipulated variables weights, respectively. This optimization problem was solved using the *scipy.optimize.minimize* routine, using the SLSQP method (tolerance $ftol = 10^{-6}$). Model constraints were treated inside the objective function calculation, and inequality constraints regarding manipulated variables were explicited to the algorithm.

Using the identified linear model, a discrete Kalman Filter was implemented to perform state estimation and model update. This was performed using the following equations for temporal and measurement updates:

$$\begin{cases} \boldsymbol{P_k^-} = \boldsymbol{F P_{k-1}^+ F^T} + \boldsymbol{Q} \\ \boldsymbol{\hat{x}_k^-} = \boldsymbol{F \hat{x}_{k-1}^+} + \boldsymbol{G u_{k-1}} + \boldsymbol{w_{k-1}} \end{cases} \tag{4.6}$$

$$\begin{cases} \boldsymbol{K_k} = \boldsymbol{P_k^- C^T}(\boldsymbol{C P_k^- C^T} + \boldsymbol{R})^{-1} \\ \boldsymbol{P_k^+} = (\boldsymbol{I} - \boldsymbol{K_k C})\boldsymbol{P_k^-} \\ \boldsymbol{\varphi_k} = \boldsymbol{K_k}(\boldsymbol{y_k} - \boldsymbol{C \hat{x}_k^-} - \boldsymbol{D u_k}) \\ \boldsymbol{\hat{x}_k^+} = \boldsymbol{\hat{x}_k^-} + \boldsymbol{\varphi_k} \\ \boldsymbol{w_k} = \boldsymbol{w_{k-1}} + \boldsymbol{\varphi_k} \end{cases} \tag{4.7}$$

The last equation represents the model correction update, which was also based on the filter innovations $\boldsymbol{\varphi_k}$.

Using this framework, different simulations were performed to evaluate controller and filter tuning. Controlled variables were selected as $[F_{in}, P_{flash}, T_{flash}, T_2]$, and manipulated variables were selected as $[x_{v,t}, x_v, Q_{aqu}, Q_{resf}, W]$.

4.5 Degradation model implementation

The model described in Section 3.2 was implemented using the parameters listed on Table 4.3. Sampling of random variables was performed using the *scipy.stats* algorithms.

Table 4.3: Degradation model parameters

λ_0	Δt	θ_1 (MJ^{-1})	θ_2	n	σ
0.001	0.05 year	5×10^{-7}	0.01	0.7	0.0001

Simulations were made to check if the expected behavior was captured by the formulation. For this, different regimes of pump power manipulation were considered, with repetitions to obtain different realizations of the stochastic process.

4.6 Crack length estimation

A particle filter was implemented in order to estimate statistical information regarding crack length in an online environment. As the model described in Equation 3.9 is stochastic, realizations of the random parameters are generated, and states are propagated considering the likelihood of the respective realization. The algorithm described by SPEEKENBRINK (2016) for a generic particle filter was adapted to the necessities of this implementation, leading to Algorithm 1. All statistical methods, such as sampling of random variables and calculations of p.d.f.s, were performed using the *scipy.stats* algorithms.

In this algorithm, μ represents the assumed initial distribution for the system state, X_k^i and w_{k-1}^i represent the i-th particle and its correspondent weight at time k, y_k represents system measurement at time k, ψ_j represents probability distribution of j-th parameter conditioned to the state and previous parameters, f represents the state evolution law, g represents the likelihood function, N_{part} is the number of particles and c is the filter resampling rate. The resampling algorithm used was the systematic resampling, described in SPEEKENBRINK (2016).

Algorithm 1 describes both SIS and SIR filters, depending on the adopted resampling rate (c). Both filters were tested with $N_{part} = 100$ in order to evaluate the quality of the obtained p.d.f. Resampling rate for the SIR filter was $c = 0.5$, and for the SIS filter $c = 0$. Particles at time $k = 0$ were sampled from a normal distribution with mean 0.001 and standard deviation 10^{-5}.

Using the filter results, a prediction of RUL was made at each measurement, considering the planned horizon for the manipulated variable. This was compared

Algorithm 1: Particle filter algorithm

```
Data: $[X^i_{k-1}, w^i_{k-1}]$, $y_k$, $\psi_j$ $(j = 1, \cdots, n_\psi)$, $f$, $g$, $N_{part}$, $c$
Result: $[X^i_k, w^i_k]$
// A priori calculations
for $i \leftarrow 1$ to $N_{part}$ do
    $p^i_\psi \leftarrow 1$; // initialize transition probability
    for $j \leftarrow 1$ to $n_\psi$ do
        Sample $\Psi^i_{k-1,j} \sim \psi_j(p_j|X^i_{k-1}, \Psi^i_{k-1,1}, \cdots, \Psi^i_{k-1,j-1})$;
        $p^i_\psi \leftarrow p^i_\psi \psi_j(\Psi^i_{k-1,j}|X^i_{k-1}, \Psi^i_{k-1,1}, \cdots, \Psi^i_{k-1,j-1})$;
    end
    $X^i_k \leftarrow f(X^i_{k-1}, \Psi^i_{k-1,1}, \cdots, \Psi^i_{k-1,n_\psi})$; // propagate particle in
        sampled path
    $w^i_k \leftarrow w^i_{k-1} p^i_\psi$; // update weight with probability of sampled path
end
$\xi \leftarrow \sum_{i=1}^{N_{part}} w^i_k$;
$w^i_k \leftarrow w^i_k/\xi,\ i = 1, \cdots, N_{part}$;
// A posteriori calculations
$w^i_k \leftarrow w^i_k g(y_k|X^i_k),\ i = 1, \cdots, N_{part}$;
$\xi \leftarrow \sum_{i=1}^{N_{part}} w^i_k$;
$w^i_k \leftarrow w^i_k/\xi,\ i = 1, \cdots, N_{part}$;
$N_{eff} \leftarrow 1/\sum_{i=1}^{N_{part}} (w^i_k)^2$;
if $N_{eff} < cN_{part}$ then
    $[X^i_k, w^i_k] \leftarrow$ Resample$(X^i_k, w^i_k)$;
end
```

to the real end-of-life time in the simulation of the specific realization. The threshold value for the health indicator was fixed at $\lambda^{lim} = 1.0$.

4.7 Health-aware control implementation

Combining the proposed methodologies from Section 4.4 and Section 4.6, a HAC structure was implemented. It consists of an MPC, with objective function written as:

$$\mathbf{J_{HAC}} = -w_{HAC}\overline{RUL} + \sum_{i=1}^{N_y}\sum_{j=1}^{N_p} {q_i}^2 (y_{i,k+j} - y^{sp}_{i,k+j})^2 + \sum_{i=1}^{N_u}\sum_{j=0}^{N_c-1} {s_i}^2 (\Delta u_{i,k+j})^2 \tag{4.8}$$

with w_{HAC} being the weighting factor between RUL extension and control objectives, and $\overline{RUL}$ being the average of RUL between the propagated particles.

Due to the randomness of the model proposed in Equation 3.9, it cannot be directly used in a deterministic optimization problem. As such, the strategy used to calculate $\overline{RUL}$ during the optimization procedure is as given in Algorithm 2.

In the algorithm, $q_k^{(i)}$ represents the quantile associated with the k-th evolution of the i-th particle, W_{end} represents pump power at the end of the control horizon, and $Q(\cdot)$ represents the inverse cumulative distribution function (c.d.f.) associated with the distribution in Equation 3.9, depending on the pump power which varies along the optimization. The set of $q_k^{(i)}$ is sampled from a standard uniform distribution, and kept constant in the optimization course.

Due to the difference between timescales, W_{end} was considered as the dominant effect in the degradation pattern, and the variation during the control horizon can be neglected. The effect of model-plant mismatch was evaluated by changing the process model between the model described in Section 3.1 and the MPC internal model itself. The influence of the number of particles (N_{part}) and the value of w_{HAC} over the closed-loop response was also evaluated.

Algorithm 2: $\overline{RUL}$ calculation algorithm

```
Data: $\lambda_0^{(i)}\ (i = 1, \cdots, N_{part})$, $\lambda^{lim}$, $q_k^{(i)}\ (i = 1, \cdots, N_{part},\ k = 0, \cdots)$, $W_{end}$
Result: $\overline{RUL}$
for $i \leftarrow 1$ to $N_{part}$ do
    $k \leftarrow 0$ ;
    while $\lambda_k^{(i)} < \lambda^{lim}$ do
        $\Delta f_k^{(i)} \leftarrow Q(q_k^{(i)}, W_{end})$ ;
        $\lambda_{k+1}^{(i)} \leftarrow \lambda_k^{(i)} + \Delta f_k^{(i)} (\lambda_k^{(i)})^n$ ;
        $k \leftarrow k + 1$ ;
    end
    $RUL^{(i)} \leftarrow k - \dfrac{\lambda_k^{(i)} - \lambda^{lim}}{\lambda_k^{(i)} - \lambda_{k-1}^{(i)}}$ ;
end
$\overline{RUL} \leftarrow \sum_{i=1}^{N_{part}} RUL^{(i)} / N_{part}$ ;
```

Chapter 5

Results and discussion

5.1 Model implementation

Some commentary about the replacement of valve flow equations (Subsection 3.1.1) is necessary. Equations developed by SOUZA (2018) were tested in the inlet conditions (313.15 K, 15 MPa), along with the proposal of this work. A sensitivity analysis was made with relation to the discharge pressure and number of integral quadrature points, whose results are shown in Figure 5.1 and Figure 5.2.

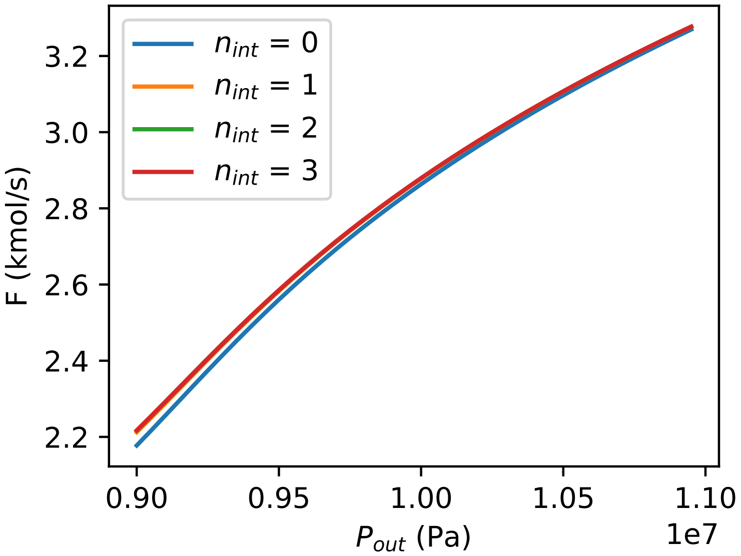

Figure 5.1: SOUZA (2018) valve model sensitivity analysis - before model correction

In the original model proposed by SOUZA (2018), higher discharge pressure, and consequently lower pressure difference in the valve, leads to a higher molar flow, which is physically unrealistic. This was due to the logarithmic term in the denominator, whose removal led to a more realistic response.

The number of quadrature points was considered to have little significance in the

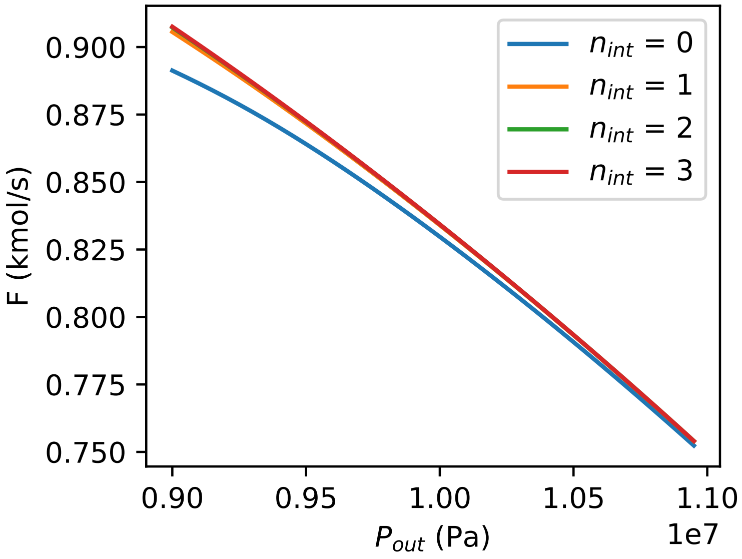

Figure 5.2: Proposed valve model sensitivity analysis - after model correction

flow results. As the system had already a reasonable number of variables and each quadrature point adds another state to the system for each valve, the trapezoidal rule (no internal quadrature points) was adopted.

In terms to the replacement of the heat exchanger model, some change on the stationary value is to be expected. The approach by theoretical stages underestimates heat transfer, in the sense that stage mixture diminishes temperature difference of cold and hot currents. When the number of stages approaches infinity, the solution will tend to the true temperature profile along the axis. But, in order to keep a reasonable number of variables, a small stage number was chosen, at the cost of accuracy when comparing to an ideal counter-current heat exchanger.

Regarding the characteristics of the mathematical system described in Equation 3.7, a sparsity pattern of the iteration matrix was generated, and is presented in Figure 5.3. Each matrix line represents a variable, and each matrix column represents an equation from the DAE system, referenced by its subindex (e.g., the last column refers to Equation 3.7n).

It can be seen that the vast majority of equations depend on the flash drum holdup, which is due to the energy recycle step. This makes the sequential solution difficult, so the use of a simultaneous approach to the solution of the system equations is justified.

In order to perform dynamic simulation of a process described by a DAE system, it is necessary to obtain a reference steady state. Even though SOUZA (2018) describes a reference steady state, the changes made to the model made a new stationary simulation needed. Due to numerical issues involving flash drum mass balances,

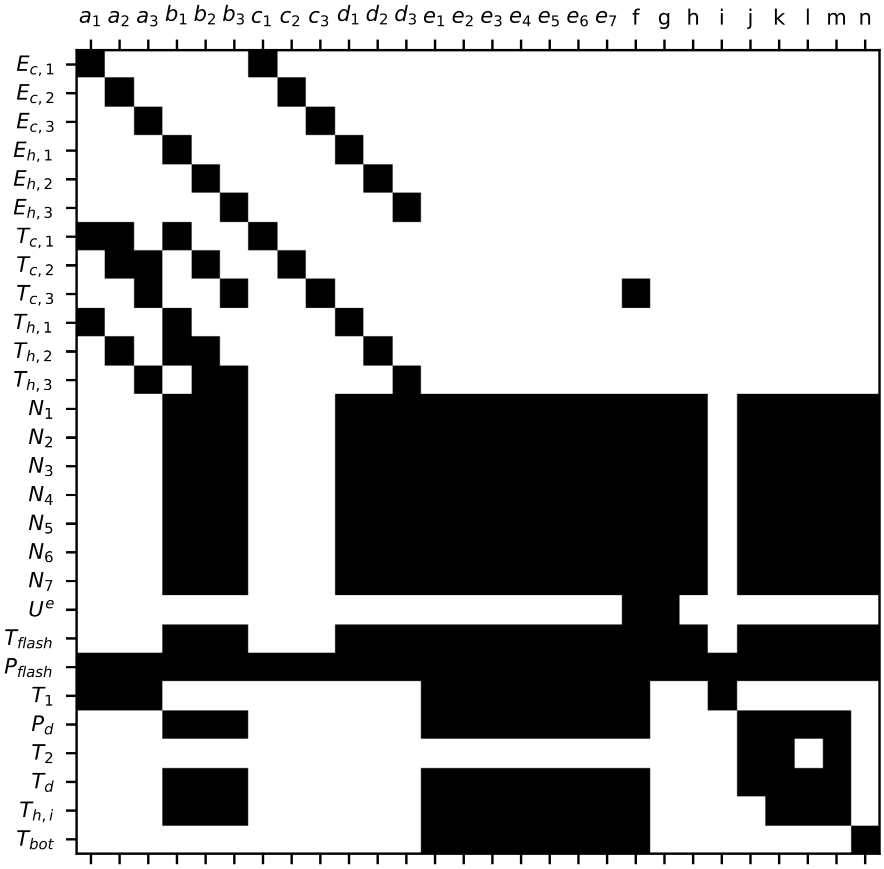

Figure 5.3: DAE system sparsity pattern (variables vs. equations in Equation 3.7)

the equation system described in Equation 3.7 could not be solved simultaneously. This motivated the use of a sequential strategy to the stationary simulation.

The first step to obtain the steady-state solution was to solve mass balances at the drum, given its operating temperature and pressure. If a inlet flow is specified, and perfect separation is assumed, it is known that stationary outlet flow solutions ought to be related to the phase fractions obtained at the flash calculation, and drum compositions ought to be equal to the inlet compositions.

Using this information, steady-state valve openings could be obtained, as presented in Table 5.1:

Table 5.1: Steady state valve openings

x_v (specified)	$x_{v,t}$	$x_{v,b}$
0.5	0.49828	0.50128

After these calculations, the next step was to find a reference steady state, corresponding to system design. This is not a trivial task, due to the numerical issues arisen by scale difference of the variables. Variable normalization was performed, to address this issue.

The initial guesses used in the stationary simulation are given by Table 5.2. These estimates were obtained by simulating each equipment separately, with the information from Table 4.1 and Table 4.2. The initial guess for the drum internal energy was obtained using mixture molar enthalpy. The heat exchanger energy holdup was obtained from the temperature estimate.

Table 5.2: Initial guesses for the free variables used in the stationary simulation

Parameter		Value	Unit	Description
	1	314.25	K	
$T_{c,k}$	2	315.83	K	Cold side stage temperature
	3	318.12	K	
	1	328.05	K	
$T_{h,k}$	2	334.51	K	Hot side stage temperature
	3	343.21	K	
T_1		312.95	K	Inlet valve discharge temperature
T_d		354.70	K	Pump discharge temperature
$T_{h,i}$		354.87	K	Top valve discharge temperature
T_{bot}		329.45	K	Bottom valve discharge temperature
W		4.259	MW	Pump power
$\dot{Q}_{aqu}$		5.478	MW	Drum heating rate
$\dot{Q}_{resf}$		-5.5	MW	Condenser cooling rate

To confirm that the obtained solution is adequate, dynamic simulations were performed using the stationary simulation solution as the initial condition. If the solution is adequate, time variation of states, and consequently of outputs, is negligible. The results of such dynamic simulation are presented in Figure 5.4.

This simulation shows that the obtained solution is close to a steady state, but there is still some time variation in flash drum liquid occupied volume, due to numerical errors associated with the simulation (possibly in the flash calculations). Also, the integrating nature of the process, evidenced in this simulation, amplifies errors of steady-state calculations in dynamic simulations. Therefore, the accurate simulation of the process requires assessment of this issue.

As there is no specific equation in the system described in Equation 3.7 directly associated with this output variable, the adopted solution was to introduce a proportional controller with pairing $x_{v,b}$ x h_{flash}, which is known to stabilize the dynamic behavior, and obtain the steady-state value as the resting position of $x_{v,b}$.

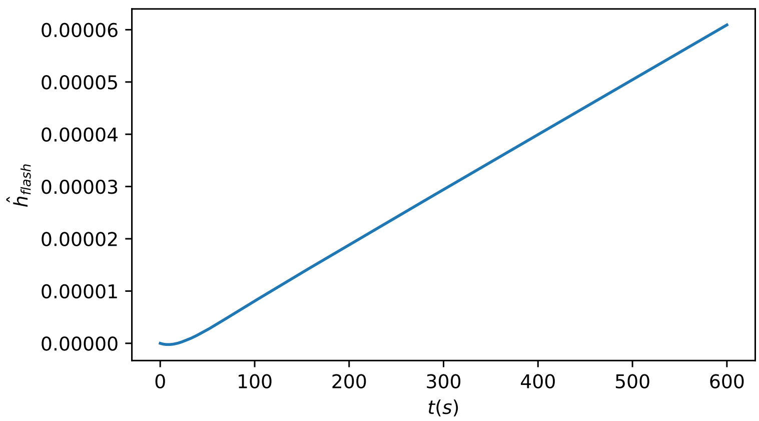

Figure 5.4: Dynamic simulation of the brute steady state

The chosen tuning to this calculation is arbitrary, and the steady-state value can be obtained regardless of it, as long as enough simulation time is used.

This closed-loop dynamic simulation is presented in Figure 5.5. It can be seen that a offset is produced, due to the absence of integral action in the controller. Despite that, stable behavior is observed, and a steady state is achieved. This steady state was then simulated in an open-loop configuration, to confirm that this solution is acceptable.

The dynamic simulation using the refined steady state as initial condition is presented in Figure 5.6. It can be seen that the temporal variations have been reduced with the solution refinement, and the numerical residue, although still present, can now be considered acceptable.

Table 5.3 presents the steady-state results, after refinement. This result was used as the initial condition for all following dynamic simulations, along with values described in Table 4.1 and Table 4.2.

Other important study to be done before step simulations are carried out is that regarding the regularization function (Equation 4.1) parameter Δt_{set}. Since R is related to the tolerance of input variable at the step start and end, it was kept constant along all simulations. Figure 5.7 presents a sensitivity analysis of Δt_{set} to an arbitrary step.

The difference of responses is perceptible for $\Delta t_{set} = 10$ s, even for the input profile. The gain is essentially the same for all simulations, and the differences are limited to the dynamic behavior. A zoomed view from step start is shown at Figure 5.8.

The difference of input profiles is not visible for $\Delta t_{set} < 1$ s, due to sampling time. But, as sampling time does not affect simulation precision, a difference between profiles for $\Delta t_{set} = 1$ s and $\Delta t_{set} = 0.1$ s can be seen in the third decimal place. Another reduction in Δt_{set} did not produce significant effects on the step response,

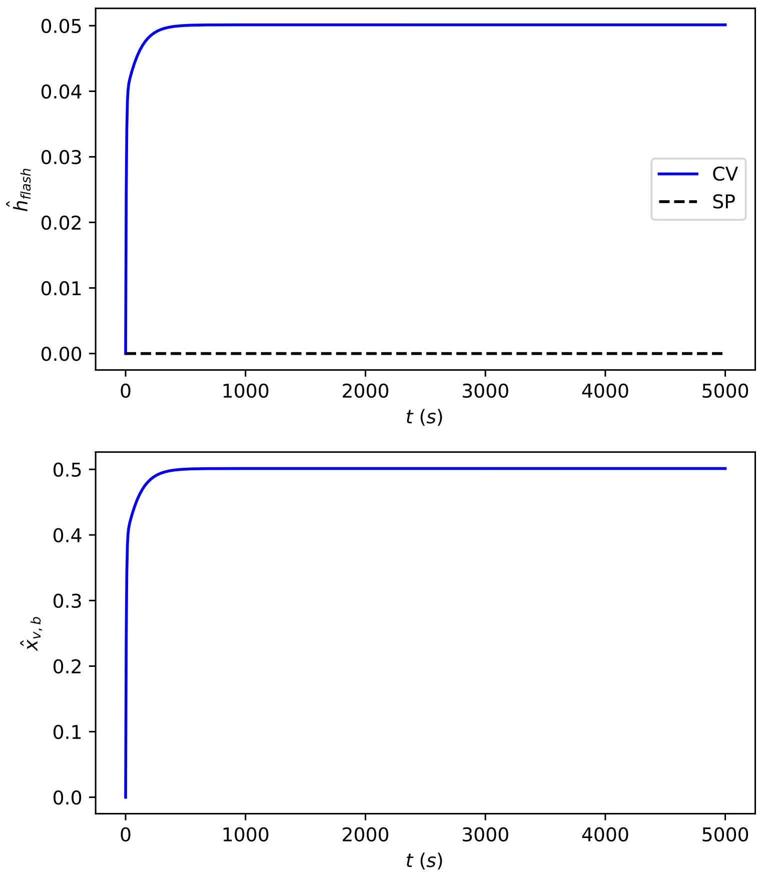

Figure 5.5: Closed-loop dynamic simulation - steady state refinement

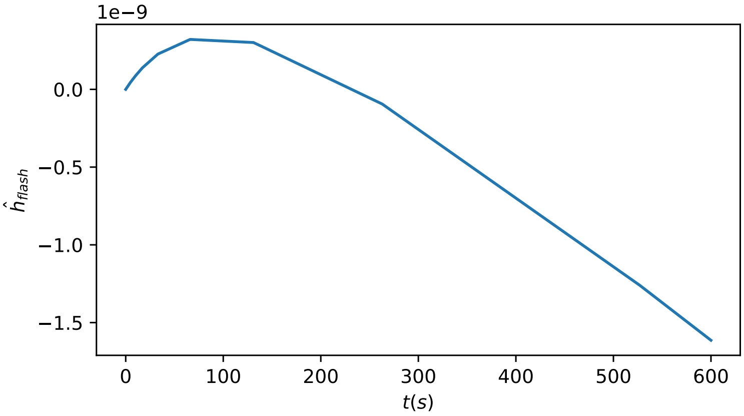

Figure 5.6: Dynamic simulation of the refined steady state

Table 5.3: Steady state results

Parameter		Value	Unit	Description
	1	-6.4567	10^7 J	
$E_{c,k}$	2	-6.0425	10^7 J	Cold side energy holdup
	3	-5.4825	10^7 J	
	1	-1.4160	10^7 J	
$E_{h,k}$	2	-1.2629	10^7 J	Hot side energy holdup
	3	-1.0557	10^7 J	
	1	314.132	K	
$T_{c,k}$	2	315.602	K	Cold side stage temperature
	3	317.773	K	
	1	326.389	K	
$T_{h,k}$	2	332.874	K	Hot side stage temperature
	3	342.018	K	
	H_2O	2.1638	kmol	
	CO_2	162.3078	kmol	
	CH_4	10.3863	kmol	
N	F_1	11.74932	kmol	Reservoir composition
	F_2	9.5855	kmol	
	F_3	11.2516	kmol	
	F_4	8.9364	kmol	
U^e		-2.2175	10^9 J	Flash drum extensive internal energy
T_1		312.971	K	Inlet valve discharge temperature
T_d		354.702	K	Pump discharge temperature
$T_{h,i}$		354.874	K	Top valve discharge temperature
T_{bot}		329.368	K	Bottom valve discharge temperature
W		4.246	MW	Pump power
$\dot{Q}_{aqu}$		5.676	MW	Drum heating rate
$\dot{Q}_{resf}$		-5.462	MW	Condenser cooling rate
x_v		0.5	-	Inlet valve opening
$x_{v,t}$		0.49828	-	Top valve opening
$x_{v,b}$		0.50131	-	Bottom valve opening

therefore Δt_{set} was fixed at 0.1 s for all dynamic simulations.

5.2 Plant stabilization

The open-loop system exhibits an integrating behavior, due to the presence of a drum in which both outlets and inlet can be freely manipulated. As such, in order to use a predictive control strategy, it is necessary to modify the system dynamics with the introduction of a controller, so that the system becomes stable. SOUZA (2018) successfully stabilized the system by implementing a PI controller with the pairing $x_{v,b}$ x h_{flash}, so it was also used in this work.

The SIMC rules of PID tuning for integrating systems are given by Table 5.4.

Table 5.4: SIMC PID tuning (SKOGESTAD, 2003)

$G(s)$	k_c	τ_I	τ_D
$\frac{k_p}{s}$	$\frac{1}{k_p \tau_c}$	$4\tau_c$	-

Identification results for the pair $x_{v,b}$ x h_{flash} are displayed in Figure 5.9. The identified transfer function is presented on Equation 5.1, consisting of an integrating system, with some dynamic compensation.

$$G(s) = \frac{-0.0032126(75.59s + 1)}{s(16.61s + 1)} \tag{5.1}$$

In order to use the SIMC rules for an integrating system, the lead-lag dynamic compensation term was neglected. The proposed tuning was then tested in servo and regulatory simulations, presented in Figure 5.10 and Figure 5.11, respectively. Regulatory simulation was performed with a step disturbance on reservoir pressure of +1 MPa.

It can be seen that, due to the presence of integral action in the controller, offset is eliminated in both servo and regulatory simulations. In servo simulation, setpoint is changed by manipulating valve opening, but the valve returns to its resting position as time passes, which is compatible with integrating systems behavior, and reinforces the discussion presented regarding Figure 5.5. On the other hand, the valve resting position is changed in the presence of input disturbances.

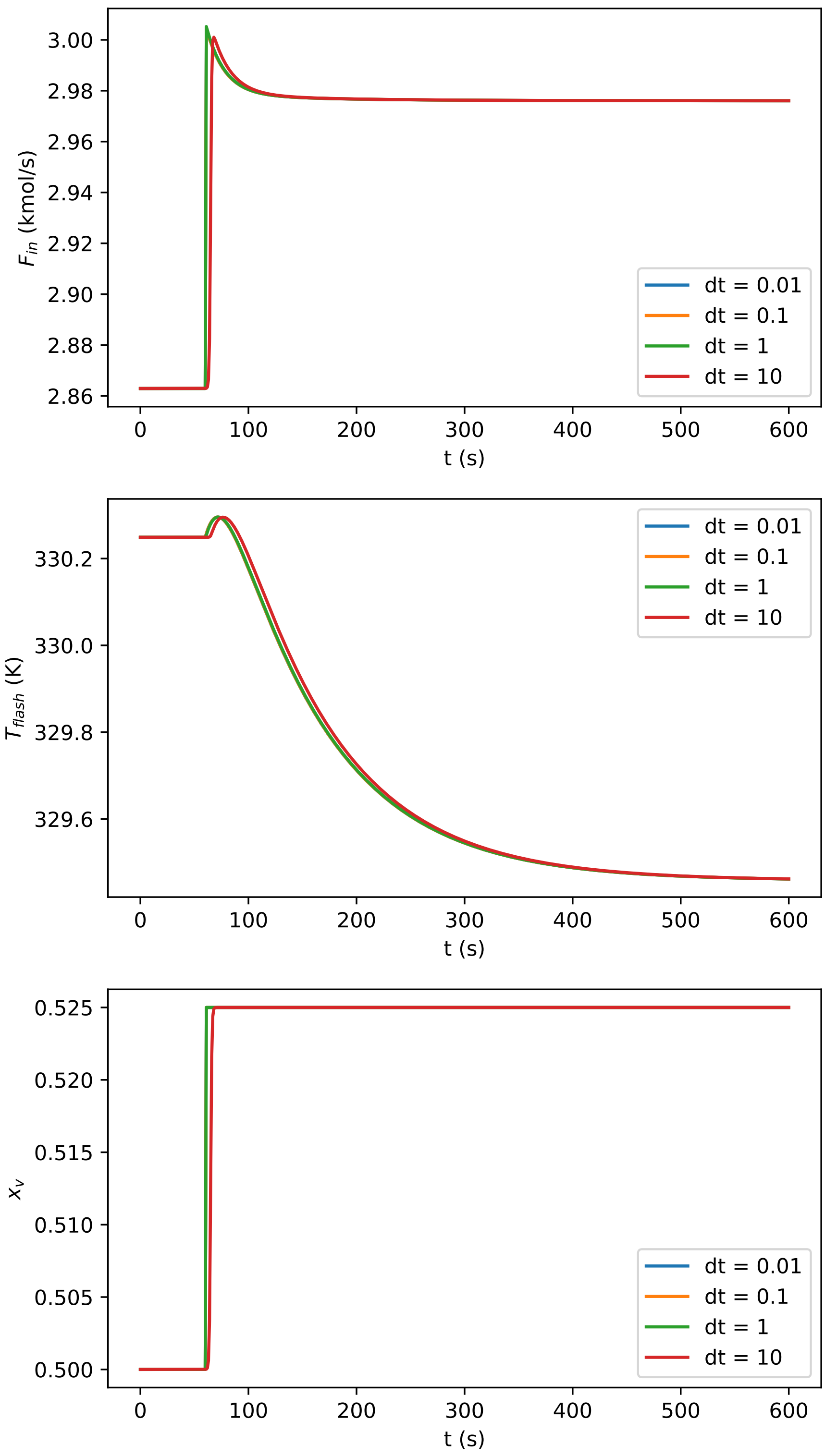

Figure 5.7: Effect of Δt_{set} in simulation results

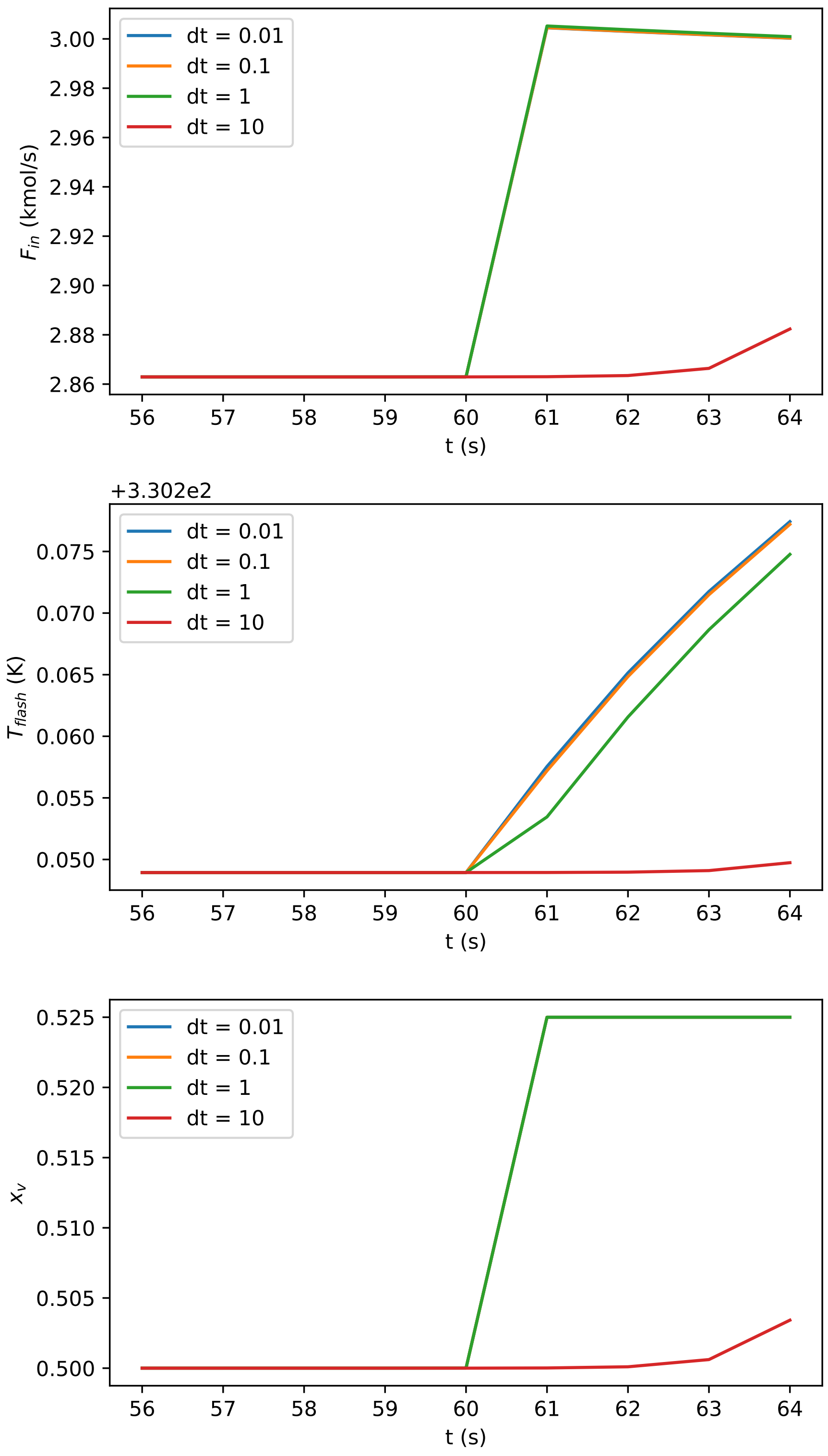

Figure 5.8: Effect of Δt_{set} in simulation results - zoomed view

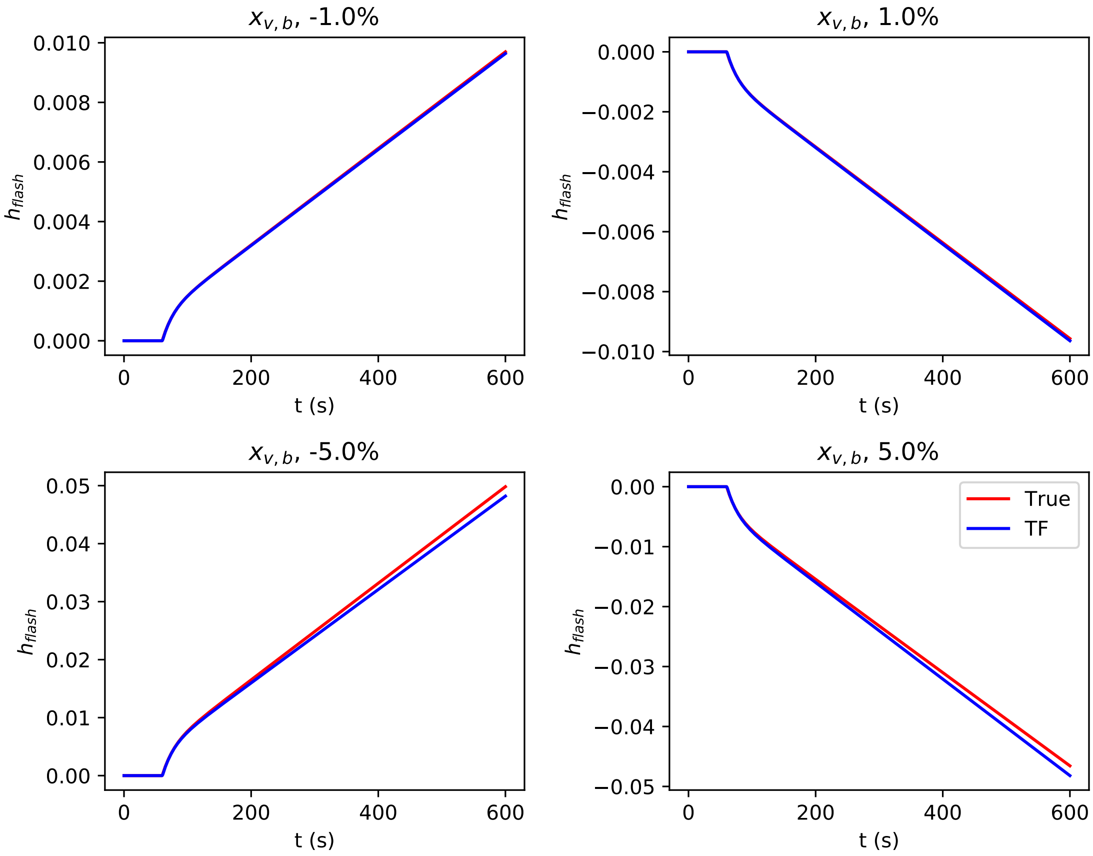

Figure 5.9: Identification results for the pair $x_{v,b}$ x h_{flash}

5.3 Model identification

The different input steps were simulated, in order to obtain representative dynamic data from the considered stationary point. Using these step response data, continuous-time transfer functions were then estimated. These results are fully presented in Appendix A, and some important remarks are brought to attention here.

System nonlinear nature is explicited by the identification of several input-output pairs, e.g. Figure A.1. Although the dynamic behavior is well captured, static gain is not constant, specially for high step magnitudes.

Flow produced by a valve is intimately related to its opening. Dynamically, the flow has a instantaneous response to the valve opening. This was observed in all valve opening-flow pairs (Figure A.5, Figure A.8). Due to this instantaneous behavior, these transfer functions had to have the same number of poles and zeros. This was taken into account when proposing a transfer function for not only these cases, but other cases with an immediate response, such as temperatures involved with the instantaneous condenser.

Identification results, in form of zeros, poles and gain, are shown in Table A.1, for the sake of space.

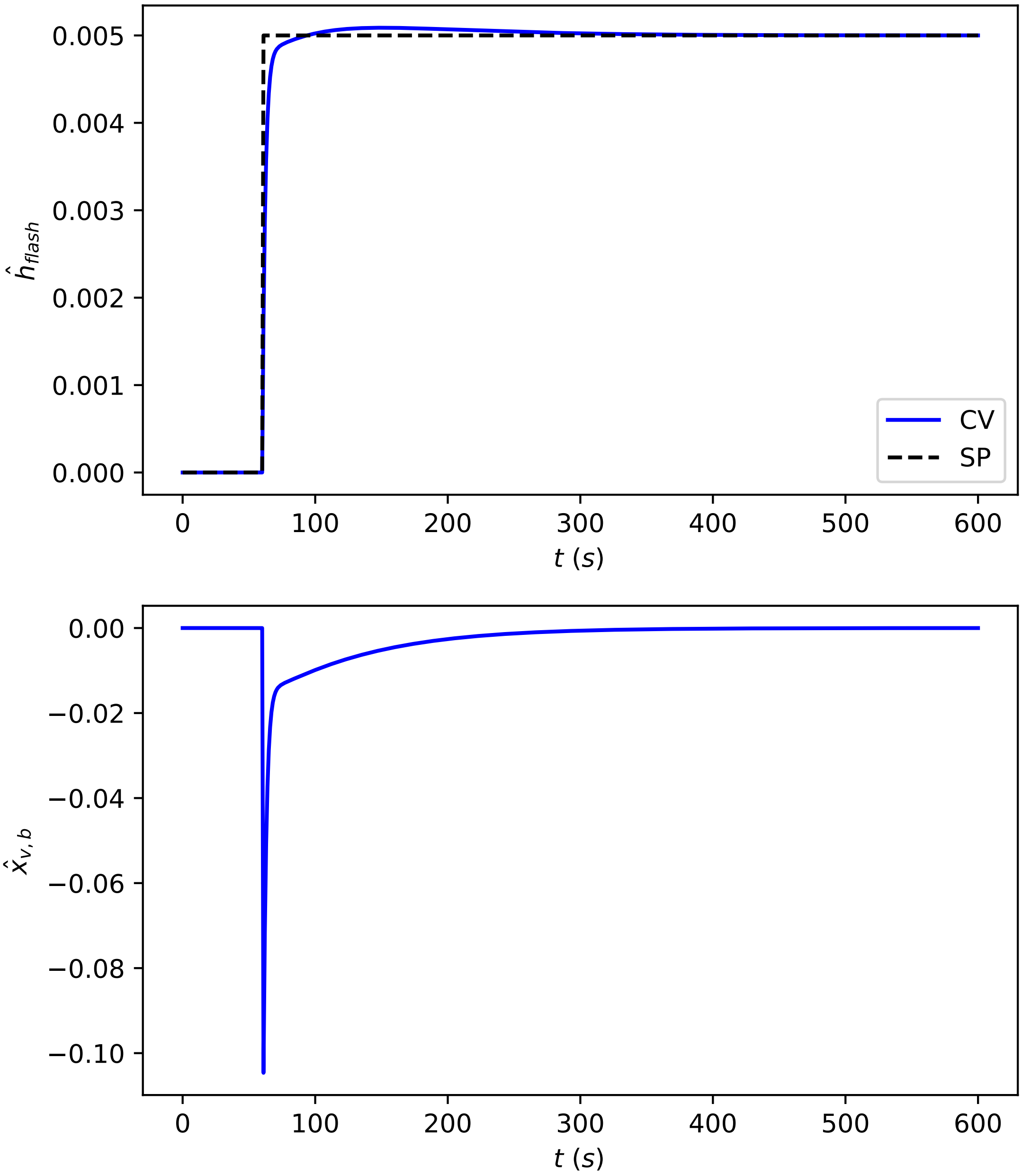

Figure 5.10: Servo test for stabilized process

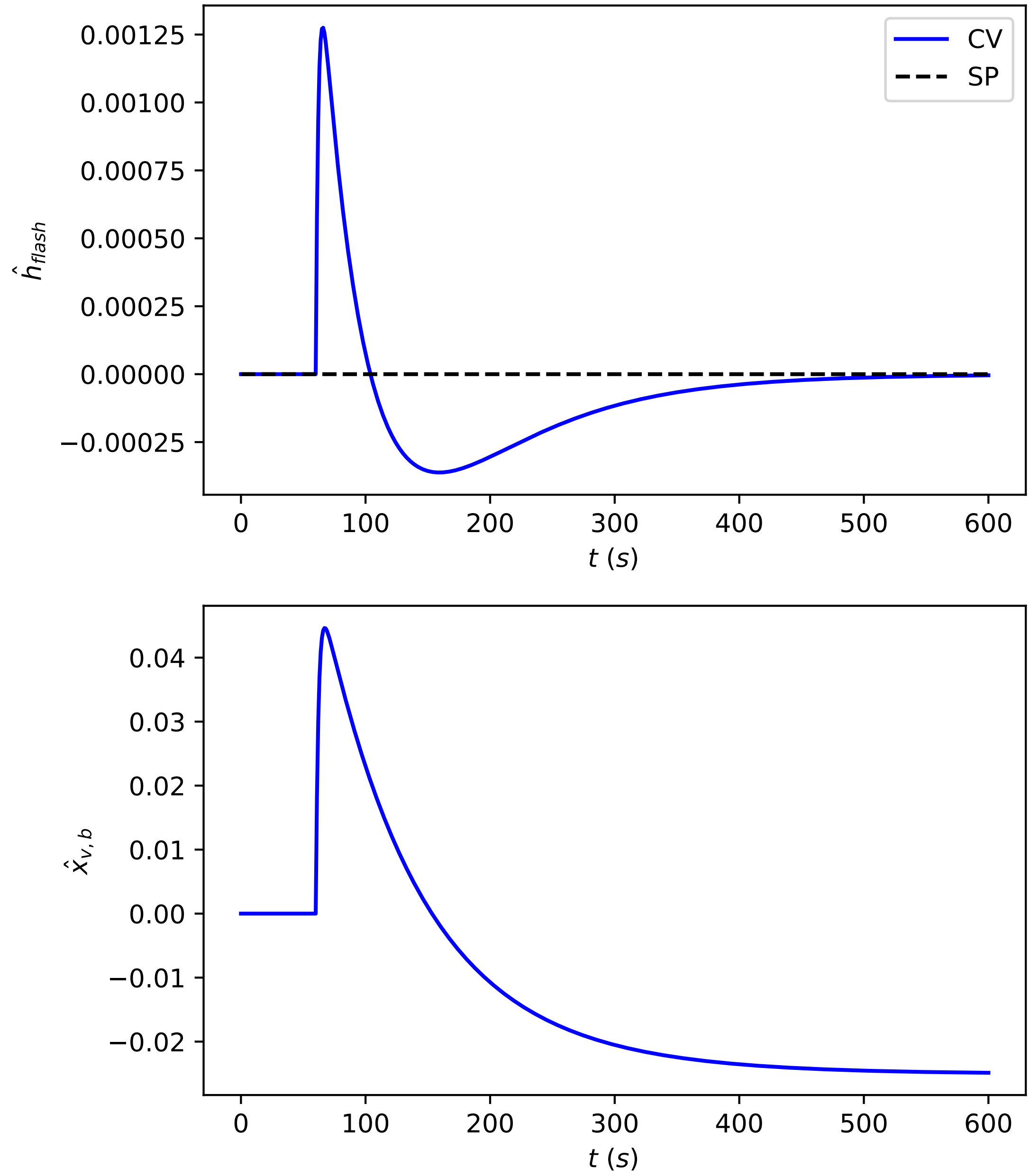

Figure 5.11: Regulatory test for stabilized process

5.4 Closed loop simulation

The model predictive control is a centralized control strategy, and its main appeal lies in interacting systems. By using an internal model, this control strategy tries to account for interaction, designing optimal control actions planning. As such, it was necessary to identify a simple model which represents the dynamic behavior of the system.

Continuous-time transfer functions were estimated due to their conciseness in representing dynamic behavior. However, the use of a Laplace-domain model for predictive control and state estimation is not straightforward. For this reason, this model was converted to a discrete state-space formulation.

The first step was to write the continuous-time transfer functions into a continuous-time state space form. As the number of freedom degrees rises in the state-space representation, there are infinite forms of representing a transfer function as a state-space system. Thus, a form can be chosen so that specific characteristics of the system can be exploited.

The chosen continuous state-space formulation was the diagonal canonical form. Any transfer function can be written in this form, as long as there are not any poles with multiplicity greater than 1. The main advantage of this form is that the state matrix is diagonal, with its diagonal being composed by the transfer function poles. OGATA (2010) describes how to write this state-space form for a SISO transfer function, relying on its partial fractions decomposition. This decomposition can be easily performed using the residue theorem (KREYSZIG *et al.*, 2011).

In order to extend this concept for MIMO systems, matrices were simply concatenated, using blocks of zeros when necessary. As each transfer function has its own set of poles, a diagonal state matrix is still possible, as long as all other matrices are assembled accordingly.

After the continuous-time state-space model is obtained, the next necessary step is the discretization of the equation system. Even though this step is not necessarily needed, as the Kalman Filter theory was developed also for continuous-discrete systems (see Kalman-Bucy filtering, SIMON (2006)), the use of a continuous-time model implies on the integration of an ODE system. Discrete-time systems, on the other hand, incorporate the result of analytical integration of the system, avoiding the use of such numerical integrators. This is specially advantageous in the implementation of MPCs, which require intensive evaluations of the model. Discrete-time models can also be incorporated in the MPC formulation, generating a QP optimization problem (TRIERWEILER and SECCHI, 2000).

The discretization of a linear state-space system, as described by SIMON (2006), requires the inversion and exponentiation of the state matrix. As such, good conditioning of state matrix is necessary. This is guaranteed by the use of the diagonal canonical form.

Some closed-loop simulation results are now presented. For the base case, tuning parameters for the controller described by Equation 4.5 were as presented in Table 5.5. For the filter, state covariance matrix Q was set to I, measurement covariance matrix was set to a diagonal matrix composed by the square of 0.1% of the output steady-state value, and initial covariance matrix P_0^+ was set to $100I$, I being an appropriately sized identity matrix.

Table 5.5: Controller tuning parameters for the base case (units consistent with values of Table 4.2 and Table 5.3)

q_i		s_i	N_p	N_c
F_{in}	$P_{flash}, T_{flash}, T_2$			
10	1	10	75	10

The simulation of a set-point change in the flash drum pressure is presented in Figure 5.12. System settling occurred at around 600 seconds, which justifies the chosen prediction horizon. This set of parameters results in a good balance between manipulation effort and settling time, as manipulated variables are smoothly guided to their final values.

It is necessary to emphasize the role of Kalman filtering in the quality of closed-loop response. Not only KF is responsible for correct state estimation, it is also responsible for model correction along the prediction horizon. To illustrate this, consider the simulation in Figure 5.13, in which KF was not used. The internal model attains the desired setpoint values, but due to modeling errors, the actual process does not attain these values, and an offset is produced.

This result illustrates the importance of state estimation in process control. Now take the simulation in Figure 5.14, in which KF is used for state estimation, but the model correction update in Equation 4.7 is not implemented.

Even though state estimation is performed reasonably, as can be seen from the proximity of internal model and process output values, the internal model remains predicting the same steady state as in Figure 5.13, leading to offset. This means that state correction is not enough to account for modeling errors. As the steady state itself needs to be corrected, KF calculations should also be accounted in the dynamic model, in similar philosophy of the feedback strategy in DMC.

In the sense of controller tuning, a more aggressive configuration was tested, using the parameters given in Table 5.6.

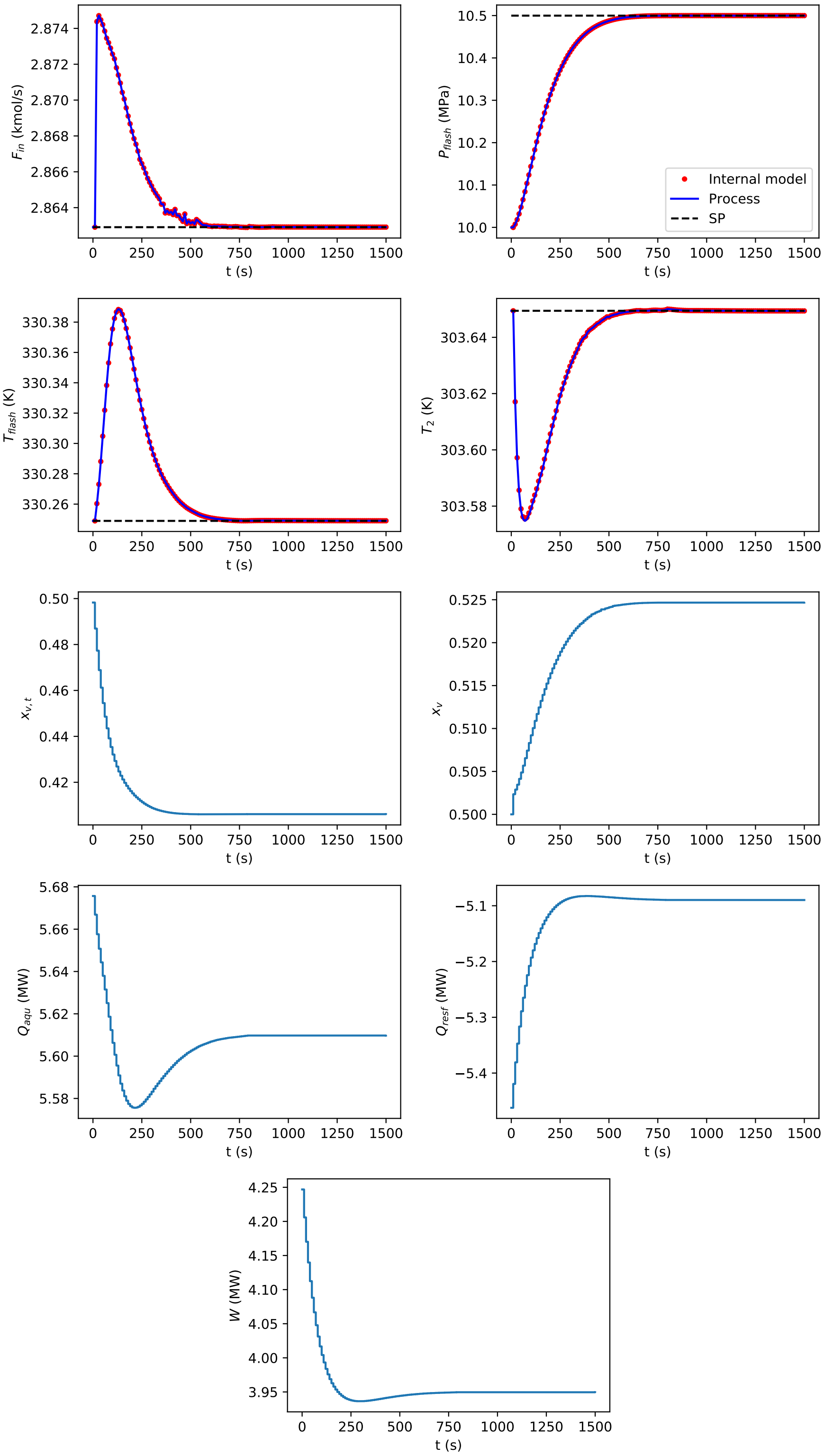

Figure 5.12: Closed loop simulation - flash pressure setpoint change - base case

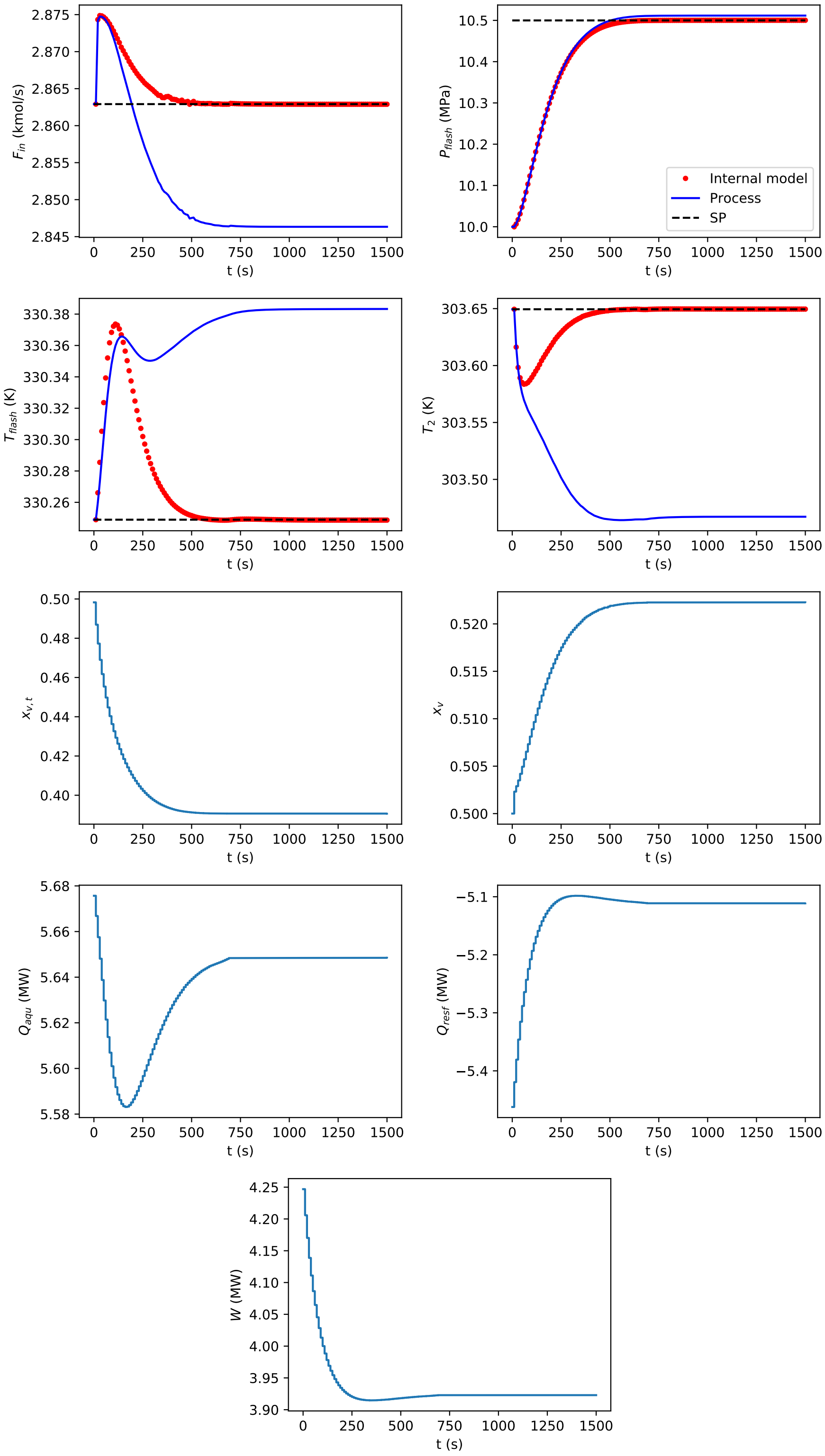

Figure 5.13: Closed loop simulation - flash pressure setpoint change - no filter

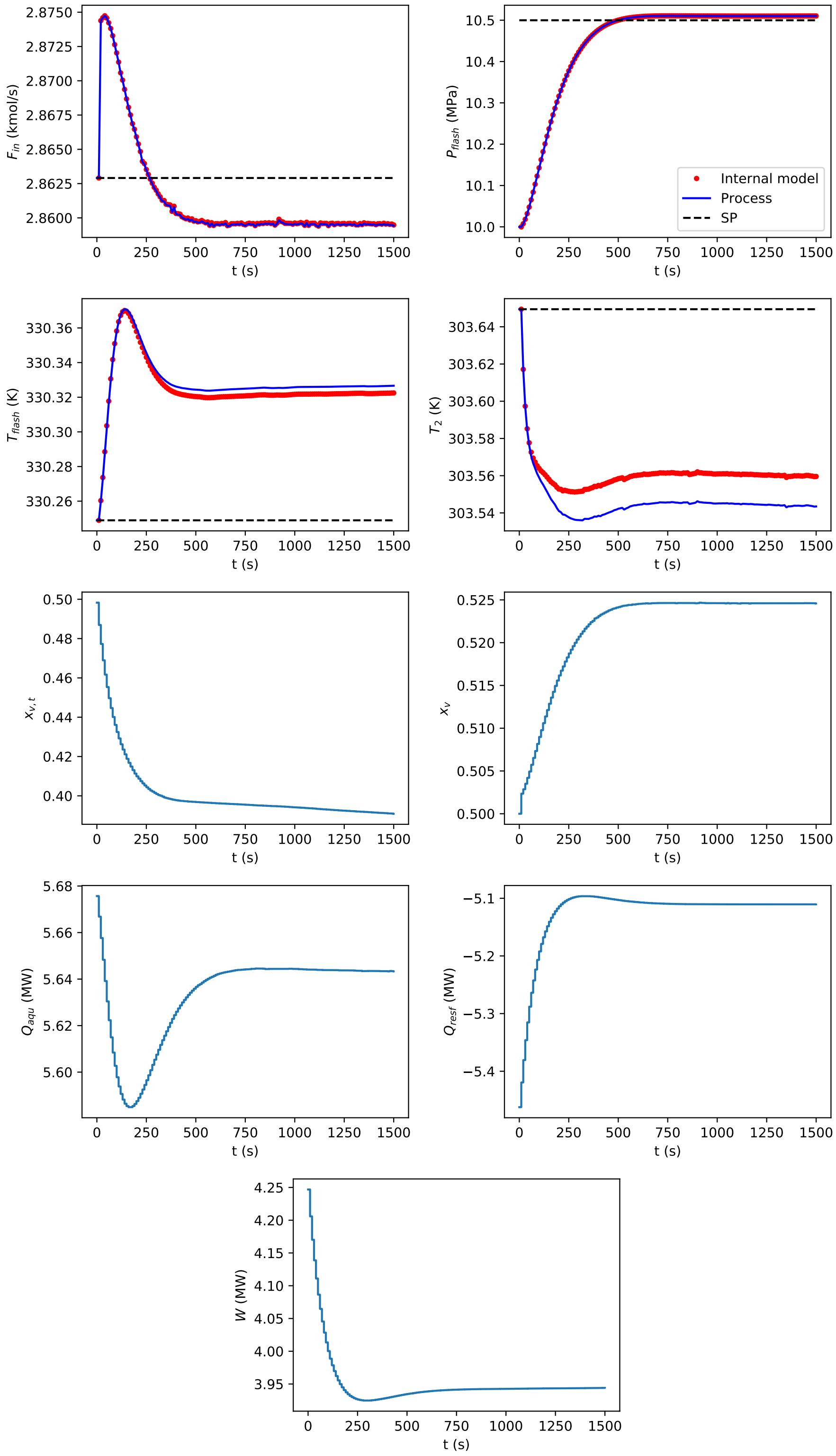

Figure 5.14: Closed loop simulation - flash pressure setpoint change - no model update

Table 5.6: Controller tuning parameters for the aggressive tuning (units consistent with values of Table 4.2 and Table 5.3)

q_i		s_i	N_p	N_c
F_{in}	$P_{flash}, T_{flash}, T_2$			
10	1	1	75	10

The simulation of this configuration is given by Figure 5.15. It can be seen that the settling time was reduced, at the expense of manipulation softness. It can also be noted that a different resting position of W and $x_{v,t}$ was attained, when compared to Figure 5.12. This is natural, since the controller has 5 freely manipulated variables, but only 4 controlled variables. In this case, there is a compensation of pump power and valve opening in the flash drum vapor outflow. To obtain a well-posed problem, there should be a term in the objective function that accounts for the resting position of the manipulated variables. For example, it could be established that the top valve opening should rest at half of its course, so it can be used for dynamic compensation when necessary.

Back to the base case, in order to check if constraints were implemented correctly, simulations with strict constraints were performed. Figure 5.16 presents a simulation in which maximum valve step was 0.001. A slower settling can be observed when compared to the base case, as well as a different resting position of manipulated variables.

Regarding constraints on manipulated variables, Figure 5.17 presents a simulation result where $x_{v,t} > 0.45$ was added as a constraint. This constraint was satisfied without any loss on controlled variables, due to the compensation in W.

This behavior was not observed in Figure 5.18, in which the implemented constraint was $x_v < 0.51$. The constraint was satisfied, but setpoints could not be attained, since there is no possible compensation among the other variables.

It is known that hard constraints on output variables may lead to infeasible optimization problems, so this was not tackled in this work. Instead, if necessary, constraints on output variables can be implemented using soft constraints (with the use of penalty factors).

Figure 5.19 shows a simulation in which all setpoints were changed at the same time. It can be seen that P_{flash} is the slowest variable to attain the setpoint value, which could be changed with a different controller tuning. Considering the open-loop dynamics, this settling time was considered reasonable.

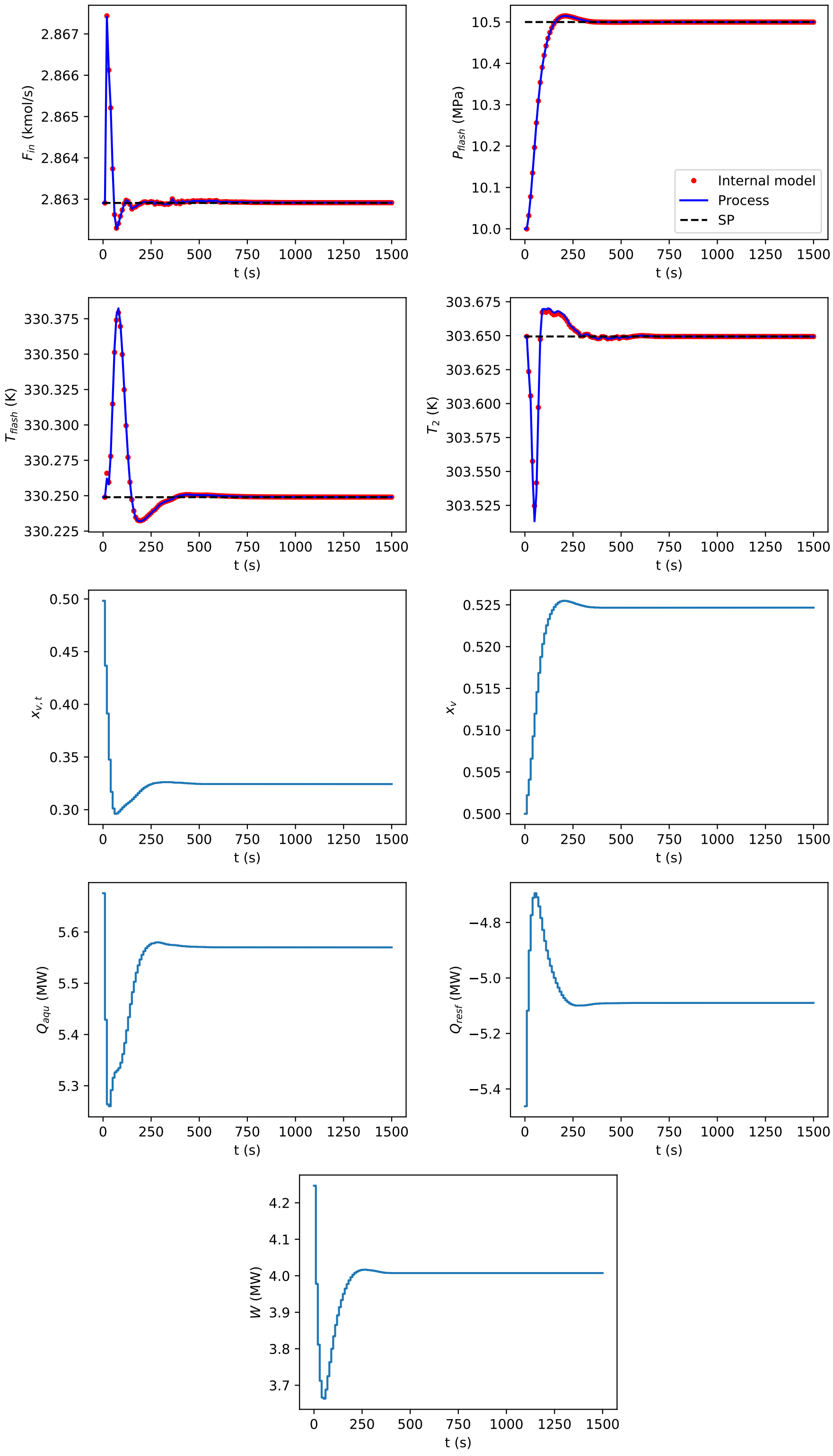

Figure 5.15: Closed loop simulation - flash pressure setpoint change - aggressive tuning

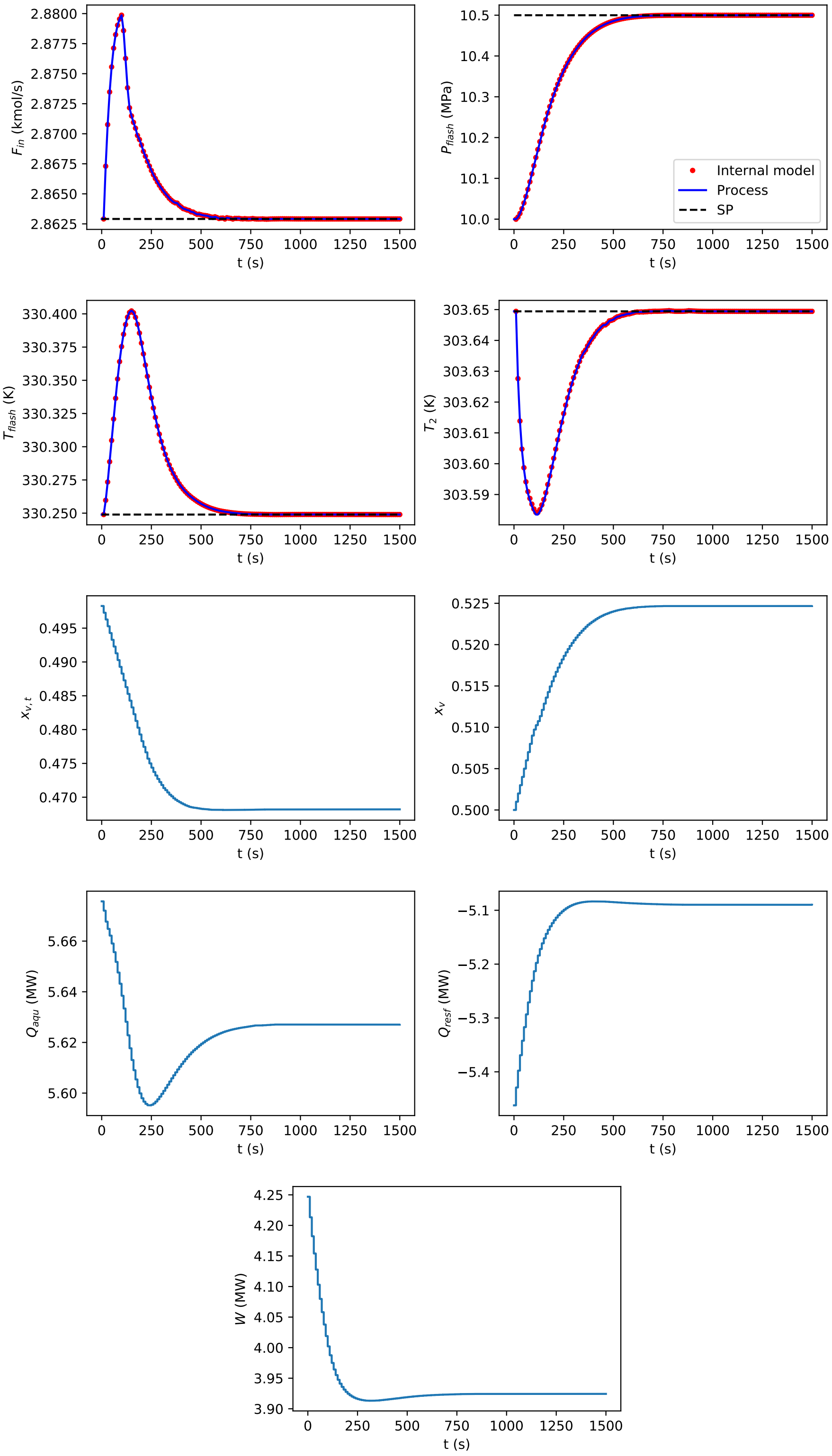

Figure 5.16: Closed loop simulation - flash pressure setpoint change - Δu constraint

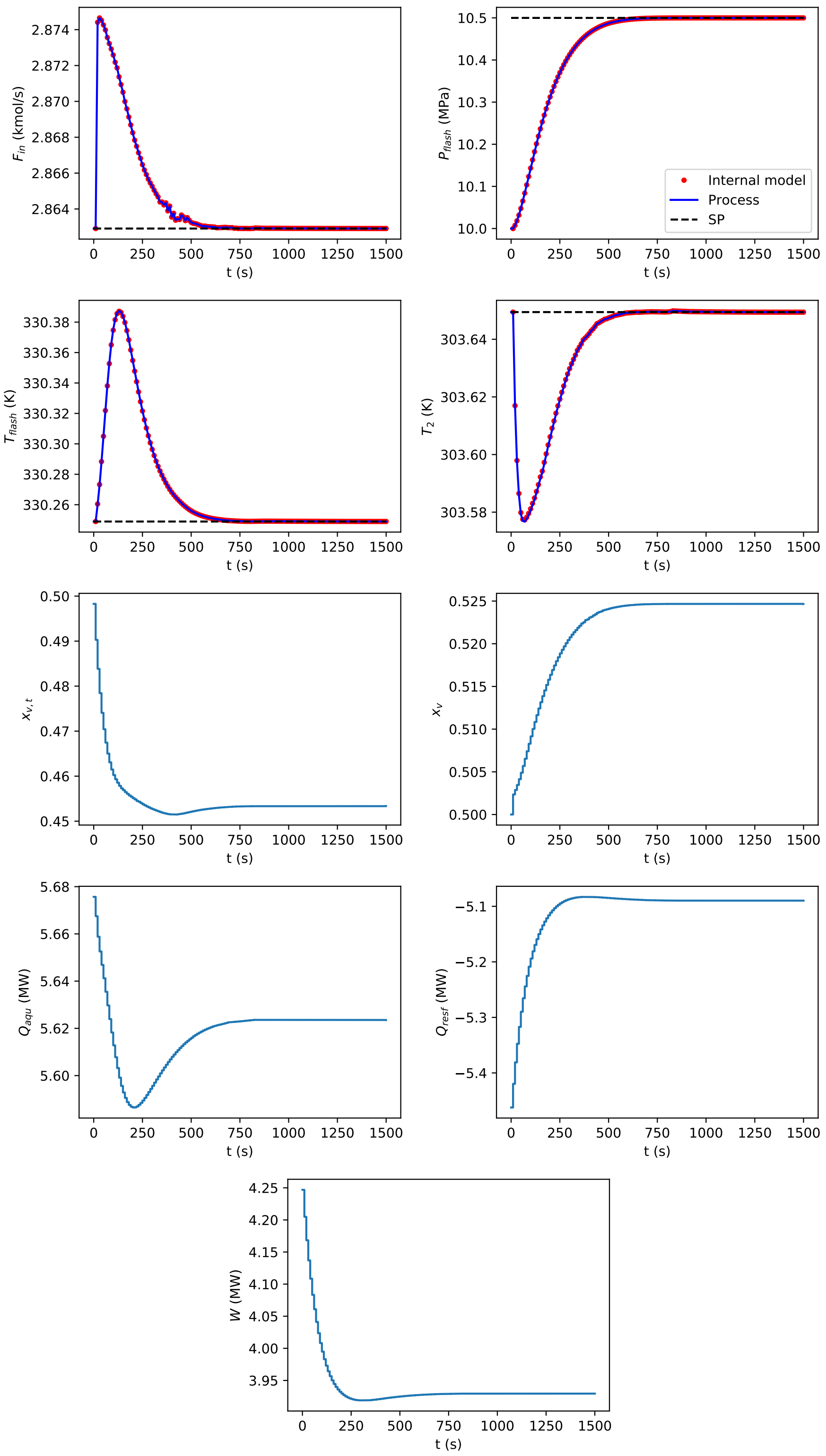

Figure 5.17: Closed loop simulation - flash pressure setpoint change - $x_{v,t}$ constraint

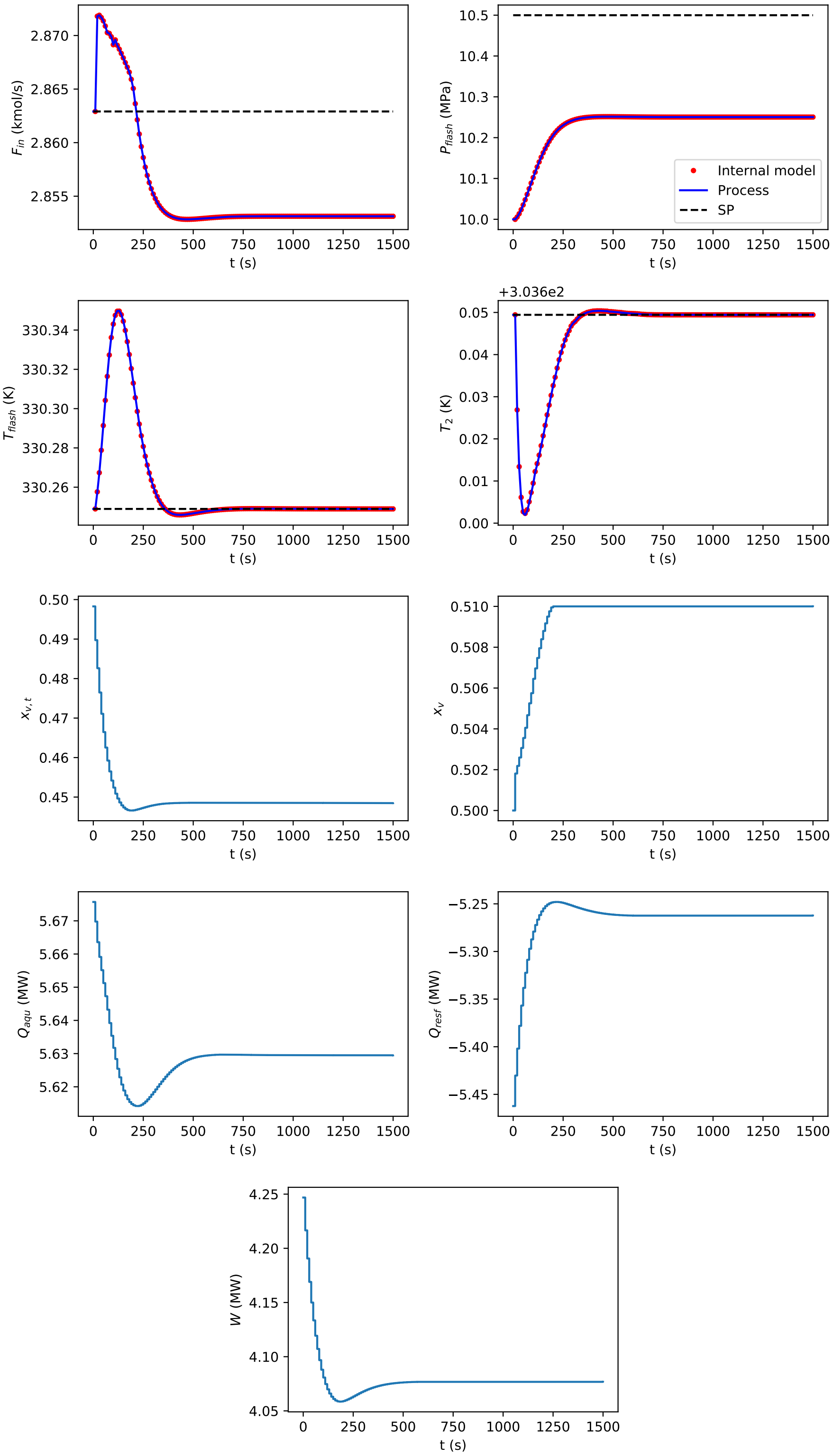

Figure 5.18: Closed loop simulation - flash pressure setpoint change - x_v constraint

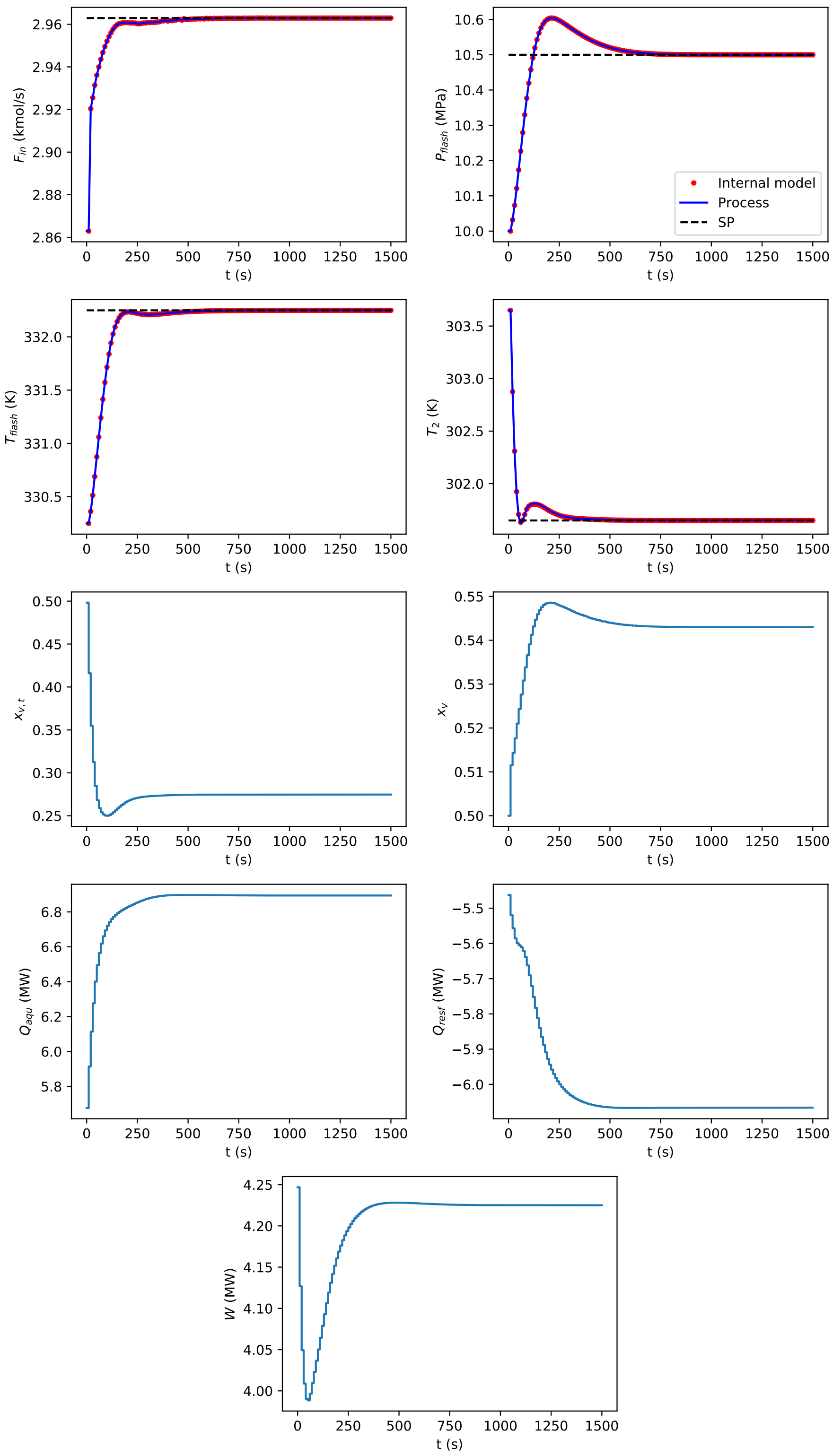

Figure 5.19: Closed loop simulation - all variables setpoint change - base case

5.5 Degradation model implementation

The proposed degradation model establishes a direct relation between equipment operation effort and its level of degradation. Unlike other common models, such as Gamma processes, degradation increments are not dependent explicitly on time. Instead, degradation increments are dependent on current state and process variables. Another level of detail in the model would be to consider degradation level as a function of feed composition, but this is beyond the scope of this work.

The simulation of degradation model in process nominal condition, which was chosen arbitrarily, yields profiles shown in Figure 5.20. The randomness of the degradation process is explicited, but also its monotonicity. Both are desired characteristics in the design of a fatigue degradation model.

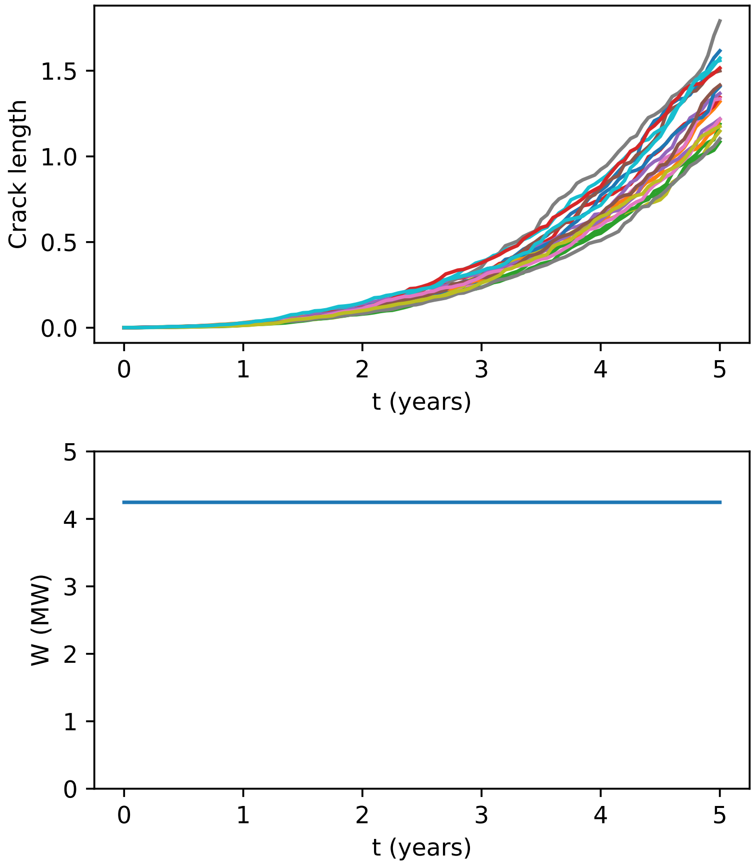

Figure 5.20: Degradation simulation in nominal condition

The degradation dependence with the manipulated variable is shown in Figure 5.21, in which is presented a simulation with decreasing pump power.

This simulation shows an interesting asymptotic behavior of the proposed model, and that is the stopping of degradation process when the pump is progressively shut down. Besides this being a hypothetical model, and experimental datasets are needed to test model predictive capability, these preliminary results encourage

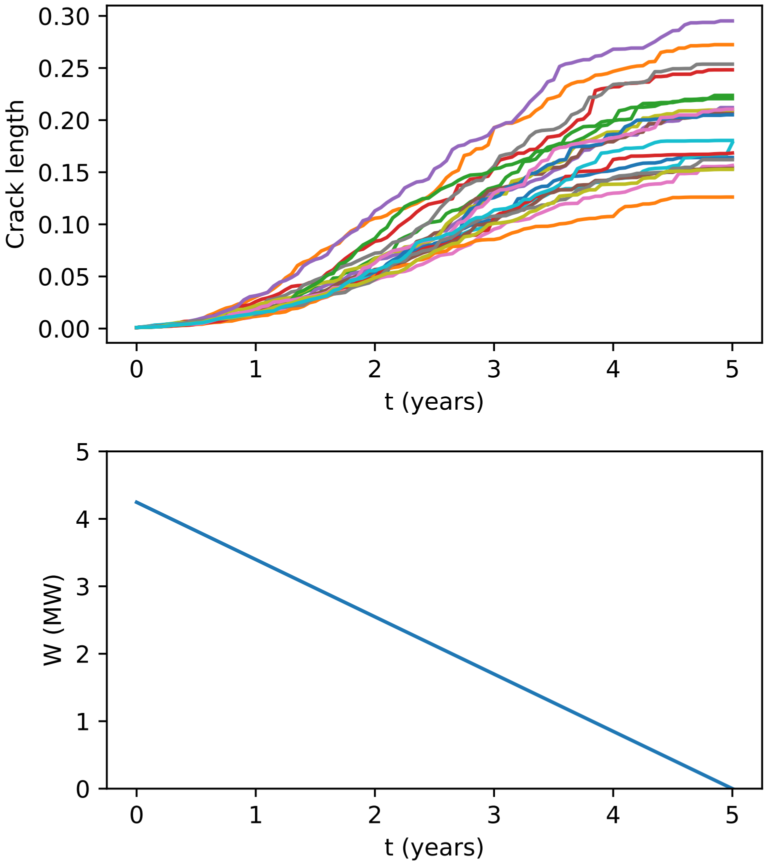

Figure 5.21: Degradation simulation in nominal condition

further investigation.

5.6 Crack length estimation

Using a representative model of the degradation process, online state estimation was performed using a particle filter. While on the process control a simple tool such as the Kalman filter was used, this problem requires a more sophisticated tool. This is due to the absence of a representative linear model of the process, and the importance of keeping the statistical information of the model.

In the first set of simulations, a constant pump power ($W(t) = 4.246$ MW) was considered. Figure 5.22 and Figure 5.23 present the results of the SIS filter implementation. The problem of weight degeneracy can be seen progressively worsening the conditioning of the particle set, to the point of only one particle having a relevant weight, and then to the point of this particle degenerating and consequently all particles ending up with indefinite weights.

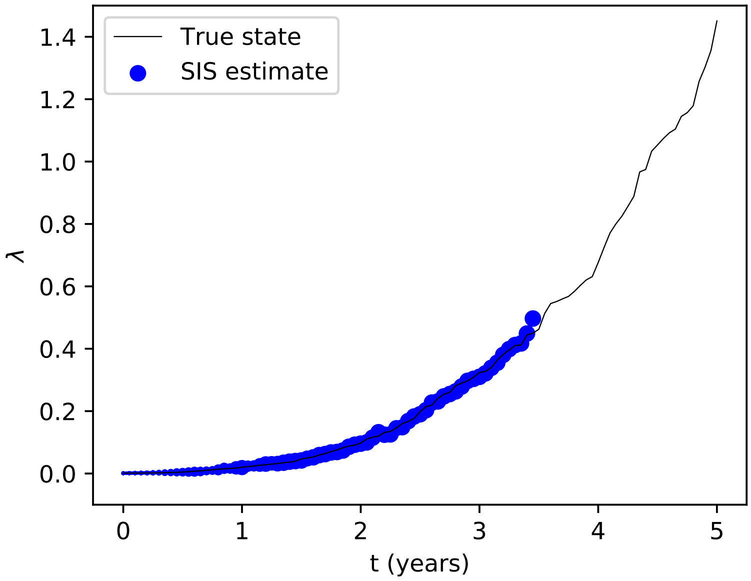

Figure 5.22: Crack length online estimation - SIS filter, *a priori* distribution (dot size represent particle weight)

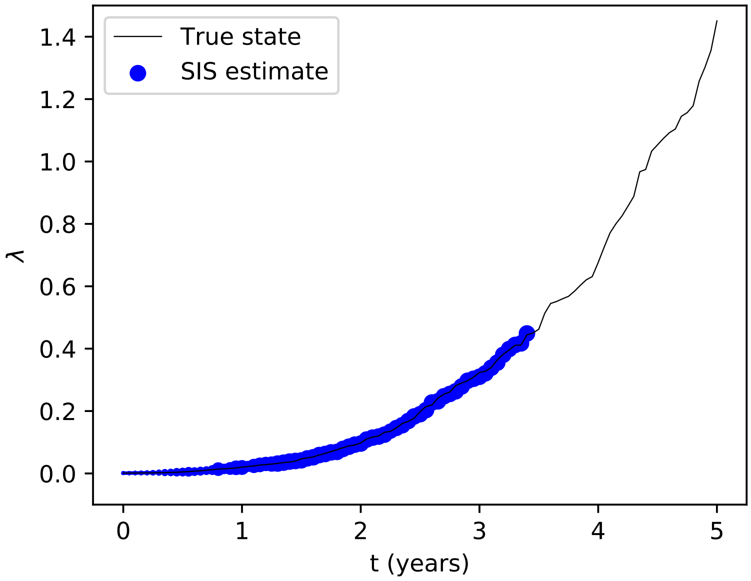

Figure 5.23: Crack length online estimation - SIS filter, *a posteriori* distribution (dot size represent particle weight)

Figure 5.24 and Figure 5.25 present the results of the SIR filter implementation. It can be seen that, by implementing the resampling step, distributions keep well-conditioned, even for longer times. Also, measurement updating effectively narrows the samples distribution, filtering the most likely updates from all *a priori* estimates.

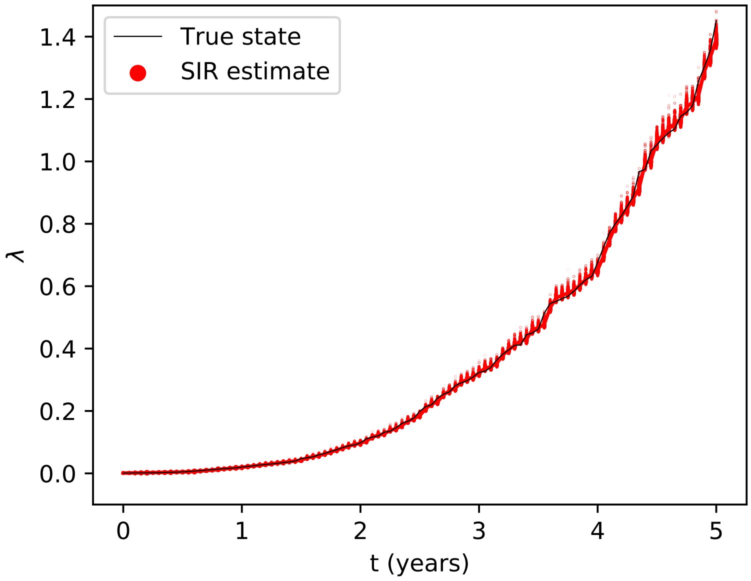

Figure 5.24: Crack length online estimation - SIR filter, *a priori* distribution (dot size represent particle weight)

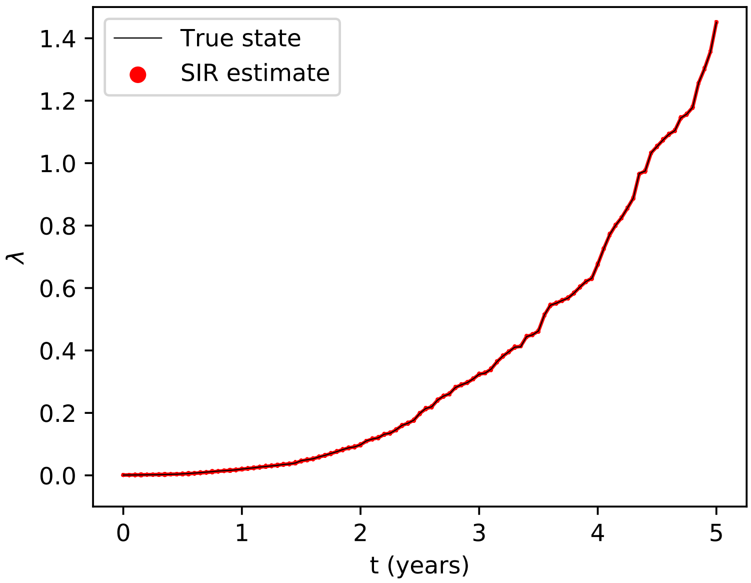

Figure 5.25: Crack length online estimation - SIR filter, *a posteriori* distribution (dot size represent particle weight)

All parameters from the stochastic model were implemented in the particle filter as the real values, meaning that no influence of modeling error was considered in this implementation. The only parameter that was changed was the likelihood function standard deviation, which was chosen to be $\sigma = 0.001$. This was done solely due to numerical issues related to the very narrow real likelihood function. This led to a more permissive filter, which means that *a posteriori* distributions are wider than necessary to describe the state.

Using the particle filter framework, RUL predictions can be performed. However, as RUL prediction generally involves a high number of state transitions, leading to high-dimensional statistical problems, it is known that a distribution reconstruction using importance sampling is nearly infeasible. The chosen strategy, then, was to retrieve information about the distributions by standard Monte Carlo sampling. As a discrete-time state transition model was used, predicted RUL belongs to a discrete set. Thus, the very number of occurrences of each realization was considered as representative of the probability.

Results of RUL prediction for the SIS and SIR filters are presented in Figure 5.26 and Figure 5.27, respectively. It can be seen that, even though both filters predict consistent average values for the RUL, SIR filter RUL distribution narrows around the real RUL as time passes. This is due to the resampling step, which redistributes particles in the most likely values. As this does not happen in the SIS filter, all particles are kept, regardless of their probability, and in RUL prediction this ultimately leads to no information update, as RUL prediction ignores particles weight.

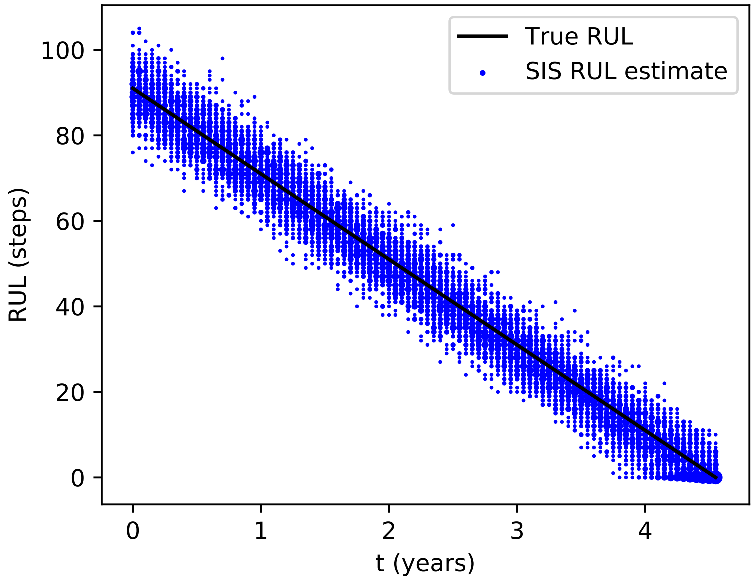

Figure 5.26: RUL prediction - SIS filter (dot size represent probability)

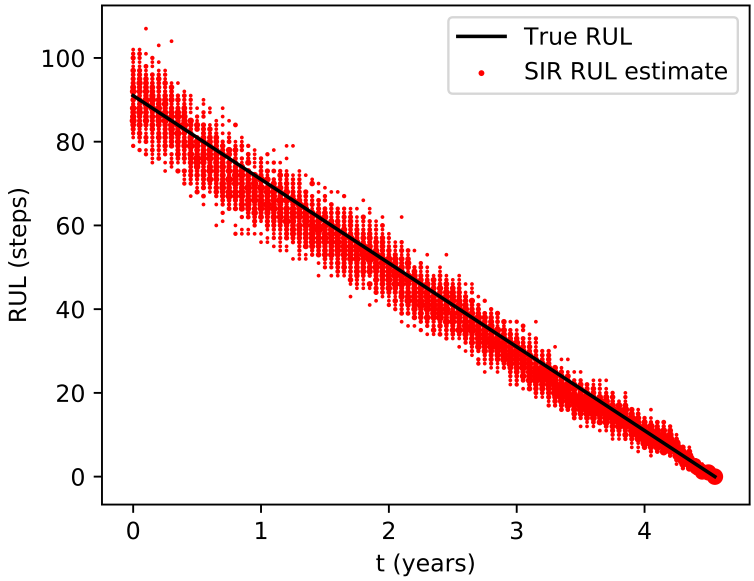

Figure 5.27: RUL prediction - SIR filter (dot size represent probability)

A set of simulations with varying pump power were also performed ($W(t) = 10.615 \times (5-t)/5$ MW, t in years). Figure 5.28 and Figure 5.29 show results for the SIS filter, and Figure 5.30 and Figure 5.31 show results for the SIR filter. Once again the failure of SIS filter is showcased, in the sense that the particles degenerate and lose statistical meaning.

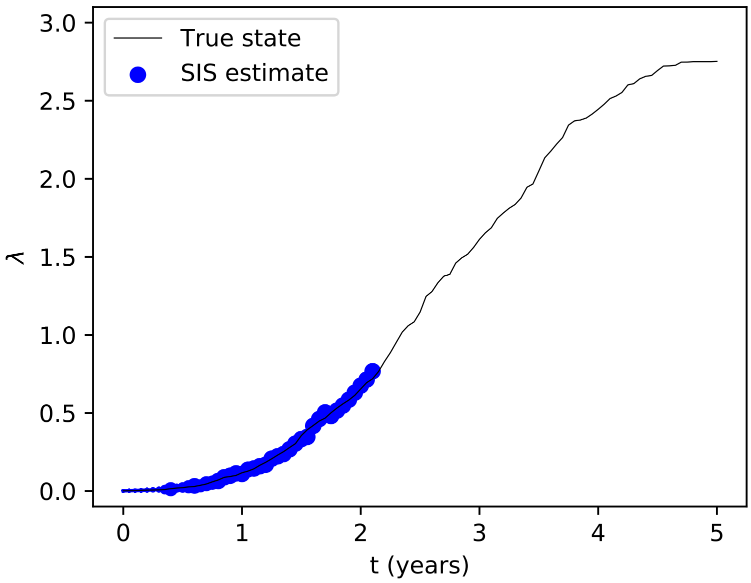

Figure 5.28: Crack length online estimation with varying pump power - SIS filter, *a priori* distribution (dot size represent particle weight)

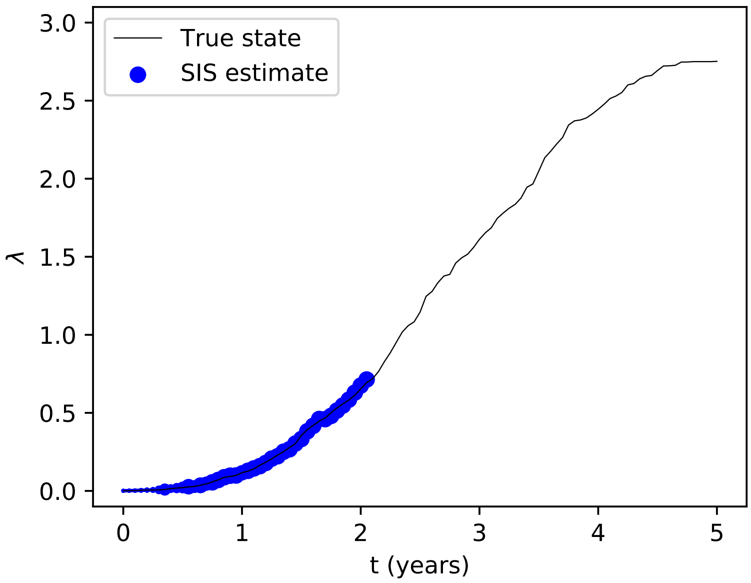

Figure 5.29: Crack length online estimation with varying pump power - SIS filter, *a posteriori* distribution (dot size represent particle weight)

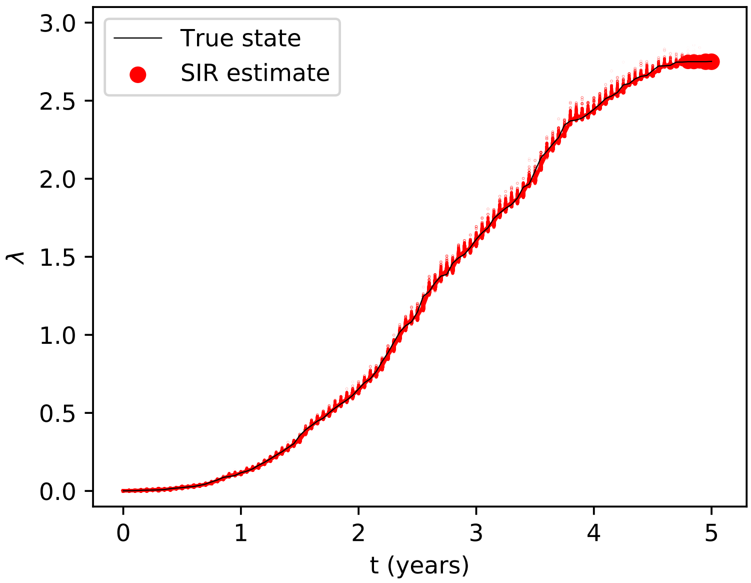

Figure 5.30: Crack length online estimation with varying pump power - SIR filter, *a priori* distribution (dot size represent particle weight)

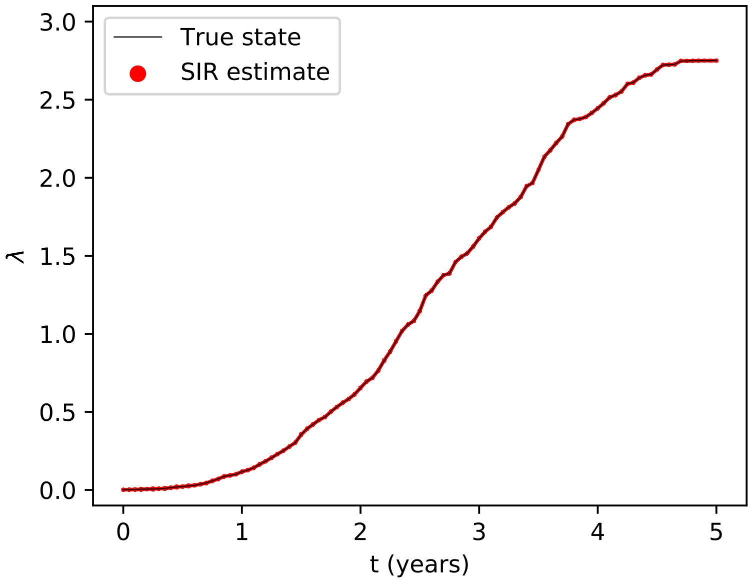

Figure 5.31: Crack length online estimation with varying pump power - SIR filter, *a posteriori* distribution (dot size represent particle weight)

In the matter of RUL prediction for this case, results for the SIR filter are given in Figure 5.32. Even though RUL prediction in early states is biased, as new measurements are incorporated, prediction progressively gets more accurate. This result encourages the use of particle filters as auxiliary tools in RUL prediction, even in the presence of modeling errors.

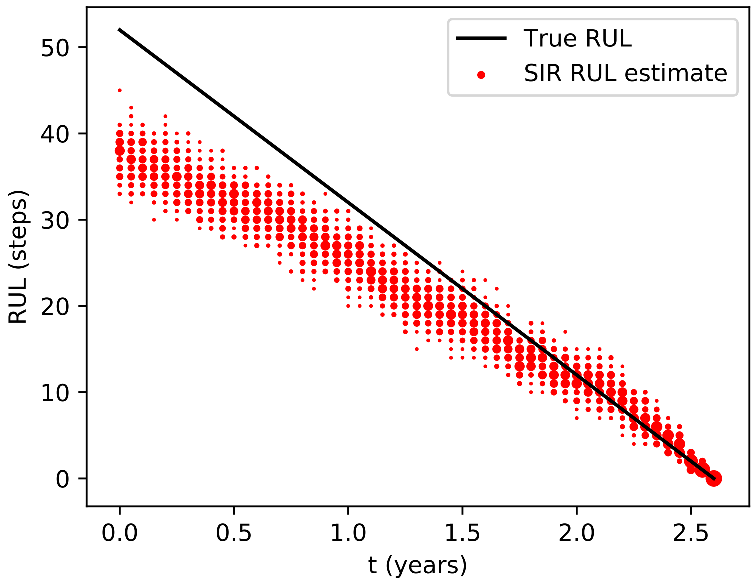

Figure 5.32: RUL prediction with varying pump power - SIR filter (dot size represent probability)

5.7 Health-aware control implementation

The HAC objective function described in Equation 4.8 aims to combine health extension and control objectives in the same optimization problem. As the control problem has 4 independent controlled variables and 5 independent manipulated variables, there is enough degrees of freedom to find an operating point which satisfies the control objectives with maximum equipment RUL. Heuristically, one should aim for the maximum top valve opening ($x_{v,t}$) to minimize pump effort and thus extend its lifetime.

The first set of simulations employed the identified model as the process model, i.e., there is no model-plant mismatch. Throughout all simulations, the used MPC tuning was as described in Table 5.5, except for the control horizon, which was $N_c = 2$. For higher control horizons, the optimization problem becomes ill-posed and thus of hard resolution.

Figure 5.33 presents results of a simulation of the HAC strategy. Due to the randomness of sampling $q_k^{(i)}$ at each control step, a steady state cannot be attained. As expected, the variable $x_{v,t}$ is brought to its allowed maximum, and W is minimized. Even though control objectives are not ignored, an offset is produced. As the operation reaches to a point where health and control objectives compete against each other, the obtained solution is a compromise between these objectives.

The effect of w_{HAC} is explored on simulations presented in Figure 5.34 and Figure 5.35, with higher and lower w_{HAC}, respectively. In the case with higher

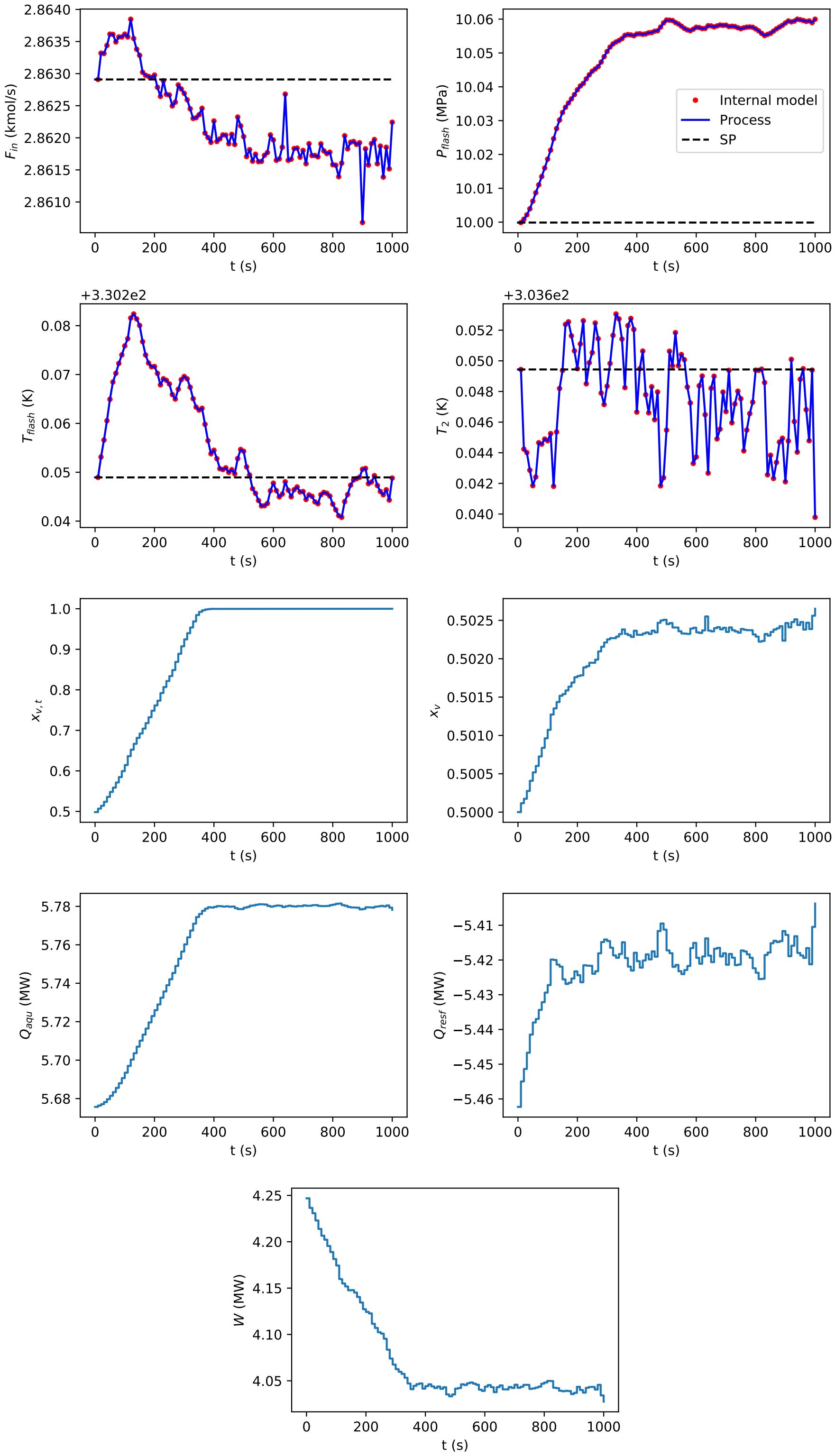

Figure 5.33: HAC simulation - $w_{HAC} = 10,\ N_{part} = 3$

w_{HAC}, further reduction on W and setpoint deviation is observed, the opposite happening in the case with lower w_{HAC}. Also, the maximization of $x_{v,t}$ occurs faster with high values of w_{HAC}, whereas the simulation with lower w_{HAC} value shows a slow, almost linear evolution of $x_{v,t}$.

Regarding the number of particles used in $\overline{RUL}$ estimation, Figure 5.36 presents results of a simulation with more particles. Comparing it to Figure 5.33, a modest reduction in the oscillation amplitude is observed. This means that the number of particles needs to be raised some magnitude orders to attain a representative mean value that stays constant with the resetting of quantile samples.

As the elevation of the number of particles showed itself not effective in stopping oscillatory behavior, a different strategy was adopted. Instead of resetting the values of $q_k^{(i)}$ at each control step, these values were kept constant throughout the closed-loop simulation. The results for this simulation are shown in Figure 5.37, in which the attainment of a steady-state value for the manipulated variables is evidenced.

Once again it was not possible to eliminate the offset, for the health extension objective becomes conflicting with the reference tracking objective. This suggests that a different approach is necessary to attain both objectives.

Regarding model-plant mismatch, Figure 5.38 displays the result of a simulation in the same conditions as Figure 5.37, but using the true process model as the controlled system. In this case, the maximization of $x_{v,t}$ is also observed, but a different value of W is attained, which is to be expected. In contrast to the perfect model case, oscillations in both manipulated and controlled variables are observed. As such, controller reconfiguration is necessary in order to account for the model-plant mismatch.

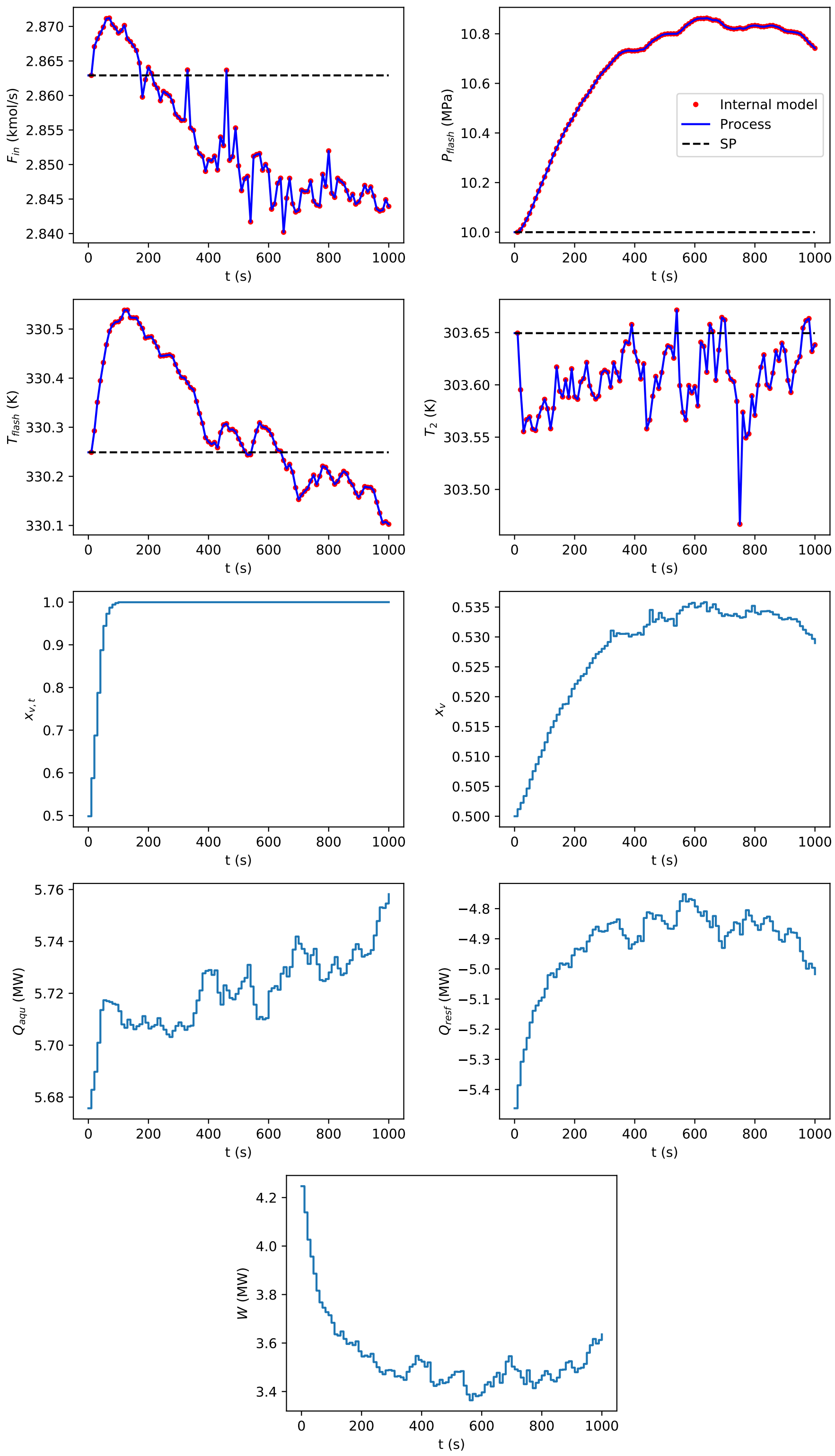

Figure 5.34: HAC simulation - $w_{HAC} = 100$, $N_{part} = 3$

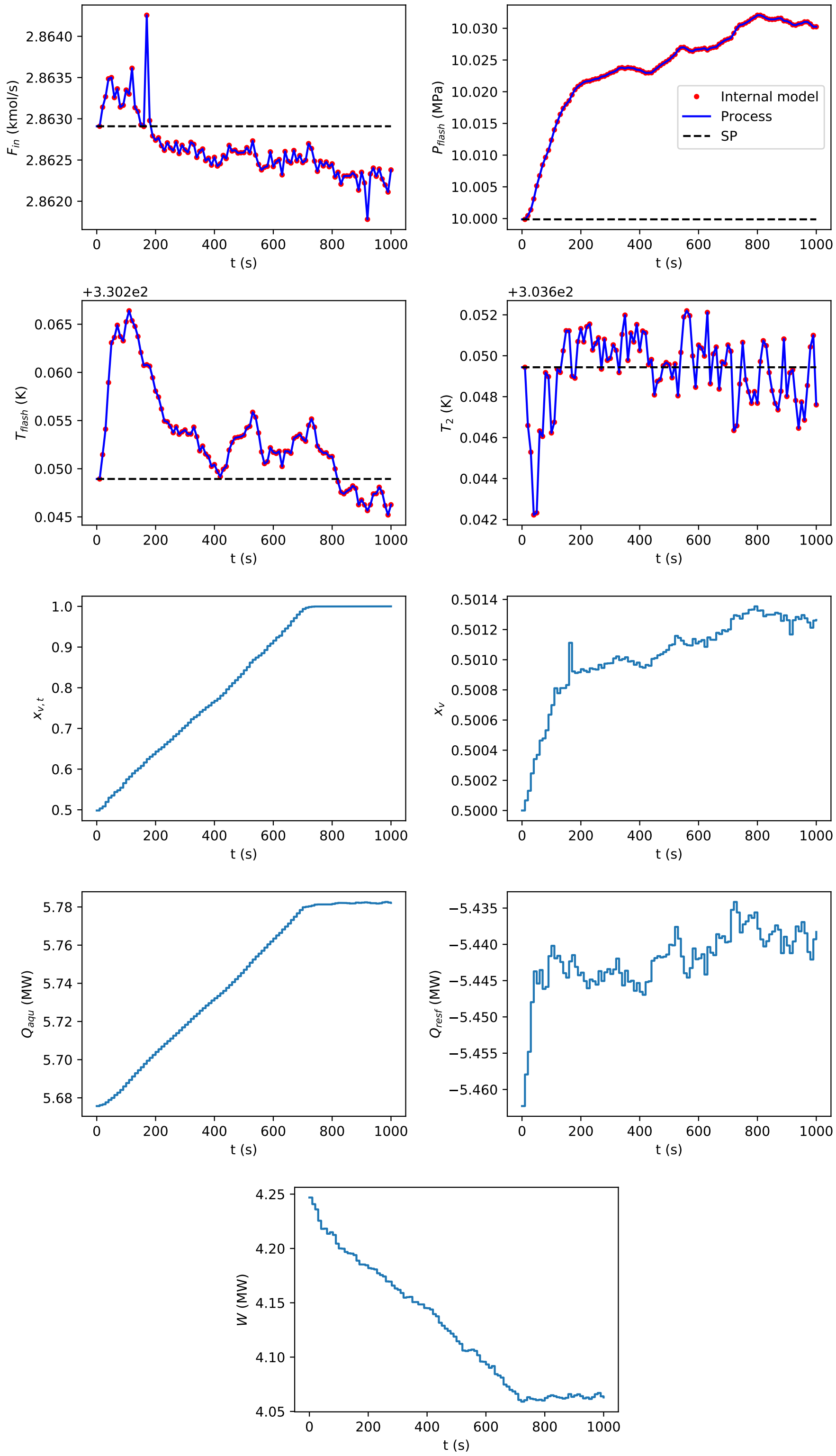

Figure 5.35: HAC simulation - $w_{HAC} = 5$, $N_{part} = 3$

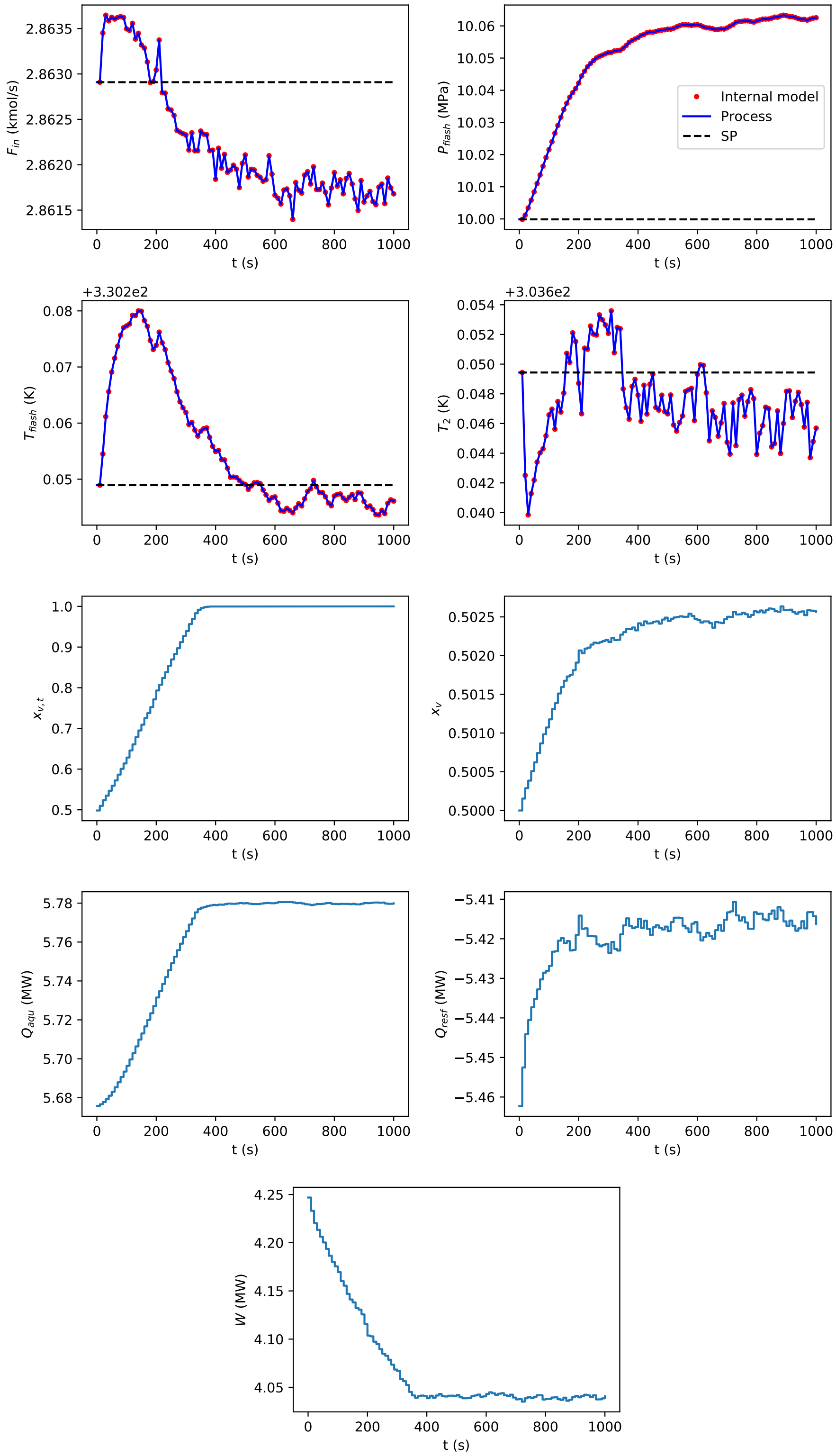

Figure 5.36: HAC simulation - $w_{HAC} = 10,\ N_{part} = 20$

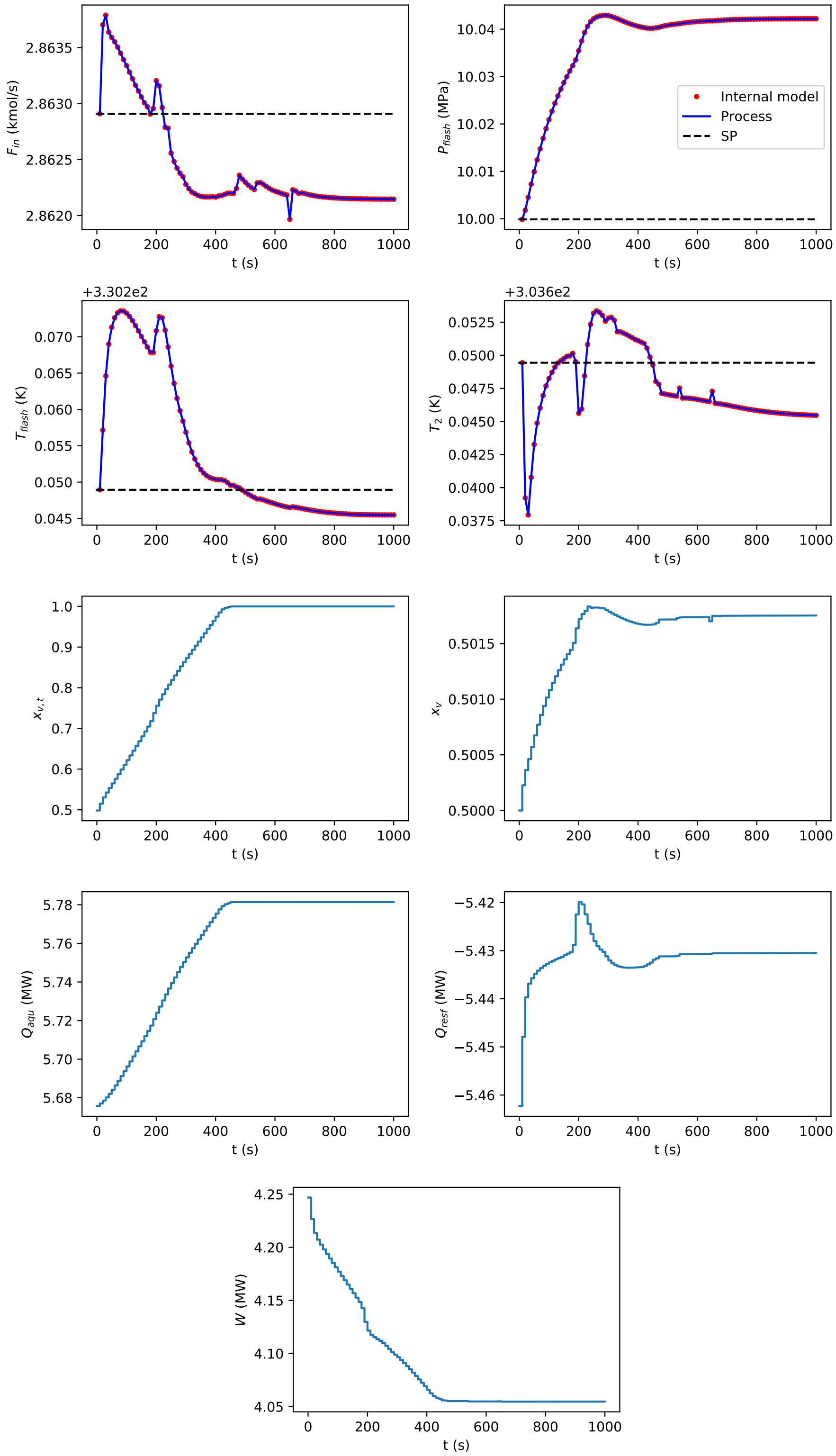

Figure 5.37: HAC simulation - $w_{HAC} = 10$, $N_{part} = 3$, fixed quantiles

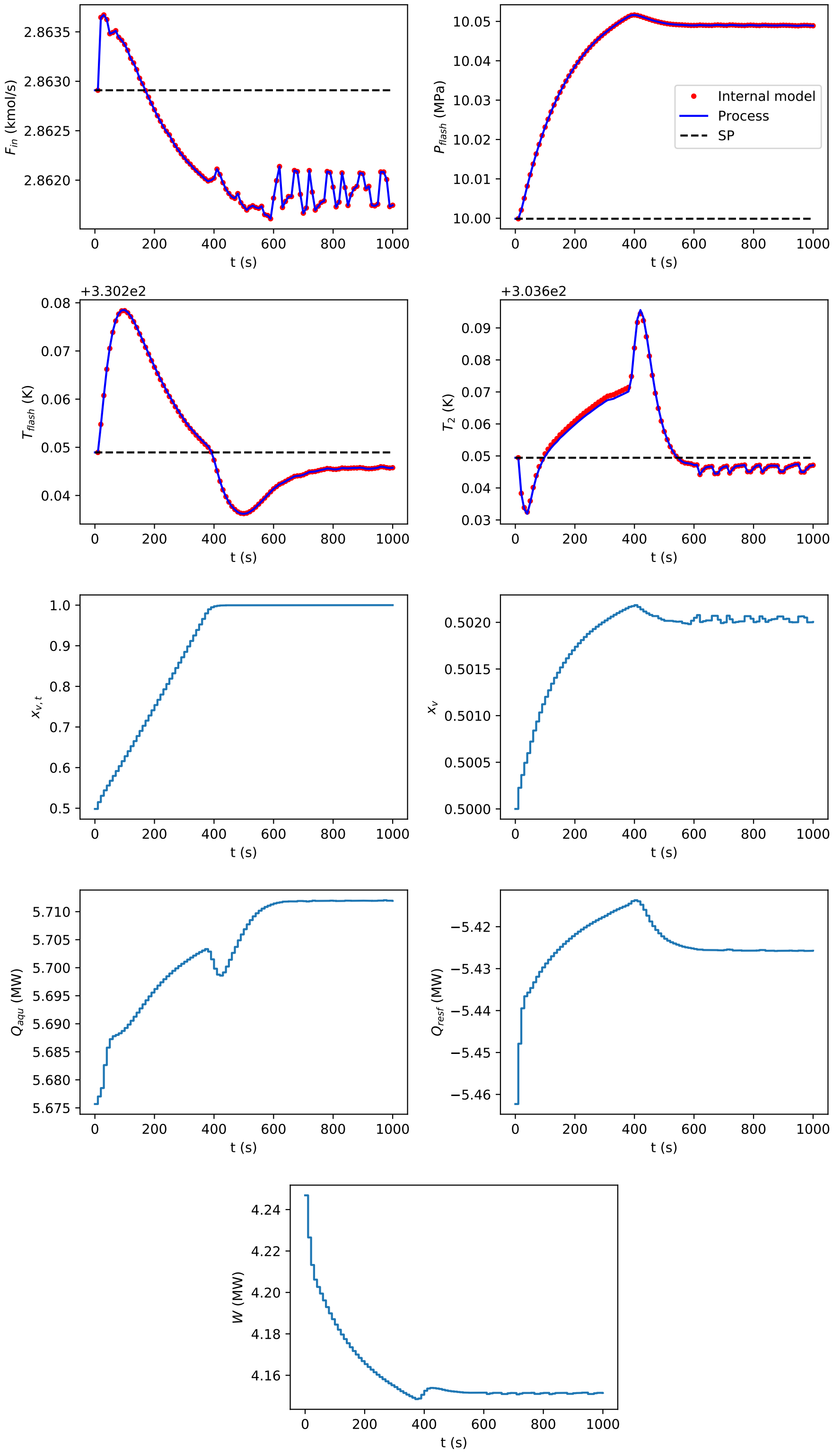

Figure 5.38: HAC simulation with true model - $w_{HAC} = 10$, $N_{part} = 3$, fixed quantiles

Chapter 6

Conclusion

In this work, motivated by the recently developed process of subsea CO_2 separation, process modeling, control and equipment prognostics were assessed.

The proposed modifications in the process model brought some conceptual corrections to the work of SOUZA (2018). The valve model, although simplified, has reasonable behavior with pressure difference. The change in the heat exchanger model resulted in a model which allows for more flexibility in simulation, as the logarithmic mean term is avoided.

The control strategies chosen in this work were the PI controller, with the main role of stabilizing plant dynamic behavior, in parallel with a MPC controller, with a linear internal model identified from the PI-stabilized plant and a KF used to correct state estimates and model bias. This strategy showed success, highlighting the importance of filter-based model correction in cases of model-plant mismatch.

A pump wear model was proposed, which reunited most desired characteristics in an equipment wear model (monotonicity, time independence, dependence with important manipulated variables). Also, dynamic simulations displayed similarity with usual degradation processes. Although this model is merely a conceptual sketch of real systems, and it has not been confronted with experimental data, the study of this kind of model is encouraged, given the results it can provide. Using this stochastic process mathematical formulation, particle filters were successfully used to estimate states and predict remaining useful lifetime.

The main efforts of this work were then consolidated in the deployment of a HAC tool to solve the proposed case study. Even though consistent results were obtained, some limitations of the proposed method were encountered, most of them concerning the contraposition of health and control objectives.

As any optimization problem with multiple conflicting objectives, the most accurate way to express it mathematically is as a multiobjective optimization problem. The objective function formulated in Equation 4.8 can be seen as simply the multiobjective problem solved by the weighted sum approach. Therefore, the treatment of this more generic problem using other tools, such as the goal attainment method, can result in more reasonable control policies.

Even though this problem was solved in an one-layer control framework, the complexity of the methods necessary to describe the degradation phenomenon suggests that this issue should be addressed in a higher control layer. With this, a higher number of particles could be used, resulting in not only a more reliable $\overline{RUL}$ estimation, but an estimate of the RUL p.d.f. itself. This would enable the analyst to work with constraints related to a confidence level to the degradation process, resulting in more robust decision-making.

Bibliography

ANDERSSON, C., FÜHRER, C., ÅKESSON, J., 2015, "Assimulo: A unified framework for ODE solvers", *Mathematics and Computers in Simulation*, v. 116, n. 0, pp. 26 – 43. ISSN: 0378-4754.

BADGWELL, T. A., QIN, S. J., 2015, "Model-predictive control in practice", *Encyclopedia of Systems and Control*, pp. 756–760.

BANJEVIC, D., 2009, "Remaining useful life in theory and practice", *Metrika*, v. 69, n. 2-3 (mar), pp. 337–349. ISSN: 0026-1335. Availability: <`http://link.springer.com/10.1007/s00184-008-0220-5`>.

BARUAH, P., CHINNAM, R. B., 2005, "HMMs for diagnostics and prognostics in machining processes", *International Journal of Production Research*, v. 43, n. 6, pp. 1275–1293.

BORDONS, C., CAMACHO, E., 2007, *Model predictive control.* Springer Verlag London Limited.

CHHIKARA, R., 1988, *The Inverse Gaussian Distribution: Theory, Methodology, and Applications*, v. 95. CRC Press.

CHOO, B. Y., ADAMS, S. C., WEISS, B. A., et al., 2016, "Adaptive Multi-scale Prognostics and Health Management for Smart Manufacturing Systems." *International journal of prognostics and health management*, v. 7, pp. 014. ISSN: 2153-2648. Availability: <`http://www.ncbi.nlm.nih.gov/pubmed/28736651http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5520667`>.

CHOO, B. Y., ADAMS, S., BELING, P., 2017, "Health-aware hierarchical control for smart manufacturing using reinforcement learning". In: *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*, pp. 40–47. IEEE, jun. ISBN: 978-1-5090-5710-8. Availability: <`http://ieeexplore.ieee.org/document/7998303/`>.

DE MEDEIROS, J. L., DE OLIVEIRA ARINELLI, L., TEIXEIRA, A. M., et al., 2018, *Offshore Processing of CO2-Rich Natural Gas with Supersonic Separator: Multiphase Sound Speed, CO2 Freeze-Out and HYSYS Implementation*. Springer.

DONG, M., HE, D., 2007, "A segmental hidden semi-Markov model (HSMM)-based diagnostics and prognostics framework and methodology", *Mechanical systems and signal processing*, v. 21, n. 5, pp. 2248–2266.

ESCOBET, T., PUIG, V., NEJJARI, F., 2012, "Health Aware Control and model-based Prognosis". In: *2012 20th Mediterranean Conference on Control & Automation (MED)*, pp. 691–696. IEEE, jul. ISBN: 978-1-4673-2531-8. Availability: <http://ieeexplore.ieee.org/document/6265718/>.

GARCIA, C. E., MORSHEDI, A., 1986, "Quadratic programming solution of dynamic matrix control (QDMC)", *Chemical Engineering Communications*, v. 46, n. 1-3, pp. 73–87.

GROSSO, J. M., OCAMPO-MARTINEZ, C., PUIG, V., 2016, "Reliability–based economic model predictive control for generalised flow–based networks including actuators' health–aware capabilities", *International Journal of Applied Mathematics and Computer Science*, v. 26, n. 3 (sep), pp. 641–654. ISSN: 2083-8492. Availability: <http://content.sciendo.com/view/journals/amcs/26/3/article-p641.xml>.

HENG, A., ZHANG, S., TAN, A. C., et al., 2009, "Rotating machinery prognostics: State of the art, challenges and opportunities", *Mechanical systems and signal processing*, v. 23, n. 3, pp. 724–739.

INCROPERA, F. P., 2006, *Fundamentals of Heat and Mass Transfer*. USA, John Wiley &#38; Sons, Inc. ISBN: 0470088400.

JARDINE, A. K., LIN, D., BANJEVIC, D., 2006, "A review on machinery diagnostics and prognostics implementing condition-based maintenance", *Mechanical systems and signal processing*, v. 20, n. 7, pp. 1483–1510.

JONES, E., OLIPHANT, T., PETERSON, P., et al., 2001. "SciPy: Open source scientific tools for Python". Availability: <http://www.scipy.org/>.

JOUIN, M., GOURIVEAU, R., HISSEL, D., et al., 2016, "Particle filter-based prognostics: Review, discussion and perspectives", *Mechanical Systems and Signal Processing*, v. 72, pp. 2–31.

KAIPIO, J., SOMERSALO, E., 2005, *Applied Mathematical Sciences*, v. 160. Springer-Verlag, New York.

KOTHAMASU, R., HUANG, S. H., VERDUIN, W. H., 2006, "System health monitoring and prognostics — a review of current paradigms and practices", *The International Journal of Advanced Manufacturing Technology*, v. 28, n. 9-10 (jul), pp. 1012–1024. ISSN: 0268-3768. Availability: <http://link.springer.com/10.1007/s00170-004-2131-6>.

KREYSZIG, E., KREYSZIG, H., NORMINTON, E. J., 2011, *Advanced Engineering Mathematics*. Tenth ed. Hoboken, NJ, Wiley. ISBN: 0470458364.

KUCHPIL, C., DUARTE, D. G., ORLOWSKI, R., et al., 2013, "Subsea Processing and Boosting in Brazil: Status and Future Vision". In: *OTC Brasil*. Offshore Technology Conference.

LEI, Y., LI, N., GUO, L., et al., 2018, "Machinery health prognostics: A systematic review from data acquisition to RUL prediction", *Mechanical Systems and Signal Processing*, v. 104 (may), pp. 799–834. ISSN: 08883270. Availability: <https://linkinghub.elsevier.com/retrieve/pii/S0888327017305988>.

LUCIA, S., FINKLER, T., ENGELL, S., 2013, "Multi-stage nonlinear model predictive control applied to a semi-batch polymerization reactor under uncertainty", *Journal of Process Control*, v. 23, n. 9 (oct), pp. 1306–1319. ISSN: 09591524. Availability: <https://linkinghub.elsevier.com/retrieve/pii/S0959152413001686>.

OGATA, K., 2010. "Modern Control Engineering. United State, America". .

PARIS, P., ERDOGAN, F., 1963, "A critical analysis of crack propagation laws", *Journal of basic engineering*, v. 85, n. 4, pp. 528–533.

PASSARELLI, F. M., 2017. "Processo e sistema para remoção de dióxido de carbono em fase densa de correntes de petróleo e gás natural". Availability: <https://www.lens.org/lens/patent/181-301-137-556-927>.

PEREIRA, E. B., GALVAO, R. K. H., YONEYAMA, T., 2010, "Model Predictive Control using Prognosis and Health Monitoring of actuators". In: *2010 IEEE International Symposium on Industrial Electronics*, pp. 237–243. IEEE, jul. ISBN: 978-1-4244-6390-9. Availability: <http://ieeexplore.ieee.org/document/5637571/>.

POUR, F. K., PUIG, V., OCAMPO-MARTINEZ, C., 2018, "Multi-layer health-aware economic predictive control of a pasteurization pilot plant", *International Journal of Applied Mathematics and Computer Science*, v. 28, n. 1 (mar), pp. 97–110. ISSN: 2083-8492. Availability: <`http://content.sciendo.com/view/journals/amcs/28/1/article-p97.xml`>.

QIN, S. J., BADGWELL, T. A., 2003, "A survey of industrial model predictive control technology", *Control engineering practice*, v. 11, n. 7, pp. 733–764.

QIN, S. J., BADGWELL, T. A., 1997, "An overview of industrial model predictive control technology". In: *AIChE Symposium Series*, v. 93, pp. 232–256. New York, NY: American Institute of Chemical Engineers, 1971-c2002.

RAUSAND, M., HØYLAND, A., 2004, *System reliability theory: models, statistical methods, and applications*, v. 396. John Wiley & Sons.

ROEMER, M. J., BYINGTON, C. S., KACPRZYNSKI, G. J., et al., 2011, "Prognostics", *System Health Management: With Aerospace Applications*, pp. 281–295.

SALAZAR, J. C., WEBER, P., NEJJARI, F., et al., 2016, "MPC Framework for System Reliability Optimization". pp. 161–177. Availability: <`http://link.springer.com/10.1007/978-3-319-23180-8{_}12`>.

SANCHEZ, H., ESCOBET, T., PUIG, V., et al., 2015, "Health-aware Model Predictive Control of Wind Turbines using Fatigue Prognosis", *IFAC-PapersOnLine*, v. 48, n. 21, pp. 1363–1368. ISSN: 24058963. Availability: <`https://linkinghub.elsevier.com/retrieve/pii/S2405896315018443`>.

SI, X.-S., WANG, W., HU, C.-H., et al., 2011, "Remaining useful life estimation – A review on the statistical data driven approaches", *European Journal of Operational Research*, v. 213, n. 1 (aug), pp. 1–14. ISSN: 03772217. Availability: <`https://linkinghub.elsevier.com/retrieve/pii/S0377221710007903`>.

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

SKOGESTAD, S., 2003, "Simple analytic rules for model reduction and PID controller tuning", *Journal of process control*, v. 13, n. 4, pp. 291–309.

SKOGESTAD, S., POSTLETHWAITE, I., 2007, *Multivariable feedback control: analysis and design*, v. 2. Wiley New York.

SOUZA, A. F. F. D., 2018, *Separação submarina óleo/CO2: concepção tecnológica, modelagem, e controle avançado*. Master Thesis, Universidade Federal do Rio de Janeiro.

SOUZA, A. F. F. D., SECCHI, A. R., DE SOUZA JR, M. B., 2019, "CO2 Subsea Separation: Concept & Control Strategies", *IFAC-PapersOnLine*, v. 52, n. 1, pp. 790–795.

SPEEKENBRINK, M., 2016, "A tutorial on particle filters", *Journal of Mathematical Psychology*, v. 73, pp. 140–152.

SUN, J., ZUO, H., WANG, W., et al., 2014, "Prognostics uncertainty reduction by fusing on-line monitoring data based on a state-space-based degradation model", *Mechanical Systems and Signal Processing*, v. 45, n. 2, pp. 396–407.

TRIERWEILER, J., SECCHI, A., 2000, "Exploring the potentiality of using multiple model approach in nonlinear model predictive control". In: *Nonlinear Model Predictive Control*, Springer, pp. 191–203.

VAN NOORTWIJK, J., 2009, "A survey of the application of gamma processes in maintenance", *Reliability Engineering & System Safety*, v. 94, n. 1, pp. 2–21.

VERHEYLEWEGHEN, A., JÄSCHKE, J., 2017, "Framework for Combined Diagnostics, Prognostics and Optimal Operation of a Subsea Gas Compression System * *This work is funded by the SUBPRO center for research based innovation, www.ntnu.edu/subpro", *IFAC-PapersOnLine*, v. 50, n. 1 (jul), pp. 15916–15921. ISSN: 24058963. Availability: <`https://linkinghub.elsevier.com/retrieve/pii/S2405896317331816`>.

VERHEYLEWEGHEN, A., GJØBY, J. M., JÄSCHKE, J., 2018, "Health-Aware Operation of a Subsea Compression System Subject to Degradation". pp. 1021–1026. Availability: <`https://linkinghub.elsevier.com/retrieve/pii/B9780444642356501790`>.

VIRKLER, D. A., HILLBERRY, B., GOEL, P., 1979, "The statistical nature of fatigue crack propagation", *Journal of Engineering Materials and Technology*, v. 101, n. 2, pp. 148–153.

WANG, X., BALAKRISHNAN, N., GUO, B., 2014, “Residual life estimation based on a generalized Wiener degradation process”, *Reliability Engineering & System Safety*, v. 124 (apr), pp. 13–23. ISSN: 09518320. Availability: <`https://linkinghub.elsevier.com/retrieve/pii/S0951832013003086`>.

Appendix A

System identification results

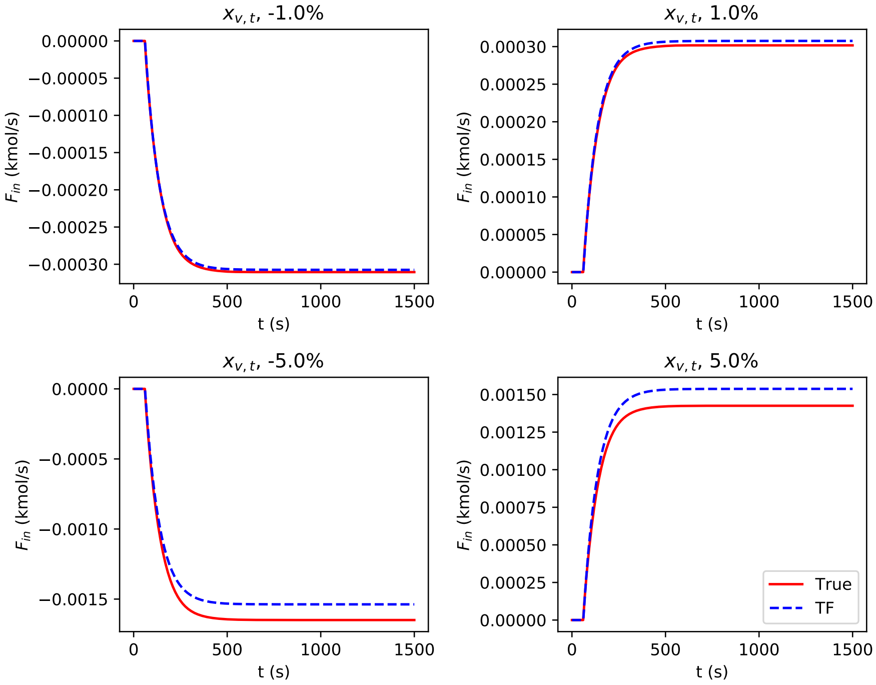

Figure A.1: Identification results for the pair F_{in} (kmol/s) x $x_{v,t}$

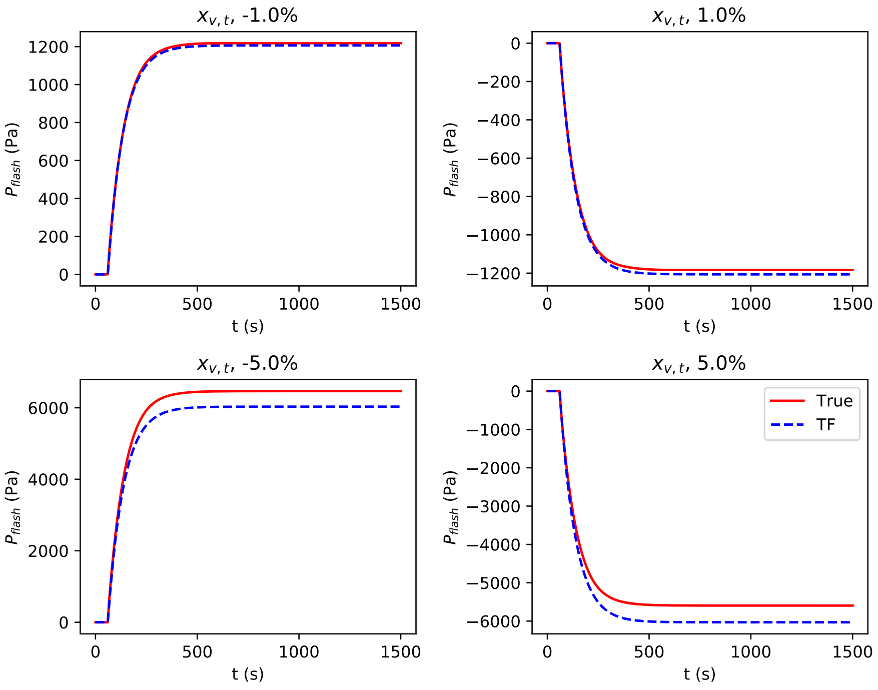

Figure A.2: Identification results for the pair P_{flash} (Pa) x $x_{v,t}$

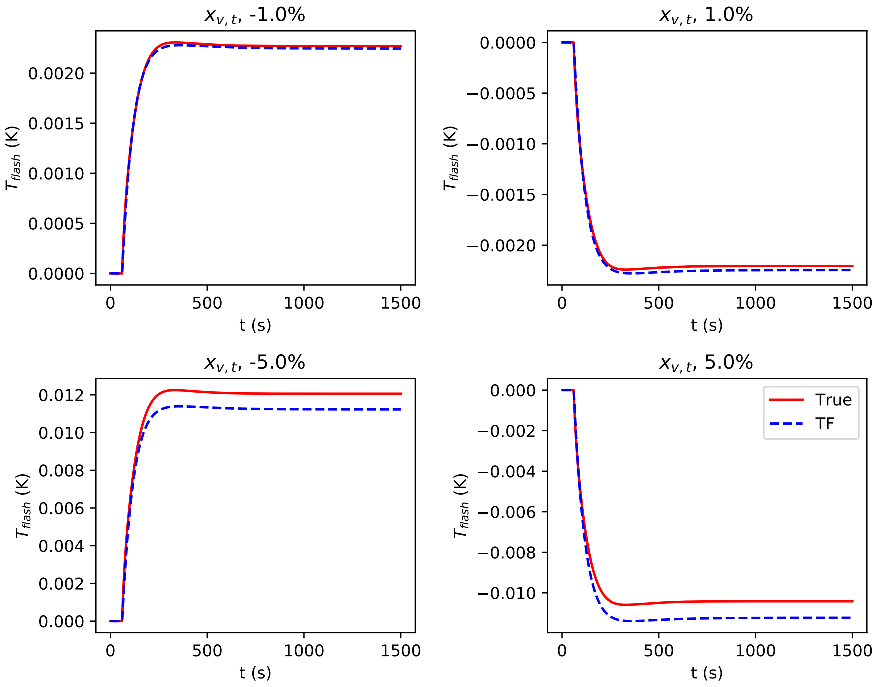

Figure A.3: Identification results for the pair T_{flash} (K) x $x_{v,t}$

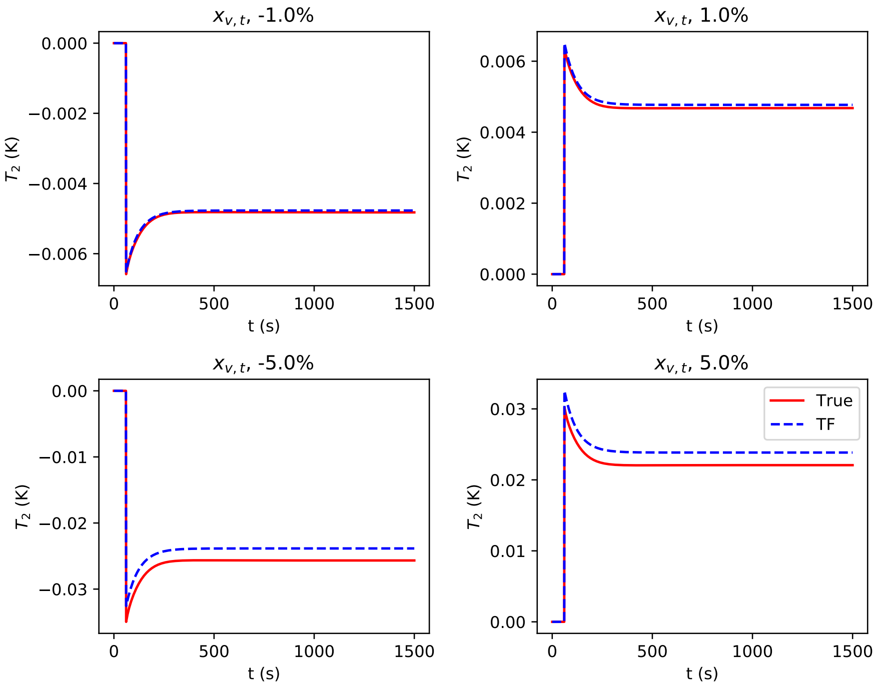

Figure A.4: Identification results for the pair T_2 (K) x $x_{v,t}$

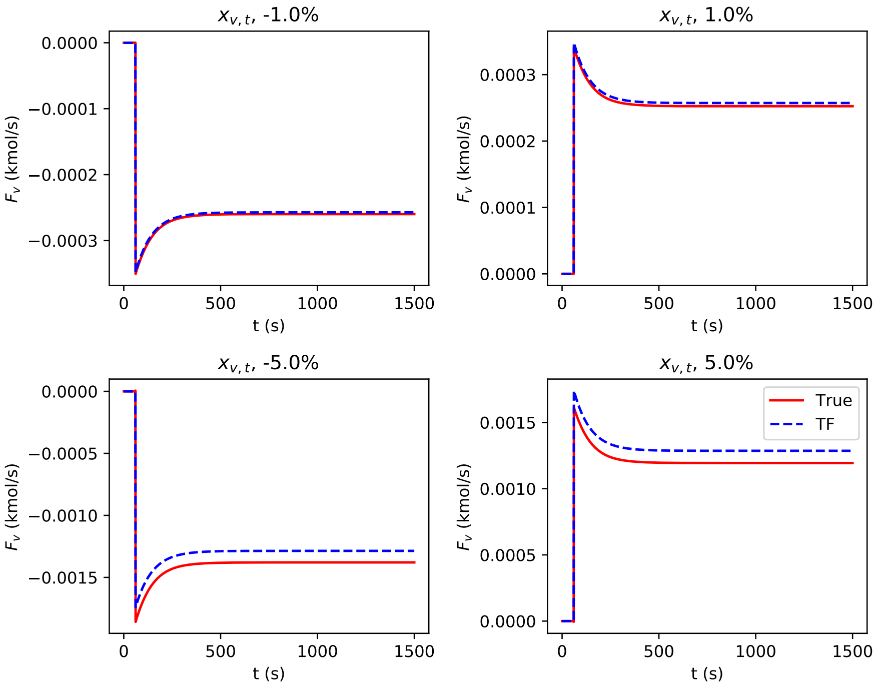

Figure A.5: Identification results for the pair F_v (kmol/s) x $x_{v,t}$

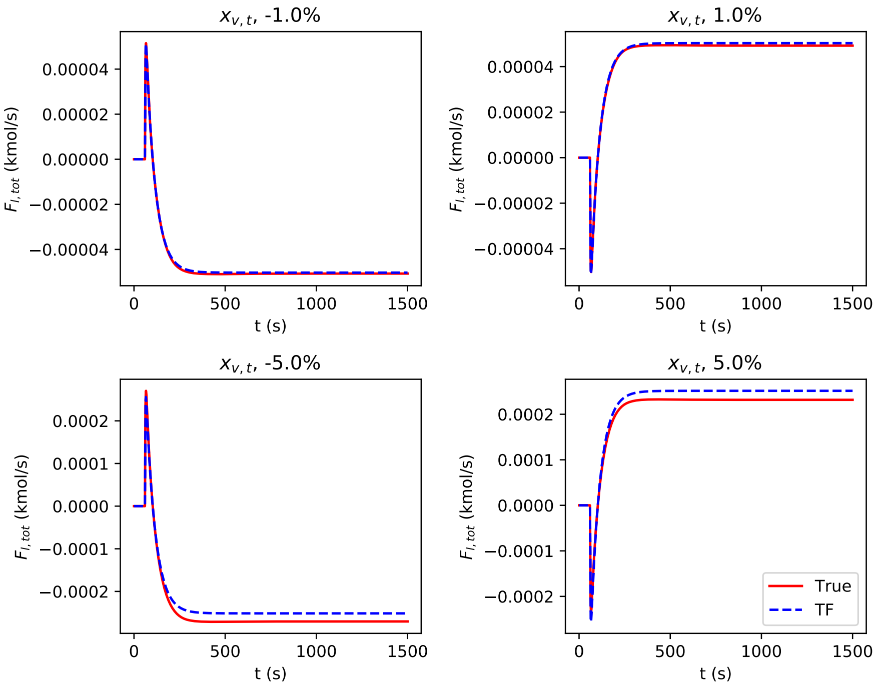

Figure A.6: Identification results for the pair $F_{l,tot}$ (kmol/s) x $x_{v,t}$

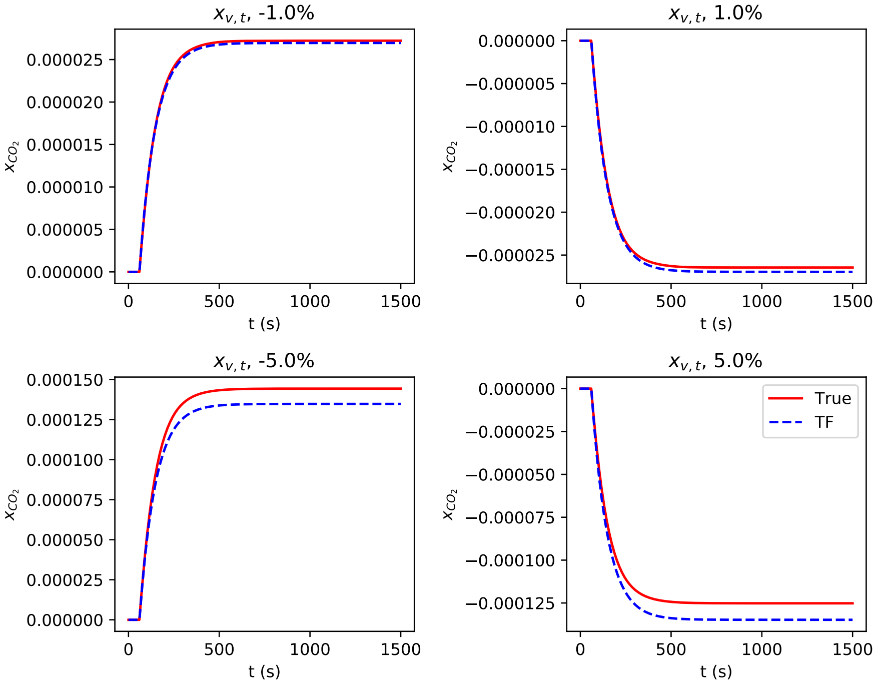

Figure A.7: Identification results for the pair x_{CO_2} x $x_{v,t}$

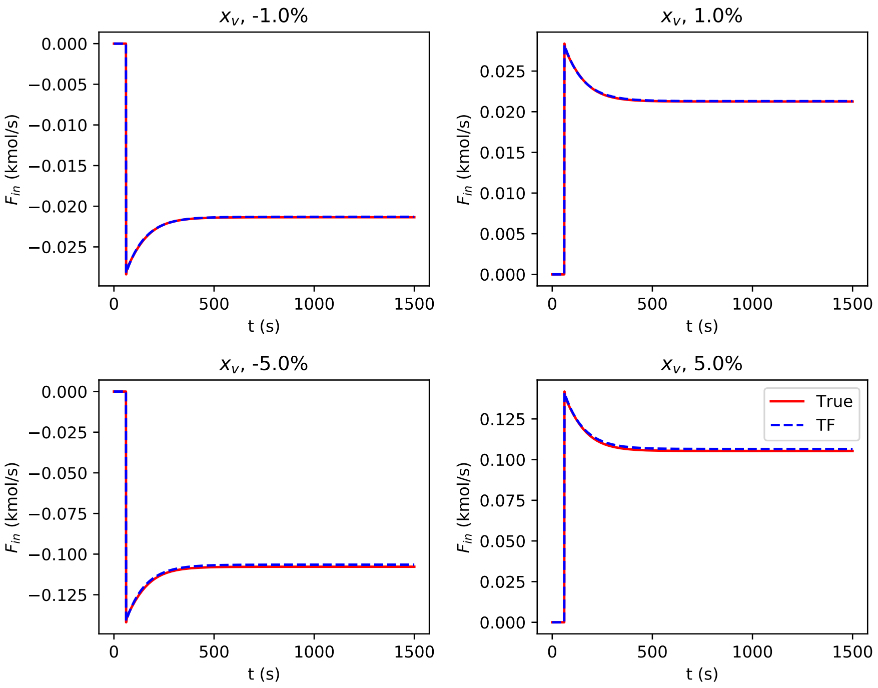

Figure A.8: Identification results for the pair F_{in} (kmol/s) x x_v

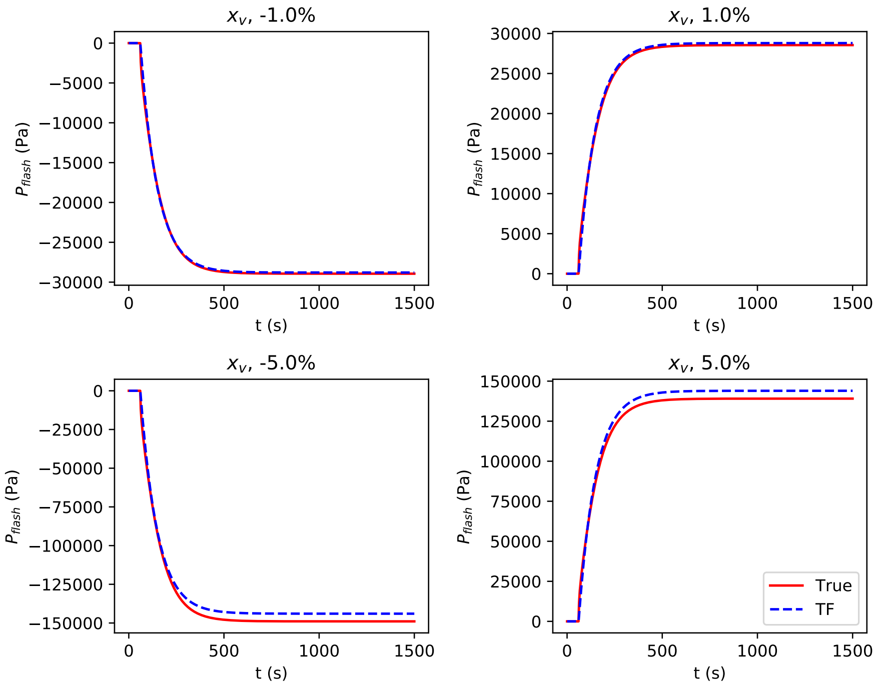

Figure A.9: Identification results for the pair P_{flash} (Pa) x x_v

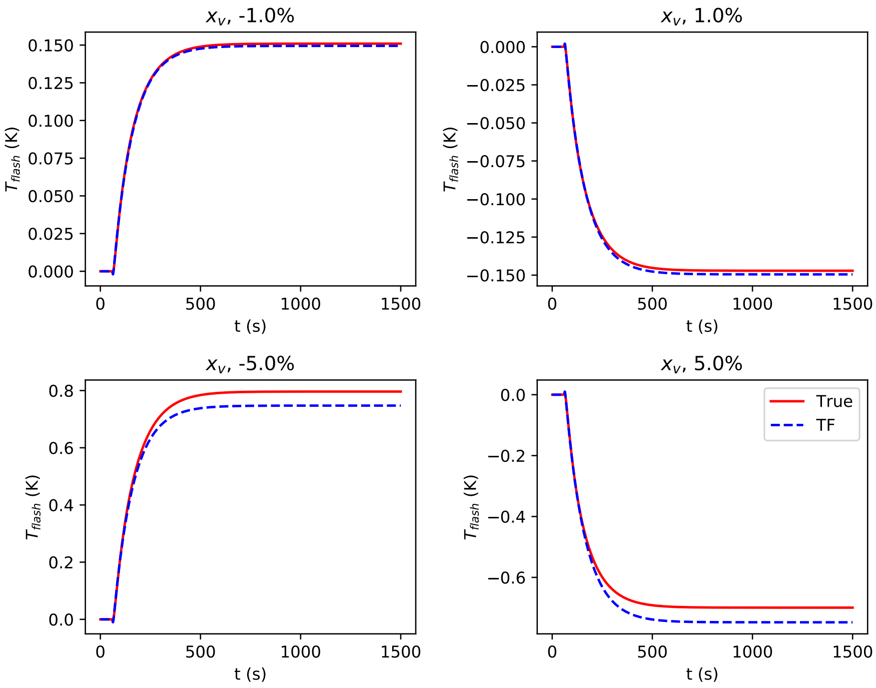

Figure A.10: Identification results for the pair T_{flash} (K) x x_v

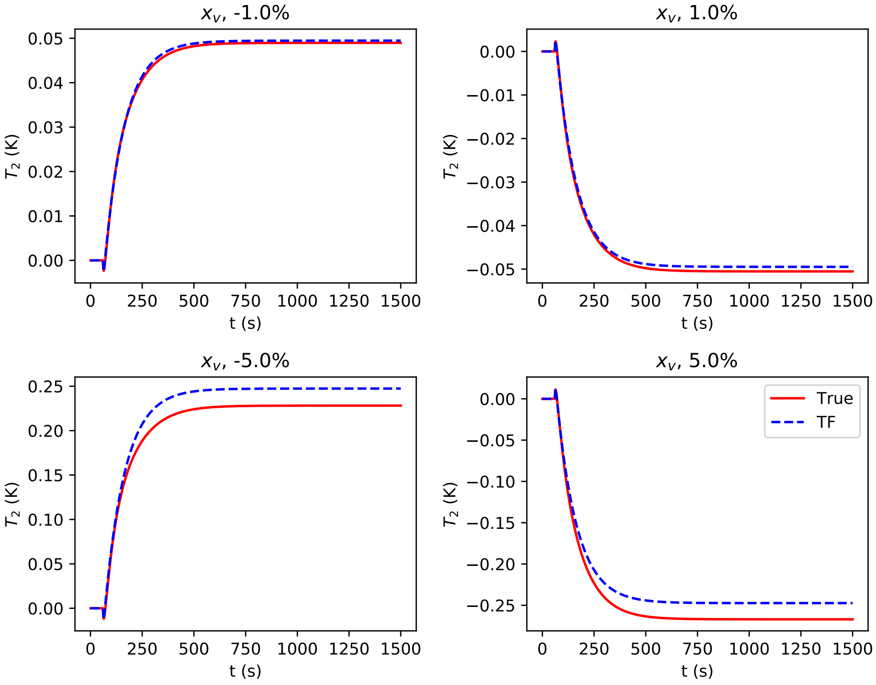

Figure A.11: Identification results for the pair T_2 (K) x x_v

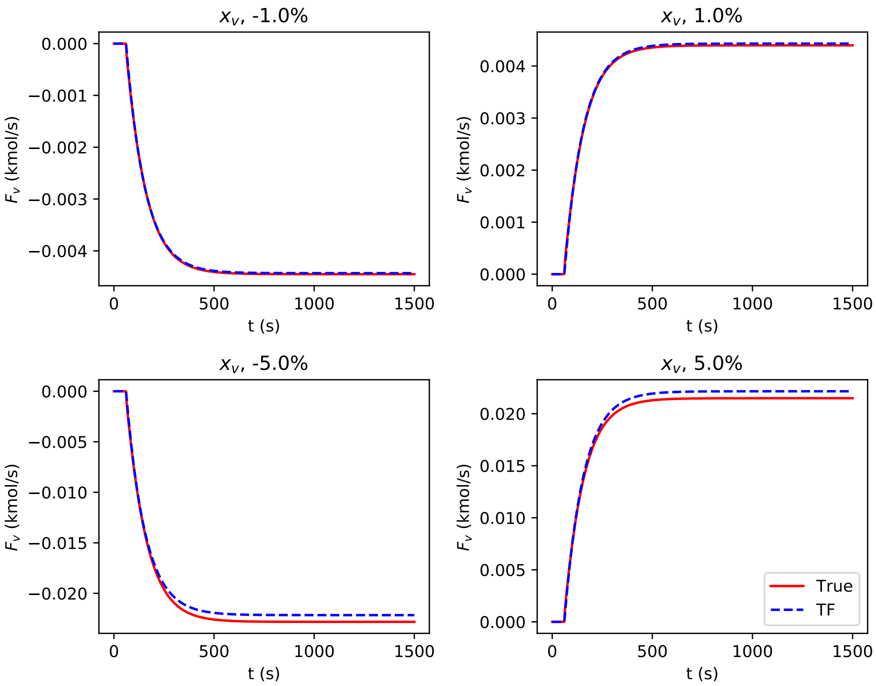

Figure A.12: Identification results for the pair F_v (kmol/s) x x_v

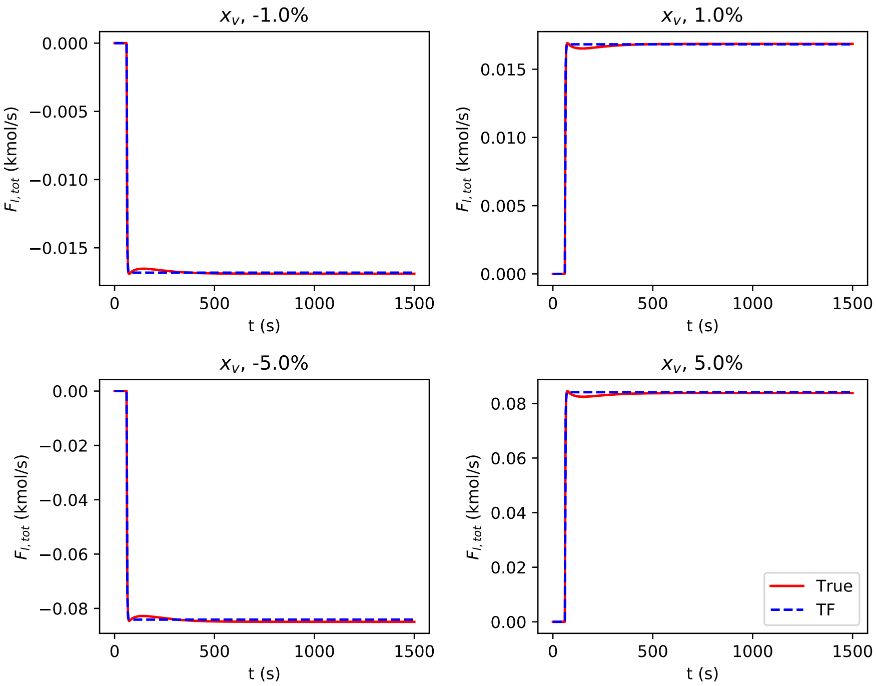

Figure A.13: Identification results for the pair $F_{l,tot}$ (kmol/s) x x_v

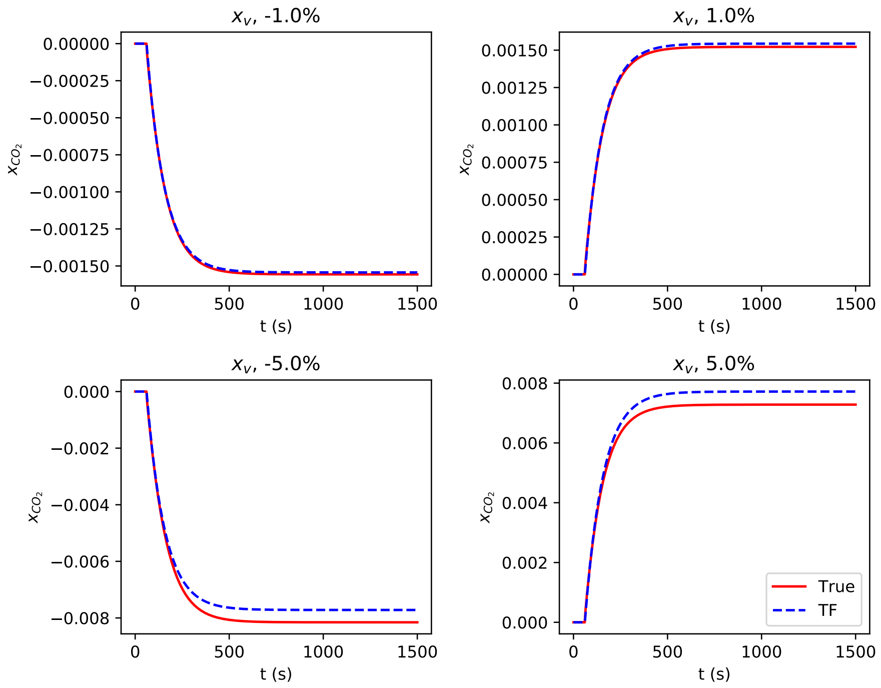

Figure A.14: Identification results for the pair x_{CO_2} x x_v

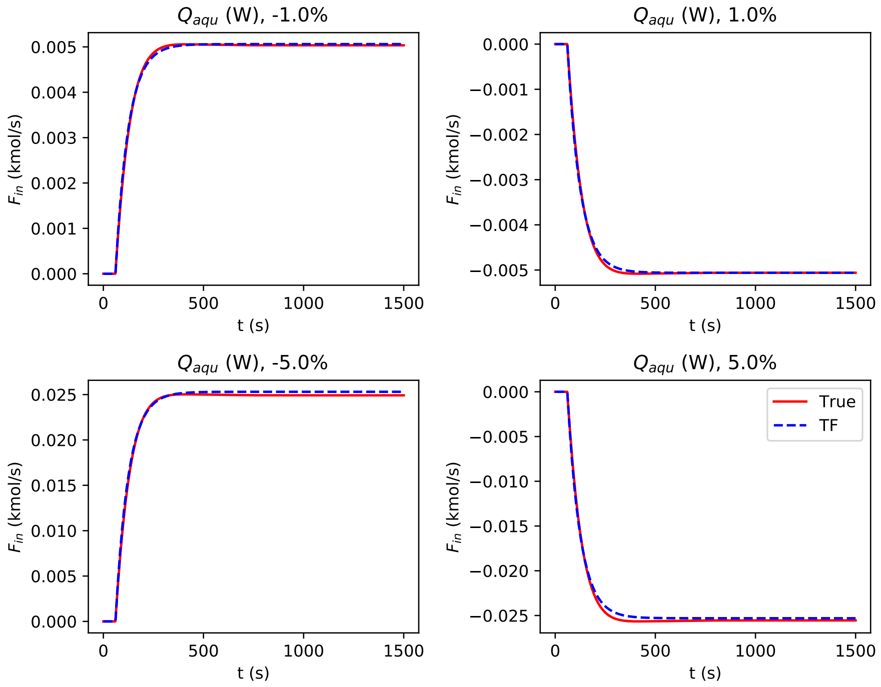

Figure A.15: Identification results for the pair F_{in} (kmol/s) x Q_{aqu} (W)

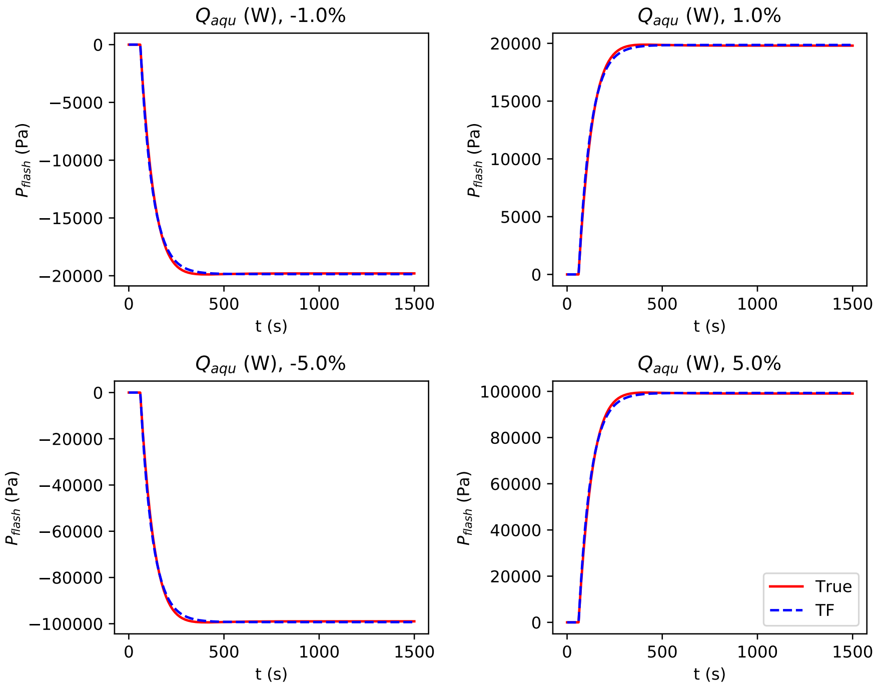

Figure A.16: Identification results for the pair P_{flash} (Pa) x Q_{aqu} (W)

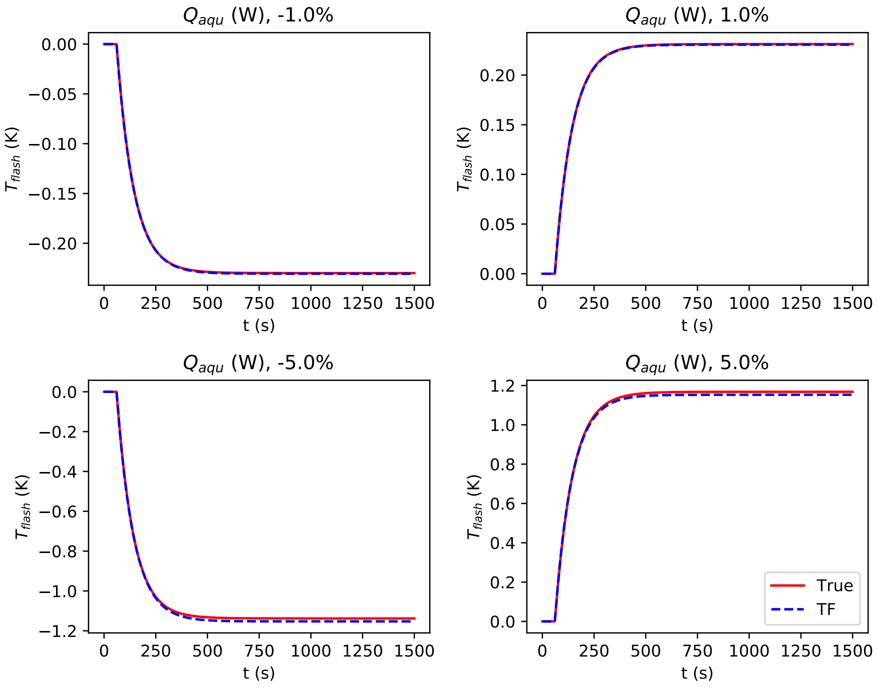

Figure A.17: Identification results for the pair T_{flash} (K) x Q_{aqu} (W)

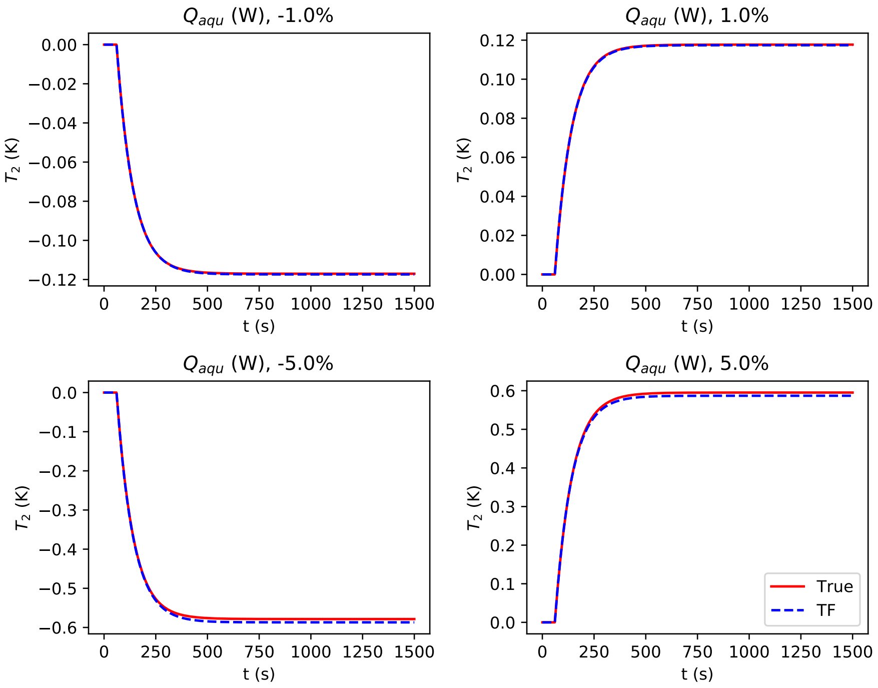

Figure A.18: Identification results for the pair T_2 (K) x Q_{aqu} (W)

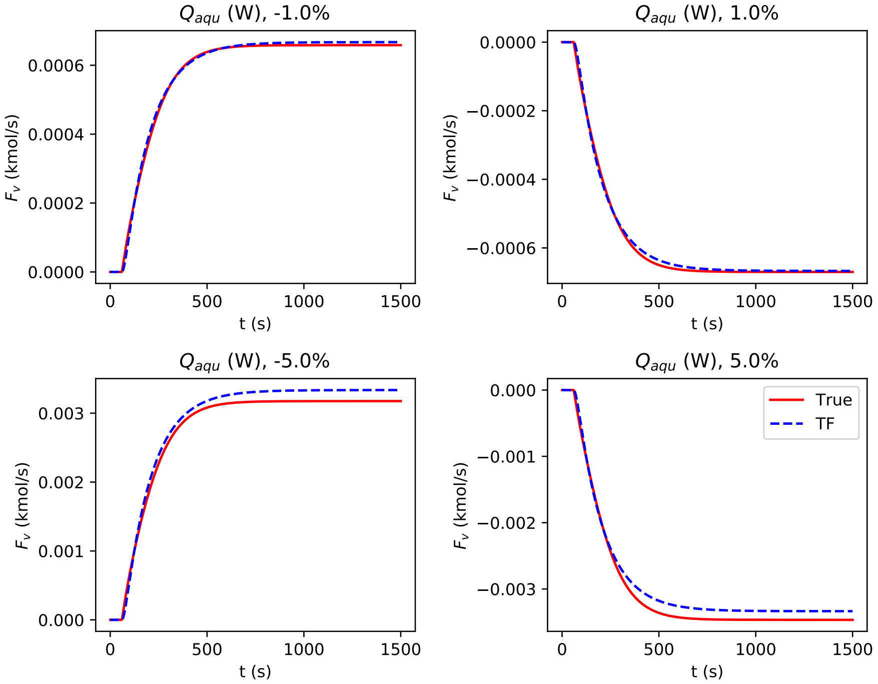

Figure A.19: Identification results for the pair F_v (kmol/s) x Q_{aqu} (W)

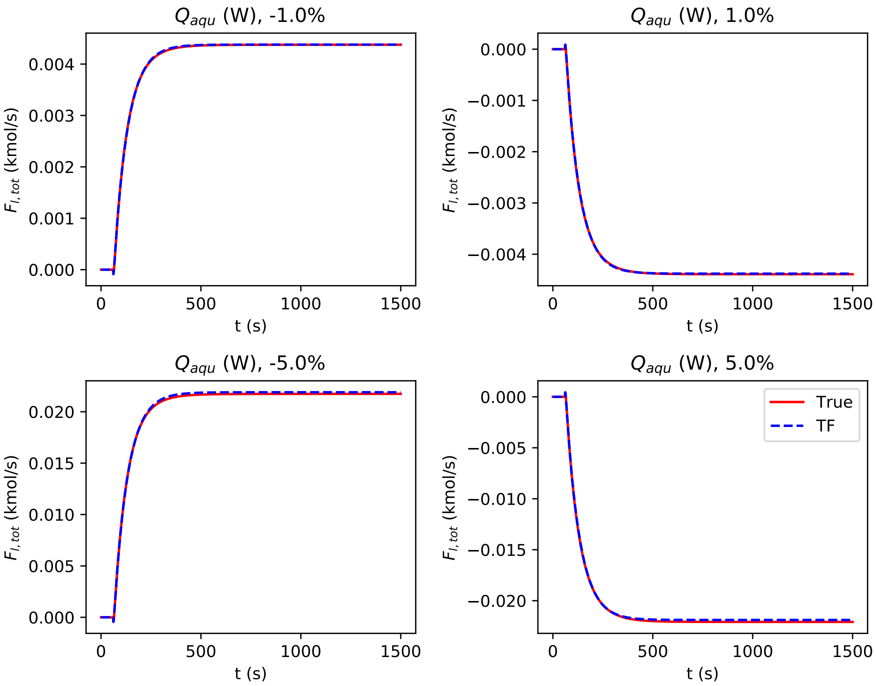

Figure A.20: Identification results for the pair $F_{l,tot}$ (kmol/s) x Q_{aqu} (W)

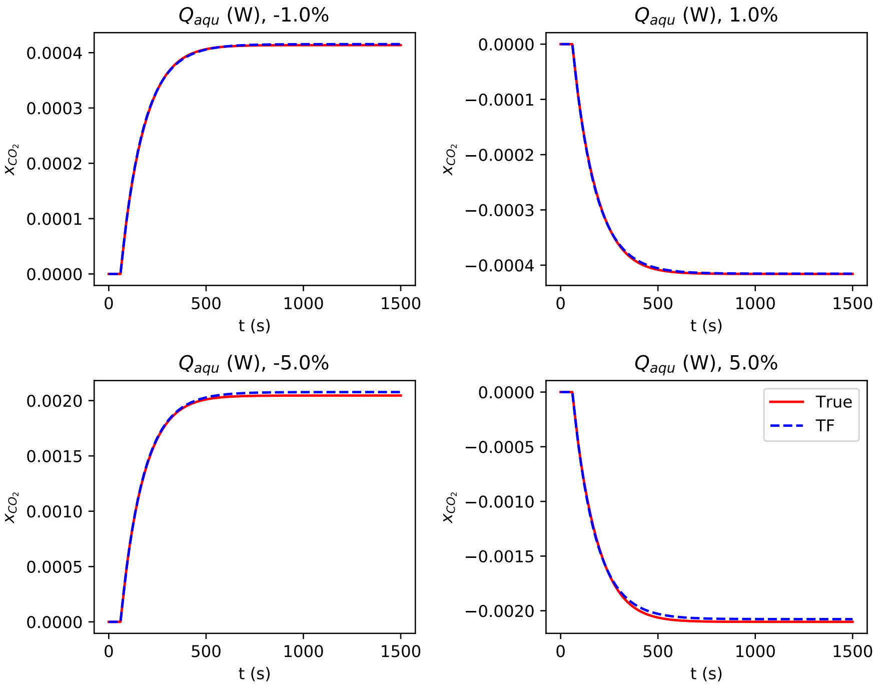

Figure A.21: Identification results for the pair x_{CO_2} x Q_{aqu} (W)

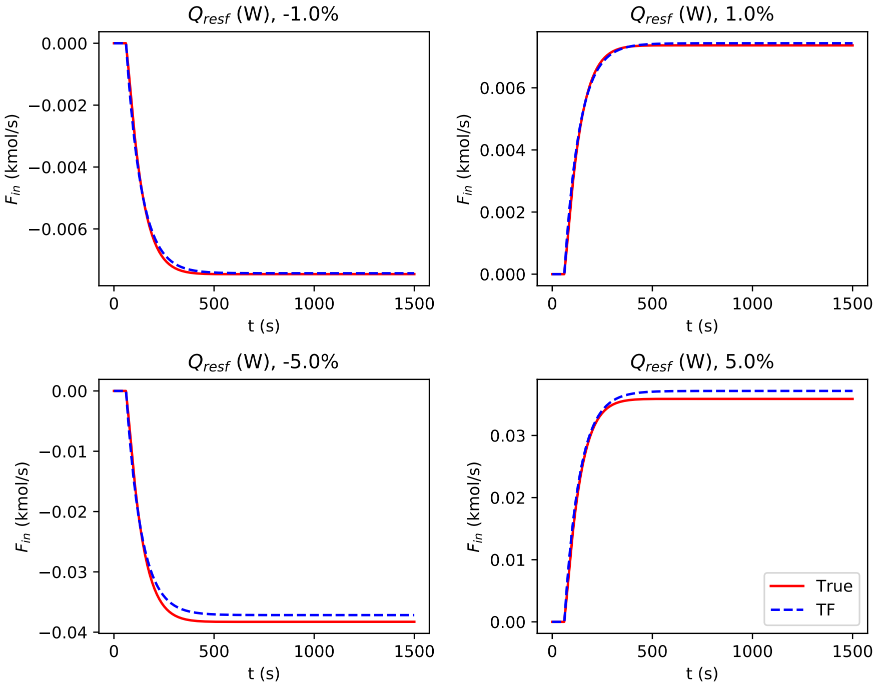

Figure A.22: Identification results for the pair F_{in} (kmol/s) x Q_{resf} (W)

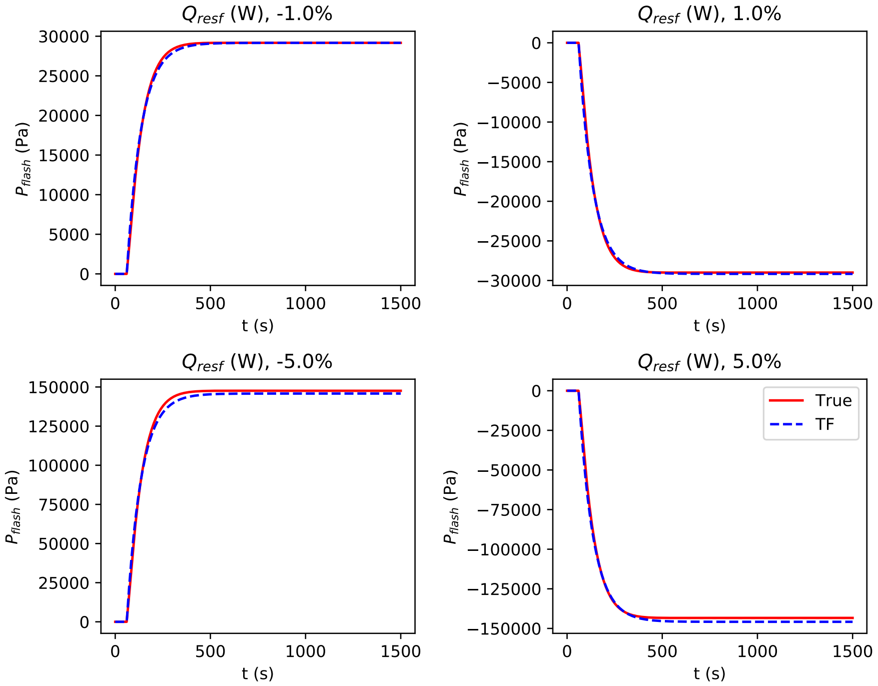

Figure A.23: Identification results for the pair P_{flash} (Pa) x Q_{resf} (W)

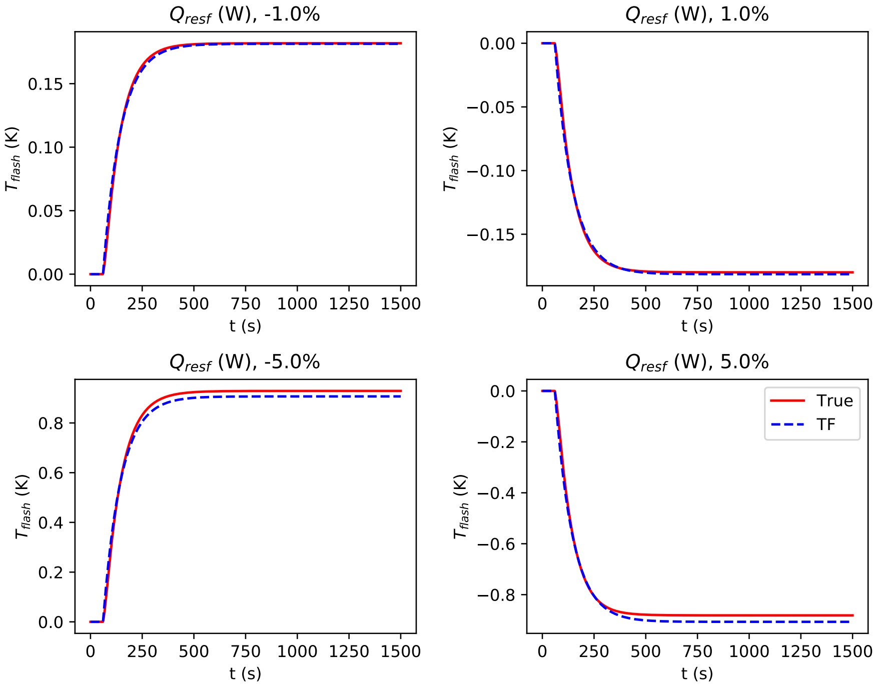

Figure A.24: Identification results for the pair T_{flash} (K) x Q_{resf} (W)

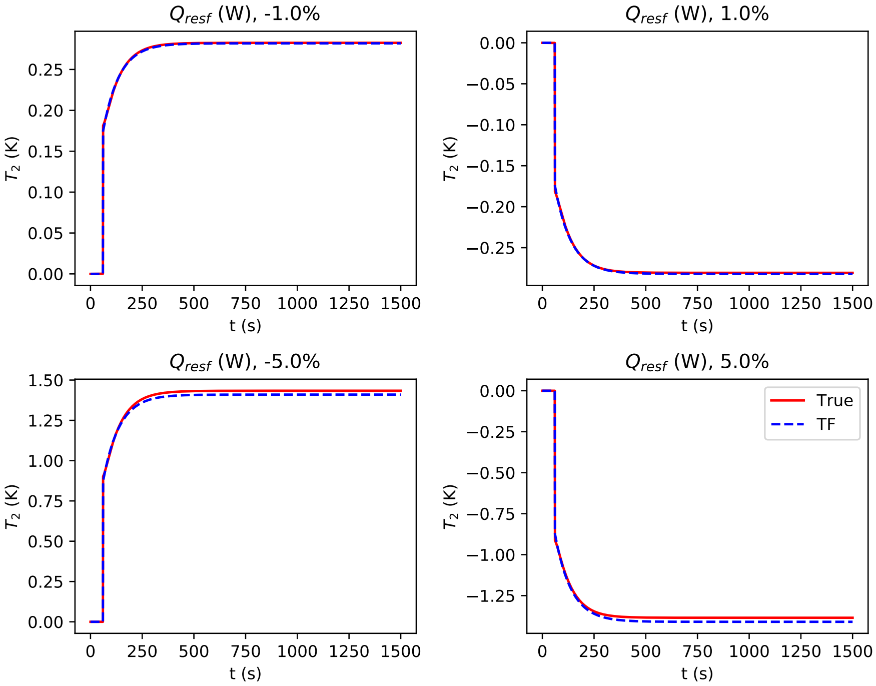

Figure A.25: Identification results for the pair T_2 (K) x Q_{resf} (W)

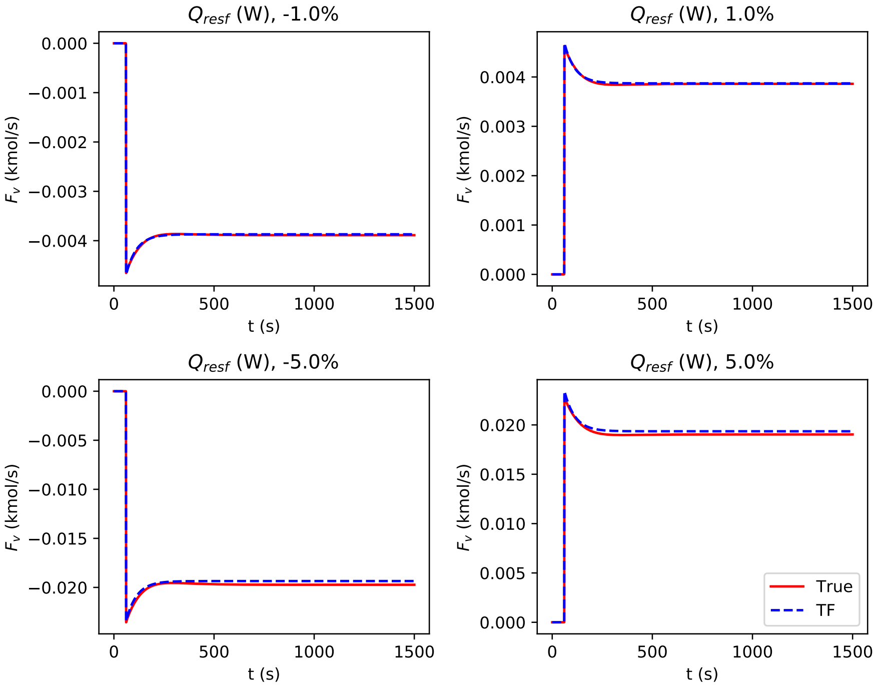

Figure A.26: Identification results for the pair F_v (kmol/s) x Q_{resf} (W)

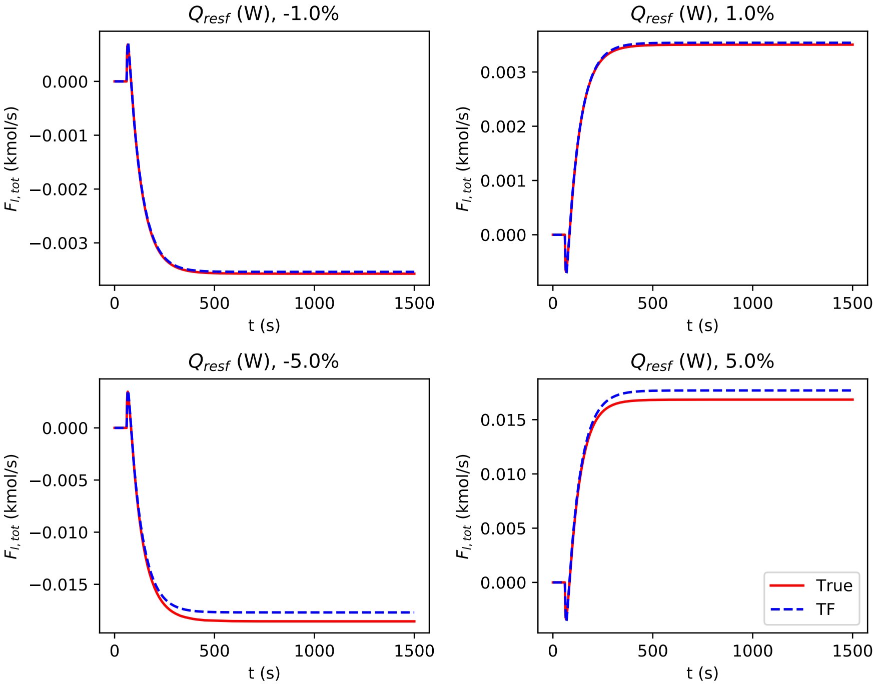

Figure A.27: Identification results for the pair $F_{l,tot}$ (kmol/s) x Q_{resf} (W)

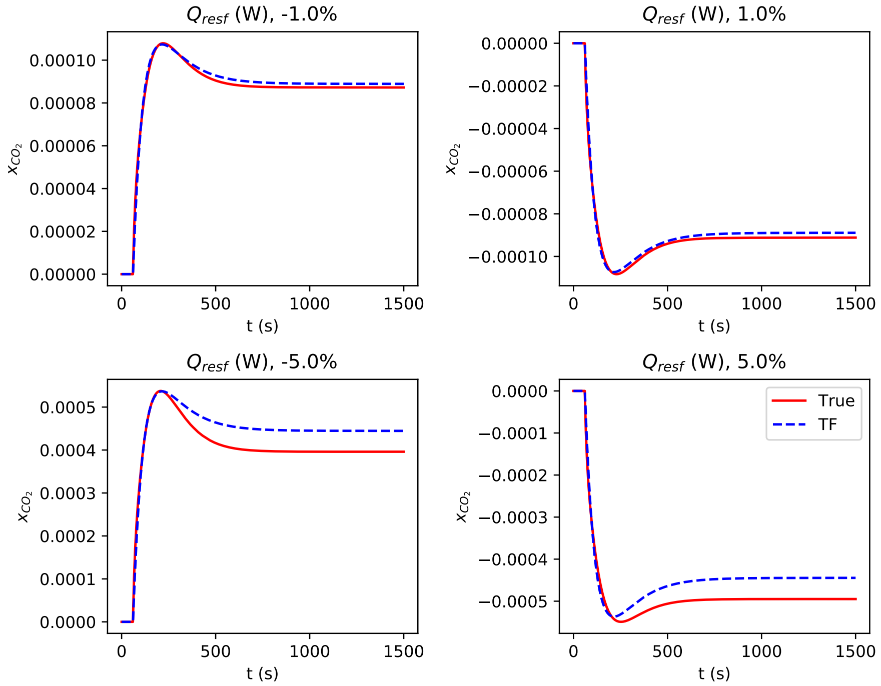

Figure A.28: Identification results for the pair x_{CO_2} x Q_{resf} (W)

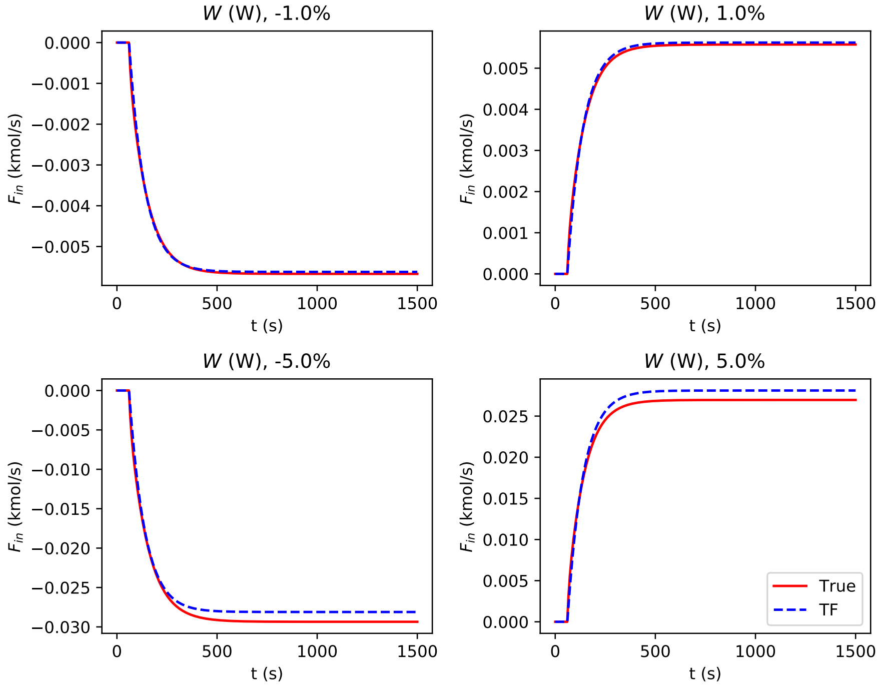

Figure A.29: Identification results for the pair F_{in} (kmol/s) x W (W)

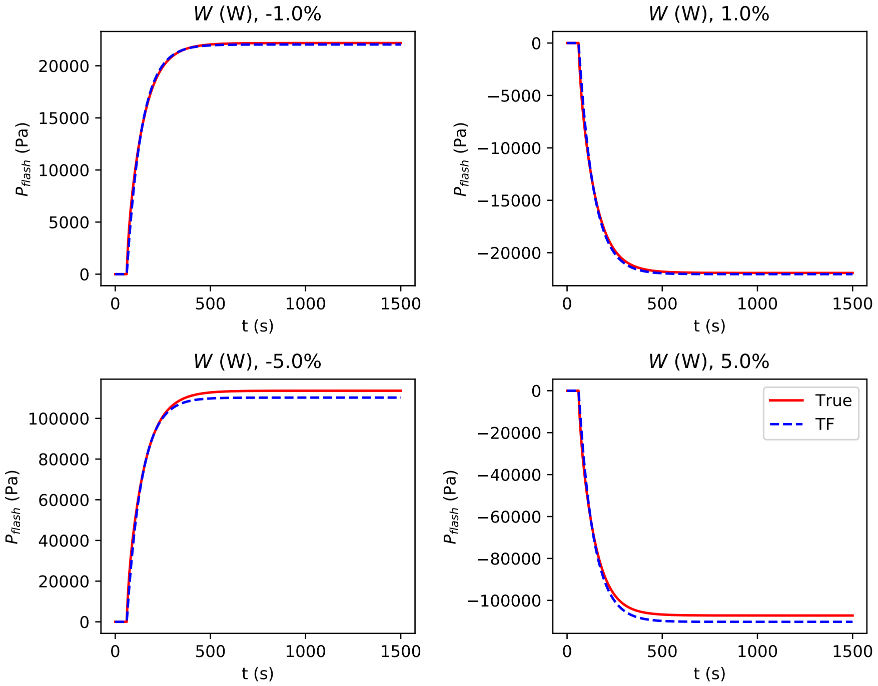

Figure A.30: Identification results for the pair P_{flash} (Pa) x W (W)

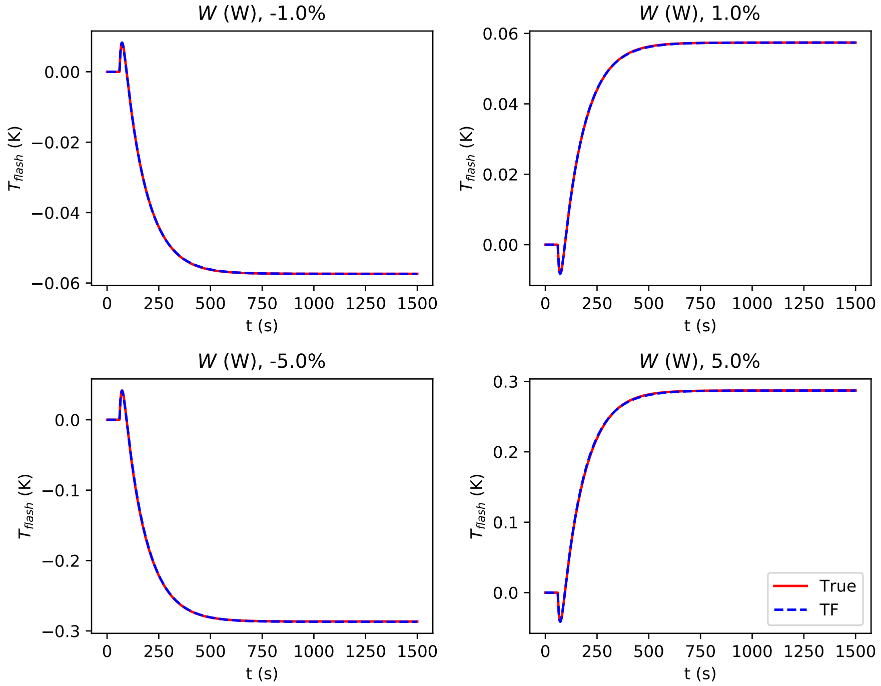

Figure A.31: Identification results for the pair T_{flash} (K) x W (W)

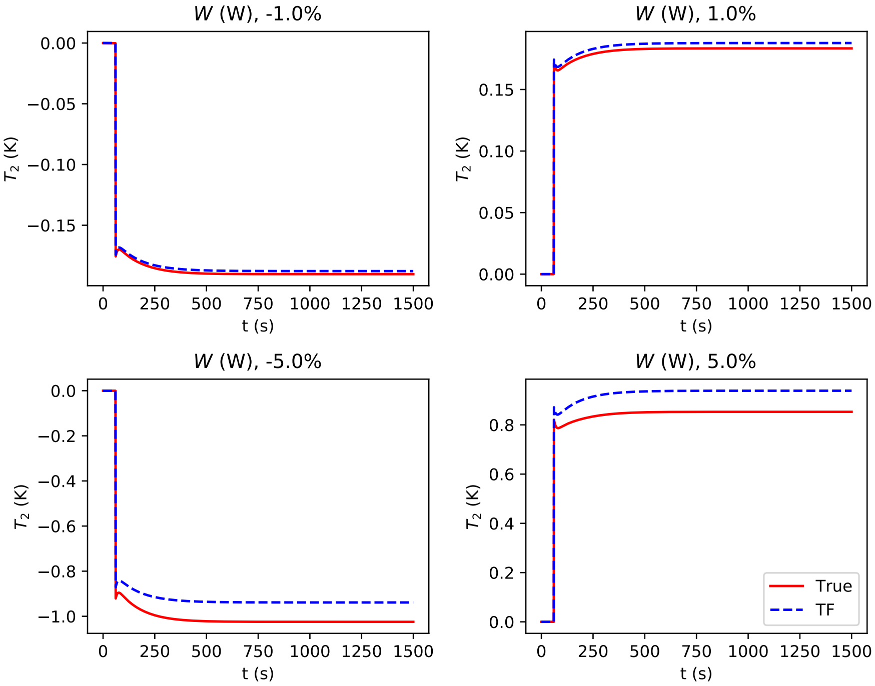

Figure A.32: Identification results for the pair T_2 (K) x W (W)

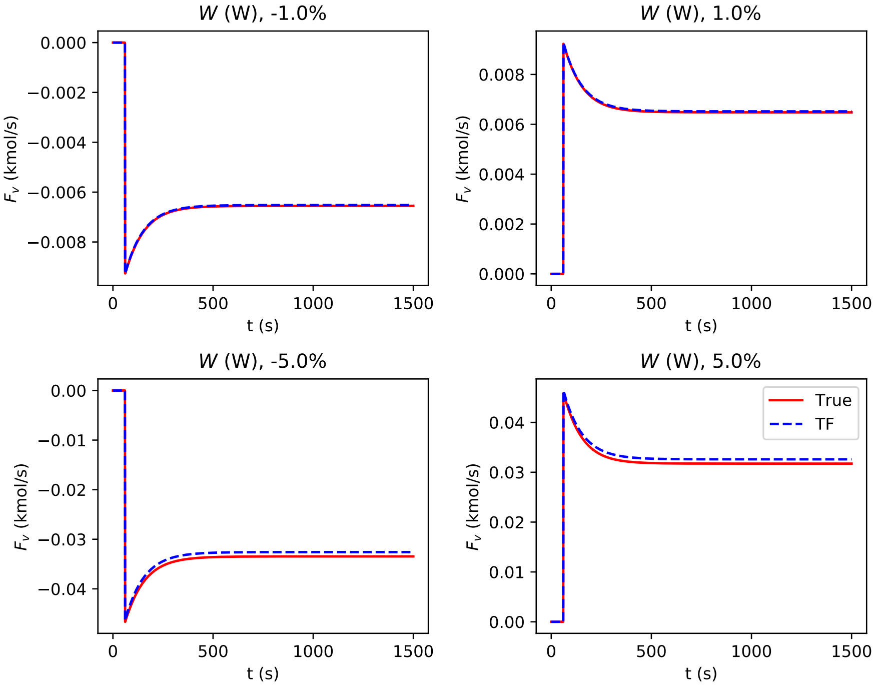

Figure A.33: Identification results for the pair F_v (kmol/s) x W (W)

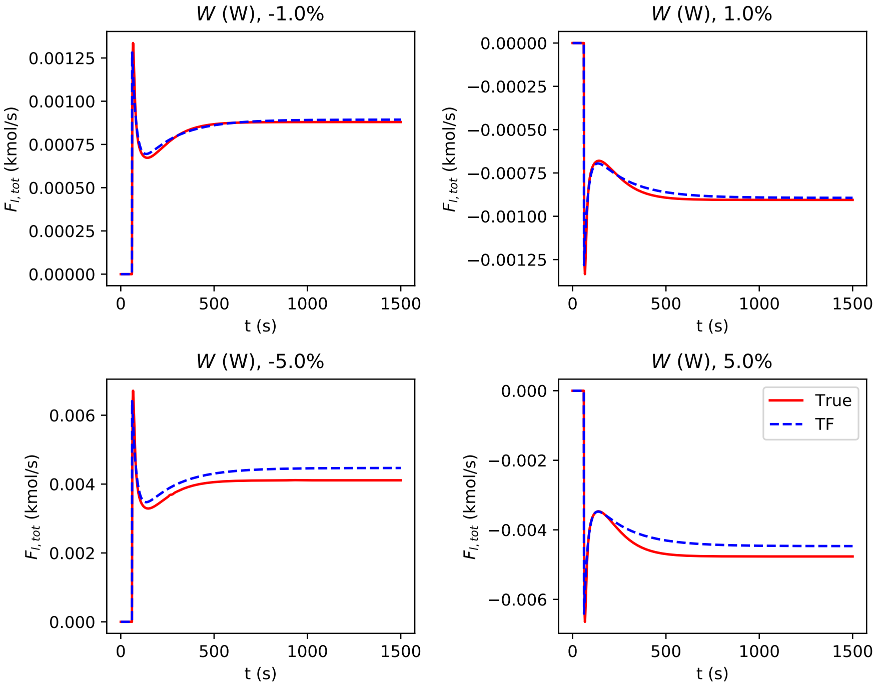

Figure A.34: Identification results for the pair $F_{l,tot}$ (kmol/s) x W (W)

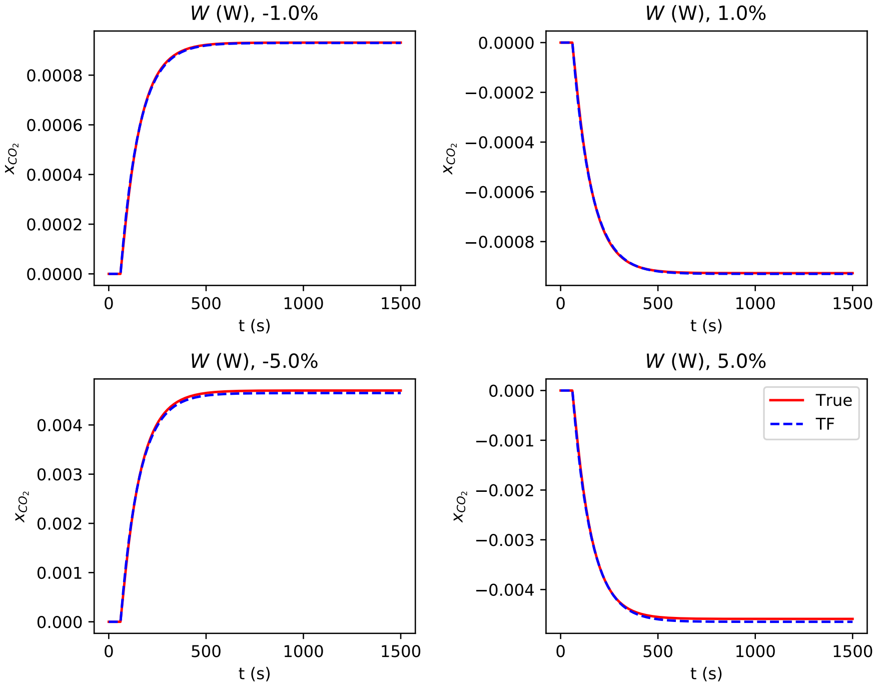

Figure A.35: Identification results for the pair x_{CO_2} x W (W)

Table A.1: Identified model parameters

Pair	Zeros	Poles	Gain
F_{in} (kmol/s) x $x_{v,t}$		-0.012914	0.00079704
P_{flash} (MPa) x $x_{v,t}$		-0.012915	-0.0031265
T_{flash} (K) x $x_{v,t}$	-0.0052529	-0.016765; -0.0056744	-0.0081611
T_2 (K) x $x_{v,t}$	-0.011613	-0.015881	1.3091
F_v (kmol/s) x $x_{v,t}$	-0.0086644	-0.011744	0.069968
$F_{l,tot}$ (kmol/s) x $x_{v,t}$	0.016141	-0.64228; -0.019574	-0.0078617
x_{CO_2} x $x_{v,t}$		-0.011286	-6.1062e-05
F_{in} (kmol/s) x x_v	-0.0079125	-0.010486	5.645
P_{flash} (MPa) x x_v		-0.01112	0.064039
T_{flash} (K) x x_v	0.2841	-0.47031; -0.010139	0.50185
T_2 (K) x x_v	0.12424	-0.61422; -0.010104	0.49423
F_v (kmol/s) x x_v		-0.010489	0.009295
$F_{l,tot}$ (kmol/s) x x_v		-0.61654	2.0745
x_{CO_2} x x_v		-0.010392	0.0032085
F_{in} (kmol/s) x Q_{aqu} (MW)		-0.015533	-0.0013851
P_{flash} (MPa) x Q_{aqu} (MW)		-0.01553	0.005434
T_{flash} (K) x Q_{aqu} (MW)		-0.012079	0.049073
T_2 (K) x Q_{aqu} (MW)		-0.012407	0.025662
F_v (kmol/s) x Q_{aqu} (MW)		-0.0071939; -0.073501	-6.215e-06
$F_{l,tot}$ (kmol/s) x Q_{aqu} (MW)	0.3237	-0.71773; -0.014549	0.0024889
x_{CO_2} x Q_{aqu} (MW)		-0.0085447	-6.254e-05
F_{in} (kmol/s) x Q_{resf} (MW)		-0.013069	-0.0017786
P_{flash} (MPa) x Q_{resf} (MW)		-0.013067	0.0069765
T_{flash} (K) x Q_{resf} (MW)		-0.011551	0.03835
T_2 (K) x Q_{resf} (MW)	-0.020418	-0.012621	3.1906
F_v (kmol/s) x Q_{resf} (MW)	-0.01559	-0.018892	-0.085869
$F_{l,tot}$ (kmol/s) x Q_{resf} (MW)	0.047388	-0.27409; -0.015471	0.0057969
x_{CO_2} x Q_{resf} (MW)	-0.004499	-0.0074535; -0.016529	4.4572e-05
F_{in} (kmol/s) x W (MW)		-0.012712	0.0016828
P_{flash} (MPa) x W (MW)		-0.012723	-0.0066043
T_{flash} (K) x W (MW)	0.034319	-0.11848; -0.0094582	-0.044136
T_2 (K) x W (MW)	-0.0099307; -0.10737	-0.0086335; -0.11501	4.1166
F_v (kmol/s) x W (MW)	-0.0074925	-0.010682	0.21884
$F_{l,tot}$ (kmol/s) x W (MW)	-0.012648; -0.012651	-0.0053482; -0.043421	-0.030537
x_{CO_2} x W (MW)		-0.010295	-2.2537e-04