NTNU

Norwegian University of
Science and Technology

Bilevel Model Predictive Control of a Semi-Batch Emulsion Copolymerisation Reactor

Ellen Kjetså

Chemical Engineering
Submission date: June 2017
Supervisor: Magne Hillestad, IKP
Co-supervisor: Peter Singstad, Cybernetica AS
Fredrik Gjertsen, Cybernetica AS

Norwegian University of Science and Technology
Department of Chemical Engineering

Abstract

The objective of this thesis is to study the implementation of an on-line bilevel model predictive control (MPC) application to an emulsion copolymerisation process in a semi-batch reactor. Previous work has been performed on a regular MPC layer as well as an off-line open-loop study of the process.

Using MPC in the process industry is becoming more common, allows for the optimisation of chemical processes. However, the implementation of MPC requires a process model that accurately depicts the behaviour of the process plant, and this is often time-consuming and cumbersome work. The nonlinear model used for this thesis was developed during the COOPOL EU-project, and has already been implemented on a pilot plant. Some differences on behaviour between the model predictions and the plant behaviour were observed.

This thesis gives an introduction to polymers, emulsion polymerisation and the most important kinetic mechanisms in free-radical emulsion polymerisation. It also provides the equations in the model of the process, which is implemented in the programming language C. The model follows a standard template that can be easily accessed by Cybernetica AS' specialised software for nonlinear real-time optimisation. The concept behind MPC is presented, as well as the concerns of the control of polymerisation processes. This provides a sufficient base for the understanding of the main purpose, implementation of a bilevel MPC control of the process.

All process plants contain a control hierarchy, from the long-term economic objectives to the fast PID controllers that ensure safe operation of the plant. This thesis presents a hierarchy of two MPC-controllers, one with a long time horizon that provides the trajectories, and the lower which strives to follow these trajectories. This aim is to discover if there is a benefit of bilevel control compared to one single MPC controller.

The main objective is to optimise the polymerisation time of the process, by allowing the temperature of the reactor and the flow rate of monomer and initiator feed to vary. However, the final product quality must remain as before. This thesis explores the possibility of controlling the process using bilevel MPC control. The results are promising. This thesis shows that the implementation of two control layers is possible, though it requires extensive tuning by the engineer. Under the original conditions of the pilot plant the polymerisation time can be decreased with around five minutes using the bilevel control structure compared to the off-line calculations in the preliminary project.

Sammendrag

Målet med denne masteroppgaven er å studere implementeringen av en on-line tonivås MPC (Model Predictive Control) applikasjon på en emulsjonskopolymerisasjon semi-batch reaktor. Tidligere arbeid har blitt utført med ett enkelt MPC nivå i tillegg til en studie av en off-line prediksjon uten tilbakekobling av prosessen.

MPC har blitt mer og mer vanlig å implementere i industrien og dette åpner for muligheten til å optimalisere kjemiske prosesser. Ulempen med MPC er at den krever en prosess-modell som nøyaktig beskriver oppførselen til anlegget. Dette er ofte tidkrevende arbeid. Den ulineære modellen som ble brukt i dette arbeidet ble utformet under EU-prosjektet COOPOL og ble brukt til å styre en pilotreaktor. Noen forskjeller mellom modellen og oppførselen til reaktoren ble observert.

Masteroppgaven gir en introduksjon til polymerer, emulsjonspolymerisasjon og de viktigste mekansimene i friradikal polymerisasjon. Dette er grunnlaget for likningene som er brukt i modellen, som er skrevet i programmeringsspråket C. En standard mal er utviklet av Cybernetica AS som fungerer bra sammen med Cyberneticas spesialiserte software til sanntidsstyring og optimalisering av ulineære prosesser. Oppgaven forklarer prinsippene bak MPC og forklarer hvorfor det er så vanskelig å styre en polymerisasjonsreaktor.

Alle prosessanlegg har et hierarisk reguleringssystem, fra langtids økonomiske aspekter til raske PID-kontrollere som sørger for at sikkerheten i anlegget blir overholdt. I denne oppgaven er det to nivåer med MPC-styring som blir studert, der det øverste laget har en lang tidshorisont og kalkulerer optimale temperatur- og føderateprofiler som det nederste laget skal følge. Målet er å undersøke om det er en fordel å bruke en tonivås løsning i stedet for ett tradisjonelt MPC lag.

Målet med oppgaven er å minimere polymerisasjonstiden til prosessen, ved å la reaktortemperaturen og fødestrømmen av monomer og initiator variere. Samtidig må den endelige produktkvaliteten være godkjent. Denne oppgaven undersøker hvordan prosessen kan reguleres ved å bruke tonivås MPC-styring. Resultatene er lovende. Oppgaven viser at et tonivås styresystem er mulig å implementere i prosessen, men at det kreves mye arbeid i tuning av lagene. Oppgaven viser i tillegg at under de opprinnelige forholdene til pilotreaktoren er polymerisasjonstiden 5 minutter kortere ved applikasjonen av tonivåssystement forhold til off-lineestimeringen i det tidligere prosjektet.

Preface

This master thesis was completed during the spring of 2017 as a compulsory part of the study program leading to a M.Sc. in Chemical Engineering at NTNU (the Norwegian University of Science and Technology). The work was realised in close collaboration with Cybernetica AS, who proposed the project. The project is an extension of the work done during a summer internship at Cybernetica in 2016, as well as a compulsory project in the M.Sc. of Chemical Engineering at NTNU the autumn of 2016.

I wish to express my gratitude to Cybernetica AS for this project, and the employees at Cybernetica for supporting my work and making my stay pleasant and interesting. I spent most of my time working in Cybernetica's offices in Fossegrenda, Trondheim, and it has been incredibly inspiring to work with people who are experts in their field and are always willing to provide valuable input and their thoughts on the project. A special thanks goes to my co-supervisors at Cybernetica, Peter Singstad and to Fredrik Gjertsen, for all the hours they have spent discussing issues that I have encountered along the way. I would also like to thank my supervisor at NTNU, Professor Magne Hillestad, for his support in the thesis. Finally I would like to acknowledge the valuable help and previous work that has been done by the contributors to the COOPOL project.

I would like to thank my fellow students for creating a creative and fun atmosphere in the office. Last but not least I would like to thank my parents, Kirsti and Frans Kjetså, for their love and support throughout the five years of university.

Ellen Kjetså,
Trondheim, June 2017

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abbreviations

COOPOL	Control and real time optimisation of intensive polymerisation processes
CTA	Chain Transfer Agent
CV	Controlled variable
DAE	Differential Algebraic Equations
DOF	Degrees of Freedom
DP	Degree of polymerisation (number of repeat units)
D-RTO	Dynamic Real Time Optimisation
DV	Disturbance variable
KF	Kalman Filter
LP	Linear Program
MIMO	Multiple input, multiple output
MHE	Moving Horizon Estimator
MPC	Model Predictive Control
MV	Manipulated variables
MWD	Molecular Weight Distribution
NLP	Nonlinear Program
OPC	Open Platform Communications Server
PDI	Polydispersity Index
PID	Proportional - Integral - Derivative controller
PVC	*Polyvinyl chloride:* widely produced plastic
QP	Quadratic Program
RECOBA	Cross-sectorial real-time sensing, advanced control and optimisation of batch processes saving energy and raw materials
RTO	Real Time Optimisation
SISO	Single input, single output
SQP	Sequential Quadratic Program

List of Symbols

A	m^2	Reactor surface area
C		Rate coefficient for radical termination
$c_i(x)$		Constraints (equality or inequality)
c_p	$JK^{-1}kg^{-1}$	Specific heat capacity
D_n		Dead polymer chain of length n
E	J	Energy
$\hat{E}$	Jkg^{-1}	Energy per unit mass
E_a	J	Activation energy
$f(x)$		Objective function
G	$Jmol^{-1}$	Gibb's free energy
H		Selection matrix to select the states that are controlled variables
H	$Jmol^{-1}$	Enthalpy
I_2		Initiator
$I*$		Initial radical after initiator decomposition
K	J	Kinetic energy
k	$mols^{-1}$	Radical exit frequency from particle
k_{CTA}		Reaction rate coefficient of radical to CTA
$k_{CTA,rev}$		Reaction rate coefficient of radical from CTA to polymer
k_d		Reaction rate coefficient of initiator decomposition
k_i		Reaction rate coefficient of free radical initiation
k_p		Reaction rate coefficient of propagation
k_p^0		Frequency factor
$k_{p,ij}$		Reaction rate coefficient of propagation between a polymer with end group i and monomer j
k_{tc}		Reaction rate coefficient of termination by combination
k_{td}		Reaction rate coefficient of termination by disproportionation
M		Monomer molecule
M_m	$gmol^{-1}$	Molecular weight of the monomer unit
$\bar{M}_m$	$gmol^{-1}$	Average molecular weight of the monomer units
M_n	$kgmol^{-1}$	Number average molecular weight
M_w	$kgmol^{-1}$	Weight average molecular weight
m	kg	Mass
$\dot{m}$	kgs^{-1}	Mass flow rate
N_A		Avogadro's number
N_P		Number of particles
N_0		Number of particles containing 0 radicals
N_1		Number of radicals containing 1 radical
n		Number of monomers in polymer chain

n_{Mi}		Number of monomer *i* in the reactor
$\bar{n}$		Average number of radicals per particle
P		Prediction horizon
P_0	Pa	Initial reactor pressure
P_n		Growing polymer chain of length n
$[P_n]$		Concentration of radicals
Q		Weighting matrix on distance to reference value
$\dot{Q}$	Js^{-1}	Heat flow
R	$Jmol^{-1}K^{-1}$	Universal gas constant
R_p		Rate of polymerisation per unit volume
$r_{p,ij}$		Reactivity ratio for propagation with monomer i or j
r_i^p		Reaction rate of monomer in the particle phase
r_i^w		Reaction rate of monomer in the aqueous phase
S_i		Probability of having a radical with end group i
$T\ K$		Temperature
T_J	K	Jacket temperature
T_R	K	Reactor temperature
t	s	Time
U	J	Internal energy
UA	$Wm^{-2}K^{-1}$	Heat transfer coefficient
u		Input, or manipulated variable
V	m^3	Reactor volume
V^p		Volume in polymer phase
V^w		Volume in aqueous phase
V_p	m^3	Volume of a particle
v		Disturbance variable
$\dot{v}_0$	m^3s^{-1}	Volumetric flow
W		Condensation product from condensation polymer
$\dot{W}$	W	Work in the system per unit time
x		Process states
$\hat{x}$		Predicted state
y		Measurement
z		Controlled variable
ϵ		Slack variable
μ		living polymerisation chain
λ		Dead polymerisation chain
ρ		Radical entry frequency into particle
ρ_{init}		Entry rate for the radicals that enter the particle directly from initiator decomposition
ρ_0	kgm^{-3}	Density
Φ	J	Potential energy
θ		Parameters

CHAPTER

ONE

INTRODUCTION

New simulation and optimisation technologies have had a huge effect on process industries the last decades. Powerful computers together with modern numerical methods facilitate the solution of complex chemical engineering problems. However, even with improved computational power, simulations and optimisation calculations will have no positive effect if not paired with accurate mathematical models representing the chemical process. Being able to control the process with model based techniques would significantly contribute to cost effectiveness of many production plants. Some chemical processes are easier to control and model than others. Processes with linear behaviour can quite easily be controlled using the latest optimisation technology, like Model Predictive Control (MPC) or can be controlled with simple PID loops[1]. Linear MPC uses empirical, linear models based on step-response behaviours in the plant, and it has been proven that this works very well. However, once the system is nonlinear, linear MPC is less effective, and the model's ability to accurately predict the future states is even more important. Polymer reactions are especially unpredictable and nonlinear, resulting in more work to build an efficient model (Bausa, 2007).

Polymers are found both everywhere, but is may best known as plastic. Plastics can be found almost anywhere, from buildings and construction, to packaging, transportation and electronic appliances. In 2008 the worldwide production of plastic was estimated to be 245 million metric tons (Statista, 2015). Plastics have very varying properties depending on their intended use. Common for plastics is that they are polymers made from organic materials, such as cellulose, natural gas, and of course, crude oil (PlasticsEurope, 1999).

Emulsion polymerisation creates a unique type of polymers, producing waterborne resins, often called latex. It is a free-radical polymerisation process where highly hydrophobic monomers are polymerised inside polymer particles by the addition of an emulsifier in the

[1] PID controller: A proportional-integral-derivative controller. A control loop feedback mechanism commonly used in industrial control systems.

process. Emulsion polymers have a wide range of applications, such as coatings, thermoplastics, paint, synthetic rubbers, etc. Emulsion polymerisation is a complex process to understand, because nucleation and stabilisation of the polymer is controlled by free radicals, in addition to other colloidal phenomena. Most commercial latex products are produced in semi-batch or continuous reactors because the reaction is exothermic and heat transfer is more easily controlled in these reactors. These reactors also allow for better flexibility and control over the end product quality.

Linear MPC is more common than nonlinear MPC in industrial applications (Qin and Badgwell, 2003). Nonlinear model predictive control (NMPC) has received much attention the last years because it provides a flexible solution for control of nonlinear multivariate systems. The NMPC will calculate a sequence of optimal control inputs over a time horizon subject to the process model and process constraints. The first input control move is applied to the plant, and the measurements from the plant are compared to the estimates of the model. A good process model will ensure stability and satisfactory performance of the NMPC.

This report will first give a process description, including theory on emulsion copolymerisation, the general kinetics and mechanisms of it, as well as details related to this specific process. An introduction to optimisation and the control of polymerisation processes will be given before a description of Cybernetica's software that has been applied. The following chapter will discuss the results of this study. The conclusion will elaborate on the significance of these results and the possibilities of further work.

This study is a continuation of the COOPOL and RECOBA EU-projects that have been ongoing since 2007. The objective for the COOPOL project was to

> develop a new process control approach, linking molecular level information and understanding of the reaction chemistry with real-time sensing, rigorous modelling based on first principles, subsequent model reduction and nonlinear model-predictive control (NMPC) with economic objectives, called dynamic real-time optimisation (DRTO) (COOPOL, 2007).

The original model for this process was developed during this project and tested on a pilot plant. The RECOBA project's objective is to use an on-line model predictive control of complex batch processes for the production of emulsion polymers, among other things.

A preliminary project was realised the autumn of 2016 as a part of the Chemical Engineering program at NTNU, see Kjetså (2016). This project was also in collaboration with Cybernetica. The aim of the preliminary project was to perform an off-line optimisation to determine the reference trajectory for the feed stream and the reactor temperature. This thesis is a continuation of the project.

1.1 Motivation

Frank Popoff, a former CEO of Dow Chemical, stated in 1996 that "Process modelling is the single technology that has had the biggest impact on our business in the last decade".

Process models are used to predict the behaviour of the process and be used in optimisation problems. Chemical processes are modelled dynamically using differential algebraic equations (DAEs) used to describe the dynamic behaviour of the system, like the mass and energy balances, while the algebraic equations describe physical and thermodynamic relations. Batch and semi-batch systems are difficult to model numerically as steady state is not reached, but the system changes over time. Furthermore, chemical processes, and especially polymerisations are typically nonlinear. To be able to improve performance and safety conditions of a batch reactor, a mathematical description of the kinetics and knowledge about the dynamic behaviour of the system is necessary. The development and validation of these dynamic models can be time-consuming and quite expensive. As a consequence, few industries invest in the developing of models and rely on the operators experience to adjust the process. Nyström (2007) provides two quotes that show a common way of thinking in the industry. "Make it work and don't worry about why", and "Don't change anything that is functioning, otherwise you will end up with problems". These attitudes originate from process engineers and operators who assume that nothing changes in the process. However, this is not the case in real life. Processes can get contaminated with chemical impurities from side reactions and unreacted initial charge. This is hard to avoid, and the contaminants will differ from batch to batch. Also, equipment gets old. Stops and disturbances cost tremendously, and compared to new investments, process optimisation will enhance production in a more cost effective way.

The polymer market is facing numerous challenges to meet a demand that is constantly changing and the pressure of cost reduction and new product development is high. To stay in business, the industry has to stay within strict product quality requirements and also introduce new products rapidly. Several of the problems encountered in the polymerisation reactors today are results of the complex kinetics and reaction mechanisms in polymerisation, and their strong nonlinearities. Moreover, many of the properties that are directly linked to the product quality are extremely difficult and/or costly to measure on-line, and will therefore only be measured at low frequency and with a significant time delay. This complicates product monitoring and control. The product quality can be defined by fundamental polymer properties such as the molecular weight distribution (MWD), composition, branching, cross-linking, etc., and, unfortunately, more than one reaction or process variable will affect these properties. Modelling of these polymerisation processes is of great industrial importance because it will allow speedy introduction of new products to the markets. In most cases the models are derived from the chemical and physical reactions of the polymerisation process to calculate reaction rates and other parameters (Yoon et al., 2004).

1.2 Scope of Work

The purpose of this report is to investigate the possibility of reducing the polymerisation time of a specific emulsion copolymerisation semi-batch reaction. Using a model that has previously been tested on a pilot plant, improvements are made on its parameters to make the model more accurate in its prediction of the temperature and value of M_n (number

average molecular weight). This study investigates the possibility of an on-line two-level MPC control of this process.

The objectives for the work in this thesis are listed below:

- Implementation of the two-level MPC control structure using Cybernetica's software.
- Deciding what each layer is controlling, and define the manipulated variables (MV's) and controlled variables (CV's), constraints and setpoints for each layer.
- Demonstrate how the two layers communicate and work together.
- Study the effect of different tuning and weighting in the layers on the polymerisation time.
- Demonstrate how the Extended Kalman Filter works to re-estimate chosen parameters when there is a mismatch between the process model and the plant model. The parameters are re-estimated based on reliable on-line measurements.

CHAPTER

TWO

FUNDAMENTALS OF POLYMERISATION

Today, synthetic polymers are widely used and are known under many different names, like plastic, rubber, resin and macromolecules. The polymer products have become essential in our everyday life, and that is why it is important to ensure efficient and economic production of polymers with the required polymer properties. The demand of the polymer product quality changes rapidly, and meeting these quality requirements is one of the biggest challenges in the polymer industry today. To understand the production of polymer, one must first understand what a polymer is. This chapter will give an introduction to polymers, how they are classified and then describe some of the different polymerisation reactions that exist.

2.1 Polymers

Polymers are found in a large variety and have shaped modern life. In nature they have existed for thousands of years: Cellulose, protein, silk, cotton and DNA strands are examples of natural polymers. Today, polymers are also produced synthetically as plastics, acrylics and glass. Polymer products can be found in food packaging, clothing, insulation, furniture, medical materials, electronic devices, information technology and to name some. Polymers are worth studying because of their properties that are so different from those of metals and other low-molecular weight materials.

Polymers are large molecules, macromolecules, with a high molecular weight. They are built up by structural units known as monomers, that form covalent bonds between themselves and create large chains that make up the polymer. For example, polyethylene is a long-chained polymer that is made up of the structural unit of ethylene ($-CH_2-CH_2-$) and is written as:

$$-CH_2CH_2CH_2- \text{ or } [-CH_2CH_2-]_n \tag{2.1.1}$$

where n represents the length of the chain, i.e. the number of monomers, in this case ethylene, that are chained together. The great versatility in the polymer's end-product qualities is due to the endless ways of the monomers can react together and create different structures of the polymers. The molecular characteristics of a polymer include the molecular weight distribution, chemical composition, branching, cross-linking and morphology.

The reaction where monomers react with another monomer or with a polymer chain is called polymerisation. The molecular characteristics of the polymer depend on the monomers used, the initiator, the emulsifier and also the process conditions, like what kind of reactor is used, monomer concentrations and temperatures.

2.1.1 Classification of Polymers

Polymers can be classified in several ways. One of the oldest methods of classification is by looking at how the polymer reacts to heat. In this system, there are two types of polymers: *Thermoplastics* and *thermosets*. Thermoplastics melt when heated and solidify when cooled. These polymers can be heated and cooled several times without changing the properties of the polymer. Thermosets, on the other hand, will melt the first time they are heated, and the consecutive times it is heated the polymer will degrade instead of melt.

An alternative approach to classifying polymers is based on their molecular structure, which divides polymers into three classes: (1) *Linear-chain polymers*, (2) *branched-chain polymers* and (3) *network* or *gel polymers*. These molecular structures are illustrated in Fig. 2.1.1. The molecular structure of the polymer depends on the functionality of the monomer that is used. For the monomers to react together and form chains, the monomer requires reactive double or triple bonds, or reactive functional groups in order to react and create polymers. The functionality of a monomer is determined by the number of these functional groups; Double bonds have a functionality of 2, while triple bonds have a functionality of 4. To form a polymer, the minimum monomer functionality is 2. Monomers with a functionality of 2 will always create linear polymer chains.

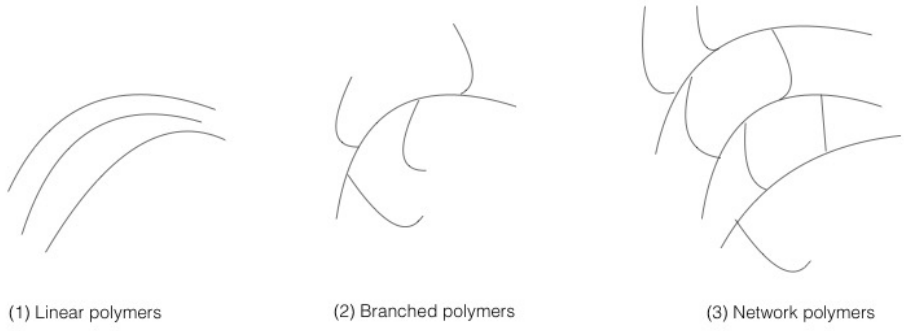

Figure 2.1.1: Illustration of the three different molecular structures of polymers: Linear, branched and network polymers.

In fact, all thermoplastics are linear polymers. In the linear chains, the repeat units are held together by strong covalent bonds, while weaker forces (secondary forces) hold the different molecules together. When the polymer is heated, the random motion of the molecules increases, and tries to overcome the secondary forces. When the secondary forces are overcome, the molecules are "free" and the polymer melts. The linear chains remain unchanged, as the covalent bonds are significantly stronger. This explains the thermoplastic nature of linear polymers (Kumar and Gupta, 2003).

Branched polymers are polymers with a linear backbone with branches that sprout from it at random places. To form the branches, the monomer must have the ability to grow in more than two directions, and so the functionality of the polymer must be greater than 2. Branched polymers are often formed at low conversion of monomers, and this implies that the branched polymer is of low molecular weight. Special techniques are applied to form branched polymers with high molecular weights. If monomers react up to large conversions, the polymers will turn into a *gel*, i.e. a three-dimensional structure with high viscosity. When a multifunctional monomer reacts, it will start out forming a linear chain before it branches, and when it reaches high conversion it will turn into a network (or gel) polymer. A network polymer will not dissolve in a solvent, but it will swell by incorporating molecules of the solvent into the free space areas of the network (Kumar and Gupta, 2003).

So far it has been assumed that the polymers are made up of just one monomer, i.e. one repetitive unit. These are called *homopolymers*. However, monomers can be made up of several types of monomers, creating *copolymers*. The distribution of the different monomers will have an impact on the polymer properties. If a copolymer is made up of two different monomers, A and B, and the monomers are randomly put together, it is called a random copolymer. If A and B alternate, it is called an alternate polymer, and if first all of monomer A reacts and creates a chain and then all of monomer B it is called a block polymer. These structures are illustrated in Eq. 2.1.2a- 2.1.2c. The different distributions are a result of the relative reactivity between the monomers. For a random distribution the reactivity between the monomers is equal, i.e. it is just as likely for a chain end A to react with monomer A as with monomer B. For a block distribution, it is much more likely for a chain end A to react with monomer A than with monomer B. Controlling the distribution of comonomers gives even more possibilities to control the properties of the polymer product.

$$ABAAAABBABABBBA \tag{2.1.2a}$$

$$ABABABABABABABA \tag{2.1.2b}$$

$$AAAAAAABBBBBBBB \tag{2.1.2c}$$

Polymers are very versatile products. Different monomers put together in different ways form all sorts of polymers. The end-product properties of the polymer are commonly described by the molecular weight distribution (MWD, see Section 3.4.5) and the copolymer distribution. The glass transition temperature T_g is a very important thermodynamic property for amorpheous polymers. It can be referred to as the melting point of amorpheous polymers. In the region above T_g the polymer is soft and rubbery, and in the region below T_g, the polymer is hard and brittle. The glass transition temperature is not very well

understood, however it is a useful quantity for two main reasons. It says something about the viscosity dependency of the temperature, and numerous other polymer properties can be correlated with T_g (Dimarzio and Gibbs, 1963).

2.2 Polymerisation

Polymerisation is the chemical reaction where polymers are synthesised from monomers by a multitude of reaction mechanisms. The monomers link together in large numbers (hundreds, thousands, tens of thousands), forming large chains and/or branched chains that result in large polymer particles. The macromolecular structure (e.g. molar mass, branching distribution, molecular weight distribution (MWD)) depends not only on the reaction mechanism, but also on on the chemical state of the monomers and the reactor configuration. Kiparissides (1996) emphasises the complex issue of the polymer product quality compared to the product quality of other shorter chain reactions. This is due to the fact that the polymer structure as well as the molecular properties of the polymer also influence the polymer's chemical, physical and thermal properties. A key to efficient production of high quality polymers is finding a good mathematical model to predict the polymer quality in terms of operating conditions, and for this an understanding of the underlying polymerisation reactions and mechanisms is required.

2.2.1 Polymerisation Reactions

Historically, two main categories of polymerisation were considered: *additive polymers* and *condensation polymers*. For additive polymers, a bifunctional monomer, M, is added to the growing polymer chain, P_n, without any part of the monomer being eliminated. The subscript n says how many monomers the chain consists of. It can be schematically represented as follows:

$$P_n + M \rightarrow P_{n+1} \tag{2.2.1}$$

This reaction step is usually very fast. Polyethylene and polypropylene are typical additive polymers.

Condensation polymers are formed by bi- or polyfunctional monomers in a reaction that eliminates a small part of the monomer. This reaction can for example take place between two growing polymer chains:

$$P_m + P_n \rightarrow P_{m+n} + W \tag{2.2.2}$$

where P_m and P_n are the polymer chains and *W*, typically a hydrogen molecule, H_2, or a water molecule, H_2O, is the condensation product. An example of a condensation polymerisation is polyesterification, and this is a reversible reaction (Kumar and Gupta, 2003).

However, as more studies were performed on polymerisation it became clear to the researchers that these two classes of reaction mechanisms were inconsistent, as some poly-

mer products could be prepared by both additive polymerisation and condensation polymerisation. The classes were redefined, and today, the two classes of polymerisation mechanisms are:

1. *Chain-growth polymerisation*: A more chemical name for additive polymerisation.

2. *Step-growth polymerisation*: Include condensation polymers, but also a reaction mechanism where no part of the monomer is eliminated.

In chain-growth polymerisation, monomers can only join active chains, and so polymerisation activity is dependent on an initiator or a catalyst. An active chain, or a "living" chain is a chain that has a reactive end monomer. Research shows that chain-growth polymerisation proceeds very quickly, before it suddenly stops and this is because the monomers only react with living chains. Consequently, the reaction mass consists mainly of monomer, dead polymer chains and only a few growing polymer chains. There are several types of chain-growth polymerisation, including coordination polymerisation, free-radical polymerisation, anionic polymerisation and cationic polymerisation. Section 2.2.2 will give an introduction to free-radical polymerisation.

Step-growth polymerisation proceeds through a reaction of the functional groups of the reactants. The monomers react with each other continuously, and not only with active chains, creating low molecular weight polymer particles and continue to react to create continuously growing chains. This way the present monomer is converted very rapidly to polymer particles. The chemical reactions in step-growth polymerisation include esterification, amidation, and transesterification among others. Chain-growth polymers and step-growth polymers have completely different molecular weight distributions, as will be discussed in Section 3.4.5 (Kumar and Gupta, 2003).

2.2.2 Free-radical Polymerisation

This section will give an introduction to the basic concepts of free-radical polymerisation. Free-radical polymerisation is a type of chain-growth polymerisation, where the polymer grows exclusively by reaction of monomer with a reactive end-group on the growing chain. The monomer is called a *repeat unit*, and the *degree of polymerisation (DP)* is the number of repeat units in the monomer chain. At least three different reaction types take place simultaneously in the reactor that are typical for free-radical polymerisation: Initiation, propagation and termination. To start the growth of the chain, an initial reaction has to take place, usually between a monomer and an *initiator* to create free radicals. A free radical is a molecule with an unpaired electron in its outer shell, making it very reactive. The polymer chain propagates by adding a new monomer to the reactive site, known as the *active centre*. Upon every addition of monomer, the active centre is transferred to the end of the chain. At termination the chains stop to grow (Lovell and El-Aasser, 1997). The next paragraphs will go into more detail on these reactions, and the complete list of reactions in free-radical polymerisation is given in Appendix A.1.

Initiation

The initiation step involves creating the free-radical active centre, which usually takes place in two steps. First the decomposition of the initiator to form two free radicals, and then these radicals react with a monomer to start the polymer chain. Free radicals can be formed in two principle ways: (1) Homolytic scission of a single bond (homolysis), and (2) single electron transfer to or from an ion or molecule (redox reaction). Homolysis is simply achieved by applying heat to the intiator. At a convenient temperature of around 50-100 °C, many compounds undergo thermolysis. Homolysis can also take place by adding radiation. Redox reactions are performed when it is necessary to keep the polymerisation reaction at low temperatures. The details of these two mechanisms can be found in Appendix B.

Eq. 2.2.3a and Eq. 2.2.3b show the process of initiation decomposition and free radical reaction with monomer, respectively.

$$I_2 \xrightarrow{k_d} 2I* \tag{2.2.3a}$$

$$I* + M \xrightarrow{k_i} P_1 \tag{2.2.3b}$$

where I_2 is the initiator, $I*$ the primary free radical, M is the monomer and P_1 the growing (living) polymer chain with one repeat unit. k_d and k_i are the reaction rates for the decomposition and free radical initiation, respectively.

The free radicals that are formed from initiator decomposition are known as *primary free radicals*, and these have to react with monomers to commence the polymerisation. However, not all free radicals from initiator decomposition react with monomers. Others are lost in side reactions.

Propagation

In the propagation reactions, monomers react rapidly with the active centre of the polymer chains. According to Lovell and El-Aasser (1997), the time required to add a new monomer is typically in the order of a millisecond, and so several thousand additions may take place within a few seconds. Eq. 2.2.4 shows the typical propagation reaction between an active chain, P_n of n repeat units, and the monomer M.

$$P_n + M \xrightarrow{k_p} P_{n+1} \tag{2.2.4}$$

where k_p is the rate of propagation. It is commonly assumed that the propagation rate is independent of the length n of the polymer chains.

Termination

There are two dominant termination reactions: Termination by combination and disproportionation, shown in Eqs. 2.2.5a and 2.2.5b, respectively, and they both take place between

two growing polymer chains. In termination by combination the two growing polymer chains are coupled together, creating one longer single polymer molecule. In disproportionation, the free radical is transferred from one chain to the other, resulting in two dead polymer chains. In this case there is an initiator fraction only in one end of the polymer molecule as opposed to from combination where there will be an initiator fraction on both chain ends. Generally both these reactions take place in polymerisation process but to different extent, depending on the monomers and the polymerisation conditions. Termination by combination is described in Eqs. 2.2.5a and termination by disproportionation is described in 2.2.5b:

$$P_n + P_m \xrightarrow{k_{tc}} D_{n+m} \tag{2.2.5a}$$

$$P_n + P_m \xrightarrow{k_{td}} D_n + D_m \tag{2.2.5b}$$

where P_n and P_m are growing polymer chains and D_n are dead polymer chains. k_{tc} and k_{td} are the rates of termination by combination and disproportionation, respectively.

Chain transfer

In addition to combination and disproportionation it is possible for the growing chain to terminate in other ways as well, and these reactions are known as chain transfer reactions. There may be *chain transfer agents (CTA)* in the reaction mixture, that capture the free radical. This is written in Eq. 2.2.6a. This is a reversible action, so the free radical CTA can activate a dead polymer molecule, see Eq. 2.2.6b. Even though a polymer chain is terminated prematurely, the concentration of actively propagating chains will remain the same.

$$P_n + CTA \xrightarrow{k_{CTA}} D_n + CTA* \tag{2.2.6a}$$

$$CTA* + D_n \xrightarrow{k_{CTA,rev}} P_n + CTA \tag{2.2.6b}$$

The chain transfer may be both intra- and intermolecular. In intramolecular chain transfer the free radical jumps from the end of the polymer to a monomer unit in the chain. This will create a branch on the polymer chain, and does not affect the number of radicals in the particle. This reaction is known as *back-biting* and mainly produces short-chain branches. The total molar mass of the polymer is not affected on this case, however, the polymer skeletal structure is of great importance to the polymer properties. In intermolecular chain transfer, the free radical is transferred to a different polymer molecule. This results in long-chain branches, and may lead to premature termination of the growth of one propagation chain, and reactivation of a dead chain that will continue to grow. A consequence of intermolecular chain transfer is that the molecular weight distribution of the polymer broadens. Also, the changes in the skeletal structure of the polymer and the molecular weight have major effects on the polymer product properties (Lovell and El-Aasser, 1997).

Free-radical Copolymerisation

Most free-radical polymerisations are in fact copolymerisation reactions. A polymer that is formed by simultaneous polymerisation of two or more different types of monomers is called a copolymer. However, including a second monomer highly complicates the reaction kinetics, and bring in different requirements. One of the main issues is understanding the different monomer reactivities and how these affect the polymer composition and sequence distribution. The details of the copolymerisation reactions are discussed in Section 3.4.2.

CHAPTER

THREE

EMULSION COPOLYMERISATION

Emulsion polymerisation is a free-radical chain polymerisation where a monomer or several monomers is polymerised in the presence of an aqueous solution of a surfactant/emulsifier to form a product, which is generally called a latex. A latex is defined as a colloidal dispersion of polymer particles in an aqueous medium. The main ingredients for an emulsion polymerisation include water, monomer, emulsifier, an aqueous initiator and chain transfer agents. It is a complicated process to describe, as the kinetic reactions, growth and stabilisation is determined by free-radical reactions in combination with colloidal phenomena.

Since its introduction on an industrial scale in the mid-1930's, emulsion polymerisation has developed into a widely used process for the production of synthetic latexes. Today, millions of tons of synthetic polymer latexes are prepared by emulsion polymerisation that can be used for a wide variety of applications (Lovell and El-Aasser, 1997). Among the uses of the latexes are synthetic rubber, latex paints, barrier coatings, adhesives and carpet backing. The major developments in emulsion polymerisation started after World War II as a result of intensive collaborative efforts between academia, industry and government laboratories. Since then, numerous patents and papers have been produced every year dealing with various aspects of emulsion polymerisation.

The aim of this chapter is to explore the different polymerisation methods before focusing on emulsion copolymerisation. The choice of reactor will be discussed, before a more thorough introduction to the reaction kinetics and mechanisms that characterise emulsion polymerisation.

3.1 Polymerisation Techniques

The different polymerisation reactions mentioned in the previous section can be implemented in several ways: Bulk polymerisation, solution polymerisation, gas-phase polymerisation, suspension polymerisation and emulsion polymerisation, and these are described in Asua (2008). In the following paragraphs the advantages and disadvantages of the different polymerisation methods will be presented.

In bulk polymerisation, the only components are the monomer and initiator. If the polymer is soluble in the monomer, the reaction mixture remains homogeneous throughout the process. If the polymer is not soluble in the monomer, phase separation occurs leading to a multiphase material. Often, as is the case in polymerisation of PVC (polyvinyl chloride, a widely used plastic), the polymer precipitates as it is formed, yielding a polymer slurry in their own monomer. The main advantage of bulk polymerisation is that a very pure polymer is produced at a high production rate per unit volume of the reactor. The main drawback is that removal of the heat of reaction is a challenge, since the high concentration of polymer in the mixture makes it very viscous. Controlling the temperature is proven to be more difficult if the reaction is a free-radical polymerisation rather than a step-growth polymerisation, since higher molecular weights are achieved by free-radical polymerisation, and therefore the viscosity in this reaction mixture is higher (Asua, 2008).

It is easier to achieve sufficient thermal control of the reactor if the polymerisation takes place in a solution. The solvent lowers the concentration of monomer, making the reaction mixture less viscous, and the heat creation per unit volume in the reactor decreases. The drawback in this case is if the solvent is environmentally unfriendly, which complicates the solvent recovery (Asua, 2008).

Suspension polymerisation gives good thermal control and avoids using a solvent. Here, drops of monomer containing initiator are suspended in water. Each of the drops will then function as small bulk polymerisations. The internal viscosity of the drop will increase with the polymerisation, while the suspension will remain at a low viscosity, keeping a sufficient heat transfer. The suspension stability and the particle size are controlled by the agitation, as well as the type and concentration of the suspension agents that are used. The polymer products typically have a diameter from 10μm to 5 mm. These products will contain suspension agents, and even though it is possible to remove some of it, there will inevitably be some left in the final product. In suspension polymerisation, only free-radical polymerisation is implemented (Asua, 2008).

Emulsion polymerisation is a polymerisation technique that leads to finely dispersed polymer particles in a continuous medium. The particles have a diameter of around 80-500 nm, and the continuous medium is often water. The product is frequently called latex. The polymerisation is an emulsion since one or more of the monomers used are not soluble in the continuous medium (water). Only free-radical polymerisation has been implemented in emulsion polymerisation. The reaction mixture includes monomer, emulsifier, water and water-soluble initiator. By agitation, the monomer droplets are dispersed, and there is enough emulsifier to surround these droplets to create a large amount of micelles. Radicals form by decomposition of the initiator in the aqueous phase, and react to form short

chains, *oligomers* in the aqueous phase. The oligomers migrate to the micelles, forming polymer particles, and the growth of these particles lead to the final latex product. The monomer micelles function as a sort of reservoir for monomer. The particle size of the latex is not determined by the size of the monomer droplets but by the size of the polymer droplets formed. It is easier to control the temperature of these reactions than for bulk polymerisation. However, the modest viscosity of the reaction medium and the presence of water that has a high heat capacity is counteracted by the fast polymerisation rate (Asua, 2008).

Water is the main ingredient in both suspension and emulsion polymerisation. It acts to maintain low viscosity in the reaction medium, and is therefore important for good heat transfer. Furthermore, it acts as a medium to transfer monomer from droplets to particles. The emulsifier also has two functions: it provides a site for particle nucleation and colloidal stability for growing polymer particles as they are adsorbed to the water-polymer interface.

3.2 Choice of Reactor

When creating a new product, the choice of reactor is of great importance and will influence the production. Operation is either continuous or by batch, as a stirred tank reactor or tubular reactor, and the difference between these reactors must be discussed to understand how they influence the polymer properties. The different stirred tank reactors are illustrated in Fig. 3.2.1. Continuous reactors are composed of a feed stream of raw material into the reactor and an exit stream of products that is removed on a continuous basis. When the stream in equals the stream out an approximate steady state is achieved in the reactor, meaning that the conditions in the reactor are consistent over time. In batch reac-

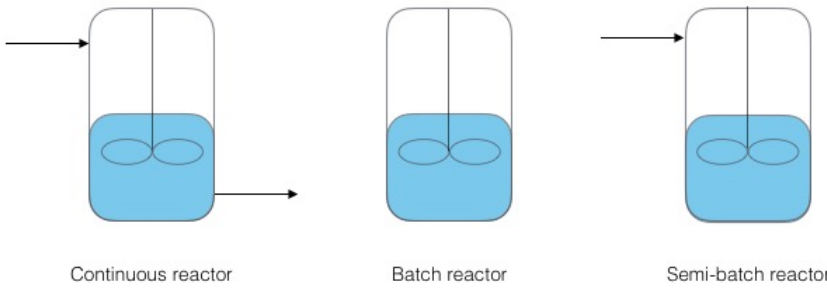

Figure 3.2.1: Illustration of the three different stirred tank reactors: continuous, batch and semi-batch reactors.

tors, all of the raw material is put into the reactor before the process starts, and product is not removed until the process is finished. Hence, the reactor conditions change over time. Batch reactors normally follow a predefined recipe to create a specific product. These recipes are based on heuristics and experience. A semi-batch reactor is a batch reactor

that is continuous in some parts. Typically, it allows for feeding more raw material into the reactor throughout the process. An example of a semi-batch reactor is when a gas of limited solubility is fed into the batch as it is gradually used up in the reaction. This is done to maintain the concentration gradient of the gas (Nyström, 2007).

Batch reactors are popular in the industry because of their versatility and flexibility due to the duration of the chemical reactions and the great number of chemical reactions they can process. However, several drawbacks to the batch reactor, especially when used for polymerisation. must be accounted for:

- They are generally less safe, for both people operating them and the environment.
- The control of polymer properties is impractical.
- The productivity is low, due to loading, emptying and cleaning of the reactor.
- Because all the monomer is loaded into the reactor at the initial stage, it is more challenging to control the temperature of the reactor.
- No batch will be the same due to unreproducible particle nucleation, jeopardising product consistency. Seeded emulsion polymerisation may be employed to avoid this problem.

In the industry, batch and semi-batch reactors are generally used to produce fine chemicals, specialities, polymers and other high value products. Batch reactors are typically used when the volumes are small, and the raw material and/or products are expensive. The reaction may be slow, as the down time for filling, emptying and cleaning the reactor may be around one hour. Semi-batch reactors are often preferred to batch reactors because of their flexibility. An initial amount of raw material (initial charge) can be fed into the reactor, and then the rest is added to the reactor over time. For a polymerisation, both the temperature of the reactor and the polymer quality can be controlled by varying the composition and amount of the initial charge, as well as the composition and flow rates of the feed. Using a semi-batch reactor in polymerisation processes allows for tailoring the polymer properties, like particle size distribution, copolymer composition, polymer architecture, morphology and molecular weight distribution (Leiza and Meuldijk, 2013).

3.3 Emulsion Polymerisation Process

An important feature with emulsion polymerisation is the heterogeneity from beginning to end of the process. It comprises water, an initiator (usually water soluble), a non-water soluble monomer and a colloidal stabiliser (surfactant/emulsifier). The polymerisation itself takes place inside monomer-swollen polymer particles, which are either formed at the start of the polymerisation, or added initially (seeded polymerisation). It is a normal misconception that the polymer particles only contain one polymer chain, when in fact there are *many* polymer chains per particle. In fact, the term "emulsion polymerisation" is a misnomer. This has historical reasons, as the original process was developed to polymerise

emulsion droplets, which in fact does not occur. The initial emulsion is not thermodynamically stable, though the final product is both colloidal and thermodynamically stable (Van Herk and Gilbert, 2013). As stated previously, semi-batch reactors are often chosen for emulsion polymerisations. In this case, the reactor is charged with an initial charge with a fraction of the formulation, and the rest of the formulation is added over time.

An *ab initio* emulsion polymerisation is the emulsification of one or more monomers in a continuous aqueous phase and stabilisation of the droplets by surfactant. In a seeded emulsion polymerisation, the reactor is initially charged with a previously prepared seed latex. The polymerisation takes place in the polymer particles that are swollen with monomer, stabilised by the surfactant and dispersed in the aqueous phase. The polymer product, the latex, is the colloidal dispersion of polymer particles in an aqueous phase (Van Herk and Gilbert, 2013). The main monomer present is non-soluble in water, and is essential for the final polymer product characteristics. If more than one monomer is present, the ratio between "hard" monomers to "soft" monomers (i.e. high or low T_g) is chosen to achieve the T_g for the final latex. Other minor monomers are added to provide some special characteristics, and chain transfer agents (CTAs) are added to control the chain architecture and the MWD of the polymer.

The polymerisation process can be divided into three intervals. The first interval is the initiation stage, the second interval is where propagation reaction dominates, and the third interval is the termination stage, when all the monomer is reacted, see Fig. 3.3.1.

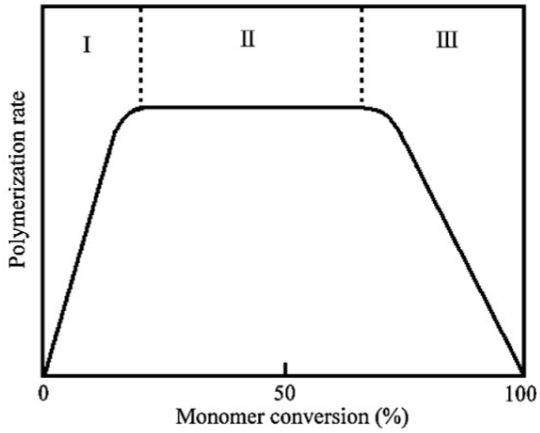

Figure 3.3.1: Illustration of the different intervals of a polymerisation reaction (Chern, 2006). Interval I is the initial stage with nucleation, in Interval II the number of droplets is assumed to be constant, and in Interval III the termination reactions dominate.

Interval I is the initial stage where particle nucleation takes place. Particle nucleation is a complex process, which is described in detail in Section 3.4.1.

In Interval II, the system is composed both of monomer droplets and polymer particles, and the number of polymer particles is assumed to be constant. The monomer that is

consumed in the polymer particle is replaced by monomer that diffuses into the particle from the monomer droplets, which function as a "reservoir" for monomers. The monomer will be partitioned between the different phases at a thermodynamical equilibrium (Asua, 2008). However, semi-batch reactions are often run under monomer starved conditions, which is more thoroughly discussed in Section 5.3.1. In this interval, the propagation reaction is dominant, and the polymer particles grow in size. The disappearance of the monomer droplets marks the end of Interval II. The conversion of monomer at the time Interval II ends depends on the swelling of monomer in the polymer particles. The higher the concentration of monomer in the polymer particle, the faster the monomer droplets disappear. So quite a lot of the monomer can polymerise during Interval III as well, though in this interval the monomer concentration in the polymer particles decreases continuously.

3.4 Mechanisms, Kinetics and Thermodynamics

The emulsion polymerisation is complex from a modelling point of view. The three different phases and compartments offers the possibility of preparing polymers with unique properties. As discussed in the previous section, the polymerisation in an emulsion polymerisation occurs in the polymer particles. The initiator is water soluble, and will initiate radicals in the aqueous phase before these are absorbed into the polymer particles. Emulsion polymerisation is operated by the free-radical polymerisation mechanisms described in Section 2.2.2. This section will discuss in more depth the mathematical equations that contribute to the modelling of the most important issues of emulsion polymerisation.

Harkins (1945) and Smith and Ewart (1948) developed the first models for emulsion polymerisation, which are still used as a basis for the models built today. Harkins was the first to formulate a qualitative theory that explained some experimental observations that served as a basis for understanding the mechanisms of emulsion polymerisation for many years. Harkins explained how the initiator decomposes in the aqueous phase and enters the micelles to form polymer particles. The polymer particles grow by absorbing monomer from the monomer droplets, thus depleting the monomer droplets. When the monomer droplets are gone, the polymer particles will continue to grow until all the monomer is consumed. Harkins also explained that only one radical would be present at a time in the polymer particle.

Smith and Ewart were the first group to quantitatively express Harkins' postulation for the first two intervals in the emulsion polymerisation (Dimitratos et al., 1994). The theory explains that the propagation rate in Interval I increases due to the increasing number of newly formed particles, while the propagation rate will remain more constant in Interval II because the number of particles remains constant (Gao and Penlidis, 2002). The Smith-Ewart theory is built on several assumptions, that will be discussed more in depth in the following sections. It states that one radical diffuses into a particle at a time, and that termination occurs when a second radical enters the particle. The radical absorption happens every 1-100 seconds. Furthermore, the Smith-Ewart model explains why polymerisation takes place in the particles rather than the monomer droplets: Large polymer particles have a lower surface area to volume ratio than the smaller monomer droplets. The probability of

a radical entering a monomer droplet is therefore significantly smaller than the probability of a radical entering a particle. Interval III was not originally a part of the Smith-Ewart theory. Following this theory, many contributions have been made to understand the modeling of emulsion polymerisation mechanisms (Dimitratos et al., 1994).

3.4.1 Particle Nucleation and Number of Particles

The nucleation stage is known as Interval I in the Smith-Ewart theory (see Fig. 3.3.1) since this is where the particle number is changing. Water, initiator, monomer(s) and surfactant (emulsifier) is added to the reactor. Since the monomer is not water-soluble it will lump together, and the emulsifier absorbs onto the surface of the monomer droplets, stabilising them. Ionic surfactants stabilise the droplets by electrostatic repulsion, while non-ionic surfactants provide steric stabilisation. In most formulations, the amount of surfactant exceeds the amount that is necessary to stabilise the monomer droplets. The excess surfactant forms micelles that may also contain monomer, with diameters typically in the order of 50-150 Å. The initiators are most often water-soluble, and form radicals in the aqueous phase. The radicals are often too hydrophilic to enter the non-aqueous phase directly, and so they react with monomer dispersed in the aqueous phase forming oligomers (very short polymerisation chains). The monomer in the aqueous phase is just a small fraction of the total monomer concentration. When the initial radicals have reacted with a few monomers the chains they are called oligomers. When the oligomers are long enough, they become so hydrophobic that they can enter the organic phase of the system, into the micelles. Here, there is more monomer present, and the polymer chains grow rapidly. The micelles are now considered to be polymer particles. This process is illustrated in Fig. 3.4.1.

The process when polymer particles are formed by oligomers entering micelles is called *heterogenuous nucleation*. A different way of creating polymer particles is by *homogeneous nucleation*, which takes place when the oligomers grow too long in the aqueous phase so that they precipitate. Heterogeneous and homogeneous nucleation may take place in the same system. For both kinds of nucleation, the polymer particles are very small in the beginning, and have an enormous increase in surface area during particle growth. They are kept stable by the continuous addition of emulsifier throughout the batch. Thermal initiators are used when the reaction is carried out at elevated temperatures (75-90 °C) while redox initiators are chosen for a reaction at lower temperatures.

As previously stated, radicals form in the aqueous phase where they react with dissolved monomers that form oligomers. These oligomers may:

1. enter the polymer particles
2. enter into the micelles (heterogeneous nucleation)
3. propagate in the aqueous phase until they are too long and precipitate (homogeneous nucleation)
4. terminate with other radicals in the aqueous phase

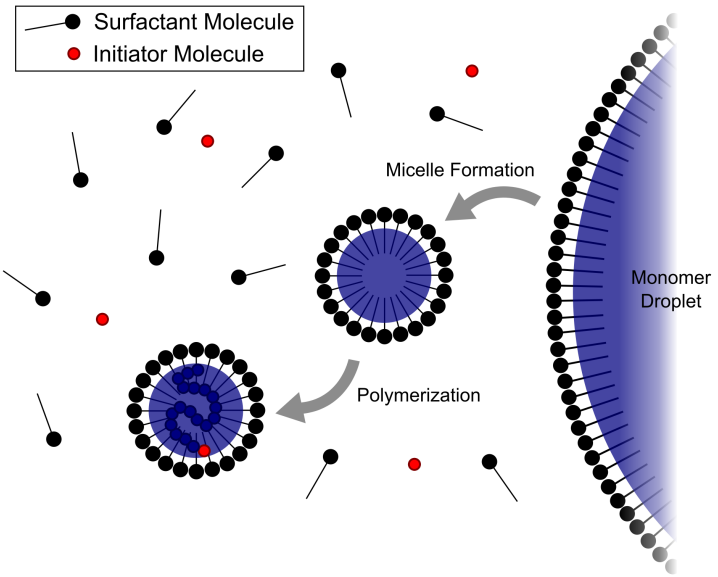

Figure 3.4.1: Illustration of the different phases in an emulsion polymerisation. The concentration of surfactant is above the critical micelle concentration and forms micelles. The monomer is hydrophobe and the excess monomer is stocked in monomer droplets. Initiation takes place in the aqueous phase, and polymerisation takes place inside a micelle.

The probability for these events to happen depend on the conditions of the system. Each little polymer particle acts like an ordinary free-radical bulk polymerisation.

Particle nucleation in emulsion polymerisation is complex and has been extensively researched over the years. Nucleation is a result of homogeneous or heterogeneous nucleation, as described previously. The number of particles that are created is very spontaneous and difficult to control. It is difficult to control because the particle number depends on the mechanism of particle formation, and it is important to control because it directly affects the reaction rate (Dimitratos et al., 1994). Furthermore, to be able to reproduce the results, the initial state is important to be similar from batch to batch. It is therefore very common to use a *seeded nucleation*, where the number of particle is predetermined (Van Herk and Gilbert, 2013). In a seeded polymerisation, the initial charge in the reactor contains a previously synthesised latex (seed), so that the nucleation of new particles is minimised and leads to better reproducibility (Asua, 2008).

3.4.2 Particle Growth

Particle growth occurs through all three intervals. The kinetics are mainly controlled by the distributions and exchange of radicals over the various phases, yielding a complex mechanism that must not be oversimplified (Van Herk and Gilbert, 2013). Several mod-

els are proposed in literature. For a homogenous batch free radical polymerisation the polymerisation rate equation is:

$$R_p = -\frac{d[M]}{dt} = k_p[M][P_n] \tag{3.4.1}$$

where R_p is the rate of polymerisation per unit volume, $[M]$ the monomer concentration, k_p the reaction rate coefficient for propagation and $[P_n]$ the radical concentration. For an emulsion polymerisation particle growth kinetics are controlled by the distribution and exchange of radicals over the various phases. One must take into account that polymerisation takes place inside the latex particles, and the polymerisation rate equation becomes:

$$R_p = k_p[M]_p \bar{n} \frac{N_p}{N_A V_p} \tag{3.4.2}$$

where $\bar{n}$ is the average number of radicals per particle, $[M]_p$ is the monomer concentration in the particle, N_p is the number of particles and V_p is the volume of the monomer-swollen polymerisation particle (Van Herk and Gilbert, 2013). It is important to understand that for an emulsion polymerisation the propagation rate does not depend on the overall concentration of monomer, but on the concentration of monomer in the polymer particle. The amount of monomer that can swell into a polymer particle depends on the reactor conditions and the monomer used. N_p will change throughout Interval I (unless the polymerisation is seeded). In Interval II the number of particles as well as the concentration of monomer in the particles is approximately constant. In Interval III the number of particles is still constant, while the concentration of monomer in the particles decrease. The propagation rate is also dependent on the average number of radicals per particle, $\bar{n}$. The work of Smith and Ewart (1948) set the ground rules, though there have been significant advances since then, as discussed in Section 3.4.3 (Van Herk and Gilbert, 2013).

Copolymerisation

The consumption of monomer is based on the rate of polymerisation. The equation for a bulk homopolymerisation is given in Eq. 3.4.1. However, if there are two different kinds of monomer, there are also two different end groups possible to the polymer chain, i.e. there are two kinds of radicals. So instead of writing the propagation rate as a function of the concentration of radicals, one writes it as a function of the concentration of radicals times the probability of having an end group i. The propagation rate of monomer i is then:

$$R_p = \sum_{i=1}^{j} k_{p,ij}[M_i]S_i[R] \tag{3.4.3}$$

with $k_{p,ij}$ the propagation coefficient for a reaction between end group i and monomer j, $[M_i]$ is the concentration of monomer i, $[R]$ is the concentration of radicals and S_i is the probability of having a radical with the end group i. The probability S_i is defined by:

$$S_i = \frac{[P_j]}{\sum_{i=1}^{j}[P_i]} \tag{3.4.4}$$

where $[P_j]$ is the concentration of radicals with j as an end group. Eq. 3.4.4 can, using a quasi steady state approximation, be written as system for two monomers:

$$S_1 = \frac{k_{p,12}[M_1]}{k_{p,12}[M_1] + k_{p,21}[M_2]} \tag{3.4.5a}$$

$$S_2 = \frac{k_{p,21}[M_2]}{k_{p,21}[M_2] + k_{p,12}[M_1]} = 1 - S_1 \tag{3.4.5b}$$

where $[M_i], i \in \{1, 2\}$ is the concentration of monomer i. The same concept can be applied when the system contains more than two different monomers.

Reaction Rates

The reaction rate depends on the concentration of the reactants, and is also dependent on the reaction rate constants. This is the case for all reactions: Initiation, termination, as well as the polymerisation reaction. All the rate constants follow Arrhenius' law:

$$k_p = k_p^0 \text{exp}(-\frac{E_a}{RT}) \tag{3.4.6}$$

in which k_p^0 denotes the frequency factor, and E_a the activation energy. This equation clearly states that the reaction rate is temperature dependent. k_p is the propagation rate coefficient for a homopolymerisation. For the simplest copolymerisation, consisting of only two monomers, the active chains may react with both types of monomers, as discussed above. The different copolymer reactions are listed below.

$$\sim M_1 + M_1 \xrightarrow{k_{p,11}} \sim M_1M_1 \tag{3.4.7a}$$

$$\sim M_1 + M_2 \xrightarrow{k_{p,12}} \sim M_1M_2 \tag{3.4.7b}$$

$$\sim M_2 + M_1 \xrightarrow{k_{p,21}} \sim M_2M_1 \tag{3.4.7c}$$

$$\sim M_2 + M_2 \xrightarrow{k_{p,22}} \sim M_2M_2 \tag{3.4.7d}$$

where $\sim M_i, i \in \{1, 2\}$ represents a radical with the chain end of monomer M_i, and $k_{p,ij}, i, j \in \{1, 2\}$ is the respective propagation rate coefficient. They are determined by the relation of the reactivity ratio, $r_{p,ij}$:

$$r_{p,ij} = \frac{k_{p,ii}}{k_{p,ij}}, \quad i, j \in \{1, 2\} \tag{3.4.8}$$

The reactivity ratio is assumed to be independent of temperature, and indicates which reaction is favoured. In the Eqs. 3.4.7, only the last monomer on the chain end is considered to affect the propagation rate, and this type of copolymerisation modelling is called the *terminal model*. The terminal model differs from the other copolymerisation model, the *penultimate model*, which considers the two last monomers on the chain end. This leads to an even more complex model.

3.4.3 Average Number of Radicals per Particle

In free-radical polymerisation, the rates are controlled by the processes of decomposition of initiator, propagation, chain transfer and termination. These processes occur in emulsion polymerisation particles as well, however the kinetics are quite different from those in a bulk polymerisation. The system is treated differently depending on the number of radicals in the polymer particle. The termination rate will be different in the polymerisation particles than it would have been for a bulk polymerisation with a similar radical concentration. Furthermore, emulsion polymerisation introduces two new mechanisms that differ from the bulk of suspension polymerisations: radical entry and exit of the polymer particle.

The polymer particles are very small and large in number, and in the reaction mixture they create a spatial separation between the growing polymer chains. Termination will only occur when two radicals meet, and the number of radicals in each particle is important to know in order to determine the termination rate. A termination reaction will decrease the number of radicals, which again decreases the termination rate. A smaller termination rate will result in longer radicals. Smith and Ewart (1948) were the first to describe three different cases in the kinetics of emulsion copolymerisation, depending on the average number of radicals per particle. $\bar{n}$ is the average number of radicals per particle.

$\bar{n} = 0.5$: There is either one or no radicals present in the particles. This is known as *zero-one kinetics*. When a second radical enters the polymer particle, this leads to instantaneous termination, either by combination or disproportionation.

$\bar{n} < 0.5$: In this case the decomposition of initiator is too slow compared to the number of polymer particles, and so there is not enough radicals to enter the particles at a sufficient speed. As a result, the probability of termination is low, leading to high molecular weights at slow reaction rates. Since the termination rate is low, the probability of chain transfer or radical desorption is higher.

$\bar{n} >> 0.5$: When the decomposition and concentration of initiator is high compared to the number of particles, the number of radicals per particle exceeds 0.5. This increases the termination rate. Consequently, the polymerisation rates and molecular weights are more similar to those of bulk polymerisations. Therefore, this third case is denoted *pseudo bulk*.

Using the zero-one or the pseudo bulk systems is a method to avoid the problem of the termination rate being dependent on the length of the polymerisation length. The zero-one system is usually applied for small particles. The original Smith-Ewart formulation defines N_0 and N_1 as the number of particles containing 0 and 1 radicals, respectively. These are normalised such that $N_0 + N_1 = 1$, and so for a zero-one system, $\bar{n} = N_1$. The equations describing the radical population takes into account radical entry (with entry frequency ρ), and radical exit (with exit frequency k), and gives the number of entering and exiting

radicals per particle per unit time. These kinetic equations are:

$$\frac{dN_0}{dt} = -\rho N_0 + (\rho + k)N_1 \tag{3.4.9a}$$

$$\frac{dN_1}{dt} = \rho N_0 - (\rho + k)N_1 \tag{3.4.9b}$$

These equations are derived by noting that the entry of one radical into a particle containing a radical, turns this particle into a particle containing no radicals because of the assumption of instantaneous termination. The entry of a radical into a particle containing no radicals, turns this particle into a particle containing one propagating radical chain.

To further develop these equations, the entry of radicals can be derived from the initiator and radicals that have exited other particles. Since radical exit rate is $k\bar{n}$ and an exit leads to a re-entry, and since $\bar{n} = N_1$ in a zero-one system, Eqs. 3.4.9 can be rewritten as:

$$\frac{d\bar{n}}{dt} = \rho_{init}(1 - 2\bar{n}) - 2k\bar{n}^2 \tag{3.4.10}$$

where ρ_{init} is the entry rate for the radicals that enter the particle directly from initiator decomposition.

The zero-one system is useful to interpret and predict data as it only contains two rate parameters, ρ and k.

In their paper, Li and Brooks (1993) discuss some further alterations and improvements to Eq. 3.4.10. The problem with Smith and Lewis' formulation is that it is difficult to obtain a general explicit analytical solution by exact procedures. Li proposes an approximation that makes the equation easier to solve. The full derivation of the equations can be found in Li and Brooks (1993):

$$\frac{d\bar{n}}{dt} = \rho - k\bar{n} + fC\bar{n}^2 \tag{3.4.11a}$$

$$\bar{n} = \frac{2\rho(1 - e^{-qt})}{(k + q) - (k - q)e^{-qt}} \tag{3.4.11b}$$

$$q = \sqrt{k^2 + 4\rho fC} \tag{3.4.11c}$$

$$f = \frac{2(2\rho + k)}{2\rho + k + C} \tag{3.4.11d}$$

where C is the rate coefficient for radical termination (s^{-1}), k is the rate coefficient of radical exit from the particle (s^{-1}), f is a coefficient that depends on N and varies between 1 and 2, and q is also a coefficient.

3.4.4 Monomer Distribution

The monomer concentration at the reaction site depends on the monomer partitioning between the different phases in an emulsion polymerisation. For a batch or continuous stirred tank reactor, the monomer will be distributed between the three phases: monomer, aqueous

and polymer phase. Semi-batch reactors, however, are often operated at what is known as *monomer starved* conditions. Under these conditions, the monomer phase does not exist, as all monomer that is added to the reactor is directly absorbed into the polymer particles. To achieve monomer starved conditions the reaction rate must be greater or equal to the monomer feed rate. Under monomer starved conditions, all monomer is partitioned between the polymer phase and the aqueous phase. In many cases it is also possible to assume that the monomer is solely in the particle phase. The monomer partitioning adds a new level of complexity to an emulsion polymerisation model. When mass balances are derived, one must also take into account the mass transfer between the phases. During Interval I and II of a batch emulsion polymerisation, the monomers partition themselves between the monomer droplets, the aqueous phase and the polymer particles. The monomer concentration in the polymer particles will influence the polymerisation rate. In Interval III there is no droplet phase, as all monomer is inside the polymer particles and no more monomer feed is added. This interval will be modelled as monomer starved conditions in a semi-batch reactor (Dimitratos et al., 1994).

The concentration of monomer in the polymer particle depends on mass transfer and the polymerisation rate, and hence the concentrations of the monomer in the different phases is given by a thermodynamic equilibrium. For a multimonomer system, the system is even more complex, and the monomer partitioning modelling involves simultaneously solving thermodynamic equilibrium equations and material balances. An empirical approach has been common in some research environments, where the monomer concentration is measured in the aqueous phase and the polymer particle is measured and a partition coefficient is measured. However, when more than one kind of monomer are involved in the reaction, this method becomes cumbersome and unreliable. Furthermore, it would be advantageous to be able to model the monomer partitioning of an emulsion polymerisation where the partition coefficient is not known. For this, a theoretical aspect is used. It was first demonstrated by Morton et al. that the monomer partitioning could be calculated by solving the thermodynamic equilibrium equations for each phase (Gao and Penlidis, 2002). The equilibrium equation are based on the principle behind the thermodynamic law that says that the chemical potential is the same in all phases are equal at equilibrium, assuming that thermodynamic equilibrium is reached quickly without mass transfer limitations. Both the empirical approach and the theoretical approach have their drawbacks and advantages (Gao and Penlidis, 2002).

If one of the monomer present is very soluble in water, a considerable amount of monomer will be present in the aqueous phase even under starved conditions. The monomeric unit may only participate in the polymerisation reactions in the particles if it diffuses through the aqueous phase, crosses the water/particle interface and diffuses into the monomer swollen particle. In this case, to obtain the monomer concentrations in each phase, the component balances of each phase must be developed, taking the transport processes into account. This requires knowledge of equilibrium concentrations, diffusion coefficients and surface areas of the various phases (Dimitratos et al., 1994).

If the process is not controlled by diffusion, the phases may be considered to be in equilibrium swelling conditions, which greatly simplifies the task of calculating the monomer concentrations in the different phases. The transport phase through the aqueous phase is

no longer regarded as the rate determining step, mainly because of agitation, which results in a convective mass-transfer process through the aqueous phase in stead of a diffusive one. This is particularly applicable for semi-batch processes (or Interval III in a batch process), as there is no separate monomer phase in the system (Dimitratos et al., 1994). The concentration of the monomer will depend on the balance between the gain in interfacial free energy caused by the increase in surface area as a result of swelling and the loss of free energy due to mixing monomer with polymer.

The theoretical approach is based on two principles: (1) The Flory-Huggins mixing model of small molecules and long polymer chains, and (2) the thermodynamic law of chemical potential at equilibrium. At equilibrium, the partial molar free energies of the monomer will have the same value in each of the phases.

$$\Delta G_p = \Delta G_d = \Delta G_a \tag{3.4.12}$$

where ΔG_p, ΔG_d and ΔG_a are the molar free energies of the polymer phase, droplet phase and aqueous phase, respectively.

The monomer partitioning is of special interest because of the starved conditions under which emulsion polymerisation in a semi-batch reactor are often carried out. The definition of starved conditions is that the there is no droplet phase, that all the monomer that enters the system is instantaneously absorbed into the particles. The amount of monomer that can dissolve in the particles depend on the monomer used, and emulsifier. Another way to define it is that the reaction rate of monomer equals the monomer feed flow rate. This ensures that there is no monomer hold-up in a droplet phase. The monomer droplets that might occur have a short lifetime.

The complete mass balance for the monomers can be written as:

$$\frac{dn_{Mi}}{dt} = \frac{d(V^R[M_i]^R)}{dt} = -V^p r_i^p - V^w r_i^w + \dot{n}_{Mi,feed} \tag{3.4.13}$$

where n_{Mi} is the total number of monomer i in the reactor, and $[M_i]^R$ is the total concentration of monomer i in the reactor. V^p is the volume of the particle phase, and V^w is the volume of the aqueous phase. The reaction rates of monomer in the particle phase and water phase are r_i^p and r_i^w, respectively. Both these rates depend on the monomer concentrations and thereby on the volumes of water and particle phase. Krämer has done significant work on determining these volumes by using partition coefficients $k_i^j, j \in \{p, d\}$. They are assumed to be independent of pressure and temperature, and are expressed as:

$$k_i^d = \frac{V_i^d/V^d}{V_i^w/V^w} \tag{3.4.14a}$$

$$k_i^p = \frac{V_i^p/V^p}{V_i^w/V^w} \tag{3.4.14b}$$

with V^d as the volume of the droplet monomer phase, and $V_i^j, j \in \{w, d, p\}$ in the water, droplet or particle phase respectively. An ideal mixture is assumed, and the different

volumes can be expressed by a system of algebraic equations.

$$V^d = \sum_{i=1}^{k} V_i^d \tag{3.4.15a}$$

$$V^p = V_{pol} + \sum_{i=1}^{k} V_i^p \tag{3.4.15b}$$

$$V^w = V_W + \sum_{i=1}^{k} V_i^w \tag{3.4.15c}$$

$$V_i = V_i^w + V_i^d + V_i^p = \underline{v}_i \cdot n_{Mi} \tag{3.4.15d}$$

where $\underline{v}_i$ is the molar volume of monomer i in the system, and V_{pol} and V_W are the volumes of the polymer and water, respectively. The volumes of polymer and water can be determined by balancing the monomer consumption in the water and polymer phase, as differential equations.

$$\frac{dV_{pol}^p}{dt} = \sum_{i=1}^{k} (V^p r_i^p + V^w r_i^w) \frac{M_i}{\rho_{pol}} \tag{3.4.16a}$$

$$\frac{dV_W^p}{dt} = \dot{V}_{W,feed} + \dot{V}_{I,feed} \tag{3.4.16b}$$

where ρ_{pol} is the average polymer density, and $\dot{V}_W$ and $\dot{V}_i$ is the volume flow rate of water and initiator, respectively, and V_i is the volume of monomer i. The initiator introduced to the system can be written as:

$$\dot{n}_{I,feed} = \dot{V}_{I,feed} \cdot [I]_{feed} \tag{3.4.17}$$

where $[I]_{feed}$ is the concentration of initiator on the feed.

From these equations, the monomer concentration in the phases $j, j \in \{w, d, p\}$ are determined by using the solution of the phase distribution calculation:

$$[M_i]^j = \frac{V_i^j}{V^j \underline{v}_i} \tag{3.4.18}$$

The system of equations, Eqs. 3.4.14 -3.4.16, may be difficult to solve since the monomer droplet phase may disappear and reappear, leading to discontinuities in simulation. Therefore, the simulation has to take care of two different cases, $V^d = 0$ and $V^d > 0$. When there is no droplet phase present, all the monomer goes into the swelled polymer particles, until they are saturated (Gesthuisen et al., 2004).

When the swelling of the particles is saturated, a droplet phase will form, creating a third phase. This droplet phase will not increase the reaction rate, since the reaction is present only inside the polymer particles and depends on the concentration of monomer in the particles. Adding more monomer will only accumulate in the droplet phase. According

to Gesthuisen et al. (2004), a droplet phase will result in poor controlability of the process, as the reaction rate cannot be changed until the monomer droplets are depleted. This makes the temperature more difficult to control, as the droplets are only depleted by reaction. Therefore, the article concludes that the droplet phase is unnecessary, potentially dangerous, and should be avoided.

3.4.5 Molecular Weight Distribution

Most polymers are made up of chains of different lengths,Therefore, the chain length of a polymer is given as a distribution of degrees of polymerisation (DP). Generally, this distribution is translated to a weight distribution, the molecular weight distribution (MWD), that characterises the polymer product. The MWD strongly affects the properties of the polymer. For example, the mechanical strength of polystyrene improves by increasing its molecular weight, but at the same time the melt viscosity increases, making processing more difficult. The MWD depends on the reaction conditions, the reaction mechanisms and reactor design. A polymer has a chain length n, i.e. the number of repeat units, including branches and endpoints. The chain length of all polymers has to be an integer number. A normal polymer has significantly more monomers in the chain than those making up the endpoints and branching, so for a homopolymer, it is possible to approximate the molecular weight of the polymer molecule to:

$$M_{m,n} = nM_m \tag{3.4.19}$$

where $M_{m,n}$ is the molecular weight of the polymer molecule, M_m is the molecular weight of a single unit and n is the number of repeat units in the polymer molecule. The MWD is the distribution of the molecular weights of the polymer molecules, as illustrated in Fig. 3.4.2. In a polymer particle the polymer molecules do not necessarily hold the same number of monomers. Even though the properties of the polymer depend on the whole distribution of molecular weights, the MWD is often characterised by an average, the number average molecular weight, M_n, or the weight average molecular weight, M_w. M_n is quite intuitive, it is the total number of monomers divided by the number of polymer chains times the molecular weight, which gives the average molecular weight of the polymer. M_w is less intuitive, but it determines the broadness of the distribution, and says something about if the polymers have the same size. The MWD can be calculated using moment equations, and simplifications are done to avoid solving hundreds of equations. Moment equations describe the system, taking into account all the different chains with different chain lengths of both the living (λ) and dead (μ) chains. They can be derived from population balances as described in Appendix A.1. The method of moments is a statistical way to represent the molecular weights. The 0^{th} live moment (λ_0) is the concentration of live radicals in the system, and the 1^{st} live moment (λ_1) is the concentration of monomers in the growing radical chain. The second moment is sometimes calculated as well, to determine the weight average molecular weight, M_w. The moment equations for the dead and alive chains are written in Eqs. 3.4.20, while the equations for M_n and M_w are given in Eq. 3.4.22a and Eq. 3.4.22b, respectively. The derivation of the moment equations can be found in Appendix A.2.

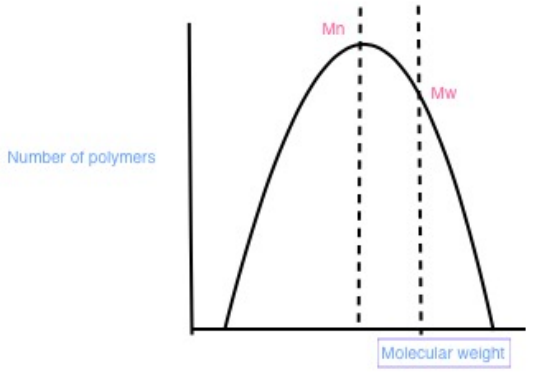

Figure 3.4.2: Illustration of the molecular weight distribution, with M_n and M_w.

$$\lambda_k = \sum_{n=1}^{\infty} n^k P_n \tag{3.4.20a}$$

$$\mu_k = \sum_{n=1}^{\infty} n^k D_n \tag{3.4.20b}$$

M_n is the number average molecular weight, and is expressed as the total number of monomer, divided by the number of chains, times the average molecular weight of the monomers, $\bar{M}_m$, (if there is more than one monomer).

$$M_n = M_m \frac{\lambda_1 + \mu_1}{\lambda_0 + \mu_0} \tag{3.4.21a}$$

$$M_w = M_m \frac{\lambda_2 + \mu_2}{\lambda_1 + \mu_1} \tag{3.4.21b}$$

Since the number of live polymer is significantly smaller than the number of dead polymers, it is often negligible. M_n can therefore be approximated:

$$M_n \approx M_m \frac{\mu_1}{\mu_0} \tag{3.4.22a}$$

$$M_w \approx M_m \frac{\mu_2}{\mu_1} \tag{3.4.22b}$$

The ratio between M_n and M_w determines the broadness of the distribution. This ratio is called the polydispersity index (PDI).

$$PDI = \frac{M_w}{M_n} \tag{3.4.23}$$

The closer PDI equals to one, the more equal in size are the polymer particles, and this effects the polymer properties. For example, the stiffness of polypropylene increases as

the PDI increases. The PDI is common between 2 and 5, though polydispersities up to 100 are known (Schork et al., 1993). For a homopolymer, M_m is the molecular weight of the monomer unit. For a copolymer an average of the weight and number of monomers is used to calculate an average monomer molecular weight.

$$\bar{M_m} = \sum x_{pi} M_{mi} \tag{3.4.24}$$

Where x_{pi} is the fractional molar composition of the polymer, and M_{mi} is the molecular weight of monomer i (Asua, 2008).

The variation in the degree of polymerisation, and therefore the number average molecular weight of the polymer, occurs for at least three reasons. The main mechanism that broadens the MWD is through the series-parallel reaction mechanisms leading to chain formation. Another mechanism that effects the MWD is the spatial differences in conditions in the reactor. Changes in temperature and monomer concentration in the reactor will influence the number average molecular weight. The final mechanism is the stochastic variations of reaction rates at a molecular level. This, however, has been proven to be insignificant towards the other two mechanisms. To conclude, the molecular weight distribution is a result of the reaction mechanisms, when all environmental variables like temperature and monomer concentration are kept constant.

3.4.6 Conversion

For this study, an important variable is the conversion of monomer to polymer because it is included in the definition of the batch time. In batch reactors, the reaction continues until equilibrium has been reached or the reactant is exhausted, and so the conversion gets higher the longer the reactant stays in the reactor. In the semi-batch reactor, like in this case, the conditions may be monomer starved, meaning that the monomer that is added to the reactor reacts almost instantaneously to polymer. Conversion says something about the amount of reactant i that reacts (or is consumed). Conversion can be defined in different ways. Eq. 3.4.25 is an expression for the overall, total, conversion.

$$X_{i,tot} = \frac{\text{Monomer reacted}}{\text{Total amount of available monomer}} \tag{3.4.25a}$$

$$X_{i,tot}(t) = \frac{n_{i,0} - n_{i,t} + \int_{t=0}^{t} \dot{n}(t)dt}{n_{i,0} + \int_{t=0}^{t_f} \dot{n}(t)dt} \tag{3.4.25b}$$

where $n_{i,0}$ is the amount of available monomer i initially in the reactor, at time $t = 0$, $n_{i,t}$ is the amount of available monomer i at time t. $\int_{t=0}^{t} \dot{n}(t)dt$ is the net flow of reactant, that is the amount of monomer i that has been fed into the reactor at time t, and $\int_{t=0}^{t_f} \dot{n}(t)dt$ is the amount of reactant that has been fed into the reactor at the final time t_f, i.e. all available monomer i. Another way of expressing the conversion is the continuous, or instantaneous, conversion. The definition of the instantaneous conversion is given in Eq. 3.4.26. This is the amount of monomer i that has reacted divided by the total number of monomer that

has been fed into the reactor at time t.

$$X_{i,inst} = \frac{\text{Monomer reacted}}{\text{Monomer fed into the reactor}} \tag{3.4.26a}$$

$$X_{i,inst}(t) = \frac{n_{i,0} - n_{i,t} + \int_{t=0}^{t} \dot{n}(t)dt}{n_{i,0} + \int_{t=0}^{t} \dot{n}(t)dt} \tag{3.4.26b}$$

3.4.7 Gel Effect

The polymer concentration in the system increases with conversion, and so does the viscosity. The rapid increase in viscosity will effect the rate at which two polymer chains encounter each other. As a result, the rate of reaction is determined by how quickly two polymer chains that have to manoeuvre their way through the tangle of dead polymer chains. This effect is called the gel effect, and can be seen as a sharp increase of the polymerisation rate due to a decrease in the termination rate.

3.4.8 General Energy Balance

To determine the temperature of the reactor, the energy balance of the system has to be written. The derivation of the complete energy balance can be found in Appendix A.3. The final temperature equation can be written as

$$\frac{dT_R}{dt} = \frac{\Delta H_f + \Delta H_J + \Delta H_{loss} - \Delta H_R + \dot{W}_{stirr}}{mc_p} \tag{3.4.27}$$

where T_R is the reactor temperature, ΔH_f is the cooling effect of the feed, ΔH_J is the cooling effect of the jacket, ΔH_{loss} is the heat lost to the environment, ΔH_R is the heat of reaction and W_{stirr} is the effect of the stirrer. m represents the mass in the reactor and c_p is the average specific heat capacity of the fluid in the reactor. As the reaction is exothermic, $\Delta H_R < 0$ by convention, and reaction will contribute to an increase in the reactor temperature.

Energy balance of the cooling jacket

A simple way of controlling the temperature in a reactor is to surround it with a jacket where some fluid (often water) circulates. This water can both cool down or heat up the reactor, depending on what is necessary. Since the reaction taking place here is exothermic, the jacket serves mostly to cool down the reactor. The effect the jacket has on the reactor temperature depends on the cooling capacity of the jacket: How well heat is transported between the reactor and the jacket. This depends on the temperatures of the reactor and jacket, the type of jacket, the area of heat exchange and the flow of fluid in the jacket. Once the reactor size exceeds around 10 m^3 (industrial size), temperature control becomes an issue. A larger reactor will always have a smaller reactor wall heat transfer area per unit

volume of the reaction mixture. This relation can be seen in Eq. 3.4.28, when a scale-up of the reactor is done, keeping the geometry consistent: The ratio between the surface of the reactor and the volume decreases, and this will affect the heat removal rate. A/V is the surface to volume ratio. The surface A is the interfacial surface between the reactor and the cooling jacket, and the volume V is the volume of fluids inside the reactor. For larger reactors, the temperature difference varies linearly with the reactor's cooling capacity. As a consequence, a point might be reached where the heat from the reaction is larger than the heat removal (Dickerson, 2015).

$$\frac{A}{V} = \frac{1}{V^{\frac{1}{3}}} \tag{3.4.28}$$

The temperature of the jacket is also modelled. It is deduced from the same principles as the reactor temperature.

$$\frac{dT_J}{dt} = \frac{\dot{m}_0 c_{p,water}(T_{water,in} - T_{water,out}) - UA(T_R - T_J)}{V_{jacket} c_{p,water}} \tag{3.4.29}$$

where T_R is the reactor temperature, T_J is the jacket temperature, $\dot{m}_0$ is the mass flow rate of water through the jacket, $c_{p,water}$ is the specific heat capacity of water, $T_{water,in}$ is the temperature of the water inlet of the jacket and $T_{water,out}$ is the temperature of the water outlet of the jacket. UA is the heat transfer coefficient and V_{jacket} is the volume of the water in the jacket.

CHAPTER

FOUR

OPTIMISATION

Optimisation became an area of academic interest after World War II. It started through operations research, attempting to optimise complex systems and phenomena. In the early stages, mostly mathematicians, physicians and economists contributed to the research. The interest in this technology for process systems engineering has evolved from an academic interest to a technology that has made, and still makes, a significant impact on the industry.

The aim of this thesis is to find the optimal, i.e. minimum, polymerisation time for a semi-batch emulsion copolymerisation reaction. This chapter will cover the theoretical aspects of optimisation, from the general solving of an optimisation problem, dynamic optimisation and to the theory behind Model Predictive Control (MPC). MPC is an advanced control technology that was first developed to meet the specialised needs of optimisation in power plants and oil refineries, but is now applied in a multitude of industries, including chemicals, food processing and aerospace applications (Qin and Badgwell, 2003).

4.1 Introduction to Optimisation

Optimisation is used everywhere. Physical systems optimise to reach a state of minimum energy, and chemical systems react to minimise their potential energy. Investors create portfolios with minimum risk while achieving a high rate of return, and manufacturers seek maximum efficiency of their production processes. In order to make use of optimisation as a tool, an objective has to be defined (the *objective function*) subject to different constraints. The constraints may be physical (e.g. when a valve is fully open, it cannot be opened more) while other constraints can be specifications on the product (e.g. purity constrains, product quality). Defining the objectives, variables and constraints for a given problem is called modelling. When the model is found, an optimisation algorithm can be applied to the model. There is no universal optimisation algorithm that can be applied to

all optimisation problems. Instead, there is a collection of different algorithms that are used depending on the optimisation problem. The choice of algorithm falls on the user, and this choice may determine if the problem is solved rapidly, more slowly or not at all. Once the optimisation algorithm has been applied, one has to be able to determine whether it was successful or not - has it succeeded in finding a solution? If it has, is this the optimal solution? Elegant mathematical formulations known as *optimality conditions* are used to verify that the variables found are in the solution of the problem. Good algorithms should posses the following properties (Nocedal and Wright, 2006, Chapter 1):

Robustness The algorithm should perform well for many different problems and for reasonable values of the starting point.

Efficiency The algorithm should not require enormous amount of computer storage and time.

Accuracy The solution found should be precise .

4.1.1 Finding the Optimal Solution

Mathematically, the optimisation problem is formulated as an objective function *f(x)* to minimise or maximise, where *x* is a vector of variables to be determined, also known as *decision variables*. Constraints, c_i, are equality and inequality functions that the unknown function x must satisfy. A solution will correspond to a local or global optimum (Bradley et al., 1977, Chapter 13). Global and local minima are illustrated in Fig. 4.1.1 and defined in Def. 4.1.

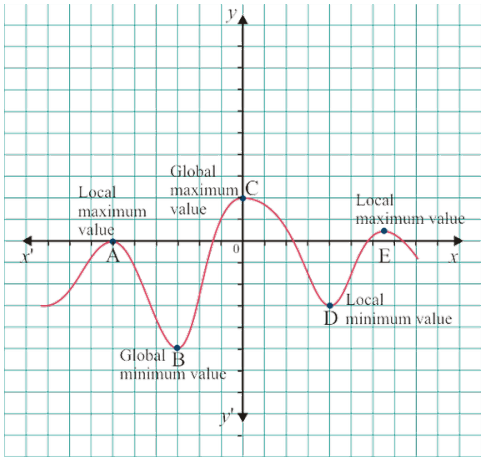

Figure 4.1.1: Illustration of the concept of global and local minima.

Definition 4.1. *Let $x = (x_1, x_2, .., x_n)$ be a feasible solution to a minimisation problem $f(x)$. We call x:*

1. *A global minimum if $f(x) \leq f(y)$ for any feasible point $y = (y_1, y_2, .., y_n)$.*

2. *A local minimum if* $f(x) \leq f(y)$ *for any feasible point* $y = (y_1, y_2, .., y_n)$ *sufficiently close to* x.

If the solution found is only a local optimum, this means that there may exist a better solution, the global optimal solution. Convexity of a problem helps define if the global optimal solution is found, and this is a fundamental concept in optimisation. If the problem is convex, any optimal solution is the global solution, according to Theorem 4.1.1 from Nocedal and Wright (2006). The concept of convexity is illustrated in Fig. 4.1.2.

Theorem 4.1.1. *When f is convex, any local minimise* x^* *is a global minimise for f. If in addition f is differentiable, then any stationary point* x^* *is a global minimise of f.*

This means that the problem is convex if:

1. The feasible region (set) is convex. I.e. a straight line segment between any two points in the set lies entirely inside the set.
2. The function is convex. I.e for any two variables x and y in a convex set satisfy the following property:

$$f(\alpha x + (1 - \alpha)y \leq \alpha f(x) + (1 - \alpha)f(y)$$

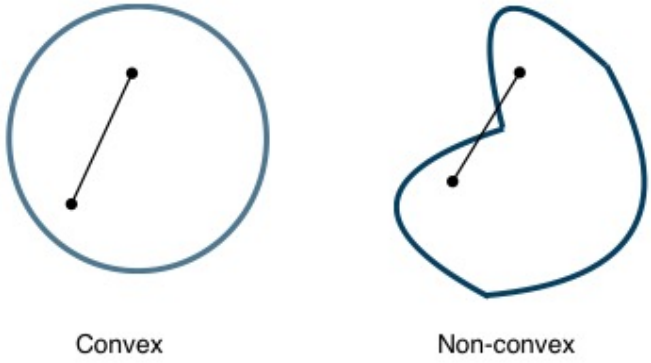

Figure 4.1.2: Illustration of the concept of convexity.

4.1.2 Optimisation of Dynamic Systems

Chemical processes are dynamic systems that change over time, and are therefore often modelled by differential equations. The variables are time dependent, the optimisation problem is solved over a time horizon, and the objective function must include time. In most cases the time is discretised, with a constant sampling time. The decision variables are chosen specifically for each problem, by the engineer. Before discussing discrete time optimisation problems any further, some general definitions have to be made clear. Some important general terms that will be used in the remainder of the thesis are:

Manipulated variables: MV (u) : A variable that can be manipulated to control the plant. These are also called control inputs.

Control variable: CV (z) : A variable for which optimal setpoints or boundaries are defined. These are also called outputs.

Disturbance variable: DV (v) : A process input. If measured, it can be controlled through feed forward control. If it is not measurable, feedback is obtained through estimation.

States (x) : Defines the different states of the system. $z = Hx$, where H is a selection matrix to select the states that are controlled variables.

Decision variables : The variables that are determined through optimisation, i.e. the outputs (z) and the inputs (u).

Degrees of freedom (DOF) : the number of process variables that can be set by the operator, or control system. In this process the degrees of freedom correspond to the number of manipulated variables that are used to control the process (Ponton, 1994).

Optimisation problem : A general way of writing optimisation problems is:

$$\min_{x \in \mathbb{R}} \quad f(x) \tag{4.1.1a}$$

$$s.t. \quad c_i(x) = 0, \quad i \in \mathcal{E} \tag{4.1.1b}$$

$$c_i(x) \leq 0, \quad i \in \mathcal{I} \tag{4.1.1c}$$

where Eq. 4.1.1a is the objective function, c_i are the constraints, $\mathcal{E}$ is the set of equality constraints and $\mathcal{I}$ is the set of inequality constraints.

Solving Optimisation Problems

Optimisation problems are classified according to their objective function and constraints. Eq. 4.1.1 is a general way to formulate an optimisation problem. When the cost function (objective function, Eq. 4.1.1a) and all the constraints are linear, the problem is a linear programming problem (LP). LP problems are widely formulated and easy to use. The most common algorithm to solve LP problems is the *Simplex algorithm* that was developed by Dantzig in the 1940's. The Simplex method is described in Nocedal and Wright (2006). A newer class of methods known as *interior-point methods* have in some cases proved to be faster for some LP problems. A quadratic programme (QP) has a quadratic cost function and linear constraints. The QP problems are commonly solved using *Active Set methods*, also described thoroughly in Nocedal and Wright (2006). In nonlinear programs (NLPs), some of the constraints and the cost function might be nonlinear. These problems are harder to solve, and require special algorithms.

Optimisation algorithms are iterative, i.e. they start out with an initial guess of the variables, and then they generate new and improved states until they reach some sort of termination criteria, hopefully at the right solution. If the problem is convex, several efficient algorithms are available to solve the optimisation problem, like the Simplex algorithm for

a linear program or the active set method for a quadratic program. However, nonlinear programs are not convex, and are therefore more complicated to solve. It is harder to find an efficient algorithm requiring little computational power. Polymerisation processes are nonlinear, therefore not convex, and this is the reason for the complexity related to optimisation of these processes.

4.1.3 The Process Control Hierarchy

Fig. 4.1.3 illustrates how a plantwide control system is often structured. The aim of the plant is to maximise an overall economic objective while maintaining the process in a safe operation mode, meeting the product quality demands and maximising the production profit (Tatjewski, 2008). The *regulatory control* layer, also called the direct control layer is responsible for the basic safety of the plant. This is the only layer that has direct access to the plant and can manipulate input variables. The *advanced process control* layer is treated as a supervisory control layer. The outputs of this layer provides the setpoints for the regulatory layer. This layer may also contain more advanced control algorithms than the ones in the regulatory control layer, for example MPC control. This layer is also slower, having feedback control on slower-varying process variables. Due to global market conditions that constantly change and are challenging to fulfil, plantwide economic control is becoming more critical. As a result, the traditional steady-state real-time optimisation of the plant, the RTO layer, is transformed into *dynamic real-time optimisation*, DRTO, based on a dynamic prediction model. This is the layer that calculates the optimal trajectories for the process, taking the constraints from the layer above and below into consideration. These trajectories are sent to the Advanced Process Control layer. The top layer, Planning and Scheduling, has a timescale of weeks to months, and ensures the product quality and production, optimises the plant economics and provides inventory constraints to the RTO layer.

Figure 4.1.3: Typical control hierarchy.

4.2 Model Predictive Control

Model Predictive control is an advanced control technique that uses a process model to predict the future behaviour of the process. It merges feedback control with dynamic optimisation. An open loop problem, as the one studied in the preliminary project, see Kjetså (2016), can predict trajectories for the process to follow. However, without feedback the trajectory is not updated to take into account model inaccuracies and disturbances. On-line control introduces feedback to the optimisation problem and the predicted trajectories can be updated accordingly.

The first generation MPC controllers were developed in the 1970's, and have since gained their popularity in the process industry. The method has been successful in the industry because of its increased performance and understandable concept. MPC was first developed to meet special control needs of a petroleum refinery. Today MPC is applied in a multitude of areas, including chemicals, food processing, automotive, and aerospace applications. In modern processing plants the MPC is a part of a multi-level hierarchy of control functions. MPC technology has been driven forward by the industry, and Qin and Badgwell (2003) could report that by the end of 1999, there were over 4,500 MPC applications worldwide, primarily found in oil refineries and petrochemical plants - and only 17 applications in the polymer industry. For nonlinear MPC (NMPC), Qin and Badgwell (2003) report that the numbers were even lower, with a total of 98 applications in industry, whereof 21 in the polymer industry. However, it must be noted that for this survey the vendors could themselves decide the definition of an *application*, and one should be careful to draw conclusions from this data. The number of MPC and NMPC applications were doubled from the previous survey by Qin and Badgwell 5 years previously. It is safe to say that the overall usage of MPC is increasing rapidly.

One of the reasons for the success of MPC is that the principle behind MPC is easy to comprehend. Imagine a multiple-input multiple-output (MIMO) process that has to satisfy constraints on the input and output variables. If a dynamic model is accurate and accessible, the model and current measurements can predict the future output values. Based on the predictions and the measurements, the appropriate changes in the input variables can be calculated (Seborg et al., 2010, Ch. 20). The success off MPC in industry is also linked to other advantages:

1. The process model captures the dynamic and static interaction between MVs, CVs and DVs.
2. The constraints on the input and output variables are considered systematically.
3. The calculation of optimum setpoints is done simultaneously with the control calculations.
4. The predictions can provide early warnings of potential problems.

The success of the MPC application is clearly dependent on the accuracy of the process model used (Seborg et al., 2010, Ch. 20).

A block diagram of an MPC controller is shown in Fig. 4.2.1. A process model predicts

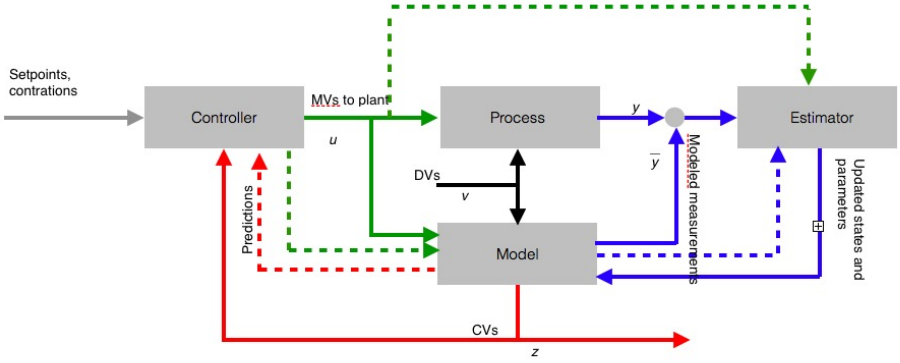

Figure 4.2.1: Schematic representation of MPC. The model receives the same input data as the real process, and estimates response of the process in state values. The estimates are compared with measurements from the real model and the difference is sent to the estimator that changes some of the parameters in the model so that the difference between the estimates and the real values are minimised. The outputs are sent in a feedback loop to the controller that calculates new input values.

the current values of the states and other variables. These are compared to the states and measurements that come from the real process. The differences between the measured values and the model estimated values are called residuals. The residuals, or the model errors, are sent as a signal to an *estimator* block, that updates the states and parameters of the model so that the estimates will lie closer to the real measurements. The model also sends its prediction to a controller, that uses these predictions to calculate the control input moves to lie within the constraints and achieve the setpoints that are given. The setpoints for the control calculations are sometimes called *targets*, and they can be constant throughout the time horizon, or be a reference trajectory that the MPC has to follow. The optimal values for the process may change frequently due to changes in the process conditions, instrumentation, or cost of energy and raw materials. The objective of the control calculations is to determine a sequence of control moves so that the predicted states reach their setpoints in an optimal trajectory.

Fig. 4.2.2 shows the predicted output $\hat{x}$, the measured output x and the MV u trajectories for a single input single output system (SISO). k is the current sampling time. At time k the MPC calculates M values of the MV for the control horizon, so that the CV will reach its setpoint at the end of the prediction horizon. Then the first control move is applied to the process. This procedure is then executed at every sampling instant. It is important that the prediction horizon is long enough to capture the dynamics of the system. The general MPC algorithm is as following:

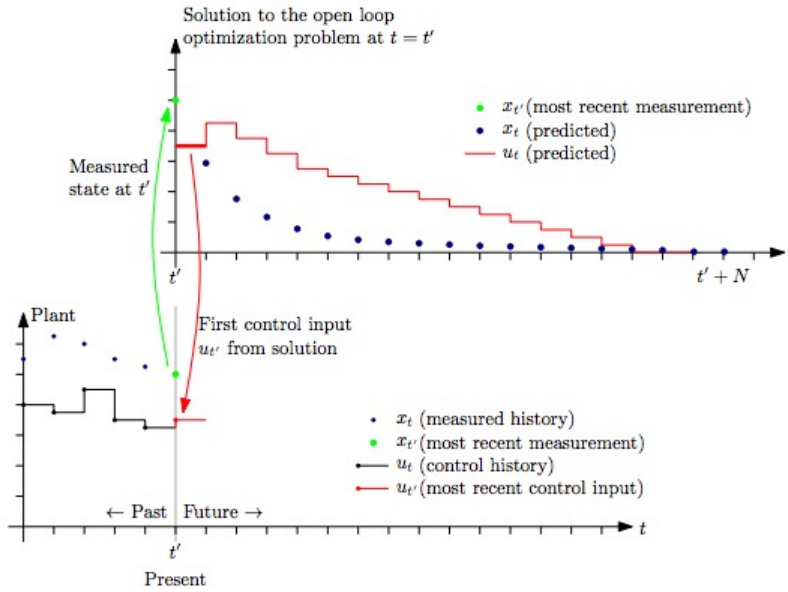

Figure 4.2.2: Illustration of the concept of MPC. The current state is read by the controller and it calculates an optimal trajectory to obtain the desired value of the output values. The first input move is applied to the process. At the next time sample, this is repeated.

Algorithm: NMPC with state feedback
for t = 0,1,2,...N **do**

(1): Get the current state x_t

(2): Solve the optimisation problem Eq. 4.2.4 on the prediction horizon from time t to time $t + N$ with x_t as the initial condition.

(3)Apply the first control move from the solution above.

end

The advantages of MPC include the possibility to operate the process closer to the constraints. This is because the optimal trajectory for the input moves is recalculated at every time sample, using the current states, and the predictions allow for closer control of the process. This is sometimes referred to as the "squeeze and shift" rule. Allowing closer control, the variance of the output values is smaller ("squeezed") and the operating set-point can be moved closer to the constraint ("shift"). Furthermore, the process model captures the interactions between the inputs, outputs and the disturbed variables. If the model and the predictions are accurate, it can early indicate a future problem that needs to be taken care of. However, the success of MPC heavily relies on the accuracy of the

process model. MPC controllers are designed to drive the process from one steady-state to another. It may receive the optimal steady-state target from an overlying controller, or it uses an internal steady-state optimiser. Qin and Badgwell (2003) summarises the main objectives of the MPC, in order of importance:

1. Avoid that the inputs and outputs violate their constraints.
2. Drive the outputs to their setpoints (steady-state optimal value).
3. Drive the MVs to their steady-state optimal values using the remaining degrees of freedom.
4. Avoid excessive movement on the MVs.
5. Control as much of the plant as possible if signals and actuators fail.

These objectives have to be translated into a mathematical problem, and undergo several approximations and trade-offs to design the basic controller. As with other design problems, there are many possible solutions, and so there are several different possible MPC formulations. However, the first step is always to read the current MVs, CVs, and DVs. The next steps include determining where the process is now, where is it heading, and what is the best way to drive the process where it should go?

4.2.1 Formulating the MPC Problem

Hovd (2004) gives a comprehensive introduction to MPC and the formulation of the MPC problem. He reveals that one of the important issues to consider is the sample time of the MPC. It is important that the computation time of the optimisation problem does not exceed the sample time. I.e. if a sample is run every 10 seconds, the computation time for the optimisation problem has to be less than 10 seconds. As a result, the optimisation problem is usually either an LP or a QP (see Sec. 4.1.1) because these require less computational time. To ensure that there exists only a unique solution for the QP that can be found efficiently with optimisation solvers, the QP problem has to be convex. Furthermore, Hovd (2004) points out that even though an LP problem may be advantageous for large optimisation problems, a QP problem will generally lead to smoother control action and more intuitive effects of the changes in the tuning parameters. A QP problem is generally easy to solve, there are several efficient algorithms to choose from. If the model or the objective function are nonlinear, the optimisation problem is nonlinear, and becomes more complicated to solve.

Since the future behaviour of the process is determined from a model, this model is crucial to the success of the MPC controller. A basic dynamic model is written with differential equations on the following continuous form:

$$\frac{dx}{dt} = f(x, u, t) \tag{4.2.1a}$$

$$y = h(x, u, t) \tag{4.2.1b}$$

$$x(t_0) = x_0 \tag{4.2.1c}$$

in which $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}^m$ is the input, $y \in \mathbb{R}^p$ is the output and $t \in \mathbb{R}$ is time. x_0 specifies the initial conditions at time $t = t_0$. Solving the differential equations will give a solution to the problem. The continuous state space model is a linear time-varying model as follows:

$$\frac{dx}{dt} = Ax(t) + Bu(t) \tag{4.2.2a}$$
$$y = Cx(t) + Du(t) \tag{4.2.2b}$$
$$x(0) = x_0 \tag{4.2.2c}$$

where $A(t) \in \mathbb{R}^{n \times n}$ is the state transition matrix, $B(t) \in \mathbb{R}^{n \times m}$ is the input matrix, $C(t) \in \mathbb{R}^{p \times n}$ is the output matrix, and $D(t) \in \mathbb{R}^{p \times m}$ allows direct coupling between u and y. In many cases, $D = 0$. Linear models are the most common because of the simplicity of finding a solution. However, the problem is normally discrete, and Hovd (2004) proposes to rewrite the model from Eq. 4.2.2 to a *discrete-time state space* model form:

$$x_{k+1} = Ax_k + Bu_k \tag{4.2.3a}$$
$$y_k = Cx_k \tag{4.2.3b}$$
$$x(0) = x_0 \tag{4.2.3c}$$

where the subscript k refers to the sampling instant, and $k + 1$ refers to the sample instant after sample k. Note that the measurement y_k generally does not depend on the input u_k, but on the input u_{k-1} which it receives from the state x_k.

In literature, x, u and y are interpreted as *deviation variables*. This means that they represent the deviations from some consistent set of variables $\{x_s, u_s, y_s\}$ around which the model is obtained. To illustrate this, if y is a temperature measurement, and the setpoint temperature $y_s = 300K$, the physical measurement of $301K$ corresponds to the deviation variable $y = 1K$.

A typical MPC problem may have the following form.

$$\begin{aligned}\min_{x,u} f(x_i, u_i) = \sum_{i=0}^{n-1} &((x_i - x_{ref,i})^\top Q_i (x_i - x_{ref,i}) \\ &+ (u_i - u_{ref,i})^\top R_i (u_i - u_{ref,i})) \\ &+ (x_n - x_{ref,n})^\top S_n (x_n - x_{ref,n})\end{aligned} \tag{4.2.4a}$$

subject to

$$x_{i+1} = g(x_i, u_i) \tag{4.2.4b}$$
$$x_0, u_{-1} = \text{given} \tag{4.2.4c}$$
$$x^{low} \leq x_i \leq x^{high} \qquad \text{for } 1 \leq i \leq n \tag{4.2.4d}$$
$$u^{low} \leq u_i \leq u^{high} \qquad \text{for } 0 \leq i \leq n-1 \tag{4.2.4e}$$

The objective function (Eq. 4.2.4a) penalises the deviation of the state x_i and input at time sample i from its desired reference value x_{ref} and u_{ref} by setting different values

in the weighting matrices Q, R and S. The reference trajectories may remain constant throughout the time horizon, or they may vary from sample to sample, $x_{ref,i}$.

The following comments can be made to this formulation (Eq. 4.2.4):

- The weighting matrices Q, R and S are assumed to be symmetric. Q may be positive semi-definite, while R and S are positive definite.
- In many cases it makes more sense to put a weight (or cost) on the measurements or CVs rather than the states. This is easily done by choosing $Q = C^\top \widetilde{Q} C$, where $\widetilde{Q}$ is the weight on the CVs.
- One may also constrain the rate of change on the inputs, giving an additional constraint: $\Delta u^{low} \leq u_i - u_{i-1} \leq \Delta u^{high}$.

Solving the Optimisation Problem

The first step in the NMPC algorithm is to solve the optimisation problem at the current time instant. Depending on the type of optimisation problem (as explained in Section 4.1.2), different algorithms are used. Since polymerisation reactions in a semi-batch reactor are highly nonlinear, this optimisation problem is an NLP. Sequential quadratic programs (SQP) are common algorithms used to solve nonlinear problems. This section will describe how the SQP algorithm works.

The SQP method is an iterative method that takes the iterate $x^k, k \in N$, as an initial condition and solves an optimisation problem in order to find the next iterate, x^{k+1}. The objective of this is to create a sequence of approximations $(x^k)_{k \in N}$ that converges to a solution x^*. One great advantage with SQP is that the iterates x^k do not have to be feasible points. The optimisation problem is a quadratic subproblem because it is relatively easy to use, and simultaneously is able to reflect some of the nonlinear properties from the nonlinear problem. Some definitions and optimality conditions are stated in order to explain the construction of this quadratic subproblem (Boggs and Tolle, 1995).

The Lagrangian function plays an important role in the optimality conditions, and it is defined in Def. 4.2.

Definition 4.2. *The Lagrangian function is defined as*

$$\mathcal{L}(x, u, v) = f(x) + u^\top h(x) + v^\top g(x) \tag{4.2.5}$$

where u^k and v^k are Lagrangian multipliers and f(x) is the objective function and h(x) and g(x) are the equality and inequality constraints, respectively.

For a point x^* to be the optimal solution for the SQP, the first order necessary optimality conditions have to be satisfied.

Definition 4.3. *For a point x^* to be a local minimum for the nonlinear program, the first order necessary optimality condition has to be satisfied:*

$$\nabla \mathcal{L}(x^*, u^*, v^*) = \nabla f(x^*) + \nabla h(x^*) u^* + \nabla g(x^*) v^* = 0 \tag{4.2.6}$$

Furthermore, the second order optimality conditions have to be satisfied.

Definition 4.4. ***The second order optimality conditions:***
1. The columns of $G(x^*)$ are linearly independent. $G(x)$ is the matrix of the gradients of the active constraints at x.

$$G(x) = (\nabla h_1, \nabla h_2, ..., \nabla h_m, \nabla g_{1a}, ..., \nabla g_{qa}) \tag{4.2.7}$$

2. The Hessian of the Lagrangian is positive definite on the null space of $G(x^*)^\top$, i.e.

$$d^\top \nabla^2 \mathcal{L} d > 0 \text{ for all } d \neq 0 \text{ such that } G(x^*)^\top d = 0 \tag{4.2.8}$$

These optimality conditions ensure that x^* is an isolated optimal point and that the Lagrangian multipliers u^* and v^* are uniquely determined (Hoppe, 2006). One of the difficulties with SQP is how to determine the quadratic subproblem. One solution is to linearise the constraints around the iterate point x^k as done in Eq. 4.2.9.

$$\min_{d_x} \quad r^{k\top} d_x + \frac{1}{2} d_x^\top B_k d_x \tag{4.2.9a}$$

$$s.t. \quad \nabla h(x^k)^\top d_x + h(x^k) = 0 \tag{4.2.9b}$$

$$\nabla g(x^k)^\top d_x + g(x^k) \leq 0 \tag{4.2.9c}$$

where $d_x = x - x^k$ and the vector r^k and the matrix B_k have to be determined. $g(x)$ are the inequality constraints, and $h(x)$ are the equality constraints. An obvious choice for a quadratic objective function would be a local quadratic approximation around x^k, making r^k the gradient and B_k the Hessian of the objective function and x^k. However, this choice is not optimal as the constraints are nonlinear. To take the nonlinear constraints into account in the objective function, the local quadratic approximation of the Lagrangian function is used instead. At a current iterate (x^k, u^k, v^k), the Taylor expansion of the Lagrangian function is as follows.

$$\mathcal{L}(x^k, u^k, v^k) + \nabla \mathcal{L}(x^k, u^k, v^k)^\top d_x + \frac{1}{2} d_x^\top \nabla^2 \mathcal{L}(x^k, u^k, v^k) d_x \tag{4.2.10}$$

In the Lagrangian function the nonlinear constraints are added to the function. The multipliers u^k and v^k are not known, but are maintained as a part of the iterative process. Letting B_k be an approximation of the Hessian $\nabla^2 \mathcal{L}(x^k, u^k, v^k)$, the approximated quadratic problem can be written as follows.

$$\min_{d_x} \quad \nabla f(x^k)^\top d_x + \frac{1}{2} d_x^\top B_k d_x \tag{4.2.11a}$$

$$s.t. \quad \nabla h(x^k)^\top d_x + h(x^k) = 0 \tag{4.2.11b}$$

$$\nabla g(x^k)^\top d_x + g(x^k) \leq 0 \tag{4.2.11c}$$

The solution of the optimisation problem of Eq. 4.2.11, d_x, is used to find the new iterate x^{k+1}, by taking a step from x^k in the direction d_x. To continue to the next iterate the multipliers need to be known. These can be chosen in many different ways, one commonly

used way is to use the optimal multipliers from the quadratic subproblem. The optimal multipliers from the quadratic subproblem are noted as u_{qp} and v_{qp}, and setting

$$d_u = u_{qp} - u^k \tag{4.2.12a}$$
$$d_v = v_{qp} - v^k \tag{4.2.12b}$$

Then, the new iterates are given by the following equation:

$$x^{k+1} = x^k + \alpha d_x \tag{4.2.13a}$$
$$u^{k+1} = u^k + \alpha d_u \tag{4.2.13b}$$
$$v^{k+1} = v^k + \alpha d_v \tag{4.2.13c}$$

where α is a step length parameter with a value between 0 and 1. When these new iterates have been calculated, the problem functions and derivatives are evaluated, and the Hessian B_{k+1} is calculated.

One has to guarantee that the QP that is solved in the SQP algorithm has a solution, i.e. that the sequence created by the algorithm will converge. This is done by including a *merit function* in the objective function, this will ensure convergence. There exist several types of merit functions. Their purpose is to penalise the objective function by increasing for every iterate, as well as penalising infeasible iterates.

The basic algorithm is as given in the book by Boggs and Tolle (1995).

Basic SQP Algorithm
Given the initial values (x^0, u^0, v^0), B_0, a merit function ϕ and a set *k=0*

1. Form and solve the QP to obtain (d_x, d_u, d_v)
2. Choose a steplength α so that the merit function holds
3. Set

$$x^{k+1} = x^k + \alpha d_x$$
$$u^{k+1} = u^k + \alpha d_u$$
$$v^{k+1} = v^k + \alpha d_v$$

4. Stop if converged
5. Compute B_{k+1}
6. Set k:= k+1, repeat from step 1.

Constraint formulations

Previously, two different types of constraints have been described: Inequality and equality constraints. However, for a multivariable control problem, it is not always possible for all constraints to be fulfilled at all times. The optimisation algorithm has to know how to handle the constraints so that the problem does not become infeasible.

Qin and Badgwell (2003) describes three different constraints that are common in industrial MPC: Hard, soft and setpoint approximations. These are illustrated in Fig. 4.2.3. Hard constraints, as shown in the top figure of Fig. 4.2.3 should under no circumstances be violated. For soft constraints, shown in the middle figure of Fig. 4.2.3, some violation may be allowed. This violation is typically minimised by the quadratic penalty in the objective function. Soft constraints may also be handled with setpoint approximations, as seen in the bottom of Fig. 4.2.3. A setpoint is defined, and a quadratic penalty is applied on both sides of the setpoint/constraint. The weight is dynamic, so that a large weight is applied only when a violation of the constraint is predicted in the prediction horizon, to bring the CV back to its setpoint.

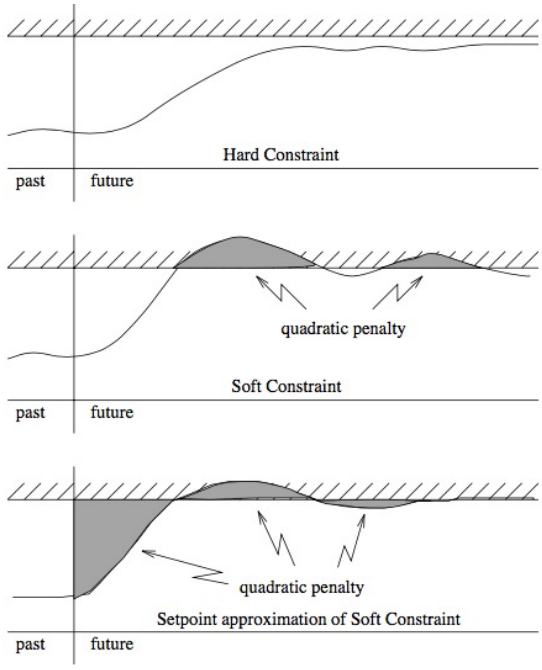

Figure 4.2.3: An illustration of the three basic types of constraints: hard constraints, soft constraints and setpoint approximation (Qin and Badgwell, 2003).

Constraints are usually formulated as inequalities. A variable has to lie within certain boundaries. For example, a flow can only be equal to or larger than zero, when the valve is completely closed, and equal to or lower than its maximum flow rate, when the valve

is completely open. When the inequality constraint is saturated, it turns into an equality constraint, called an *active constraint*. When a constraint is active, one degree of freedom is lost, because it is saturated and can not be used to control the process in case of any disturbances.

Feasibility is important when solving the MPC problem. An optimisation problem is *infeasible* if there does not exist no set of values for which all constraints are satisfied (Hovd, 2004). Problems with infeasibility in MPC controllers may occur when operating close to a constraint and large disturbances occur so that it is not possible to fulfil the constraint. It is desirable that the MPC controller does not "give up" and terminate when faced with an infeasible problem, but rather that it effectively moves the process into an operating region where all the constraints are feasible (Hovd, 2004). This is why soft constraints are often preferred in the MPC calculation. Soft constraints provide a feasible solution to the problem, but adds an extra weight in the objective function as slack variables, see Eq. 4.2.17. As long as the constraints are not violated, the slack variables equal zero.

For most MPC applications the MV constraints are hard constraints. This is because the MV constraints are usually linked to physical properties, like the maximum opening of a valve. It is also possible to define the rate of change in these MVs as constraints (Δu). These are usually defined to keep the controllers in a controllable range, and to avoid abrupt and large movement in one single execution. Hard constraints may lead to a problem with feasibility. Furthermore, hard constraints may destabilise an otherwise stable system (Hovd, 2004). This phenomenon is quite rare, and can be solved by using a soft constraint formulation for the output constraints.

CV and MV trajectories

MPC uses three different options to specify the future CV behaviour: a setpoint, a zone or a reference trajectory. These are illustrated in Fig. 4.2.4.

All MPC controllers provide the possibility of driving a certain variable to a setpoint, with deviations on both sides that are penalised in the objective function. In general, using a constant setpoint may lead to aggressive behaviour with very large input adjustments. This is especially important when the process model differs significantly from the real plant behaviour. Constraints on the input change can be added to avoid this behaviour, or the setpoint is dynamic, a trajectory for the MPC to follow that leads from the current setpoint to the desired steady state value (Qin and Badgwell, 2003).

Also, all controllers provide the possibility of keeping the CV inside a zone defined by upper and lower boundaries. One way to implement a zone is by applying soft lower and upper constraints on the CV, but other options are also available. Reference trajectories are used when the CV is required to follow a specific path. This can be calculated as a first or second order curve drawn from the current value of the CV to the setpoint, with the speed of the response determined by one or more trajectory time constants.

Another option is to use a precalculated trajectory that is given to the MPC. CV deviations from this trajectory will be penalised. When the trajectory is calculated to a reach a spe-

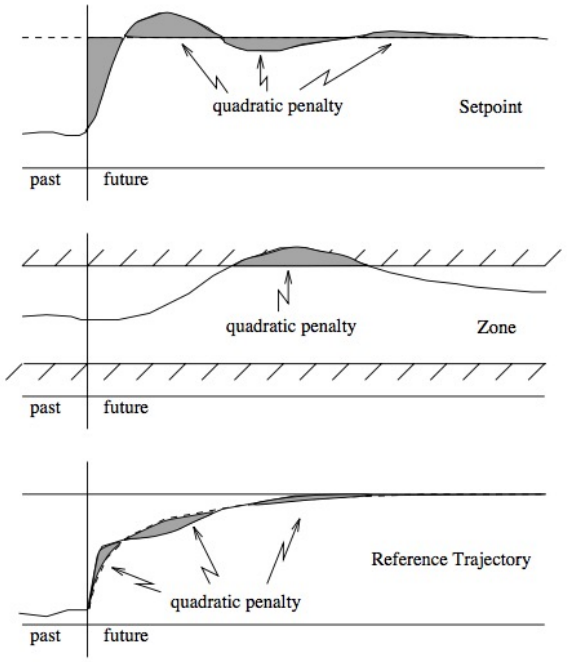

Figure 4.2.4: An illustration of the three options for predicting future CV behaviour; setpoint, zone or reference trajectory (Qin and Badgwell, 2003).

cific setpoint, one drawback with the use of a reference trajectory is that it will penalise the CV for moving too quickly towards the setpoint. This may happen due to unmeasured disturbances. However, if the CV moves too quickly due to model mismatch, the penalisation is beneficial, as it slows the response down and avoids a large overshoot. A similar setpoint or trajectory may be applied to the MVs (Qin and Badgwell, 2003).

Horizon Parametrisation

The industrial MPC usually predicts the future behaviour of the CVs over a finite set of future time intervals, called the *prediction horizon*. The length of this horizon, P, is a basic tuning parameter, and is chosen long enough to capture the steady state effects of the computed MV moves. The prediction horizon is illustrated in the top figure of Fig. 4.2.5.

Many MPC controllers use a multiple point output horizon. This means that all variables are calculated for each time interval, i.e. at every sample, and the number of variables to calculate will increase linearly with the prediction horizon. However, if the optimisation problem is large, the amount of variables to calculate is enormous. To reduce the computation time, the variables are only calculated at certain points, called *coincidence points*. This concept is illustrated at the bottom of Fig. 4.2.5. The coincidence points can be cho-

sen separately for each CV, depending on which ones are more important to monitor and if they need to respond quickly.

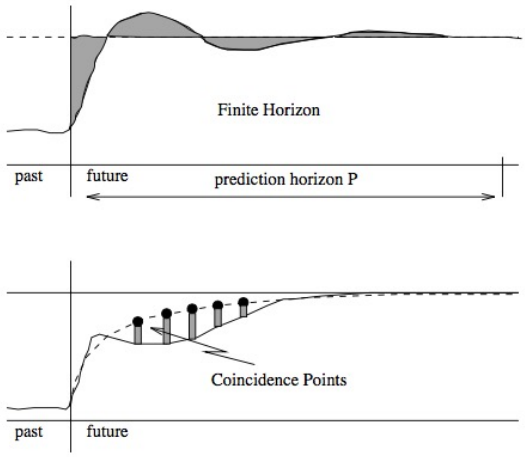

Figure 4.2.5: An illustration of the finite prediction horizon and a subset of the horizon, called coincidence points (Qin and Badgwell, 2003).

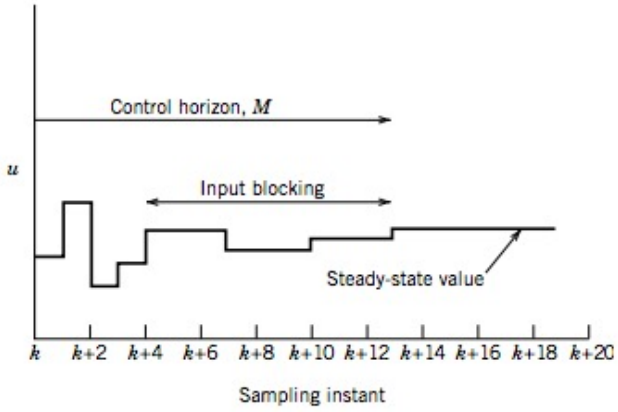

Figure 4.2.6: An illustration of input blocking to reduce computation time (Seborg et al., 2010).

A similar selection of points in the time horizon can be done to the MV profile. The original way is for the MPC controller to calculate the MV trajectory at every point in the *control horizon*, of length M. Performance increases with M, but so does the computational cost. MV parametrisation can be done and is called *input blocking*. It is up to the engineer to determine the number of the blocks and the length of the intervals. The number and length of the MV intervals is chosen to approximate a desirable predictive curve shape of the specific MVs. Control input blocking will reduce runtime (Foss and Heirung,

2013). Fig. 4.2.6 is an illustration of the concept of input blocking, with fine blocking in the beginning and larger blocks towards the end of the time horizon.

Zero and First Order Hold

The parametrisation can be of zero or first-order hold. If the variable is of *zero-order hold*, the variable is blocked for the specified sample period. The zero-order hold is an interpolation between sample points by holding each sample until the next sample point. This generates a staircase-like approach. First-order hold is linear interpolation, and connects the sample points by straight line segments. These types of interpolation are used frequently in signal treatment (Oppenheim, 2011). In this case it is used in the predictions of the input variables. By default, the input variables are zero order hold, resulting in the staircase-like prediction, as seen in Fig. 4.2.7. Sometimes, however, it is best to have a smooth input trajectory, and in this case one can apply the first-order hold. A smooth input trajectory can be advantageous if the input variable has direct influence in the optimisation problem, as it does in this work. It results in less aggressive behaviour.

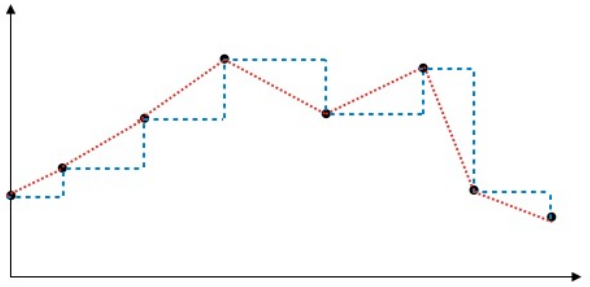

Figure 4.2.7: An illustration of zero and first-order hold on the input parametrisation, in blue and red, respectively.

Weights and Tuning

Recall the standard MPC formulation given in Eq. 4.2.4. This can be slightly reformulated, taking the MV and CV parametrisation into consideration. Instead of evaluating the MVs and CVs at every time sample, it is only done in the specified samples. Furthermore, the CVs (z) are often included directly in the objective function, and not the states (x). The reformulation of the objective function on a discrete-time form is shown in Eq. 4.2.15:

$$J = \frac{1}{2}\sum_{j=1}^{P}\sum_{i=1}^{n_z}(z_{k+j,i} - z_{ref,k+j,i})^2 Q_{j,i} + \frac{1}{2}\sum_{j=0}^{M-1}\sum_{i=1}^{n_u}\Delta u_{k+j,i}^2 S_{j,i} \tag{4.2.15}$$

where P is the number of evaluation points for the CV, and M is the number of MV blocked intervals. $Q_{j,i}$ is the CV weights and $S_{j,i}$ the MV move penalties, and n_z and

n_u are the numbers of CVs and MVs, respectively. The weighting matrices Q and S are diagonal:

$$Q = \begin{pmatrix} Q_1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & & & \vdots \\ \vdots & \ddots & Q_{n_z} & \ddots & & & \vdots \\ \vdots & & \ddots & Q_1 & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & & \ddots & Q_{n_z} & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & \ddots \end{pmatrix}$$

and

$$S = \begin{pmatrix} S_1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & & & \vdots \\ \vdots & \ddots & S_{n_u} & \ddots & & & \vdots \\ \vdots & & \ddots & S_1 & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & & \ddots & S_{n_u} & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & \ddots \end{pmatrix}$$

These matrices do not take the sample subscript j into account. This is because they are usually constant throughout the prediction horizon. The dimensions of the matrices are:

$$dim(Q) = (P * n_z) \times (P * n_z) \tag{4.2.16a}$$
$$dim(S) = (M * n_u) \times (M * n_u) \tag{4.2.16b}$$

The weights Q_i and S_i are determined by the engineer corresponding to each CV and MV. For example, z_1 is the temperature and z_2 is the pressure, each to be kept at its setpoint. However, it is more important that the temperature is kept at its setpoint than the pressure. Choosing $Q_1 > Q_2$ will penalise a temperature deviation from its reference harder than a pressure deviation.

However, this objective function will not ensure feasibility at all times, i.e. at some point the constraints will be broken. To avoid infeasibility, *slack variables* are added to the objective function.

$$\min J = \frac{1}{2}\sum_{j=1}^{P}\sum_{i=1}^{n_z}(z_{k+j,i} - z_{ref,k+j,i})^2 Q_{j,i} + \frac{1}{2}\sum_{j=0}^{M-1}\sum_{i=1}^{n_u}\Delta u_{k+j,i}^2 S_{j,i} \tag{4.2.17a}$$

$$+ r_1^\top \epsilon + \frac{1}{2}\epsilon \text{diag}(r_2)\epsilon$$

s.t.
$$x_{i+1} = g(x_i, u_i) \tag{4.2.17b}$$
$$x_0, u_{-1} = \text{given} \tag{4.2.17c}$$
$$z^{low} - \epsilon \leq x_i \leq x^{high} + \epsilon \tag{4.2.17d}$$
$$0 \leq \epsilon \leq \epsilon^{max} \tag{4.2.17e}$$
$$u^{low} \leq u_i \leq u^{high} \tag{4.2.17f}$$
$$\Delta u^{low} \leq \Delta u \leq \Delta u^{high} \tag{4.2.17g}$$

where ϵ is the slack variable and r_1 and r_2 are the tuning parameters. ϵ^{max} is usually set to infinity to avoid infeasibility. r_1 and r_2 are penalties for crossing CV limits. Both the added terms including the slack variable are positive terms, hence the desire to drive them to zero. Said differently, the slack variables are only non-zero if the constraints are violated (Foss and Heirung, 2013).

4.2.2 State Estimation

An important factor for the MPC is that it uses the current variable values and measurements at each time sample as the initial conditions to start the predictions. However, in most chemical processes it is not possible to measure all states on-line. Variables that are easy to measure include pressure, flow and temperature. Concentration measurements are infrequent and with a time delay, leading to inaccuracy in the model. State estimation is used to estimate the states used in the model at each time sample. Qin and Badgwell (2003) call the state estimation the output feedback stage, and the controller uses the current states to estimate the dynamic states of the system. This could be because a measurement is not available at the exact sample time, but comes with some time delay. When using only predictions of the states, cumulative effects of model inaccuracy and disturbances that are not measured may lead to inaccurate predictions. The lack of on-line measurements has been a driving force to develop state estimators. The major results in state estimation was done by Kalman (1960,1961) and Leuenberger (1964) (Krämer, 2005).

A state estimator works with a dynamic mathematical model that is simulated alongside the real process using the same inputs and initial conditions (that are known) as the real process. The model simulates the measurements, and the simulated measurements are compared to the real measurements. The error between the measurements is sent back to the model and used as a feedback correction term. Without this feedback, there is no guarantee that the simulated states are equal to the real ones.

Several methods exist to design the feedback function for linear and nonlinear systems. For nonlinear systems, there are three subgroups of estimators: (1) Differential-geometric

approaches, (2) approaches with guaranteed stability and (3) approaches based on approximations, mainly linearisations. Differential-geometric approaches require a special class of nonlinear models that are often met by mechanical systems, but not by chemical systems. A *proportional observer* was proposed by Lopez-Arenas et al. (1997) for a chemical system, where the states were detectable and stable but not observable. There are two examples of state estimators that guarantee stability, the *High Gain Observer* and the *Sliding Mode Observer*. The high gain observer has a disadvantage towards measurement noise. It divides the model into a linear dynamic part and a nonlinear Lipschitz continuous part. The sliding mode observer approach divides the system into an observable linear part, nonlinear terms dependent on the measurements, and nonlinear terms not described by measurements. It guarantees stability with a large design effort. State estimators based on linear approximations of the models include the *Extended Kalman Filter (EKF)*, the *Extended Leuenberger Observer* and the *Moving Horizon Estimator*. All three of these use linearisations of the nonlinear system around the current time sample. Because of the nonlinear dynamics of the model, the linearisation is only valid in the nearest proximity to the operating point and only local stability can be considered. The EKF is the most-used approach, despite having non-intuitive tuning (Krämer, 2005).

State Estimation Concepts

Considering a linear system written as:

$$\dot{x}(t) = Ax(t) + Bu(t), \qquad x(0) = x_0 \tag{4.2.18a}$$

$$y(t) = Cx(t) \tag{4.2.18b}$$

$$\text{with} \qquad x \in \mathbb{R}^n, A \in \mathbb{R}^{n\times n}, u \in \mathbb{R}^m, B \in \mathbb{R}^{n\times m}, y \in \mathbb{R}^q, C \in \mathbb{R}^{n\times q} \tag{4.2.18c}$$

The state estimator for this system can be written as:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + K(y(t) - \hat{y}(t)), \qquad \hat{x}(0) = \hat{x}_0 \tag{4.2.19a}$$

$$\hat{y}(t) = C\hat{x}(t) \tag{4.2.19b}$$

$$\text{with} \qquad \hat{x} \in \mathbb{R}^n, A \in \mathbb{R}^{n\times n}, u \in \mathbb{R}^m, B \in \mathbb{R}^{n\times m}, \hat{y} \in \mathbb{R}^q, C \in \mathbb{R}^{n\times q} \tag{4.2.19c}$$

where the hat symbol $\hat{x}$ denotes the estimated value, and K is the estimator gain. The different state estimators uses different algorithms to calculate the estimator gain. If K is small, the contributions from the difference between the measurements and the estimated measurements on the estimation of the state variables are small. This is normally the case if the precision of the measurement is poor, or the model is considered to be very accurate. And vice versa, with a large value of K, the opposite is true, and the measurements and estimated measurements are very reliable. It is important to note that the value of K may change over time and depend on the measured outputs.

Definition of a State Estimator

Krämer (2005) gives two conditions that the state estimator has to fulfil. If the initial state is known, the estimator reconstructs the process. If the initial state is not known, or inac-

curate, the state estimator should converge to the correct state, i.e. that the initial states are re-constructable.

Definition 4.5. ***Simulation condition****: For identical initial conditions of the system and the observer* $(\hat{x}(t=0) = x(t=0))$ *it follows that* $\hat{x}(t) = x(t), \quad \forall t > 0$.

Definition 4.6. ***Convergence condition****: If* $\hat{x}(t=0) \neq x(t=0)$ *for* $t \to t_\infty$*, the estimator error* $(e(t) = \hat{x}(t) - x(t))$ *has to converge to zero* $(e \to 0)$*. One can equivalently write, for a shorter time:*

$$\forall \epsilon > 0 \quad \exists T \quad \textit{such that} \quad \frac{||e(t)||}{||e(T)||} < \epsilon \qquad \forall t < T \tag{4.2.20}$$

Detectability and Observability

The system has to be observable for state estimation to be applied. In his thesis, Krämer (2005) gives the definition of observability in a system, as stated Def. 4.7.

Definition 4.7. ***Observability:*** *A linear system as given in Eq. 4.2.18 is observable if there is a finite time T such that the knowledge of* $u(t)$ *and* $y(t)$ *that is available is sufficient to estimate the initial state* $x(0)$.

For a nonlinear system, observability is divided into global and local observability.

Definition 4.8. ***Global observability:*** *A nonlinear system is globally observable, if all initial states* $x_0 \in \mathbb{X}_0$ *are observable for all inputs* $u(t)$.

Definition 4.9. ***Local observability:*** *A nonlinear system is locally observable in* $x_1 \in \mathbb{X}_0$ *if the initial states* $x_0 \in \mathbb{X}_0$ *are observable around* x_1.

$\mathbb{X}_0$ is the set of all possible initial state vectors, and a subset of $\mathbb{X}$, the set of all solution vectors of the system. Simply put, the system is observable if the measurable output contains information on all the state variables. Local observability is important for nonlinear systems operated at specific conditions. In this case the nonlinear system is linearised around the point, and the matrices A and C from Eq. 4.2.18 are the Jacobians of the nonlinear model equations. For systems that are not operated around a single operating point, global observability should be checked. This is especially important for batch and semi-batch processes. However, it can generally not be shown if a nonlinear system is globally observable. In these cases, observability is analysed using the linearised system.

In polymerisation, the MWD is not observable by using the available measurements of monomer conversion and temperature, and must therefore be calculated using only the prediction model (i.e. $K = 0$). However, since many mathematical models for polymerisation are robust, it has been shown that accurate estimation of copolymer properties is possible (Asua, 2008).

4.2.3 The Kalman Filter

When it comes to state estimation for nonlinear systems there is no single solution available that is clearly better than other strategies. Many strategies for nonlinear estimation have been presented over time, most of them being extensions of the celebrated Kalman Filter (KF). One has to choose the estimator that gives the best trade off between estimation accuracy, numerical robustness, computational burdens and ease of implementation. The most common estimator for nonlinear processes is known as the Extended Kalman Filter (EKF), which was first presented by Gelb, Kasper, Nash, Price and Sutherland in 1974. The EKF is based on a first order Taylor approximation of state transitions and observation equations about the state trajectories. Therefore, application of the filter implies assumptions that the required derivatives exist and that they can be obtained within reasonable effort (Nørgaard et al., 2000). The Taylor approximations produce insufficiently accurate representations of the state variables in many cases, and this may lead to quite a large bias or convergence problems.

A continuous time Kalman Filter can be written in the following form:

$$\dot{x} = Ax + Bu + \xi(t) \qquad \xi \sim \mathcal{N}(Q, 0) \tag{4.2.21a}$$
$$x(0) = x_0 + \xi_0 \qquad \xi_0 \sim \mathcal{N}(P_0, 0) \tag{4.2.21b}$$
$$y(t) = Cx + \phi(t) \qquad \phi \sim \mathcal{N}(R, 0) \tag{4.2.21c}$$

where x, u, y, ξ and ϕ are the n-dimensional state vector, the r-dimensional vector of the control variables and the m-dimensional vectors of the measurement errors, respectively. ξ and ϕ are assumed to be normally distributed zero mean random processes. The Kalman Filter minimises the sum of the expected values of the quadratic estimation error:

$$J = \sum_{i=1}^{n} E((\hat{x}_i - x_i)^2) \tag{4.2.22}$$

There are two ways to derive the Kalman Filter: As Kalman did it himself from stochastic interpretation, or the minimisation of a pseudo linear quadratic regulator problem. Both methods give the same results, however the latter provide covariance matrices, **Q** and **R**, that are mere weighting matrices and have no stochastic significance.

The KF equations are a coupled ODE system given by:

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}) \tag{4.2.23a}$$
$$\dot{P}(t) = AP(t) + P(t)A^\top + Q - P(t)C^\top R^{-1} CP(t) \tag{4.2.23b}$$
$$K(t) = P(t)C^\top R^{-1} \tag{4.2.23c}$$

Definition 4.5 is fulfilled if the model is accurate and Definition 4.6 is fulfilled if the matrix P_0 is symmetric and positive semi definite, and the matrices **Q** and **R** are symmetric and positive definite. In most cases, only the non-diagonal elements have non-zero values, indicating from a stochastic viewpoint that the error terms in the states and the measurements are independent. The Kalman Filter is tuned by setting the **R** matrix according to the

measurement error and by finding **Q** through trial and error. **P** is found using engineering judgement (Krämer, 2005).

For a discrete Kalman Filter, the ordinary differential equations have to be transformed into difference equations by integrating over one sampling interval:

$$x_{k+1} = x_k + \int_{t_k}^{t_{k+1}} (A^c x + B^c u + \xi^c) dt \tag{4.2.24a}$$
$$= Ax_k + Bu_k + \xi_k$$
$$y_k = Cx_k + \phi_k \tag{4.2.24b}$$

where the index c denotes the continuous form. Kalman originally derived his equations on a discrete form. The discrete Kalman Filter is also based on the minimisation of the difference between the measured and estimated value, and as such it is optimal for the provided covariance matrices. The discrete Kalman Filter can be written in a predictor-corrector form:

Correction: The correction uses the predictions from the last step $(\hat{x}_{k|k-1}, P_{k+1|k})$ and the current measurements to correct the prediction.

$$K_k = P_{k|k-1}C^\top (CP_{k|k-1}C^\top + R)^{-1} \tag{4.2.25a}$$
$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - C\hat{x}_{k|k-1}) \tag{4.2.25b}$$
$$P_{k|k} = (I - K_k C_{k|k-1})P_{k|k-1} \tag{4.2.25c}$$

Prediction: The prediction uses the undisturbed model to predict the states for the next time step $\hat{x}_{k+1|k}$ and the solution of the optimisation to predict the covariance matrix of the measurement error $P_{k+1|k}$:

$$\hat{x}_{k+1|k} = C\hat{x}_{k|k} \tag{4.2.26a}$$
$$P_{k+1|k} = AP_{k|k}A^\top + Q \tag{4.2.26b}$$

Extended Kalman Filter (EKF)

For nonlinear systems, like polymerisation reactions, the model yields nonlinear differential equations as given in Eq. 4.2.27.

$$\dot{x} = f(x,u) + \xi(t) \qquad \xi \sim \mathcal{N}(\mathbf{Q}, 0) \tag{4.2.27a}$$
$$x(0) = x_0 + \xi_o \qquad \xi_0 \sim \mathcal{N}(P_0, 0) \tag{4.2.27b}$$
$$y(t) = h(x) + \phi(t) \qquad \phi \sim \mathcal{N}(\mathbf{R}, 0) \tag{4.2.27c}$$

ξ and ϕ are the vectors for model error and measurement errors, respectively, and are assumed to be zero mean random processes, described by the covariance matrices P_0 for the initial condition, **Q** for the model error and **R** for the measurement error.

The derivation of the Extended Kalman Filter can be found in numerous textbooks. The EKF approximates the optimal filter around a linearisation of the nonlinear system. The

discrete EKF is based on the solution of the minimised expected values of the estimator error and is therefore a first order approximation of the optimal filter for the provided covariance matrices. The EKF equations can be written, as for the discrete KF equations, on a predictor-corrector form. It is common to sort the variables into *a priori*, before correction, and *a posteriori*, after correction, variables. The equations for the EKF are listed below:

Correction: The correction uses the predictions from the last step $(\hat{x}_{k|k-1}, P_{k+1|k})$ and the current measurements to correct the prediction.

$$K_k = P_{k|k-1}H^\top(H_{k|k-1}P_{k|k-1}H_{k|k-1}^\top + R)^{-1} \tag{4.2.28a}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - h\hat{x}_{k|k-1}) \tag{4.2.28b}$$

$$P_{k|k} = (I - K_k H_{k|k-1})P_{k|k-1} \tag{4.2.28c}$$

Prediction: The prediction uses the undisturbed model to predict the states for the next time step $\hat{x}_{k+1|k}$ and the solution of the optimisation to predict the covariance matrix of the measurement error $P_{k+1|k}$ (Krämer, 2005):

$$\hat{x}_{k+1|k} = F(\hat{x}_{k|k}, u_k) \tag{4.2.29a}$$

$$P_{k+1|k} = A_{k|k}P_{k|k}A_{k|k}^\top + Q \tag{4.2.29b}$$

with

$$A_{k|k} = \frac{\partial F}{\partial \hat{x}}\Big|_{\hat{x}_{k|k}} \tag{4.2.30a}$$

$$H_{k|k-1} = \frac{\partial h}{\partial \hat{x}}\Big|_{\hat{x}_{k|k-1}} \tag{4.2.30b}$$

The specific algorithm deployed by the Cybernetica CENIT software is a Divided Difference Kalman Filter, as presented in Nørgaard et al. (2000). Conceptually this is very similar to the EKF, however the implementation is quite different. The EKF is based on first-order Taylor approximations of state transitions and observation equations about the estimated state trajectory. To apply this filter assumptions must be made that the required derivatives exist and that they can be obtained within reasonable effort. Furthermore, the Taylor linearisations provide insufficient accuracy in many cases, resulting in a significant bias or even convergence problems. The divided difference Kalman filter uses an interpolation formula, that unlike the Taylor approximation does not require derivatives, only functional evaluations (Nørgaard et al., 1998). This allows for easy implementation of the filters, and enables state estimation even in the sample points in which the derivatives are undefined. However, the computational burden will be equal to or higher than for the EKF. The difference between the EKF and the divided differences Kalman filter lies in the matrices, which for divided differences replace the matrix products of Jacobians in the EKF by covariance matrices. This gives the same state update as for the EKF, and the difference alone lies in the update of the covariance matrices.

Tuning of the Kalman Filter

The initial enthusiasm when the Kalman Filter was introduced quickly diminished since the noise statistics had to be provided to design the filter. In general, the noise statistics are tweaked until satisfactory behaviour is acceptable instead of tuning it properly to achieve better results (Naik et al., 2015). When Kalman himself proposed the filter, he required the statistics of the process and the measurement noise to be known, and he only dealt with state estimation. Tuning the filter varies from ad hoc, heuristic and rigorous methods, though all include obtaining the filter statistics of P_0, $\mathbf{Q}$ and $\mathbf{R}$. The ad hoc method is an arbitrary tuning and may lead to inaccurate results. The rigorous approach is often very time consuming. The heuristic approach offers an appealing middle path between inaccurate results and hard exact solutions (Naik et al., 2015). The approach to tuning is the same for the Kalman Filter and the EKF. The covariance matrices determine how much process noise, measurement noise and integrated white noise there is in the system.

If $P_0 = \mathbf{Q} = 0$ then the filter does not learn anything from the measurements. The $\mathbf{R}$ matrix can be determined from the measured data. The P_0 can be tricky: usually all the off-diagonal elements are set to zero while the diagonal elements are set relative to each other. The choice of P_0 is important, as it will determine the steady state filter behaviour given by $\mathbf{Q}$. $\mathbf{Q}$ is also difficult to determine, it injects uncertainty into the state equations, aiding the filter in learning from the measurements. It also controls the steady state filter response. A large value of $\mathbf{Q}$ will lead to a large steady state uncertainty of the estimates, and vice versa for a small value of $\mathbf{Q}$ (Naik et al., 2015). In general the filter parameters are tuned off-line using simulated data before application to on-line and real-time applications.

The aim of the tuning is to obtain a quick response and correction of the parameters. However, the Kalman filter must not be so aggressive that it follows the measurement noise. The higher the covariance on the process noise, the more aggressive behaviour, since the measurements are considered to be quite certain. Likewise, a small covariance on the measurement noise means that there is little error on the measurement and will lead to aggressive behaviour of the Kalman Filter estimation.

CHAPTER

FIVE

CONTROL OF POLYMERISATION PROCESSES

Latex products are often composed by more than one polymer. The copolymer chains are formed in an emulsion by simultaneous reaction of the monomers. The copolymers have different properties to the polymers that are built up of only one kind of monomer (homopolymer latexes). The desired properties of the polymer is dictated by the market. Typically the industrial production of emulsion copolymers use monomers that give hard latexes and monomers that lead to soft latexes. Styrene and methyl methacrylate are two monomers that give hard latexes[1], and butyl acrylate will give soft latexes. By choosing the amount of each of the monomers, controlling the distribution of them and the molecular weight distribution of the monomer allows for the production of many different emulsion polymer products. Including acrylic and methacrylic acid to the emulsion will significantly improve the colloidal stability of the latex product. Batch, semi-batch and continuous reactors are used in emulsion polymerisation. Typically, these are well-stirred tank reactors, and the semi-batch reactor is mostly used because of its versatility. However, tubular reactors provide a large heat transfer area/reactor volume ratio in continuous operation which is attractive. This is not much used in emulsion polymerisation due to a high risk of phase segregation, fouling and pipe clogging (Leiza and Meuldijk, 2013).

This chapter will focus on the main issues of controlling a polymerisation process, mainly regarding the control of the reactor temperature and feed flow rate. The difficulties considering on-line measurements and the lack thereof will be reviewed, as well as the reason for implementation of a bilevel MPC structure. Finally, some safety issues in the control of polymerisation processes will be mentioned.

[1] Hard latex: polymer with a high glass transition temperature T_g.

5.1 Optimisation of Batch Reactors in the Process Industry

In the early days, the objective of polymerisation control was to make theories of particle nucleation and growth and use this to predict and control the process. With modern digital computer technology, modelling for process control has become increasingly realistic. The chemical industry has tougher competition, which makes process optimisation a natural choice to reduce costs, increase production and improve product quality. The objective for optimising batch and semi-batch operation is often economic, i.e. to reduce operational costs. The optimisation problem will commonly give optimal feed rates and temperatures, and take multiple operational constraints into consideration.

In a polymerisation process, there may be multiple, conflicting objective functions. The optimisation technology to handle multiple variables is relatively new, it emerged the latest half of the twentieth century. Most of the companies that conduct emulsion polymerisation have developed their own technology that is confidential. Therefore, research papers and articles on clever computer control procedures and modelling are limited. Another discord between the scientific community and the industrial processes is the difference in objective. The scientific research in emulsion polymerisation is to a large extent about the kinetics and mechanisms. While kinetic models are important in control of the process, they only predict the molecular level properties and not the macroscopic properties that are important to the industry.

Min and Ray did a considerable amount of preliminary work and presented a pioneering modelling framework in 1974-1978. The main challenge when modelling is to find appropriate model simplifications while the model is sufficiently accurate. A sufficiently accurate model may be used in model predictive control. Extensive research has been performed on this topic the last 10-15 years. The easiest models of batch reactors are those for lab- or pilot sized reactors. The main challenge when up-scaling the reactor model is the temperature control. When the reactor increases in size, the surface/volume ratio decreases, turning sufficient heat transfer into an issue.

5.2 Characterising the Control Problem

As mentioned in the previous chapters, control of polymerisation reactions is complicated, and the reasons for this are multiple. First of all, polymers are performance materials, meaning that its market value depends on the polymers end-of-use properties, i.e. the market demands the polymer product quality. This includes rheological characteristics and transition temperatures as well as mechanical characteristics. The polymerisation operation conditions must therefore ensure that the end-product quality is attained. If one or more of the desired product qualities are not met, the market value of the polymer product will decrease. Another difficulty linked to the product quality is that once one product property is correct it might have worsened another desired property. An example is for homopropylene, where rubber is introduced to increase the impact of resistance on ho-

mopropylene resins. However, this will decrease the flexural modulus of the bend (the tendency for a material to bend), which is undesirable. The optimal production will therefore have to take into account a trade-off between two properties.

The relationship between the molecular structure and the end-product quality is not well understood, and is based on empirical observations and testing. As a result, the phenomenological models will not completely reflect how operation variables affect the molecular structure of the polymer, and so also the polymer quality. Furthermore, the relationship between the operation variables and the polymer properties are highly nonlinear, which means that the well-developed linear optimisation theory is of little use when applied to polymer reactions. As a result, advanced nonlinear optimisation theory is used in most polymer industry.

The polymer reaction is susceptible to different instabilities, which may be caused by for example viscous, thermal and kinetic effects. For example, if the system viscosity increases uncontrollably, the heat transfer coefficient decreases drastically, and the heat removal is not sufficient. This is a safety issue, and to avoid this safety procedures must be present to guarantee that the process will be steered into safe operating conditions.

The initial decision of polymerisation control lies in defining the objectives, constraints and decision variables. Most of the polymer product properties are not measurable online. This means that control procedures cannot rely directly on these properties, but they have to be estimated by using measurable variables and process models.

5.3 Polymerisation Reaction Control Problems

The control of the polymerisation reaction is challenging, even though the the semi-batch reactor provides a great advantage of flexibility in the process. For emulsion polymerisation the products can be tailored to fit specifications in composition, molecular weight distribution and particle morphology. The drawback with using a semi-batch reactor is relatively low productivity. This is compensated by using large reactors.

The semi-batch reactor will contain an initial charge, that consists of water, emulsifier, some initiator and a seed. The seed is used to avoid the lack of reproducibility of the nucleation stages, when the seed is produced *in situ*. Also, the nucleation stage is highly scale sensitive. The rest of the ingredients are fed into the reactor at an optimal feed flow rate.

The amount of initial seed is less than 5 wt.% of the total polymer in the final product, and therefore its properties are negligible compared to the copolymer properties. However, the amount of polymer and the size of the seed are important. For a given amount of seed, the smaller the particle size the higher the number of particles, and thus the higher specific surface area per unit volume of water. A higher specific surface area per unit will decrease the likelihood of a secondary nucleation during the monomer addition period.

When it comes to the effect of the amount of emulsifier, Leiza and Meuldijk (2013) states that the higher the concentration of emulsifier in the recipe, the higher the number of poly-

mer particles and the lower the particle size. If all of the emulsifier is loaded with the initial charge, there is a possibility of coagulation of the particles in the reaction mixture. This is explained by particle growth with simultaneous polymerisation and monomer absorption, and the fractional surface coverage of emulsifier will decrease when no more emulsifier is added to the mixture. As a result of the reduced fractional surface coverage, coagulation will occur for the particles with emulsifier below its critical value for colloidal stability. Therefore, it is recommended to split the total amount of emulsifier to the initial charge and the feed. This will provide enough emulsifier to avoid the fractional surface energy to be reduced under its critical value for colloidal stability during the monomer addition period.

The amount and distribution of initiator will also impact the system, however it will effect the product properties different from system to system. Leiza and Meuldijk (2013) claims that the feed of initiator generally is not a good control variable. However, the radical production rate by initiator decomposition will influence the rate of formation of oligomer radicals in the aqueous phase, and so the entry rate of radical oligomers into the particles. The time a polymer chain will propagate until it is terminated is directly linked with the entry rate, and by that on the radical production rate which is linked to the initiator concentration. Therefore, the initiator concentration will influence the particle size distribution. Furthermore, Leiza and Meuldijk (2013) state that care should be taken when introducing initiator in the feed stream. If the local concentration of initiator is too high, this may lead to coagulation of the latex, as the colloidal stability limit can be exceeded. For colloidal stability, fast mixing of the feed stream with the reactor contents is a necessity, and this can be achieved by feeding close to the impeller tip.

The monomer that is not included in the initial charge in the reactor will be fed into the reactor at a predefined feed rate profile. The monomer may be added neat, or as monomer pre-emulsified in water. Leiza and Meuldijk (2013) states that the polymerisation rate and the copolymer properties in terms of molecular weight will be different from feeding neat monomer and pre-emulsified monomer. This is due to the monomer transport from the monomer phase to the polymer phase and the reacting polymer particles. When neat monomer is fed into the reactor, the resistance against monomer transport between the two phases is larger compared to the case of feeding pre-emulsified monomer. The reason for this difference lies in the specific mass transfer area per volume of the aqueous phase, which is considerably smaller when neat monomer is fed, compared to pre-emulsified monomer. For pre-emulsified monomer feed, the polymerisation is governed on polymerisation kinetics, whereas for a neat monomer feed the polymerisation is governed by mass transfer limitation. As a consequence, the molecular weight control is better for a pre-emulsified monomer feed than a neat monomer feed.

5.3.1 Control of Reactor Temperature

When the reaction is exothermic, a cooling jacket is added to the reactor to transfer the excess heat away. This jacket may be designed in several ways, but it all includes a colder fluid, mostly water, that circulates around the reactor. Sometimes there are also cooling

coils inside the reactor. The cooling capacity of the cooling jacket depends on the temperature of the water and the reactor, the flow of the water and the specific heat transfer coefficient that says something about how easily heat is transferred in the material between the two fluids.

Temperature control of the reactor becomes an issue when the size of the reactor increases, mostly when the reactor size exceeds $10m^3$ (Leiza and Meuldijk, 2013). A large reactor volume indicates a smaller reactor wall heat transfer area at the reactor wall per unit volume of reaction mixture. This means that the heat transfer rate per unit volume decreases when up-scaling the reaction. This is shown in Eq. 5.3.1.

$$\dot{Q}_{transfer} = \frac{UA}{V_W}(T_R - T_J) \tag{5.3.1}$$

where $\dot{Q}_{transfer}$ is the heat transferred per unit time, UA is the heat transfer coefficient, V_W is the volume of the jacket and T_R and T_J are the temperatures of the reactor and the jacket, respectively. The heat generation can be controlled by the monomer concentration in the polymer particles. This is done by manipulating the monomer feed rate: The lower the feed rate, the lower the concentration of monomer in the particle phase, the lower the reaction rate the lower the heat generation. In a semi-batch reactor it is possible to achieve pseudo-steady-state conditions when the monomer feed rate is low. At this point, the monomer feed rate equals the polymerisation rate, and so the monomer concentration in the particle remains constant. These conditions are known as *starved conditions*, and are illustrated in Fig. 5.3.1. Under these starved conditions, the heat generated is generally lower than the heat transfer, and so the reactor temperature can be controlled by manipulating the temperature of the jacket.

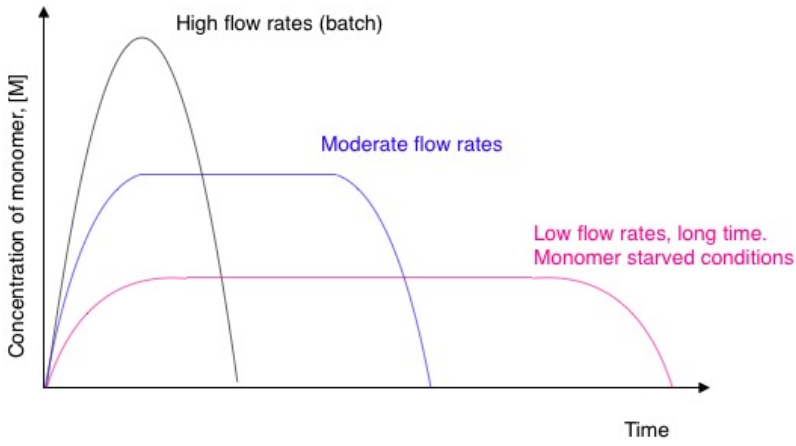

Figure 5.3.1: An illustration of the concentration of monomer droplets for a high, moderate and low feed flow rates of monomer. Low feed rates are called monomer starved conditions.

Fig. 5.3.1 also shows the monomer concentration when the monomer feed rate is a lot larger than the polymerisation rate, i.e. almost batch reactor conditions. During these conditions, the heat generated in the reactor from the polymerisation may exceed the maximum heat transfer, $UA(T_R - T_J^{min})$, where T_J^{min} is the minimum temperature that the fluid in the jacket may have. If this is the case, that the heat generated exceeds the cooling capacity, the reactor temperature increases and may lead to a thermal runaway. The lower the monomer feed rate, the longer the batch time and the lower the productivity.

5.3.2 Control of Molecular Weight Averages and Molecular Weight Distribution (MWD)

Other than to ensure safe and economic operation, the aim of controlling a polymerisation reaction is to obtain the right end-product quality at the end of the batch. The molecular weight distribution of the polymer has significant impact on the polymer end-use properties. It is dependent on the reactor operating conditions, which change over time in a semi-batch reactor. The molecular properties of the polymer may as a consequence change continuously.

In most cases on-line measurement of end-product quality is not possible. The mathematical model must therefore relate the measurable variables to the end-use properties. The molecular weight of the polymer is directly linked to the polymer chain length, and therefore with all the kinetics and mechanisms of polymerisation and chain growth. Both termination and transfer suppress the growth of the polymer. Unless the transfer is to a dead polymer chain, it does not affect the total number of radicals in the polymer particle, and does therefore not affect the rate of polymerisation. In termination by combination, two polymer chains meet and form one long chain, while termination by disproportionation creates one longer and one shorter polymer chain. The type of termination will influence the MWD (Dimitratos et al., 1994).

The molecular weight distribution has been attempted controlled for several homogeneous polymerisation processes. Controlling the MWD in an emulsion copolymerisation process proves to be even more complicated due to the compartmentalisation of the system. Also radical entry and exit to the polymer particle has to be considered, in addition to the termination and the transfer reactions (Dimitratos et al., 1994). Control of the molecular weight distribution is normally obtained by manipulating the concentration of chain transfer agent (CTA). However, the reactor temperature, the initiator and monomer concentration as well as the batch time may also be used to control the MWD. Asua (2008) gives a thorough description of the control of the MWD, and states that when the rate constant ratio between propagation and transfer to CTA is dependent on the temperature, it is possible to use the reactor temperature to control the MWD. However, Asua (2008) does not recommend using temperature to control the MWD, as it has a slow response, and can also be a safety issue. Nevertheless, this is exactly what is being done in this study. The reason for this is that the reactor temperature is easy to measure on-line, and gives reliable measurements. Also, as Dimitratos et al. (1994) explains, the MWD is directly linked to the kinetics, which in turn depends on the reactor temperature.

Furthermore, Asua (2008) states that the initiator or monomer feed is not recommended for controlling the MWD either, because these are also strongly coupled with the energy balance terms, like the reactor temperature. For these reasons, Asua (2008) highly recommends using the CTA to control the MWD. However, this may cause some problems with the reactor mixing. Recent developments in controlling free-radical polymerisations has given the possibility to control the MWD with a higher accuracy than before. Therefore, manipulation of the monomer and initiator stream will affect the value and the shape of the MWD.

Even though Asua (2008) is sceptical to the use of the feed rate to control the MWD, he predicts that better accuracy in models will allow for this in the near future. Several other papers, like Alhamad et al. (2005a) and Farkas et al. (2004) claim that the feed flow rate *can* be used to control the MWD. However, Farkas et al. (2004) acknowledges the difficulties in tailoring the MWD through control of initiator and monomer flows, and addresses the issues that are specific to the control objectives, reactor type and polymerisation chemistry.

5.4 On-line Monitoring

To achieve closed-loop control of a reactor and ensure safe operation as well as the required product quality, some on-line monitoring of variables is required. On-line monitoring is also important as it gives much information of the process that can be useful for modelling and parametrisation. The sensors that are used for on-line monitoring can be divided into two categories:

1. sensors for monitoring *process variables*, or *reactor operation conditions*.
2. sensors for monitoring the *trajectory of polymer properties* during polymerisation.

The first group of sensors include temperature, pressure, level and flow rate measurements, and are performed at the plant site. The sensors of the second group that monitor the polymer quality are very hard to develop. However, they provide more useful information of the polymer quality than the sensors in the first category for closed-loop control strategies.

The last decades the technology behind the category 2 sensors has evolved significantly. This is mainly due to advancements in computer, electronic and process control technologies in the mid-80's that allowed for automation of sensors that up until that time only had been used for off-line monitoring. An example is the development of on-line gas chromatography analysis of the reactor content. Furthermore, in the 90's, fibre optics provided for the use of different spectroscopic techniques to determine the polymer quality.

However, there are still polymer properties that can only be determined by off-line and time consuming monitoring, taking samples of the reaction mixture and sending it to the lab for analysis. Some of these properties include the MWD, branching, cross-linking density and gel content. Among the properties that cannot be measured on-line, some can be estimated by combining mathematical models and the available measurements. The

next section will give a brief presentation of the available on-line monitoring and state estimation techniques.

5.4.1 Sensors for On-line Monitoring of Polymer Quality

Asua (2008) provides Fig. 5.4.1, describing the different sensors that are available for the on-line monitoring of several polymer properties, with the amount of information they provide as a function of the difficulty of implementation. The ideal placement for a sensor would be in the upper left corner of the graph. The polymerisation rate can be measured

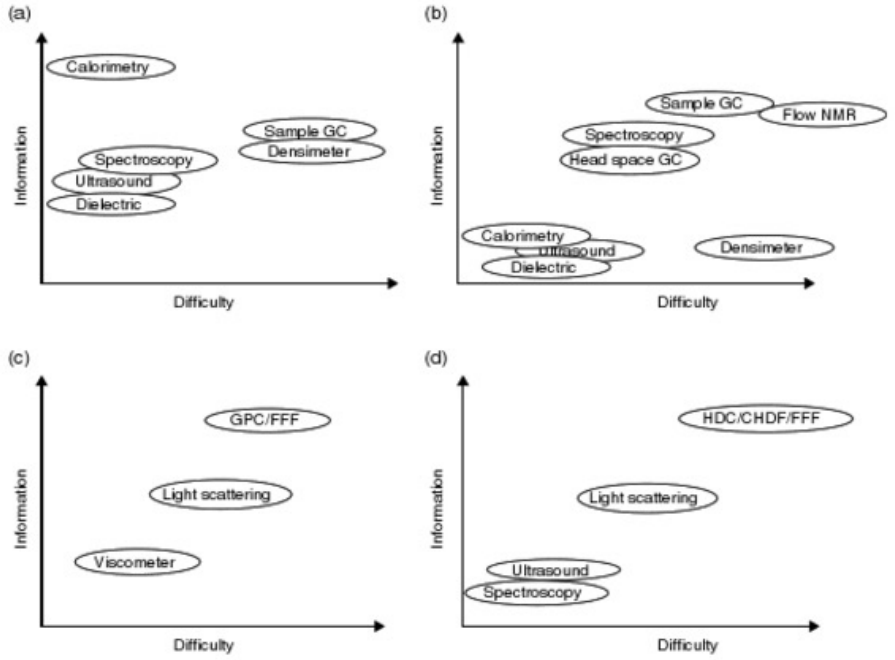

Figure 5.4.1: Charts for sensor selection: (a) polymerisation rate, (b) comonomer concentration, (c) molecular weight distribution, (d) particle size distribution.

using several different techniques. The most convenient for industrial reactors, calorimetry, determines the heat of reaction by solving the energy balances of the reactor and the cooling jacket. This measurement can also determine monomer conversion or monomer concentration. In the case of homopolymerisation this is straight forward, but requires a mathematical model for copolymerisations. There exist many more ways to determine the monomer concentration and conversion that are explained more thoroughly in Asua (2008). For the measurement of the molecular weight distribution, depends on the nature of the polymer product. The classes that are distinguishable are:

- *Soluble polymers:* Polymers that dissolve in a solvent so that the molecular weight

can be monitored on-line by means of gel permeation chromatography and light scattering equipment. This category includes both linear and branched polymers. However, the determination of an accurate MWD for branched polymers is more sophisticated, due to a combination of refractive index, viscosity and light scattering detectors that are used.

- *Insoluble polymers:* These polymers are non-soluble in a solvent and usually have very high molecular weights because the polymer particles are fully cross-linked. In these cases it is not possible to measure the MWD on-line. Currently, the MWD is measured by taking a sample of the reaction mixture and analysing the soluble part of the polymer using GPC/SEC equipment. The insoluble part is given as a gel fraction, and so characterised as an insoluble amount. Additional information regarding the cross-linking density can be obtained using other equipment and methods.
- *Polyolefins:* Polyethylenes and polypropylenes are homopolymers and copolymers that may be dissolved in chlorinated solvents at high temperatures. Their MWD can be measured using the high temperature and pressure fractionation equipment.

The times when on-line measurement of the MWD is not possible, state estimation becomes the method of choice. The easiest estimation is for linear polymers, and using the effect that CTA has on the average molecular weight. There is a direct relation between the unreacted CTA and the number chain length: $DP_n = R_P/R_t = k_p[M]/k_{tr}^{CTA}[CTA]$. Using this ratio it is possible to achieve a good estimate of DP_n. This method has successfully been used for linear polymers and copolymers in emulsion polymerisation (Asua, 2008). When the polymer is insoluble, it complicates the state estimation, and open-loop state estimators can be used. The accuracy of the mathematical model used will determine the estimation capability of the molecular weight.

5.5 Bilevel Control

Process plants have for technological and economic reasons become increasingly complex comprising thousands of measurements and control loops (Larsson and Skogestad, 2000). The large scale systems include several subsystems and can be difficult to control using a centralised control structure. Due to these difficulties, other control structures have been developed and implemented during the last 40 years, like completely decentralised structures, distributed control systems and hierarchical structures. This leads to a functional decomposition of the control problem for the whole plant, assigning different partial control objectives to different parts of the plant (Tatjewski, 2008). The structural decisions in the selection of manipulators and measurements and the decomposition into smaller subproblems is referred to as *plantwide control* (Larsson and Skogestad, 2000). This section will focus on hierarchcal control of multilevel systems.

Morari published a paper in 1982, reviewing plantwide control, where he presented two ways of decomposing the optimisation problem: (1) Vertical (multi-layer), where the difference between the layers are in the frequency adjustments of the input, and (2) horizontal decomposition, where the system is divided into non-interactive parts (Larsson and

Skogestad, 2000). The common vertical, or hierarchical control structure for plant-wide control is explained in Section 4.1.3. This thesis studies the RTO layer and the advanced process control layer. It can also be argued that it is not the RTO layer that is studied, but rather the advanced control layer that is divided into two sub-layers.

In hierarchical multilayer systems, the control action is performed by regulators working at different time scales. According to Scattolini (2009), this is useful in at least two cases: (1) When the overall process behaviour is characterised by at least two different time-scales, and (2) in plantwide optimisation when optimisation and control algorithms work at different rates to compute both optimal targets and the effective control actions to reach these targets. For this study, hierarchical control is mainly included due to the latter reason. Mintz et al. (2016) has a descriptive paper on bilevel MPC control. It describes how distributed MPC schemes are categorised based on the direction of the information flow.

From a historical point of view, multilevel control and optimisation is closely related to Stacklebergs economic problem in the field of game theory (Colson et al., 2007). This theory will be briefly described. Consider an economic planning process that involves interacting agents at two distinct levels: A group of individuals (*leaders*) that issue directives to the remaining agents (*followers*). In the Stackleberg games, the leader is assumed to anticipate the behaviour of the followers, which allows them to choose the optimal strategy accordingly. McGill and Bracken were the first to consider bilevel programs in a series of papers in the 1970's. These papers dealt with applications in the military field, and in production and marketing decisions. McGill and Bracken called these problem *mathematical programs with optimisation problems in the constraints*. The terms *bilevel* and *multilevel programming* were introduced by Candler and Norton in their paper from 1977 (Colson et al., 2007).

In many systems, the reactions and objectives of process plants can be characterised on two different time scales, and it is therefore preferential to perform control at two different time scales (Scattolini, 2009). A regulator that acts on the low frequencies (i.e. long time scale), calculates the control action u_{slow} of the manipulated variables that affect the slow control variables. It also calculates the references for the fast control variables, u_{fast}^{ref}, x_{fast}^{ref} and y_{fast}^{ref}. A second regulator uses these reference values as inputs, and computes u_{fast}, the fast control action, and solves the optimisation problem at a higher rate. A conceptual scheme of this architecture is reported in Fig. 5.5.1.

Many systems, both industrial, economical and sociological, can be described by a hierarchical structure. The highest layer of the hierarchy corresponds to slow dynamics over a long time horizon. The computed control inputs must be provided by a subsystem lower in the hierarchy that operates at a higher frequency. In turn, these subsystems must be controlled at a higher rate and can be placed at an intermediate layer in the hierarchy (Scattolini, 2009). Fig. 5.5.2 is a schematic representation of this three-layer structure. The regulator at the higher level computes its desired control inputs, which are the reference values of the immediately lower layer. An example is a hybrid vehicle, with an internal combustion engine and an electric motor. At a higher level the regulator must compute the torque which is required in order to satisfy the drivers load request and optimise energy management of the system. At the lower level, the engines must provide

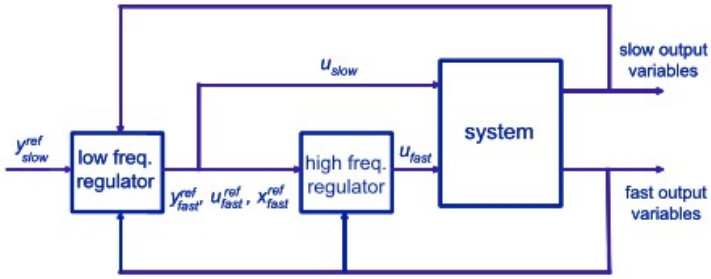

Figure 5.5.1: Schematic representation of the two different time-scale control structure (Scattolini, 2009).

the torque requested in the prescribed time and under operational constraints. To guarantee feasibility of the references computed at a higher level in these kinds of hierarchical structures, some additional information must be sent from the bottom of the hierarchy and up. Moreover, the regulators in the lower layers must guarantee that the solution of the tracking problems have an adequate level of accuracy, so that the mismatch between what is required by the higher level and what is provided by the lower level does not destroy any fundamental properties, such as stability and performance.

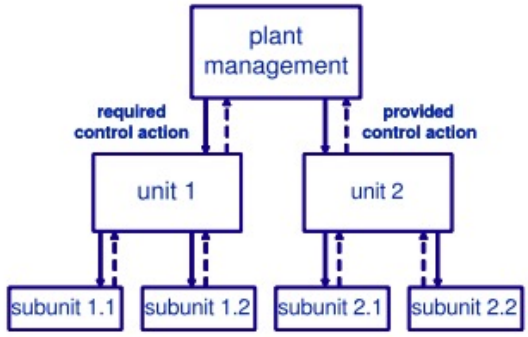

Figure 5.5.2: Schematic representation of a three-layer control structure (Scattolini, 2009).

This multilayer hierarchical control structure corresponds to a classic cascade feedback control system where the outer loop is the slow dynamics and corresponds to the highest layer in the hierarchy, while the inner loop is the fast dynamics. In industrial control systems, the fastest dynamics is usually associated with actuators, while the slow dynamics control the setpoints of the actuators, working towards a specific product quality or economic target. Usually, an MPC is used for the slow control system, while PI & PID regulators control the lower layers. The cascade control is made according to a frequency

decoupling principle, where the dynamics of the feedback loops are so different that all other loops can be assumed steady state. If this is not the case, or the cascade control requires a more sophisticated design, an MPC can be implemented in any layer of the hierarchy (Scattolini, 2009).

In the process industry, it is common to design the control structure as it is discussed in Section 4.1.3, with a higher level RTO and an MPC layer underneath as stated in Larsson and Skogestad (2000). The RTO layer is performed to compute the optimal operating conditions with respect to an economic criterion. The MPC guarantees that the target values that are received from the RTO layer are attained. In this case as well, the lower level can transmit information to the higher level. This approach is very popular in process control, and the design of the RTO layer represents a fundamental role (Scattolini, 2009). The model it uses has to be periodically updated to take into account slow disturbances. Furthermore, there must be a coherence between the model used in the upper layer for the design phase, and the lower layer where the MPC control is implemented. There must also be a steady-state target optimisation to guarantee that the input and output calculated by the RTO are feasible and as close as possible to the desired setpoints (Scattolini, 2009).

Kadam et al. (2003) provides a study of a two-level approach for the integration of model based dynamic real time optimisation (D-RTO) and the control of industrial processes. The objective is to minimise an objective function subject to the dynamic model, equality and inequality constraints and given initial states. An off-line optimisation is not sufficient to solve the problem due to disturbances, inaccuracies in the model, and uncertain parameters. Hence, the problem is re-optimised several times, taking process measurements into account. This implies a closed-loop D-RTO control strategy that implements the process measurements gathered at each sampling interval. However, the relevant dynamics will be too fast to implement closed-loop real-time optimisation on the chosen sampling frequency for such a complex model. The solution is to create a two-level control strategy that includes an upper layer, the D-RTO layer, solving the optimisation problem given by Eqs. 5.5.1, and the lower layer, the MPC layer, solving the optimisation problem given by Eqs. 5.5.2. These operate at two different time scales, with a lower sampling frequency for the D-RTO layer so that it has the time to compute an optimal solution.

$$\begin{aligned}
\min \quad & \bar{\Phi}(\bar{x}, u^{ref}, t_0, t_f) && \text{(5.5.1a)}\\
\text{s.t.} \quad & 0 = \bar{f}(\dot{\bar{x}}, \bar{x}, u^{ref}, \bar{d}, \bar{t}), && \text{(5.5.1b)}\\
& \bar{x}(t_0) = \bar{x}_0, && \text{(5.5.1c)}\\
& y^{ref} = \bar{g}(\bar{x}, u^{ref}, \bar{d}, \bar{t}), && \text{(5.5.1d)}\\
& 0 \geq \bar{h}(\bar{x}, u^{ref}, \bar{d}), && \text{(5.5.1e)}\\
& \bar{t} \in [\bar{t}_o, \bar{t}_f]; \quad \bar{t}_{0_{i+1}} = \bar{t}_0 + \Delta\bar{t}; \quad \bar{t}_{f_{i+1}} = \bar{t}_{f_i} + \Delta\bar{t} && \text{(5.5.1f)}
\end{aligned}$$

$$\min \quad \int_{t_{0_i}}^{t_{f_i}} [(y - y^{ref})^\top Q(y - y^{ref}) + (u - u^{ref})^\top R(u - u^{ref})] d\tau \qquad (5.5.2a)$$
$$+ (\widetilde{x}_N - \bar{x}_N^{ref})^\top P(\widetilde{x}_N - \bar{x}_N^{ref})$$
$$\text{s.t.} \quad 0 = \widetilde{f}(\dot{\widetilde{x}}, \widetilde{x}, u, \widetilde{d}, \widetilde{t}), \qquad (5.5.2b)$$
$$\widetilde{x}(t_0) = \widetilde{x}_0, \qquad (5.5.2c)$$
$$y = \widetilde{g}(\widetilde{x}, u, \widetilde{d}, \widetilde{t}), \qquad (5.5.2d)$$
$$0 \geq \widetilde{h}(\widetilde{x}, u^{ref}, \widetilde{d}), \qquad (5.5.2e)$$
$$\widetilde{t} \in [\widetilde{t}_o, \widetilde{t}_f]; \quad \widetilde{t}_{0_{i+1}} = \widetilde{t}_0 + \Delta\widetilde{t}; \quad \widetilde{t}_{f_{i+1}} = \widetilde{t}_{f_i} + \Delta\widetilde{t} \qquad (5.5.2f)$$

where $\Delta\bar{t}$ and $\Delta\widetilde{t}$ is the sampling frequency for the D-RTO layer and MPC layer respectively. The D-RTO problem in Eq. 5.5.1 determines the optimal trajectories u^{ref} and y^{ref} for all relevant process variables so that the objective function $\bar{\Phi}$ is minimised and the constraints $\bar{h}$ are satisfied. For this work, the only objective in $\bar{\Phi}$ is to minimise the polymerisation time. The process model $\bar{f}$ used for the optimisation should be of satisfactory quality to predict good results for a large range of process dynamics. For the closed-loop D-RTO layer, the optimisation problem is solved over the time horizon $[t_{0_i}, t_{f_i}]$. The sampling time has to be sufficiently large to capture the slow process dynamics, yet sufficiently small to make flexible economic operation possible. It is not necessary to re-optimise the upper layer at each sampling time $\widetilde{t_0}$, the sampling time for the MPC layer.

The MPC problem (Eq. 5.5.2) is set up to follow the trajectories calculated by the D-RTO layer. The sampling time $\widetilde{t_0}$ has to be significantly shorter than the $\bar{t}_0$ to capture the fast process dynamics. Kadam et al. (2003) states that the process model may be the same in the two layers, but may also be slightly different. The requirement, he continues, is that the process model in the MPC problem, $\widetilde{f}$, is simple enough to be solved faster than the sampling time $\Delta\widetilde{t_0}$. The states that are used in the model come from process measurements estimated by for example the Extended Kalman Filter. Also, it is not necessary for the MPC layer to make predictions over the complete time horizon $\bar{t}_{f_i}$, it can be quite a lot shorter and only operate to follow the given trajectories.

To recap, the economic objective, or in this case the polymerisation time, is handled in the D-RTO layer, as well as soft constraints, like the end-product quality. Disturbance rejection is handled in the MPC layer. The alternative to bilevel optimisation is a single-level configuration with MPC control that optimises plant economics at the controller sampling frequency. With a single-level structure, the issues regarding model inconsistencies and conflicting objectives between the two layers are avoided. However, implementing a bilevel approach will significantly decrease the problem size and the computation time.

5.6 Safety and Control Issues in Polymerisation Reactors

The safety record of the chemical industry is generally good, though fires, explosions and incidences involving hazardous chemical reactions do happen. The control of chemical reactions is an important aspect of chemical manufacture. The industry manufactures nearly all products, including polymers, through the control of reactive chemicals. Even though the production of the chemicals normally are carried out without incident, the chemical reaction may get out of control occasionally. This could be due to the use of wrong materials, new operating conditions, or unexpected delays related to equipment failure (Barton and Rogers, 1997). The few accidents in the chemical industry may be caused by technical failures, human failures and the chemical reaction itself, because of limited knowledge of the thermochemistry and chemical reactions in the process (Asua, 2008, Ch. 8).

According to Leiza and Asua (1997), the main objectives to be fulfilled in the production of dispersed polymers are safety, production rate and product quality. In that order. In polymerisation reactions the reactor temperature must be kept under safe limits to avoid thermal runaway. Barton and Rogers (1997) states that runaway can be a particular problem for unsteady-state batch reactors, where the rate of reaction, an thereby the heat of reaction production, varies with time. Safety also comprises environmental aspects, so the violation on environmental regulations on the plant environment but also on the final product must be avoided.

Polymerisation reactions are exothermic, and if the cooling system fails, an uncontrolled thermal runaway reaction may occur, leading to increased temperatures. Elevated temperatures may lead to secondary reactions that further generate heat or non-condensable gases, that may lead to increased pressure in sealed or inadequately vented reactors. Though runaway polymerisation accidents fortunately are not common, they make up a large part of the industrial accidents (Manders et al., 2011). To prevent accidents, the equipment and facilities have to be kept to specific standards, the operators have to be qualified, and the models used for control need adequate knowledge and understanding of the underlying chemical reactions (Asua, 2008, Ch. 8). Moreover, automation and on-line control of the process will also help prevent technical and human failure. Advanced control strategies for polymerisation reactions should aim to increase production, achieving the required polymer product quality, and also avoid unsafe situations.

Manders et al. (2011) states that a common method to safeguard a reactor is to use pressure relief valves on the top of the reactor. However, this is not adaptable for all polymerisation processes. Emulsion polymerisation reactors are required to fulfil a wide range of application property requirements, and it may be difficult to determine worst-case scenarios. If a deviation from the recipe occurs, this may lead to situations far worse than expected (Manders et al., 2011).

CHAPTER

SIX

PROCESS DESCRIPTION

6.1 Description of the Process

The process is a free-radical emulsion copolymerisation in a semi-batch reactor in a monomer-starved regime. The polymer is made up of four different monomers: A, B, C and D[1], where A and B are the main components and are not soluble in water. Therefore an emulsifier is added. The initiator is aqueous, and extra water is also added to the reactor. Fig. 6.1.1 is an illustration of the process. The whole process starts with an empty reactor and several tanks that will be emptied into the reactor during the batch: One containing the initial charge loaded into the reactor (some water, initiator, seed and emulsifier), one containing the monomers (Tank 1) and one containing the initiator (Tank 2). In addition there are some other tanks containing water that are added towards the end. The reaction takes place in the reactor at around $85\,^{\circ}\mathrm{C}$, so the reactor is slowly heated while the initial charge is fed into it. When the right temperature is reached (and all the initial charge is in the reactor), the heating stops. At this point, the initiator tank and monomer tank is fed into the reactor at a fixed ratio. The exothermic reaction that occurs in the reactor prevents the temperature to drop (and some cooling might be necessary to keep the temperature from rising too high). When both these tanks are empty, some extra water is added to the reactor to finalise the product and the tank is cooled down.

[1]The monomers can not be given up as a part of the confidentiality agreement with Cybernetica and their industrial partners.

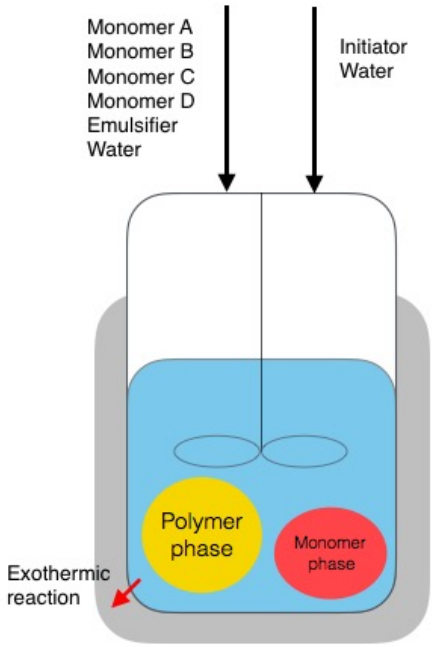

Figure 6.1.1: Illustration of the process. There are three phases in the reactor and a stirring element. The reaction is exothermic, cooled by a surrounding jacket, and monomer A, B, C, D as well as water, initiator and emulsifier is added to the reactor throughout the batch.

Several assumptions have been made in order to simplify the model:

- It is a seeded emulsion copolymerisation in an ideally mixed semi-batch reactor.
- The emulsion is formally divided into three phases: Continuous aqueous phase, monomer droplets and polymer particles. The volumes of these three phases are considered to be additive.
- The following species are in the system: Monomers A, B, C and D, water, initiator, emulsifier, dead and live polymer chains. Emulsifier is treated as water in the reaction balances. Chain transfer to monomer as well as gel effect is considered, though chain transfer to polymer is excluded from the model.
- There are two types of radicals in the system: Radicals in the aqueous phase as a result of initiation or desorption from the polymer phase, and radicals distributed among polymer particles. The latter are a result of radical absorption into the particles from the aqueous phase.
- The amount of monomer C and D dissolved in the polymer particles and monomer

droplets is neglected, as well as the amount of monomer A and B in the aqueous phase. For monomer A and B, linear phase equilibrium is assumed due to constant partition coefficients.

- Polymerisation kinetics is processed by the population balance of polymer moments, which is implemented in the format generated by the software PREDICI software[2].
- The model only describes Interval II and III in the emulsion polymerisation (see Section 3.3). Since this is a seeded polymerisation, the total number of polymer particles in the reactor is considered to be constant during the simulation.
- The model includes a heat balance of the reaction mixture and of the reactor jacket.

The model equations consists of both differentiable and algebraic equations. The theory behind some of them are described in Section 3.4. The model consists of the following equations:

- Total balance of all four monomers.
- Balance of monomer A and B in the polymer phase.
- Balance of all monomers incorporated into polymer chains.
- Balance of radicals both in the polymer and the aqueous phase.
- Balance of polymer moments for living and dead chains.
- Heat balance of the reaction mixture and of the jacket.

The model written in C is extensive, comprising 38 states (X) and 55 derived output variables (Z). There are 98 constants and 47 parameters, many of them correction factors used for tuning the EKF.

6.2 Control System

Figure 6.2.1 illustrates a typical setup of a semi-batch emulsion polymerisation control system. Flowmeters control the flow of initiator and monomer and their ratio, and a temperature sensor registers the temperature of the flow. An assumption in the model is that the reactor is well mixed due to the work provided by a stirrer. The temperature of the reactor may be monitored on-line with a temperature indicator in the reactor. Furthermore, the temperature of the reactor is controlled by the temperature of the water in the cooling jacket that surrounds the reactor. Both the flow and the temperature of the water circulating in the jacket can be controlled. Usually there are two streams into the jacket, one hot and one cold, that determines the jacket inlet temperature, controlled with PI or PID controllers. The water in the jacket can also be recycled. The difference in the water inlet and the water outlet temperature of the jacket can be determined and says something about the heat transfer between the jacket and the reactor. The interfacial area between the jacket

[2]PREDICI is a simulation package for the modelling and dynamic simulation of macromolecular processes. (See: http://www.cit-wulkow.de/products/predici-polymerisation)

and the reactor will partly determine the heat transfer, and is determined by the design of the jacket. In this study, the jacket is designed to maximise the interfacial area, formed as a coil around the outside of the reactor.

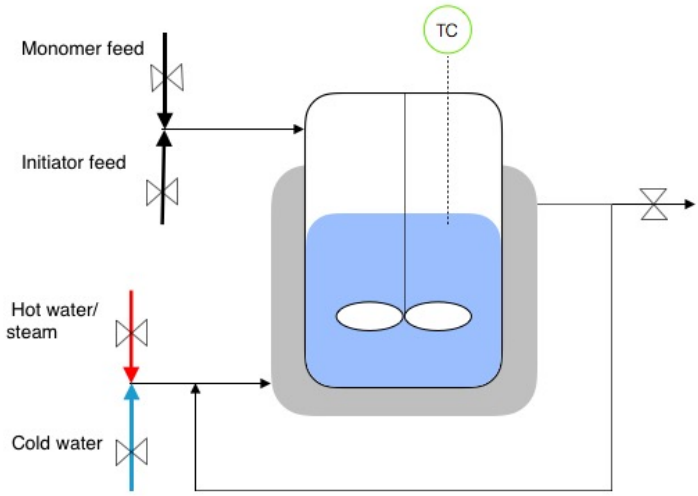

Figure 6.2.1: Illustration of a typical control scheme of an emulsion polymerisation semi-batch reactor.

CHAPTER

SEVEN

INTRODUCTION TO CYBERNETICA'S SOFTWARE AND MODEL SET UP

Cybernetica AS is a company that develops tailor made model based control systems and delivers NMPC applications based on mechanistic models. They have developed their own software, designed especially for this purpose, and these have been used for the work described in this thesis. ModelFit is used for the parameter estimation and model verification, RealSim works as a theoretical plant replacement, and CENIT is the nonlinear optimisation tool. The software is developed to work well together, and all of tools use the same model implemented in C. This model defines all states, inputs, outputs, controlled variables, parameters and constants, as well as other variables that are mainly used internally in the software. Fig. 7.1.1 indicates how the different software send information.

7.1 Software

All the software communicate with the model implemented in C. To solve the differential algebraic equations (DAEs), different solvers are integrated in the Cybernetica model template. The most simple one, and the one used in this project is the Euler solver. The most sophisticated solver is CVODE[1]. However, any solver can be implemented to solve the DAEs.

[1] CVODE is a solver developed by SUNDIALS, that solves ordinary differential equations and DAEs that are stiff or nonstiff.

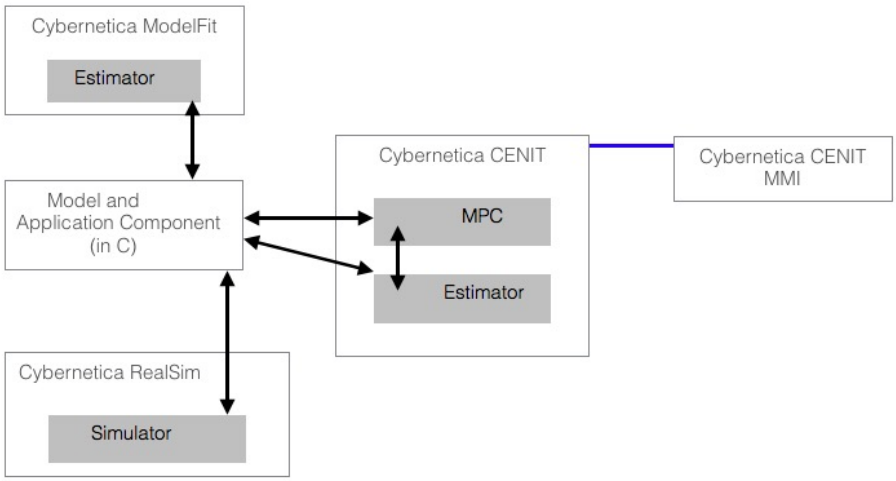

Figure 7.1.1: Illustration of how the software used in this study shares information. The three software are ModelFit, RealSim and CENIT, all using the same model written in C. CENIT MMI is the CENIT GUI. CENIT is the on-line optimisation software, RealSim is a theoretical plant replacement, and ModelFit is used for off-line parameter estimation.

7.1.1 ModelFit

Cybernetica ModelFit is developed by Cybernetica AS as an off-line estimation tool of model states and parameters for model validation. It ensures that the process models are well adjusted to the real plant. It is a powerful tool for simulating ballistic models and to support parameter estimation, which is a critical phase in optimisation and model based control projects. ModelFit performs off-line calculations and uses data logged from the process.

ModelFit can be used to simulate the model off-line and is easily used for plotting the states, inputs and measurements. An important tool in the ModelFit software is the procedure to optimise the time-invariant parameters. It allows the user to choose the parameters to optimise and then weight the measurements that the parameters should be optimised to. ModelFit uses the least squares method in order to make the deviation between the simulated value and the measured value to be as small as possible. The mathematical expression of this is given in Eq. 7.1.1.

$$\min_{\eta\in\theta} \sum_{k=1}^{n_{k_y}} (y_{p,k}(x^k, u^k, \theta) - y_{m,k})^2 \tag{7.1.1a}$$

$$s.t. x_{k+1} = f(x^k, u^k, \theta) \tag{7.1.1b}$$

$$y_{p,k} = g(x^k, u^k, \theta) \tag{7.1.1c}$$

In this equation, $y_{p,k}$ and $y_{m,k}$ represent the predicted and measured values respectively for the sample point k. n_{k_y} is the number of valid measurements, and η is the selection of parameters to be optimised (among all parameters θ). The model is discrete, so k represents a point in time. The optimisation problem is subject to a lot of discrete constraints, implying that the model equations must be satisfied for any η. Since the model equations are nonlinear, this is a nonlinear optimisation problem and has to be solved by an adequate solver. ModelFit uses an SQP solver[2]. ModelFit can also use an off-line Kalman Filter that will adjust the parameters throughout the simulation to make the model fit better to the data points. However, this was not used for this project, and will not be further discussed.

7.1.2 RealSim

Cybernetica RealSim, a plant replacement process simulator, and serves as a simulation tool and test-bench for on-line control applications. It is used to test the controller off-line and tune it prior to installation. It allows for open loop testing and debugging of the controller. RealSim also provides an interface to the user as an operator, where it is possible to change different parameters, as if a real operator was working on the plant.

RealSim is used together with CENIT, and is a real-time simulator for testing various control applications on the plant. It is runs a separate instance of the same model as the control application. RealSim and CENIT use an OPC server[3]to synchronise, so that the simulation can be more rapid than the real-time process. RealSim provides the measurements for CENIT and the MPC. This data is is read by CENIT that provides control input action that is applied for the next sample run by RealSim. The main reason for using an OPC server is to ensure that the simulated conditions are as identical as possible to the on-line conditions. Then it is not necessary to change anything in the model depending on if the simulation is off-line or on-line.

RealSim simulates the plant behaviour, and works as an interface to the operator. The "plant" can run, pause or just calculate one sample. When using RealSim, the current data values are visible in the main window. It is possible for the user to modify parameters and inputs when the simulator is paused. A screenshot of the RealSim GUI is represented in Fig. 7.1.2.

7.1.3 CENIT

Cybernetica CENIT is an NMPC controller software based on nonlinear mechanistic models, and it allows for control of continuous or batch process units in the economically optimal way. The models that are used in CENIT are specifically developed for the process. However, there is always some uncertainty in a mathematical process model, that CENIT can efficiently compensate for with its estimators. CENIT has two integrated estimators:

[2]SQP: Sequential Quadratic Programming. This is an iterative method for nonlinear optimisation. See Section 4.2.1.

[3]OPC server: Open Platform Communications Server. Allows communication of real-time plant data from different manufacturers.

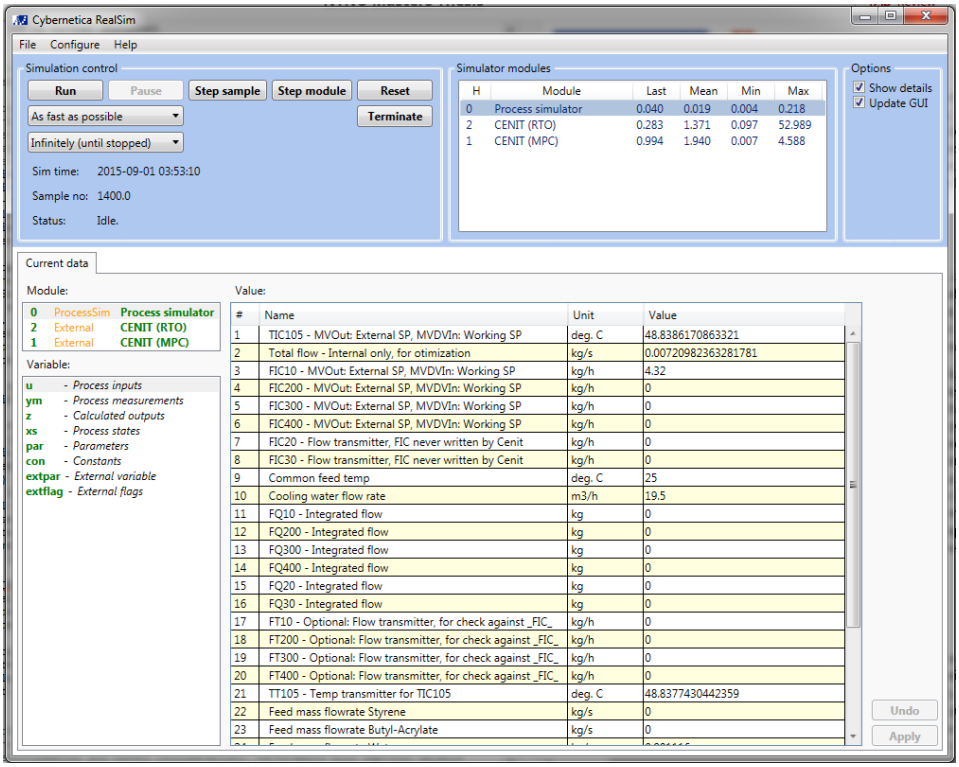

Figure 7.1.2: Screenshot of the RealSim GUI. The variables from RealSim as well as the two layers (MPC and RTO) are visible, and also the possibility to run and pause the simulations.

the Kalman filter and the Moving Horizon Estimator (MHE). It communicates with the plant via OPC.

CENIT MMI is the CENIT interface. It allows the user to easily adjust settings in CENIT, for example by turning on and off constraints, MVs and changing setpoints and weights. Furthermore, the CENIT MMI allows for plotting the different variables, both the current states and the predictions. This is a useful tool in order to see the result of the optimisation.

CENIT uses an SQP algorithm, as described in Sec. 4.2.1, to solve the NLP. The algorithm is built in the way that CENIT searches for a better solution by providing small perturbations in the control inputs. This creates a sensitivity matrix that determines how the controlled variables respond to these perturbations. CENIT will choose to apply the perturbations that provide the best result.

7.2 Communication Set Up

The aim for this thesis is to set up a two-layer control structure where the upper layer predicts a profile that the lower layer works to follow. This requires two different instances of CENIT that operate simultaneously, and communicate with each other as well as with RealSim. The idea is that the upper level has a prediction horizon that is long enough to see the end of the batch. It calculates the optimal feed flow rates and temperature profiles to obtain the correct M_n value for a short polymerisation time. The lower layer does not observe the end of the batch until late in the simulation when the end of the batch is very near. The upper layer has a longer prediction horizon, and therefore also more variables to calculate. To avoid a too high computational cost, this layer the sample time is larger than for the MPC layer. The MPC layer has a shorter sampling time, and works to follow the reference trajectories provided by the RTO layer and reject disturbances. So the upper layer works at a lower frequency than the lower layer. Even though both layers optimises using MPC control, the upper layer will be referred to as the RTO layer and the upper layer as the MPC layer.

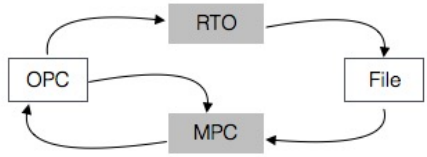

Figure 7.2.1: Illustration of how the RTO layer and MPC layer communicate.

The two layers are made as two different instances of CENIT that are operated simultaneously and connected to RealSim via OPC. RealSim works as a simulation of the real plant. First of all, a RealSim sample is run to provide measurements. Then the RTO layer is run to provide the optimal reactor temperature profile, and then the MPC layer starts working. It is important to know exactly how the layers communicate. To simplify matters, data is only sent in one direction, from the RTO to the MPC layer. As RealSim functions as the plant, it provides all the states and measurements for both the RTO layer and MPC layer to read and implement. However, only the MPC layer has the possibility to write to RealSim. All communication between the RTO layer and MPC layer goes through a file. One of the decisions that have to be made is then what information that has to be sent from the RTO layer to the MPC layer. This set-up is illustrated in Fig. 7.2.1.

7.2.1 Controlling each layer

The implementation itself of two layers is not the main issue, the difficulty lies in choosing how to control them and how they work together. Each layer has its purpose, and contributes to the overall goal of the process, which to obtain the correct value of M_n at

the shortest amount of time. As the value of M_n is difficult to measure on-line, the control of the reactor depends on other measurements. The reactor temperature and the flow rate are two variables that are possible to measure continuously and during on-line operation. Furthermore, both the reactor temperature and the monomer and initiator concentration in the reactor (indirectly the feed flow rate) influence the end-product quality M_n.

The role of the RTO is to determine the reactor temperature trajectory and the monomer/initiator feed flow rate trajectory for the entire prediction horizon, i.e. from the current position in time with the corresponding updated states and the whole prediction horizon. The MVs in the RTO layer is the derivative of the reactor temperature, and the total feed flow rate. The setpoint is the polymerisation time, which is set to zero so that the optimisation problem will always work towards a minimal time. There are constraints on the value of M_n, the reactor temperature and a max feed flow rate. The RTO layer writes the entire reactor temperature profile and feed flow rate profile to a file, that will be read by the MPC layer. Since the RTO layer predicts over the whole batch time, i.e four hours, there are several more variables to calculate and the computation time is relatively high. However, it is not necessary to predict the profiles every sample, i.e. every 10^{th} second. The main requirement is that the computation time is lower than the sample time. The sample time for the RTO layer is chosen to be two minutes.

The MPC layer receives the temperature and feed rate profiles, and strives to follow these by using the jacket temperature T_J and the total feed flow rates as MVs. The MPC layer contains the complete temperature equations, and is therefore more accurate. However, M_n is not taken into account in this layer, it is supposed that following the given trajectories, the value of M_n will be satisfactory. The MPC layer has a prediction and control horizon of 30 minutes, significantly shorter than the RTO layer. However, the sampling time is only 10 seconds. The aim is that the short sampling time will take care of disturbances and model errors to keep the process on the given trajectories. With a sampling time of 2 minutes for the RTO layer and 10 seconds for the MPC layer, this means that the RTO layer is run once every 12^{th} MPC sample. The different setpoints and constraints are recapitulated in Table 7.2.1 and 7.2.2.

MV:	
Derivative of reactor temperature	u_dTR
Total feed flow rate	uMass_tot
Setpoints:	
Polymerisation time	0 min
Constraints:	
Reactor temperature	80-93 °C
M_n	
Sufficient cooling capacity	
Max feed stream	

Table 7.2.1: MVs, setpoints and constraints for the RTO layer.

MV:	
Jacket temperature	T_J
Total feed flow rate	uMass_tot
Setpoints:	
Temperature profile	
Feed flow rate profile	
Constraints:	
Reactor temperature	80-93 °C
Jacket temperature	15-120 °C
Max feed stream	

Table 7.2.2: MVs, setpoints and constraints for the MPC layer.

CHAPTER

EIGHT

SIMULATIONS AND ANALYSIS

In the industry today, optimisation and control of polymerisation reactors using nonlinear MPC is not that common because of the time and effort required to create an accurate process model. Currently, the process described in Sec. 6.1 follows a predetermined recipe, stating the feed stream flow rate and a constant reactor temperature, that has proven to give the correct end-of-batch quality of M_n in approximately 195 min. This recipe has been developed through trial and error, and the experience of the operators. However, there is no guarantee that this recipe is the optimal one, and the possibilities to adjust the process should there be any disturbances are slim.

The aim of this thesis is to establish a bilevel MPC control structure on the process described in Section 6.1, with the superior motive to reduce the polymerisation time. This chapter will provide the results of the different simulations and a discussion around them. The first part of the chapter will briefly present the results from the preliminary project. Then follows the results of the work for this thesis, mainly discussing the effect of weighting and tuning of the layers in order to achieve satisfactory process behaviour. Because of the confidentiality agreement with Cybernetica and their industrial collaborations, some of the exact variables, model equations, weights and parametrisation cannot be given specifically.

8.1 Preliminary Results

In advance of this master's thesis, a smaller preliminary project was realised (see Kjetså (2016)). The aim of the project was to reduce the polymerisation time of the batch reactor by off-line NMPC optimisation, with only one NMPC application. By solving the NMPC problem off-line, optimal reference trajectories were created to be followed by the process

to achieve a shorter polymerisation time. The project explored different cases, using one, two and three degrees of freedom to control the process.

First of all, the project explored the effect different constant temperatures and constant feed flow rates had on the final value of M_n. The results showed that to achieve a higher value of M_n, the reactor temperature had to be lower, and the feed flow higher, a result that is also supported by Alhamad et al. (2005a) and Alhamad et al. (2005b). In accordance to this, a low value of M_n at the end of the batch could be achieved with a low feed flow rate and a higher temperature. This result can be seen in coherence with the reaction rates, initiation vs. termination and propagation vs. termination. When M_n is lower for a higher temperature it means that the termination rate is higher compared to propagation, creating shorter chains. When the feed stream is higher, then so is the concentration of initiator and monomer, and propagation is more likely to happen than termination, resulting in longer, fewer chains and a higher M_n value.

For 1 DOF, the reactor temperature was kept constant while the feed flow rate profile was optimised. The results of the simulations showed that the polymerisation time was slightly shortened compared to the original polymerisation time at $85\,^{\circ}\mathrm{C}$, and even shorter at a higher temperature, at $90\,^{\circ}\mathrm{C}$. This can be explained by the increase in polymerisation rate at higher temperatures, so that the conversion criteria is reached earlier.

In the original recipe, the feed flow contains monomer and initiator at a constant ratio and the total flow is one manipulated variable. Allowing the system two degrees of freedom, both the temperature profile and the feed flow rate profile were calculated. With no constraints on the cooling capacity the feed stream was kept at the maximum value and the polymerisation time was minimised. With cooling constraints the polymerisation time increased significantly, due to a slower feed stream. During the simulations for two degrees if freedom it became clear that the dependencies in the reactor temperature and cooling demand and heat created from reaction of the monomer present was complex.

An experiment was performed where the feed stream was divided into two manipulated variables, one for the monomer and emulsifier stream and the other for initiator stream. This was done partly because the system, and the value of M_n showed a very high sensitivity to the concentration of initiator. Allowing for three degrees of freedom, the polymerisation time was further reduced, and with cooling constraints, the polymerisation time remained low.

The results in the preliminary project were promising showing that the polymerisation time could be reduced by applying an off-line NMPC control scheme on the process. This master's thesis takes the next step and controls the process using on-line NMPC instead of only calculating the trajectories in the first sample. The trajectories that are calculated in the first sample might be ideal at that point in time. However, disturbances, uncertain measurement, process noise and errors in the mathematical model will result in estimated model states that do not correspond to plant measurements, and the trajectories are no longer accurate. To get back on track, feedback is implemented in the control algorithm, allowing for recalculation of the trajectories. Another difference in this thesis from the project is the inclusion of the cooling capacity from the surrounding jacket. For the project it was mainly assumed that the cooling capacity of the cooling jacket was ideal. When a

constraint was put on the cooling capacity, these were hard constraints at certain values valid for the whole prediction horizon. However, the cooling capacity depends on the reactor temperature (see Section 8.3) and is not constant throughout the prediction horizon. This improvement was introduced in the model used in the thesis.

In the simulations for this thesis, the feed stream is considered as one MV, feeding monomer, emulsifier and initiator into the reactor at a constant ratio, since this is what is used in the real process. The underlying MPC layer strives to follow the optimal temperature and feed flow rate trajectories that are calculated by the upper layer. The optimal trajectory is recalculated at a lower sampling frequency in the RTO layer (every 2 minutes), while the lower MPC layer will run every 10 seconds. The RTO layer will have a prediction horizon of four hours that sees the end of the batch, while the MPC layer has a shorter time horizon of 30 minutes in order to decrease computation time. The two layers contain different CVs and MVs, as explained in Section 7.2.1.

8.2 Defining the Polymerisation Time

The overall objective of this study is to minimise the polymerisation time of the process. This is taken care of in the RTO layer, with an optimisation problem formulated in words as follows:

$$\begin{aligned} \min \quad & J = PolymerisationTime \\ \text{subject to} \quad & \\ & \text{Model constraints (model functions)} \\ & \text{CV constraints} \\ & \text{MV constraints} \end{aligned}$$

The setpoint of the objective function is set to 0 minutes so as to minimise the polymerisation time. The challenge was how to define the CV *PolymerisationTime*.

The polymerisation time is defined from the start of the batch to the point in the process where the tank that initially contained the monomer mixture is completely emptied into the reactor, and the total conversion of monomer A has reached 99 %. The start of the batch is considered to be when an initial charge containing some initiator, emulsifier and water has been added to the reactor, and the contents has been heated up to the starting temperature of $85\,^{\circ}\mathrm{C}$. At the end of the batch, when the conversion of monomer A has reached 99 % and all the monomer has been introduced to the reactor, all the differential equations in the model are set to zero, which makes it easy to identify the end of the batch.

Initially, *PolymerisationTime* was equal to the current batch time, a function that increased linearly with time until the end of the batch where the time remained constant. However, the optimal trajectory of the feed stream was close to no stream near the end of the batch. The tank did not empty rapidly into the reactor and the conditions for the condition for the end of the batch was not reached. To avoid this problem the polymerisation time was

redefined to:

$$PolymerisationTime = \frac{\text{mass left in tank}}{\text{feed flow rate}} \tag{8.2.2}$$

In this new definition, *PolymerisationTime* is not a very intuitive variable. The unit is still time (seconds), but it starts as a large value and will decrease throughout the batch. The idea was that as the feed flow rate also being an MV, this would create a win-win situation: The mass in the tank would steadily decrease, bringing the polymerisation time closer to its setpoint. Keeping the feed flow rate high would also contribute to minimising the time. However, the same problem as before was observed, when the tank was close to empty, the feed flow rate decreased drastically and it took a long time for the batch to end. An explanation for this is that as the polymerisation time decreased, the smaller the deviation from the setpoint became. The sensitivity in the polymerisation time will decrease, so the optimal solution to the problem is keeping some monomer in the tank.

The third way to define the polymerisation time is to merge the two previous ones:

$$PolymerisationTime = \text{current time} + \frac{\text{mass left in tank}}{\text{feed flow rate}} \tag{8.2.3}$$

The *PolymerisationTime* variable as defined in Eq. 8.2.3 is also quite abstract. Instead of decreasing or increasing steadily throughout the horizon, the aim is to keep the value constant. If the feed flow stream remains constant, then so should the polymerisation time. Also, it will be large enough to provide sensitivity in the cost function.

8.3 Temperature Control System

The reactor temperature, T_R, is a complex variable and depends on many different factors. The polymerisation reaction is exothermic, meaning that the reaction creates heat that has to be transported away to keep the reactor close to constant temperature. Furthermore, the reaction is temperature dependent since the reaction rate coefficients follow Arrhenius' law. This implies that a higher temperature in the reactor creates a higher reaction rate, creating more heat of reaction, which in turn heats the reactor even more. If no cooling is provided to the reactor, T_R will continue to increase, resulting in thermal runaway of the reactor. The reaction rate is also higher the more monomer that is present, so a high feed flow rate of monomer and initiator provides a higher reaction rate, creating more heat in the reactor. However, the feed flow is cold, providing a cooling effect to the reactor. Some heat is lost to the environment. Finally the jacket surrounding the reactor provides temperature control of the reactor. The complete reactor temperature equation is included in the MPC layer and is defined in Eq. 3.4.27.

In the RTO layer the reactor temperature is directly determined from u_dT_R, the input variable, and this layer does not include the complete temperature equation. However, there needs to be a link between the cooling capacity and the temperature profile that the RTO layer calculates. Without this link the RTO layer will predict temperature profiles that are impossible for the MPC layer to follow. Without the constraint on the cooling capacity, the temperature profile in the reactor will be calculated as if a perfect cooling

jacket surrounded the reactor that could transport away all the extra heat. This results in a mismatch between the RTO layer and the MPC layer, since the RTO layer has no indication that there is a limit to the cooling capacity.

A constraint on the cooling capacity has to be defined such that the RTO layer is aware of its limitations. For the constraint to be dynamic in the prediction horizon, a new variable is defined that is the difference between the *CoolingCapacity* and the *CoolingDemand*. *CoolingDemand* is essentially the inverse of the reactor temperature equation (Eq. 3.4.27), calculating how much cooling is required by the jacket in order to maintain the desired reactor temperature. Eq. 8.3.1 shows the calculation of the *CoolingDemand* variable:

$$CoolingDemand = \frac{dT_R}{dt}\dot{m}c_p - (-\Delta H_R + \Delta H_{feed} + \Delta H_{loss}) \tag{8.3.1}$$

The *CoolingCapacity* is a variable that is modelled on the jacket surrounding the reactor, and gives the effect of the cooling of the jacket, and is equal to the cooling available from the jacket, ΔH_J, as given in Eq. 8.3.2.

$$\Delta H_J = -UA(T_R - T_J) \tag{8.3.2}$$

where *UA* is the heat transfer coefficient, and the higher this number is, the better the heat transfer between the reactor and the jacket. Eq. 8.3.2 shows that when $\Delta H_J > 0$, the jacket is heating the reactor, and when $\Delta H_J < 0$ the reactor is being cooled. In order to control the reactor temperature, there must always be more cooling capacity than the cooling required.

$$CoolingDemand < CoolingCapacity \tag{8.3.3}$$

It is common to use the maximum cooling capacity, *MaxCoolingCapacity*, in order to loosen the constraint. Eq. 8.3.4 gives the maximum cooling capacity, with the current value of the reactor temperature and the minimum value of the jacket temperature (Leiza and Meuldijk, 2013).

$$MaxCoolingCapacity = \Delta H_J^{max} = -UA(T_R - T_J^{min}) \tag{8.3.4}$$

where T_J^{min} is the minimum jacket temperature available. The cooling capacity constraint is not as tight, and is given in Eq. 8.3.5.

$$CoolingDemand < MaxCoolingCapacity \tag{8.3.5}$$

8.4 Establishing Communication Between the Layers

The first part of this study was the implementation of the two layer operation in CENIT, as described in Section 7.2. The idea is that the RTO layer determines an optimal profile that is written to a file. This file is read by the MPC layer, and is set as the reference trajectory the MPC has to follow. To ensure that the communication was working, only the temperature profile was optimised, and the feed rate was constant.

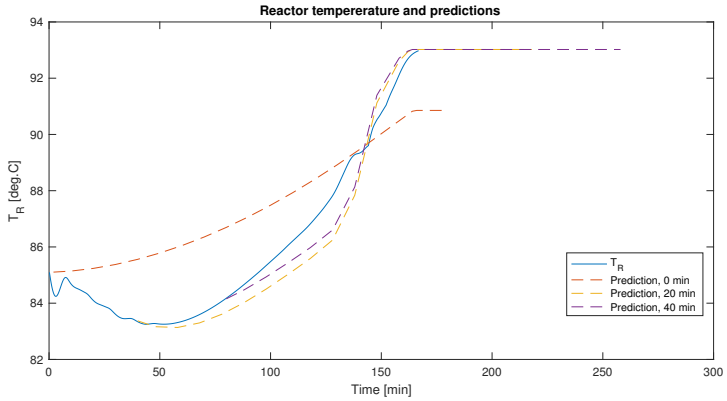

Figure 8.4.1: The first simulation where the two layers work correctly together. The solid blue line is the simulated trajectory, and the dashed lines are the reference trajectories at different points in time.

Fig. 8.4.1 shows the first simulation where the data is correctly transferred from one layer to the other. The solid blue line is the actual reactor temperature profile that is simulated as a result of the input moves from the MPC layer. The dashed lines are the reference trajectories at different times. However, the MPC layer does not show optimal behaviour as the reactor temperature does not follow the initial trajectory. This could be caused by inappropriate initial conditions, different constraints and the difference in the way the reactor temperature is modelled in the two layers. One can see that the actual reactor temperature becomes closer to the optimal trajectories throughout the batch. This may be due to initial conditions that have to stabilise the first minutes. The initial conditions include the initial reactor temperature and the initial jacket temperature. For this study they are both chosen to be $85\,^{\circ}\mathrm{C}$. In the real process, however, the initial temperatures are determined during the initial heating of the reactor.

After this initial experiment, the configuration of the layers was as described in Tables 7.2.1 and 7.2.2. The MPC CENIT Kernel also receives the feed stream profile in addition to the reactor temperature profile from the is also read from the file. Fig. 8.4.2 gives the RTO profile and the MPC prediction at three different points in time, and the simulated feed stream. There is a coherence between the RTO profile and the predicted trajectory of the MPC. However, the trajectory is constantly recalculated in the MPC layer (every

sample, i.e. every 10 seconds), resulting in a different final trajectory for the feed stream. The following sections will discuss the action taken to avoid the difference between the predicted trajectories and the final simulated trajectory.

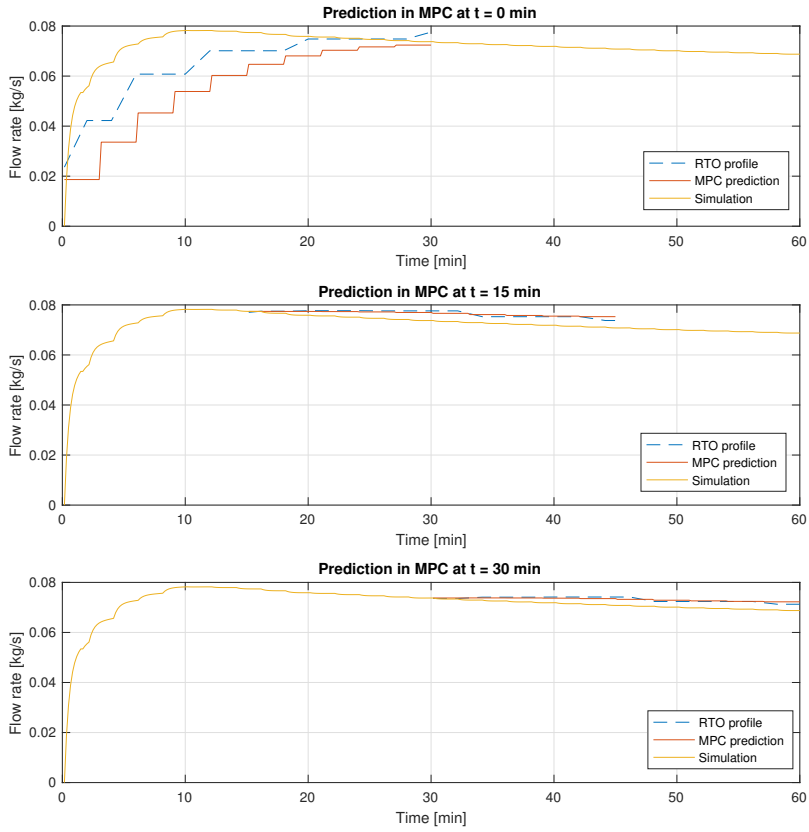

Figure 8.4.2: The MPC layers predictions of the feed stream at three different points in time, and also the simulated trajectory of the feed stream.

8.5 Modification of the Heat Transfer Coefficient

The main difference between the RTO and the MPC layer is how the reactor temperature and the jacket temperature is modelled. The RTO layer does not include a detailed temperature equation, but uses the derivative of the reactor temperature u_dTR as a decision variable, an MV. The MPC layer uses the complete reactor temperature model, where the heat of reaction, cooling from the feed, heat loss to the environment, and the cooling effect

from the jacket is taken into account. As discussed Section 5.3.1, controlling the reactor temperature using the cooling effect of the jacket is not an issue until dealing with industrial sized reactors. The model used for this thesis is scaled for a pilot reactor, and the heat transfer coefficient *UA* for the transfer of heat through the wall of the reactor is sufficiently large. The heat transfer coefficient is dependent on both the material of the wall of the reactor, as well as the surface area between the wall and the surrounding jacket. In addition, when the reactor is being fed, more and more polymer will be made, the viscosity will increase and the heat transfer will be less efficient.

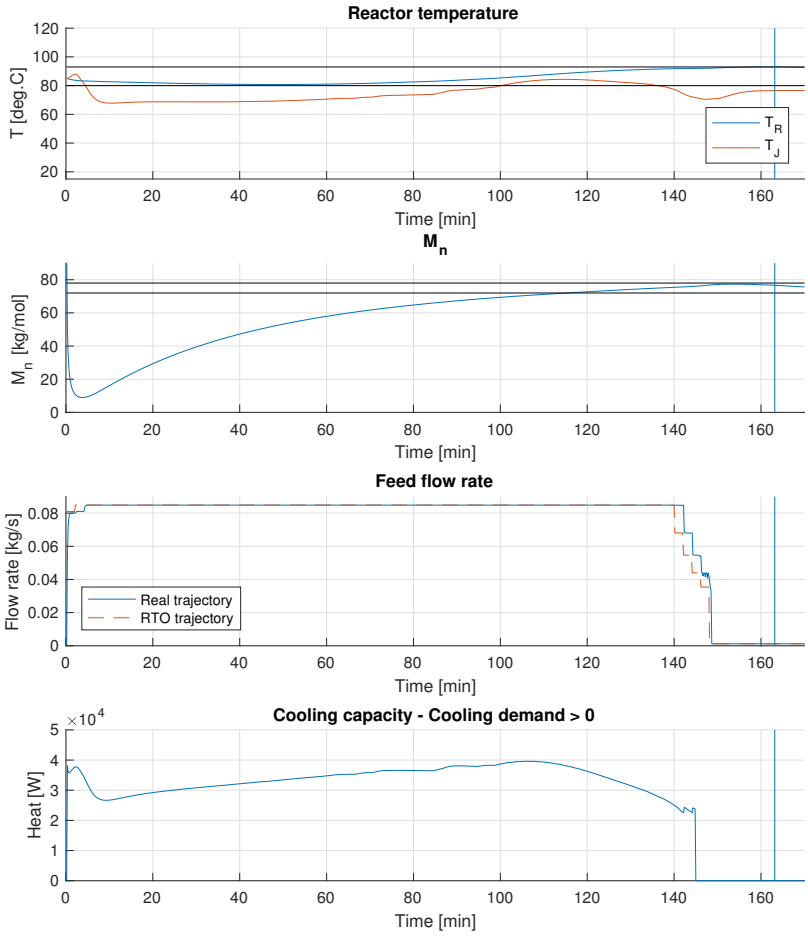

Figure 8.5.1: Simulation of the results under the original conditions of the pilot plant. The polymerisation time is 163.17 minutes.

Simulating the process with sufficient heat transfer is not of much interest, the simulation is shown in Fig. 8.5.1. The optimal result is to maximise the feed flow rate, which is possible since the heat of reaction that is created is compensated for by a sufficient heat transfer rate. The reactor temperature stays within its constraints, the cooling capacity constraint is not violated and the M_n has a satisfactory value at the end of the batch. Furthermore, the polymerisation time under these original conditions of the pilot plant is 163.15 minutes, which is shorter than the polymerisation time of the preliminary project under the same conditions which was 167.25 minutes. The two-layer approach has the potential to reduce the polymerisation time. However, the more interesting approach is to study how the optimisation problem reacts when the cooling from the jacket is not sufficient, meaning that the heat transfer coefficient is reduced, $UA/10$. This requires the optimisation problem to optimise the feed flow stream to reduce the heat of reaction and not simply set the feed stream to maximum value.

8.5.1 Mismatch Between the RTO and MPC Layer

The first issue encountered after reducing the heat transfer coefficient ($UA/10$) is that the RTO layer still acts as though it has a perfect cooling capacity, and provides reference trajectories that the MPC layer is not able to follow. This is illustrated in Fig. 8.5.2, where the blue line represents the final simulated trajectory, the dashed lines are the reference trajectories from the RTO layer and the dotted lines are the predictions provided by the MPC layer. At the initial time, the RTO provides a temperature profile that is within the reactor temperature constraints. This profile, however does not correspond with the reactor temperature prediction of the MPC layer. Even with constraints on the reactor temperature in the MPC layer, the MPC is not able to avoid an overshoot in temperature. The RTO re-optimises and predicts that the temperature should decline back between the constraints. The MPC layer, on the other hand, does not have enough cooling capacity in the jacket to follow the predicted temperature profile and the reactor temperature continues to rise. However, the mismatch between the predictions are not as significant for the feed stream profile. As seen in the bottom graph in Fig. 8.5.2, the feed flow rate profile is maximised in the RTO layer. The MPC prediction also follows this profile.

The overshoot effect in Fig. 8.5.2 can be considered to be a weighting problem, where the MPC layer follows the feed rate profile rather than the temperature profile, or that the reactor temperature constraints are not weighted enough in the MPC layer. The constraints are not hard constraints as this may easily create an infeasible problem and result in no solution to the optimisation problem, as discussed in Section 4.2.1. An infeasible problem may occur if two or more constraints are mutually exclusive, meaning that both cannot be fulfilled at the same time. A hard constraint does not allow for any slack on the constraint. Only quadratic and linear weights are applied on the reactor temperature constraints, and these have to be chosen large enough to impact the objective function.

To ensure that the RTO layer is aware of the heat transfer limitations, the cooling constraint described in Section 8.3, Eq. 8.3.5, is added to the optimisation problem. This is an inequality constraint indicating that the cooling capacity, i.e. the maximum cooling effect of the cooling jacket, must always be larger or equal to the cooling demand, i.e. the

cooling necessary to keep the desired reactor temperature. In CENIT this constraint is posed as a soft constraint with high linear and quadratic values. It could also have been posed as a hard constraint because of the importance of keeping the heat of reaction low. However, as stated previously, hard constraints may lead to infeasibility and no solution to the optimisation problem.

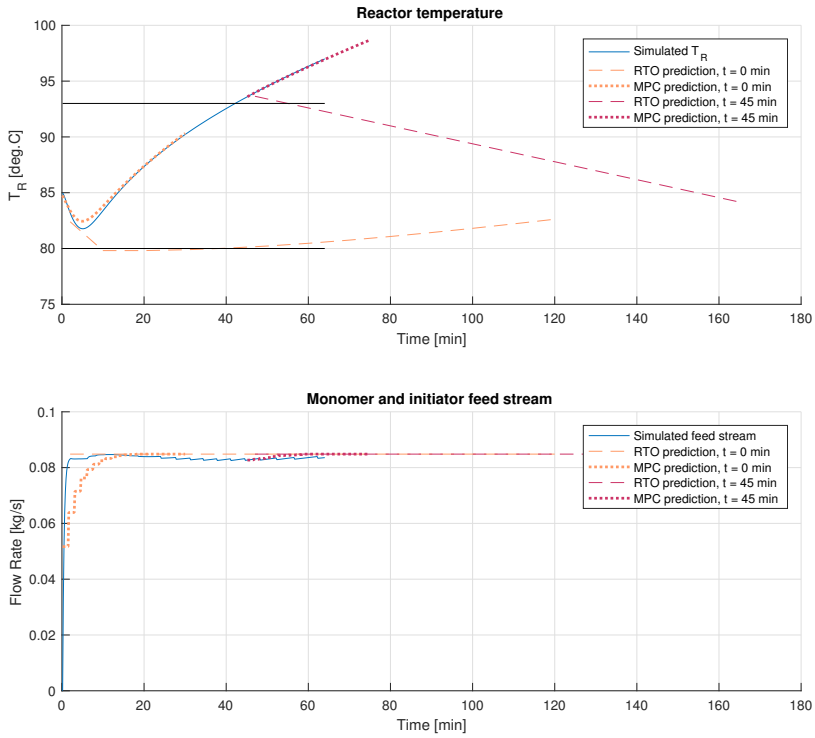

Figure 8.5.2: With no constraints on the cooling capacity and reduced heat transfer coefficient (UA), the RTO layer is not able to provide realistic temperature and feed rate profiles that the MPC layer can follow.

Fig. 8.5.3 plots the different variables from one of the first simulations when the new cooling capacity constraint is added to the optimisation problem. The polymerisation time is marked with the vertical line, and is 210 minutes. The cooling constraint variable is shown in the fourth plot, and shows the variable lying close to its constraint at zero. There are two important details to point out in this plot: M_n slightly breaks its constraint at the end of the batch, and the cooling capacity constraint is broken significantly at the end of the batch. To ensure that the tank is emptied at the end of the batch, control of the feed rate is lost when the tank is almost empty, and the recipe rate is applied to the remaining contents of the tank. However, a pulse of monomer and initiator that is added to he tank

will create a large amount of reaction heat, that the jacket cannot remove. This results in a "spike" in the cooling capacity constraint, because the reaction continues a bit after the feed has stopped, and so the cooling effect of the feed is lost. This spike is not avoided by adjusting the jacket temperature, as it is already at its minimum temperature.

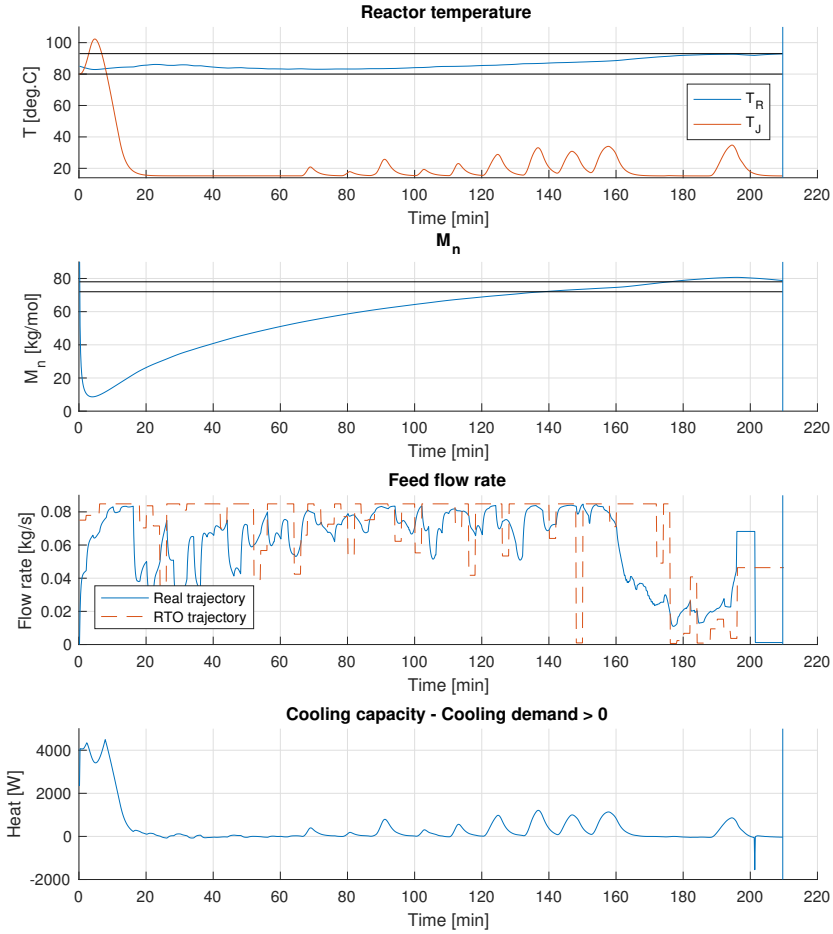

Figure 8.5.3: Constraints on the difference between the max cooling capacity and the cooling demand. Heat transfer coefficient: $UA/10$

The violation of the constraint on M_n can be explained by the feed stream rate. This is illustrated in the third plot in Fig. 8.5.3, where the dashed red line gives the inputs from the RTO layer and the blue solid line gives the MPC input. One can observe a major

mismatch in the two trajectories. M_n is only a part of the optimisation problem for the RTO layer and is not important in the MPC layer. However, M_n is highly dependent on the feed stream as it contains both monomer and initiator. The RTO creates the optimal feed stream trajectory to obtain the correct end-product quality, and when this is not followed, the end-product quality cannot be guaranteed to be satisfactory.

Significant oscillations are observed in the jacket temperature, the cooling capacity, and the feed stream, indicating that the system is not stable, and that the variables are reliant on each other. There is a connection between the jacket temperature and the feed stream. When the feed stream increases, one can observe a slight augmentation in the jacket temperature. This is because of the cooling effect from the feed stream, that is so significant that it requires the jacket temperature to increase. This cooling effect and how it impacts the system is discussed in detail in Section 8.5.2. The oscillations in the feed stream are a result of the cooling capacity constraint. When the constraint is broken in the prediction horizon, the system reacts by decreasing the feed flow. Then, to minimise the polymerisation time, the feed is increased, creating more heat which violates the cooling capacity constraint.

In Fig. 8.5.3 the tank is forced to empty. When there is only 10 kg of monomer remaining in the tank, the control of the feed rate is removed, and the tank is forced to empty into the reactor. This is visible in the feed stream profile as the final block structure right before 200 minutes. However, this is not ideal for the optimisation problem. Preferably the feed rate should be optimised for the whole tank, also the end. This is the case in Fig. 8.5.4. In this simulation, the process has been re-tuned, to obtain a better general result. The weighting problem will be further discussed in Section 8.6. For the simulation in Fig. 8.5.4 the polymerisation time is 219 minutes, slightly longer than for the previous case. This is due to the feed stream which is slightly lower. The oscillations have disappeared, and the MPC layer follows the RTO predictions on the feed stream significantly better.

However, the cooling capacity constraint is still broken at the end of the batch, at the last spike in the feed stream. The jacket temperature constraint is already active, implying that the jacket temperature input is saturated, which means that one degree of freedom is lost and that the the jacket temperature cannot be used to avoid the violation of the cooling capacity constraint.

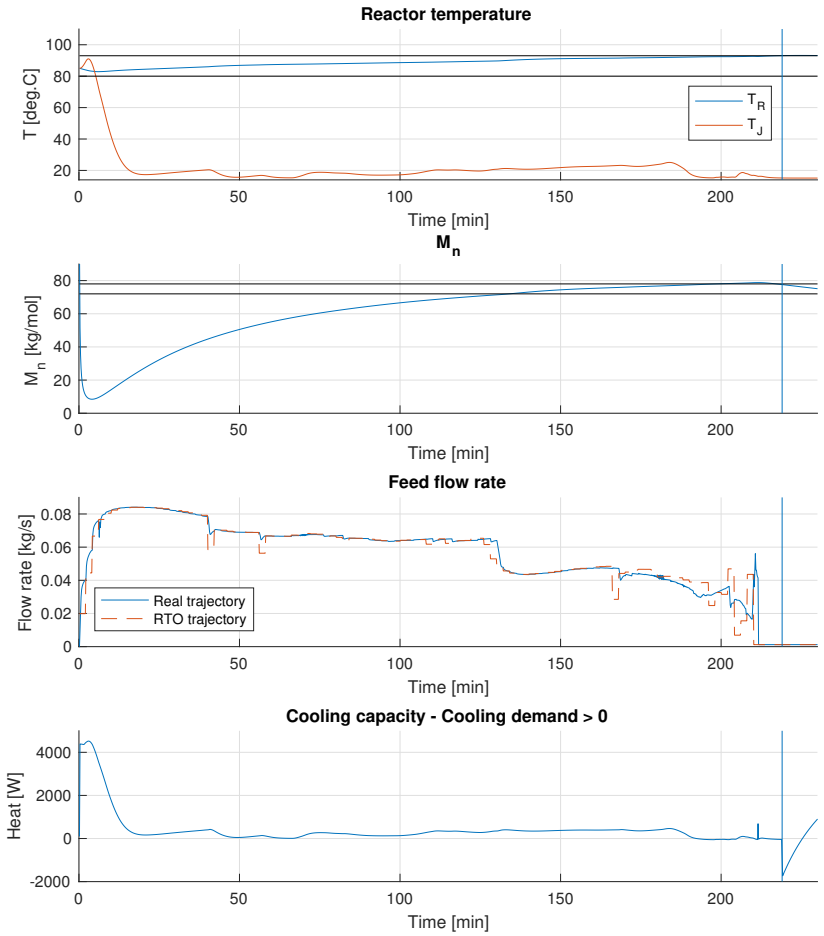

Figure 8.5.4: Simulation of process with reduced heat transfer coefficient: $UA/10$.

The simulated feed stream follows the predictions from the RTO layer relatively well. The development of the feed stream predictions are shown in Fig. 8.5.5, where the prediction in the RTO layer at $t = 0$ min, $t = 20$ min and $t = 80$ min is shown. Since the model is perfect, i.e the same model is used in RealSim for calculation of the "real" process measurements and for the estimations of the measurements and states from CENIT, the initial predicted profile at $t = 0$ should be able to hold throughout the batch. However, this is not the case as the feed stream increases higher than the initial profile the first 50 min of the polymerisation reaction, and then is slightly lower than the initially predicted profile. At the re-optimisation at 20 min the profile fits well, and also at 80 min. The

incompatibility between the two layers is most distinct at the end of the batch, resulting in a longer polymerisation time than predicted.

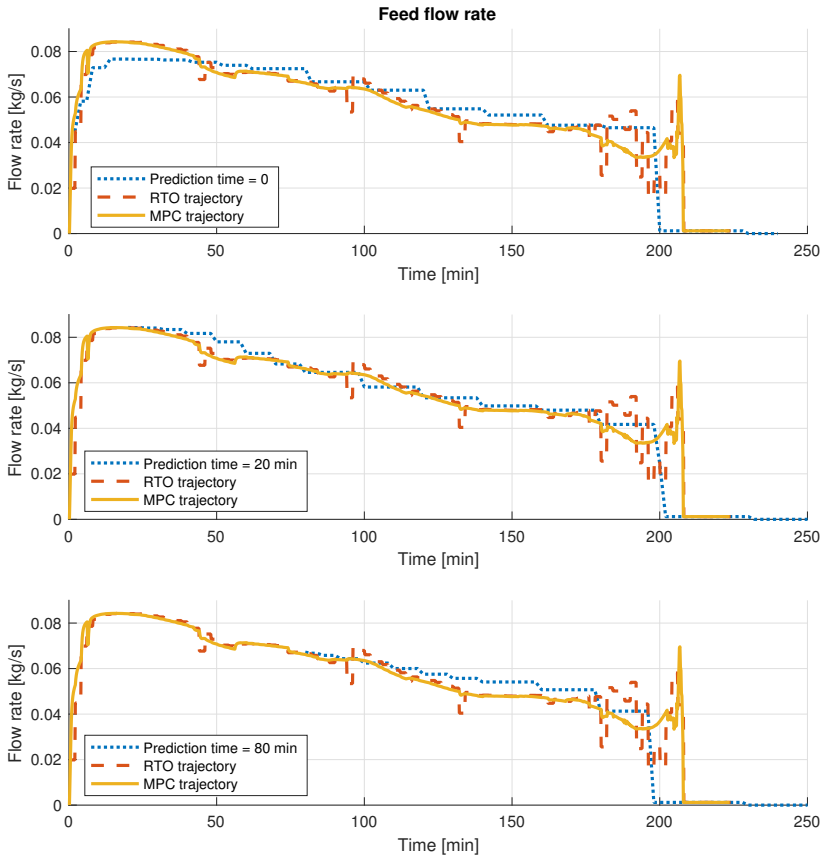

Figure 8.5.5: Simulation of process with reduced heat transfer coefficient: $UA/10$. The Figure shows the predictions of the feed stream trajectories at different sample times and compares it to the final simulated trajectory.

8.5.2 The Cooling Effect of the Monomer/Initiator Feed Stream

Fig. 8.5.3 shows quite large oscillations in the feed stream profile. They are a result of the cooling capacity constraint which is violated in the prediction horizon. The cooling effect of the feed stream is very large, around half of the effect of the heat of reaction. When the feed flow rate decreases, the constraint is held for a time until the heat of reaction increases and the constraint is broken, resulting in a new decrease in the input feed stream. Because

of the instant cooling effect, the optimisation problem may also find it optimal to cool the reactor by increasing the feed. This again leads to the cooling capacity constraint being broken because of the increase in heat of reaction that follows an increase in monomer concentration. To avoid these big responses, the MV of the feed stream is in the MPC layer set to first order hold, as explained in Section 4.2.1. This removes the blocks and creates a smoother input prediction and a less aggressive behaviour of the cooling capacity constraint.

In all the previous simulations there has been a "spike" that breaks the cooling capacity constraint at the end of the batch. This can be explained by the abrupt stop of feed stream. The feed stream contributes to a large cooling effect in the reactor. It enters at $25\,^{\circ}\mathrm{C}$, and the reactor temperature lies around $90\,^{\circ}\mathrm{C}$. When the feed stops, the cooling effect is lost. However, there is still a good amount of monomer and initiator that is reacting and creates heat.

The main issue with the "spike" is linked to the way the CENIT algorithm works. The cooling capacity variable is weighted linearly as a linear weight will still provide sensitivity of the constraint in the optimisation problem even when the constraint is only slightly broken. This can be considered as a harder way to weight the constraints than using only quadratic weights. The downside of linearly weighted constraints is that only the largest violation of the constraint is considered and weighted. This allows for other smaller violations of the constraints earlier in the batch.

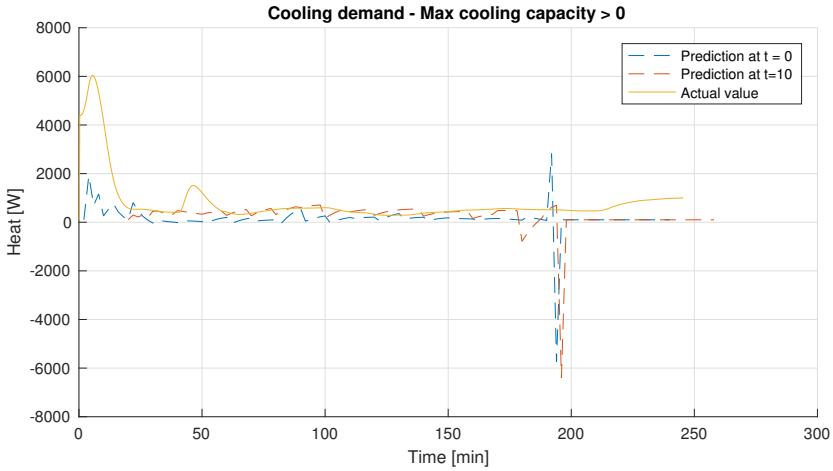

Figure 8.5.6: Figure presenting the cooling capacity constraint. In the predicted trajectories the constraint is violated with "spike", though the final simulated variable shows that the constraint is not violated.

The RTO layers does not predict a rise in temperature as a reaction to the violation of the constraint. Fig. 8.5.6 shows that even though the cooling capacity constraint is never broken, the predictions early in the simulations show the spike. However, the optimisation

problem adapts further out in the batch. To avoid the spike, the feed flow rate is gradually decreased instead of ending abruptly. This is illustrated in Fig. 8.5.7.

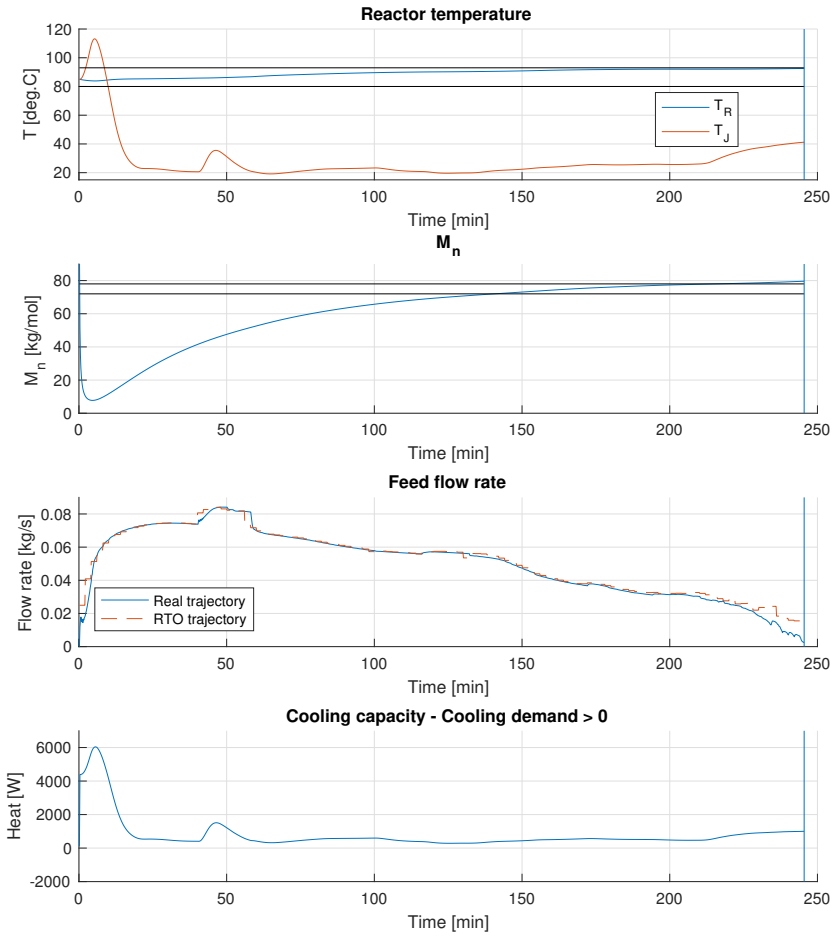

Figure 8.5.7: Simulation of process with reduced heat transfer coefficient: $U_j/10$.

An option that was explored to avoid this problem was to "remove" the spike in the prediction horizon. When the tank was nearly empty, the cooling constraint variable was forced to 0, keeping the constraint active, but not violated. However this does not provide satisfactory behaviour, since the spike is still present in the plant. The same problem as previously was observed where the RTO layer predicts a temperature profile that the MPC cannot follow. The problem with removing the spike is that the increase in reactor tem-

perature will still occur, but the RTO layer will not adjust the feed stream to avoid this. This effect is only an issue at the end of the batch. The MPC layer only sees 30 minutes into the future, and does not predict this rise in temperature until it is too late. Even with a constraint on the reactor temperature in the MPC layer, it is not capable of avoiding the increase in reactor temperature. This is also because the jacket temperature is already saturated, so this degree of freedom is lost. These effects are plotted in Fig. 8.5.8

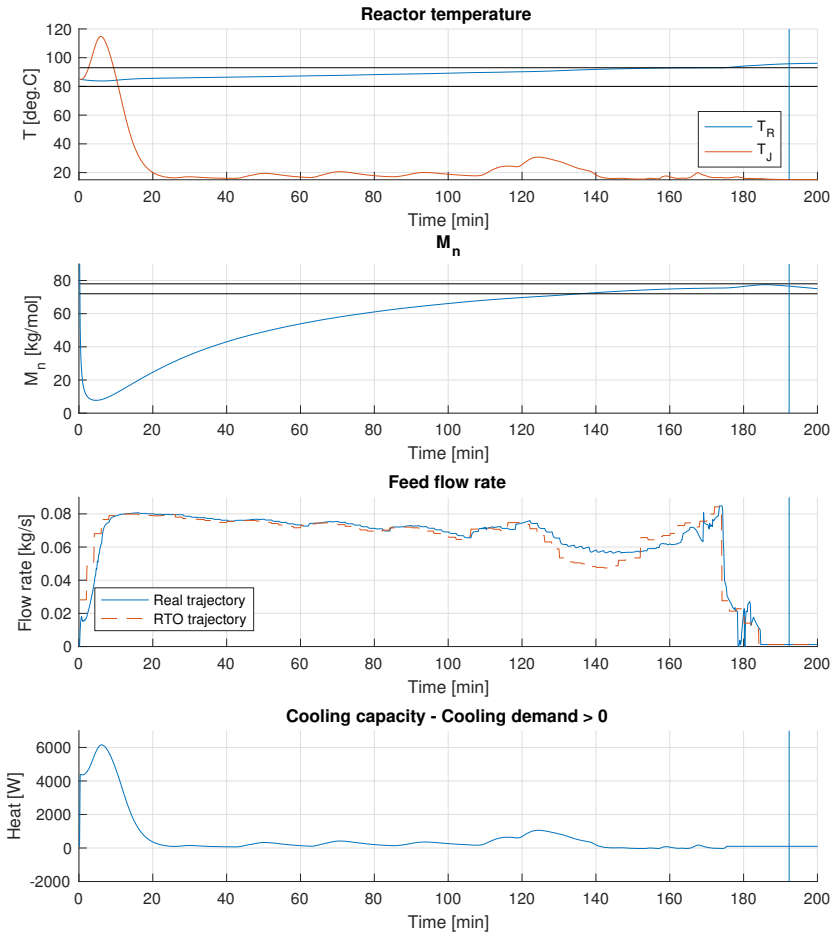

Figure 8.5.8: Simulation of process when the heat created at the end of the batch is not observable by the model in both the RTO and the MPC layer. This results in violation of the temperature constraint at the end of the batch.

The polymerisation time in Fig. 8.5.8 when the spike is removed from the model is 192 minutes, shorter than the polymerisation time of 245 minutes in Fig. 8.5.7 with the spike. The comparison in the reactor temperature and feed stream in these two cases can be observed in Fig. 8.5.9.

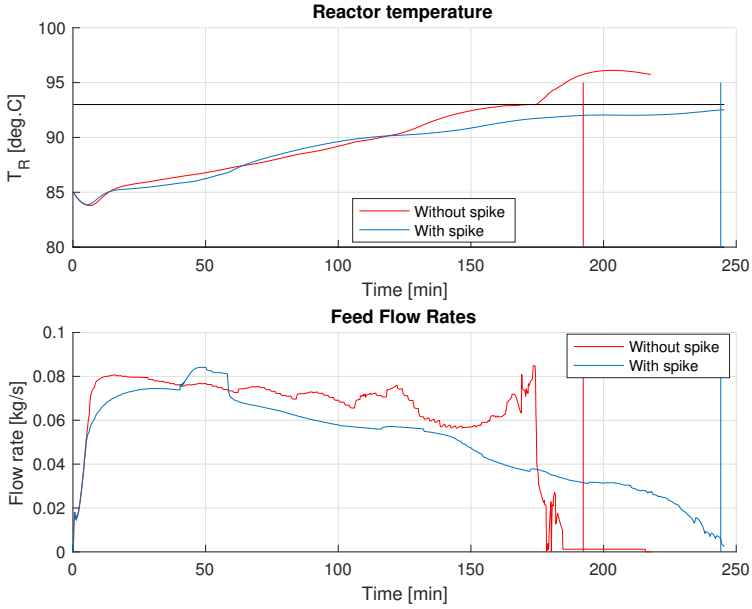

Figure 8.5.9: The simulations of the temperature and feed stream when the spike is observable and when it is not.

Fig. 8.5.9 shows that when the spike is not present in the predictions, the reactor temperature constraint is violated and so this simulation is not a solution to the optimisation problem. The reactor temperature profiles that are created by the RTO layer and the MPC layer are plotted in Fig. 8.5.10. The RTO layer never predicts the rise in temperature, not at the start of the batch, nor at sample $t = 150$ min, directly before the event occurs. The MPC layer detects the temperature rise at the sample at 150 minutes and at 170 minutes, but is not able to prevent it.

To recapitulate before moving on: to avoid violation of the reactor temperature constraint, which happens at a sudden stop of feed stream, the feed stream is gradually decreased in the optimal solution of the problem. However, the RTO layer is not able to predict this at the initial time samples, but adapts further out in the batch. Furthermore, when the feed stream decreases steadily the polymerisation time increases. Nevertheless this avoids violation of the reactor temperature constraints and cooling capacity constraints.

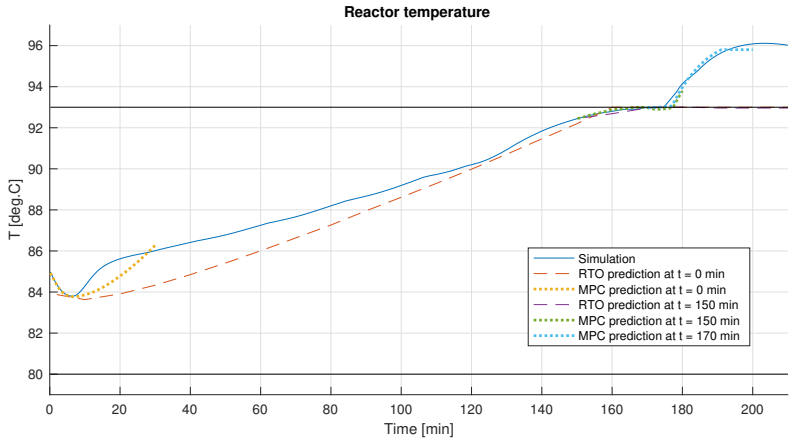

Figure 8.5.10: Plot of the prediction of the reactor temperature at different times in the RTO and the MPC layer when the spike removed from the model.

It is apparent that the feed stream provides a considerable cooling effect to the reactor. When the feed stops and this cooling is no longer available, the jacket is not able to compensate, resulting in the violation of the cooling capacity constraint, or a longer polymerisation time because the feed stream decreases slowly towards the end of the batch. To avoid this situation, the cooling capacity has to be weighted much harder to avoid the spike to violate the constraint. Another solution is to remove the cooling effect from the feed from the cooling demand equation all together. This yields a lower feed flow rate and a higher jacket temperature. Implementing this change in the cooling demand also improves the safety regulation of the reactor. Should the feed suddenly stop during operation for some reason, the jacket will be able to remove the excess heat and avoid thermal runaway. When this is done for the heat transfer coefficient divided by ten (UA/10) the polymerisation time becomes so long that the RTO layer does not see the end of the batch. Adjusting the heat transfer coefficient it is found that UA/4 is an interesting case to study. The result is plotted in Fig. 8.5.11. The polymerisation time in this case is 235 minutes, which is shorter than when the spike is considered. However, the time is not really comparable between the two cases, since the heat transfer number is not the same.

Examining the simulation in Fig. 8.5.11 more closely, in which the cooling effect of the feed is removed from the cooling capacity constraint and $UA/4$, the same tendencies as previously can be observed. Even though the MPC follows the RTO prediction of the feed stream very well during almost the entire batch, when the end of the batch is near it does not follow. The simulation is valid, since none of the constraints are violated. However, the new setup did not result in the desired behaviour. The idea was that removing the cooling effect from the feed stream would allow the feed stream to end abruptly, and compensate for the loss of cooling by reducing the temperature of the jacket. A small dip in the jacket temperature is observed, but it rises again quickly.

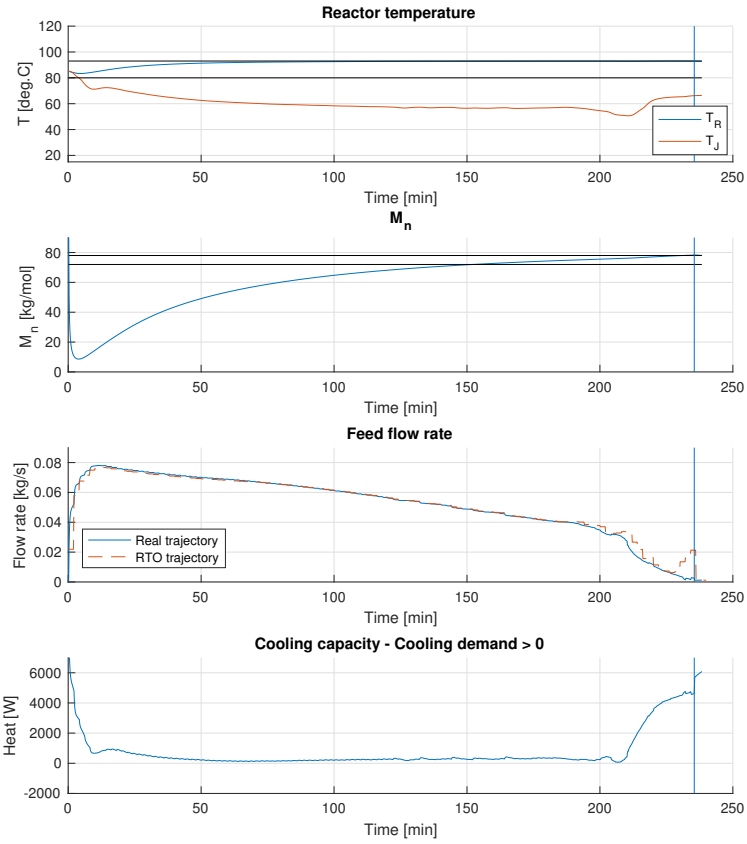

Figure 8.5.11: Plot of the solution when the cooling effect from the feed stream is removed from the *CoolingDemand* equation. The heat transfer coefficient is divided by 4. None of the constraints are violated. The jacket temperature is not saturated, and can be used to cool the reactor in case the feed stream should suddenly stop prematurely.

8.6 The Effect of Tuning and Parametrisation

Some of the previous figures have already shown the importance of weighting, and how much the solution of the optimisation problem may vary for different tuning. It is important to identify which variables the problem is sensitive to. Changing the weight of some parameter may yield no difference in the polymerisation time, while others result in greater variations. For the MPC layer, the challenge lies in which reference trajectory it should prioritise to follow, the reactor temperature or the feed stream of monomer and

initiator. In the early stages of the work when the cooling effect from the feed stream was included in the cooling capacity variable, the MPC layer attempted to control the reactor temperature with the feed stream in stead of the jacket temperature. This could be a result of the instant cooling effect of the feed stream, while there is a 2 minute time constant in the response of the jacket temperature. For the RTO layer the challenge is obtaining the shortest polymerisation time while remaining within the constraints of the final value of M_n and the reactor temperature. Also, the RTO layer must provide feasible profiles that the MPC can follow. This section will discuss the sensitivity in the polymerisation time for different weights on the polymerisation time, following the reference trajectory of the reactor temperature and the parametrisation and following the reference trajectory of the feed stream.

8.6.1 Polymerisation Time

One of the objectives in this thesis is to minimise the polymerisation time in the process. In the optimisation problem, the *PolymerisationTime* variable is the objective function, and the setpoint is set to 0. *PolymerisationTime* is parametrised regularly throughout the prediction horizon, with a coincidence point every 4^{th} sample. Due to the definition of the *PolymerisationTime* variable, Eq. 8.2.3, the setpoint will never be attained, but this will force the system to work towards a short polymerisation time. The variable can be weighted differently in the Q weighting matrix, implying the importance of reaching a short polymerisation time. However, if it is weighted too hard, this could mean that other constraints are violated.

Fig. 8.6.1 shows the solution to the optimisation problem with different weights on the polymerisation time. The vertical line represents the polymerisation time, and some sensitivity to the weighting of the setpoint can be observed. When the setpoint is weighted harder, the polymerisation time decreases. This is mainly due to the feed rate profile which is provides higher feed stream nearer the end of the batch. In all three cases the other constraints are obeyed. However, from the lowest weighting to the highest, there is a 10 minute difference, a 4.2 % reduction, in the polymerisation time. Though there is a sensitivity in the system to the weighting of this setpoint, it does not have an enormous effect. Table 8.6.1 shows the different polymerisation times for the different weights on the setpoint.

Weight	Polymerisation time [min]
0.5	235.10
1	234.00
10	225.00

Table 8.6.1: Table showing the polymerisation time for different weights on the polymerisation time setpoint in the RTO layer.

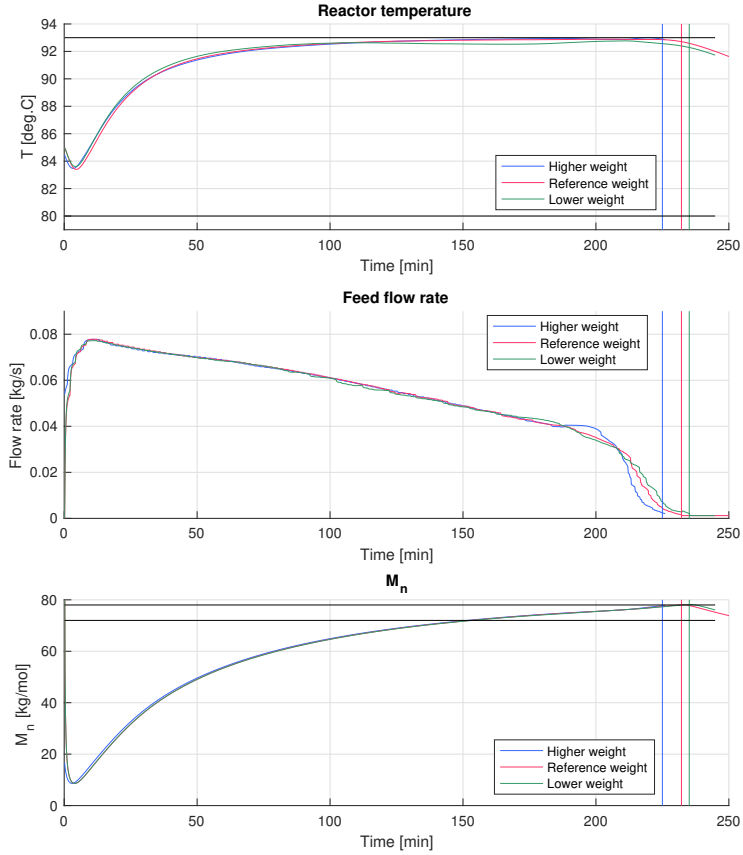

Figure 8.6.1: Plot of the solutions for different weights on the polymerisation time setpoint. The polymerisation time is represented by the vertical line in the corresponding colour. None of the constraints are violated, and weighting the polymerisation time more in the RTO layer results in a shorter polymerisation time.

8.6.2 Reactor Temperature Profile

The temperature in the reactor is important to control, which is why it is important that the temperature model is accurate. Thermal runaway must be avoided for safety reasons. The temperature will also influence the polymerisation time, as the reaction kinetics depend on the temperature, and it will also influence the value of M_n. The reactor temperature profile is calculated in the RTO layer and read by the MPC layer. This profile is used as a reference trajectory the MPC has to follow. The RTO layer uses the derivative of the temperature as an MV to determine the reactor temperature. This results in a smooth

temperature profile, and not a blocked one, as it would have been if the reactor temperature was used as an input directly.

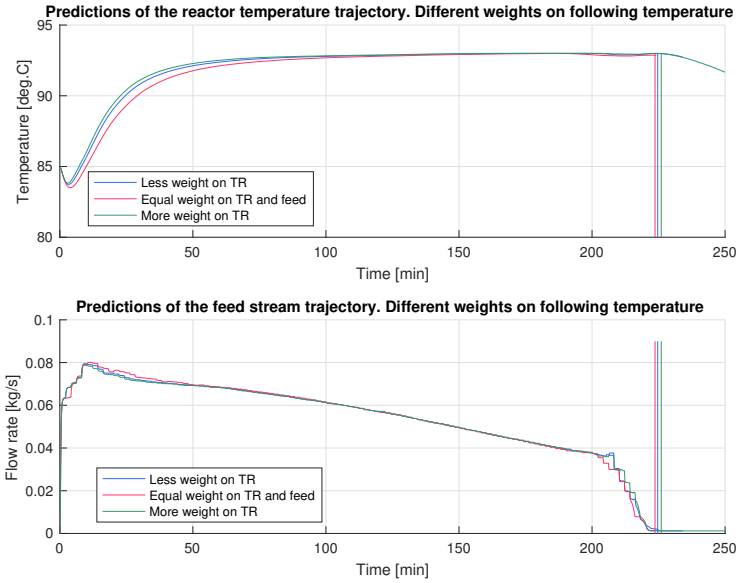

Figure 8.6.2: Plot of the solutions for different weights on following the reactor temperature profile. This figure shows that there is little sensitivity in the polymerisation time to different weights on the temperature reference trajectory..

In the MPC layer the feed stream and the jacket temperature are the two inputs that will affect the reactor temperature. The jacket temperature setpoint is parametrised evenly throughout the prediction horizon. A 2 minute time constant is added on the jacket temperature equation, meaning that from the setpoint is applied to the process the effect is not instant but it will take two minutes until this setpoint is reached. This time constant is added so the model corresponds better with the real-life conditions. Changing the jacket temperature setpoint does not mean that the jacket temperature reaches this value immediately. There are three tuning parameters regarding the temperature profile in the MPC layer: weighting on following the temperature trajectory, penalty on the movement of the input (for the MPC layer this is the inlet jacket temperature setpoint) and parametrisation of the reactor temperature. As discussed in Section 4.2.1, the higher the frequency of coincidence points, the tighter the control of the reactor temperature. Fig. 8.6.2 shows the effect of weighting the reference profile of the temperature more or less compared to the feed stream profile. There is almost no sensitivity in the polymerisation time, though equal weights on the reference trajectories minimises the polymerisation time slightly. This could be because with equal weights the optimisation problem has the freedom to choose which trajectory to follow. The exact polymerisation times for different weighting on following the reference trajectory is shown in Table 8.6.2. When weights are equal

on following the reactor temperature trajectory and the feed flow rate trajectory (Qfeed = Qtemp = 1), the polymerisation time is 1.33 minutes shorter than for Qtemp = 20, i.e. a decrease in polymerisation time of 0.6 %.

Weight	Polymerisation time [min]
0.5	224.00
1	223.67
20	225.00

Table 8.6.2: Table showing the polymerisation time for different weights on following the given RTO reactor temperature trajectory.

8.6.3 Monomer and Initiator Feed Stream Profile

The feed stream influences the polymerisation time directly, because the rate determines how rapidly the tank containing monomer is emptied into the reactor. The larger the feed stream, the lower the polymerisation time. However, under conditions with a lower heat transfer coefficient, a high feed stream will result in a high reaction rate and much heat of reaction that needs to be transported away. As a consequence, the feed flow rate is lowered, and the polymerisation time increases.

In the RTO layer, it was shown that the input variable for the feed rate is sensitive to the weight on the polymerisation time setpoint. The heavier the polymerisation time is weighted, the quicker the tank wants to empty, the higher the feed rate. In the MPC layer, the input variable for the feed rate depends on how hard following the feed rate reference trajectory is weighted, the penalty on change in input value, Δu, and how the input variable is parametrised.

Feed Stream Trajectory

When setting up the control structure the decision on the MVs and CVs in the two layers will influence the behaviour and the solution of the system. Figs. 8.6.3 and 8.6.4 show the effect of the feed rate on the problem, with $UA/4$ and the cooling effect of the feed stream removed from the cooling demand equation for solutions to different setups. The green line represents the recipe rate of the feed stream which the MPC has to follow, resulting in a 1DOF system. This gives by far the shortest polymerisation time corresponding to the average polymerisation time of the pilot plant (represented with the vertical lines in the corresponding colours). The cooling capacity constraint is violated, but the jacket temperature is able to compensate so that the reactor temperature remains within its constraints. However, the recipe rate is given for the pilot process with sufficient heat transfer.

The red line represents the feed stream when the RTO input is applied directly. This gives a 2DOF system in the RTO layer, but 1DOF system in the MPC layer. The polymerisation time is relatively short, and M_n and the reactor temperature are within its constraints. Since the termination of the feed stream is not as abrupt as the recipe rate, the cooling

capacity constraint is not violated as much as for the recipe rate, but it is still violated. The jacket temperature remains within its constraints and so does M_n and the reactor temperature. The only time the cooling capacity constraint is not violated is when the MPC layer is also using the feed as an input, for the blue and yellow plots. All the other variables are also within their constraints. The polymerisation time is a few minutes shorter when the MPC layer weights following the RTO profile feed. The exact polymerisation times are given in Table 8.6.3.

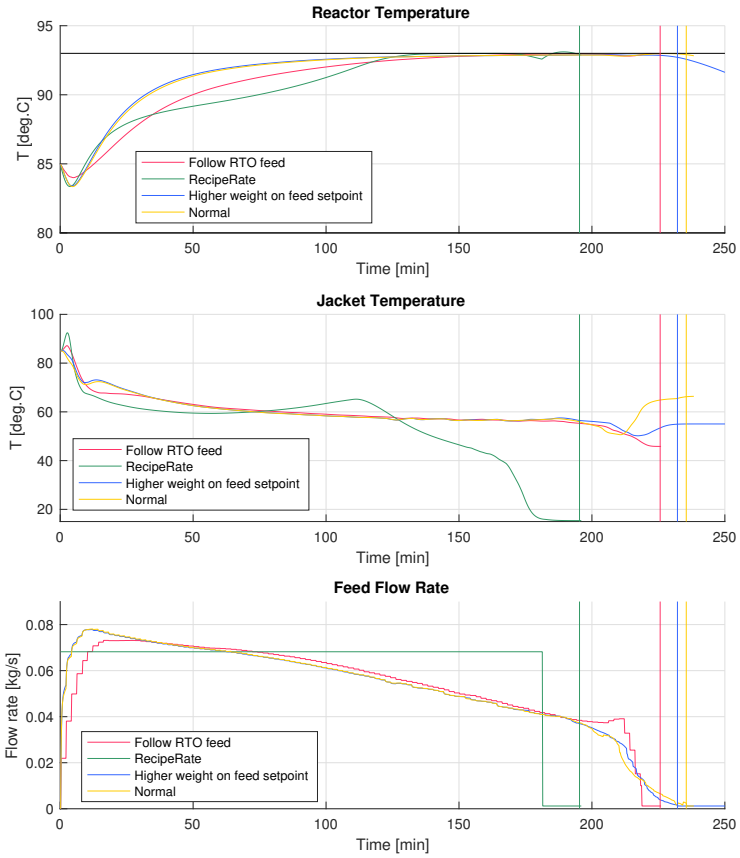

Figure 8.6.3: Plot of the solution trajectories for the reactor temperature, jacket temperature and feed stream for different conditions on the feed stream. The heat transfer coefficient is reduced (UA/4). The reactor temperature constraint is slightly violated for the recipe feed flow rate.

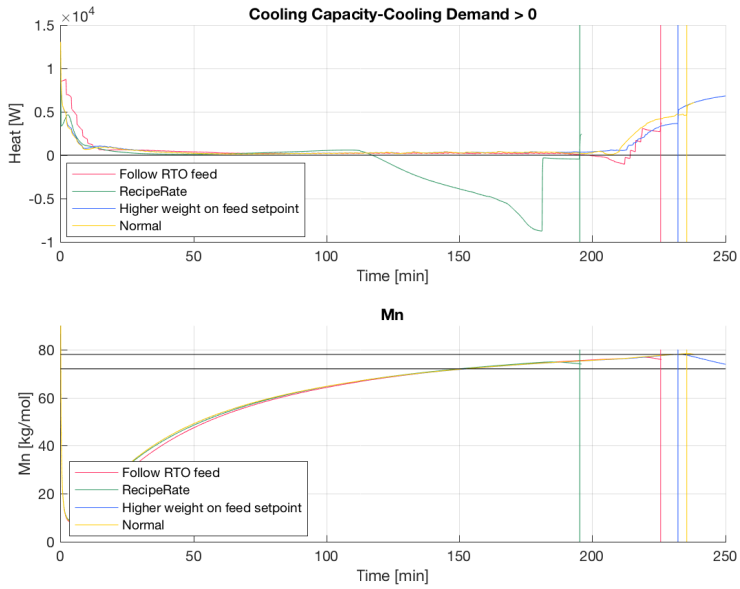

Figure 8.6.4: Plot of the solution trajectories for M_n and the cooling capacity constraint for different conditions on the feed stream. The cooling capacity constraint is violated both for the recipe feed rate and when the RTO layer input moves were applied directly.

Feed stream	Polymerisation time [min]
Recipe Rate	195.33
Direct RTO trajcetory	225.67
Original weights	238.30
Harder weights on feed trajectory	235.48

Table 8.6.3: Table showing the polymerisation time for different feed stream trajectories.

The Effect of Parametrisation

The parametrisation, i.e. the input blocking described in Section 4.2.1, could also affect the optimisation problem. The aim of input blocking is to decrease the number of decision variables but still follow the dynamic tendencies of the variable had it been continuous. In the preliminary project, the trajectory was calculated off-line at the beginning of the batch. Therefore there were shorter blocks in the beginning of the prediction horizon and near the end. This captured the rapid increase and decrease of the flow rate, and had larger blocking in the middle where the flow rate was approximately constant. However, for an on-line simulation, the control horizon is constantly moving. Half-way in the simulation the rapid dynamics of the decrease in the feed flow is situated around the large input blocks.

For the on-line optimisation problem, it could be advantageous to keep the input blocking in the RTO layer at equal intervals, so that the dynamics remain present throughout the prediction horizon.

However, Fig. 8.6.5 shows that the parametrisation does not effect the final polymerisation time, nor the solution to the optimisation problem significantly. The simulated feed flow rates are nearly identical. In the initial sample, the even parametrisation prediction has a slightly shorter polymerisation time, but for the rest, the prediction of the polymerisation time is identical. In the final simulated trajectory, there are some oscillations for the uneven blocking, but this could probably be removed with different tuning.

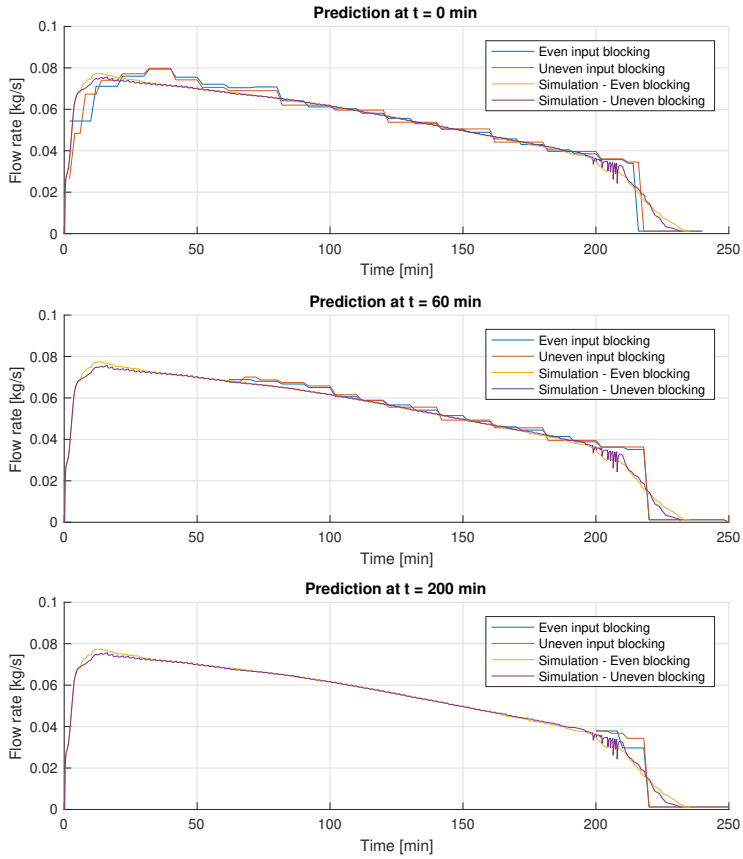

Figure 8.6.5: Plot of the prediction of the feed stream at different sample times for different parametrisations of the feed stream input variable.

The main reason for input blocking is to reduce the computation time, which has to be lower than the time between each sample. For this problem, the sample time is two minutes in the RTO layer. Keeping the old input blocking with fewer blocks gave a maximum computation time of around 35 seconds. Introducing more input blocks increases the computation time slightly, but it still lies well within the limits.

Tuning the Layers

The solution of the optimisation problem will depend on the penalties set for changing the input setpoint, and also the weights determining the importance of following the assigned feed stream trajectory. The values of the weights and penalties are not important in themselves, it is the relative value between them that is important.

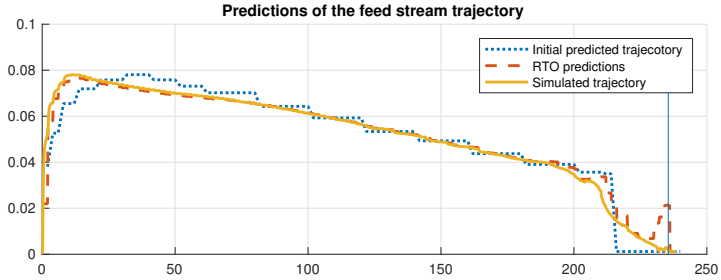

Figure 8.6.6: Plot of feed stream and the initial prediction of the feed stream profile. Illustrates that the initial prediction is not followed, and that the MPC lies ahead of the RTO predictions.

The first order hold, as described in Section 4.2.1 smooths the input moves. A property to the first order hold is that it looks into the future. To smooth out the input trajectory, it has to know the following sample value. When the first order hold is applied to the feed stream MV in the MPC layer, this implies that the MPC feed stream profile has a tendency to lie one step ahead of the RTO layer. As a result, the predicted feed stream from the RTO layer and the one from the MPC layer are not identical. This effect is shown in Fig. 8.6.6, where the first 20 minutes, during the increase in feed flow rate, the simulated trajectory (the yellow line) constantly lies above the RTO predicted input moves (the red dashed line). The same effect can be seen after around 200 minutes in Fig. 8.6.6 and to the end of the batch. The MPC feed starts decreasing more rapidly than the RTO trajectory suggests. The RTO layer predicts an abrupt stop in the feed. The MPC, however, observes the decrease in feed and adjusts accordingly by starting the descent in flow rate earlier. To avoid the premature decrease in the feed flow rate, the reference trajectory was set to a constant value when the end of the batch was observable for the MPC layer.

For some reason, the MPC layer does not stop the feed flow rate as abruptly as the RTO indicates that it could. One explanation could be the first order hold effect described in the previous paragraph. However, an attempt was made to see if this behaviour was a result of a too hard penalty on the input movement. This resulted in a oscillatory behaviour between

each RTO sample - it was too easy to change the input value. However, at the end of the batch, the MPC layer follows the RTO trajectory well and the feed flow rate ends quite abruptly. This is plotted in Fig. 8.6.7.

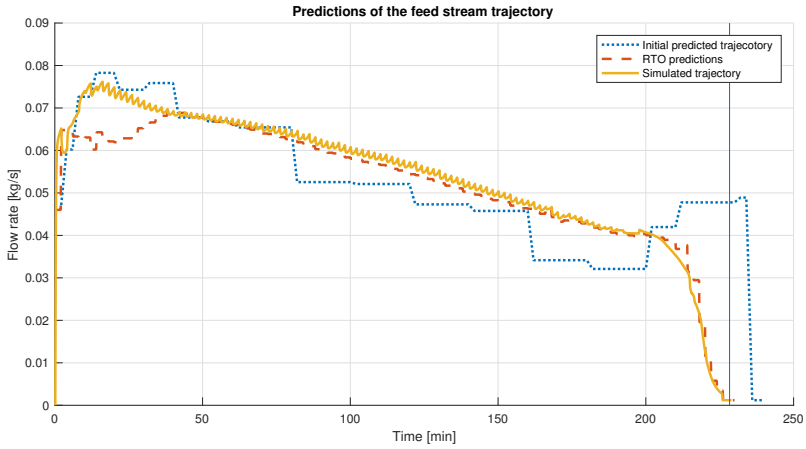

Figure 8.6.7: Plot of feed stream and the initial prediction of the feed stream profile. Almost no penalty on input movement.

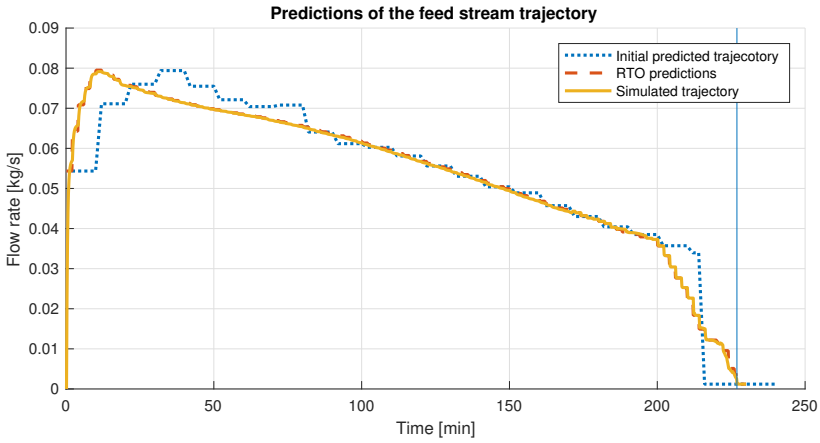

Figure 8.6.8: Plot of feed stream and the initial prediction of the feed stream profile. Almost no penalty on input movement at the end of the batch.

The best result was obtained when logic was included in the code allowing the weights to change at a certain point in the batch. Until the end of the batch, the old weights on input moves were applied to avoid the oscillatory behaviour observed in Fig. 8.6.7. At the end of the batch, these weights were almost removed, allowing the feed stream in the MPC

layer to end more abruptly. The result of this simulation is plotted in Fig. 8.6.8. The initial predicted trajectory of the RTO layer is not followed, however the MPC follows the RTO input directly, and the feed stream ends quite quickly.

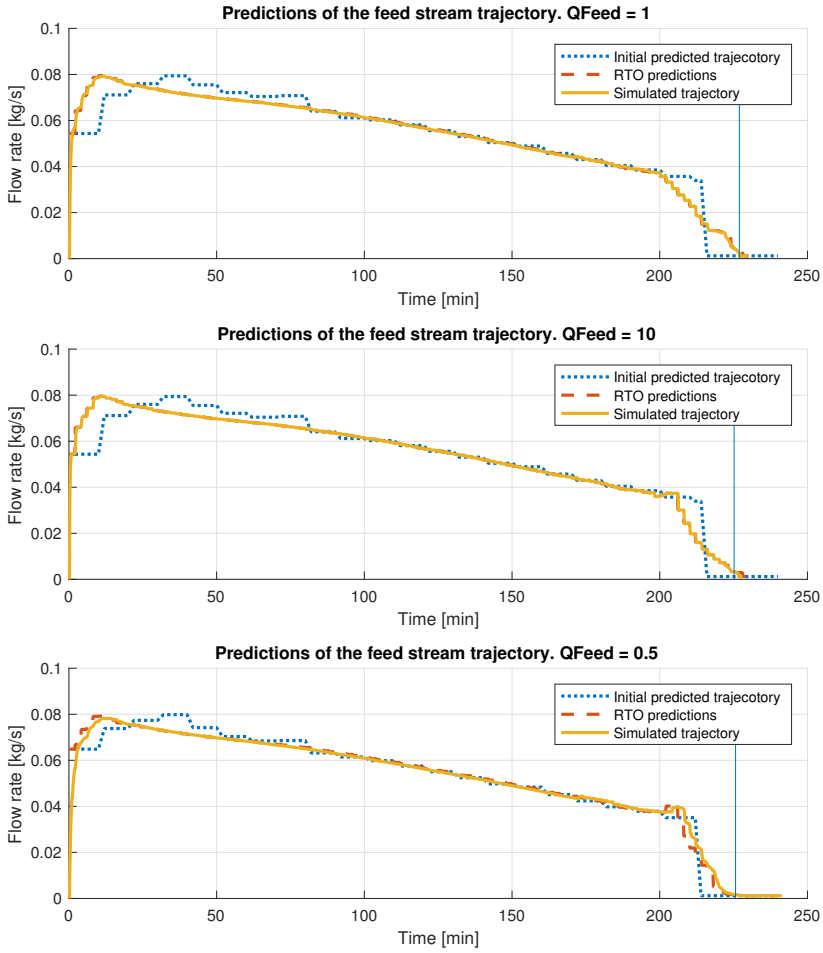

Figure 8.6.9: Plot of feed stream and the initial prediction of the feed stream profile, with three different weights on following the RTO trajectory. The shortest polymerisation time is obtained for Qfeed = 10, when the MPC layer strives to let the feed flow rate follow the RTO reference trajectory.

As for the reactor temperature reference trajectory, it is possible to weight the deviation from the reference trajectory from the feed flow rate in the MPC layer. The simulations from applying different weights to the following of the reference trajectory can be studied

in Fig. 8.6.9. There is no great difference in the polymerisation time, which is shown in Table 8.6.4. The polymerisation time is slightly shorter when the RTO is followed more closely, a result which is correlated with Fig. 8.6.3 where the RTO trajectory gives a better polymerisation time than when the MPC layer chooses the feed more freely. The exact values of the polymerisation time are listed in Table 8.6.4. The difference between the highest and lowest polymerisation time is 1.77 minutes, which corresponds to a 0.8 % decrease in polymerisation time.

Weight	Polymerisation time [min]
0.5	225.67
1	226.94
10	225.17

Table 8.6.4: Table showing the polymerisation time for different weights on following the given RTO feed stream trajectory.

8.7 Including the Kalman Filter

Up until now the model has been identical for the model estimates and for the measurements provided by RealSim, which functions as a process plant replacement. However, the model is never perfect, and the estimates will never match the measurements perfectly. This mismatch is also due to disturbances in the real plant as well as process and measurement noise. This mismatch can in most cases be reduced by introducing a Kalman Filter to the setup - or rather an Extended Kalman Filter since the model is nonlinear. The function of the EKF is described in Section 4.2.3. The EKF will change the parameters of the model so that the estimations better fit the measurements. The theory behind the Kalman Filter is explained in Section 4.2.3. Since there is a mismatch between the model used for estimation of the variables and the process, i.e. the measurements, the trajectories that are provided in the first samples will not give the correct end-product quality. It is important that the Kalman Filter quickly picks up on the error between estimates and measurements and changes the chosen parameter accordingly. The aim is that the model will adapt, recalculate the trajectories and provide the correct end-product quality. This section will explore the optimal solutions of the problem to a mismatch in different parameters and different tuning of the EKF.

8.7.1 Uncertainties in the Correction Factor for the Heat Transfer: KK_U_j

The jacket temperature plays an important role in determining the reactor temperature, but so does the heat transfer coefficient *UA*. This coefficient is uncertain and depends - among other things - on the viscosity and amount of the fluids in the reactor, and it is a challenging variable to estimate. The model includes a correction factor, KK_U_j, that is set equal to 1 by default. By changing this correction factor to 1.5 in CENIT but keeping the original

plant conditions, the CENIT model considers the heat transfer to be greater than it is in reality.

Setting up the EKF, one must choose which parameter to estimate and which measurement to rely on. KK_U_j was chosen with the corresponding reactor temperature as the measurement since this is a continuous measurement and fairly reliable. As a rule of thumb, there should be as many measurements as the number of parameters to estimate. The tuning of the EKF is done by setting values on the standard deviation from the measurement noise, white noise and process noise. CENIT and RealSim provide three different values for the measured variable: (1) RealSim provides y_m, the measured variable as if RealSim were the real plant, (2) y_p which is the *a priori* value predicted by CENIT without the updated correction from the measurements at the current sample, and (3) y_e the estimated value (*a posteriori*) after the Kalman Filter has re-tuned the parameters. Depending on the tuning, the predicted value or the measured value will be trusted more, and y_e will lie somewhere in between.

In CENIT there are three tuning parameters for the noise statistics, on the measurement noise, process noise and white noise. The process noise is modelled to enter through the parameters. The noise is added to the parameters before the state vector is updated. The white noise is added to the parameter vector and the state vector afterwords. The measurement noise is not the direct noise, but an indication of how uncertain the measurements are considered to be. It is added to the measurement vector after RealSim has calculated them. The noise can be added relative to the parameter or additive to the parameter, shown in Eqs. 8.7.1a and 8.7.1b, respectively.

$$par\_v[i] = xpar[i](1 + v[i]) \tag{8.7.1a}$$

$$par\_v[i] = xpar[i] + v[i] \tag{8.7.1b}$$

where par_v is the updated parameter vector, $xpar$ is the *a priori* parameter vector and v is the noise vector. The values in the noise vector can be the same for all parameters, or it can be defined specifically for one parameter, if for example one measurement is more uncertain than others. For the simulations in this thesis, the noise is considered to be added relatively to the parameters and states, like in Eq. 8.7.1a.

When the parameter mismatch lies on the same parameter that is being estimated in the Kalman Filter it is easy to determine whether it finds the correct value for the parameter or not - and how quickly. This is illustrated in Fig. 8.7.1 where the development of the estimation of KK_U_j is shown for different tuning of the process noise and the measurement noise. The white noise is constant at $1e-4$. For all the different tuning the KK_Uj parameter is estimated back to the correct value of 1 quite rapidly, within the first 40 minutes of the batch. The blue and green lines are almost identical, the only difference being the amount of process noise. The red line corresponds to more measurement noise, and this is the case where the estimator uses the longest time to estimate the correct value of the parameter because the measurements are considered to be very uncertain compared to the process model. Contrarily, when the process noise and the measurement noise are equally small, the estimator is very quick to find the correct parameter value.

The effect of the different tuning on the process outputs can be seen in Fig. 8.7.2, as well as the polymerisation time. M_n lies within its constraints, and so does the reactor temperature. The largest variation to the different tuning can be observer in the plot for the feed flow stream. The polymerisation time is shortest in the case with the largest measurement error (the red line) because it takes longer for the estimator to reduce the parameter in CENIT. However this does not affect the end-product quality of the polymer. The exact polymerisation times are stated in Table 8.7.1.

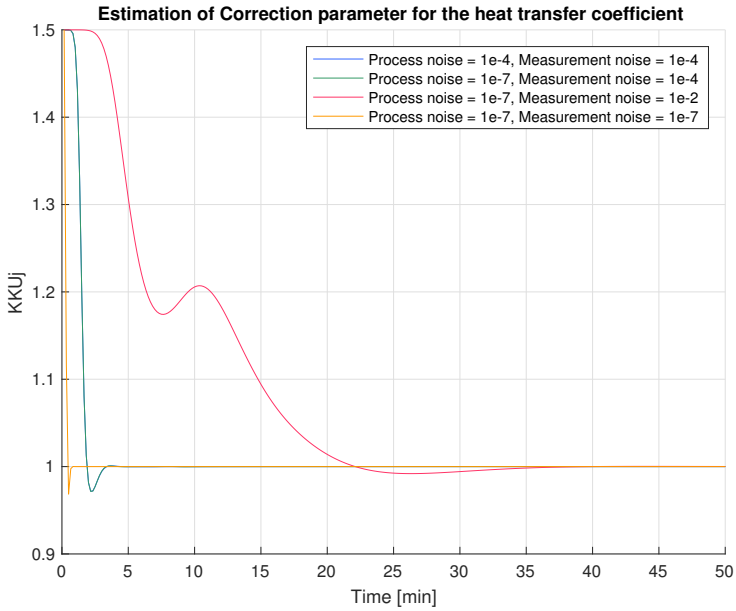

Figure 8.7.1: Plot of the development in KK_Uj for different EKF tuning.

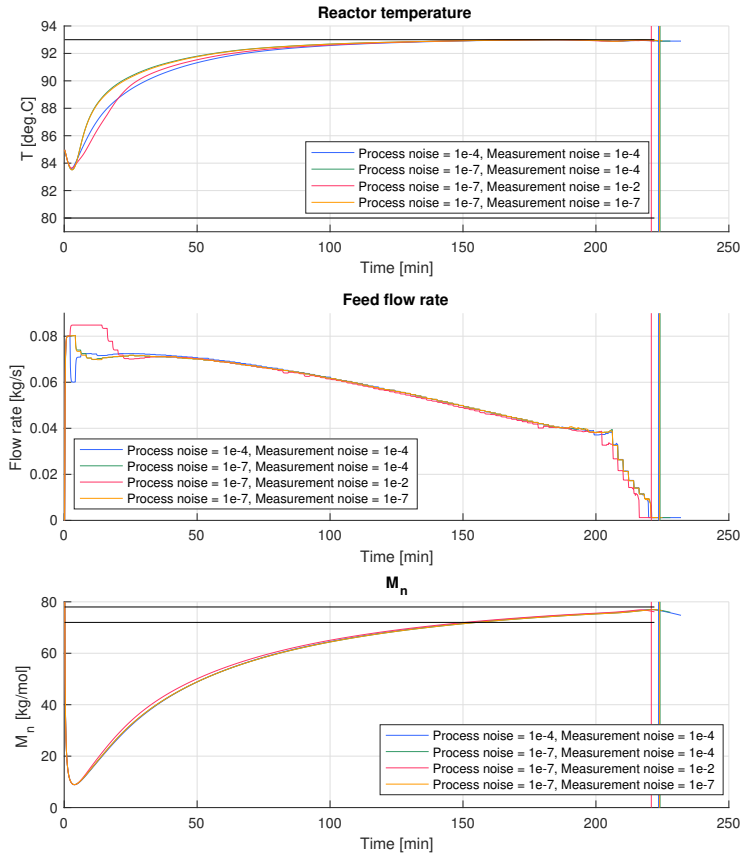

Figure 8.7.2: Plot of the reactor temperatures, feed flow rate and value of M_n for different EKF tuning.

Tuning	Polymerisation time [min]
Process noise = 1e-4, Measurement noise = 1e-4	223.57
Process noise = 1e-7, Measurement noise = 1e-4	224.12
Process noise = 1e-7, Measurement noise = 1e-2	220.83
Process noise = 1e-7, Measurement noise = 1e-7	223.93

Table 8.7.1: Table showing the polymerisation time for different tuning in the EKF with a mismatch between the plant model and the state estimation model in the correction factor for the heat transfer coefficient, KK_Uj.

8.7.2 Mismatch in Two Parameters: KK_U_j and KK_P

In this section there is a mismatch in two parameters between the model used for the estimates and for the model representing the plant. KK_U_j is still chosen as the uncertain parameter for the Kalman Filter to adjust, and the measurement of the reactor temperature is still used for the comparisons between the model and the plant. This means that there is only one parameter to correct the model difference, when there are two parameters that are different. The parameters that are different in the model are chosen to be the correction factor for overall propagation, KK_P, and the correction factor for heat transfer, KK_U_j. The parameters have been changed so that they counteract each other, one leading to better heat transfer and one to worse. In the plant model the parameters are their original values, $KK\_U_j = 1$ and $KK\_P = 0.7$, while in the estimator model they are changed to $KK\_U_j = 1.5$ and $KK\_P = 0.6$.

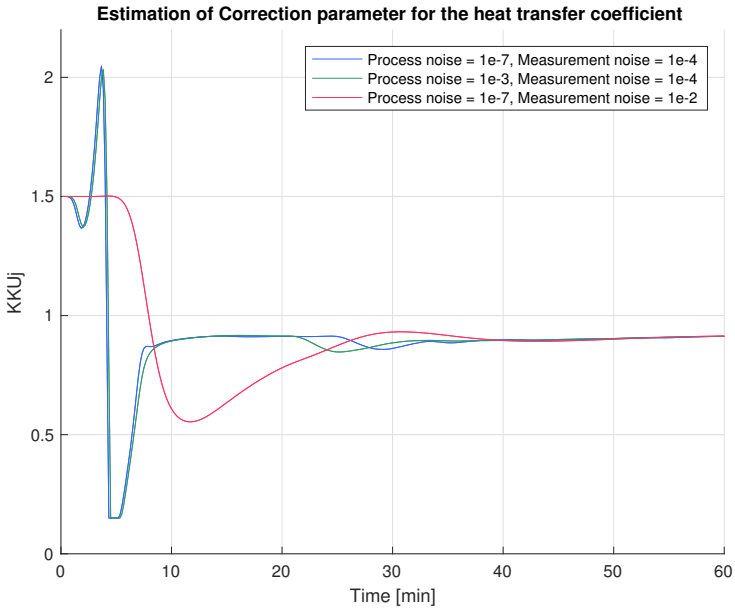

Figure 8.7.3: Plot of the development in KK_Uj different EKF tuning with model mismatch in two parameters.

Different tuning to the measurement noise and the process noise are chosen, while the white noise remains constant at $1e-4$. The simulation of the development in KK_U_j can be observed in Fig. 8.7.3. There is almost no difference in the development for different tuning on the process noise, supporting the result from the previous section. When the measurement noise is increased, the EKF takes longer to stabilise the parameter. Since there is a mismatch in two parameters, there is a more oscillating behaviour in the estimation of the parameter compared to in the previous case, in Fig. 8.7.1. The simulations

of the other process variables are plotted in Fig. 8.7.4. The slow re-estimation of KK_P when high measurement noise is considered results in a longer polymerisation time. This also results in a slight oscillating behaviour in the feed flow rate at the start of the batch. Even with the mismatch in the model, the constraints are not violated, and the polymerisation times correspond with the polymerisation times without the mismatch. The exact polymerisation times are written in Table 8.7.2.

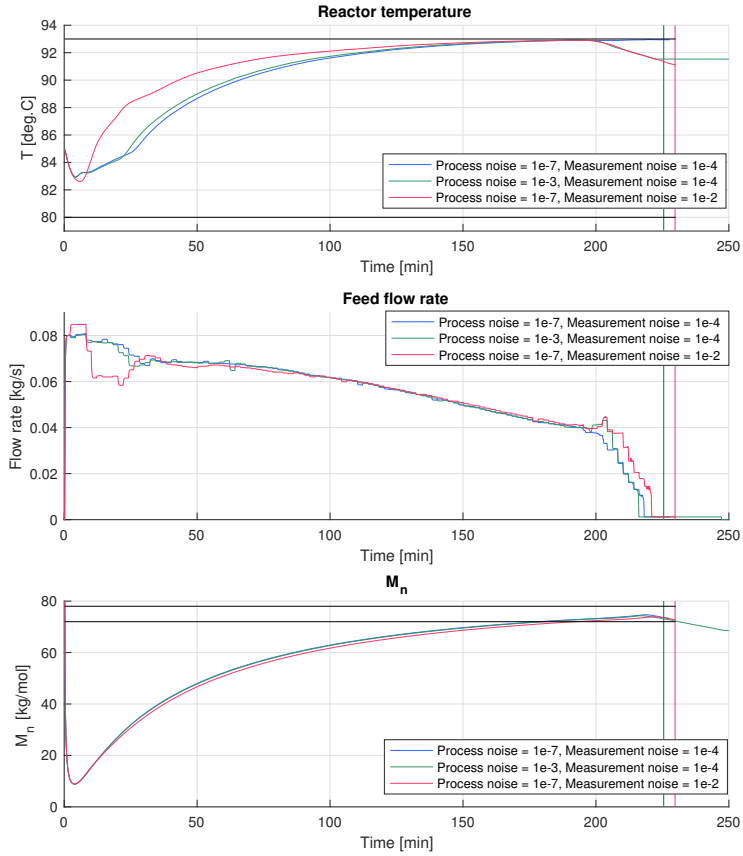

Figure 8.7.4: Plot of the reactor temperature, feed flow rate and development of M_n for different EKF tuning with model mismatch in two parameters.

Tuning	Polymerisation time [min]
Process noise = 1e-7, Measurement noise = 1e-4	225.43
Process noise = 1e-3, Measurement noise = 1e-4	225.51
Process noise = 1e-7, Measurement noise = 1e-2	229.77

Table 8.7.2: Table showing the polymerisation time for different tuning in the EKF with a mismatch between the plant model and the state estimation model in the correction factor for the heat transfer coefficient, KK_U_j, and the correction factor for all propagation, KK_P .

8.8 Results and Discussion

This section will summarise the results that have been presented in this chapter. It has been shown that the changing the weights in the objective function in both the MPC and RTO layer as well as implementing logic in the code so that these weights may change in the time horizon will influence the solution of the optimisation problem. The main interest lies in whether bilevel control of the process will result in a shorter polymerisation time or not. The average original polymerisation time is 195.03 minutes. The preliminary project optimised the process off-line, proving that the polymerisation time could be reduced to 167.25 minutes under the original conditions (allowing a slightly higher maximum feed stream). The interest in keeping a maximum feed flow rate is to keep the process under starved conditions.

Fig. 8.5.1 shows that with the implementation of a two-layer control system, the polymerisation time can be further reduced to 163.17 min. These times are for a perfect model, meaning that the model used for the simulation of the plant measurements is identical to the estimated states in CENIT. However, the main work for this master's thesis has been done for an "industrial" sized reactor, with a reduced heat transfer coefficient. Under these conditions the previously calculated polymerisation times are not comparable to the new with a tweaked heat transfer coefficient. In Figs. 8.6.3 and 8.6.4 the recipe rate is applied to the process with a reduced heat transfer coefficient, and the cooling constraint is significantly violated.

In order to compare the results from the simulations with a reduced heat transfer coefficient ($UA/4$) a base case was created. Figs. 8.8.1 and 8.8.2 show the simulated variables for different constant feed rates, the original *RecipeRate*, *RecipeRate*0.8* and *RecipeRate*0.5*. The vertical lines represent the polymerisation time. The green line is the simulation for the pilot plant, where there is sufficient cooling capacity. This is presented in the first plot in Fig. 8.8.2, where the cooling capacity constraint is never violated. In fact the variable is considerably higher than zero, indicating that there is plenty of cooling capacity left.

The red and blue lines are simulations under the new conditions with reduced heat transfer coefficient, and also with the cooling effect from the feed stream removed from the cooling capacity constraint variable. The aim is to find a constant recipe rate that is comparable to the recipe rate used with sufficient heat transfer. In both cases, for *RecipeRate*0.8* and *RecipeRate*0.5*, the cooling constraint is slightly violated at the end of the batch, where the feed stream suddenly stops. Furthermore, the cooling capacity constraint variables are not

nearly as large as for the original case, meaning that the system has less cooling capacity available. However, for both cases the reactor temperature lies within its constraints, and the end-product quality is acceptable.

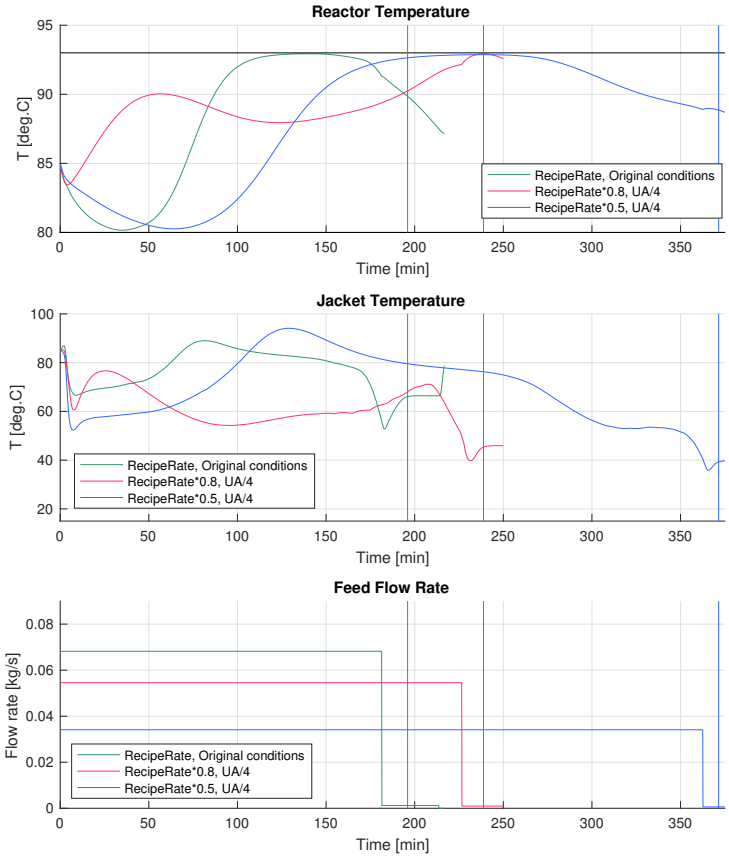

Figure 8.8.1: Plot of the reactor temperature, jacket temperature and the feed flow rate for different constant feed flow rates.

It is interesting to observe the interaction between the reactor and the jacket temperature. Since the feed flow rate is constant, the only degree of freedom is the reactor temperature which is determined in the RTO layer and the jacket temperature which is determined in the MPC layer to be able to follow the given reactor temperature trajectory. For all three cases the jacket temperature decreases abruptly at the end of the feed stream, to compensate for the loss of cooling effect. For the red line, the jacket temperature lies mainly below the reactor temperature, always providing a cooling effect. For the blue

line, however, during the batch the heat of reaction created is not sufficient to increase the reactor temperature, and so the jacket temperature must step in to heat the reactor. In many industrial applications the jacket is not able to heat the reactor, and this is added as a safety measure to avoid thermal runaway. Nevertheless there is no question that a high reactor temperature is desirable for a fast reaction to reach the conversion goal of 99 % of the monomer to polymer. The exact polymerisation times corresponding to the different cases are given in Table 8.8.1.

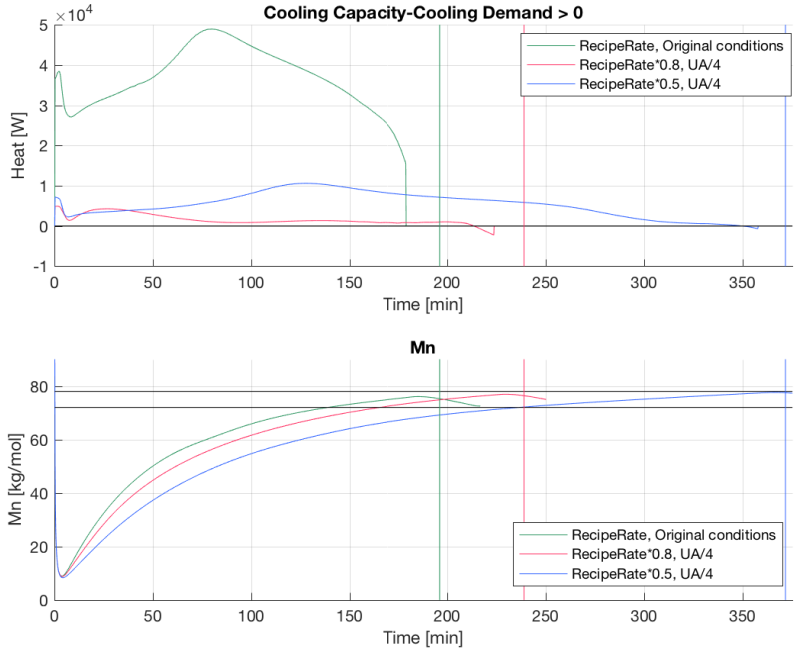

Figure 8.8.2: Plot of the cooling capacity constraint and M_n for different constant feed flow rates.

Weight	Polymerisation time [min]
RecipeRate, UA	196.00
*RecipeRate*0.8*, $UA/4$	238.80
*RecipeRate*0.5*, $UA/4$	371.50

Table 8.8.1: Table showing the polymerisation time for the original recipe feed rate and sufficient heat transfer coefficient, and two different constant feed flow rates for a reduced heat transfer coefficient.

The shortest polymerisation time obtained for the different simulations studied in this thesis was 223.67 min (in Section 8.6.2), a reduction of 15.13 min in polymerisation time for the base case of *RecipeRate*0.8*, a decrease of 6.3 %, and a reduction of 147.83 min

compared to *RecipeRate*0.5*, a decrease of 39.8 %. Even though these base cases are constructed, it can be assumed that the polymerisation time can be decreased when applying on-line optimisation on the process.

This chapter studied the solutions to different conditions in the optimisation problem. The challenges lie in the mismatch between the layers, where the complete jacket and reactor temperature is not a part of the RTO layer. Furthermore, there is a difference in the constraints in the two layers, for example regarding the constraints on M_n which are not a part of the MPC layer. The trajectories from the RTO layer have to be followed, if not the final value of M_n risks to violate its constraints. Tuning the optimisation problem is important to achieve satisfactory behaviour, and also affects the polymerisation time. In the RTO layer, harder constraints on the *PolymerisationTime* variable will actually reduce the polymerisation time. In the MPC layer, most effect on the polymerisation time is seen when tuning the penalty on the feed stream input moves and the weigth on the reactor temperature trajectory.

The RTO layer has a longer prediction horizon that sees the end of the batch and creates trajectories to minimise the polymerisation time while staying within the constraints on the end-product quality M_n and the reactor temperature constraints. The MPC layer does not see the whole picture, but has a higher sampling frequency to take care of disturbances and model errors, and works to follow the given trajectories.

The main benefit of on-line optimisation with NMPC, and thus the reason for the shorter polymerisation time, is allowing the variables to lie close to their constraints, a result of the so-called “squeeze-and-shift” rule. This can be observed because the optimisation allows the cooling capacity constraint to lie right up against the minimum value. By re-optimising throughout the batch, any disturbances and mismatches between the model and the plant will be taken care of through the EKF.

CHAPTER

NINE

CONCLUSION AND FURTHER WORK

9.1 Conclusion

The work on this master's thesis has entailed extensive literature research to understand the concepts and mechanisms that lie behind the emulsion copolymerisation process. Further research was done on optimisation problems and the solving of real-time nonlinear MPC problems. A two-level NMPC control system of the process was implemented in Cybernetica CENIT. Simulations were completed under different operating conditions and different tuning in the two layers. The aim was to understand how the two layers could work together with the upper RTO layer providing reference trajectories of the reactor temperature and feed stream for the lower MPC layer to follow. The sensitivity to different weights on certain parameters and the polymerisation time was studied.

9.1.1 Reducing the Polymerisation Time

The primary goal of implementing a two-level control structure is to reduce the polymerisation time of this semi-batch emulsion copolymerisation process. Previous experiments of a one-layer control system has been realised on a pilot plant. For a smaller sized reactor, the cooling capacity from the jacket is sufficient to remove the excess heat from the reaction. This is mainly due to the surface to volume ratio that is sufficiently big. To provoke a different solution to the problem, the heat transfer coefficient was decreased to a convenient value. The original polymerisation time from the pilot plant is 195 min. Under the same conditions, the two-layer MPC control system gives 165 minutes. The average polymerisation time for the pilot plant experiments were 195.03 minutes, and simulations with the pilot plant recipe rate the same conditions gave 196 minutes in polymerisation time. However, maximising the feed stream to achieve the shortest polymerisation time does not

make a very interesting optimisation problem. In industrial reactors the surface to volume ratio is a lot smaller, so that heat transfer becomes a limiting factor. In order to determine if the two-level optimisation structure has succeeded in reducing the polymerisation time, a base case was constructed. The solution to a new constant recipe rate that corresponded to the new conditions with a reduced heat transfer coefficient was simulated.

A great amount of time was spent studying the effect of different tuning. Since the two-layer approach is new to this process, and to Cybernetica's software in general, it was important to see how the two layers communicated. Even working with a perfect model, that has no mismatch between the plant model and the model used for estimation of the states and controlled variables, the MPC had problems following the different reference trajectories. Also, the initial prediction, at least for the feed stream trajectories, were rarely followed perfectly. These differences in the layers are due to the variations in the models used in the RTO and MPC layer. The two CENIT cores solve two different optimisation problems. They have different prediction horizons, different objective functions and different MVs. Furthermore, the temperature equations for both the reactor and the jacket temperatures are different. All these differences contribute to the mismatch between the layers and the need to tune them correctly.

Weighting the polymerisation time in the RTO layer will shorten the polymerisation time, because a higher sensitivity in the polymerisation time can be detected from changes in the feed stream and temperature. For the MPC layer, the shortest polymerisation time was provided by equal weights on following the temperature trajectory and the feed stream trajectory. A reason for this could be that when the weight is equal the optimisation problem can decide which one to diverge should it be necessary. If the temperature profile is weighted harder, the feed stream will have to do the adjustments, and vice versa. The feed rate profile proved to be determining for the success of the simulation. Especially towards the end of the batch it was critical that the feed stream trajectory was followed to ensure the correct end-product quality.

As stated several times throughout the thesis, a model is never a perfect representation of the plant. Small differences between the estimation model and the plant model were applied, and the Extended Kalman Filter activated. The aim was to study how quickly the EKF was able to correct the differences and if the polymerisation time still was reduced and constraints held. For the small changes made, the result was successful. If tuned correctly, implementing a two-layer control structure of the process can to be advantageous even with the amount of work required to write the process model and tune the layers.

9.1.2 Computation Time

The main advantage with the implementation of two-level MPC is the reduction in computation time. It allows for more computationally heavy problem in the upper layer and a faster, easier problem underneath. Applying the correct constraints and defining the correct variables are essential to the success of the control scheme. From the base case it can be deduced that there will be a reduction in the polymerisation time, because applying NMPC to a process allows it to work closer to the constraints. However, tight control

requires the calculation of many variables at frequent sampling intervals.

For the model used in this thesis the maximum computation time for the RTO layer was 90 seconds, well below the 2 minute sampling time. For the MPC layer the maximum computation time was around 7-8 seconds, with an average computation time of 2-3 seconds. Compared to a one-layer control scheme with a sampling time of 10 seconds, the average was 8-9 seconds, with the computation time sometimes exceeding the sampling time.

9.2 Further Work

The next step of this project is to implement this two-level control system to the real plant to observe how the parameters fit. Initially it could suffice with a pilot plant, but a real size experiment should be carried out to examine the behaviour when the heat transfer is not sufficiently big to allow maximum feed stream. This would allow for new parameter estimations and model validation. Further work can also be done in the model. M_n is not the only property that determines the end-product quality. M_w should be added to the model as well to determine the broadness of the molecular weight distribution. Also, the cooling effect of the feed should be studied more closely, and maybe a time constant should be introduced as it has been on the effect of the jacket temperature.

In this study, the upper layer did not include the complete energy balance to calculate the reactor temperature and the jacket temperature. This was done to simplify the calculations in this upper layer. The average computation time per sample was around 12 seconds, far below the maximum of 2 minutes. However, this simplification in the prediction of reactor temperature resulted in some difficulties regarding infeasible trajectories. An idea would be to test out the complete temperature balance in the RTO layer as well to give more accurate trajectories. Since the RTO layer is only predicted every 2 minutes, there is time for heavier computations.

BIBLIOGRAPHY

Alhamad, B., Romagnoli, J., Gomes, V., 2005a. Advanced modelling and optimal operating strategy in emulsion copolymerization: Application to styrene/mma system. Chemical Engineering Science 60 (10), 2795–2813.

Alhamad, B., Romagnoli, J. A., Gomes, V. G., 2005b. On-line multi-variable predictive control of molar mass and particle size distributions in free-radical emulsion copolymerization. Chemical Engineering Science 60 (23), 6596–6606.

Asua, J., 2008. Polymer reaction engineering. John Wiley & Sons.

Barton, K., Rogers, R., 1997. Chemical reaction hazards. Gulf Professional Publishing.

Bausa, J., 2007. Model based operation of polymer processes–what has to be done? In: Macromolecular symposia. Vol. 259. Wiley Online Library, pp. 42–52.

Boggs, P. T., Tolle, J. W., 1995. Sequential quadratic programming. Acta numerica 4, 1–51.

Bradley, S., Hax, A., Magnanti, T., 1977. Applied mathematical programming.

Chern, C., 2006. Emulsion polymerization mechanisms and kinetics. Progress in polymer science 31 (5), 443–486.

Colson, B., Marcotte, P., Savard, G., 2007. An overview of bilevel optimization. Annals of operations research 153 (1), 235–256.

COOPOL, 2007. Coopol: Objectives. `http://www.coopol.eu/Objectives.html`, accessed: 2016-11-05.

Dickerson, K., 2015. Heat transfer in reactor scale-up. Ph.D. thesis, WORCESTER POLYTECHNIC INSTITUTE.

Dimarzio, E. A., Gibbs, J. H., 1963. Molecular interpretation of glass temperature depression by plasticizers. Journal of Polymer Science Part A: General Papers 1 (4), 1417–1428.

Dimitratos, J., Elicabe, G., Georgakis, C., 1994. Control of emulsion polymerization reactors. AIChE journal 40 (12), 1993–2021.

Farkas, E., Meszena, Z. G., Johnson, A. F., 2004. Molecular weight distribution design with living polymerization reactions. Industrial & engineering chemistry research 43 (23), 7356–7360.

Foss, B., Heirung, T. A. N., 2013. Merging optimization and control. Norwegian University of Science and Technology.

Gao, J., Penlidis, A., 2002. Mathematical modeling and computer simulator/database for emulsion polymerizations. Progress in polymer science 27 (3), 403–535.

Gesthuisen, R., Krämer, S., Engell, S., 2004. Hierarchical control scheme for time-optimal operation of semibatch emulsion polymerizations. Industrial & engineering chemistry research 43 (23), 7410–7427.

Hoppe, R. W., 2006. Chapter 4: Sequential quadratic programming. https://www.math.uh.edu/~rohop/fall_06/Chapter4.pdf, accessed: 2016-12-14.

Hovd, M., 2004. A brief introduction to model predictive control. http://www.itk.ntnu.no/fag/TTK4135/viktig/MPCkompendium%20HOvd.pdf, NTNU.

Kadam, J., Marquardt, W., Schlegel, M., Backx, T., Bosgra, O., Brouwer, P., Dünnebier, G., Van Hessem, D., Tiagounov, A., De Wolf, S., 2003. Towards integrated dynamic real-time optimization and control of industrial processes. Proceedings foundations of computer-aided process operations (FOCAPO2003), 593–596.

Kiparissides, C., 1996. Polymerization reactor modeling: a review of recent developments and future directions. Chemical Engineering Science 51 (10), 1637–1659.

Kjetså, E., 2016. Optimization of a semi-batch emulsion copolymerization reaction. NTNU, Department of Chemical Engineering.

Krämer, S., 2005. Heat Balance Calorimetry and Multirate State Estimation Applied to Semi-Batch Emulsion Copolymerisation to Achieve Optimal Control. Shaker.

Kumar, A., Gupta, R. K., 2003. Fundamentals of polymer engineering, revised and expanded. CRC Press.

Larsson, T., Skogestad, S., 2000. Plantwide control-a review and a new design procedure. Modeling, Identification and control 21 (4), 209.

Leiza, J. R., Asua, J. M., 1997. Feedback control of emulsion polymerization reactors. In: Polymeric Dispersions: Principles and Applications. Springer, pp. 363–378.

Leiza, J. R., Meuldijk, J., 2013. Emulsion copolymerisation, process strategies. Chemistry and Technology of Emulsion Polymerisation, Second Edition, 75–104.

Li, B.-G., Brooks, B. W., 1993. Prediction of the average number of radicals per particle for emulsion polymerization. Journal of Polymer Science Part A: Polymer Chemistry 31 (9), 2397–2402.

Lovell, P. A., El-Aasser, M. S., 1997. Emulsion polymerization and emulsion polymers. Wiley.

Manders, L. G., Meister, M., Bausa, J., Hungenberg, K.-D., 2011. Online model-based process safety concepts in polymerization. In: Macromolecular Symposia. Vol. 302. Wiley Online Library, pp. 289–296.

Mintz, Y., Cabrera, J. A., Pedrasa, J. R., Aswani, A., 2016. Bilevel model predictive control. arXiv preprint arXiv:1611.04477.

Naik, N., Gemson, R., Ananthasayanam, M., et al., 2015. Introduction to the kalman filter and tuning its statistics for near optimal estimates and cramer rao bound. arXiv preprint arXiv:1503.04313.

Nocedal, J., Wright, S., 2006. Numerical optimization. Springer Science & Business Media.

Nørgaard, M., Poulsen, N. K., Ravn, O., 1998. Advances in derivative-free state estimation for nonlinear systems. Tech. rep., Informatics and Mathematical Modelling, Technical University of Denmark, DTU.

Nyström, A., 2007. Modeling and simulation of a multi phase semi-batch reactor. SIMS 2007, 173.

Nørgaard, M., Poulsen, N. K., Ravn, O., 2000. New developments in state estimation for nonlinear systems. Automatica 36 (11), 1627 – 1638.
URL http://www.sciencedirect.com/science/article/pii/S0005109800000893

Oppenheim, A., 2011. Interpolation, signals and systems. https://ocw.mit.edu/resources/res-6-007-signals-and-systems-spring-2011/lecture-notes/MITRES_6_007S11_lec17.pdf, accessed: 2017-05-05.

PlasticsEurope, 1999. How plastic is made. http://www.plasticseurope.org/what-is-plastic/how-plastic-is-made.aspx, accessed: 2016-11-05.

Ponton, J. W., 1994. Degrees of freedom analysis in process control. Chemical Engineering Science 49 (13), 2089–2095.

Qin, S. J., Badgwell, T. A., 2003. A survey of industrial model predictive control technology. Control engineering practice 11 (7), 733–764.

Scattolini, R., 2009. Architectures for distributed and hierarchical model predictive control–a review. Journal of process control 19 (5), 723–731.

Schork, F., Deshpande, P., Leffew, K., 1993. Control of polymerization reactors. Marcel Dekker, Inc., New York.

Seborg, D. E., Mellichamp, D. A., Edgar, T. F., Doyle III, F. J., 2010. Process dynamics and control. John Wiley & Sons.

Statista, 2015. Global plastic production from 1950 to 2014 (in million metric tons). https://www.statista.com/statistics/282732/global-production-of-plastics-since-1950/, accessed: 2016-11-05.

Tatjewski, P., 2008. Advanced control and on-line process optimization in multilayer structures. Annual Reviews in Control 32 (1), 71–85.

Van Herk, A., Gilbert, R., 2013. Emulsion polymerisation. Chemistry and Technology of Emulsion Polymerisation, Second Edition, 43–73.

Yoon, W. J., Kim, Y. S., Kim, I. S., Choi, K. Y., 2004. Recent advances in polymer reaction engineering: modeling and control of polymer properties. Korean Journal of Chemical Engineering 21 (1), 147–167.

APPENDIX

A

DERIVATION OF EQUATIONS

A.1 Population Balances

The reactions for a free-radical polymerisation can be described by the following equations (Kiparissides, 1996):
Initiation
$I_2 \xrightarrow{k_d} 2I*$
Chain Initiation Reaction:
$I* + M_j \xrightarrow{k_i} P_{j,1}$
Propagation Reactions:
$P_{j,m} + M_i \xrightarrow{k_{p,ij}} P_{i,m+1}$
Chain Transfer to Monomer Reactions:
$P_{j,m} + M_i \xrightarrow{k_{trm}} D_{j,q} + P_{i,1}$
Chain Transfer to Solvent (CTA reactions):
$P_{j,m} + S \xrightarrow{k_{CTA}} P_{j,q} + D_{j,m}$
Chain Transfer to Polymer:
$P_{j.m} + D_{i,q} \xrightarrow{k_{trp}} P_{i,q} + D_{j,m}$
Termination by Disproportionation:
$P_{j,m} + P_{i,n} \xrightarrow{k_{td}} D_m + D_n$
Termination by Combination:
$P_{j,m} + P_{i,n} \xrightarrow{k_{tc}} D_{m+n}$
Intramolecular Transfer:
$R_{j.m} \xrightarrow{k_{intra}} R_{j,m}$ or $R_{i,m}$

From these, the reaction rates and population balances can be derived. The population

balances keep track of the lengths of the polymerisation chains (Yoon et al., 2004).

$$\frac{d[I_2]}{dt} = -2fk_d[I_2] \tag{A.1.1a}$$

$$\frac{d[I*]}{dt} = 2fk_d[I_2] - k_i[I*][M_j] \tag{A.1.1b}$$

$$\frac{d[M_j]}{dt} = -k_i[I*][M_j] - k_p[M_j]\sum_{n=1}^{\infty}[P_n] - k_{trm}[M_j]\sum_{n=1}^{\infty}[P_n] \tag{A.1.1c}$$

$$\frac{d[P_1]}{dt} = k_i[I*][M_j] - k_p[M_j][P_1] + (k_{trm}[M] + k_{CTA}[S])\sum_{n=2}^{\infty}[P_n] - (k_{td} + k_{tc})[P_1]\sum_{n=2}^{\infty}[P_n] \tag{A.1.1d}$$

$$\frac{d[P_n]}{dt} = k_p[M]([P_{n-1}] - [P_n]) - (k_{trm}[M] + k_{CTA}[S])[P_n] - (k_{td} + k_{tc})[P_n]\sum_{n=1}^{\infty}[P_n], \qquad n \geq 2 \tag{A.1.1e}$$

$$\frac{d[D_n]}{dt} = (k_{trm}[M] + k_{CTA}[S])[P_n] + k_{td}[P_n]\sum_{n=1}^{\infty}[P_n] + \frac{1}{2}k_{tc}\sum_{m=1}^{n-1}[P_n][P_{n-m}], \qquad n \geq 2 \tag{A.1.1f}$$

A.2 Moment Equations

The MWD can be derived from the population balance equations (rate equations) that describe polymers of different chain lengths. However it is impractical to solve an enormous equation sets, and keeping track of the length of the chains at all times. Instead a statistical approach is used, where the molecular weight averages are used as a measure of the molecular weight properties. The averages are found by solving molecular weight moment equations, that are derived from the polymer population balances.

For a homopolymerisation the moment equation is as follows:

$$\lambda_k = \sum_{n=1}^{\infty} n^k[P_n] \tag{A.2.1}$$

where λ_k is the k^{th} moment, n is the number of monomer units and $[P_n]$ is the concentration of polymer with a chain length n. The 0^{th} moment represents the total concentration of polymer, while the 1^{st} moment represents the total weight of the polymer. The moment

equations are usually derived for the dead and live polymer chains.

$$\lambda_0 = \sum_{n=1}^{\infty}[P_n] \tag{A.2.2a}$$

$$\lambda_0 = [P_1] + \sum_{n=2}^{\infty}[P_n] \tag{A.2.2b}$$

$$\frac{d\lambda_0}{dt} = \frac{d[P_1]}{dt} + \sum_{n=2}^{\infty}\frac{d[P_n]}{dt} \tag{A.2.2c}$$

The population balance equations for P_1 and P_n are given in Eqs. A.1.1d and A.1.1e, and are inserted in Eq. A.2.2c.

$$\begin{aligned}\frac{d\lambda_0}{dt} &= k_i[I*][M_j] - k_p[M_j][P_1] + (k_{trm}[M] + k_{ts}[S])\sum_{n=2}^{\infty}[P_n] \\ &- (k_{td} + k_{tc})[P_1]\sum_{n=2}^{\infty}[P_n] + \sum_{n=2}^{\infty}\{k_p[M]([P_{n-1}] - [P_n]) \\ &- (k_{trm}[M] + k_{ts}[S])[P_n] - (k_{td} + k_{tc})[P_n]\sum_{n=1}^{\infty}[P_n]\}\end{aligned} \tag{A.2.3a}$$

A.3 Energy Balances

The general statement for conservation of energy can be written as follows:

{Rate of energy accumulation} =	{Rate of energy entering the system by inflow}	- {Rate of energy entering the system by outflow}
	+ {Rate of heat added to the system}	+ {Rate of work done to the system}

(A.3.1)

Eq. A.3.1 can be written with defined variables.

$$\frac{dE}{dt} = \dot{m}_0\hat{E}_0 - \dot{m}_1\hat{E}_1 + \dot{Q} + \dot{W} \tag{A.3.2}$$

where the dots indicate mass per time, and the hat indicates an energy per unit mass. The unit is kJ/s. $\dot{W}$ is the work term and can be split into three parts: $\dot{W}_f$, the work that is done by the flow streams moving material into the reactor, $\dot{W}_s$, the work done by the stirrer and $\dot{W}_b$, the work done when moving the system boundary.

$$\dot{W} = \dot{W}_f + \dot{W}_s + \dot{W}_b \tag{A.3.3}$$

The work provided by the feed, $\dot{W}_f$ can be expressed by the volumetric flow rate over the area the feed is inserted to and the pressure at that point. Since this is a semi-batch reactor,

there is no outflow.

$$\dot{W}_f = \dot{v}_0 A_0 P_0 = \dot{m}_0 \frac{P_0}{\rho_0} \tag{A.3.4}$$

The energy terms are composed of many forms of energy, internal, kinetic and potential energy.

$$\hat{E} = \hat{U} + \hat{K} + \hat{\Phi} \tag{A.3.5}$$

Enthalpy is commonly used to describe the amount of a system. The definition of enthalpy is given in Eq. A.3.6.

$$H = U + PV \tag{A.3.6}$$

Eq. A.3.2 can be rewritten to the following equation:

$$\frac{d}{dt}(U + K + \Phi) = \dot{m}_0(\hat{H} + \hat{K} + \hat{\Phi})_0 + \dot{Q} + \dot{W}_s \tag{A.3.7}$$

The kinetic and potential energy is negligible, and using Eq. A.3.6 to find an expression for U:

$$\frac{dH}{dt} - P\frac{dV}{dt} - V\frac{dP}{dt} = \dot{m}_0\hat{H} + \dot{Q} + \dot{W}_s - P\frac{dV}{dt} \tag{A.3.8}$$

Assuming the pressure is constant:

$$\frac{dH}{dt} = \dot{m}_0\hat{H} + \dot{Q} + \dot{W}_s \tag{A.3.9}$$

The enthalpy H is a function of temperature, pressure and number of moles.

$$dH = \left(\frac{\partial H}{\partial T}\right)_{P,n_j} dT + \left(\frac{\partial H}{\partial P}\right)_{T,n_j} dP + \sum_j \left(\frac{\partial H}{\partial n_j}\right)_{P,T,n_k} dn_j \tag{A.3.10}$$

The first partial derivative is the definition of the heat capacity, $C_p = mc_p$. The second derivative equals zero, since pressure is assumed to be constant. The third partial derivative expresses the partial molar enthalpies, $\bar{H}_j$.

$$\left(\frac{\partial H}{\partial n_j}\right)_{P,T,n_k} = \bar{H}_j \tag{A.3.11}$$

Inserting Eq. A.3.10 into Eq. A.3.9, gives:

$$mc_p\frac{dT_R}{dt} + \sum_j \bar{H}_j\frac{dn_j}{dt} = \dot{m}_0\hat{H} + \dot{Q} + \dot{W}_s \tag{A.3.12}$$

$\dot{Q}$ is the heat introduced to the system, so the heat coming from the jacket surrounding the reactor, but also the loss of heat to the environment. $\dot{m}_0\hat{H}$ is the heat introduced to the reactor through the feed stream.

$$\dot{Q} = \Delta H_J + \Delta H_{loss} \tag{A.3.13a}$$

$$\dot{m}_0\hat{H} = \Delta H_f \tag{A.3.13b}$$

$$\frac{dT_R}{dt} = \frac{\Delta H_f + \Delta H_J + \Delta H_{loss} - \Delta H_R + \dot{W}_{stirr}}{mc_p} \tag{A.3.14}$$

APPENDIX

B

INITIATOR MECHANISMS

For a free-radical polymerisation to commence, there has to be initiator present that gets activated. There are two main ways for the initiator to be activated, through homolysis of redox reactions.

B.1 Homolysis

Homolysis in chemical reactions, is the dissociation of a molecule by breaking the chemical bond in such a way that both fragments contain one of the originally bonded electrons. An example is chlorine, Cl_2:

$$Cl - Cl \rightarrow 2Cl^- \tag{B.1.1}$$

The energy involved in this process is called the bond dissociation energy. This is a quite large energy, and therefore homolysis will only occur under certain circumstances:

- Ultraviolet radiations.
- Heat, also known as thermal decomposition. Certain intramolecular bonds are weak enough to split spontaneously at the addition of a small amount of heat.

B.2 Redox Reaction

Redox reactions is often the reduction of a hydrogen peroxide by iron. However, also Cr^{2+}, V^{2+}, Ti^{3+}, Co^{2+} and Cu^{2+} can be used in stead of a ferrous ion.

$$H_2O_2 + Fe^{2+} \rightarrow Fe^{3+} + OH^- + OH* \tag{B.2.1}$$