MODEL PREDICTIVE CONTROL OF HIGH-INDEX DAE SYSTEMS WITHOUT INDEX REDUCTION

Charles Rajiv Cavalcante do Lago

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: Príamo Albuquerque Melo Junior
Maurício Bezerra de Souza Júnior

Rio de Janeiro
Fevereiro de 2019

MODEL PREDICTIVE CONTROL OF HIGH-INDEX DAE SYSTEMS
WITHOUT INDEX REDUCTION

Charles Rajiv Cavalcante do Lago

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. Príamo Albuquerque Melo Junior , D.Sc.

Prof. Argimiro Resende Secchi, D.Sc.

Prof. Bruno Didier Olivier Capron, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
FEVEREIRO DE 2019

Lago, Charles Rajiv Cavalcante do

Model Predictive Control of High-Index DAE systems without Index Reduction/Charles Rajiv Cavalcante do Lago. – Rio de Janeiro: UFRJ/COPPE, 2019.

XVIII, 75 p.: il.; 29, 7cm.

Orientadores: Príamo Albuquerque Melo Junior
Maurício Bezerra de Souza Júnior

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

Referências Bibliográficas: p. 61 – 67.

1. Differential-Algebraic Equations. 2. Model Predictive Control. 3. Process Identification. 4. Direct Initialization. I. Melo Junior, Príamo Albuquerque *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Química. III. Título.

I dedicate this work to my family, in special my father,
Héber César, and in memoriam of my grandfather,
José Borges Filho.

Acknowledgments

I am thankful to all cosmic entities who contribute directly or indirectly for this work and for maintaining this whole universe.

Among the human beings, I first would like to thank Ataíde Neto for his inspiring dedication and ever present willingness to help. To him I owe my journey on DAE understanding.

I also would like to thank Sergio Castaño, who gives support on control issues for thousands of desperate students and provided the control implementation used on simulations presented here.

I would like to thank all teachers who presented inspiring classes and make great effort to educate as much students as possible. Thank you Jackson Araujo for the stimulating classes on numerical calculus and control during graduation.

Thank you Argimiro Secchi for being this bright professor spreading knowledge to the humankind.

A special thanks to Príamo and Maurício, for being professors, advisors and friends during this journey through DAE and control fields of knowledge.

Thank you Dona Rosa and Rose for helping us by keeping the laboratory and our minds clean and organized.

Thanks to all my friends for supporting me during several situations and tolerate my social and philosophical daydreams.

Thanks to all authors whos compiled centuries of knowledge into books, articles and videos.

Special thanks to all women who contributed on building the wall of knowledge,

special remarks for Marie Curie and Linda Petzold for their contribution on chemistry, physics and DAEs.

This study was partially financed by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

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

CONTROLE PREDITIVO DE SISTEMAS DE EADS DE ÍNDICE SUPERIOR SEM REDUÇÃO DE ÍNDICE

Charles Rajiv Cavalcante do Lago

Fevereiro/2019

Orientadores: Príamo Albuquerque Melo Junior
Maurício Bezerra de Souza Júnior

Programa: Engenharia Química

O controle de processos tem, em sua representação matemática tradicional, a descrição de processos por meio de equações diferenciais ordinárias, EDOs. Uma alternativa mais geral de representar os processos dinâmicos a serem simulados e controlados é através de sua modelagem por meio de equações algébrico-diferenciais, EADs. O controle de processos é classicamente representado por controladores PIDs (Proporcional, Integral e Derivativo), tendo como camada superior o controle preditivo baseado em modelo matemático, ou apenas controle preditivo, MPC (*Model Predictive Control*). Há uma crescente atividade acadêmica no desenvolvimento do controle ótimo, uma abordagem de controle também baseada em modelo, na qual são utilizados algoritmos de otimização. O presente trabalho utiliza uma técnica de simulação computacional de sistemas de EADs, chamada inicialização direta, que consiste em inicializar o modelo a partir de uma condição estacionária e utilizar uma função de regularização para realizar a transição de uma condição à outra. Essa técnica foi utilizada para identificação e controle de quatro processos descritos por EADs de índice superior. Os processos são descritos por EADs de índice 2 sendo o modelo *benchmark* de condensador utilizado por Pantelides, dois modelos de reatores, um isotérmico e outro não isotérmico, e um modelo de *flash* reativo adimensional. Os processos descritos por modelos foram identificados para funções de transferência e controlados com um controlador MPC do tipo GPC (*Generalized Predictive Control*). Ficou evidenciado que a redução de índice pode levar a comportamentos incoerentes e os modelos identificados lineares, baseados em função de transferência, representaram adequadamente o comportamento local de modelos EAD de ordem superior.

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

MODEL PREDICTIVE CONTROL OF HIGH-INDEX DAE SYSTEMS WITHOUT INDEX REDUCTION

Charles Rajiv Cavalcante do Lago

February/2019

Advisors: Príamo Albuquerque Melo Junior
Maurício Bezerra de Souza Júnior

Department: Chemical Engineering

Process control has in its traditional mathematical representation the process description by means of ordinary differential equations, ODEs. A more general alternative for representing the dynamic process to be simulated and controlled is through its discription in differential-algebraic equations, DAEs. The process control is classically represented by PID controllers (Proportional, Integral and Derivative) and having in the upper layer the model predictive control strategies, MPC. It presenting nowadays a growing research activity on the development of optimal control, a control technique also based on a model, in which optimization techniques are used. The present work employs a computational simulation technique for DAE systems, known as direct initialization, which consists of initializing the model from a stationary condition and using a regularization function to carry out the transition from one condition to the other. This approach was used to identify and control four processes described by high-index DAEs. The processes are described by index 2 DAEs being the benchmark model of a condenser used by Pantelides, two models of reactors, one isothermal and other non-isothermal, and a non-dimensional model of reactive flash drum. The processes described by models were identified to transfer functions and controlled with MPC controller of the GPC (Generalized Predictive Control) type. It was evidenced that the index reduction can lead to incoherent behaviors and the identified linear models, based on transfer function, adequately represented the local behavior of high-index DAE models.

Contents

List of Figures

List of Tables

List of Symbols

B_i	Dimensionaless Heat of Reaction, p. 25
D_{ai}	Damköhler Number for Reation i, p. 25
K_i	Phase Equilibrium Constant, p. 25
K_{eq}	Reaction Equilibrium Constant, p. 22
$N1$	Minimum Costing Horizon, p. 12
$N2$	Maximum Costing Horizon, p. 12
Np	Prediction Horizon, p. 12
Nu	Control Horizon, p. 12
O_i	Higher Order of Derivative Found During Index Reduction , p. 32
η	Regularization Function, p. 8
γ_i	Dimensionaless Activation Energy in the Arrhenius Equation, p. 25
λ	Dimensionaless Heat of Vaporization, p. 25
ν	Differential Index of DAE Model, p. 32
ν_i	Index of Dependend Variable i, p. 32
ν_k	Zero Mean White Noise, p. 12
ϕ	Vapor Phase Fraction, p. 25
θ	Dimensionaless Temperature, p. 25
ξ	Regularization Parameter, p. 8
x_{iF}	Molar Feed Flow Rate Fraction of Component i, p. 25

x_i	Molar Liquid Phase Fraction of Component i, p. 25
y_i	Molar Vapor Phase Fraction of Component i, p. 25

List of Abbreviations

ARX	AutoRegressive eXogenous, p. 9
BDF	Backward Differentiation Formula, p. 7
CARIMA	Controlled Auto-Regressive Integrated Moving Average, p. 9
CSTR	Continuous Stirred Tank Reactor, p. 20
DAE	Differential Algebraic Equations, p. 2
DASSLC	Differential Algebraic System Solver in C, p. 17
DASSL	Differential Algebraic System Solver, p. 14
DAWRS	Differential-Algebraic Waveform Relaxation Solver, p. 33
DMC	Dynamic Matrix Control, p. 2
EMSO	Environment for Modeling, Simulation, and Optimization, p. 32
FODT	First Order plus Dead Time, p. 36
GPC	Generalized Precitive Control, p. 9
IMC	Internal Model Control, p. 9
LTI	Linear Time Invariant, p. 44
MIMO	Multi-Input Multi-Output, p. 13
MPC	Model Predictive Control, p. 1, 2
MSE	Mean Squared Error, p. 42
NMPC	Nonlinear Model Predictive Control, p. 2
ODE	Ordinary Differential Equation, p. 8
PEQ	Chemical Engineering Program of COPPE/UFRJ, p. 33

PID	Proportional Integral Derivative, p. 2
PSIDE	Parallel Software for Implicit Differential Equation, p. 14
QDMC	Quadratic Programming Solution of Dynamic Matrix Control, p. 9
SISO	Single Input Single Output, p. 11
SODT	Second Order plus Dead Time, p. 36

Chapter 1

Introduction

As a way for clarifying the reader about the thoughts that drove the research on this dissertation, some information about the subject matter addressed here is presented and the inspiration behind it is pointed out.

1.1 Motivation

Controlling a process is somehow applying our human knowledge to different types of equipment and driving the desired variables through a trajectory that please us under certain criteria. To control a process one needs to know it minimally, such by describing it through a phenomenological mathematical model or by considering it as a "black box", which only looks for the input and output data and design the controller with an amount of data that gives reasonable results (ROSSITER, 2003). As the tuning is still a non-intuitive step to have the full control structure running (WASCHL *et al.*, 2011), the computational simulation is an important key in the implementation of control strategies.

The development of the MPC (Model Predictive Control) added resources to the control theory by allowing engineers and researchers to use more complex mathematical models to describe the process phenomena and deal with multivariable systems more easily, without the need to design complex decoupling structures such those needed for PID (Proportional Integral Derivative) controllers (CAMACHO and BORDONS, 1999). One of the great advantages of the MPC over traditional PID is that it foresees the process behavior based on an internal model. How far in time the model predict the future behavior of the process is called prediction horizon.

The MPC refers to a philosophy of control, having several different algorithms using certain common principles. One principle is that predictions are performed using a predicting model; this model can be any suitable one for the process (ROSSITER, 2003). With this in mind, the model can be obtained from a step response, such as in DMC (Dynamic Matrix Control) (CUTLER and RAMAKER, 1979), or from a nonlinear model, such as in the case of NMPC (NonLinear Model Predictive Control) (QIN and BADGWELL, 2000). This freedom in choosing the model used for prediction step gave rise to the idea of using a model based on DAEs (Differential-Algebraic Equations) in the control area (ROSSITER, 2003). Models based on DAEs are powerful tools for describing physical problems, where the variables of interest can be expressed by different equations and constitutive relationships are easily replaced (BRENAN *et al.*, 1996).

DAE systems are an active research field nowadays, being of interest for engineers in control, chemical, mechanical and electrical process (AZEVEDO-PERDICOÚLIS and JANK, 2007; BAUM *et al.*, 2017; HACHTEL *et al.*, 2018; HASSKERL *et al.*, 2017; HÖCKERDAL *et al.*, 2018; LI, 2010; WANG, 2010; YE *et al.*, 2017), with researchers in diverse fields making great effort to combine the DAE theory to control strategies, from classical PID to more advanced ones such as MPC or optimal control (BIEGLER *et al.*, 2012). The need for a more complete and general theory for DAEs and algorithms that can deal with those systems inspires researchers to follow different approaches and allows precious contributions from different fields.

Researchers and students have growing interest on DAE systems as can be seen in the willingness to improve teaching and presenting this subject even at the level of undergraduate students (VIANNA JR and NASCIMENTO, 2005). The characteristic of DAE models to represent physical phenomena and its more general aspect regarding ODEs tend to make researchers investigate their proprieties and intensify the study on this subject in order to make it more general.

The current developments in DAE simulation and control are frequently based on avoiding dealing directly with high-index systems, by applying some strategy to converting then into ODEs or index-1 DAEs. Several solvers for high-index systems uses some index reduction to convert the system into an index-1 or ODE to solve it, but they do not compare or solve the original system, which can have a different behavior when submitted to disturbances.

This dissertation reviews the literature in Section 2.5 on high-index DAE for simulation and control and shows how to deal directly with high-index systems by using a solver capable of integrating high-index DAEs in its implicit form, which is the most generic form that DAEs appears. Rigorous techniques for consistent initialization are in general difficulty to apply and this drives the opportunity of applying simpler strategies to overcome such difficulties (KRÖNER *et al.*, 1997). The direct method of initialization has its benefits due to the great difficult of initializing DAE systems using standard solvers. Finding consistent initial conditions is one of the hardest tasks when dealing with DAEs, especially those of high index. Additionally, the performance of a model predictive control strategy in keeping a high-index system at a stationary point is investigated. The impact of index reduction is analyzed when performing the identification of the processes to apply the MPC control.

1.2 Objective

The present work applies an approach called direct method of initialization to simulate computationally DAE systems to complete three main objectives:

i. Identify high-index DAE models to transfer function models, using “black box” identification;

ii. Apply control strategies to high-index DAE-based processes using the identified transfer functions, where the high-index DAE models are used as virtual plants;

iii. Study the impact of index reduction when simulating disturbances in the DAE models.

The first and second objectives are due to the difficulty in simulating high-index DAEs. The literature provide just a few general technique for solving a generic DAE system, the mostly techniques presented are usually restricted to DAEs of a certain structure or require some index reduction to solve the system. The third objective is a consequence of the first two, as the high-index solution is not always known and the index reduction is the most common approach when dealing with those systems. Besides the well-known drift-off effect (SOARES and SECCHI, 2005), the impact of such technique is not usually mentioned, making possible to researchers drawing conclusions based on incorrect behavior when simulating the reduced systems in substitution of the original DAE.

1.3 Dissertation Structure

Chapter 1 presents the motivation of this dissertation and the objectives defined to complete it. Chapter 2 provides an overview about important fundamentals regarding DAEs and MPC. The specific requirements for simulating high-index DAEs via direct initialization are provided and topics related to effects of index reduction are covered. The MPC section presents a historic review and the specific aspects regarding the MPC algorithm chosen. Chapter 3 explains the methodology applied to simulate systems described by high-index DAE models in order to control them with a linear MPC. In Chapter 4 the results of the simulation performed for identification and control are presented and the discussion about the effects observed. The conclusion from this study, its results and the methodology proposed are presented in Chapter 5.

Chapter 2

Literature Review

This chapter brings relevant information to understand Differential-Algebraic Equations and its current relationship with Model Predictive Control. Information about important characteristics when numerically simulating DAEs and its difficulties are presented, as well as the possible benefits of working with this type of equation system. The fundaments of Model Predictive Control are presented in order to introduce the reader about this advanced control strategy. The last topic in this chapter describes the research carried out in these two fields concomitantly, providing the basis for the development of this work.

2.1 Differential-Algebraic Equations (DAEs)

The representation of a DAE system in its fully implicit form is given according to Equation 2.1:

$$F(x, x', z, t) = 0 \tag{2.1}$$

in which the differential variables are x and the algebraic variables are represented by z. The term x' represents the derivative of x with respect to t and t is the independent variable, usually time. These systems occur when there is a coupling of differential equation with algebraic equations, also called constraints. Systems in the form of DAEs arise naturally in the process of phenomenological modeling in various engineering fields (BRENAN *et al.*, 1996). The most common way for representing and dealing with DAEs is in the semi-explicit by showing, clearly, which are the differential and the algebraic equations, as expressed in Equation 2.2:

$$\begin{cases} x' = F(x, z, t) \\ 0 = G(x, z, t) \end{cases} \tag{2.2}$$

The development of DAE's theory is intimately related to initial value problems. But the differences between the ODEs and DAEs arise when trying to numerically simulate them as pointed by PETZOLD (1982). DAE systems require that all equations be solved simultaneously and solvers addressed to solve ODEs are not always suitable for DAEs depending on its structure. Different types of DAEs are reported by BRENAN *et al.* (1996) as linear constant coefficient, linear time varying, semi-explicit linear, semi-explicit nonlinear. This work focuses on fully implicit DAEs, as given in Equation 2.1. The solution of DAEs differs from the ODEs by the difficulties associated with their simulation. The DAE systems require consistent initial conditions to start the simulation while the ODEs can have arbitrary initial conditions. In order to simulate a DAE systems, one needs to characterize it beforehand, describing the index and the dynamic degrees of freedom (LEITOLD and GERZSON, 2010).

The solution of DAE systems can be performed by different approaches: the so called direct or indirect methods (VIEIRA and BISCAIA JR., 2001). There is also a structural approach proposed by PANTELIDES (1988) in which graph theory is used to determine the minimum set of equations to be differentiated in order to convert the original system into an equivalent form of explicit ODEs. Several works applied this technique (COSTA JR., 2003; MURATA, 1996; SOARES and SECCHI, 2005). This approach of reducing DAEs to ODEs requires the symbolic or numerical evaluation of several derivatives, increasing the size of the problem and inserting new variables, that usually indicate how much the constraints have been violated. These drawbacks make attractive using the DAE system on its original form.

Some benefits of working directly with DAE systems are: (i) it is not always possible to reformulate the DAE system to an explicit ODE system, (ii) the variables keep their original physical interpretation, (iii) it is easier to vary parameters and relationships with implicit systems, (iv) the algebraic equations typically represent conservation laws, that should be kept invariant. Others benefits can be found in BIEGLER *et al.* (2012) and MURATA (1996).

The literature reports that a possible effect when working with index reduced systems is the called "drift off" effect, in which the reduced system presents increasingly integration errors compared to the original system. The effect is exemplified in SOARES and SECCHI (2005) for the classical index-3 pendulum system described in BRENAN *et al.* (1996). Another difficulty, when working with reduced systems, is that it accepts more solutions than the original system and the differentiation process may cause eventual loss of process information.

2.1.1 Index of a DAE System

As discussed above, the solution of DAE systems presents more difficulties compared to purely differential systems. An indication of the difficulty when dealing with DAE systems is the so called differential index, defined as the minimum number of times that all or part of the original system has to be differentiated relatively to the independent variable, usually time, so the system is reduced to a system of ODEs. According to this definition, a system of ODEs is a particular case of DAEs, in this case of index 0 (BRENAN *et al.*, 1996). This and several other index definitions can be found in SCHULZ (2003). To exemplify the most common index characterization one may start with an index-1 DAE as presented in Equation 2.3, where all the algebraic variables are explicit in the algebraic equation, $G(x, z, t)$, and only one differentiation is needed to have an explicit system of ODEs. By differentiating $G(x, z, t)$ with respect to t, it produces an explicit expression representing z' and z, characterizing the differential equation for z.

$$\begin{cases} x' = F(x, z, t) \\ 0 = G(x, z, t) \end{cases} \tag{2.3}$$

When the algebraic variables, z, are not present in the algebraic equations, it can be seen that the system is of high-index (index 2 or higher), as more than one differentiation will be needed to have an explicit set of ODEs, as exemplified in Equation 2.4.

$$\begin{cases} x' = F(x, z, t) \\ 0 = G(x, t) \end{cases} \tag{2.4}$$

In this case, the expression $G(x, t)$ needs to be differentiated at least twice to have an explicit expression for z' and z, indicating that the index of such system is equal to or greater than 2. There are cases where more differentiations are needed and the variables present different indices, as exemplified with Equation 2.5.

$$\begin{cases} x' = F(x, y, t) \\ y' = K(x, y, z, t) \\ 0 = G(x, t) \end{cases} \tag{2.5}$$

The example above is at least an index-3 system, as the algebraic expression $G(x, t)$ needs to be differentiate three times to have an explicit expression for z' and z. The variables x and y are differential ones and z is the algebraic variable. While reducing the index of the whole system, 2.5, the index of each variable is determined, for this case the indices of x, y and z are at least of index 1, 2, and 3, respectively. More details about the procedure to determine the index of each variable can be found in LIOEN *et al.* (1998).

Just reinforcing the concept, the index determination is performed by differentiating the minimum number of equations to have an explicit set of ODEs. As the index reduction needs to be carried out to determine which is the differential index of the system, the researchers tend to use this technique to solve high-index systems by reducing the index and solving the reduced system. The solution of high-index DAE system via index reduction can be done by different ways: (i) direct differentiation, which consist on differentiating the entire set of equations until it becomes an explicit set of ODEs; (ii) differentiation and substitution, in which some algebraic constraints are differentiated and substituted on the differential equations; and (iii) forming an augmented DAE system with the new equations discovered by the differentiation process (SANTAMARÍA and GÓMEZ, 2015). These approaches have the inconvenience of possible causing loss of information through differentiation, increasing the size of the problem by inserting new variables or violating certain relationships that should be kept invariant, as already discussed above.

As the DAE systems presents different types of behavior depending on the equations involved, the index of a system may be hard to find and can also fluctuate depending on the dynamics or the initial condition given (QUINTO, 2010). Also high-index systems may arise depending on the assumptions adopted such as equilibrium, incompressibility and fast reaction kinetics among others (KUMAR and DAOUTIDIS, 1999).

2.2 Direct Method of Initialization

This approach uses the property that a steady state is always a consistent initial condition, as demonstrated by KRÖNER *et al.* (1997) and used by VIEIRA (1998). The dynamic behavior can be interpreted as a disturbance in a steady state, as reported by VIEIRA and BISCAIA JR. (2001) in several examples.

Another inconvenience when dealing with DAEs is that the methods for integration are susceptible to discontinuities during the process. If a discontinuity is present during the integration, the integration process may fail and another set of consistent initial condition has to be calculated in order to proceed the solution, which is a characteristic of multi-step methods, such BDF (Backward Differentiation Formula). Abrupt changes in parameters or variables tend to make the integration process fail and DAE solvers are typically sensitive to sudden changes (VIEIRA, 2001).

As a way of guaranteeing the continuous integration of the systems submitted to discontinuous disturbances such the step changes, the smoothing of the disturbance can be adopted through a regularization function, $\eta(t - ts, \xi)$, with the characteristic of being continuous and limited between 0 and 1. The argument $(t - ts)$ represents the time when the transition occurs, growing from 0 to 1 continuously. The parameter ξ defines how close to the ideal step the function behaves (VIEIRA and BISCAIA JR., 2001). Equation 2.6 follows the formulation used by (QUINTO, 2010). Other works proposed different functions, depending on the system simulated (VIEIRA, 1998; VIEIRA, 2001):

$$\eta = \frac{1 + tanh[\xi(t - t_s)]}{2} \tag{2.6}$$

Using this regularization function, perturbations in discrete domain are applied as described by Equation 2.7. This equations makes the transition between two conditions in the process, using different inputs, u. Starting the simulation with the previous input, u_{k-1}, and applying the regularization function, η, to switch from the previous to the current input, u_k. Using the previous input, the integration starts are under consistent initial conditions and the function 2.6 switches from the previous to the new condition. The transition behavior depends on parameter ξ and the time t_{ks}, which defines the time where the transition will occur.

$$u_k = u_{k-1} + (u_k - u_{k-1}) \cdot \eta(t - t_{ks}, \xi) \tag{2.7}$$

With this approach, the simulation of the control actions remain connected, guaranteeing the progress of the integration method under perturbations applied between one sampling time and another.

2.3 Process Identification

Process identification is a way of establishing a direct correlation model between input (manipulated or disturbed) variables and output (controlled or not) variables of a system, from data showing variations in inputs and their respective impacts on outputs. The identification process is fundamental to a number of control engineering practices. The process model provides essential information about the behavior of the process over time and relevant characteristics for control design.

Process identification emerges when dealing with processes whose governing phenomena are not clearly understood or are too complex or not suitable for the proposed application (OGUNNAIKE and RAY, 1999). There are several forms and approaches for modeling and identification. In general, real processes are not known in their ultimate reality, the models are abstract ways for describing the world as we conceive it. When models based on phenomenological laws are not suitable for simulation or are impracticable, the process of identification is conducted on the basis of available information, usually through the use of input and output data. The so-called "black box" methods only take process data for parameters estimation and are not necessarily valid throughout the whole operational range, while "gray box" methods use some phenomenological or empirical information about the process together with available process data (OGUNNAIKE and RAY, 1999).

The identification of linear models takes advantage of the characteristic that even non-linear processes can be represented by a linear model in many cases, if sufficiently close to the collection of process data used for identification. MATLAB$^{\text{TM}}$ environment has a tool with several possibilities for process identification, namely, the System Identification Toolbox$^{\text{TM}}$, which uses several internal tools to identify linear and non-linear models, using techniques such as maximum likelihood, prediction error minimization, Hammerstein-Wiener modeling, ARX (AutoRegressive eXogenous) models among others (MATHWORKS, 2018b).

2.4 Model Predictive Control

Philosophically, MPC can be interpreted as reflecting the human behavior regarding the control strategy (ROSSITER, 2003). This feature presents the relatively intuitive way in which the algorithms that follow this strategy can be developed, since our behavior in control situations can be exemplified according to the concepts used in the various algorithms covered by the MPC strategy. The historical determination of the beginning of MPC is difficult to do, since these techniques have been developed and applied over several years by the industry until its presentation through academic research (MACIEJOWSKI, 2002). The basic concepts date back to the 1960s, with the concept of receding horizon by PROPOI (1963) (OGUNNAIKE and RAY, 1999). The academic development of this type of strategy began in the late 1970s, with the development of the algorithms MAC (Model Algorithmic Control) by RICHALET *et al.* (1978), better known by the name of the software in which it was implemented, IDCOM (OGUNNAIKE and RAY, 1999), and the widely diffused DMC (Dynamic Matrix Control), recognized as one of the most popular predictive control algorithms, published by CUTLER and RAMAKER (1979). Over the years, improvements have been proposed based on the DMC, such as the QDMC (Quadratic Programming Solution of Dynamic Matrix Control) by (GARCIA and MORSHEDI, 1986).

In 1987, the algorithm named GPC (Generalized Precitive Control) by (CLARKE *et al.*, 1987a,b) was published mentioning that the new algorithm would incorporate the family of long-range predictive controllers, being suitable for systems difficult to control before, such the ones with multiple dead times, non-minimal phase and even open-loop unstable systems (CLARKE *et al.*, 1987a). The original GPC algorithm has several vulnerabilities, such as a lack of stability and robustness, noise susceptibility among others, but several proposals were presented and the robustness of the GPC controller could be improved according to certain modifications (ROSSITER, 2003). The use of low-pass filters, such as the T filter, and the application of the Internal Model Control (IMC) paradigm, proposed by GARCIA and MORARI (1982), have improved the performance and enabled a better understanding of the stability, robustness and performance of MPC proposals (OGUNNAIKE and RAY, 1999). The GPC was originally proposed with the CARIMA (Controlled Auto-Regressive Integrated Moving Average) model for prediction, but its formulation may be adequate for state space representation or following the internal model proposal (ALBERTOS and ORTEGA, 1989; ROSSITER, 2003).

The MPC approach has some basic fundamentals which makes possible to adapt the control to the most varied processes. The design and implementation of a MPC strategy, according to any algorithm, can be done in an intuitive and properly way (ROSSITER, 2003). To exemplify how the MPC performs the control action, with the predictive aspects, the following analogy with human behavior is presented. As stated earlier, human action, when facing control situations, tends to contain certain characteristics in common with MPCs. The prediction horizon can be interpreted as how far in the future one makes predictions. For humans, it would be how long are the forecasts about future events. As in human experience, the predictions on MPC are performed every moment, at each sampling time the controller foresees how the plant will behave based on its internal model. Another characteristic is the control horizon, which can be linked to the amount of actions that will be taken in order to drive the plant from one situation to the desired trajectory. The control horizon is the number of actions calculated, which is the degrees of freedom of the system and is usually chosen to be smaller than prediction horizon. The optimization is the step where the best control actions are chosen in order to achieve the desired objective. As the concept of optimality is relative, the optimization evaluates the performance with an objective function, considering the factors that the controller designer understand to be better to describe the problem. A scheme illustrating a generic MPC strategy is presented in Figure 2.1 (CARVALHO, 2015; OGUNNAIKE and RAY, 1999). Normally the control horizon is smaller than the prediction horizon, which allows to reduce the number of calculations made by assuming the control actions after the control horizon will remain constant. This concept was implemented in DMC and used on GPC (CLARKE *et al.*, 1987a). The whole process of predicting future behavior based on a model and determining the next control actions is carried out at every sampling time, just like humans do, by constantly making new projections based on obtained information and the results generated. This characteristic of

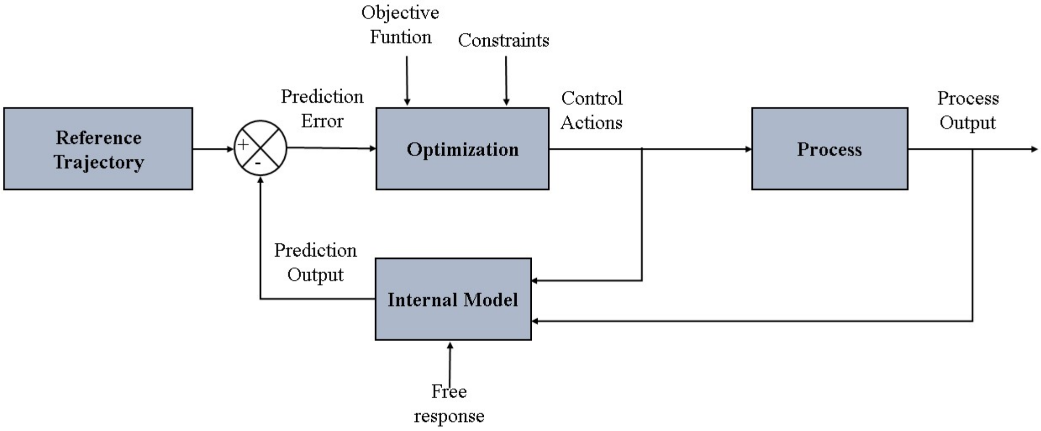


Figure 2.1: Generic scheme of a MPC strategy.

constant updating the predictions refers to the concept of receding horizon. These concepts can be visualized in Figure 2.2.

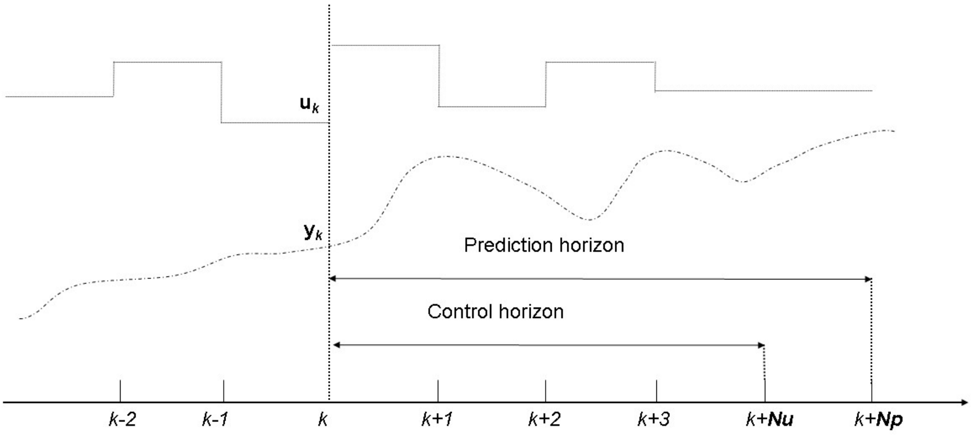


Figure 2.2: Diagram of prediction and control horizon on MPC strategy.

Figure 2.2 schematically represents how MPC sees the events in a SISO (Single Input, Single Output) application, k is the sampling time, u_k is the control action, y_k is the plant behavior. Np and Nu are the prediction and control horizons, respectively. At every sampling time, indicated in Figure 2.2 as k, the whole process to determine what will be the next sequence of Nu actions is performed, but only the first action is effectively implemented. For the GPC algorithm, the prediction starts from a minimum horizon, $N1$, and extends through Np sampling times until $N2 = Np + N1$, this allows dealing directly with systems with dead time. The approach in DMC starts from the first sampling time, making predictions from over the entire prediction horizon which can be a waste of computational time in cases of long dead time.

2.4.1 CARIMA Model

The acronym for Controlled Auto-Regressive Integrated Moving Average, CARIMA, is a popular model for representing transfer functions in the Z-domain with the inclusion of a stochastic term, $\frac{T(z)\nu_k}{\Delta}$. Its standard form is (ROSSITER, 2003):

$$a(z)y_k = b(z)u_k + \frac{T(z)\nu_k}{\Delta} \tag{2.8}$$

The terms $a(z)$ and $b(z)$ are polynomials of z-domain containing the parameters of the model, the denominator and numerator of a transfer function, respectively. The term $T(z)$ is usually considered as a control design function, which can act as a low pass filter in a closed loop performance. The term ν_k and Δ are a zero mean noise and the discrete delay operator $(1 - z^{-1})$, respectively. By incorporating a stochastic term into the model, the CARIMA model intrinsically incorporates the possibility of rejecting small disturbances in the controlled variables (CAMACHO and BORDONS, 1999). The common use of the CARIMA model is according to Equation 2.9, with the term $a(z)\Delta$ grouped as $A(z)$ (ROSSITER, 2003):

$$A(z)y_k = b(z)\Delta u_k + T(z)\nu_k \tag{2.9}$$

The CARIMA model inside GPC allows the control to reject low frequency disturbances in the process by internally incorporating the term $\frac{T(z)\nu_k}{\Delta}$, which can be understood as a representation of disturbances, making it possible to control process even in cases of uncertainty of parameters or mismatch between model and plant (ROSSITER, 2003). The term $T(z)$ usually assumes the value of 1, but other polynomials can be proposed to improve the disturbance rejection in different frequencies (ROSSITER, 2003).

2.4.2 Generalized Predictive Control (GPC)

The formulation of GPC proposed by CLARKE *et al.* (1987a) uses the CARIMA model to predict the plant behavior and a polynomial division technique for calculating future responses. Predictions are made for the entire prediction horizon using complicated algebra with the Diophantine equation. However, other methods can be implemented in order to make the algebra for the predictions calculation simpler and clearer, as proposed in ALBERTOS and ORTEGA (1989), which uses the characteristic of the CARIMA model to predict one step ahead and organizes the prediction matrix using recursion through a Toeplitz-type matrix, according to the matrix H below, which is well conditioned and of simple inversion (ROSSITER, 2003). The terms hi are the impulse response coefficients obtained from the transfer function models which gives the prediction N steps ahead.

$$H = \begin{bmatrix} h_1 & 0 & 0 & 0 \\ h_2 & h_1 & 0 & 0 \\ \vdots & \ddots & h_1 & 0 \\ h_N & h_{N-1} & \cdots & h_1 \end{bmatrix} \in \boldsymbol{R}^{\boldsymbol{NxN}} \tag{2.10}$$

The objective function used in the GPC is posed according to Equation 2.11, in which the process response, y_{k+j}, is compared with the reference trajectory, y_{ref}, and the increment of control actions, Δu_{k+j-1}, are accounted for and weighted by the factors, $\delta(j)$ and $\lambda(j)$, respectively. The weighting factors can be used to compensate the scales of the variables, if nondimensionalization is not performed *a priori*.

$$F_{obj} = \sum_{j=N1}^{N2} \left\{ \delta(j) \left[y_{k+j} - y_{ref} \right]^2 \right\} + \sum_{j=1}^{Nu} \left\{ \lambda(j) \left[\Delta u_{k+j-1} \right]^2 \right\} \tag{2.11}$$

The optimizer will minimize the objective function in the interval between 1 and $N2$ relative to control actions, Δu. After the control horizon, the control increments are set to zero in order to reduce the computational effort. Originally, the GPC was proposed without restrictions, however, the structure of the GPC and of the MPCs in general allows the inclusion of restrictions during the optimization process. This makes the solution difficult, but it is a great advantage of the MPC over classical control, as well as easily dealing with MIMO (Multi-Input Multi-Output) systems in a simple and direct way (OGUNNAIKE and RAY, 1999).

2.5 Simulation and Control of DAEs

Researchers and engineers from different fields faced high-index DAE systems as a challenge for several years. The simulation of such systems began with simple cases and research studies started to recognize that DAE systems differ from ODEs, but have an intrinsic relationship with stiff ODEs. PETZOLD (1982) published an article explaining the differences between these two types of mathematical models. An active area of research is the simulation of process models that are typically described by high-index models, like the reactive flash and reactive distillation (BONILLA *et al.*, 2012; COSTA JR., 2003; GONÇALVES *et al.*, 2007; HARNEY *et al.*, 2013; SANTAMARÍA and GÓMEZ, 2015).

Regarding simulation of DAE systems, the first time a direct method was used to the consistent initialization of a DAE model appears to be in the work of CUILLE and REKLAITIS (1986), where a cubic polynomial is used to reinitialize the derivatives and physical information is used to give consistent initial conditions, starting from a stationary state. The most famous code for integration of DAEs is the DASSL code (Differential Algebraic System Solver), suitable for index-1 systems and published by PETZOLD (1989).

LEIMKUHLER *et al.* (1991) used forward finite differences to approximate high-order derivatives and employed user information to define consistent initial condition to solve index-1 and semi-explicit index-2 DAE systems.

ALBET *et al.* (1994) proposed a modified Euler method to calculate the derivatives after suddens changes, such as step inputs, by dropping the error control of the integration method.

CAMPBELL and MOORE (1994) proposed the solution of high-index DAEs by iterative minimum squares using a derivative array, solving the augmented system composed of the original equations plus the derivatives of the algebraic equations. The method was not suitable for control studies because it was computationally demanding due to decomposition of the Jacobian matrix in singular values.

GOPAL and BIEGLER (1998) applied a linear programming approach to find consistent initial conditions to reinitialize the integration after discontinuities. MURATA (1996) and COSTA JR. (2003) applied automatic differentiation to create the augmented system in order to solve high-index systems for the Pantelides condenser, distillation column and pendulum.

VIEIRA (1998); VIEIRA (2001) used the direct method of initialization to simulate DAE systems with different regularization functions.

SOARES and SECCHI (2005) proposed a direct method of initialization based on a new algorithm for index reduction that proved to be more robust than the classical codes.

GERDIN (2006) used the MODELICA language to simulate DAEs by applying the Pantelides method for index reduction.

GONÇALVES *et al.* (2007) used the MATLAB$^{\text{TM}}$ solver for stiff ODEs, namely, ODE15s, to solve a reduced system describing a flash drum.

SOARES (2007) developed a simulation environment capable to solve high-index DAEs by applying an improved version of Pantelides algorithm for index reduction.

The simulation of an index-2 system describing a flash separation process is presented by LIMA *et al.* (2008), the authors used PSIDE, the high-index solver suitable for systems up to index 3 (SOARES and SECCHI, 2005).

LI (2010) used the MODELICA language to simulate centrifugal chillers by applying the direct method of initialization in different conditions of operation and control.

BONILLA *et al.* (2012) simulated a distillation process in packed columns by reducing the model from index 2 to index 1. A structural analysis was performed by MCKENZIE *et al.* (2015) in order to reduce DAEs to index 1 or 0 using MODELICA language.

The control studies focusing on DAE approach seems to start with MCLELLAN (1994) who applied a DAE perspective on nonlinear control, by solving the whole control problem as a DAE system, but the process was described by a set of ODEs.

KUMAR and DAOUTIDIS (1995a,b, 1996, 1999) explored the control of DAE systems of indices 1 and 2 with feedback regularization and applied the differentiation of the whole system to convert the DAEs into ODEs. They also used algebraic manipulation of the constraints in order to insert the algebraic variables into the ODE equations.

The development of control and DAE focused on optimal control simulation in this new millennium, presenting several works dealing with optimal control, but usually separating the process from the constraints. The whole problem is characterized as an DAE, but the processes are usually described by ODEs.

CONTOU-CARRERE and DAOUTIDIS (2005) developed a method to represent DAEs in state space form for feedback control by changing variables in order to remove the control variables from hidden constraints.

SANTAMARÍA and GÓMEZ (2015) created a hybrid index DAE model for a NMPC, using the reduced index-1 model to compute the consistent initial conditions and them switched to the index-2 model in order to reduce the computational effort and the size of the system solved.

UPPAL *et al.* (2017) used the DAE approach to simulate the optimal control of an aerial problem. The process itself is a system of ODEs and the constraints characterize the DAE system. The final model is discretized with the control variables and the alpha method is used to integrate the system (CAMPBELL, 1995; PARIDA and RAHA, 2009).

2.6 Remarks on the Review

The information presented above aims at providing the reader with the basic concepts and ideas about DAEs and MPC. The main point about these two subjects together is that the research on simulating and controlling those types of models is still under development and the MPC strategy has been improved by dealing directly with DAEs due to its generic aspect in modeling and optimization. It was found that the common practice when facing high-index DAE models is to apply some strategy to reduce the index and solve the reduced order system by the most common solvers.

The simulation of high-index DAE models has been proved to be useful to improve the understanding of diverse processes and the DAE formulation allows the researcher to test different constitutive relationships, as well as to test different correlations between variables and parameters in the models. It was found that researchers tend to avoid dealing directly with high-index DAE models in control studies, when these types of models appear, the most common practice is to reduce the index.

This work deals with this type of models directly, with no algebraic manipulation or index reduction approach applied in order to solve the resulting system. The applied approach was directed to GPC algorithm, which is a control technique that deals with wide classes of systems described by the CARIMA model. More detailed information about DAE can be found on (BIEGLER *et al.*, 2012; BRENAN *et al.*, 1996; GERDIN, 2006; SCHULZ, 2003) and for a detailed explanation on MPC the following references are recommend (CAMACHO and BORDONS, 1999; OGUNNAIKE and RAY, 1999; ROSSITER, 2003).

Chapter 3

Methodology

The following information is a description of what was performed in order to simulate and control process systems described by high-index DAEs models. A flowchart illustrating the sequence of each stage is presented in the beginning of the chapter and the case studies are described next. An example of index characterization is presented for an index-2 model describing an isothermal CSTR. The steps for numerical simulation are presented focusing on the direct method of initialization and the process identification procedure is presented as performed in industry for cases where no phenomenological information is used, only time data from input and output disturbances. The last topic explains how the control is implemented by tuning the design parameters based on process dynamics.

3.1 Methodology Flowchart

The flowchart of this dissertation methodology is presented in Figure 3.1. The first step of this work is to decide on how to control a chemical process governed by differential-algebraic equations. Among the available control theories, it was decided to use MPC strategy. The algorithm chosen was the GPC due to its flexibility to handling different types of models. The predictive model used, CARIMA, is an input-output model that has internally a representation of low frequency noise. The processes chosen are described in KUMAR and DAOUTIDIS (1999), HARNEY *et al.* (2013) and PANTELIDES (1988). The GPC, as most MPC algorithms, uses discrete control actions to control the plant, which can be a real or a virtual process. As the control algorithm applies discrete actions, as discussed in the previous chapter, this causes difficulties to the integration process. Many DAE solvers, such as DASSLC (Differential Algebraic System Solver in C) are sensitive to step changes during the integration process (VIEIRA and BISCAIA JR., 2001), which requires the application of direct initialization methods also in the control algorithm. A simple regularization function was used to enable the simulation of the model: the

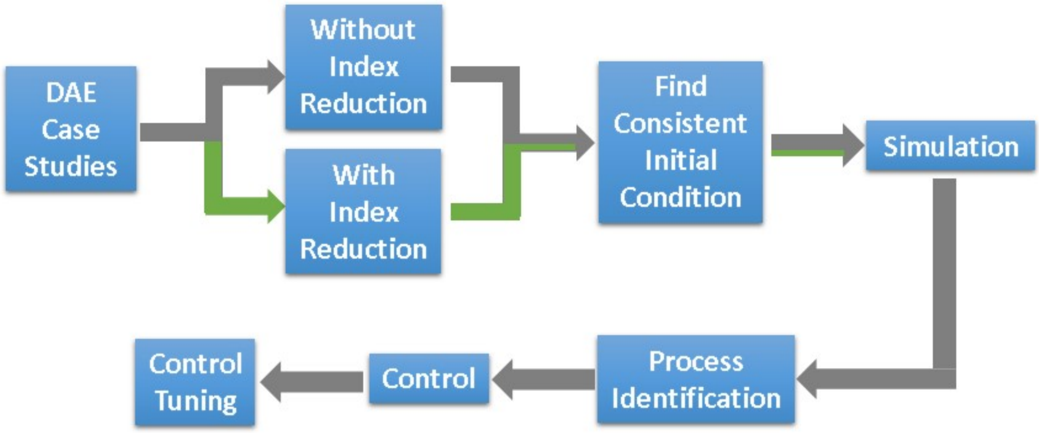


Figure 3.1: Methodology flowchart.

tangent hyperbolic function, that simulates the transition process from zero to one continuously during the integration procedure, thus converting the discrete signal into a continuous one (QUINTO, 2010).

The continuous control action is used inside the models to allow the transition from one state to another during the integration process. The direct method of initialization set the initial condition as a stationary state, by setting all derivatives to zero and finding the solution by means of the FSOLVE tool in MATLAB's$^{\mathrm{TM}}$ environment. FSOLVE was used to find the stationary point and uses the "trust-region-dogleg" optimization algorithm to solve the nonlinear equations, $F(x, x', z, 0) = 0$, with the standard parameters of each model. It is a nonlinear algebraic system solver, which is used once, to determine the condition where the simulation will start.

The next two tasks of this dissertation work are then performed: process identification and control tuning. To perform the process identification, the output data obtained from the simulation, $y(t)$, is stored together with the step changes applied and inserted in the System Identification Toolbox$^{\mathrm{TM}}$ of MATLAB$^{\mathrm{TM}}$, a toolbox that performs the process identification by means of different strategies on parameter estimation. With the time data from the simulation, the response from DAE models is represented by transfer functions and used in the GPC algorithm to control the process in the form of a CARIMA model. The last step in the dissertation process is to tune the GPC controller after analyzing the simulation, trying different parameters for tuning following some heuristics rules based on the open loop dynamic behavior of the models simulated (ROSSITER, 2003).

The index reduction is performed only for analyze the impact of reduction when simulating disturbances on the models. The identification and control were performed only for the index-2 models.

3.2 Case Studies of High-Index DAE Models

Four index-2 DAE models were chosen for simulation and control. The choice for these models was based on their chemical engineering relevance and suitability for the methodology presented here. The first case study is the Pantelides condenser, which appears in several studies regarding DAE systems. The model is an index-2 DAE and is used because its relevance and possibility of comparison with other works. The second and third case studies are systems presented by KUMAR and DAOUTIDIS (1999) for which control studies were performed and the DAE models were obtained from quasi-equilibrium assumptions which enables the comparison between the purely differential and the DAE responses. The fourth case study is a intrinsically high-index DAE model (ANDRADE NETO, 2018), presented in dimensionless form to enable better correlation between variables and parameters.

3.2.1 Case 1: Pantelides' Condenser

The first case is the condenser model presented by PANTELIDES (1988), illustrated in Figure 3.2, which can be considered as a benchmack on the DAE literature. The model is not phenomenologically consistent, as it does not cover all phenomena presented in this type of process, such as mass transfer between phases. However it

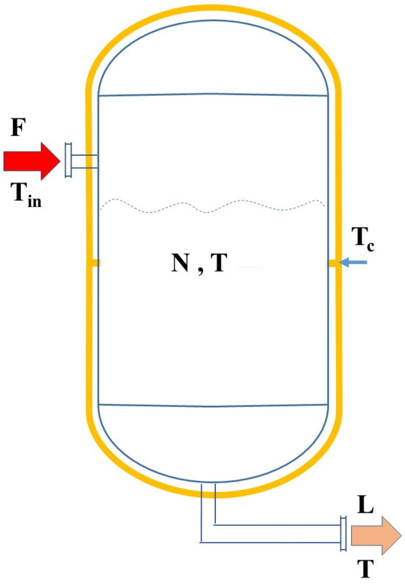


Figure 3.2: Ilustration of Pantelides condenser.

Table 3.1: Parameters used for the Pantelides' condenser simulation.

Parameter	Description (unit)	Value
T_{in}	Vapor temperature on feed (K)	373
V	Condenser volume (m^3)	1
UA	Global coefficient and heat exchange effective area (J/mol)	10^4
T_C	Cooling fluid temperature (K)	283
c_p	Heat capacity at constant pressure ($J/molK$)	33.5
ΔH	Heat of vaporization (W/m^2)	$4.5 \cdot 10^4$
R	Universal gas constant ($J/molK$)	8.314
A	Antoine constant A	$1.2 \cdot 10^{10}$
B	Antoine constant B	3816
C	Antoine constant C	-46
F	Molar feed flow rate (mol/s)	50

serves as a reference to test the approach of direct initialization and is a benchmark for the DAE literature. The index-2 model is presented in Equations 3.1a to 3.1d.

$$0 = \frac{dN}{dt} - F + L \tag{3.1a}$$

$$0 = Nc_p\frac{dT}{dt} - Fc_p\left(T_{in} - T\right) - L\Delta H - UA\left(T_c - T\right) \tag{3.1b}$$

$$0 = PV - NRT \tag{3.1c}$$

$$0 = P - Ae^{\left(-\dfrac{B}{T+C}\right)} \tag{3.1d}$$

Equation 3.1a refers to the global molar balance, with N being the total molar holdup and F and L, the inlet and outlet flow rates, respectively. Equation 3.1b is the energy balance and Equations 3.1c and 3.1d are the algebraic constraints, referring to the ideal gas assumption and the Antoine relationship for the saturation pressure. In this model the differential variables are N and T and the algebraic variables are L and P. The parameters used to simulate the Pantelides condenser are presented in Table 3.1. The chosen control configuration is for a SISO case, using F as manipulated variable and L as controlled variable.

3.2.2 Case 2: Isothermal CSTR

The second simulated model is the isothermal CSTR (Continuous Stirred Tank Reactor) where two reactions occur in series, $A \leftrightarrow B \rightarrow C$, with the first being faster than the second. The process is illustrated in Figure 3.3 and the model is presented in Equations 3.2a to 3.2e. The parameters used for the simulation are

Table 3.2: Parameters used for the isothermal CSTR simulation.

Parameter	Description (unit)	Value
C_{A0}	Initial concentration of A (mol/L)	10
V	Reactor volume (L)	10
k_1	Reaction rate $A \rightarrow B$ (min^{-1})	50
k_2	Reaction rate $B \rightarrow C$ (min^{-1})	2
K_{eq}	Equilibrium constant for the first reaction	2.25
F	Molar feed flow rate (L/min)	4

listed in Table 3.2.

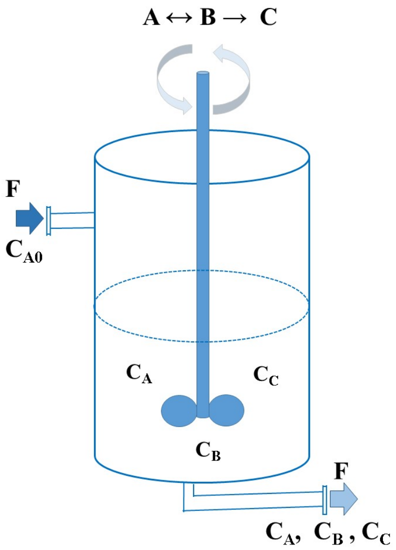


Figure 3.3: Illustration of isothermal CSTR with two reactions in series.

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A0} - C_A) - R_1 \tag{3.2a}$$

$$\frac{dC_B}{dt} = -\frac{F}{V}C_B + R_1 - R_2 \tag{3.2b}$$

$$\frac{dC_C}{dt} = -\frac{F}{V}C_C + R_2 \tag{3.2c}$$

$$0 = C_A - \frac{C_B}{K_{eq}} \tag{3.2d}$$

$$0 = R_2 - k_2 C_B \tag{3.2e}$$

The differential variables are C_A, C_B and C_C and the algebraic variables are R_1 and R_2. The algebraic constrains, Equations 3.2d and 3.2e, refer to the reaction rates, R_1 and R_2, with the assumption that the first reaction is much faster than the second. This makes the species A and B remain in equilibrium, defined by the equilibrium constant, K_{eq}. The isothermal CSTR model has a representation in ODE form by not taking into account the fast kinetics of the first reaction, what changes the model by replacing the Equation 3.2d by Equation 3.3.

$$0 = R_1 - k_1(C_A - \frac{C_B}{K_{eq}}) \tag{3.3}$$

3.2.3 Case 3: CSTR with a Heating Jacket

The third case is also a process that could be described by an explicit set of ODEs, but, due to the equilibrium assumption, the model is represented by a DAE system. The original model was proposed by KUMAR and DAOUTIDIS (1999), describing a CSTR with a heating jacket, illustrated in Figure 3.4, in which the heat transfer is so efficient that the heating jacket is assumed to be in thermal equilibrium with the reactor, generating the equilibrium constraint $T = T_j$. In the process C_A and C_B are the molar concentration of components A and B in the reactor, T is the reactor temperature and T_j is the heating jacket temperature. The ODE model can be obtained from Equations 3.4a to 3.4e, but its representation after thermal equilibrium assumption is presented by Equations 3.5a to 3.5e. The differential variables are C_A, C_B, T and T_j and the algebraic variable is Q. The fluid parameters in reactor and heating jacket are assumed to be the same, *i.e.*, $\rho_h = \rho$ and $c_{ph} = c_p$. The remaining parameters are listed in Table 3.3. This process is configured for a MIMO case, with manipulated variables F and F_h and controlled variables C_B and T.

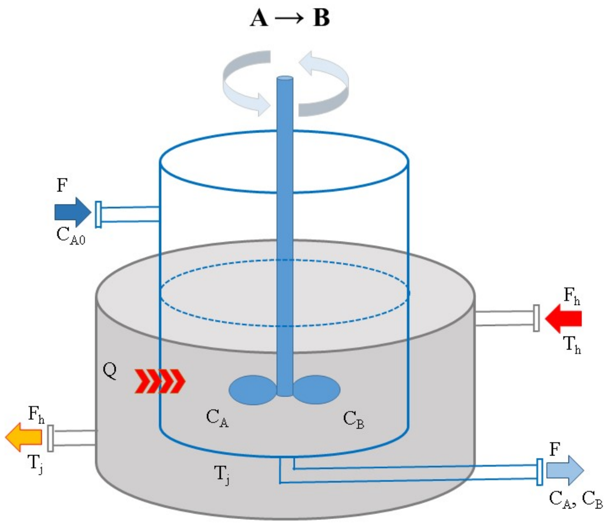


Figure 3.4: Illustration of reactor with heating jacket.

Table 3.3: Parameters used for the simulation of reactor with heating jacket.

Parameter	Description (unit)	Value
C_{A0}	Initial concentration of A (mol/L)	5
V	Reactor volume (L)	10
V_h	Heating jacket volume (L)	0.1
c_p	Heat capacity at constant pressure (J/gK)	6
E	Activation energy of Arrhenius equation $(J/molK)$	$5 \cdot 10^4$
k_0	Pre-exponential factor of Arrhenius equation $(L/molmin)$	1e9
T_A	Feed flow rate temperature (K)	300
T_h	Heating fluid temperature (K)	375
ΔH_r	Heat of reaction (J/mol)	$2 \cdot 10^4$
ρ	Molar density of liquid (g/L)	600
F	Inlet feed flow rate (L/min)	3
F_h	Heating fluid flow rate (L/min)	0.1

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A0} - C_A) - k_0 e^{\left(\frac{-E}{RT}\right)} C_A \tag{3.4a}$$

$$\frac{dC_B}{dt} = \frac{F}{V} C_B + k_0 e^{\left(\frac{-E}{RT}\right)} C_A \tag{3.4b}$$

$$\frac{dT}{dt} = \frac{F}{V}(T_A - T) - \frac{\Delta H_r}{\rho c_p} k_0 e^{\left(\frac{-E}{RT}\right)} C_A + \frac{Q}{\rho V c_p} \tag{3.4c}$$

$$\frac{dT_j}{dt} = \frac{F_h}{V_h}(T_h - T_j) - \frac{Q}{\rho_h V_h c_{ph}} \tag{3.4d}$$

$$0 = Q - UA\,(T_j - T) \tag{3.4e}$$

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A0} - C_A) - k_0 e^{\left(\frac{-E}{RT}\right)} C_A \tag{3.5a}$$

$$\frac{dC_B}{dt} = \frac{F}{V} C_B + k_0 e^{\left(\frac{-E}{RT}\right)} C_A \tag{3.5b}$$

$$\frac{dT}{dt} = \frac{F}{V}(T_A - T) - \frac{\Delta H_r}{\rho c_p} k_0 e^{\left(\frac{-E}{RT}\right)} C_A + \frac{Q}{\rho V c_p} \tag{3.5c}$$

$$\frac{dT_j}{dt} = \frac{F_h}{V_h}(T_h - T_j) - \frac{Q}{\rho_h V_h c_{ph}} \tag{3.5d}$$

$$0 = T_j - T \tag{3.5e}$$

3.2.4 Case 4: Reactive Flash Drum

The fourth DAE model describes a reactive flash drum written in dimensionless form. The model was proposed by ANDRADE NETO (2018) and is similar to the one presented by HARNEY *et al.* (2013) but, instead of four components, the model studied here accounts for only three; A, B and C. The model considers two reactions in series, $A \to B \to C$. The model here has a different energy balance from the one presented by HARNEY *et al.* (2013), as it uses the dimensionless temperature instead of the heats. The time is rewritten in dimensionless form using the mean residence time, as described in Appendix A. The number of variables is reduced to four, being three differentials, x_A, x_B and θ, and one algebraic, ϕ. The process is illustrated in Figure 3.5. In this model, components, A and B, fractions in the feed stream are given by x_{AF} and x_{BF}, respectively. The process model is given in Equation 3.6a to 3.6d and parameters used for the reactive flash drum simulation are presented in Table 3.4.

Constants K_A, K_B and K_C refer to the equilibrium of the components between the liquid and vapor phases. The variable ϕ is the flow rate fraction that leaves the vessel in vapor phase. The parameters D_{aA} and D_{aB} are the Damköhler numbers regarding the reactions, involving species A and B, respectively. The parameters B_1 and B_2 are related to the heats of reaction, γ_A and γ_B are the dimensionless activation energies for each reaction and λ is the dimensionless heat of vaporization. The term Q is the dimensionless heat added to the vessel and θ is the dimensionless temperature defined by $\theta = \dfrac{T - T_{ref}}{T_{ref}}$. Equation 3.6d is the algebraic constraint, coming from the equilibrium assumption between the components in liquid and vapor phases. The molar fractions of the vapor phase, y_i, are related to the molar fractions of the liquid phase, x_i, by the relationship $y_i = K_i x_i$.

$$\frac{dx_A}{dt} = x_{AF} - (1-\phi)\,x_A - \phi K_A x_A - D_{aA} x_A e^{\left(\frac{\gamma_A}{\theta+1}\right)} \tag{3.6a}$$

$$\frac{dx_B}{dt} = x_{BF} - (1-\phi)\,x_B - \phi K_B x_B - D_{aB} x_B e^{\left(\frac{\gamma_B}{\theta+1}\right)} + D_{aA} x_A e^{\left(\frac{\gamma_A}{\theta+1}\right)} \tag{3.6b}$$

$$\frac{d\theta}{dt} = 1 - \theta - \lambda\phi + Q + B_1 D_{aA} x_A e^{\left(\frac{\gamma_A}{\theta+1}\right)} + B_2 D_{aB} x_B e^{\left(\frac{\gamma_B}{\theta+1}\right)} \tag{3.6c}$$

$$0 = K_A x_A + K_B x_B + K_C\,(1 - x_A - x_B) - 1 \tag{3.6d}$$

The equilibrium constants, K_i, are assumed to be related to the total and saturation pressures, *i.e.* $K_i = \dfrac{P_i^{sat}}{P}$. The saturation pressure is calculated using the Antoine equation, conform Equation 3.7a to 3.7c. The total pressure, P, is assumed to

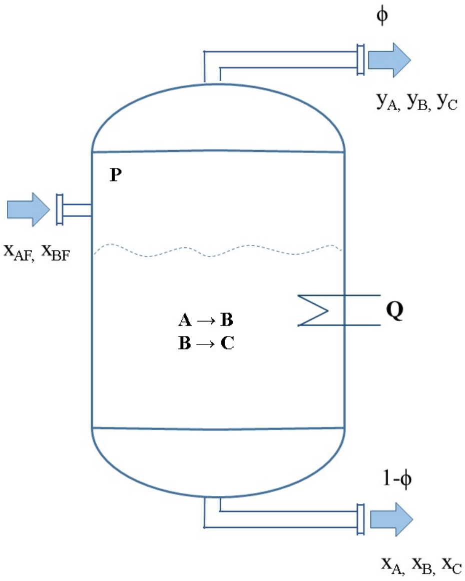


Figure 3.5: Illustration of vessel for reactive flash.

be in the vessel and was used as a manipulated variable, after testing the process response to variations in Q and D_{aA}. The process is configured as SISO, with manipulated variable P and controlled variable x_B. P was chosen as manipulated variable due to the high sensitivity of the system regarding this parameter. The model nondimensionalization procedure is presented in Apendix A.

$$P_A^{sat} = exp\left[21.3066 - \frac{2428.2}{T_{ref}(\theta+1) - 35.388}\right] \tag{3.7a}$$

$$P_B^{sat} = exp\left[25.1431 - \frac{6022.18}{T_{ref}(\theta+1) - 28.25}\right] \tag{3.7b}$$

$$P_C^{sat} = exp\left[23.8578 - \frac{6085.25}{T_{ref}(\theta+1) - 26.15}\right] \tag{3.7c}$$

3.3 Index Characterization

The DAE literature reports several index concepts (SCHULZ, 2003), but the one used to characterize the DAE systems studied here is the differentiation index, hereafter called only index. As stated in the previous chapter, the index is defined as the minimum number of times that part or all equations should be differentiated to obtain a system of explicit set of ODEs (BRENAN *et al.*, 1996). This definition

Table 3.4: Parameters used on reactive flash drum simulation.

Parameter	Description (unit)	Value
x_{AF}	Molar fraction of component A in feed stream	0.8
x_{BF}	Molar fraction of component B in feed stream	0.2
D_{aA}	Damköhler number of reaction A→B	0.01
D_{aB}	Damköhler number of reaction B→C	0.02
γ_A	Adimensional activation energy of reaction A→B	6.7821
γ_B	Adimensional activation energy of reaction B→C	3.8784
λ	Adimensional heat of vaporization	15.521
Q	Adimensional heat added	1.3430
B_1	Adimensional heat of reaction A→B	35.9778
B_2	Adimensional heat of reaction B→C	99.8517
T_{ref}	Reference temperature (K)	298
P	Total pressure (Pa)	101325

shows the intrinsic relationship between DAEs and ODEs, being the DAEs a general case of ODEs, where the last can be understood as an index zero DAE system.

The representation of DAE in this work follows the acronym DAE plus the index, for instance, DAE2, refering to an index 2 DAE system. In some cases where the DAE model is simulated together with its reduced versions, the models will be named with the same standard, changing only the index value, such as DAE2, DAE1 and DAE0 represent the system of index-2 DAE and its reduced versions to index 1 and 0, respectively. The DAE0 is the fully reduced DAE model to an explicit ODE. In cases where the DAE model is obtained from an explicit ODE from equilibrium assumptions, such as the models from KUMAR and DAOUTIDIS (1999), *i.e.* Cases 2 and 3 described previously, the original model will be identified by the ODE acronym.

To exemplify the previous paragraph, Figure 3.6 shows a simulation of Case 3, the CSTR with heating jacket, represented both as ODE and the DAE models. The DAE system is simulated as the index 2 and the reduced versions of index 1 and index 0, DAE2, DAE1 and DAE0, respectively. The model with no equilibrium assumption is designated as ODE, the index 2 DAE model is obtained by supposing thermal equilibrium between the heating jacket and the reactor.

The simulation using the reduced index models was performed to check the consistency of the models with the index reduction approach. A numerical problem reported when reducing high index DAE systems is the so called “drift off” effect, where the solution of the reduced DAE differs from the original model, as exemplified for the classical index-3 mechanical Pendulum model (SOARES and SECCHI, 2005). The “drift off” effects occur due to the index reduction method applied, where only

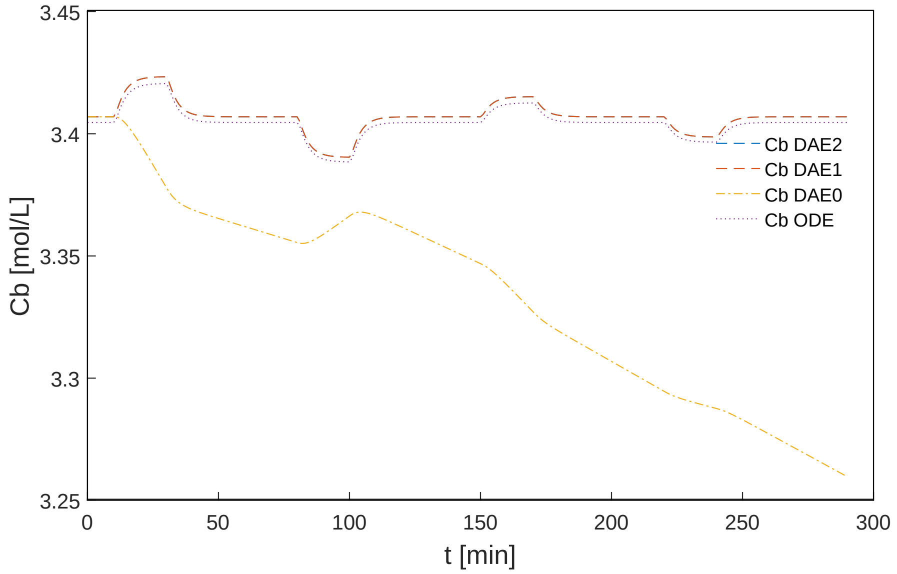


Figure 3.6: Example of an index-2 DAE model simulated together with its index reduced equivalents and the original ODE model without the assumption of thermal equilibrium.

the differentiated equations remains in the model and the solution drift off the correct one after longer integration time, as can be seen in Figure 3.7.

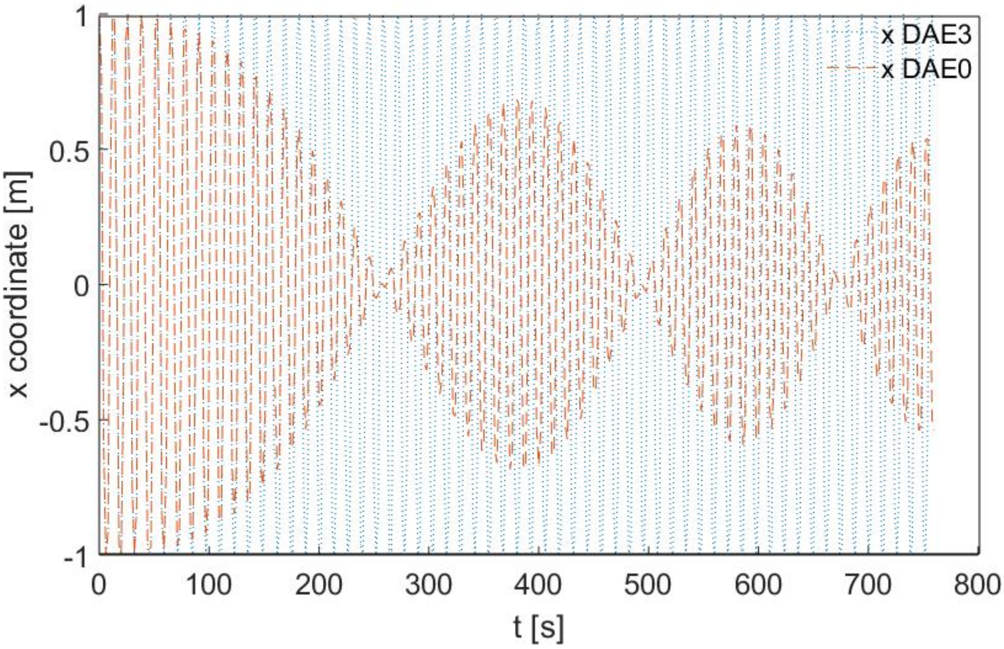


Figure 3.7: Example of "drift off" effect on solution of a reduced index 0 from the index-3 pendulum model presented in BRENAN *et al.* (1996).

As the DASSLC and other high index solvers use the index of each variable to solve high-index DAE systems, the index reduction will be examplified for case 2, the isothermal CSTR model, in which two reactions in series occur, $A \leftrightarrow B \rightarrow C$, with the first reaction considered faster. This leads to the assumption of equilibrium of the intermediate product, B, giving rise to an index-2 DAE system.

The phenomenological model consists of the component molar balances and the kinetic equations, which are usually inserted in the differential equations, as the reader can see in Equations 3.8a to 3.8e. To determine the index of each variable one needs to carry out the differentiation without algebraic substitution, in order to check the higher order derivative of each variable, as explained in more detail in (LIOEN *et al.*, 1998). In this model the differential variables are C_A, C_B and C_C and the algebraic variables are R_1 and R_2.

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A0} - C_A) - R_1 \tag{3.8a}$$

$$\frac{dC_B}{dt} = -\frac{F}{V}C_B + R_1 - R_2 \tag{3.8b}$$

$$\frac{dC_C}{dt} = -\frac{F}{V}C_C + R_2 \tag{3.8c}$$

$$0 = R_1 - k_1(C_A - \frac{C_B}{K_{eq}}) \tag{3.8d}$$

$$0 = R_2 - k_2C_B \tag{3.8e}$$

With the equilibrium assumption of the intermediate product, B, the model becomes an index-2 DAE where the algebraic variable R_1 is not present in the constraints anymore, as illustrated in Equations 3.9a to 3.9e:

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A0} - C_A) - R_1 \tag{3.9a}$$

$$\frac{dC_B}{dt} = -\frac{F}{V}C_B + R_1 - R_2 \tag{3.9b}$$

$$\frac{dC_C}{dt} = -\frac{F}{V}C_C + R_2 \tag{3.9c}$$

$$0 = C_A - \frac{C_B}{K_{eq}} \tag{3.9d}$$

$$0 = R_2 - k_2C_B \tag{3.9e}$$

To exemplify the procedure of index reduction and the determination of the index of each variable, the reduction of the index-2 DAE model in Equation 3.9 will be described by the following operations. First, the algebraic constraints, Equations 3.9d and 3.9e, are differentiated with respect to time, producing Equations 3.10a to 3.10b:

$$0 = \frac{dC_A}{dt} - K_{eq}^{-1}\frac{dC_B}{dt} \tag{3.10a}$$

$$0 = \frac{dR_2}{dt} - k_2\frac{dC_B}{dt} \tag{3.10b}$$

After substitution of Equations 3.9a and 3.9b in Equation 3.10, a new algebraic constraint, Equation 3.11a, and a new differential equation, Equation 3.11b, are obtained:

$$0 = \frac{F}{V}(C_{A0} - C_A + K_{eq}^{-1} C_B) - R_1(1 + K_{eq}^{-1}) + K_{eq}^{-1} R_2 \tag{3.11a}$$

$$0 = \frac{dR_2}{dt} - k_2(-\frac{F}{V} C_B - R_1 - R_2) \tag{3.11b}$$

The algebraic Equation 3.11a is known as a *hidden constraint*, and also needs to be satisfied at the initial condition and during the integration process. As another differentiation is needed to transform it into a differential equation, the DAE model has index 2. To have the DAE model fully reduced to an explicit set of ODEs, represented in this work like DAE0, the Equations 3.10a or 3.11a need to be differentiated once again. This may cause loss of constants and/or information, making the index reduction a hard, tedious, error risking and messy algebraic manipulation. Taking the derivative of Equation 3.11a means a second derivative of Equation 3.10a. As some DAE solvers are not suitable for implicit integration, the full substitution of derivatives would be required. As a second derivative of the first algebraic constraint is needed, the higher order of derivatives for C_A and C_B is 2, as seen in Equation 3.12:

$$0 = \frac{d^2 C_A}{dt^2} - K_{eq}^{-1} \frac{d^2 C_B}{dt^2} \tag{3.12}$$

Performing the necessary differentiation and substitutions in the Equation 3.12, assuming K_{eq}, F and V constants, an explicit differential expression is obtained for R_1, as seen in Equation 3.13. Depending on what solver is used to integrate the system, further substitution needs to be carried out, as not all solvers can deal with implicit systems.

$$\frac{dR_1}{dt} = \left(\frac{K_{eq} + 1}{K_{eq}}\right) \left[\frac{F}{V}(-\frac{dC_A}{dt} + K_{eq}^{-1} \frac{dC_B}{dt}) + K_{eq}^{-1} \frac{dR_2}{dt}\right] \tag{3.13}$$

After this procedure, the index 2 DAE model can be represented by its reduced index 1 and index 0, DAE1 and DAE0, respectively, as expressed by Equations 3.14a to

3.14e and 3.15a to 3.15e, as follows:

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A0} - C_A) - R_1 \tag{3.14a}$$

$$\frac{dC_B}{dt} = -\frac{F}{V}C_B + R_1 - R_2 \tag{3.14b}$$

$$\frac{dC_C}{dt} = -\frac{F}{V}C_C + R_2 \tag{3.14c}$$

$$0 = \frac{F}{V}(C_{A0} - C_A + K_{eq}^{-1}C_B) - R_1(1 + K_{eq}^{-1}) + K_{eq}^{-1}R_2 \tag{3.14d}$$

$$\frac{dR_2}{dt} = k_2(-\frac{F}{V}C_B - R_1 - R_2) \tag{3.14e}$$

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A0} - C_A) - R_1 \tag{3.15a}$$

$$\frac{dC_B}{dt} = -\frac{F}{V}C_B + R_1 - R_2 \tag{3.15b}$$

$$\frac{dC_C}{dt} = -\frac{F}{V}C_C + R_2 \tag{3.15c}$$

$$\frac{dR_1}{dt} = \left(\frac{K_{eq}+1}{K_{eq}}\right)\left[\frac{F}{V}(-\frac{dC_A}{dt} - K_{eq}^{-1}\frac{dC_B}{dt}) + \frac{F'}{V}\left(C_{A0} - C_A + K_{eq}^{-1} \cdot C_B\right) + K_{eq}^{-1}\frac{dR_2}{dt}\right] \tag{3.15d}$$

$$\frac{dR_2}{dt} = k_2(-\frac{F}{V}C_B + R_1 - R_2) \tag{3.15e}$$

The term F' is the derivative of manipulated variable, which is the derivative of the regularization function used in this work, as appears on Equation 3.16.

$$F' = \frac{1}{2}\left\{sech\left[\xi\left(t - t_s\right)\right]\right\}^2 \cdot F \tag{3.16}$$

The solution of the numerical integration of these models using the solver DASSLC are presented in Figure 3.8 and 3.9. The models were submitted to step changes in different directions and magnitudes of manipulated variable F. The models integration starts with the same initial condition. The models DAE2 and DAE1 present the same solution, the model ODE, which is the purely differential, behaves closely to the DAE2 and DAE1 but presenting a small shift. The fully reduced model, DAE0, presents a completely different behavior, becoming unstable. The Figure 3.9 shows a local zoom in the models' response in order to clarify the behavior of each model when perturbations on the manipulated variable is present.

The higher orders of the derivatives and the indices of each variable for the

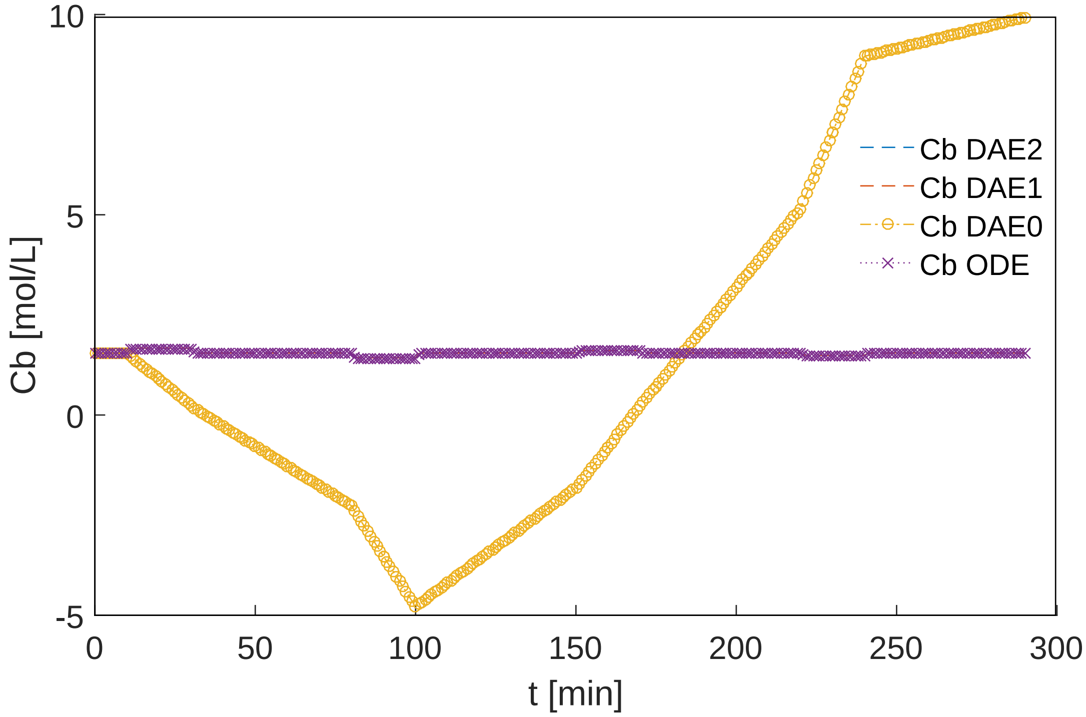


Figure 3.8: Isothermal CSTR $A \leftrightarrow B \rightarrow C$ simulated with different models with respect to the differential index.

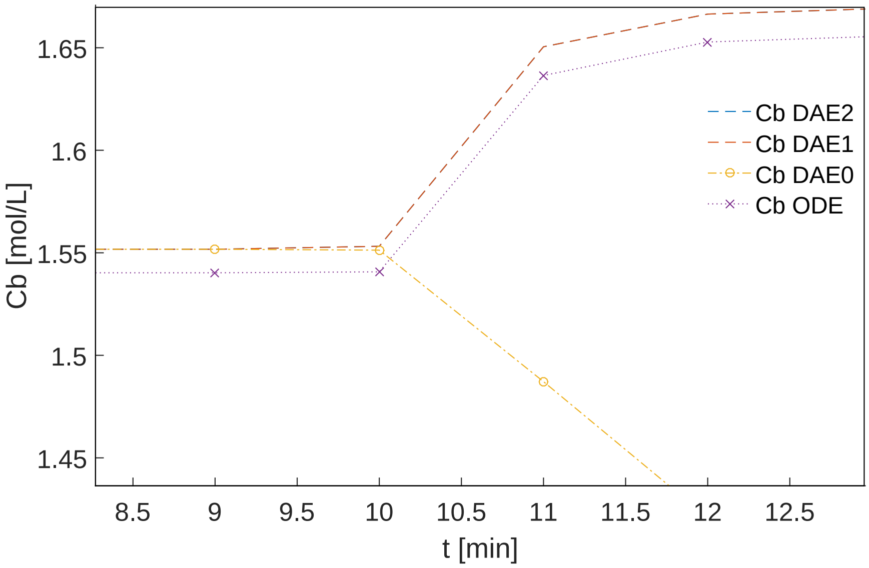


Figure 3.9: Zoom on the simulation results for the different models of isothermal CSTR.

isothermal CSTR are presented in Table 3.5. In order to perform the simulation with DASSLC, the vector of indices of the variables needs to be provided. It is formed by the index of each dependent variable, ν_i, determined by Equation 3.17 (LIOEN *et al.*, 1998), making the relationship of the differential index, ν, and the

Table 3.5: Higher order derivatives and indices of each dependent variable.

	C_A	C_B	C_C	R_1	R_2
O_i	2	2	1	1	1
ν_i	1	1	2	2	2

higher order of derivatives, O_i, found during the index reduction procedure.

$$\nu_i = 1 + \nu - O_i \tag{3.17}$$

3.4 Numerical Simulation

For DAE systems up to index-1, several integrators are available, consisting in some cases of modified ODE solvers, whereas solvers for DAE systems of index greater than 1 usually require some strategy of index reduction or the systems to be expressed in a specific structure, such as the semi-implicit or Hessenberg form (VIEIRA and BISCAIA JR., 2001). MATLAB$^{\text{TM}}$ environment itself has only one embedded integration routine capable of integrating implicit DAE systems, ODE15i, applicable only for DAEs of index up to 1 (MATHWORKS, 2018a). The software MATHEMATICA$^{\text{TM}}$ also has an embedded solver for ODEs and DAEs, but is limited to index-1 too, when high index systems are presented an index reduction needs to be defined and applied to the system (WOLFRAM, 2018). The brazilian software for process simulation, EMSO (Environment for Modeling, Simulation, and Optimization) is an equation-oriented process simulator that has high-index DAE solvers for index up to 3 (PSIDE and MEDBF) and higher (DASSLC).

One situation in which DAEs may appear is when dealing with system of ODEs with time scales so apart from each other that some equations become algebraic relationships. This is sometimes referred as quasi-equilibrium assumptions, a powerful approach to solve complex systems of ODEs, by converting them into algebraic systems. Researchers when facing stiff systems may use these equilibrium assumptions to solve their equations, such as in KUMAR and DAOUTIDIS (1997). The stiffness of a system was once defined as a measurement of the diversity of time scales, meaning that the same system possesses very fast and very low process simultaneously (SHAFIE, 2013). Another way to understand stiffness is the magnitude difference between the module of the real part of the largest and smallest characteristic values of the Jacobian matrix of the system (DA SILVA, 2013); this understanding implies that high-index DAE systems have infinit stiffness, as they have at least one characteristic value equal to zero. One practical way for understanding stiffness related to numerical solutions of ODEs and DAEs is that the step size required by the integration method to keep it stable is smaller than the step required for describing the solution. The concept of stiffness is important for DAEs solution as the methods applied for those systems are usually adequate to solve stiff systems (GONÇALVES *et al.*, 2007; MATHWORKS, 2018a; WOLFRAM, 2018).

Only a few solvers are able to simulate high index DAE systems in the fully implicit form, as examples of those are the PSIDE (Parallel Software for Implicit Differential Equation) (LIOEN *et al.*, 1998) and MEBDFI (Modified Extended Backward Differentiation Formula Implicit) (ABDULLA and CASH) that deal with implicit systems of index up to 3. The DASSLC routine is able to solve implicit systems of any index, when provided with the vector of differential index of the variables (SECCHI, 2007). The first research on DAEs performed at PEQ/COPPE/UFRJ (Chemical Engineering Program of COPPE/UFRJ) is due to SECCHI (1992), who developed the solver DAWRS (Differential-Algebraic Waveform Relaxation Solver) for solution of DAE systems by the waveform relaxation method using parallel computation.

DASSLC version 3.8 was chosen for this work due to its capability to integrate fully implicit systems of any index, as discussed above, having as outputs both the values of the variables and their derivatives. It allows the obtaining of consistent initial conditions to both the variables and the derivatives, eliminating the need of the integrator itself to use an internal routine to determine the derivatives at the initial point (SECCHI, 2007). The direct method of initialization of DAE systems is applied due to the complexity involving the rigorous methods, especially for high-index DAEs. The rigorous methods usually apply the index reduction and/or structural analysis to DAE systems in order to determine the dynamic degrees of freedom and the hidden constraints.

A stationary condition is found to be a consistent initial condition for any index system, but may not work for all situations (KRÖNER *et al.*, 1997). In the present work, the stationary condition is used as initial condition by setting all derivatives to zero and using the nonlinear algebraic system solver, FSOLVE, of MATLAB$^{\text{TM}}$, to solve the non-linear system. VIEIRA and BISCAIA JR. (2001) use the assumption that a dynamic process can be interpreted as a perturbation applied to steady state conditions. This assumption is useful to overcome the reinitialization difficulty, which is a major drawback when working with high index DAE systems. The reinitialization step is done by converting the discontinuous step changes into continuous input through regularization functions that remove the discontinuities from sudden changes, making smooth transitions, as discussed in the previous chapter. There are several proposed functions in literature, but only one was used for identification and control. The direct method applied in this work consists on starting the simulation from a stationary point found initially and then making all transitions using the sigmoid function, represented in Figure 3.10 for different values of ξ. The regularization parameter used here is $\xi = 50$ due to results on simulating the models for different values of this parameter.

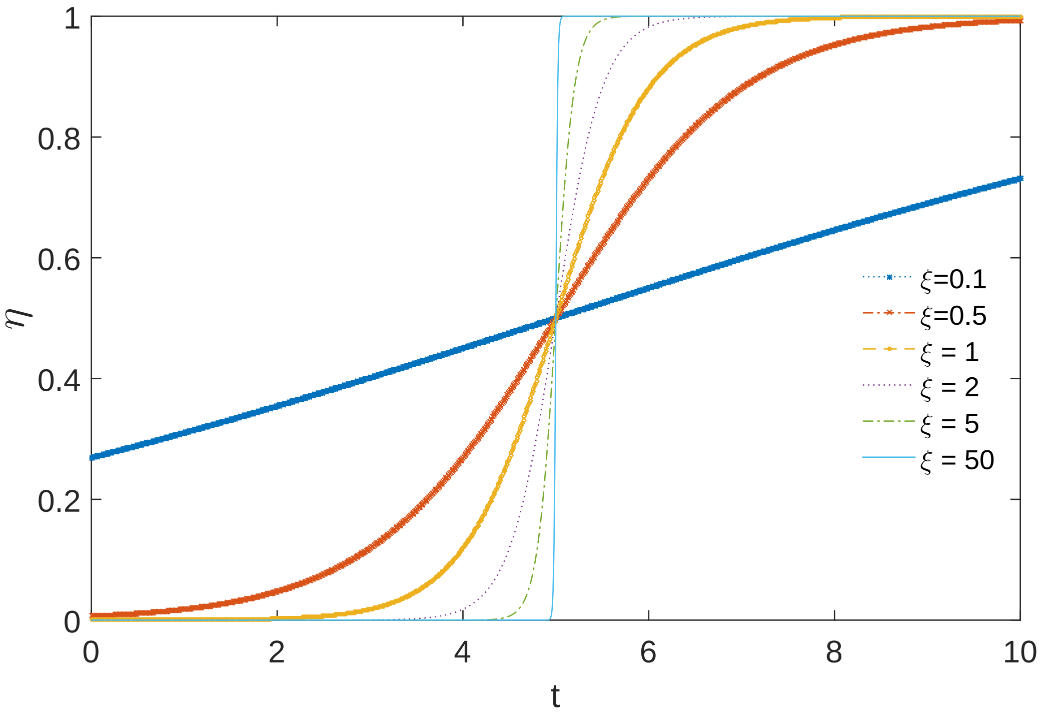


Figure 3.10: Regularization function simulated with different values of ξ.

3.5 Process Identification Procedure

The identification procedure performed to represent the DAE models as transfer functions is based on what is called “black box” identification. The DAE models were assumed to represent real processes and were used as virtual plant. The “black box" procedure treats the process as a black box in which no phenomenological information about the internal process is available, but only input and output data. This procedure is used where no information is available about the process or the process description is too complex to be simulated or would be too time consuming. In the present work the identification is performed by applying step changes on manipulated variables and collecting the response data in the output of the process, as illustrated in Figure 3.11.

Defining the step changes that will be applied to the process, the signal is passed internally to the models to make the transition between the two states. The identi-

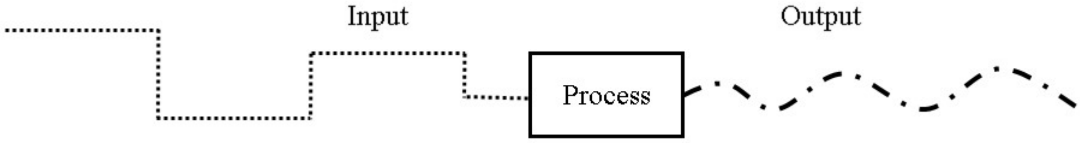


Figure 3.11: Illustration of black box identification process.

fication process was performed using MATLAB$^{\text{TM}}$ and the models were integrated with DASSLC. In order to compute the responses to the disturbances, the simulation starts from a stationary condition and the regularization function is used to make the transition from one state to another. The regularization function is placed inside the model to make the transition as smooth as possible.

With the consistent initial condition taken as a steady state, the DASSLC solution for the process with no disturbance is a straight line, as shown in Figure 3.12

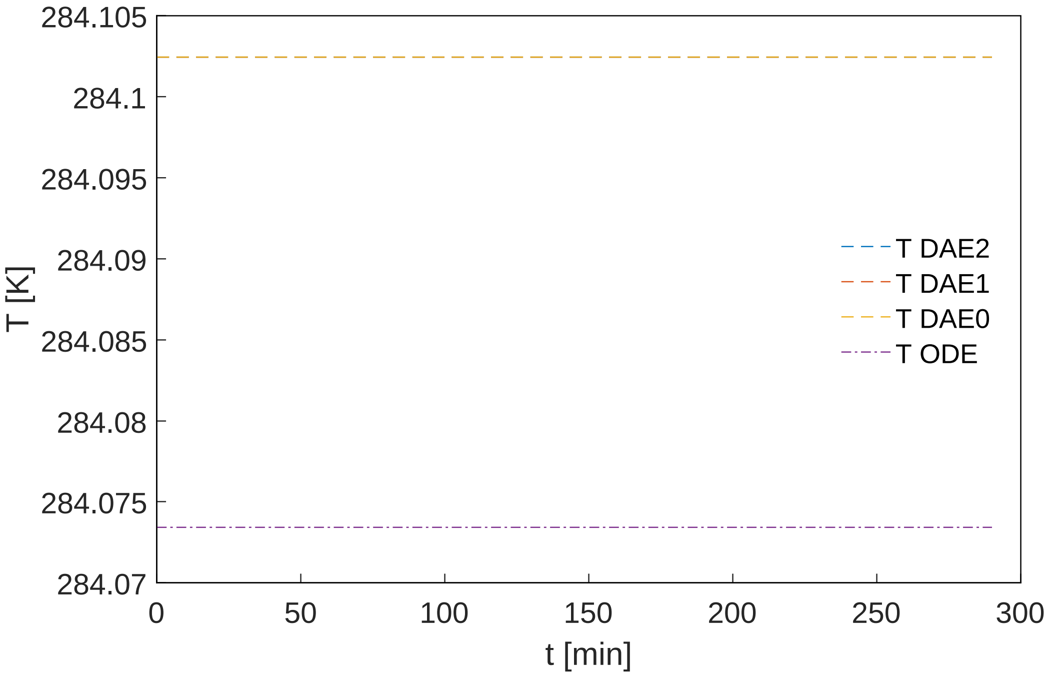


Figure 3.12: Simulation of a stationary condition for all models of Case 3 with no disturbances, solution for T.

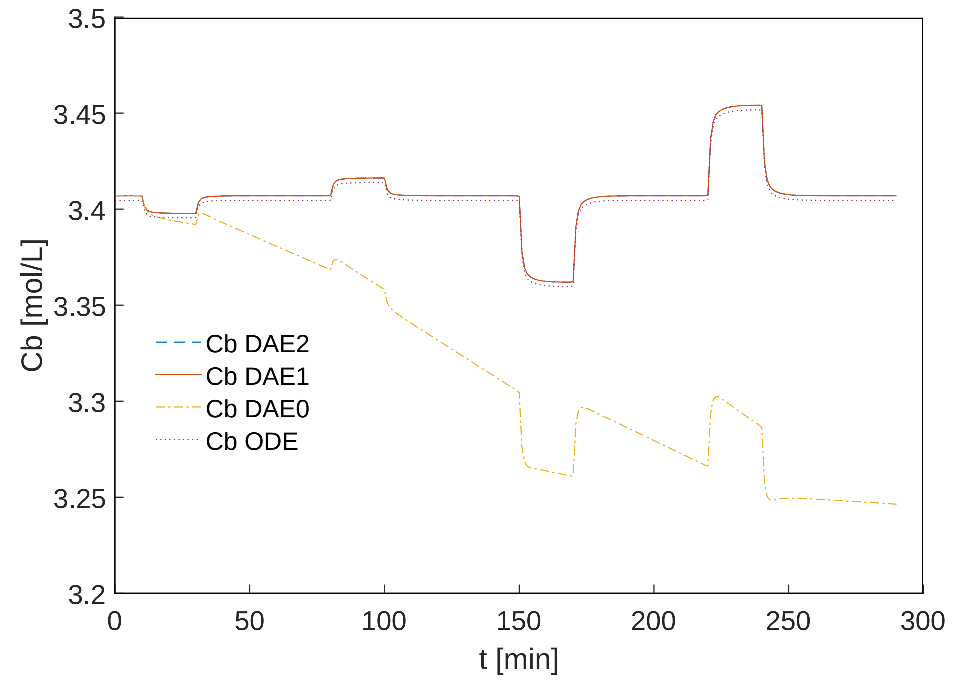
In Cases 1, 2 and 3, the reduced forms of index-1 present the same solution of index-2 as will be shown in Chapter 4. The original model of Case 2 and 3, which are purely differential, without the equilibrium assumptions differs slightly from the DAEs solution of index 1 and 2. With the stationary condition stored, the simulation of the step changes can be carried out for identification. The algorithm simulates up to four pulses, behaving as a sequence of eight steps. The simulations of the DAE models with step changes of different directions and magnitudes were performed to evaluate process non-linearities. The steps were simulated for changes of ±1, ±5 and ±10% in the manipulated variables. The inputs and the outputs were transformed in deviation variables, by subtracting the steady state value from the respective variable. A simulation of Case 3 for steps of ±1 and ±5% is presented in Figure 3.13, for the sake of illustration.

After applying the steps, the deviation variables are transferred to System Identification Toolbox$^{\mathrm{TM}}$ of MATLAB$^{\mathrm{TM}}$ where parameter estimation will be performed to obtain transfer functions from the responses. The system requires the user to define what type of transfer function will be estimated, by providing the number of poles and zeros. The estimation process was performed for first or second order transfer functions, as it is a common practice in process identification to postulate transfer functions with few parameters as first order plus dead time, FODT, or second order plus dead time, SODT, as candidate models. The MATLAB$^{\mathrm{TM}}$ toolbox gives the results of the identified model. In the following, the simulation of the transfer functions is performed and compared with the with DASSLC results.

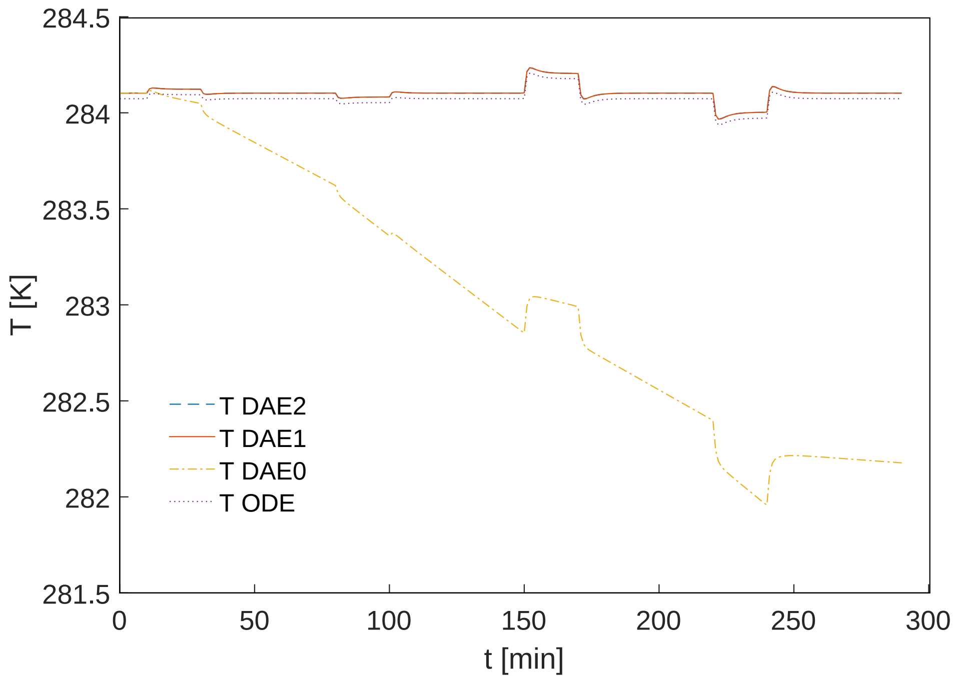
While performing the identification, information regarding control design can be obtained,*e.g.* response time. The prediction horizon is an important designing parameter and is closely related to the dynamics behavior of the process. The prediction horizon was chosen to be long enough to encompass the entire dynamics of the identified processes, which is the number of sampling times required to cover the whole transient process.

3.6 GPC Tuning

MPC tuning is an active research area and is still most of the times based on heuristic rules regarding process dynamics and robustness (ROSSITER, 2003; WASCHL *et al.*, 2011). The tuning procedure was performed here by trial and error. The prediction horizon was set sufficiently high to cover the process dynamics. The sampling time was defined to be around 1/10 of total dynamic response time. The control horizon was set to low values, keeping all the other parameters fixed, the control horizon was tested ranging from 2 to 5 of sampling times, as the prediction horizons did not exceed 10 sampling times. The models were simulated as described by the case studies in Section 3.2, with no scaling necessary on manipulated or controlled variables. The weight matrices were kept as identities during the horizons tuning, the weights were changed to do the final adjustments, when no improvement was seen from the horizons changing. The GPC used is supplied with the identified transfer functions in continuous time and performs the discretization according to the sampling time defined for the process. The algorithm used is based on the origi-



(a) Case 3 solution for C_B.



(b) Case 3 solution for T.

Figure 3.13: Solution for C_B and T when applying steps of ±1 and ±5% in F for Case 3.

nal GPC, but with constraints implemented inside the optimization stage. The GPC MATLAB$^{\text{TM}}$ code was provided by GIRALDO (2018) and adapted with DASSLC to simulate the DAE models as the virtual plant, as one may see in Figure 3.14.

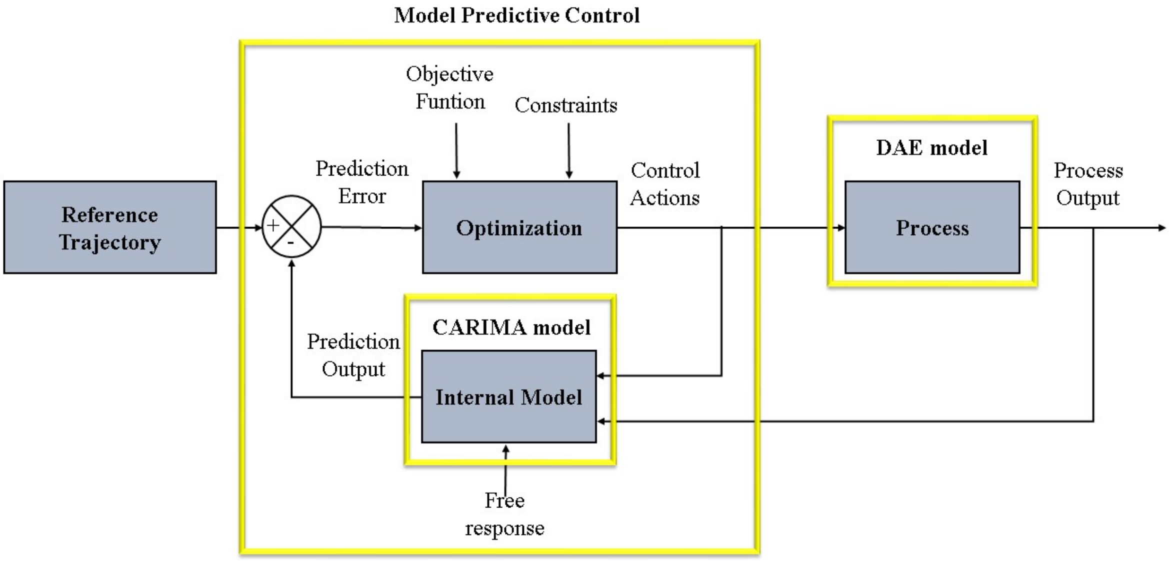


Figure 3.14: Illustration of GPC scheme used with CARIMA model and DAEs as virtual plant.

3.7 Remarks on the Methodology

This chapter detailed the steps of the methodology proposed in this dissertation, which aims performing the control of models represented by high-index DAE systems. In order to study the control characteristics in a particular process, the simulation gives important insights about what mostly affects the process and what are the best tuning parameters for the controller. Even in processes where the physical phenomena are well known but traditional phenomenological modeling becomes complex or very computationally demanding, black box identified linear models are a well suitable alternative to characterize the input/output relationships. By studying the behavior of processes as they are described by the modeling engineers, one may have better results and improve the relationship between the modeling and control research. The most important aspect of this dissertation is its generic characteristic regarding what type of mathematical model is simulated, as if considering a generic implicit high-index DAE model, a whole family of other models can be simulated using the same approach. Another generic aspect is the GPC algorithm, which enables the control engineer to handle several different processes described by transfer functions, what differs from another common MPC algorithm used, the DMC, which requires the process to be described by a step response matrix, called dynamic matrix. The main contribution of this work is in the identification process and control using the DAE models without index reduction. By using the high-index DAE models as they are originally presented by the whole set of equations, without using any index reduction technique, this work differs from the others by applying the simulation technique to the MPC control strategy and solving the DAE models as they are produced. The process identification converting high-index DAEs to transfer functions gives only a localized representation of the DAE models, but seems to be sufficient if the control is defined to operate close to the stationary point.

Chapter 4

Results and Discussion

This chapter compiles the results obtained during the identification and control of the high-index DAE systems studied in this work. The results are presented in three different categories: the impact of the regularization function when using the direct method of initialization; those obtained during the process identification stage; and those regarding the process control.

4.1 Selection of the Regularization Parameter

First, as pointed by VIEIRA (1998), the regularization function used can modify the model behavior depending on its formulation. To evaluate the impact of the regularization parameter, ξ, simulation with different values of this parameter were carried out for the index-2 DAE models of the case studies presented in Chapter 3. To exemplify, the results of Case 2 are presented in Figure 4.1, where no difference on the behavior of the process can be seen for ξ bigger than 1. For values of ξ below 1, the regularization parameter interferes on the dynamic solution of the DAE models, so this can be used by the researcher to describe the dynamics of the actuator, that can be, for example, a valve that will affect the process. Simulations for the other case studies are presented in Appendix B.

Based on the results for different ξ, the value of $\xi = 50$ was found to be close enough to an ideal step, with no interference on the DAE solution, and the identification stage of this work was performed with this value. The results point out that the shape of the regularization function interfere with the simulation results, being important for the control engineer to check what is the dynamics of the equipment used as input, in order to simulate as close as possible to the real situation.

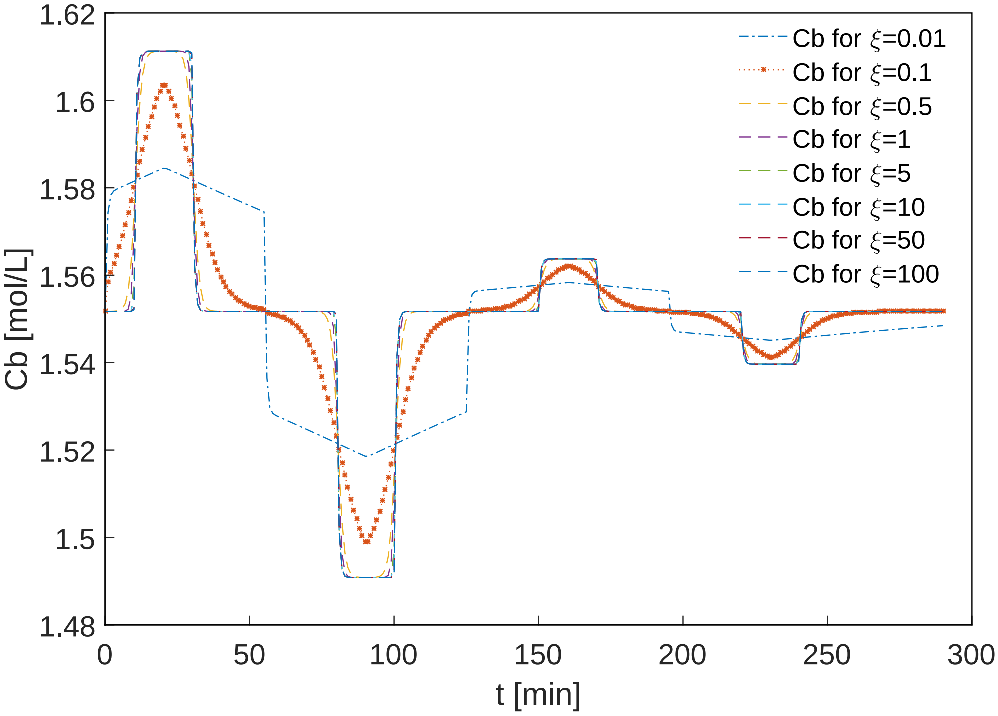


Figure 4.1: Simulation model DAE2 of Case 2 with different values of ξ.

4.2 Process Identification Results

As discussed previously, the identification of the processes studied here was performed by applying step changes with the aid of the regularization function to obtain the processes responses. The steps are grouped in two sets of 4 steps. The first set is performed with steps of ±1 and ±5% and the second set with steps of ±5 and ±10%. Another set of steps, of magnitude of +1, +5, +10 and +20% is applied to check the consistency of the transfer functions identified compared to the DASSLC integrated responses.

4.2.1 Identification for Case 1 - Pantelides Condenser

The first case study of this dissertation, the Pantelides condenser, was simulated with different step magnitudes, first with steps of ±1 and ±5%. The data obtained from this simulation was introduced into the System Identification Toolbox$^{\text{TM}}$ and the parameters of the transfer function were estimated. The simulation results for the output C_B using the original index-2 DAE model and its reduced form to index-1 and index-0 are presented in Figure 4.2. The simulation shows the same response for both models DAE2 and DAE1 systematically. For this example no difference was observed when the index reduction is applied once but the second reduction, to represent the DAE0 model could not be simulated, this can be understood as a difficulty of applying index reduction techniques.

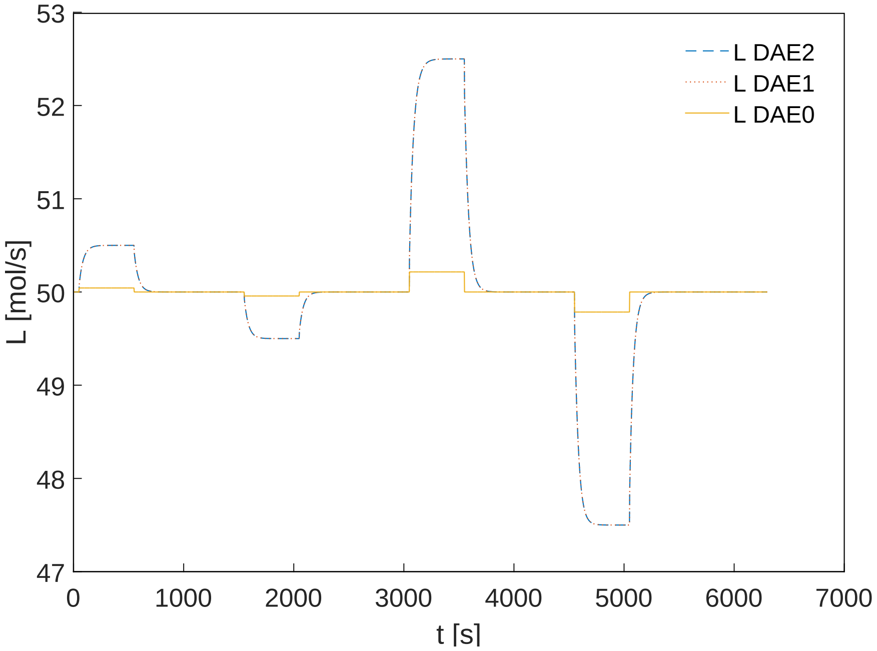


Figure 4.2: Simulation results of Case 1 for step of ±1 and ±5% on F.

With the input and output data into the System Identification Toolbox$^{\text{TM}}$, the first order transfer function described by Equation 4.1 was obtained from time data with the DAE2 model with 97.37% of fit to estimation data and MSE (Mean Square Error) of $5.687 \cdot 10^{-4}$:

$$G(s) = \frac{0.9939}{34.09s + 1} \tag{4.1}$$

Different steps were simulated to check if the transfer function response follows the model. Figure 4.3 show the index-2 DAE and the identified transfer function responses to different step changes. In the figures the DASSLC response is plotted with the transfer function response for the same inputs. As can be seen, even in case of extrapolation, *i.e.* input of 20%, the transfer function represents very well the DAE model.

4.2.2 Identification for Case 2 - Isothermal CSTR

The second case study is simulated with four different models. The rate based model consisting purely of ODEs and the DAE models: the index-2 obtained from the equilibrium assumption and its reduced forms to index 1 and 0. Figure 4.4 shows the simulation results for step changes of ±1 and ±5% on input F. This case study has all models available and integrated by DASSLC. At the beginning of the simulation, the ODE model differs slightly from DAE models, including the

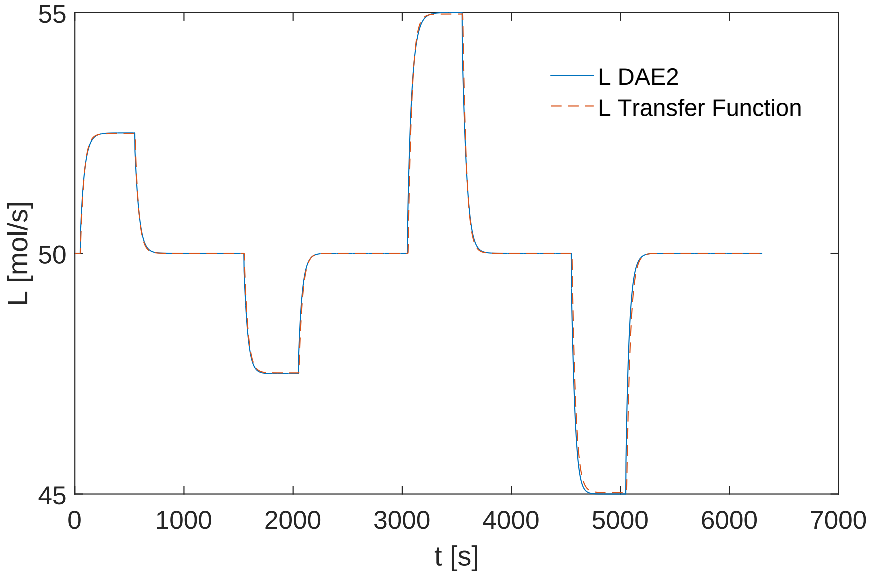


Figure 4.3: Comparison between transfer function and DAE response of Case 1 for steps of ±5 and ±10% on F.

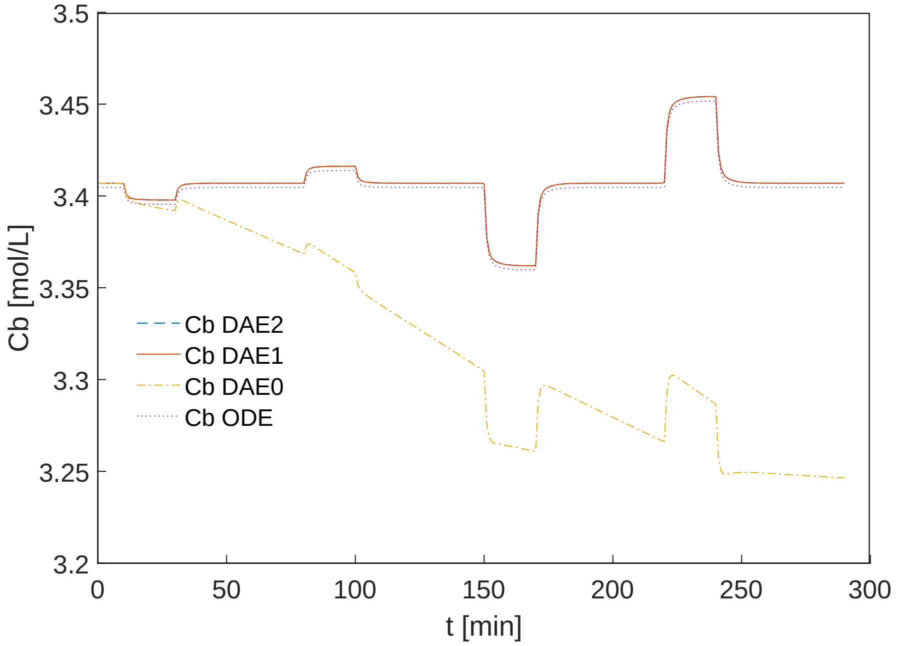


Figure 4.4: Simulation results of Case 2 for step of ±1 and ±5% on F.

reduced ones, except for index-0. The DAE models, DAE2, DAE1 and DAE0 present the same response when simulated from the same condition, as can be seen in the beginning of the simulation in Figure 4.4. The little gap between the results of DAE and ODE is expected as they are different models, as the equilibrium assumption introduces some deviation. But when the steps are applied, the index-0 DAE behaves completely different from the other models, DAE or not. The index-2 and index-1 DAE models present the same response to the step changes. The first order transfer function identified with fit of 89.11% and MSE of $6.333 \cdot 10^{-6}$ with the simulated time data is described by Equation 4.2. The comparative results between DASSLC and the transfer function are presented in Figures 4.5 and 4.6. The comparison between DAE2 and transfer function responses shows a high accuracy of the identified model, even in case of extrapolation which is shown in Figure 4.6, where steps of higher magnitude, 10 and 20%, also presented similar responses.

$$G(s) = \frac{0.3017}{0.0358s + 1} \tag{4.2}$$

The difference between the transfer function and the DASSLC results are in part due to the localy linearized aspect and the discretization of transfer function performed by MATLAB$^{\text{TM}}$ to compare the two simulations, but there is a difference in the gain, as showed in Figure 4.7. This is expected due to linearization forced by identification in a LTI (Linear Time Invariant) form, which is only a localized representation. Differently from Case 1, when steps of magnitude larger than the ones used on process identification were applied, the difference in the models appeared slightly in the gain of each model, which is the situation of steps of 10 and 20% in Figures 4.6 and 4.7. The cumulative shift to the right is due to numerical simulation used.

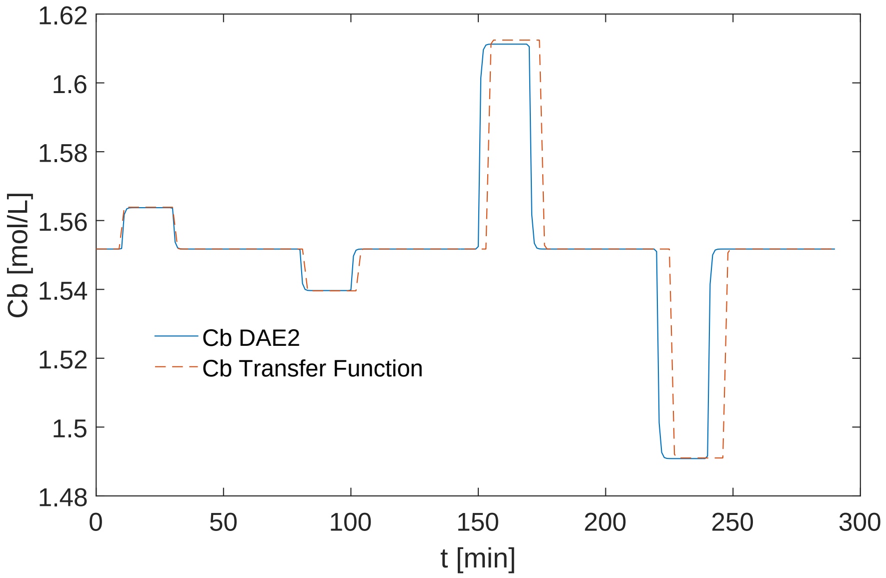


Figure 4.5: Comparison between transfer function and DAE response of Case 2 for steps of ±1 and ±5% on F.

4.2.3 Identification for Case 3 - CSTR with a Heating Jacket

The third case also could be simulated with all models, the purely differential and the DAE-based. The models present the same characteristic of Case 2, with the ODE model behaving similar to the DAEs, except for the little gap introduced

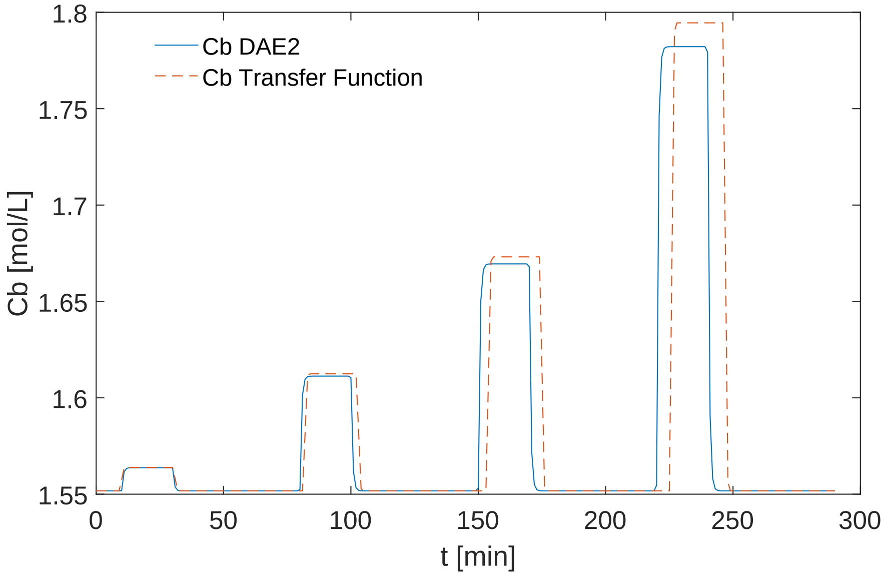


Figure 4.6: Comparison between transfer function and DAE response of Case 2 for steps of +1, +5, +10 and +20% on F.

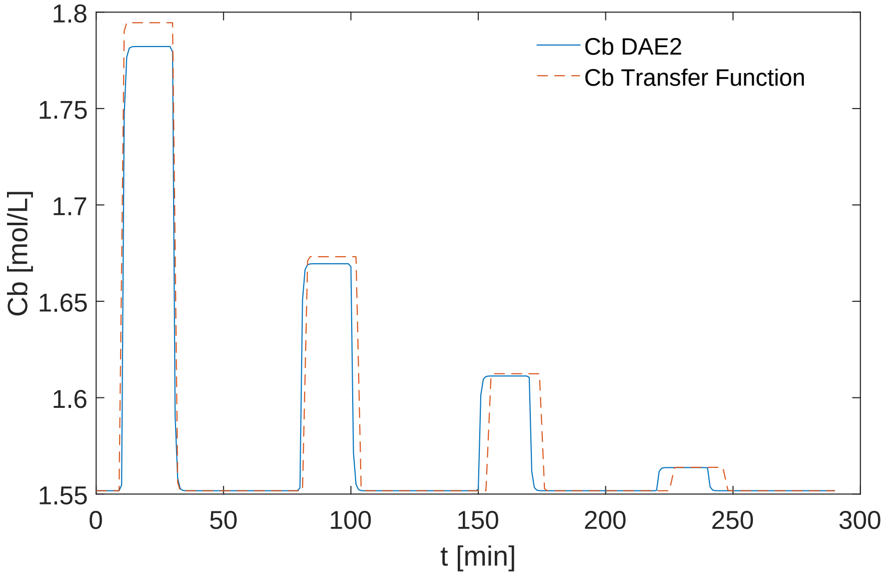
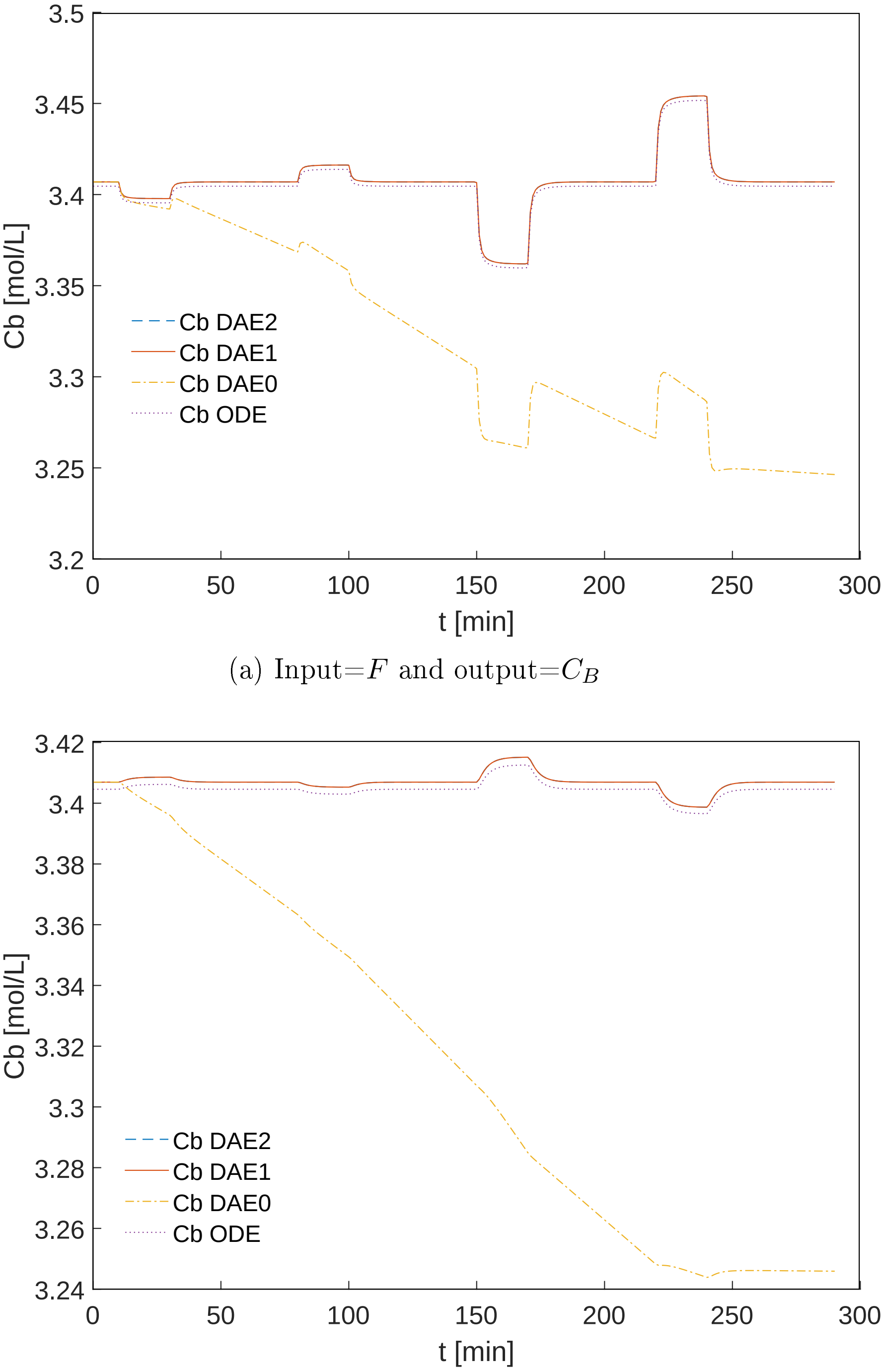


Figure 4.7: Comparison between transfer function and DAE response of Case 2 for steps in reverse order of Figure 4.6.

by the quasi-equilibrium assumption. The same behavior is presented when the simulation starts from stationary condition, all the DAE models behave the same and the ODE model presents a small shift. But the fully reduced index-0 DAE presents an unstable behavior when the inputs are applied. As this model was presented for MIMO control, the inputs F and F_h and outputs C_B and T simulated for identification are presented in Figures 4.8 and 4.9.

The identified transfer functions are presented in Equation 4.3. The comparative

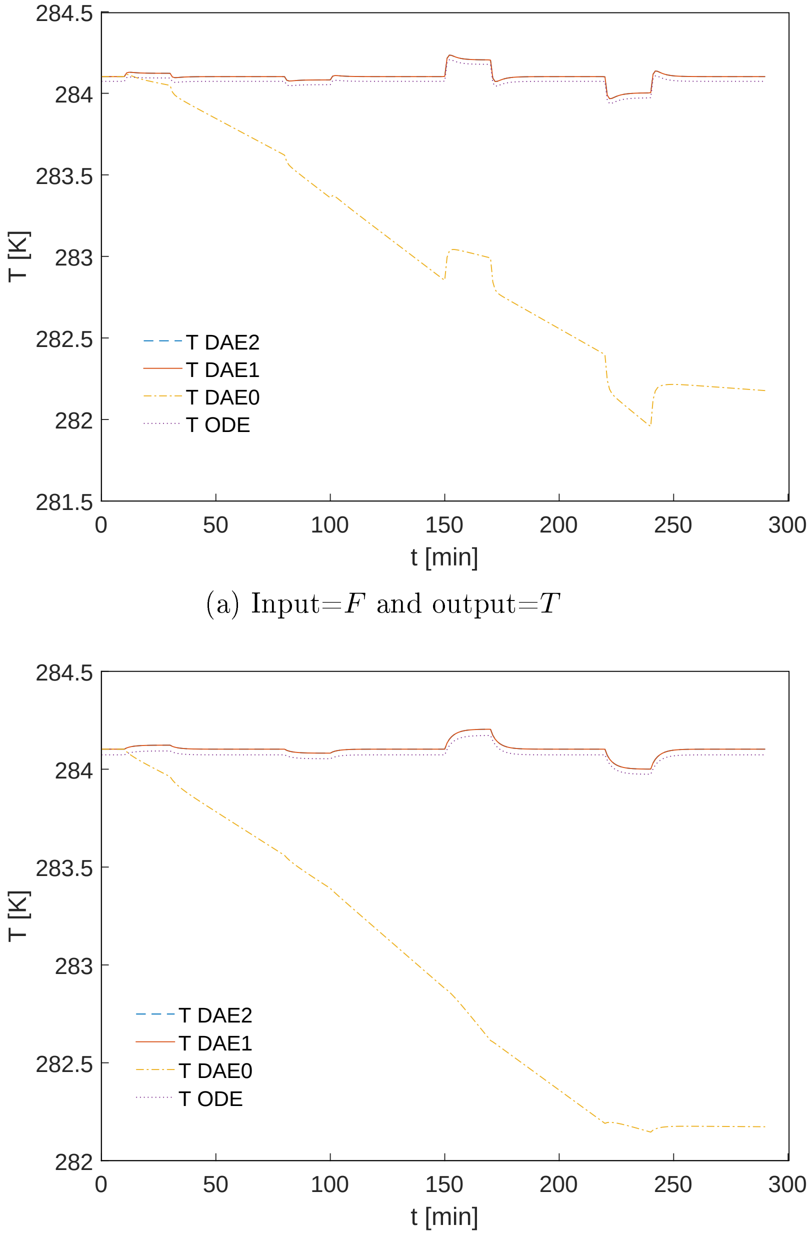


(a) Input=F and output=C_B

(b) Input=F_h and output=C_B

Figure 4.8: Simulation results of Case 3 for output C_B and steps of ± 1 and $\pm 5\%$ on F and F_h.

simulation between the transfer functions and DASSLC responses is presented in Figures 4.10 and 4.11. A comparison for steps varying from 20% to 1% is also presented in Figures 4.12 and 4.13. The System Identification Toolbox$^{\mathrm{TM}}$ reports the statistical information for each identified model. All the identified transfer functions are first order ones, except $G(2,1)$ that is a second order with a negative zero. The



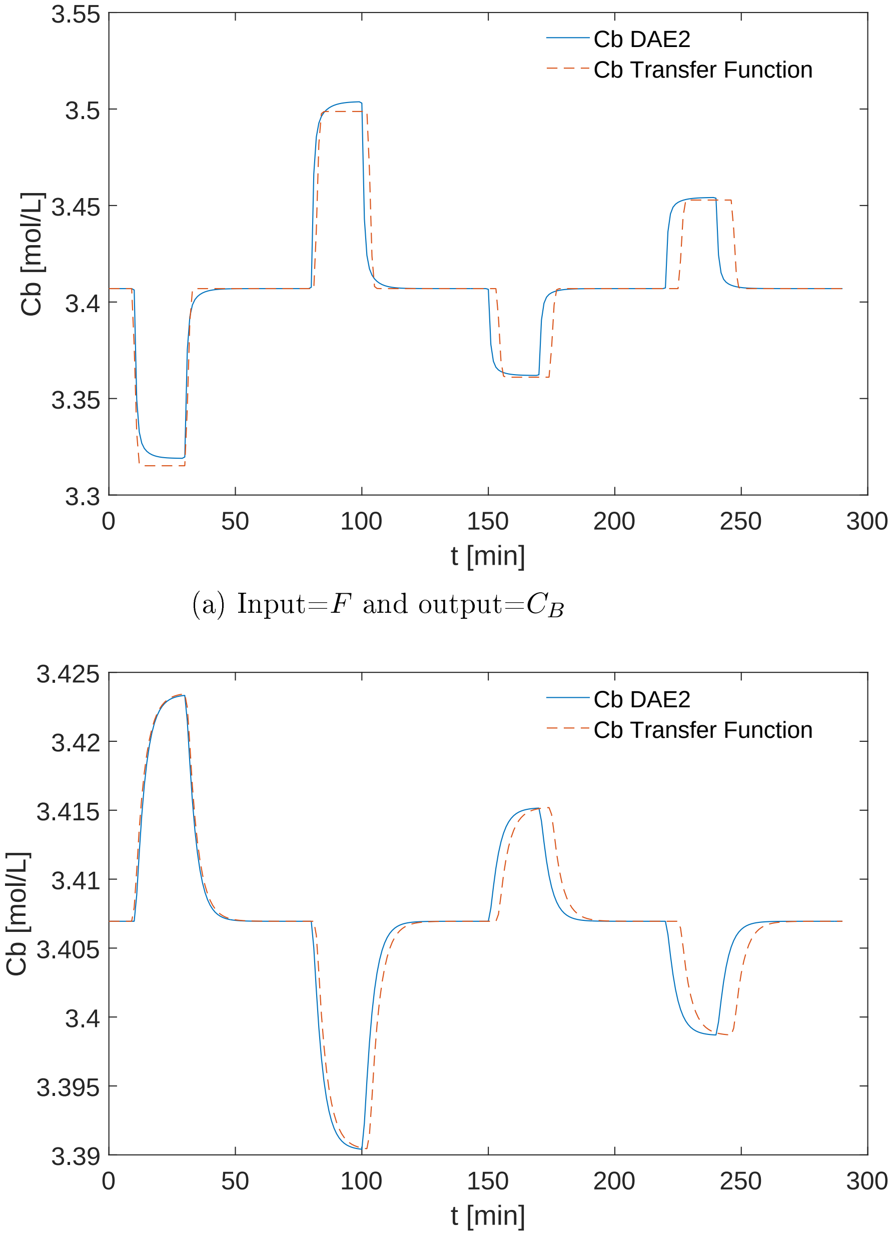
(a) Input=F and output=T

(b) Input=F_h and output=T

Figure 4.9: Simulation results of Case 3 for output T and steps of ± 5 and $\pm 1\%$ on F and F_h.

fit of $G(1,1)$ was 91.65% and MSE of $2.094 \cdot 10^{-6}$, for $G(1,2)$ the fit was 98.67% and MSE of $1.434 \cdot 10^{-9}$, for $G(2,1)$ the fit was 88.83% and MSE of $2.297 \cdot 10^{-5}$ and for $G(2,2)$ the fit was 93.94% and MSE of $4.657 \cdot 10^{-6}$.

$$G(s) = \begin{bmatrix} \dfrac{-0.2905}{0.2468s + 1} & \dfrac{1.6707}{3.802s + 1} \\[2ex] \dfrac{26.97s + 5.797}{s^2 + 22.36s + 8.505} & \dfrac{19.9418}{2.5304s + 1} \end{bmatrix} \tag{4.3}$$



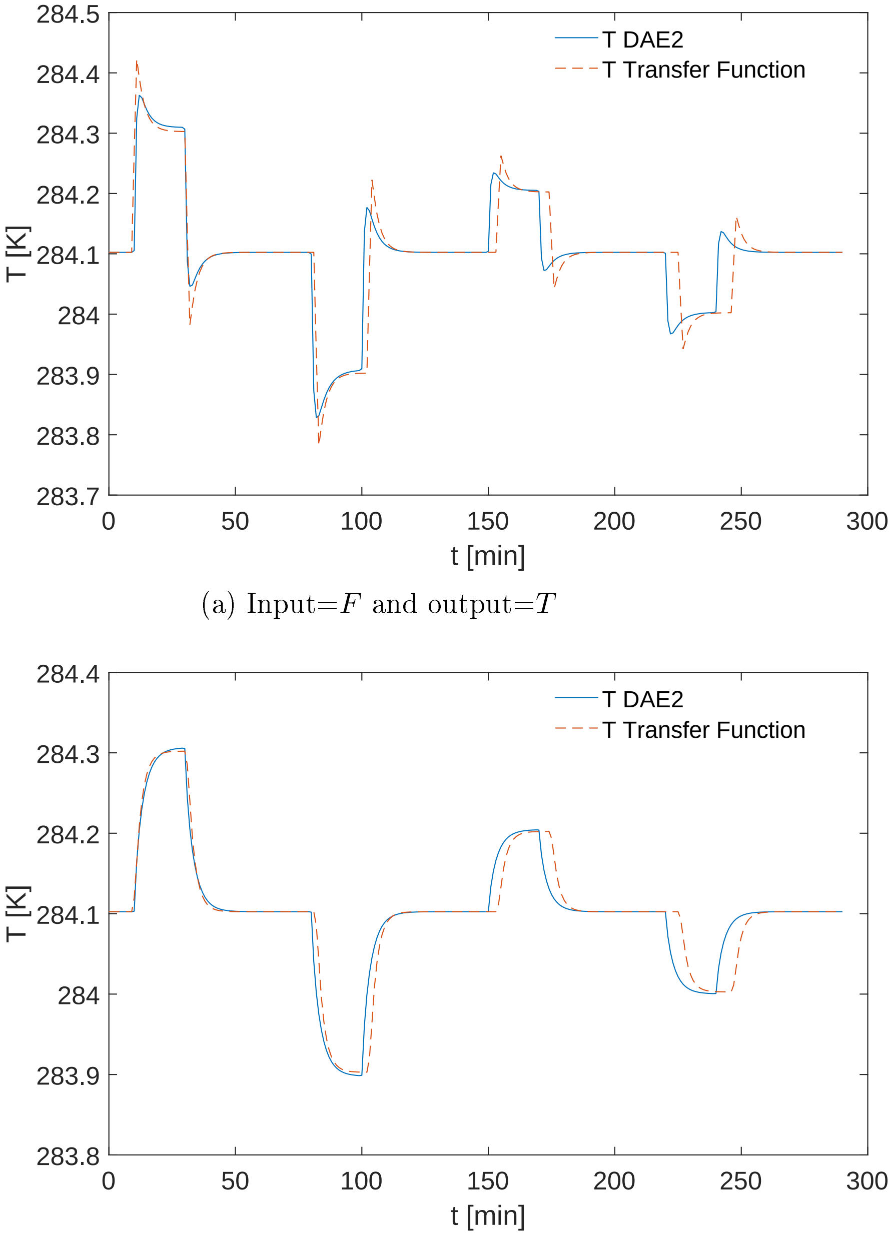
(a) Input=F and output=C_B

(b) Input=F_h and output=C_B

Figure 4.10: Comparison between transfer function and DAE response of Case 3 for output C_B and steps of ±10 and ±5% on F and F_h.

4.2.4 Identification for Case 4 - Reactive Flash Drum

The fourth case analyzed here is the reactive flash drum model, with the assumption that the reference temperature, T_{ref}, is equal to the feed temperature, T_F. The model is a dimensionless representation of a reactive flash process, in which the vapor phase fraction, ϕ, is the only flow explicitly defined in the model. The liquid

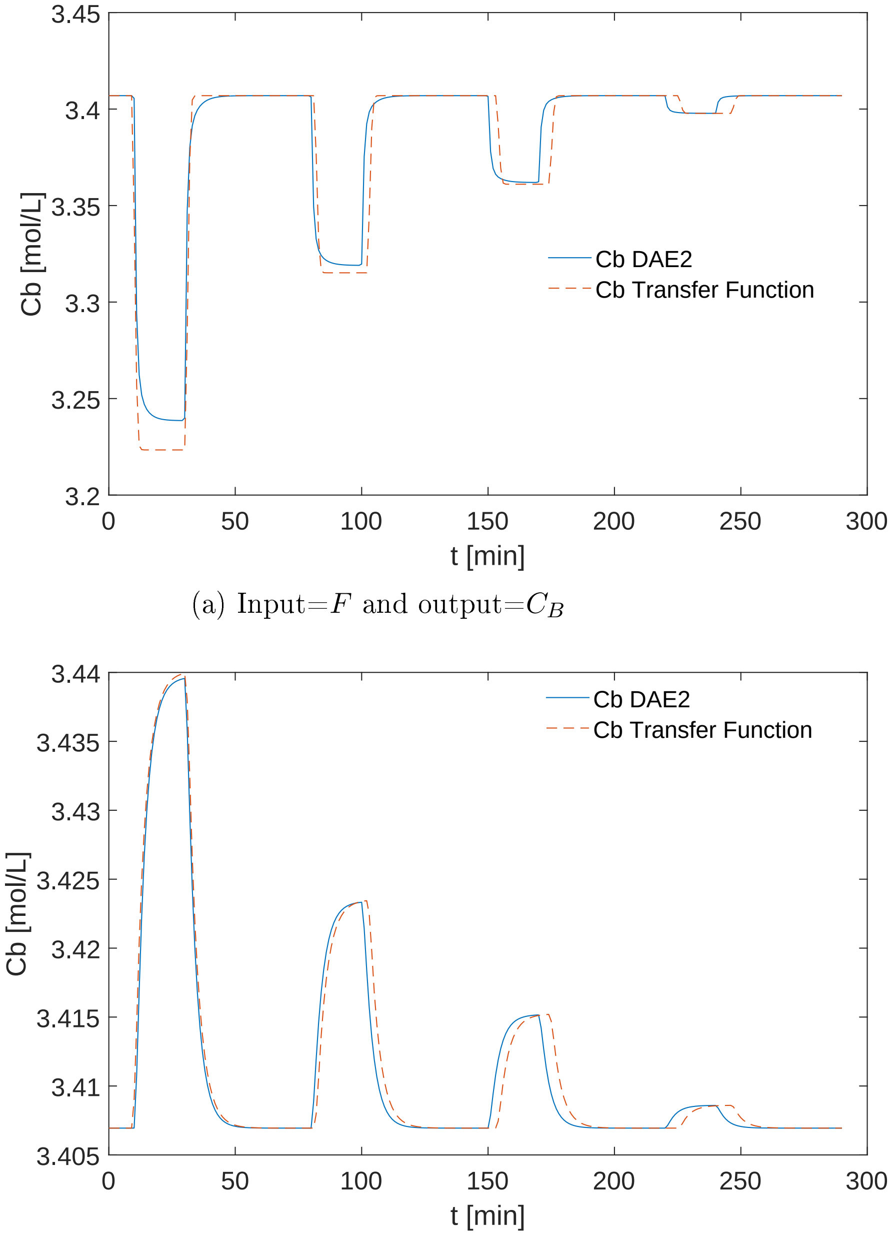


(a) Input=F and output=T

(b) Input=F_h and output=T

Figure 4.11: Comparison between transfer function and DAE response of Case 3 for output T and steps of ±10 and ±5% on F and F_h.

phase fraction, l, is described by the following relationship, $\phi + l = 1$. The pressure, P, was chosen as input to control the liquid fraction of B, x_B, after analyzing the process sensitivity to this parameter. Also, the heat added to the system, Q, was investigated, but its influence in the output variable was smaller if compared to the pressure. As in the Case 1, only two models were successfully simulated: the index-2 and its reduced form to index-1. The fully reduced index-0 DAE was not simulated due to persistent error during integration, probably because inconsistency in the

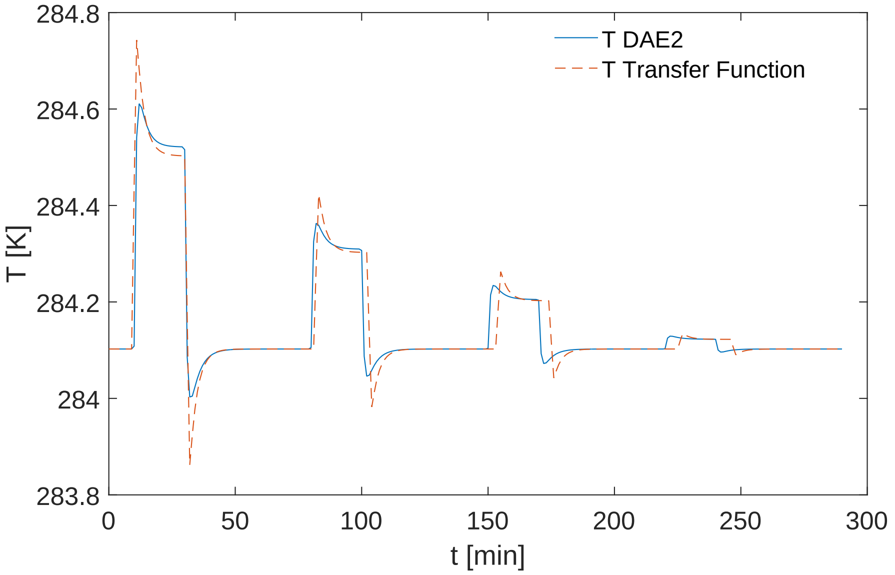


(a) Input=F and output=C_B

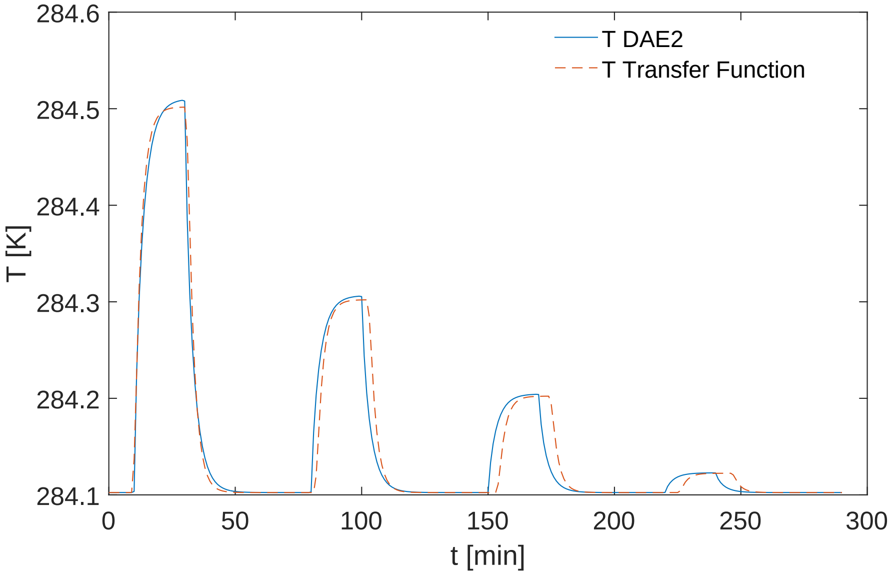
(b) Input=F_h and output=C_B

Figure 4.12: Comparison between transfer function and DAE response of Case 3 for output C_B and steps of +20, +10, +5 and +1% on F and F_h.

initial condition as the DASSLC failed to proceed. This case presented a different behavior between index-2 and index-1 models, the responses were completely different, as can be seen in Figure 4.14. With only one index reduction, the response to disturbance gives completely different results, making the index reduction an approach that needs extra care when used, specially for complex systems, such as the reactive flash drum. To illustrate the difference between the disturbances applied on P and Q, Figure 4.15 shows the response of x_B when applied a step change of ±10%



(a) Input=F and output=T



(b) Input=F_h and output=T

Figure 4.13: Comparison between transfer function and DAE response of Case 3 for output T and steps of +20, +10, +5 and +1% on F and F_h.

on P and Q, respectively. The disturbance applied on P affects directly the algebraic constraint, as the liquid-vapor equilibrium constant, K_i, depends on it through the relationship $K_i = \dfrac{P_i^{sat}}{P}$. The disturbance applied on Q did not cause any difference between the models responses, but the disturbance applied on P causes the reduced index-1 model to behave differently from the index-2 model.

The comparison between the DASSLC and transfer function was simulated for different step changes - +20, +10, +5 and +1% - and is presented in Figure 4.16.

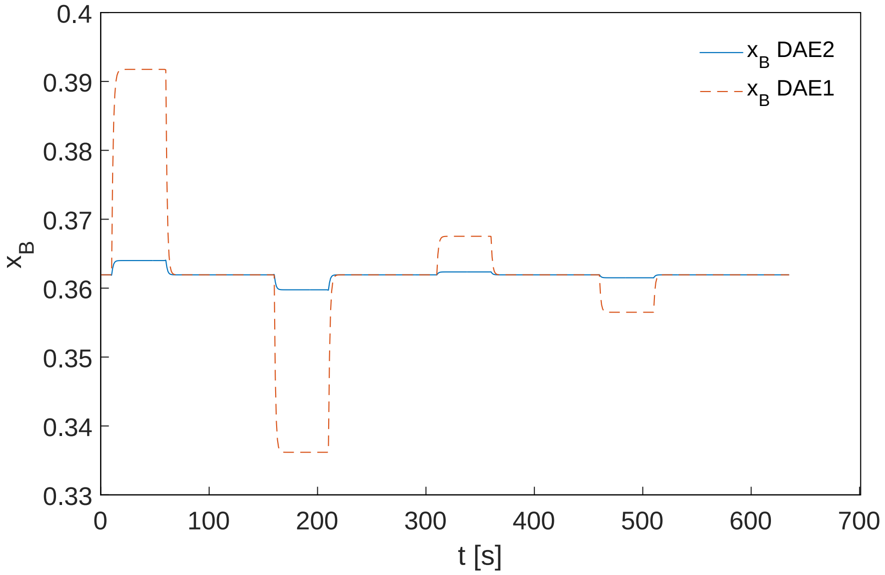


Figure 4.14: Simulation results of Case 4 for step of ±5 and ±1% on P.

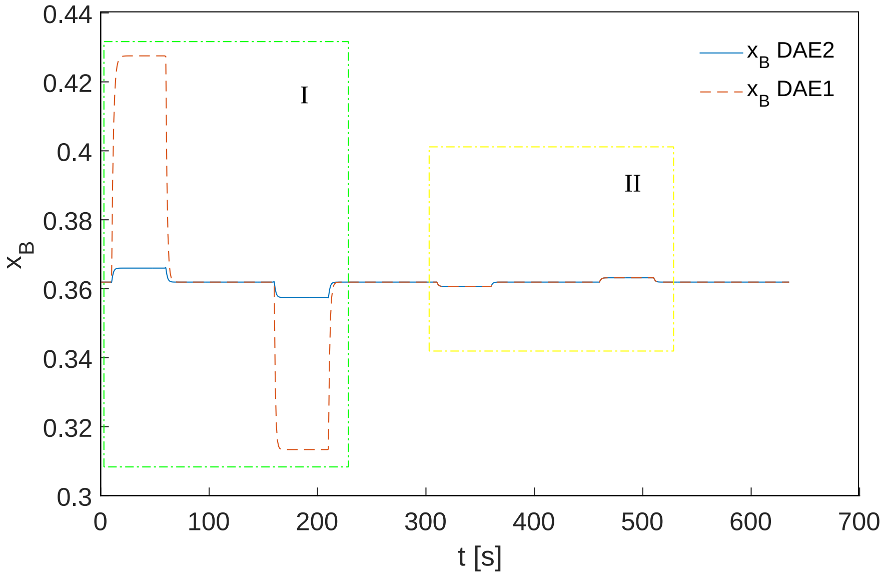


Figure 4.15: Simulation results of Case 4 for step of ±10% on P, interval I, and ±10% on Q, interval II.

A difference in the gain is observed for steps larger than 10%, but the control algorithm chosen is expected to handle such discrepancy in the predicting model, as long as the direction is the same as the process (please refer to Section 4.3 for this investigation). The first order identified transfer function is described by Equation 4.4. The simulation fit for the transfer function was 96.37% and MSE of $9.587 \cdot 10^{-10}$.

$$G(s) = \frac{0.0423}{0.8453s + 1} \tag{4.4}$$

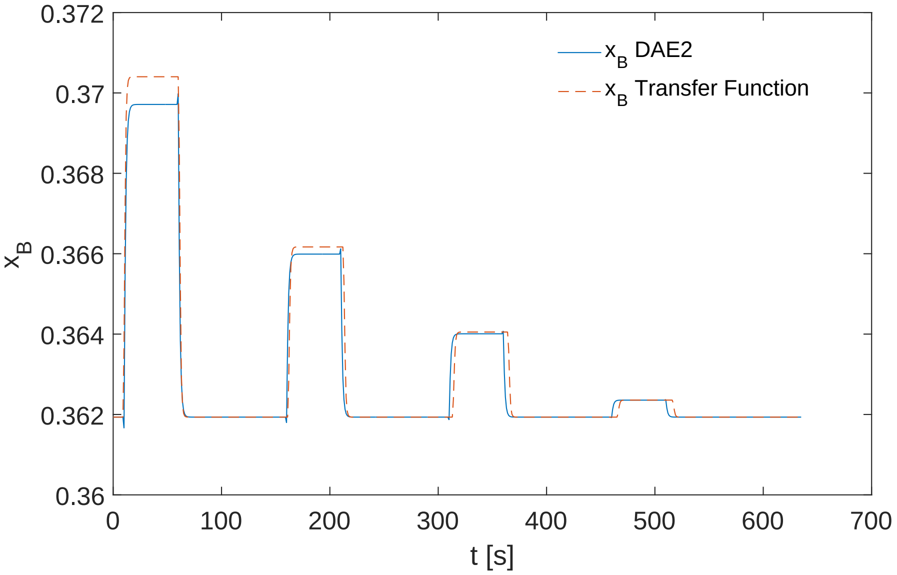


Figure 4.16: Comparison between transfer function and DAE response of case 4 for output x_B and steps of +20, +10, +5 and +1% on P.

4.3 Control Results

After representing the DAE models with transfer functions, those functions are supplied to the GPC and are discretized using the defined sampling time. Firstly the prediction horizon is fixed to cover the main dynamics of the process and then the control horizon is changed to improve performance. The weight matrices are adjusted in the final tuning. The control simulation consists in setpoint changes and disturbances in controlled variables. The results of the control problems treated here show the strong capability of the MPC strategy to deal with a complex process even when the predicting model is not completely accurate. The disturbance simulated are of constant changes, such as +10% on the measurement of controlled variable. Some problems using Gaussian noise show the high sensitivity of the GPC to high frequency noise, but performs reasonably well to rejecting low frequency noise, as well as the constant change applied.

Case 1 uses the inlet flow rate as manipulated variable, F, to control the liquid flow rate, L, as illustrated in Figure 4.17. The disturbance in the controlled variable is an increase of 10% on the output L between the period of 800 to 900 seconds. The setpoint changes were simulated with changes ranging between 10 and 20% in L in the period from 0 to 700s. It can be seen that the controller exhibited a good servo and regulatory behavior.

For Case 2 in Figure 4.18, the control simulation is for input F and output C_B. Different setpoint changes were applied from 0 to 300s and one disturbance rejection was investigated, when an increase of 10% was applied on C_B. The setpoint changes

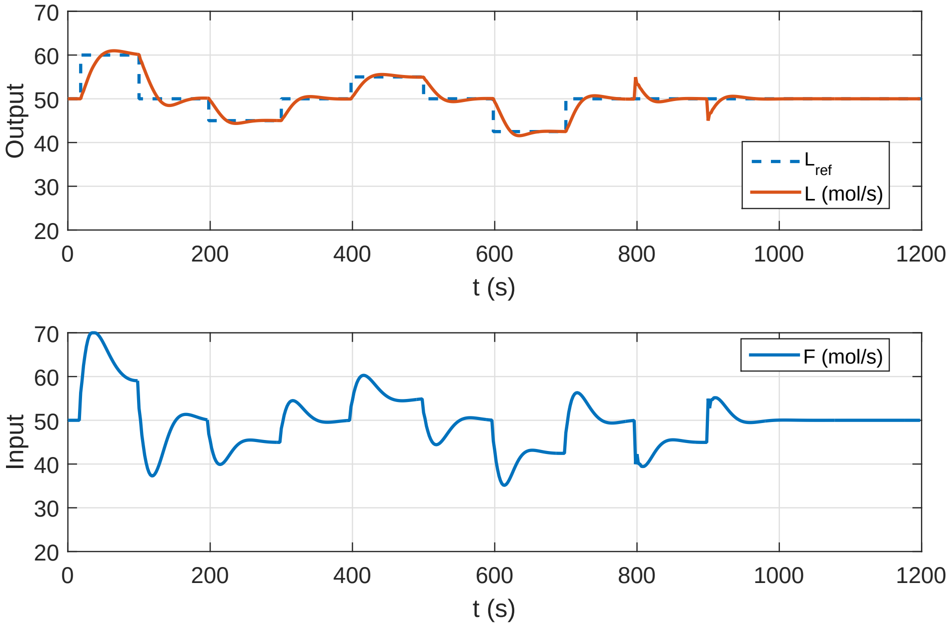


Figure 4.17: Control simulation of Case 1 for different setpoint changes and disturbance of +10% on controlled variable L.

range from 5 to 15% of C_B in the period from 0 to 300s. The disturbance simulated on C_B is of 10% from 350-400s. Again, it can be noticed that the controller followed the set-point changes and rejected the disturbance.

Case 3 is a MIMO system with inputs F and F_h and outputs C_B and T, with results in Figure 4.19. The setpoint change for C_B is implemented during the period of the simulation designated by the interval I and the change in setpoint of T is indicated in interval II. The disturbances on measurements of C_B and T are indicated in intervals III and IV, respectively.

In Case 4 the total pressure, P, is used as manipulated variable and the controlled variable is x_B. Different setpoint changes were simulated in the period of 0-300s and disturbances were of two different kinds: the first one was to simulate a disturbance in x_B of +10% in the period 350-400s and a second one simulating random noise from 0 to 10%, in the interval 450-500s. The disturbance rejection considering the deterministic disturbance is well performed, but the GPC seems to be too sensitive to white noise in the outputs, making the manipulate variable oscillate in the same frequency. To improve disturbance rejection in GPC, the internal model can be modified with an inclusion of other polynomial as proposed by DE SOUZA (1989), this was left for future work as the rejection of disturbances is related to offset free behavior. The control results using P as manipulated variable are shown in Figure 4.20.

The control parameters used on the case studies analyzed here are listed in Table

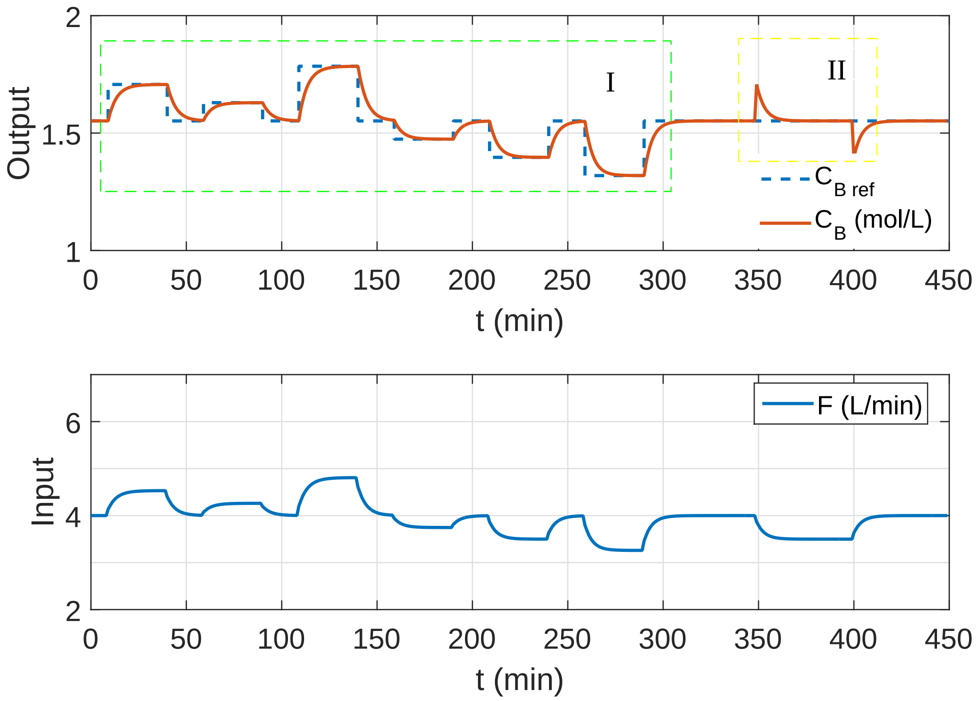


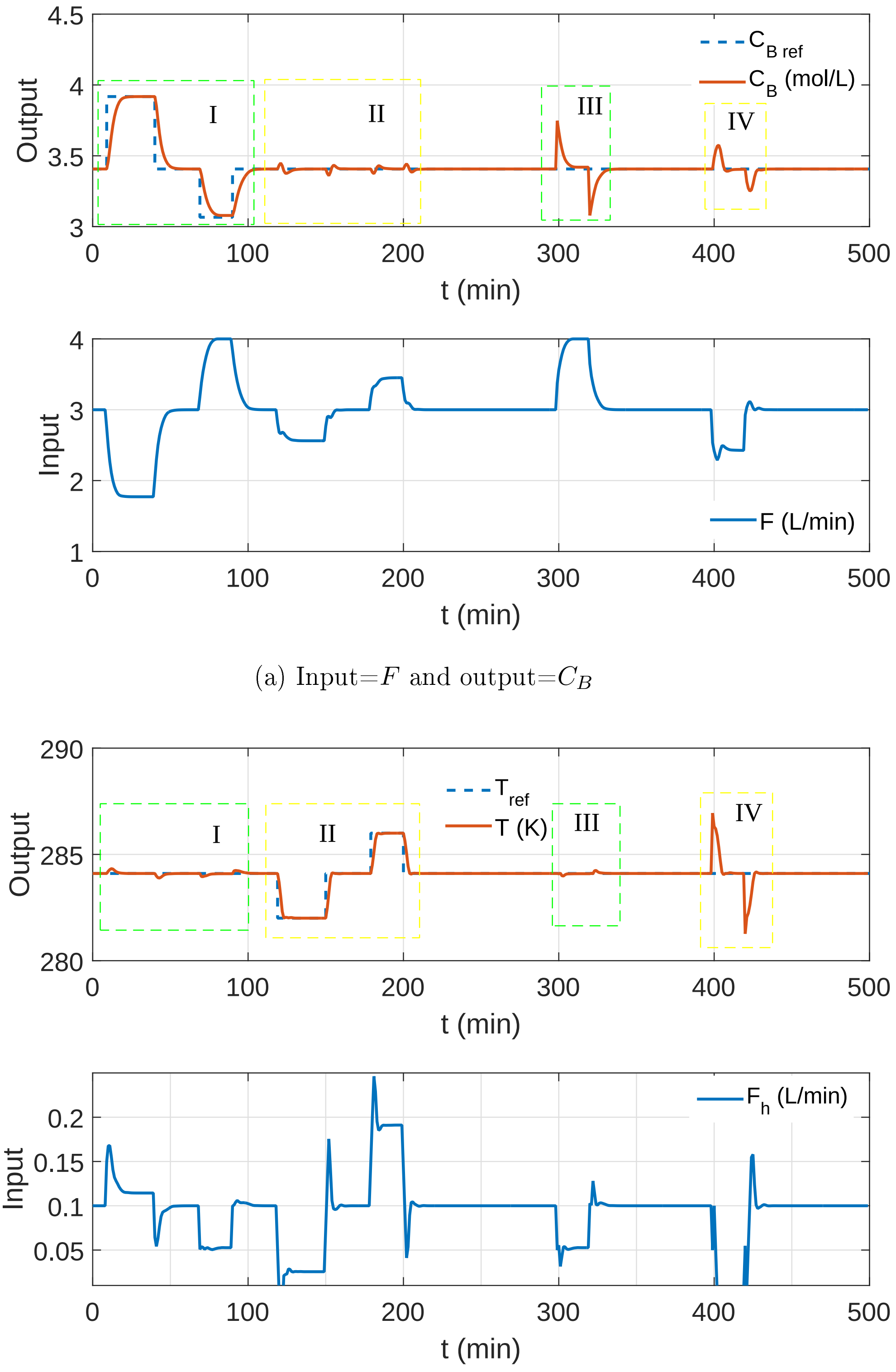
Figure 4.18: Control simulation of Case 2 for different setpoint changes, interval I, and disturbance of +10% on controlled variable C_B, interval II.

4.1. The Case 3 present two values of weights of objective function, δ and λ, one for each input and output variables. The parameter δ weights the offset difference and the parameter λ weights the control action variance.

Table 4.1: Parameters used in the control problems addressed in the work

	Case 1	Case 2	Case 3	Case 4
Np	5	5	10	10
Nu	3	2	5	5
δ	1	1	C_B: 1 | T: 0.1	70
λ	1	1	F: 1 | F_h: 1	1
$t_s(s)$	2	60	60	1

An alternative manipulated variable for the reactive flash drum problem is the heat added to the system, Q, which was varied when changes were applied in the setpoint by ±5, ±10 and ±15% in the controlled variable, x_B. The control could not achieve setpoints higher than +5% for x_B, but could follow decreasing setpoints of the value of x_B and reject disturbance of 10% in x_B, as shown in Figure 4.21.



(a) Input=F and output=C_B

(b) Input=F_h and output=T

Figure 4.19: Control simulation of Case 3 for setpoint changes in C_B, interval I, and T, interval II. Disturbances of 10% on C_B, interval III, and 1% on T, interval IV.

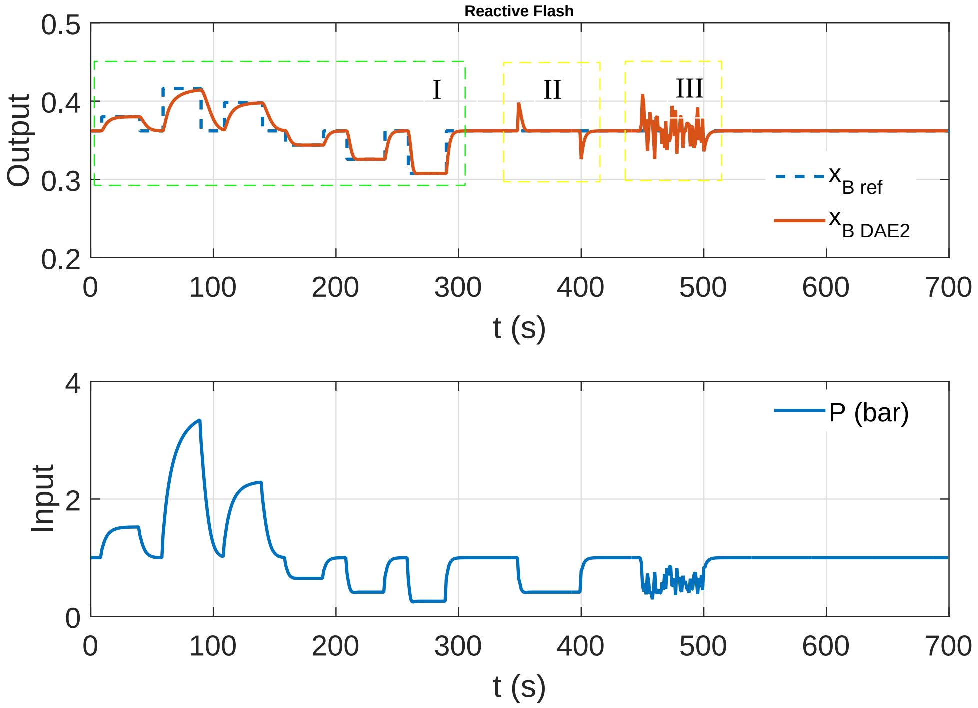


Figure 4.20: Control simulation of Case 4 using P as manipulated variable for different setpoint changes, interval I. Disturbances of +10%, interval II, and random noise of $0 - 10\%$, interval III, on controlled variable x_B.

4.4 Remarks on the Results

The results presented above showed the convenience of using the direct method of initialization to simulate and control of the high-index DAE systems. The responses for open loop dynamic simulations found for the reduced index systems were shown to be spurious because they may differ from the response of the original models, the DAE or ODE. The control simulation benefited by the direct method by removing discontinuities inherent to control actions and the use of regularization function appears to be another convenient technique to simulate systems sensitive to abrupt changes on variables or parameters.

In regard to the simulation and control results, the use of DAE models as they are proposed (without index order reduction) is of great importance. There are cases where only one differentiation of the algebraic constraint could cause spurious results compared with the original DAE, as described in Case 4. Cases 1, 2 and 3 presented the same responses to disturbances for the DAE2 and DAE1 models but presented a completely different response for DAE0 model. These results show that index reduction, generically speaking, is not a convenient approach to simulate DAE systems and needs extra care when applied. The understanding of the process, as well as the expected responses, need to be compared, if possible, with the original

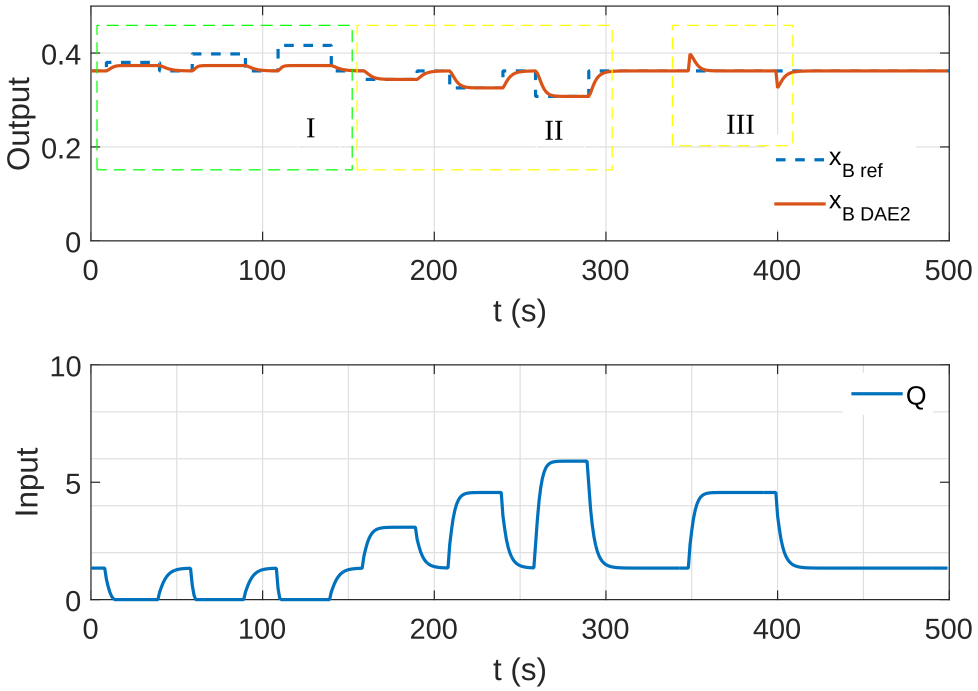


Figure 4.21: Control simulation of Case 4 using Q as manipulated variable for positive and negative setpoint changes, interval I and II, respectively. Disturbance of +10%, interval III, on controlled variable x_B.

model.

In regard to the regularization function, one needs to select carefully the regularization parameter and how the input will be applied, if a more realistic description of the behavior of the actuator is wanted. The process identification needs to be carefully studied if the index reduced DAE system is chosen. The different behavior when reduced systems are disturbed creates a warning to the application of this technique. The identified transfer functions showed a very good correlation to the DAE models, but with some deviations in few situations, but, for control proposes, the identification gives useful data, as one may see in the high value of fit and in the small MSE for the identified models.

Chapter 5

Conclusions

An exploratory study about the simulation and control of processes described by Differential-Algebraic Equations was carried out in this work. The large amount of information needed to understand reasonably well the intrinsic characteristics of DAE systems and their peculiarities regarding dynamic simulation and control application worth the time dedicated to study it.

The results presented here show that dealing directly with high-index DAE systems as they are described may be a better approach due to different behavior introduced by the index reduction technique. In simple systems, as described by Cases 1, 2 and 3, the index reduction shows different conclusions when compared to the fully reduced system of index-0. The Cases 1 and 4 could not even be simulated for index-0. In a little more complex case, such as Case 4, the result of index reduced system is somewhat worrisome, as the system with only one step of index reduction gives completely different results, if this possibility of spurious results is dismissed, wrong conclusions may be drawn based on results masked by index reduction technique. In Cases 2 and 3, where the the systems are integrated with the DAE models, the behavior of the DAEs agrees with those from the ODE system associated, but the index 0 differs from all when submitted to step changes. This also indicates the benefit of using the system in its original form, without index reduction or algebraic manipulation.

Regarding the direct method of initialization, it proved to be a convenient approach when dealing with high-index DAE systems, but it is still limited to the stage of finding consistent initial conditions. The use of stationary point for consistent initialization proved to be convenient. To overcome such difficulties, one may need to compare the parameters and the scales of variables in order to give the integration method the best numerical condition to successfully integrate such systems.

Regarding the MPC strategy chosen, the GPC proved to be of great efficiency for tracking setpoint changes and rejecting low frequency disturbances, but lacks of robustness when white frequency noise is present in the measured variables. This is reported to be improved by designing low pass filters using the $T(z)$ parameter on CARIMA model with a polynomial that rejects noise based on poles (DE SOUZA, 1989; ROSSITER, 2003). The GPC also proved to be a convenient controller when working with model mismatch between the predicting and the plant model. The main contribution of this work is at last to show how index reduction technique can interfere in the results when studying simulation and control of systems described by high-index DAE models and how to apply the current techniques on simulating such class of systems.

5.1 Recommendations for Future Work

The developed approach used the GPC algorithm, which is a control technique that deals with ample classes of systems described by CARIMA model. In future studies another control technique can be addressed, the NMPC, by using as the internal model a DAE system describing the process. A comparison with other solvers can be studied in other to check consistency of the results. Different regularization functions can be tested to study its impact and different index reduction procedures.

Bibliography

ABDULLA, T., CASH, J., "5 Solver MEBDFI", .

ALBERTOS, P., ORTEGA, R., 1989, "On generalized predictive control: Two alternative formulations", *Automatica*, v. 25, n. 5, pp. 753–755.

ALBET, J., LE LANN, J., JOULIA, X., et al., 1994, "Operational policies for the start-up of batch reactive distillation column". In: *Institution of Chemical Engineers Symposium Series*, v. 133, pp. 63–63. HEMSPHERE PUBLISHING CORPORATION.

ANDRADE NETO, A. S., 2018, "Modelo adimensional de flash reativo algébrico-diferencial de índice 2", *Qualification exam - Programa de Engenharia Química-COPPE-UFRJ*.

AZEVEDO-PERDICOÚLIS, T.-P., JANK, G., 2007, "A 2-DAE system gas network model in view to optimisation as a dynamic game". In: *Multidimensional (nD) Systems, 2007 International Workshop on*, pp. 57–64. IEEE.

BAUM, A.-K., KOLMBAUER, M., OFFNER, G., 2017, "Topological solvability and DAE-index conditions for mass flow controlled pumps in liquid flow networks", *Electronic Transactions on Numerical Analysis*, v. 46, pp. 394–423.

BIEGLER, L. T., CAMPBELL, S. L., MEHRMANN, V., 2012, *Control and optimization with differential-algebraic constraints*. SIAM.

BONILLA, J., LOGIST, F., DEGRÈVE, J., et al., 2012, "A reduced order rate based model for distillation in packed columns: Dynamic simulation and the differentiation index problem", *Chemical Engineering Science*, v. 68, n. 1, pp. 401–412.

BRENAN, K. E., CAMPBELL, S. L., PETZOLD, L. R., 1996, *Numerical solution of initial-value problems in differential-algebraic equations*, v. 14. Siam.

CAMACHO, E., BORDONS, C., 1999, *Model predictive control*. Springer Verlag.

CAMPBELL, S. L., 1995, "High-index differential algebraic equations", *Journal of Structural Mechanics*, v. 23, n. 2, pp. 199–222.

CAMPBELL, S. L., MOORE, E., 1994, "Progress on a general numerical method for nonlinear higher index DAEs II", *Circuits, Systems and Signal Processing*, v. 13, n. 2-3, pp. 123–138.

CARVALHO, D. F., 2015, "Controlador Preditivo Otimizado Aplicado ao Controle de Velocidade de Motor CC", *XII Congresso de Pesquisa, Ensino e Extensão da UFG*.

CLARKE, D. W., MOHTADI, C., TUFFS, P., 1987a, "Generalized predictive control—Part I. The basic algorithm", *Automatica*, v. 23, n. 2, pp. 137–148.

CLARKE, D., MOHTADI, C., TUFFS, P., 1987b, "Generalized predictive control—part II. Extension and Interpretations", *Automatica*, v. 23, n. 2, pp. 149–160.

CONTOU-CARRERE, M.-N., DAOUTIDIS, P., 2005, "An output feedback precompensator for nonlinear DAE systems with control-dependent state-space", *IEEE transactions on automatic control*, v. 50, n. 11, pp. 1831–1835.

COSTA JR., E. F. D., 2003, *Resolução automática de equações algébrico-diferenciais de índice superior*. Ph.D. Thesis, Programa de Engenharia Química-COPPE-UFRJ (In Portuguese).

CUILLE, P., REKLAITIS, G., 1986, "Dynamic simulation of multicomponent batch rectification with chemical reactions", *Computers & chemical engineering*, v. 10, n. 4, pp. 389–398.

CUTLER, C., RAMAKER, B., 1979, "Aiche national meeting", *Houston, USA, WP5-B*.

DA SILVA, D. V. A., 2013, *Aprimoramento de métodos numéricos para a integração numérica de sistemas algébrico-diferenciais*. Master Thesis, Programa de Engenharia Química-COPPE-UFRJ (In Portuguese).

DE SOUZA, J., 1989, *MB Controle Preditivo de Longo Alcance: Uma Investigação*. Ph.D. Thesis, Dissertação de mestrado, COPPE, UFRJ. Rio de Janeiro.

GARCIA, C. E., MORARI, M., 1982, "Internal model control. A unifying review and some new results", *Industrial & Engineering Chemistry Process Design and Development*, v. 21, n. 2, pp. 308–323.

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.

GERDIN, M., 2006, *Identification and estimation for models described by differential-algebraic equations.* Ph.D. Thesis, Institutionen för systemteknik.

GIRALDO, S. A. C., 2018, "Control GPC", *https://controlautomaticoeducacion.com /control-predictivo/control-gpc/*, (dez.).

GONÇALVES, F., CASTIER, M., ARAÚJO, O., 2007, "Dynamic simulation of flash drums using rigorous physical property calculations", *Brazilian Journal of Chemical Engineering*, v. 24, n. 2, pp. 277–286.

GOPAL, V., BIEGLER, L. T., 1998, "A successive linear programming approach for initialization and reinitialization after discontinuities of differential-algebraic equations", *SIAM Journal on Scientific Computing*, v. 20, n. 2, pp. 447–467.

HACHTEL, C., KERLER-BACK, J., BARTEL, A., et al., 2018, "Multirate DAE/ODE-Simulation and Model Order Reduction for Coupled Field-Circuit Systems". In: *Scientific Computing in Electrical Engineering: SCEE 2016, St. Wolfgang, Austria, October 2016*, pp. 91–100. Springer.

HARNEY, D. A., MILLS, T. K., BOOK, N. L., 2013, "Numerical evaluation of the stability of stationary points of index-2 differential-algebraic equations: Applications to reactive flash and reactive distillation systems", *Computers and Chemical Engineering*, v. 49, pp. 61–69.

HASSKERL, D., SUBRAMANIAN, S., HASHEMI, R., et al., 2017, "State estimation using a multi-rate particle filter for a reactive distillation column described by a DAE model". In: *Control and Automation (MED), 2017 25th Mediterranean Conference on*, pp. 876–881. IEEE.

HÖCKERDAL, E., FRISK, E., ERIKSSON, L., 2018, "Real-time performance of DAE and ODE based estimators evaluated on a diesel engine", *Science China Information Sciences*, v. 61, n. 7, pp. 70202.

KRÖNER, A., MARQUARDT, W., GILLES, E., 1997, "Getting around consistent initialization of DAE systems?" *Computers & Chemical Engineering*, v. 21, n. 2, pp. 145–158.

KUMAR, A., DAOUTIDIS, P., 1995a, “Control of nonlinear differential-algebraic-equation systems with disturbances”, *Industrial & engineering chemistry research*, v. 34, n. 6, pp. 2060–2076.

KUMAR, A., DAOUTIDIS, P., 1995b, “A DAE framework for modeling and control of reactive distillation columns”. In: *Dynamics and Control of Chemical Reactors, Distillation Columns and Batch Processes (Dycord’95)*, Elsevier, pp. 99–104, b.

KUMAR, A., DAOUTIDIS, P., 1996, “Feedback regularization and control of nonlinear differential-algebraic-equation systems”, *AIChE journal*, v. 42, n. 8, pp. 2175–2198.

KUMAR, A., DAOUTIDIS, P., 1997, “High-index dae systems in modeling and control of chemical processes”, *IFAC Proceedings Volumes*, v. 30, n. 6, pp. 393–398.

KUMAR, A., DAOUTIDIS, P., 1999, *Control of nonlinear differential algebraic equation systems with applications to chemical processes*, v. 397. CRC Press.

LEIMKUHLER, B., PETZOLD, L. R., GEAR, C. W., 1991, “Approximation methods for the consistent initialization of differential-algebraic equations”, *SIAM Journal on Numerical Analysis*, v. 28, n. 1, pp. 205–226.

LEITOLD, A., GERZSON, M., 2010, “Structural decomposition of process models described by higher index DAE systems”, *Computer Aided Chemical Engineering*, v. 28, pp. 385–390.

LI, PENGFEI; LI, Y. S. J. E., 2010, “Consistent initialization of system of differential-algebraic equations for dynamic simulation of centrifugal chillers”, *Journal of Building Performance Simulation*, v. 5, n. 2, pp. 115–139.

LIMA, E. R., CASTIER, M., BISCAIA, E. C., 2008, “Differential-algebraic approach to dynamic simulations of flash drums with rigorous evaluation of physical properties”, *Oil & Gas Science and Technology-Revue de l’IFP*, v. 63, n. 5, pp. 677–686.

LIOEN, W. M., DE SWART, J. J., VAN DER VEEN, W. A., 1998, *Specification of PSIDE.* Stichting Mathematisch Centrum.

MACIEJOWSKI, J. M., 2002, *Predictive control: with constraints.* Pearson education.

MATHWORKS, R., 2018a, "Solve Differential Algebraic Equations (DAEs)", *https://www.mathworks.com/help/matlab/math/solve-differential-algebraic-equations-daes.html*, (dez.).

MATHWORKS, R., 2018b, "Create linear and nonlinear dynamic system models from measured input-output data", *https://www.mathworks.com/products/sysid.html*, (dez.).

MCKENZIE, R., PRYCE, J., NEDIALKOV, N., et al., 2015, "Regularization of nonlinear DAEs based on Structural Analysis", *IFAC-PapersOnLine*, v. 48, n. 1, pp. 298–299.

MCLELLAN, P., 1994, "A differential-algebraic perspective on nonlinear controller design methodologies", *Chemical engineering science*, v. 49, n. 10, pp. 1663–1679.

MURATA, V., 1996, *Caracterização simbólica de equações algébrico diferenciais por um sistema de álgebra computacional com aplicações na engenharia química.* Ph.D. Thesis, Programa de Engenharia Química-COPPE-UFRJ (In Portuguese).

OGUNNAIKE, B. A., RAY, W. H., 1999, *Process dynamics, modeling, and control*, v. 1. Oxford University Press New York.

PANTELIDES, C. C., 1988, "The consistent initialization of differential-algebraic systems", *SIAM Journal on Scientific and Statistical Computing*, v. 9, n. 2, pp. 213–231.

PARIDA, N. C., RAHA, S., 2009, "The α-method direct transcription in path constrained dynamic optimization", *SIAM Journal on Scientific Computing*, v. 31, n. 3, pp. 2386–2417.

PETZOLD, L., 1982, "Differential/algebraic equations are not ODEs", *SIAM Journal on Scientific and Statistical Computing*, v. 3, n. 3, pp. 367–384.

PETZOLD, L., 1989, "DASSL Code (Differential Algebraic System Solver)", *Computing and Mathematics Research Division, Lawrence Livermore National Laboratory, Livermore, CA, USA.*

QIN, S. J., BADGWELL, T. A., 2000, "An overview of nonlinear model predictive control applications". In: *Nonlinear model predictive control*, Springer, pp. 369–392.

QUINTO, T., 2010, *Abordagem Algébrico-Diferencial da Otimização Dinâmica de Processos com Índice Flutuante.* Ph.D. Thesis, Programa de Engenharia Química-COPPE-UFRJ (In Portuguese).

RICHALET, J., RAULT, A., TESTUD, J., et al., 1978, "Model predictive heuristic control", *Automatica (Journal of IFAC)*, v. 14, n. 5, pp. 413–428.

ROSSITER, J. A., 2003, *Model-based predictive control: a practical approach.* CRC press.

SANTAMARÍA, F. L., GÓMEZ, J. M., 2015, "Index hybrid differential–Algebraic equations model based on fundamental principles for nonlinear model predictive control of a flash separation drum", *Industrial & Engineering Chemistry Research*, v. 54, n. 7, pp. 2145–2155.

SCHULZ, S., 2003, *Four lectures on differential-algebraic equations.* Relatório técnico, Department of Mathematics, The University of Auckland, New Zealand.

SECCHI, A., 1992, *Simulação dinâmica de processos químicos pelo método da relaxação em forma de onda em computadores paralelos.* Ph.D. Thesis, Programa de Engenharia Química-COPPE-UFRJ (In Portuguese).

SECCHI, A. R., 2007, *DASSLC User's Manual Version 3.2.* Relatório técnico.

SHAFIE, S., 2013, "Implementation of Modified SIRK Method on Solving Stiff Ordinary Differential Equations", *Int. J. Human. Manage. Sci*, v. 1, n. 1, pp. 8–12.

SOARES, R. D. P., SECCHI, A. R., 2005, "Direct initialisation and solution of high-index DAE systems". In: *European Symposium on Computer-Aided Process Engineering-15, 38th European Symposium of the Working Party on Computer Aided Process Engineering*, v. 20, pp. 157–162.

SOARES, R. P., 2007, *Depuração para simuladores de processos baseados em equações.* Ph.D. Thesis, Departamento de Engenharia Química-UFRGS (In Portuguese).

UPPAL, T., RAHA, S., SRIVASTAVA, S., 2017, "Trajectory feasibility evaluation using path prescribed control of unmanned aerial vehicle in differential algebraic equations framework", *The Aeronautical Journal*, v. 121, n. 1240, pp. 770–789.

VIANNA JR, A. D. S., NASCIMENTO, M. L. D., 2005, "Equações algébrico diferenciais para ensino em engenharia química", *Congresso Brasileiro de Ensino de Engenharia.*

VIEIRA, R., 1998, *Metodos Diretos Para Iniciaçáo de Sistemas Algebrico–Diferenciais.* Master Thesis, M. Sc. Dissertation, PEQ/COPPE/UFRJ (In Portuguese).

VIEIRA, R., 2001, "Técnicas de Inicialização de Sistemas Algébrico-Diferenciais", *Doctor on Science Thesis, Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ, Brasil.*

VIEIRA, R., BISCAIA JR., E., 2001, "Direct methods for consistent initialization of DAE systems", *Computers & Chemical Engineering*, v. 25, n. 9-10, pp. 1299–1311.

WANG, H.-S., 2010, "Derivation of the optimal guidance law for space vehicle based on a DAE approach". In: *System Science and Engineering (ICSSE), 2010 International Conference on*, pp. 407–412. IEEE.

WASCHL, H., ALBERER, D., DEL RE, L., 2011, "Automatic tuning methods for MPC environments". In: *International Conference on Computer Aided Systems Theory*, pp. 41–48. Springer.

WOLFRAM, A., 2018, "Numerical Solution of Differential-Algebraic Equations", *https://reference.wolfram.com/ language/tutorial/NDSolveDAE.html*, (dez.).

YE, Y., RUAN, J., LU, Z., et al., 2017, "Transient stability modes of DFIG in large disturbance based on reduced-order DAE models". In: *Power & Energy Society General Meeting, 2017 IEEE*, pp. 1–5. IEEE.

Appendix A

Dimensional and Dimensionless Models of the Reactive Flash Drum

The nondimensionalization of the reactive flash drum model explained by (ANDRADE NETO, 2018) is shown here. The model is based on two reaction in series $A \to B \to C$, considering the total molar holdup, N, constant and assuming energy dynamics much faster in vapor than in liquid. First the global molar balance is presented according to Equation A.1. With N constant, the derivative is null, $\dfrac{dN}{dt} = 0$. The term $\Delta\nu_j = \sum_{i=1}^{Ncomp} \nu_{i,j}$ is the component i stoichiometric balance for reaction j and r_j is the reaction rate using Arrhenius equation, $r_j = k_{0j} e^{\left(\dfrac{-E_j}{RT}\right)} C_j$. The term V_L is the liquid volume. It is assumed that the reactions occur only in liquid phase. The term $\Delta\nu_j = 0$ is null due to the stoichiometric balance for reactions in series, as no molar variation is supposed. With those conditions, the global molar balance reduces to the expression in Equation A.2., where F is the feed molar flow rate, and L and V are the liquid and vapor outlet, respectively.

$$\frac{dN}{dt} = F - L - V + \sum_{j=1}^{Nreac} \Delta\nu_j r_j V_L \tag{A.1}$$

$$0 = F - L - V \tag{A.2}$$

The component molar balance is presented in Equation A.3, with $R_i = \sum_{j=1}^{Nreac} \nu_{i,j} r_j$. Using the assumption of N constant, the equation becomes equation A.4

$$\frac{d\left(Nx_i\right)}{dt} = Fx_i - Lx_i - Vx_i + R_i V_L \tag{A.3}$$

$$\frac{dx_i}{dt} = \frac{F}{N} x_{iF} - \frac{L}{N} x_i - \frac{V}{N} y_i + \frac{R_i V_L}{N} \tag{A.4}$$

The component balances for two reactions are presented for the components A and B, equation A.5. The component C is obtained through the summation relationship, $x_A + x_B + x_C = 1$.

$$\frac{dx_A}{dt} = \frac{F}{N}x_{AF} - \frac{L}{N}x_A - \frac{V}{N}y_A + \frac{k_{0A}e^{\left(\frac{-E_A}{RT}\right)}C_AV_L}{N} \tag{A.5a}$$

$$\frac{dx_B}{dt} = \frac{F}{N}x_{BF} - \frac{L}{N}x_B - \frac{V}{N}y_B + \frac{k_{0B}e^{\left(\frac{-E_B}{RT}\right)}C_BV_L}{N} \tag{A.5b}$$

Using the relationship $C_iV_L = N_i$ and $\frac{N_i}{N} = x_i$, the equation A.5 becomes equation A.6.

$$\frac{dx_A}{dt} = \frac{F}{N}x_{AF} - \frac{L}{N}x_A - \frac{V}{N}y_A + k_{0A}e^{\left(\frac{-E_A}{RT}\right)}x_A \tag{A.6a}$$

$$\frac{dx_B}{dt} = \frac{F}{N}x_{BF} - \frac{L}{N}x_B - \frac{V}{N}y_B + k_{0B}e^{\left(\frac{-E_B}{RT}\right)}x_B \tag{A.6b}$$

The energy balance is based on the total enthalpy, H^t, which is composed by the liquid and vapor enthalpies, H^L and H^V, through $H^t = H^L + H^V$. By assuming negligible vapor holdup, the liquid enthalpy becomes greater than vapor enthalpy, $H^V << H^L$, by assuming the dynamics of vapor enthalpy much faster that of the liquid, $\frac{dH^V}{dt} << \frac{dH^L}{dt}$, the following relationship is produced, $\frac{dH^t}{dt} = \frac{dH^L}{dt}$. Using the specific enthalpy expressions; $H^L = Nh_L$, $dh_L = \bar{c}_p dT$, $\bar{c}_p = \sum_{i=1}^{Ncomp} x_i c_{pi}$, the total enthalpy can be expressed as $\frac{dH^t}{dt} = N\bar{c}_p\frac{dT}{dt}$. The total energy balance using the previous relationships is expressed in equation A.7.

$$N\bar{c}_p\frac{dT}{dt} = Fh_F - Lh_L - Vh_V + \dot{q} - \sum_{i=1}^{Nreac} \Delta H_r R_i V_L \tag{A.7}$$

Reorganizing the expression and opening the summation we have equation A.8

$$\frac{dT}{dt} = \frac{Fh_F}{N\bar{c}_p} - \frac{Lh_L}{N\bar{c}_p} - \frac{Vh_V}{N\bar{c}_p} + \frac{\dot{q}}{N\bar{c}_p} - \frac{\Delta H_r^{AB} k_{0A}e^{\left(\frac{-E_A}{RT}\right)}C_AV_L}{N\bar{c}_p} + \frac{\Delta H_r^{BC} k_{0B}e^{\left(\frac{-E_B}{RT}\right)}C_BV_L}{N\bar{c}_p} \tag{A.8}$$

By substituting $\frac{C_iV_L}{N\bar{c}_p} = x_i$ and the expression for enthalpy; $h_F = h_{ref} +$

$\overline{c}_p\left(T_F - T_{ref}\right)$, $h_L = h_{ref} + \overline{c}_p\left(T - T_{ref}\right)$, $h_V = h_L + \Delta_{vap}$, with $h_{ref} = 0$, we have the following expression, equation A.9.

$$
\begin{aligned}
\frac{dT}{dt} = \frac{F\left(T_F - T_{ref}\right)}{N} - \frac{L\left(T - T_{ref}\right)}{N} - \frac{V\left[\Delta H_{vap} + \overline{c}_p\left(T - T_{ref}\right)\right]}{N\overline{c}_p} + \\
\frac{\dot{q}}{N\overline{c}_p} - \frac{\Delta H_r^{AB} k_{0A} e^{\left(\frac{-E_A}{RT}\right)} x_A}{\overline{c}_p} + \frac{\Delta H_r^{BC} k_{0B} e^{\left(\frac{-E_B}{RT}\right)} x_B}{\overline{c}_p}
\end{aligned}
\tag{A.9}
$$

Further nondimensionalization is performed on temperature, T, and time, t, using the following expressions, equation A.10.

$$\theta = \frac{T - T_{ref}}{T_{ref}} \tag{A.10a}$$

$$\tau = \frac{t}{N/F} \tag{A.10b}$$

Applying equation A.10 on energy and component balances we have the following system, equation A.11.

$$\frac{dx_A}{d\tau} = x_{AF} - \frac{Lx_A}{F} - \frac{Vy_A}{F} - \frac{Nk_{0A} e^{\left[\frac{-E_A}{RT_{ref}\left(\theta + 1\right)}\right]} x_A}{F} \tag{A.11a}$$

$$\frac{dx_B}{d\tau} = x_{BF} - \frac{Lx_B}{F} - \frac{Vy_B}{F} - \frac{Nk_{0B} e^{\left[\frac{-E_B}{RT_{ref}\left(\theta + 1\right)}\right]} x_B}{F} + \frac{Nk_{0A} e^{\left[\frac{-E_A}{RT_{ref}\left(\theta + 1\right)}\right]} x_A}{F} \tag{A.11b}$$

$$
\begin{aligned}
\frac{d\theta}{d\tau} = \theta_F - \frac{L}{F}\theta - \frac{V\Delta H_{vap}}{F\overline{c}_p T_{ref}} - \frac{V}{F}\theta + \frac{\dot{q}}{\overline{c}_p T_{ref} F} - \frac{\Delta H_r^{\hat{A}B}}{\overline{c}_p T_{ref}} \frac{Nk_{0A} e^{\left[\frac{-E_A}{RT_{ref}\left(\theta + 1\right)}\right]} x_A}{F} \\
- \frac{\Delta H_r^{BC}}{\overline{c}_p T_{ref}} \frac{Nk_{0B} e^{\left[\frac{-E_B}{RT_{ref}\left(\theta + 1\right)}\right]} x_B}{F}
\end{aligned}
\tag{A.11c}
$$

Based on equation A.11, the nondimensionalization is performed applying the expressions of Equation A.12 and A.13, producing the final model used on simulation, Equation A.14.

$$l = \frac{L}{F} \tag{A.12a}$$

$$\phi = \frac{V}{F} \tag{A.12b}$$

$$D_{aA} = \frac{Nk_{0A}}{F} \tag{A.12c}$$

$$D_{aB} = \frac{Nk_{0B}}{F} \tag{A.12d}$$

$$\gamma_A = \frac{-E_A}{RT_{ref}} \tag{A.12e}$$

$$\gamma_B = \frac{-E_B}{RT_{ref}} \tag{A.12f}$$

$$\lambda = \frac{\Delta H_{vap}}{\bar{c}_p T_{ref}} \tag{A.12g}$$

$$B1 = \frac{\Delta H_r^{AB}}{\bar{c}_p T_{ref}} \tag{A.12h}$$

$$B2 = \frac{\Delta H_r^{BC}}{\bar{c}_p T_{ref}} \tag{A.12i}$$

$$Q = \frac{\dot{q}}{\bar{c}_p T_{ref} F} \tag{A.13a}$$

$$x_C = 1 - x_A - x_B \tag{A.13b}$$

$$y_i = K_i x_i \tag{A.13c}$$

$$0 = y_A + y_B + y_C - 1 \tag{A.13d}$$

$$0 = K_A x_A + K_B x_B + K_C \left(1 - x_A - x_B\right) - 1 \tag{A.13e}$$

$$\frac{dx_A}{dt} = x_{AF} - (1-\phi)\, x_A - \phi K_A x_A - D_{aA} x_A e^{\left(\frac{\gamma_A}{\theta+1}\right)} \tag{A.14a}$$

$$\frac{dx_B}{dt} = x_{BF} - (1-\phi)\, x_B - \phi K_B x_B - D_{aB} x_B e^{\left(\frac{\gamma_B}{\theta+1}\right)} + D_{aA} x_A e^{\left(\frac{\gamma_A}{\theta+1}\right)} \tag{A.14b}$$

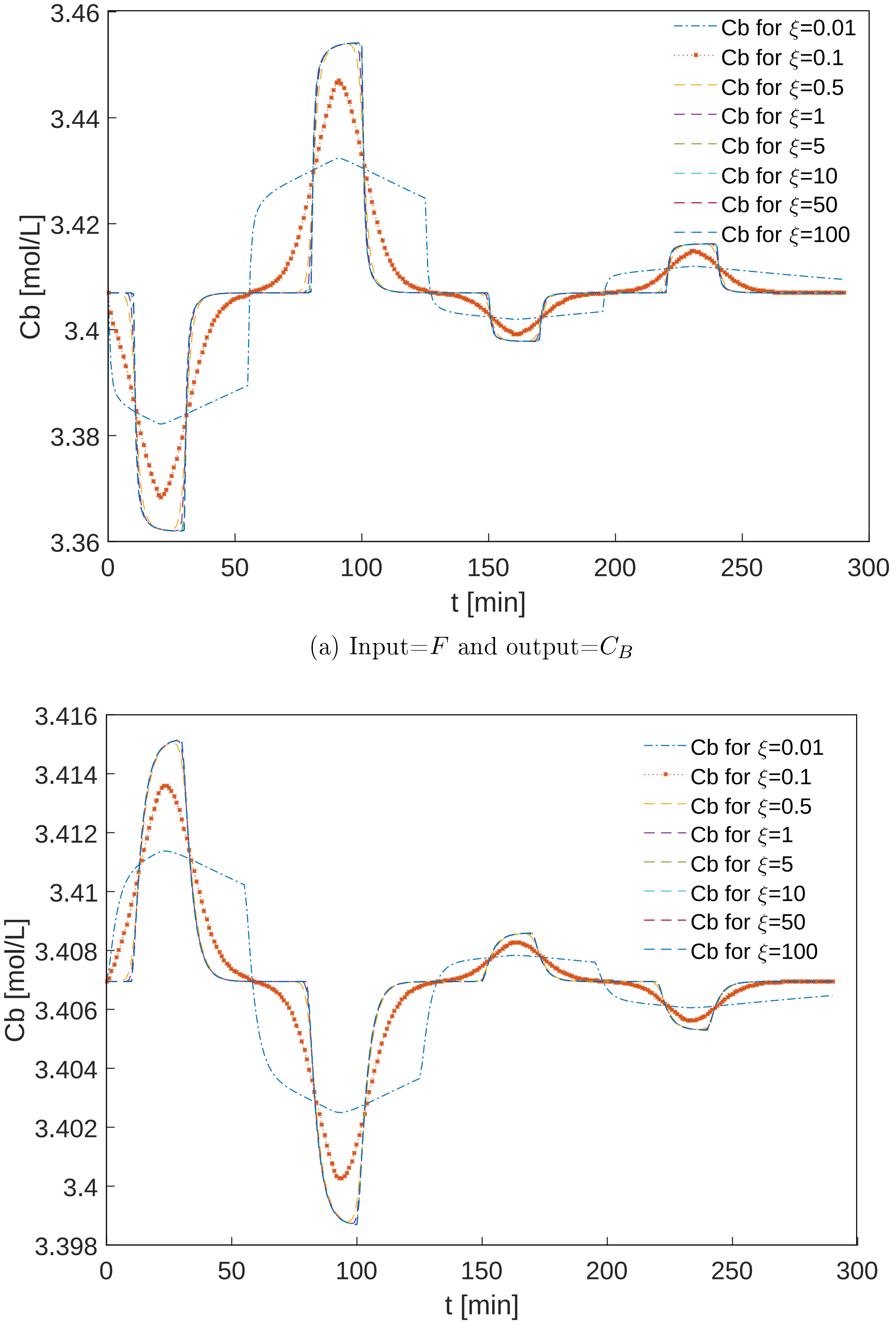
$$\frac{d\theta}{dt} = 1 - \theta - \lambda\phi + Q + B_1 D_{aA} x_A e^{\left(\frac{\gamma_A}{\theta+1}\right)} + B_2 D_{aB} x_B e^{\left(\frac{\gamma_B}{\theta+1}\right)} \tag{A.14c}$$

$$0 = K_A x_A + K_B x_B + K_C \left(1 - x_A - x_B\right) - 1 \tag{A.14d}$$

Appendix B

Simulation of Different Regularization Parameters

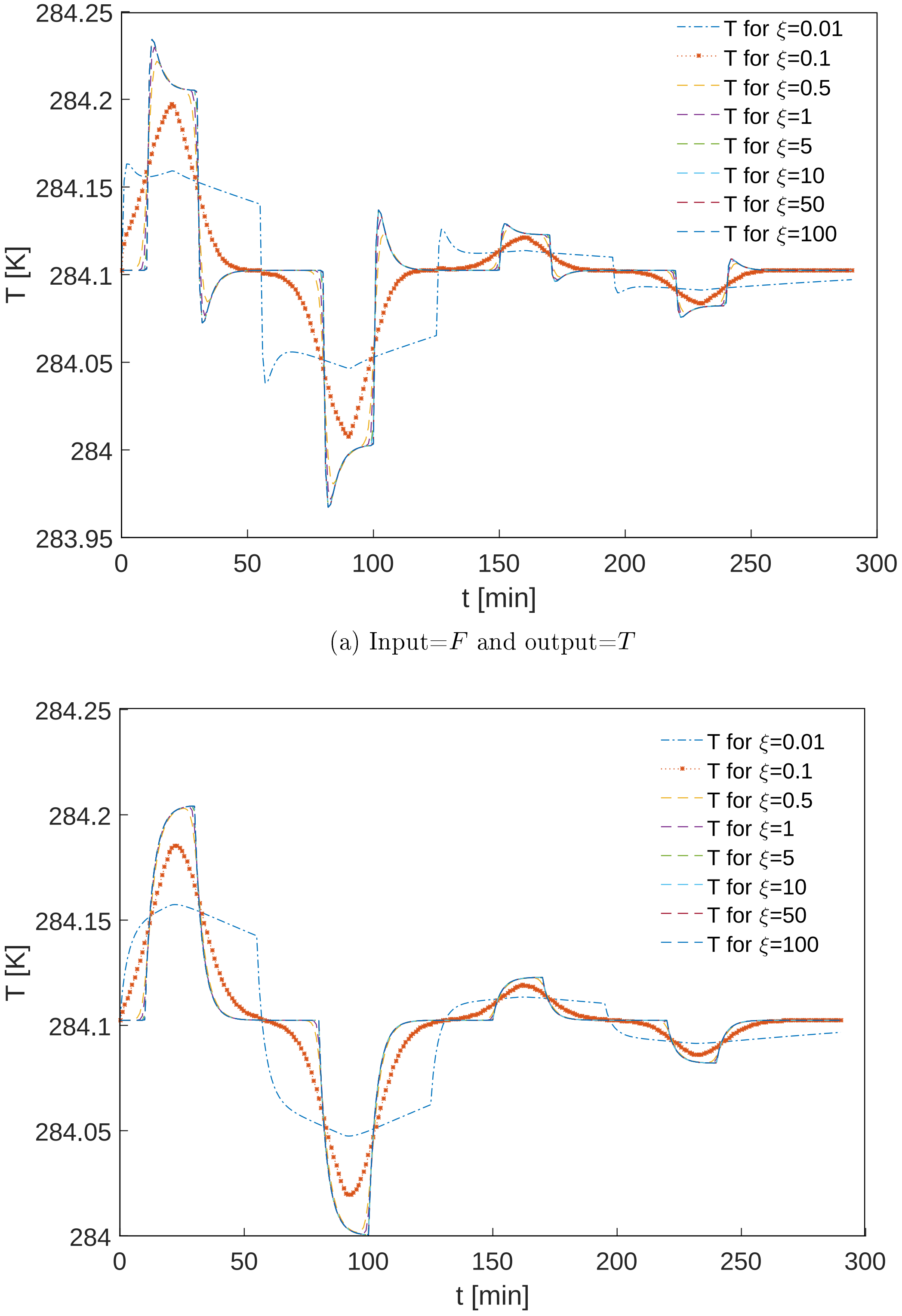
The impact of the regularization parameter for Cases 1, 3 and 4 is presented in Figures B.3, , B.1, B.2 and B.4. The simulations were performed for the index-2 DAE models using the manipulated and controlled variables applied on control study. The simulation of Case 2 is presented in Section 4.1. In Case 4 the value of $\xi = 0.01$ is not simulated because the solver could not integrate the system for this condition. Case 1 is the one that shows the smaller impact of the regularization parameter, as can be seen in Figure B.3, with a small smoothing of the responses. Case 3, Figures B.1 and B.2, is the one mostly impacted with the regularization parameter, presenting large differences in the responses for different values of ξ. Case 4 is the only that could not be simulated for the chosen values of ξ, but also present a small impact, such as Case 1, in the responses for the values simulated.



(a) Input=F and output=C_B

(b) Input=F_h and output=C_B

Figure B.1: Simulation of Case 3 with different values of ξ. Results for output C_B with disturbances in F and F_h.



(a) Input=F and output=T

(b) Input=F_h and output=T

Figure B.2: Simulation of Case 3 with different values of ξ. Results for output T with disturbances on F and F_h

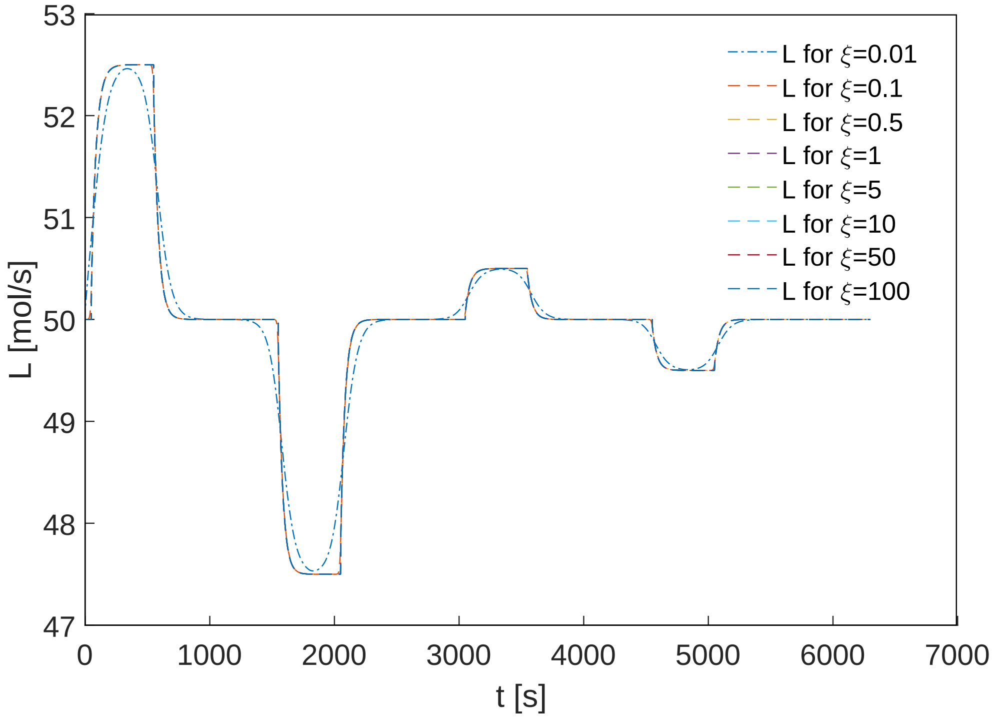


Figure B.3: Simulation of Case 1 with different values of ξ. Results for output L and disturbances in F

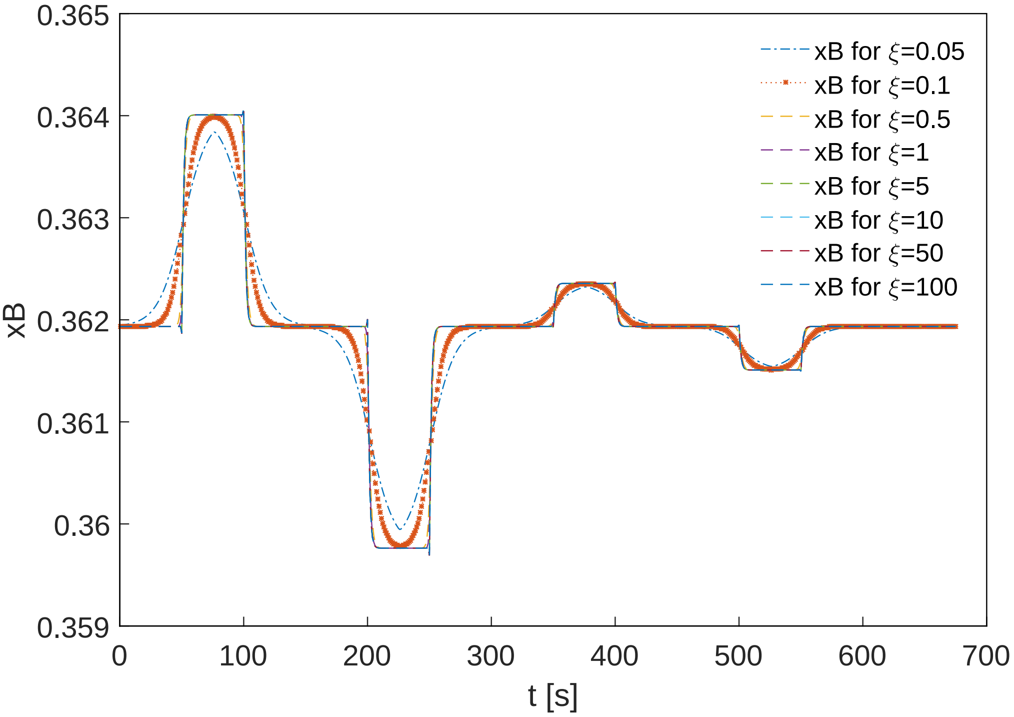


Figure B.4: Simulation of Case 4 with different values of ξ. Results for output xB and disturbances in P