Eindhoven University of Technology

MASTER

Controlling a tubular low-density polyethylene reactor using model predictive control with multiple models

Hennissen, J.J.M.

Award date:
2002

Link to publication

Disclaimer
This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

Eindhoven University of Technology
Department of Electrical Engineering
Control Systems

TU/e technische universiteit eindhoven

Controlling a Tubular Low-Density Polyethylene Reactor using Model Predictive Control with Multiple Models

by J.J.M. Hennissen

Master of Science thesis

Project period: March 2002 – December 2002
Report number: 02A/07
Commissioned by: Prof. Dr. Ir. A.C.P.M. Backx

Supervisors: Dr. Ir. H.M. Falkus, IPCOS Technology BV, Boxtel

The Department of Electrical Engineering of the Eindhoven University of Technology accepts no responsibility for the contents of M.Sc. theses or practical training reports

Preface

This is the end report of my Master Thesis at IPCOS Technology BV. The thesis is done at the Dutch office of IPCOS in Boxtel. The main subject of this practical work is Model Predictive Control (MPC) using multiple models applied to control a Tubular Low Density Polyethylene (LDPE) reactor. This report is meant for people with basic knowledge of both conventional control theory and Model Predictive Control theory.

Readers who want to know more about the process inside of the tubular LDPE reactor are relegated to chapter 2. The information that can be found in that chapter is the of result of a literature search. The literature search report can also be found in this report in appendix A. Readers who want to know more about the identification of a model for a process, are relegated to chapter 3. Those who want to know more about the actual controlling of the process and the usage of multiple models, are relegated to chapter 4 and 5.

I would like to thank my supervisors Dr.ir. H.M. Falkus and Dr.ir. J.H.A. Ludlage for their guidance which made it possible for me to end my thesis successfully. Also I want to thank IPCOS Technology BV and its employees for the possibilities offered to me to do my master thesis. Last but not least, I want to thank everyone I forgot to mention.

Boxtel, 26th November 2002

Jeroen Hennissen

Summary

Controlling a non-linear process with a model based controller is not easy. The usage of non-linear models to control the process is difficult and sometimes even impossible (for instance, the MPC controller is not able to handle non-linear models). The usage of multiple linear models is that case a possibility to enlarge the operating range. The production of polyethylene is such a non-linear process. There are three kinds of polyethylene: High-Density, Middle-Density and Low-Density Polyethylene. Each kind is produced in different kind of reactors under different circumstances. In this Master Thesis the control of the production of Low-Density Polyethylene (LDPE) in a tubular reactor comes across.

The development of a Model Predictive Controller is done using a simulator of this extremely non-linear process. The model described in (Gup85) is implemented in a C++ program and the interface of the simulator is made in Visual Basic. The interface communicates with the INCA Environment: a commercial product of IPCOS Technology BV which is being used for the optimal control of an industrial process.

Because of the non-linearity of the process, several operating points with accompanying work ranges are introduced. These operating points are also called product grades, because at a certain work point one specific product is produced. For each operating point a model is identified. First some pretesting is done in order to learn the behaviour of the process. After pretesting, data is logged, analysed and used in *INCAModeler* to identify a FIR-model. The FIR-model can be verified using independent validation data. The identified models turn out to predict the behaviour of the process well enough to use them to control the process.

To control the process a strategy is developed. In this strategy is stated that the outputs density and meltindex, which determine the quality of the produced polymer, are the most important outputs to control. Their value should be as close as possible to the ideal value. The other outputs, like the temperature inside and outside the reactor, should always stay in a certain zone and, if possible, follow their ideal value too. Using this strategy for the controller of product grade A gives well results. The controller is able to make small steps at the ideals of the quality-determining outputs. Also the addition of noise does not tremendously influence the performance of controllers. The controller is not able to deal with periodic disturbances due to the physical delays in the process. Partly periodic disturbances at the quality outputs of the process are compensated by the controller.

Switching between grades can be done by loading a so called cluster model. A cluster model describes a bigger operating range and contains more than one product grades which equal each other concerning their behaviour (i.e. their transfers have almost the same dynamics, delays and gains). The operating range of each included grade lies in the operating Range of the cluster. It turned out that if two grades lay close to each other, a grade model is in most cases good enough to switch to another grade in the cluster. At the most, some tuning parameters of the controller should be changed to give the controller enough freedom to reach the target grade.

When two grades lay far away from each other, switching between them can not be done with one grade model. In some cases one cluster model is enough and in other cases more than one cluster model is needed. Switching is done as follows: first the cluster (containing the current grade) model

is loaded. If necessary, tuning parameters will be adjusted. The controller, using the cluster model, moves the process state to the target grade. After reaching this grade, the model and tuning parameters belonging to that grade are loaded. It is tested that this method works. It is also demonstrated that using one model is in some cases not enough.

It is recommended that further improvements of the simulator are done. Several dynamics of the process should be simulated. Also solving differential equations, which describe the process, should be improved. Further work can be done in the automatic switching of models and tuning parameters. Developing optimal trajectories for optimal switching between product grades can also be a further activity.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

Controlling a process with non-linear behaviour can be very difficult. When a controller uses a model of the process, that model can be linear or non-linear. Using a non-linear model is not possible in the current MPC software and harder to identify. Using only one linear model is in most cases not enough. A model predictive controller is a controller which uses a model to control the process. In the petrochemical industry MPC is more and more implemented. In other industries MPC is not applied that much. One of the reasons for this is the non-linear behaviour of a process in the chemical industry while a model predictive controller needs a linear model. The goal of this thesis is to research the usage and performance profit when simply using multiple linear models to control a non-linear process. The process used for this research is the production of Low Density Polyethylene.

Polyethylene is a raw material for plastics and is being produced for over 60 years now. It's structure and production is still studied with the goal to produce a better and more pure product. Roughly, there are 3 kinds of polyethylene (PE): Low Density PolyEthylene (LDPE), Medium density PolyEthylene (MDPE) and High Density PolyEthylene (HDPE). Especially the first and the last one are discussed a lot in the literature. Those two kinds are also produced the most. In 1990 approximately 25 million tonnes per year were produced; 65% of that was LDPE and the other 35% was HDPE.

In this report the production of LDPE is discussed, especially the production of LDPE under high pressure in a so called tubular (derived from the word tube) reactor. The reactor consists of a small pipe with diameter of 5 to 10 cm. The tube itself is thicker because of a thick cooling jacket to keep the temperature in the reactor at a certain value. The length of the reactor can be huge: 500 till 1500 meter! Because of this huge length, the reactor is build in a spiral form. The picture at the right shows such a reactor.

One of the aims of this master thesis is the implementation of a simulation of the tubular LDPE reactor in the programming language C++. Also the development of an user interface is done. The following aim is the implementation of the simulator with an environment called INCA. The INCA environment is a commercial product of IPCOS Technology BV and is being used for the optimal control of an industrial process. The controller, part of this environment, is a so called Model Predictive Controller.

Model Predictive Control (MPC) is used to control the polymerisation process in the tubular reactor. MPC has some advantages such as the possibility to introduce constraints. The fact that a temperature may not exceed a certain value is one constraint that must be implemented. Also the assignment of certain ranks to outputs (i.e. a specific output may be more important than some other outputs), is possible with the INCA MPC. For optimal control, a control-strategy is developed: which outputs are the most important to control and are there any constraints to take care of. Also the performance profit when using multiple models, in stead of one, is researched.

The structure of the report is as follows. In the second chapter the working of the tubular reactor will be discussed. Especially the chemical reactions and some properties of the reactor are discussed. Also a short introduction to the mathematical model (used to simulate the reactor) and the simulation environment are discussed.

The third chapter will pay attention to identification of a model of the LDPE plant. The complete identification procedure consist of several phases. Each phase will be explained and an example will be given. Also an introduction to product grades is made. A product grade can best described as a set of operating conditions and values of in- and outputs, for which a certain product is produced.

In chapter four the actual controlling of the reactor will be discussed. Not only the controller setup will come across, but also the control strategy will be explained. Also a short explanation of MPC is given. The main part of chapter four will be the performance of the implemented MPC controller. The performance of one product grade will be discussed. Several steps on the indicators of the product quality are made. Also the process is disturbed with noise, periodic disturbances and an extra temperature offset.

In the fifth chapter multi-modelling will be introduced. Because of the non-linear behaviour of the process and the fact that our controller uses a linear model, better performance can be achieved when using more than one linear model to control the process. Subjects that will be discussed are the chosen linear models, clustering of those models and the usage of multiple models. A strategy for switching between different products is discussed and a switching example is given.

Finally, in the last chapter conclusions about the implementation of a MPC controller in a tubular LDPE reactor, will be made. Also recommendations about possible improvements of the simulator, and extensions of this demonstrator will be given.

Chapter 2

The Tubular LDPE Reactor

2.1 The process

The goal of a LDPE reactor is to convert monomer into polymer. The monomer converted in our Tubular LDPE reactor is Ethylene. The chemical structure of Ethylene consist of a double electron-connection between the two carbon-atoms of the Ethylene-molecule. One of the properties of Ethylene is the fact that it is not excited about that double connection. In fact all molecules with a double electron-connection are not excited about such a connection. They rather have a single connection with other (carbon-)atoms.

A so called free-radical (i.e. very reactive molecule) likes to have a connection more. That wish follows from the fact the molecule consist of atoms with a non-full shell. Atoms with a non-full shell want to have a full shell and are therefore very reactive. The smaller the atoms, the less and the smaller the shells and therefore the atom is more reactive. In our tubular reactor, that free radical initially comes from the initiator hydrogen-peroxide (H_2O_2).

The free-radical breaks the double electron-connection in Ethylene. As a result, the Ethylene molecule becomes a free-radical. In that way, it can react with another Ethylene molecule (i.e. break open the double connection of that molecule) and build a longer molecule consisting of two Ethylene molecules. This is shown in figure 2.1.

Figure 2.1: *Conversion of Ethylene into Polyethylene*

One side of the molecule just reacted is still very reactive and can react with an another Ethylene molecule. In this way a long chain of Ethylene molecules arises. It is also possible to add co-monomers into the reactor. By doing this, side chains can be generated when the co-monomer has more carbon-atoms (e.g. use Buthylene with four C-atoms as co-monomer).

The building of a chain can stop in several ways. The first method is when a reaction occurs between the reactive part of the polymer-chain and a hydrogen-atom. The second way is when the reactive part of the chain reacts with an Ethylene molecule which gives away it's free-radical to another molecule

which did not have a free-radical part already. The third possibility that could happen is a reaction between two polymer-chains into a very long polymer-chain.

It can be noticed, very different kinds of chains can be found in a tubular LDPE reactor. It is very usual that also monomer is received at the end of the reactor. In fact, only 15 to 20 percent of the monomer fed into the reactor is converted. The composition of different chains and their lengths determine the kind of product. The longer a chain, the lower the viscosity, the stiffer the final plastic produced with this polyethylene. In table 2.1 the elementary reaction equations can be seen. Radicals are denoted with a *R* and dead (no longer containing a free-radical part) polymer-chains are denoted with a *D*.

Table 2.1: *Elementary polymerisation reactions*

1. Initiation:	$I \xrightarrow{k_d} 2R\cdot$
2. Chain initiation reactions:	$R\cdot + M_1 \xrightarrow{k_{I1}} R_1$
3. Propagation:	$R_x + M_1 \xrightarrow{k_p} R_{x+1}$
4. Termination by combination:	$R_x + R_y \xrightarrow{k_{tc}} D_{y+x}$
5. Termination by disproportionation:	$R_x + R_y \xrightarrow{k_{td}} D_y + D_x$
6. Chain transfer to monomer:	$R_x + M_1 \xrightarrow{k_{tm}} D_x + R_1$
7. Chain transfer to solvent or chain transfer agent:	$R_x + S \xrightarrow{k_{ts}} D_x + R\cdot$
8. Chain transfer to polymer (intermolecular transfer):	$R_x + D_y \xrightarrow{k_{tp}} D_x + R_y$
9. Intramolecular transfer (backbitting):	$R_x \xrightarrow{k_b} R_x$
10. Scission of radicals:	$R_x \xrightarrow{k_\beta} \mathrm{D}_{x-y}^{=} + R_y$
11. Retardation by impurities (or oxygen):	$R_x + \ldots \xrightarrow{k_r} D_x$
12. Decomposition of ethylene:	$2C_2H_4 \xrightarrow{k_{dec}} 2C + 2CH_4 + heat$
	$C_2H_4 \xrightarrow{k_{dec}} 2C + 2H_2 + heat$

R_x denotes live radicals and D_x denotes dead polymer-chains with length x.

Normally the formation of a chain takes a while. Therefore the pressure in the reactor is very high (2000 till 3000 atmospheres). Besides, the reaction is strong exothermic. Due to the produced heat the molecules move fast (i.e. they have a lot of kinetic energy). The high pressure and high temperature cause very often collapsing of the molecules in the reactor. In this way the reaction takes place very fast. As soon as all of the free radicals have reacted (a short while after the initiator is gone), the reaction stops and the temperature slowly drops.

The reason for the dropping of the temperature are the external cooling jackets. These jackets take care that the temperature in the reactor does not exceed a certain level. At 320 à 330 °C the reactor may explode due to the high pressure, so cooling is very important. Due to the construction of the reactor and its cooling jacket, only the mixture of monomer and polymer at the edge of the tube can be cooled. In the centre of the tube, the mixture has the highest temperature and flows best. The mixture at the edge is a little syrupy.

The entire reactor is divided in a number of zones. Each zone has its own function; for example a preheating zone, reaction zone and cooling zone. In the preheating zone, the feed of monomer, co-monomer and initiator is heated to a certain temperature. By doing this, the reaction in the reaction zone can start earlier. In a reaction zone new monomer, co-monomer and initiator are added to the flow. In this zone, the actual reaction takes place. After a reaction zone, another reaction zone could be placed. In that second reaction zone fresh monomer, co-monomer and initiator can be added to the flow. This is of influence on the reaction in that zone and the final end product. It is influenced by the drop of the temperature caused by the feed which is colder than the flow in the zone. Also the formed

chains can grow further, so the properties of the end-product can be changed very flexible. The above explanation becomes clearer when looking at figure 2.2.

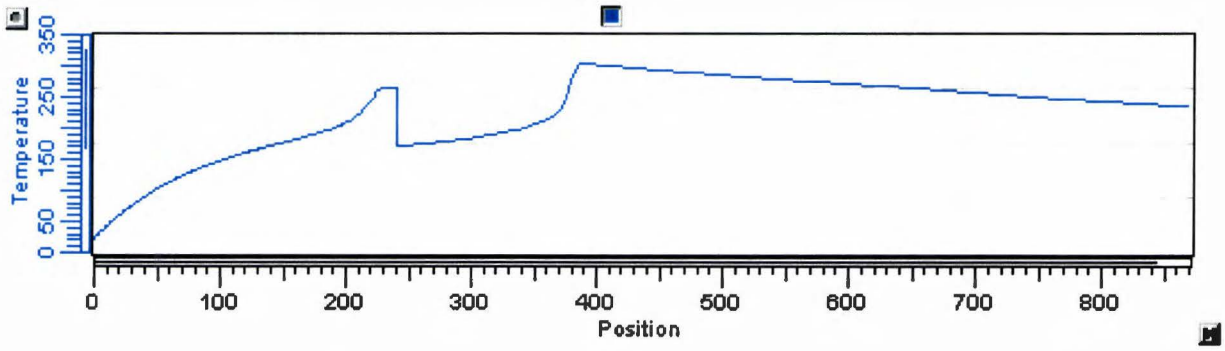

Figure 2.2: *A possible temperature profile inside the reactor*

As can be seen in the figure above, the temperature rises in the first zone due to the reaction that takes place. When all free radicals in the zone are used, the reaction stops and the temperature stabilises. When fresh monomer, etc, is added to the flow the temperature drops (this can be seen at approximately 230 meter). After that, the temperature rises again till all free radicals have reacted. When no free radicals are available anymore, the temperature slowly drops due to the cooling jackets around the reactor. In figure 2.2 this occurs at approximately 380 meter.

With simple reasoning the behaviour of the profile to changes of inputs of the reactor, can be explained. For example, when more initiator is added in zone 1 or 2, the peak in the temperature profile will move to the right. When the feed added to the flow has a higher temperature, the peaks of the profile will be higher.

More information about the tubular LDPE reactor can be found in (Kip93).

2.2 Using a Computer Model

There are many computer models of a tubular LDPE reactor introduced the past decades. There are even more articles written about those computer models. All models have their pros and cons. Because of different neglections and assumptions, using different models will lead to different simulation results. The used model comes from S.K. Gupta, et al. (Gup85) and an initial version by IPCOS using Fortran as a programming language.

The model describes a tubular reactor divided in two reaction zones and one cooling zone. At the beginning of the first and second zone (the reaction zones), fresh monomer, co-monomer (in the simulation called solvent) and initiator are added at a certain temperature. The model consists of a set of non-linear differential equations. These equations describe the behaviour of various molecular species, energy and total mass in the reactor. These equations are also coupled with other equations describing the variation of several parameters (i.e. kinetic and physical) with respect to the operating conditions in the reactor.

Using a model of a real plant will lead to simplifications and assumptions. The ones used in this model and its implementation are:

- Plug-flow within the reactor (i.e. no axial mixing)
- No backmixing within the reactor

- Ideal radial mixing
- The temperature in the cooling zone is assumed to be constant and homogeneous
- No pressure drop over the reactor

The model is implemented in *Microsoft Visual Studio C++*. Therefore some adjustments of the old *Fortran* source code were made. Not only for implementation reasons (for instance: Fortran does not have pointers which can make the program faster), but also because equivalent functions did not exist anymore. For instance: for solving the differential equations a specific *NAG (Numerical Algorithms Group)* routine for *Fortran* is used. The same NAG routines for C++ are not available anymore (in Fortran the routines were called D02PAF and D02XAF, in C++ the routines were called D02PAC and D02XAC). Therefore equivalent routines for C++ are used (D02PVC and D02PDC), which unfortunately operate different (e.g. other arguments, etc.). Some little changes in order to improve the performance and results of the simulation, such as the determination of peaks in the temperature profile, are done.

All differential equations (and a short description) can be found in (Gup85) and in the source code in the document (Hen02a) which has be written in an early stage of the thesis.

2.3 The complete Setup

What is needed when setting up a complete demonstration environment? First a simulation of the process is needed. Also a controller to control the process is needed. There should be a communication method between those two parts. Also other parts, such as a data-logging tool and a program to view values of inputs and outputs, can be useful.

IPCOS Technology developed an environment with tools that fulfill these functionalities. The environment is called INCA. The main communication protocol in INCA is OPC. All used variables are stored in a central *OPC dataserver*. The *OPC dataserver* is the core of the simulation environment.

All INCA Tools communicate with that *OPC DataServer*. A scheme of all used tools and the communication-ways can be seen below in figure 2.3.

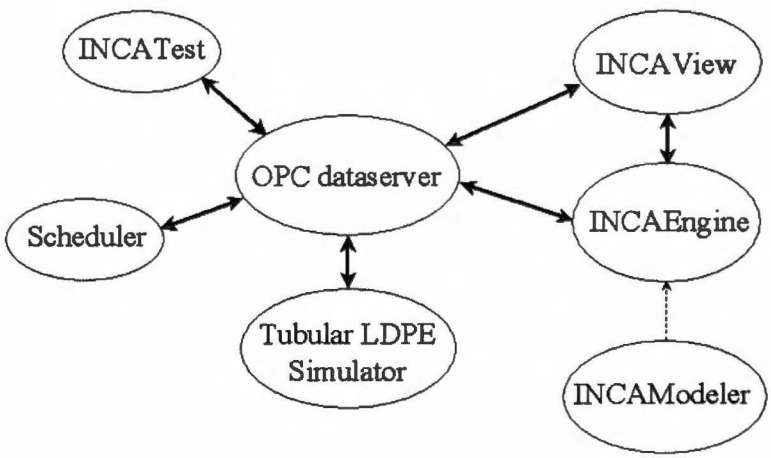

Figure 2.3: *Scheme showing communications between the used INCA tools*

INCATest is one of those programs that communicates with the dataserver. It can be used to plot the value of certain variables against time. It can also be used to log the values of one or more variables

against time (and samplenumber). *INCATest* can also be used to do automatic testing. This will be described more extensively further on in this report.

INCAEngine is the actual MPC controller which is used to control the process. In closed loop, it reads all process values and calculates new setpoints for the Manipulated Values (MV's) of the process. All values used by *INCAEngine* can be watched in *INCAView*, which is actually the user interface of the engine. Both products and its settings will be treated further on.

The last online program is the Scheduler. It schedules the modules (such as *INCATest* and the simulator) directly connected to the *OPC DataServer*.

Also one off-line INCA tool is used. That off-line tool is called *INCAModeler*. It can be used to identify a model out of data logged by *INCATest*. The program will be described more extensively further on in this report.

The programs described in this section are very simple described. More details about these programs and the INCA Environment can be read in the INCA Course and Manual (IPC01).

One of the main parts outside the INCA environment, is the simulator of the process to be controlled. The complete simulation of the tubular LDPE reactor contains more than just the C++ program code. The C++ code is compiled as *Dynamic Link Library* (DLL) which communicates with a program written in *Visual Basic*. A screenshot of this program can be seen in figure 2.4.

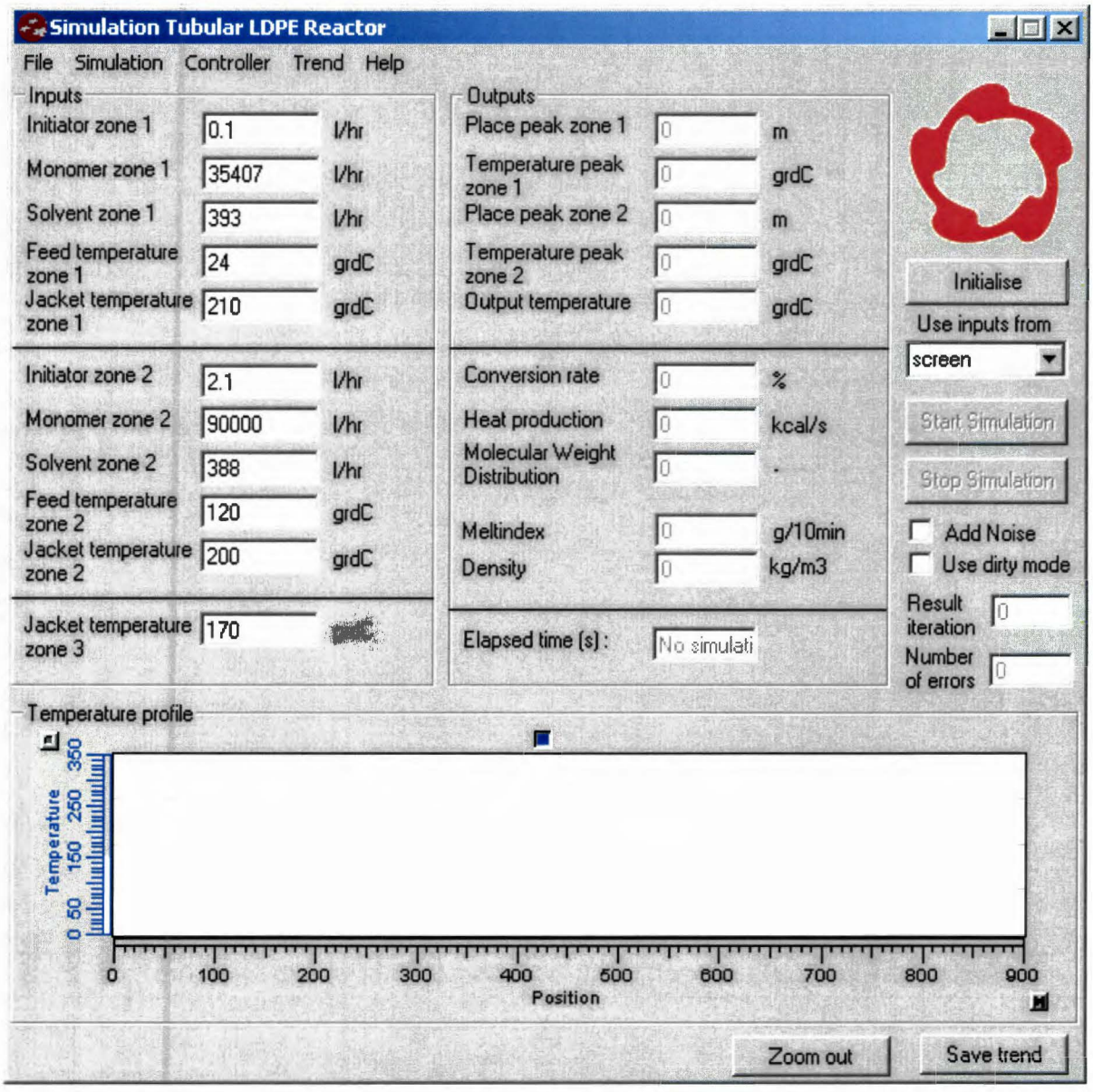

Figure 2.4: *Screenshot of simulation program*

The inputs of the reactor to be simulated can be obtained in two ways: the user may enter them into the user interface, or they are read out of the *OPC DataServer*. The simulator-program uses the input variables to calculate the output variables. While simulating, all variables (including input- and output-variables) are written into the *OPC DataServer*. In that way they are known by other programs that communicate with the dataserver. During the simulation the temperature profile of the reactor (as discussed on page 15) is plotted.

In figure 2.4 several inputs and outputs can be seen. Not all inputs are used to control the process. The feed of monomer and co-monomer (solvent) are not used to manipulate the process.
Also not all outputs are being controlled. The outputs conversion rate, heat production and molecular weight distribution are not included in the MPC controller *INCAEngine*. The outputs place and temperature of the peaks in the temperature profile are included in the Engine configuration, but they are

not all used to control the process. These variables are another way to describe the quality of the produced polymer (i.e. they are correlated to the outputs meltindex and density) and sabotage the control of the meltindex and density. The values of the temperature outputs (temperature peaks and output temperature) are not as important as the quality-determining outputs and are relative free to move, as long as they stay in a certain zone (for safety reasons).

More information about the simulator-program can be found in (Heno2b).

Chapter 3

Identification

As already told in the introduction, MPC has some advantages such as the implementation possibility of constraints. The most important part of a Model Predictive Controller is the model. When a model is not accurate enough, it is hard to control the process well (enough). So, the model should fit on the process behaviour. In practice, the model never fits the behaviour of the process exactly. Due to noise and little changes of the process equipment (e.g. a valve or a pipe), the calculated model does not fit for 100%. Normally this should not be a problem. The most important parts of a model are the described dynamics, gains and possible delays. When those parameters almost fit for all transfer functions, the controller should be able to control the process.

How do one gets such a model, and how can be verified that the model is accurate enough? The model can be calculated (or better: identified) by examining the outputs of a plant after excitation of one or more input(s). One can not simply excite all inputs of a plant and expect to identify a model out of the logged data. For instance, exciting two inputs at the same time and looking how one output changes can not lead to a model (the change of the output can be caused by both or just a single excited input) The complete identification procedure contains some phases, which should all be done with care. All those phases will be discussed in the next sections of this chapter.

3.1 Pretesting

The first phase of identification is the pretesting phase; i.e. learn how the process behaves. The goal is to determine the different dynamics (slow or fast), gains (positive or negative) and delays of each transfer ($\frac{output}{input}$). Also non-linearities are determined. This can be done by putting little steps at an input and observing the behaviour of all outputs. When the steps at a certain input differ in size, the non-linearity of the process can be determined.

The simulation of the tubular LDPE reactor is very non-linear. The gain of a transfer (including its sign) differs dependent on the process conditions (i.e. the values at the inputs of the reactor). The dynamics of a certain transfer also differ when too large steps are made at one (or more) input(s). When steps are small enough, the several gains and dynamics almost stay the same. Actually, these non-linearities are determined by pretesting and they set the size of the operating range around a certain point. That operating point belongs to a certain set of input values.

The gains of the process simulation differ very much when exciting an input too much. Therefore in the pretesting phase, all inputs were excited with very small steps (equal of size) after each other. Normally it takes a while before the steady state (belonging to that set of inputs) is found. To quicken this process, the option 'dirty mode' is introduced in the simulation. In dirty mode, no dynamics are

taken into account and the steady state is calculated immediately. It may be clear that this mode is only used in the pretesting phase. In practice the process also has dynamics that must be simulated.

The changes applied to each input are done by hand. The logging of the data (both input and output variables) is done with *INCATest*. This can be seen in figure 3.1.

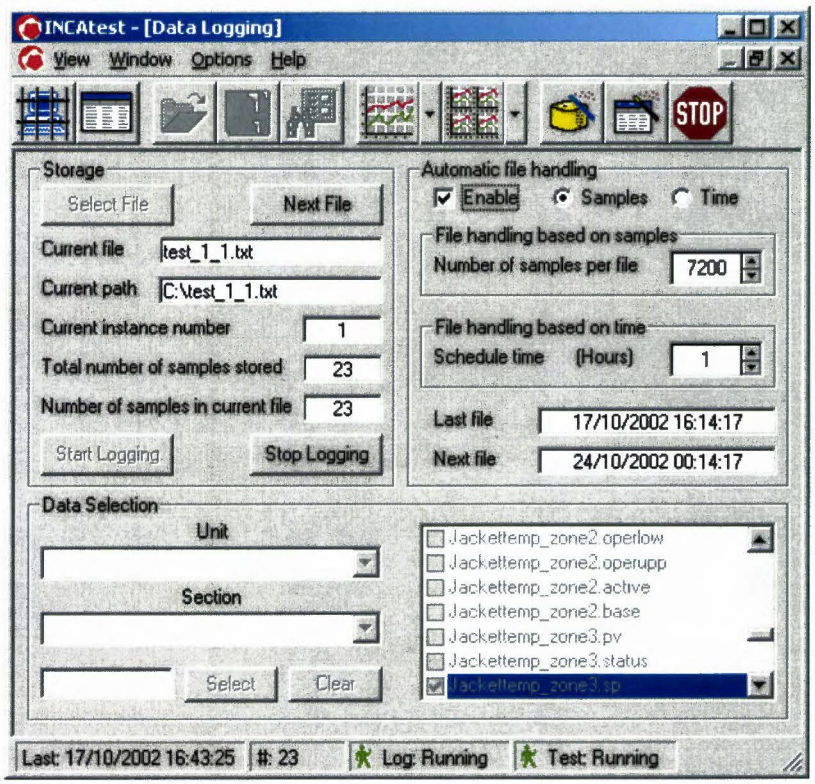

Figure 3.1: *Data on the dataserver can be logged with* INCATest

When the process is in steady state in a certain operating point, the operating range, in which the process behaves almost linear, must be determined. One aspect of linear behaviour is a constant gain for each transfer $\frac{output}{input}$ of the process.
After logging the data, the values of the outputs can be plotted against the changed input. The derivative of the line when doing this, is the gain of the transfer. So, when the line is almost straight, the gain is almost constant for those values of the changed input. This becomes clearer when looking at figure 3.2. In that figure, the temperature peak in zone 1 is plotted against the feed temperature in that zone. Approximation of the curve by a straight line can be used to determine the steady state gain.

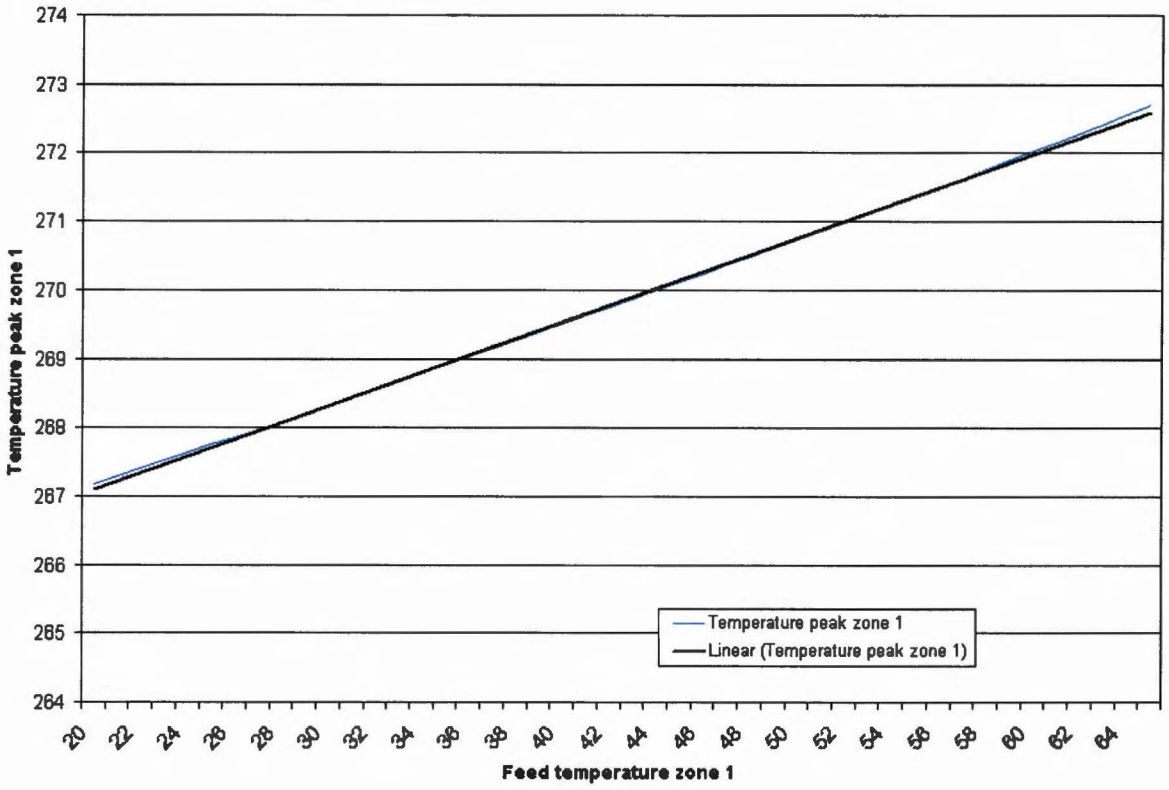

Figure 3.2: *Temperature peak zone 1 plotted against the temperature of the feed in zone 1*

Such plots are made from all outputs when changing a single input. In the same way all inputs were excited. Doing so for each operating point, an operating range in which the gains of the transfers almost stay the same, can be found. Of course dynamics should be taken into account, but other tests learned that the dynamics did not change as much as the gains did.

3.2 Several Operating Points and Areas

Because of the non-linear behaviour of the process, several operating points with accompanying ranges have been defined. These operating points and ranges are defined in table B.1 to B.8 which can be found in appendix B. The fact that these points were not random chosen but belong to several product-grades, will be explained later in this report.

Per operating point pretesting is done in order to determine the range and behaviour of the process when changing a single output. All other phases of the identification-process are done per operating point. These phases will be described in the next subchapters.

3.3 Logging Data using Test Signals

For identification, the logged data should fulfil some conditions. First, only one input may be excited (i.e. a step put on it) at the same time. When more than one input are excited, it is not guaranteed the

transfers are identified correctly. Inputs should be excited independent of each other. An input should be excited several times with several values (i.e. different step sizes should be chosen). In this way the gain of the system can be calculated more accurately and non-linearities are determined. The last condition is about the duration of an excitation. It should last until every output has reached steady state. After that, a new excitation may be started. It must be noticed that in practice more than one input is excited at the same time in order to save time. When doing tests on a real plant, the production process is in someway disturbed which should last as short as possible. When knowledge about the process behaviour (e.g. the size of the operating range in which the process behaves almost linear) is available, the tests can be done faster and the production process is disturbed as short as possible.

Every input should be excited several times and an excitation should last until every output has reached steady state. This means a complete test to identify all transfer functions in the system, can take a while. The option automatic testing in *INCATest* could therefore be an option in order to save some time. Figure 3.3 shows a screenshot of the option Automatic testing.

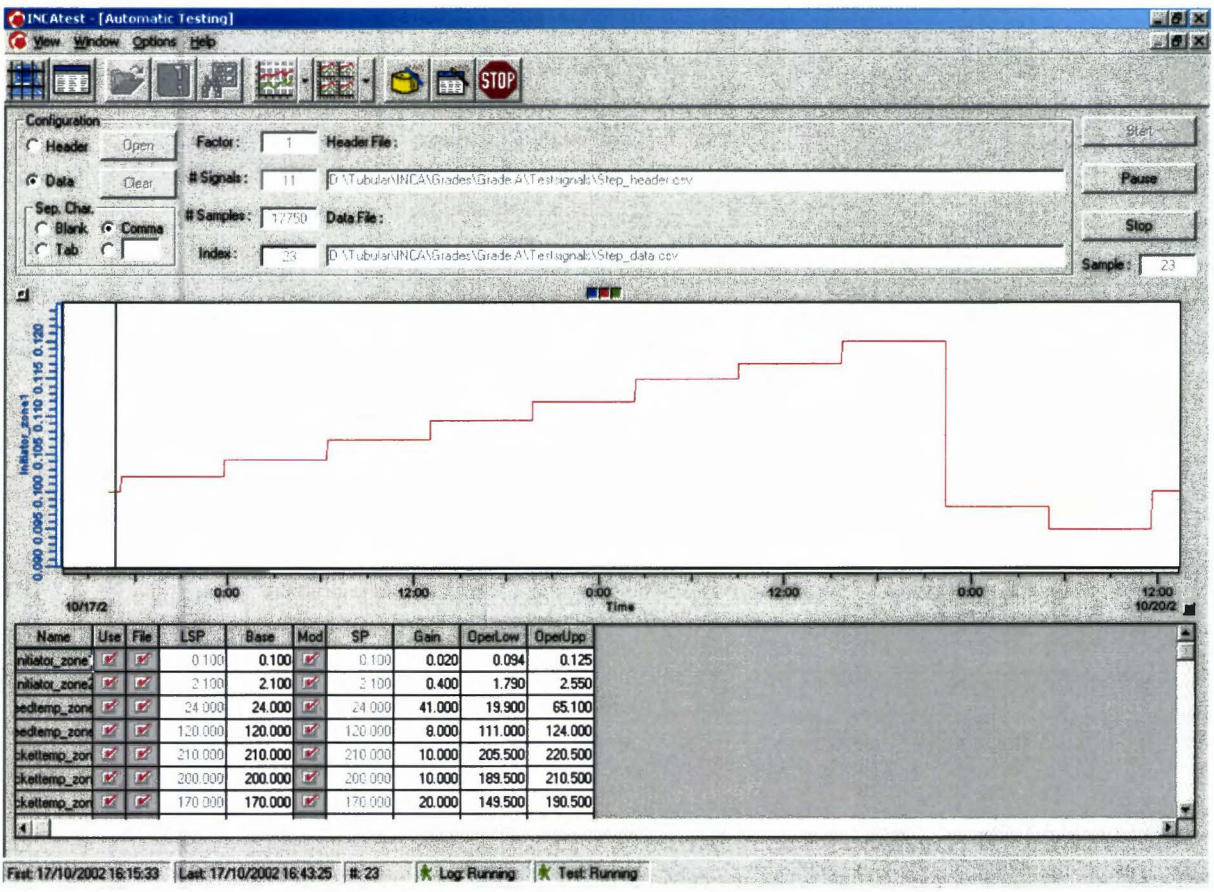

Figure 3.3: *The option Automatic Testing in INCATest*

Automatic testing uses two files: a header and a data file. The header file contains information about the duration of the test and for each input (MV's and in some cases DV's) the gain, operator and engineering boundaries. The data file contains for each input the normalised value for that sample. For each model thats needs to be identified, such a testsignal must be designed because the operating point and range differ per model.

3.4 Analysing the logged Data

After logging the data with variations due to the automatic testing, the generated file(s) should be analysed. Possible inconsistences should be fixed. A sensor could have measured a completely wrong value for one or more samples. Identification algorithms could have problems with outliers and the identified model will not be as accurate as possible. Therefore outliers should be fixed.

A program like *MS Excel* can be used to analyse the data files. The figure below is a plot of the output 'place peak zone 2' when changing inputs using a testsignal described before. It may be clear that the outlier must be wrong and should be corrected. This correction is done using peak shaving. In practice such outliers can be caused by a failing sensor. In this case, a possible explanation is wrong temporary solution (i.e. discontinuity between the solutions) of the differential equations used to simulate the reactor.

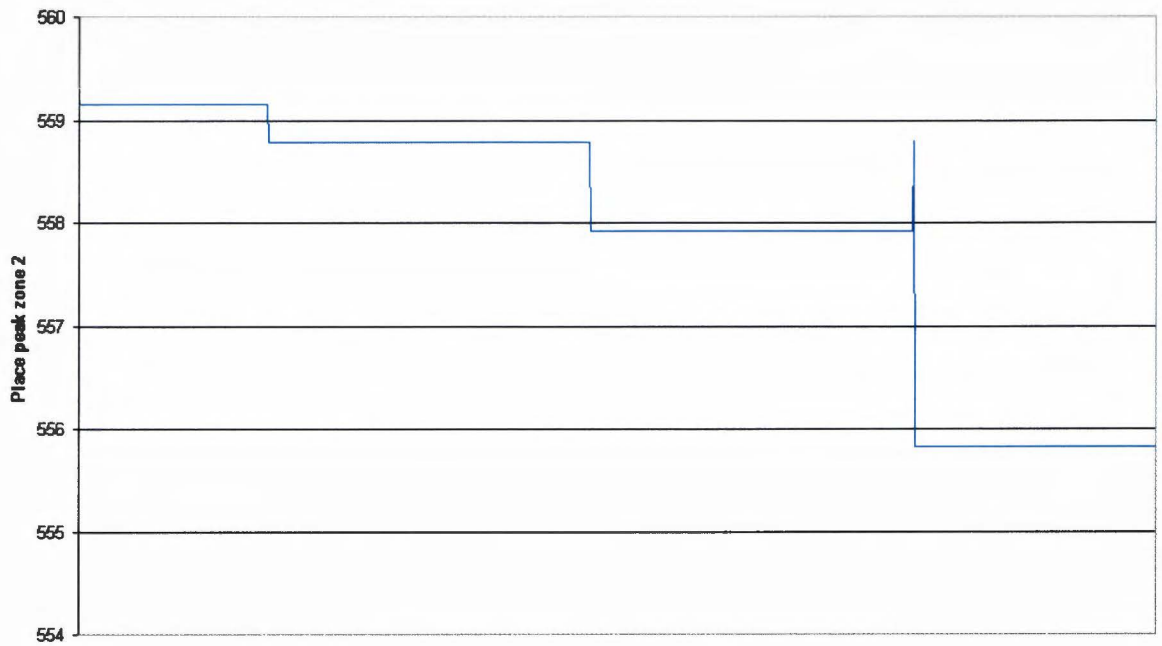

Figure 3.4: *The logged data for the place of peak in zone 2 contains a outlier*

After correcting outliers some parameters for the future identification should be determined. When calculating a FIR-model, the duration of each response and the delays before the response starts, can be used as à-priori knowledge. These parameters are determined opening the logged data-files in *MS Excel*, but also another program could be used to do so. The parameters (in number of samples) for each transfer in each grade (this will be explained further on) can be found in Appendix C on page 85.

3.5 Identify a Model using INCAModeler

As already mentioned in the previous chapter, *INCAModeler* is an offline tool part of the INCA environment. Its goal is to identify a model out of data by building so called cases. In a case, the inputs (called independents) and outputs (called dependents) are selected in the wanted sequence. There are several

kind of models that can be identified using the modeler: FIR, State-Space, output error and equation-error models. For each kind of model certain parameters, such as number of FIR parameters (in case of identifying a FIR model) and known delays and gains, can be filled in. More about *INCAModeler* can be read in its manual (IPC01). A screenshot can be seen in figure 3.5.

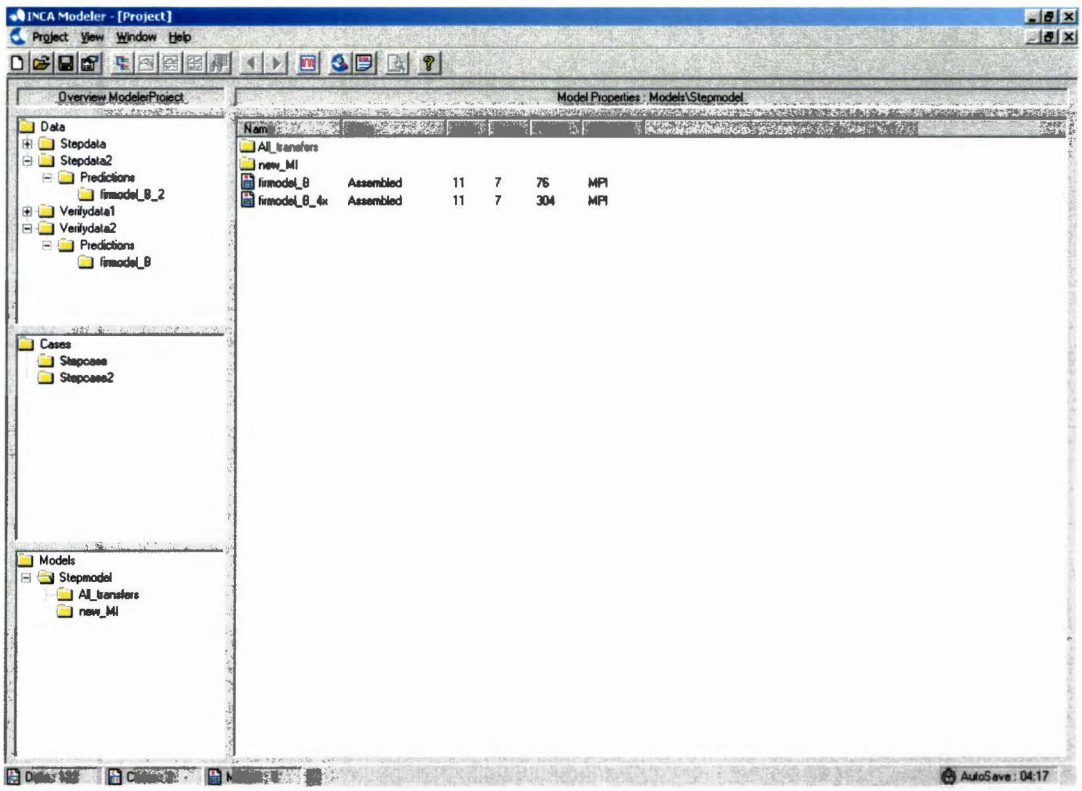

Figure 3.5: *Screenshot of INCAModeler*

After building a case, the model can be calculated and viewed. It is decided to identify a FIR model instead of a state-space model because of the advantage that comes along with a FIR model: known process-knowledge can be used to identify a better model. Known parameters as gains, delays and zero-transfers can be defined when building the case. When identifying a FIR model, the algorithm used to identify the FIR model is Least Squares. More about this identification procedure and techniques used in the INCAModeler can be found in (Fal94). The screenshot in figure 3.6 shows a part of the step responses of the identified model for grade B.

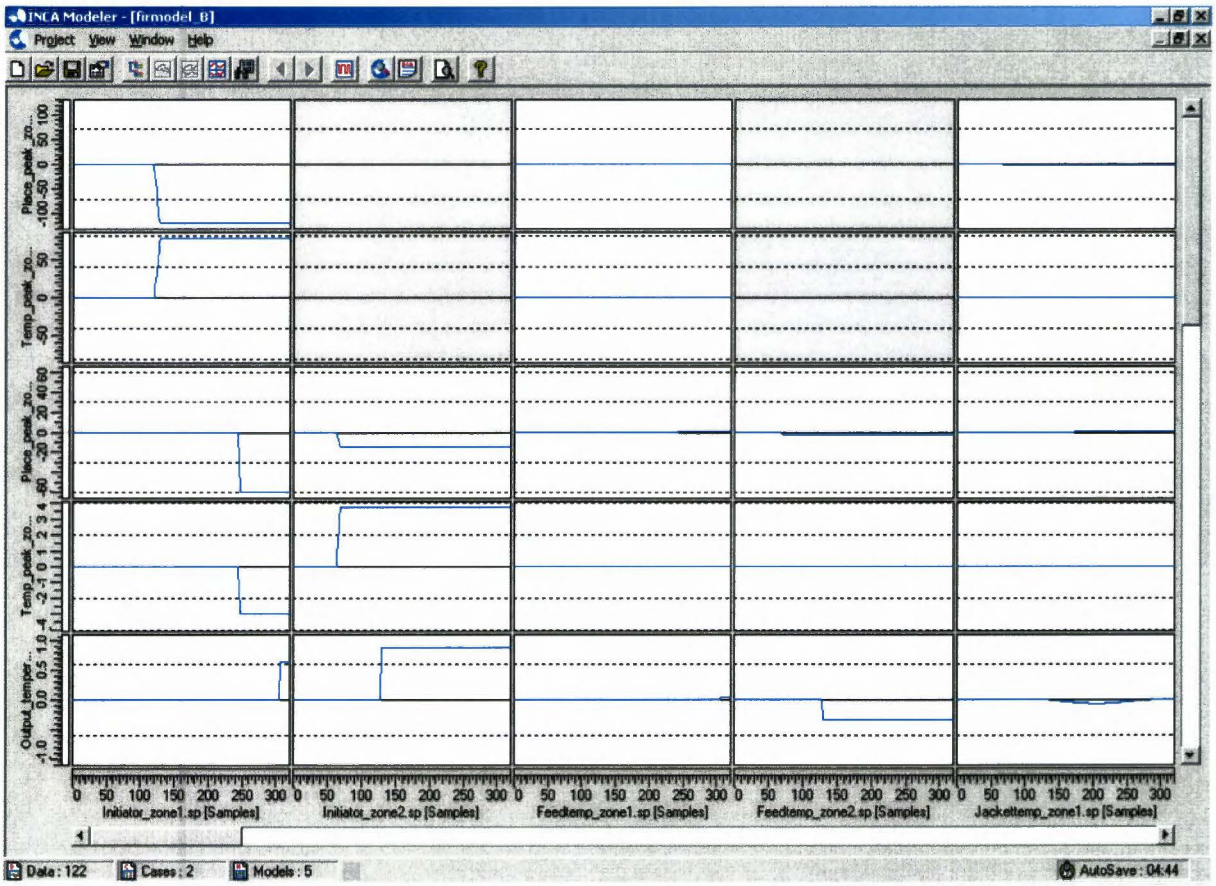

Figure 3.6: *Screenshot of plotting a model in INCAModeler*

The complete step responses of the identified model for grade B, can be found in Appendix D.

3.6 Verification of the identified Models

To determine how good the identified model actually is, the model needs to be validated. This can be done by making predictions with the found model. One can use the already imported logged data of the inputs to predict the outputs. When the model is perfect, the predicted outputs should fully fit on the logged outputs. In practice this is never the case; there are always small deviations between the predicted and measured output values. Also in this simulation small deviations occur. The reason for this is the non-linearity of the process and the fact that the identified model is linear (i.e. simplification failures appear). The screenshot in figure 3.7 shows this for the model of grade B.

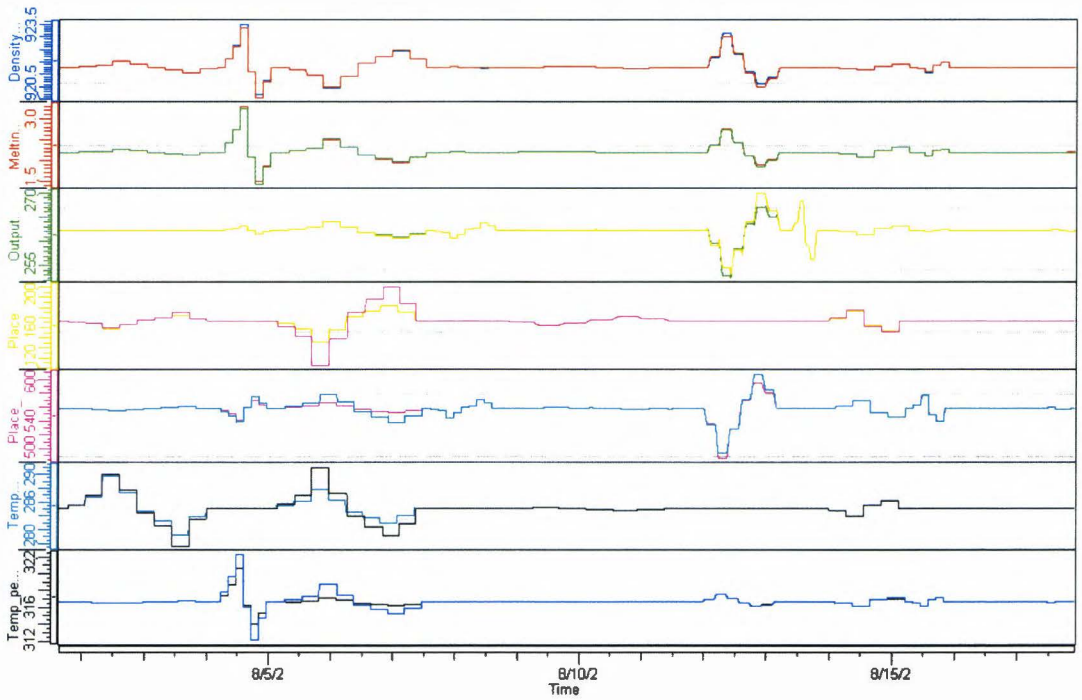

Figure 3.7: *Predicted and measured outputs using the known step data and the model of grade B for predictions*

The legend for both figure 3.7 and figure 3.9 can be found in the figure below. The values behind the description belong to figure 3.9 at the point where the predicted variable 'Temp_peak_zone2' differs the most from the measured value.

[Time]	: 8/3/2 14:11
Density.Prd	: 920.41
Meltindex.Prd	: 1.245
Output_temperature.Prd	: 272.59
Place_peak_zone1.Prd	: 164.6
Place_peak_zone2.Prd	: 617.9
Temp_peak_zone1.Prd	: 288.45
Temp_peak_zone2.Prd	: 313.2
Density.pv	: 920.14
Meltindex.pv	: 1.143
Output_temperature.pv	: 275.17
Place_peak_zone1.pv	: 167.9
Place_peak_zone2.pv	: 620.5
Temp_peak_zone1.pv	: 288.45
Temp_peak_zone2.pv	: 224.2

Figure 3.8: *Legend belonging to figure 3.7 and figure 3.9. .PRD are the predicted outputs and .PV are the measured outputs*

A better judgement of the model (in this case, the model of grade B) can be made when validation data (i.e. data which is not used to identify the model) is used. In this validation data, several inputs may be excited at the same time since in practice this will be the case. As can be seen in figure 3.9, the predicted outputs differ from the measured outputs, but the behaviour is mainly the same.

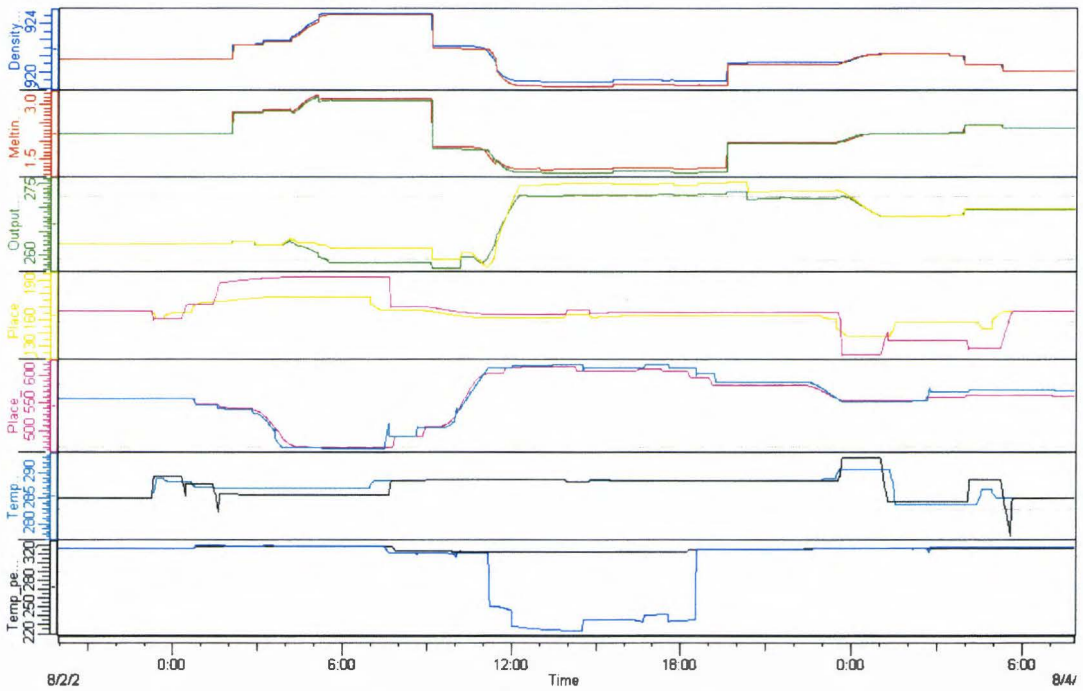

Figure 3.9: *Predicted and measured output using independent verification data*

As can be seen, the prediction differs from the measured outputs at some places. There are peaks which are not predicted. The reason for these peaks is a discontinuity in the (temporary) solutions of the differential equations describing the simulation.
Also the behaviour of the measured temperature peak in zone 2 (the bottom plot in figure 3.9) differs very much from the predicted value. The explanation for this, is a non-linearity which occurs when a certain input (or a combination of inputs) is excited too much. That non-linearity is not included in the linear FIR model used for the prediction. When the value of every input stays within the operating range, it can be concluded that excitation of more than one input at the same time makes the operating range smaller. So, it is smart to use a smaller range when tuning the controller using this model. When operating outside an operating range, the controller will still be able to control the process, but not as good as possible. Because of the small operating ranges of the models and the large range needed to reach all wanted product grades, multiple models are needed.

One last note: in this situation, the models are identified using step-responses. No noise was added to the process simulation during the model-identification phase. Therefore simple step-responses are enough to identify a model. However, in practice the process contains noise (i.e. measured process data is disturbed). In that case a PRBNS test can be very useful to reduce the disturbance influence on the test and improve the quality of the model. PBRNS testsignals have the advantage that the testing time is shorter and amplitudes can be smaller. In that way, the production process is less disturbed. Also the dynamics of the process can be better estimated.

Chapter 4

Controlling the Process

4.1 Conventional Controllers

The main target of this master thesis is to control the Tubular LDPE reactor simulation. This is done using Model Predictive Control. In the real plant, also conventional PID controllers can be found. Figure 4.1 shows the usage of such a PID controller.

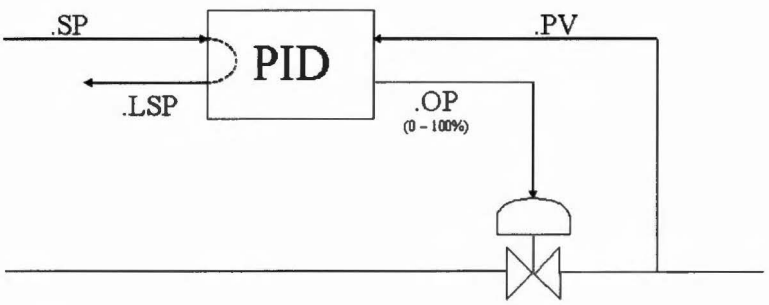

Figure 4.1: *PID Controllers are also applied in a MPC environment*

In the figure a flow (for instance: a flow of initiator) is controlled by a valve. This valve is actuated by the output of a PID controller. Of course this PID controller needs some input in order to determine how to actuate the valve. It gets a setpoint or reference value from an operator or a supervisory controller such as the MPC controller *INCAEngine*. It also gets the real process value of the realised flow from a sensor in the pipeline. The PID controllers are not replaced by the MPC because of the advantages they have. The PID controllers can act very fast on changes of, for example, a valve. When the MPC must also control these outputs, a smaller sample time is needed which could introduce numeric problems. Another reason are unstable parts in the plant. Those parts can be stabilised by conventional PID controllers. When the MPC goes offline for some reason, the process must still be stable and controllable by an operator. Since PID controllers are already present in the plant and the advantages they have, they are in most cases not replaced.
However, in this simulation environment, these controllers are not implemented outside the simulation. The configuration of these controllers is not further discussed in this report.

4.2 Implementation of a Model Predictive Controller

4.2.1 Basic MPC concepts

In this section some basic MPC concepts are discussed. A MPC uses a process model to predict the behaviour of the process. Based on that prediction, a calculation is done of the future behaviour that is as close as possible to the desired behaviour. Thus, the MPC has two main tasks: predict the future behaviour and optimise it under defined constraints. The prediction is based on the past and future inputs and the last measured outputs.

The optimisation procedure finds future inputs to bring the future outputs as close s possible to their desired values. The optimisation task consists of two subtasks: static optimisation and dynamic optimisation. First the static optimisation is done based on equation 4.1 and equation 4.2 (assuming there are p inputs and q outputs).

$$min\{(u_{ideal}(1) - u_{ss}(1))^2 + \ldots + (u_{ideal}(p) - u_{ss}(p))^2\} \tag{4.1}$$

$$min\{(y_{ideal}(1) - y_{ss}(1))^2 + \ldots + (y_{ideal}(q) - y_{ss}(q))^2\} \tag{4.2}$$

The equation must solved taking constraints and maximum steps to be made, into account. The static optimisation is solved using a QP-solver.
The dynamic optimisation determines how the inputs and outputs move in the future and if they can move more optimal (faster for example).

For simplicity, a MPC controller can be compared with a chess computer. A chess computer also calculates future moves and (depending on the play level) makes the smartest (optimal) move. There are several parameters to (fine)tune the controller. The most important parameters will come across in the next sections.

More about Model Predictive Control can be found in (Bac92).

4.2.2 Controller Setup

In chapter 2 could be read the tubular reactor is controlled with the MPC controller called *INCAEngine*. This section will describe how the controller is implemented and (fine)tuned. The method described is applicable for each operating point and its range, but only one is described. Operating point A (see table B.1) is chosen without a special reason.

The tubular LDPE simulator has 11 process-inputs and 10 process-outputs. The simulator has more inputs which can be seen as settings for the simulator. Therefore the inputs relevant for controlling the process are called control-inputs. Also not all outputs are controlled: some outputs of the simulator are indicators of the process simulation.

The *INCAEngine* knows three kinds of variables: Manipulated Variables (MV's), Disturbance Variables (DV's) and Controller Variables (CV's). Both MV's and DV's can be seen as inputs of the process with the slight difference that MV's can be manipulated by the controller (i.e. their value can be changed). The value of a DV cannot be changed by the controller and can therefore be seen as a measured disturbance on the process. CV's can be seen as outputs of the process.

The MV's of this LDPE reactor simulation are:

- Initiator Feed Zone 1
- Initiator Feed Zone 2
- Feed Temperature Zone 1

- Feed Temperature Zone 2
- Jacket Temperature Zone 1
- Jacket Temperature Zone 2
- Jacket Temperature Zone 3

The DV's of this LDPE reactor simulation are:

- Monomer Feed Zone 1
- Monomer Feed Zone 2
- Solvent (Co-monomer) Feed Zone 1
- Solvent (Co-monomer) Feed Zone 2

The CV's of this LDPE reactor simulation are:

- Place Peak Zone 1
- Peak Temperature Zone 1
- Place Peak Zone 2
- Peak Temperature Zone 2
- Output Temperature
- Density Polymer
- Meltindex Polymer

4.2.3 Control Strategy

The first thing to do when tuning a (MPC) controller is to decide how the controller should behave (i.e. a control strategy must be developed). A controller can be tuned in such a way that some overshoot (or undershoot) occurs when putting a step at an outputs ideal-value. However, there are cases in which this overshoot is unwanted because of negative consequences for the process. A controller can be tuned to follow a step very fast (with some overshoot) or it can be tuned to be slow but more accurate (i.e. performance is more important). Besides performance, the robustness of the controller can be influenced by the tuning parameters.

When the process is in a certain grade, it is decided that the controller should be very fast and accurate. This is not as easy as it looks, because robustness and performance are always each other's opposite. It should be fast because the controller should be able to suppress fluctuations due to disturbances at inputs and outputs of the process or other disturbances. Of course the controller should be as accurate as possible; when producing a product with a certain density and meltindex, the product should have as small as possible pollution of polymer with other specifications. Therefore a tradeoff between robustness and performance has to be made (i.e. an optimal compromise between fast and accurate control has to be found).

The *INCAEngine* is a MPC controller which can take constrains into account. As can be imagined a variable may not exceed a certain boundary. For instance, a temperature may not exceed a certain value because of unsafe situations that could arise. When both upper- and lower-bounds are introduced, a certain band (or zone) around the variables setpoint is created. It could be wishful that the value of the

variable never crosses a boundary and leaves that zone. Other kind of constrains are the rate of change per variable. For example: it could be unwanted that a valve is opened (or closed) too fast.

A second nice thing of the *INCAEngine* is the possibility of ranking. Each ideal value (i.e. reference value) and zone can be ranked. Each CV has an idealrank and a zonerank. MV's only have idealranks because it is decided internally in the controller that the zones of a MV have greatest rank. This means a MV's zone may never be exceeded. When choosing correct boundaries it can be prevented that the plants operating conditions drift to unsave values.

The strategy to control the process is the following. The CV's density and meltindex are the most important output variables since they determine the quality of the polymer. When the temperature inside the reactor becomes too high, the pipe could burst open. So, all temperature variables need to stay in their zone. Also it is nice to keep the temperature of the polymer that comes out of the reactor, at a certain constant value. The temperature profile inside the reactor determines the quality of product. Because the quality is also determined by the variables density and meltindex, it is impossible to keep the temperature profile unchanged. These outputs variables are too much coupled. In practice the quality of the polymer is determined by the density and meltindex of the product. Therefore, these two outputs are more important to control than the places and the height of the temperature peaks. This explanation is expressed in ranking terms in table 4.1.

Table 4.1: *Ranking of variables (the higher the rank, the lower the priority)*

Manipulated Variables	Ideal Rank	Controlled Variables	Ideal Rank	Zone Rank
Initiator Feed Zone 1	22	Place Peak Zone 1	30	5
Initiator Feed Zone 2	22	Peak Temperature Zone 1	25	5
Feed Temperature Zone 1	21	Place Peak Zone 2	30	5
Feed Temperature Zone 2	21	Peak Temperature Zone 2	25	5
Jacket Temperature Zone 1	20	Output Temperature	15	5
Jacket Temperature Zone 2	20	Density Polymer	10	5
Jacket Temperature Zone 3	20	Meltindex Polymer	10	5

The constrains are set up as follows. For each variable there are three kinds of boundaries: operator boundaries, engineering boundaries and valid boundaries. The identified model used in de MPC controller is only valid in the operating range it describes. The model may be good enough outside its range where values are still valid, but this is certainly not guaranteed. The engineering boundaries are therefore chosen equal with the boundaries of the operating range of the used model.

The operator boundaries are chosen safely chosen inside the engineering boundaries. The valid boundaries are chosen much higher (or lower in case of the lower boundary) than the engineering boundaries. In practice the values of these boundaries are the maximum or minimum value for which the sensor, valve, etc. operates as it should. A simple example: a certain feed can never be less than zero!

Two important parameters of the MPC are control horizon and prediction horizon. The control horizon is the horizon over which MV's may be moved during the dynamic optimisation. The prediction horizon is the horizon over which the CV's are predicted. In figure 4.2 these two parameters are explained in a graph.

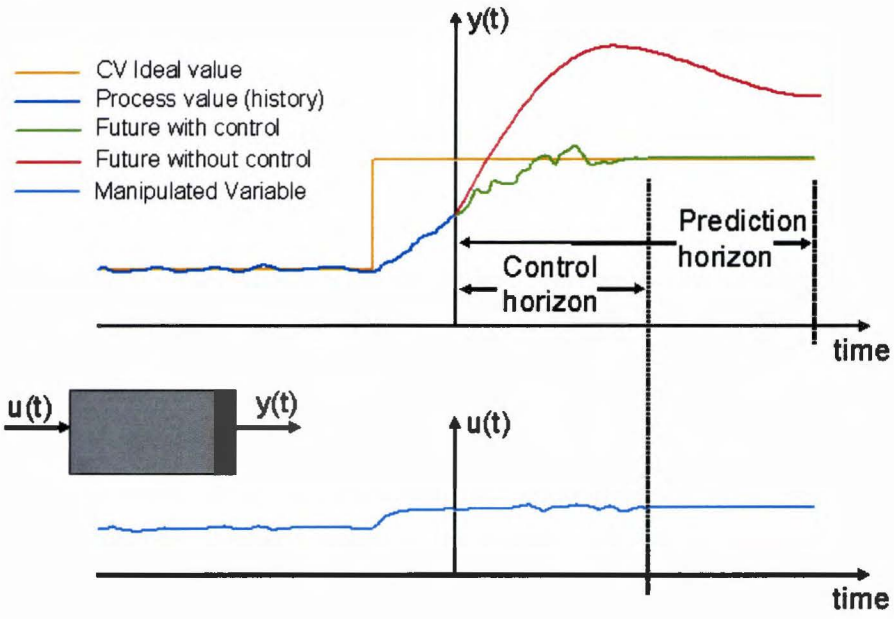

Figure 4.2: *Graphical explanation of control and prediction horizon*

There is no rule of thumb when choosing the control horizon. The larger the control horizon, the better the performance but the more complex the optimisation problem. In this simulation using process dynamics lead to a control horizon of 41 samples. When the control horizon is determined, the predict horizon can be chosen by a rule of thumb: prediction horizon = control horizon + maximum number of FIR parameters in the used model. In this case, the prediction horizon is chosen to be 120 samples.

Further tuning parameters such as maximum steps to be made per cycle, steady state weighting factors and dynamic weighting factors can be setup in order to influence the behaviour of the controller in a positive way. Most of these parameters are determined by trial and error. Fortunately there are some rules of thumb. First choose the weighting factor equal for all variables. When the predicted steady state is very important for one specific variable, the steady state weighting can be changed in such a way that reaching the steady state of the specific variable is more important than for other variables. When the dynamics of certain output should be improved, the dynamic weight parameter can be changed. In *INCAEngine* these weights are inverse weights, so a lower number means a higher weight and thus a higher importance.

4.3 Results of MPC Implementation

Eight grades are introduced. Per grade one model is identified and used to tune the MPC controller. Describing the performance and results of each controller would be too much information for this time. Therefore only one controller (of one grade) is discussed. Product grade A is chosen without further specific reasons.

4.3.1 Step at the Density's Ideal Value

The controller belonging to process grade A is now discussed. In our simulation grade A has a density of 921.12 $\left[\frac{kg}{m^3}\right]$ and a meltindex of 0.796 $\left[\frac{g}{10\ min}\right]$. The controller is tested when putting steps on both

quality variables.

Figure 4.3 shows a screenshot when putting a negative step at the ideal value of the density. The ideal value is decreased to 921 $\left[\frac{kg}{m^3}\right]$. Later on the ideal value is set back to its starting value.

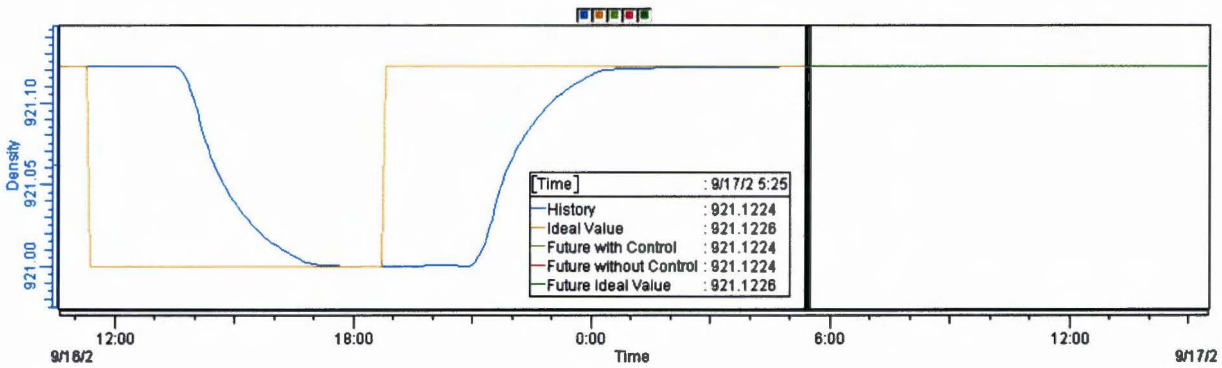

Figure 4.3: *Results of a step at the density's ideal value*

After the change of the density's ideal value, it takes a while before the process value decreases and follows the new ideal value. This is due to physical delays in the process; when a change is made at an input (for example: more initiator is fed into the reactor), it takes a while before that change is noticeable at the end of the reactor where the density and meltindex is determined. The same applies to the positive step at the density's ideal value. In figure 4.4, showing the MV 'Jacket Temperature zone 2', it becomes clear that the controller takes action direct after the change ideal value; it just takes a while before this action becomes noticeable.

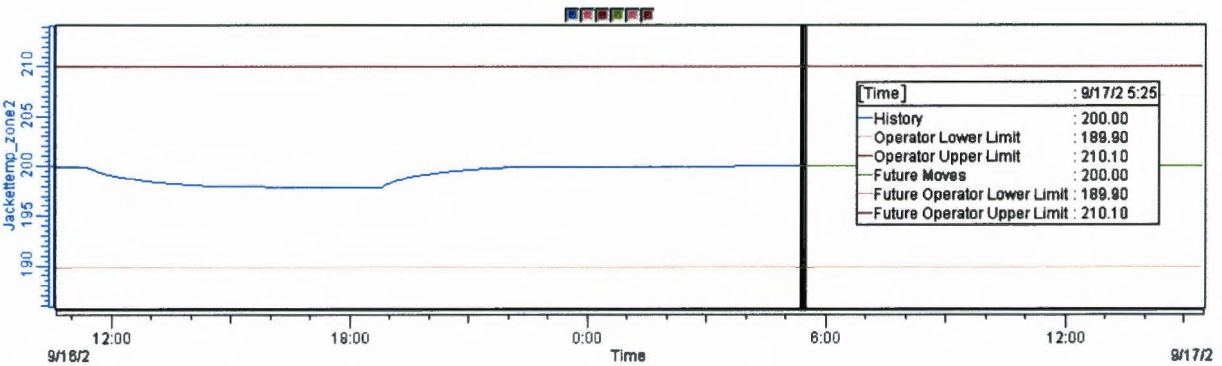

Figure 4.4: *Moves of the MV Jacket Temperature zone 2 when making steps at the density's ideal value*

It can be seen in figure 4.3 that the new ideal value of the density is reached in approximately 7 hours. Based on historical knowledge it can said that in practice this is also the case. When two grades lay close to each other (i.e. making small steps is enough), the time to switch is less than when two grades are completely different. In the last case, the switching time could be 10 to 16 hours and in the most worse case even more than 20 hours. In figure 4.3 the new value is reached without overshoot.

The change of density influences the values of other outputs. One of the most important outputs is

the meltindex, since this one is also a direct indication for the product quality. Figure 4.5 shows the change of the meltindex due to the step at the density's ideal value.

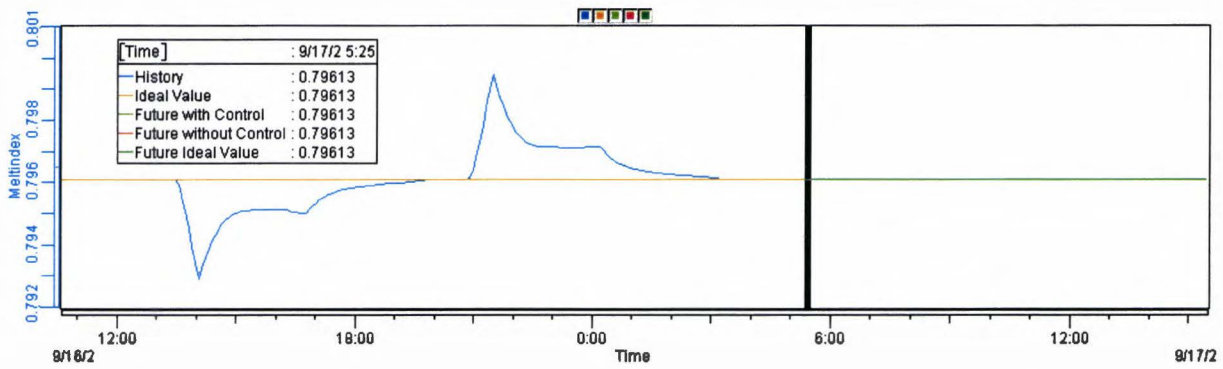

Figure 4.5: *Change of meltindex due to a step at the density's ideal value*

As can be seen in the above figure, the meltindex decreases when a negative step at the density is made. The reason for this is the fact that the meltindex and density are coupled. When one or more inputs are changed in order to correct the density's process value, it has also influence at the process value of the meltindex. The meltindex decreases very quickly but increases back to its ideal value again. The sharp peak in the meltindex' value is probably caused by sudden discontinuities in the simulation. Reaching the ideal value happens much slower because also other demands (with higher ranks) should be satisfied (i.e. the density's ideal value should be reached). Also other constraints, such as the boundaries of the MV's, limit the freedom of the controller to satisfy all demands (not occurred in this case).

4.3.2 A noisy Density Process Value

In practice the in- and outputs of the process will probably be noisy. The simulator is able to add noise (with a certain amplitude) to the in- and outputs of the process. That noise can be integrated so that its pseudo random changes are not as wild as normal. Also can be chosen to filter (using a low-pass filter) the outputs with their noise. In order to test the controllers reaction to noise, the output value density is made noisy with pseudo random white noise. A noise amplitude of 0.02 $\left[\frac{kg}{m^3}\right]$ on a value of 921 $\left[\frac{kg}{m^3}\right]$ is chosen. The output is filtered with a low-pass filter $\frac{0.2}{s+0.2}$. The same step as before at the ideal of the density is made. Only now, the output is noisy. The result can be seen in figure 4.6.

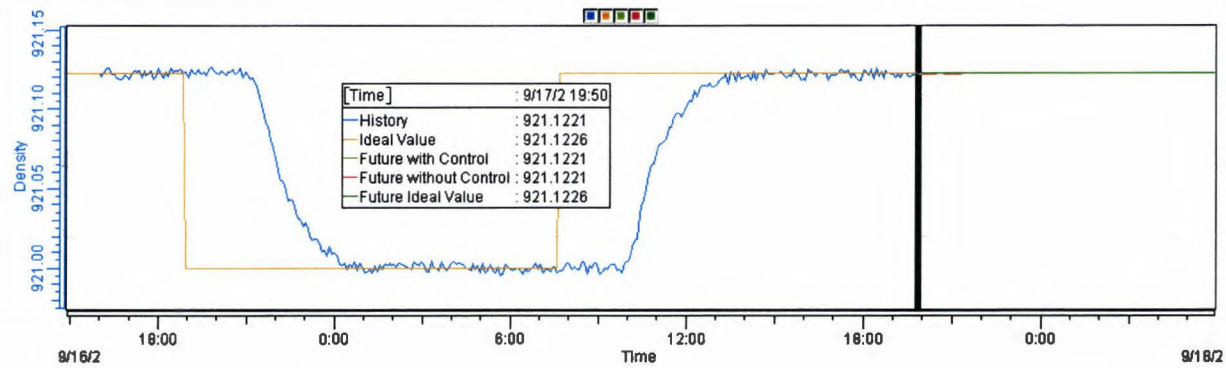

Figure 4.6: *Results of a step at the density's ideal value with added noise*

In practice all outputs will be a little bit noisy. Making only one output (the density is this case) noisy is not realistic, but is done to show the effect of noise at outputs in little steps. As can be seen, the controller is well able to reach the new ideal value. Again, approximately 7 hours are needed to do so. The output disturbances at the density value are fed back to the controller. It seems the controller is (almost) insensitive for these disturbances: the settling time stays the same as before and no extra overshoot occurs.
Again also other CV's are influenced by the negative step at the product density. The screenshot below (figure 4.7) shows the meltindex during the change of density. When comparing figure 4.7 with figure 4.5, one can conclude the noise at the value of the density has almost no influence on change of the meltindex. Only a few small deviations are visible but the time, needed to correct the value of the meltindex, is the same as in figure 4.5.

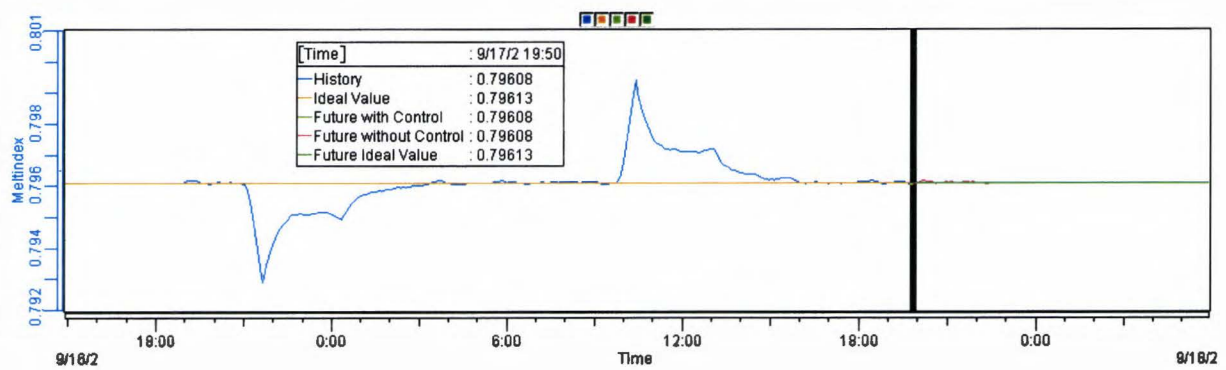

Figure 4.7: *Change of meltindex due to a step at the density's ideal value with noise*

4.3.3 More Noise at the Process

In most cases not only a step at the density of the product, but also a step on the meltindex must be made in order to achieve the wanted product quality. As one can imagine both outputs could be a little bit noisy at the same time. This situation is simulated with a noise amplitude of 0.02 $\left[\frac{kg}{m^3}\right]$ at the output density and a noise amplitude of 0.001 $\left[\frac{g}{10\ min}\right]$ at the output meltindex. Both outputs are filtered with a low-pass filter $\frac{0.2}{s+0.2}$.

Figure 4.8 shows the behaviour of the output density when a step down (from 921.1226 to 921 $\left[\frac{kg}{m^3}\right]$) and back is made. As can be seen, the controller is almost invincible for the added noise and brings the process value of the density to the new ideal value.

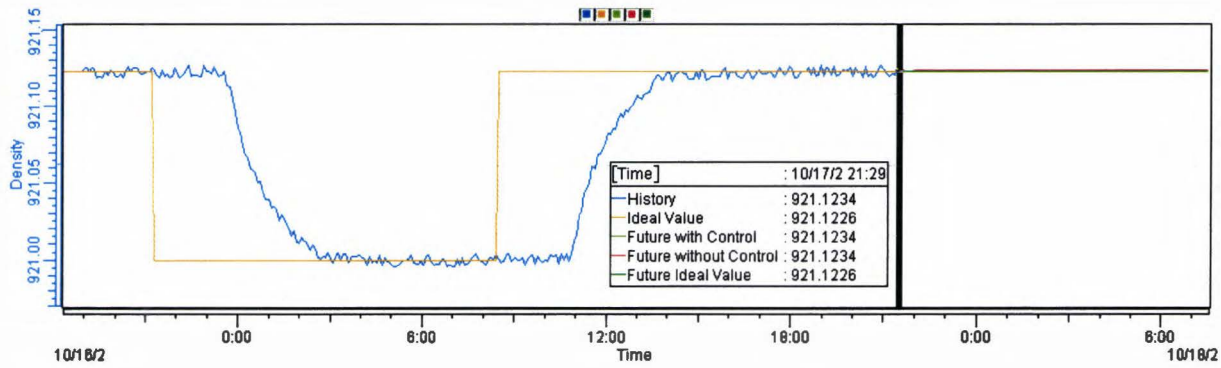

Figure 4.8: *Change of density due to a step at both the density's and meltindex's ideal value, both with noise*

Figure 4.9 shows the behaviour of the meltindex of the produced product. Both the change due to the step at the meltindex ideal value and the change due to the step at the density's ideal value, can be seen in this figure. After some delay, the value of the meltindex first decreases while the value should increase. This seeming wrong behaviour is due to the density's value which is changing at that moment (compare figure 4.9 with 4.8). Again, the new ideal value of the density is reached in approximately 7 hours. The new value of the meltindex is reached in approximately 8 hours.

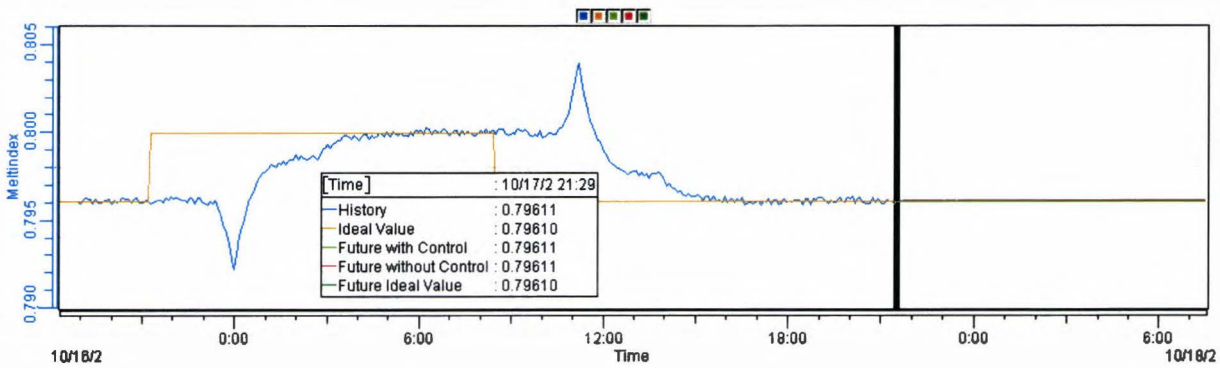

Figure 4.9: *Change of meltindex due to a step at both the density's and meltindex's ideal value, both with noise*

Of course not only the outputs can be noisy. Disturbances at the inputs and states of the process are also possible. Table 4.2 shows the amplitudes of the noise per input and output. For simplicity reasons there is no noise added to states of the process.

Table 4.2: *Amplitude of the added noise at the inputs and outputs of the process*

Input	Noise Amplitude	Output	Noise Amplitude
Initiator Feed Zone 1	0.001	Place Peak Zone 1	0
Initiator Feed Zone 2	0.01	Peak Temperature Zone 1	0.01
Feed Temperature Zone 1	0.1	Place Peak Zone 2	0
Feed Temperature Zone 2	0.1	Peak Temperature Zone 2	0.01
Jacket Temperature Zone 1	0.1	Output Temperature	0.01
Jacket Temperature Zone 2	0.1	Density Polymer	0.02
Jacket Temperature Zone 3	0.1	Meltindex Polymer	0.001

A step is made at the density's ideal value with noise (which can be seen in table 4.2) at the inputs and outputs of the process. Figure 4.10 shows the response of the density when a step at the density's ideal is made with noisy circumstances.

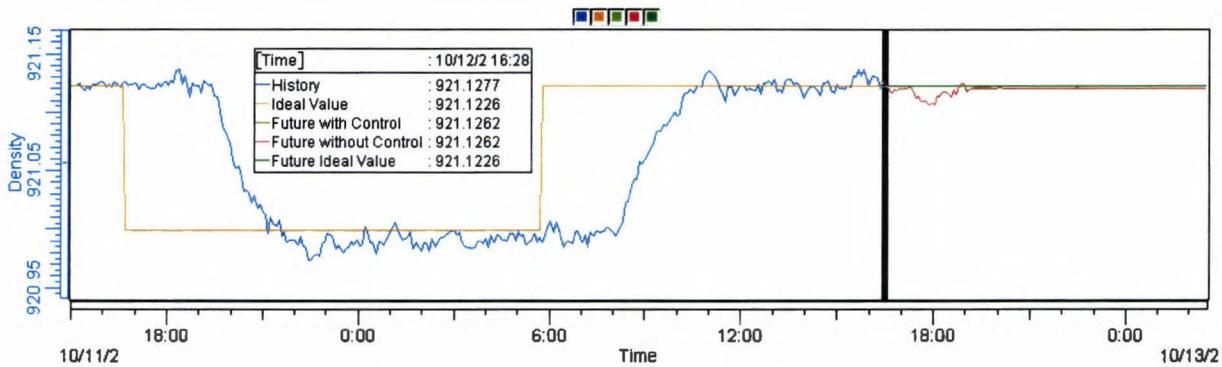

Figure 4.10: *Change of density due to a step at both the density's ideal value, with noise on all inputs and outputs*

From figure 4.10 it becomes clear the noise at the output is larger than before. Thus the noise at the inputs has influence on the outputs, as expected of course. It is more difficult to reach the new ideal value when a step is made. Also keeping the process value as close as possible at the ideal value, is more difficult. There is some undershoot which is not as clear as before due to the added noise. The settling time is now approximately 13 hours. Keeping the value of the meltindex ideal value is even more difficult. Not only this variable changes due to the step at the density's ideal value, also the noise at inputs and outputs make the process value oscillate round the ideal value. Figure 4.11 shows this meltindex behaviour.

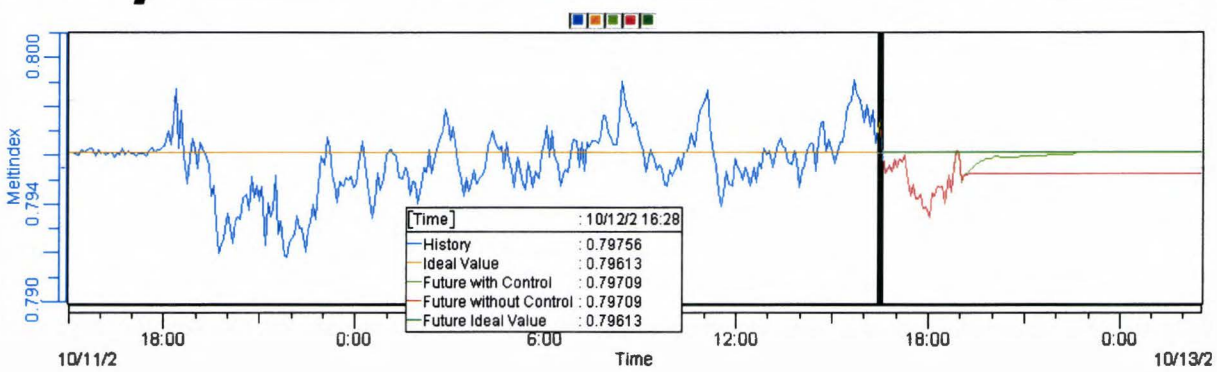

Figure 4.11: *Change of meltindex due to a step at both the density's ideal value, with noise on all inputs and outputs*

Another thing thats becomes clear in figure 4.11 are the delays in the process. At the beginning of the trend, there is noise added to the inputs (see figure 4.12), but it takes a while before that noise influence is noticeable at the end of the reactor. Therefore in the first part of figure 4.11 only the added output noise can be seen. After the delay time the input noise is also noticeable.

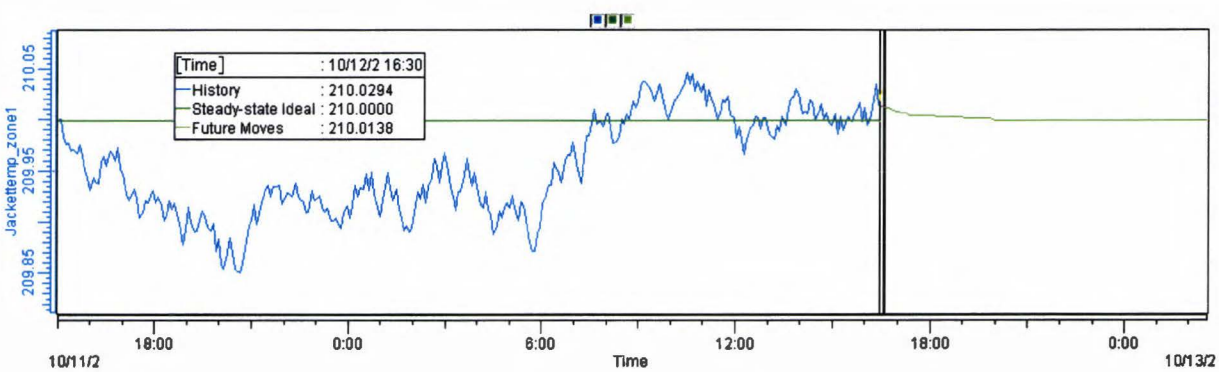

Figure 4.12: *Noise is added to the input Jacket Temperature Zone 1*

4.3.4 Slow Periodic Disturbances

Fast fluctuations such as pseudo random white noise are very difficult for a MPC controller to suppress. However, suppressing slow fluctuations should not be problem. This case is researched by adding a slow periodic signal to the outputs of the process. It is decided to disturb the quality outputs density and meltindex with a sinus and noise. The sinus has a period of 24 hours (i.e. the disturbance differs during a day of production). The amplitude of the sinus is 0.01 $\left[\frac{kg}{m^3}\right]$ for the output density and 0.001 $\left[\frac{g}{10\ min}\right]$ for the output meltindex. The added noise has the same amplitude as the periodic disturbance, but is filtered with a low-pass filter $\frac{0.2}{s+0.2}$.

First the situation where the controller is switched off (open loop) is researched. Figure 4.13 shows the density and figure 4.14 shows the meltindex for 24 hours.

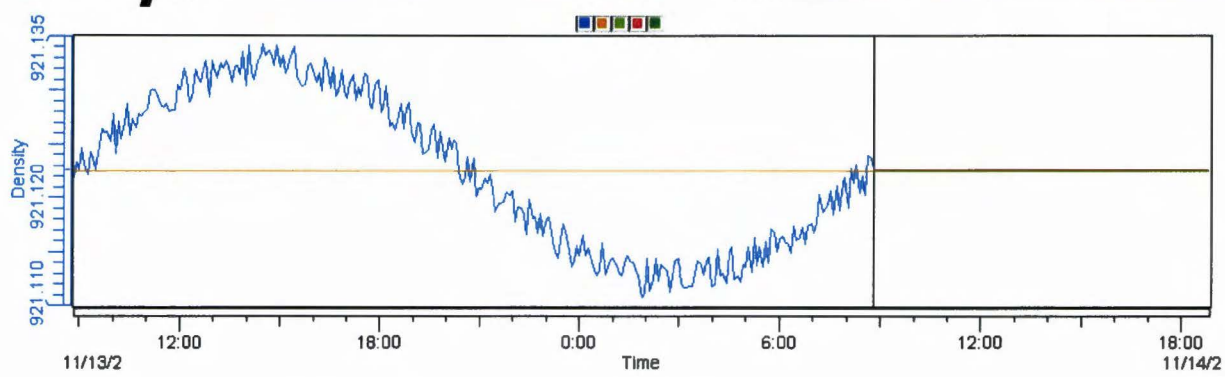

Figure 4.13: *Open loop density disturbed with periodic signal and noise*

The periodic disturbance can be seen very clearly. The value of the density ranges from 921.1108 to 921.1344 $\left[\frac{kg}{m^3}\right]$. The value of the meltindex ranges from 0.7950 to 0.7973 $\left[\frac{g}{10\ min}\right]$.

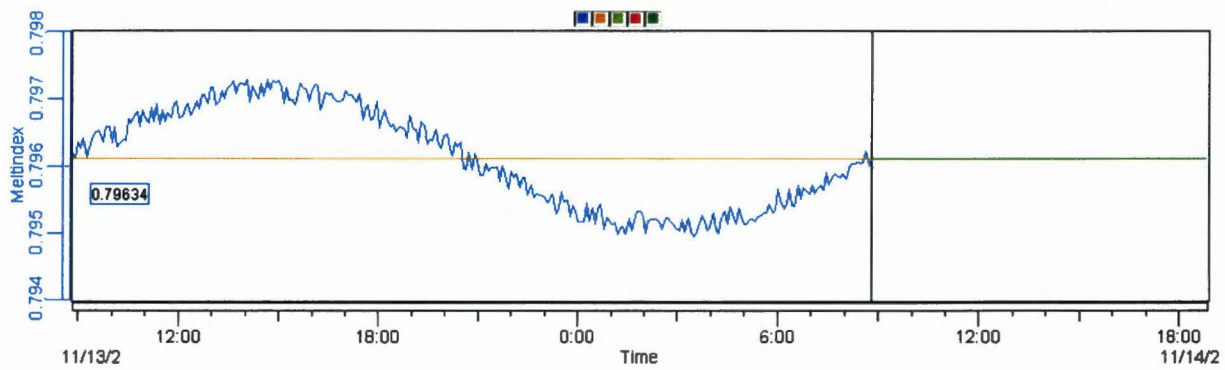

Figure 4.14: *Open loop meltindex disturbed with periodic signal and noise*

When closing the loop, the controller should be able to suppress at least the slow disturbance. Figure 4.15 shows the density and figure 4.16 shows the meltindex for 24 hours.

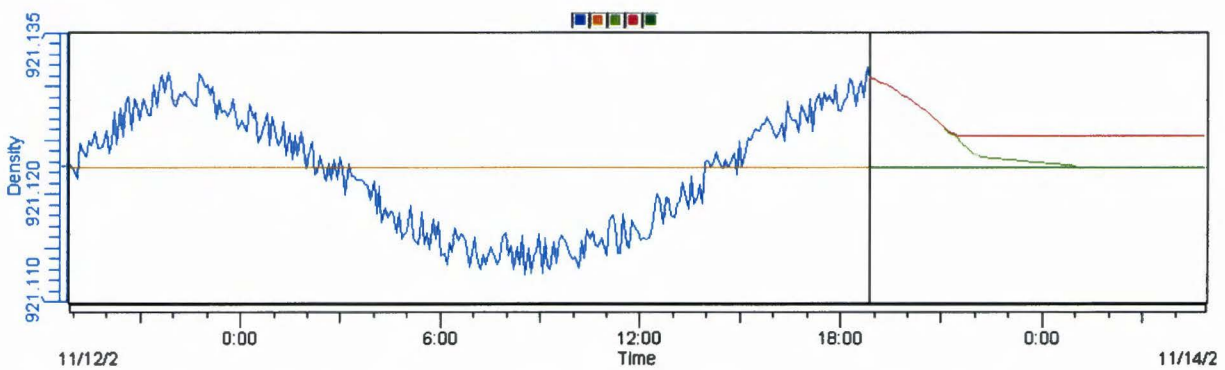

Figure 4.15: *Closed loop density disturbed with periodic signal and noise*

Again, the periodic disturbance can be seen very clearly. The periodic disturbance is not gone, but its period time and amplitude has changed. The change in amplitude is very small and can only be seen

when zooming in. The value of the density now ranges from 921.1126 to 921.1314 $\left[\frac{kg}{m^3}\right]$. The value of the meltindex now ranges from 0.7952 to 07970 $\left[\frac{g}{10\ min}\right]$.

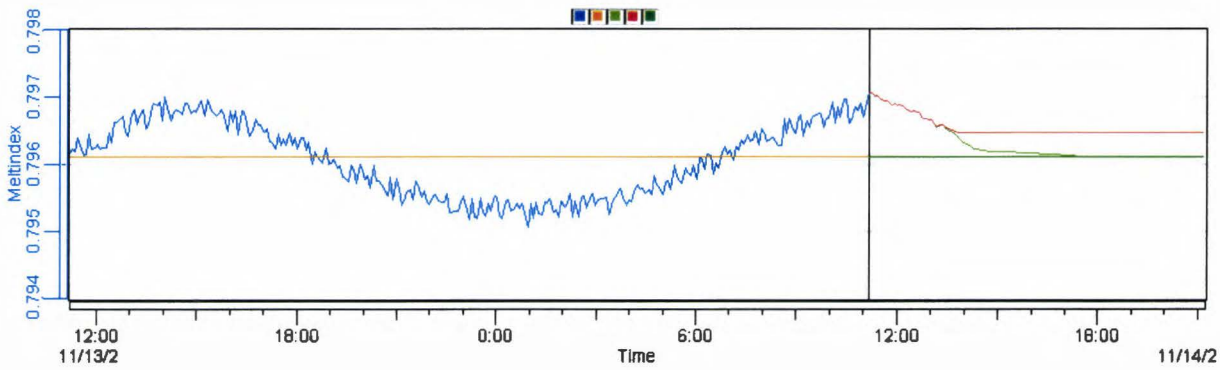

Figure 4.16: *Closed loop meltindex disturbed with periodic signal and noise*

It was expected that the controller would suppress the slow periodic disturbance. The reason for not doing this (or actually just a little bit) are the delays. To resists against the change of density and meltindex, several inputs (especially the feed and jacket temperatures) should be changed. Unfortunately these changes take a while before they are noticeable at the end of the reactor. In figure 4.17 can be seen that the controller changes the input 'Jackettemp_zone1' immediately after the distortion at the quality outputs is noticed by the controller.

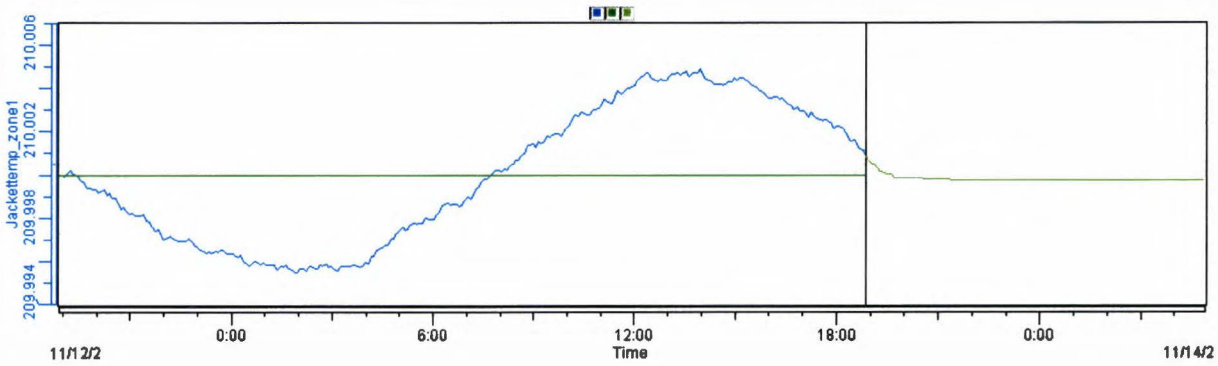

Figure 4.17: *In closed loop inputs such as 'Jackettemp_zone1' are actuated directly to correct the quality outputs*

When the process did not have (physical) delays, the disturbance would be suppressed. This can be seen in figure 4.18. This figure shows the values of the density and the meltindex directly at the process outputs. It also shows the disturbed density and meltindex which are fed back into the controller (i.e. the periodic disturbance is added on the way from the process outputs to the controller inputs).

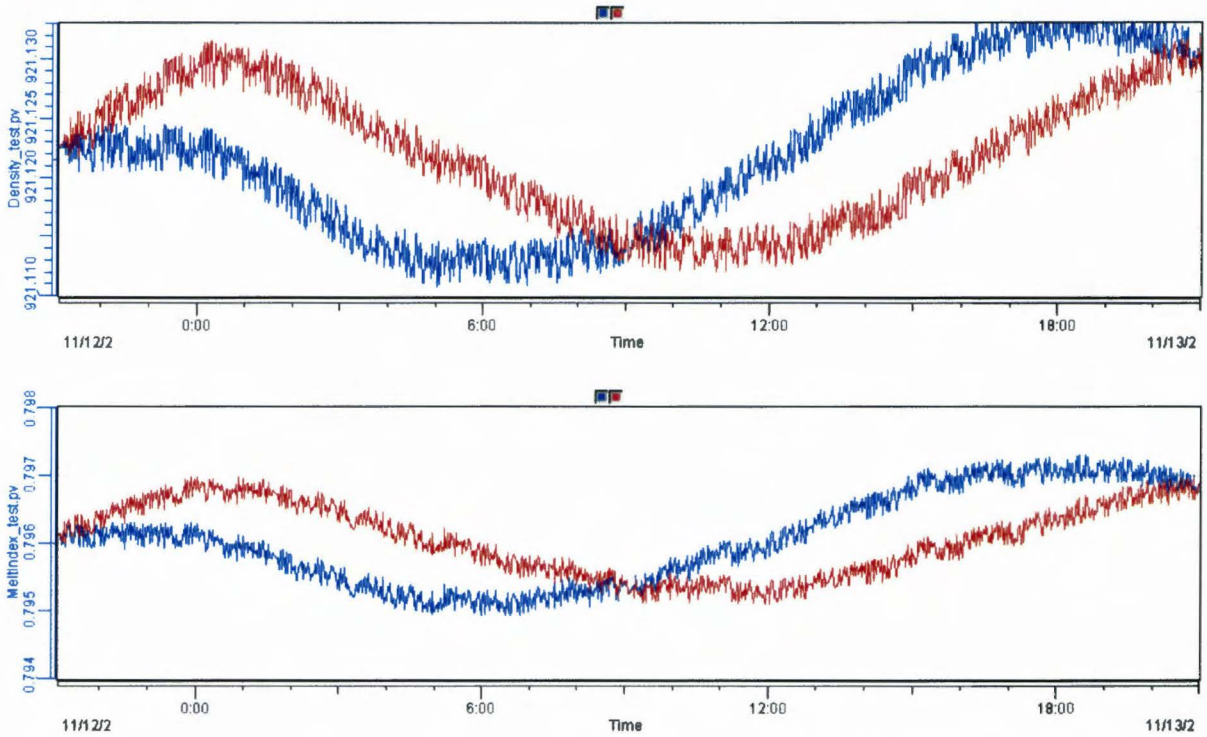

Figure 4.18: *Density and meltindex directly at the process outputs (blue) and before entering the controller (red)*

When there would be no delay, the blue trend in figure 4.18 would be shifted backwards in time and the slow periodic disturbance would be completely or almost completely suppressed.

Another way to make sure the controller can suppress this periodic disturbance, is to define this disturbance as a measured disturbance. If one knows the periodic disturbance is due to temperature changes during a day, the disturbance is approximately known and could be implemented as knowledge of the controller.

To prove the controller is able to suppress a disturbance, another partly periodic disturbance is added to the process outputs density and meltindex. The disturbance consists of a quarter of the just added sinusoidal disturbance and stays the same after the quarter of the period time. The amplitude of the sinus and the noise is the same as before. Figure 4.19 shows density when the controller is switched off (open loop). Figure 4.20 shows the meltindex.

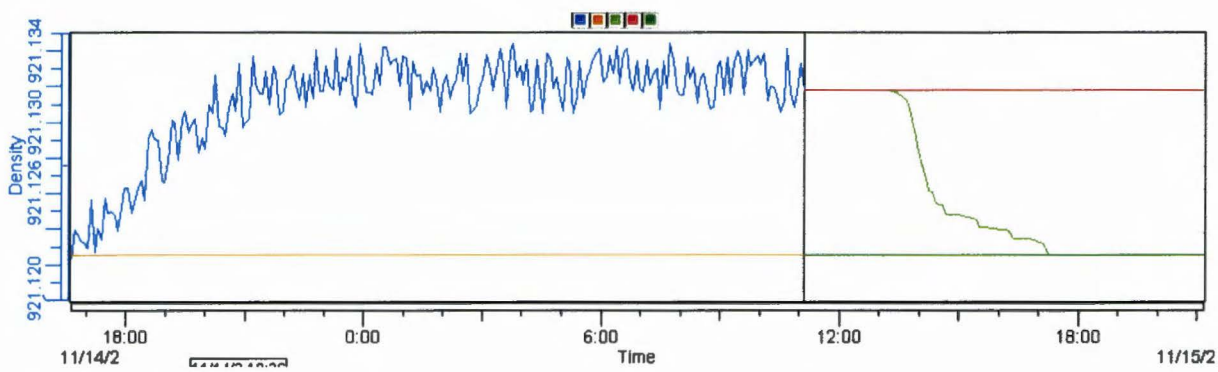

Figure 4.19: *Open loop density with partly periodic disturbance and noise*

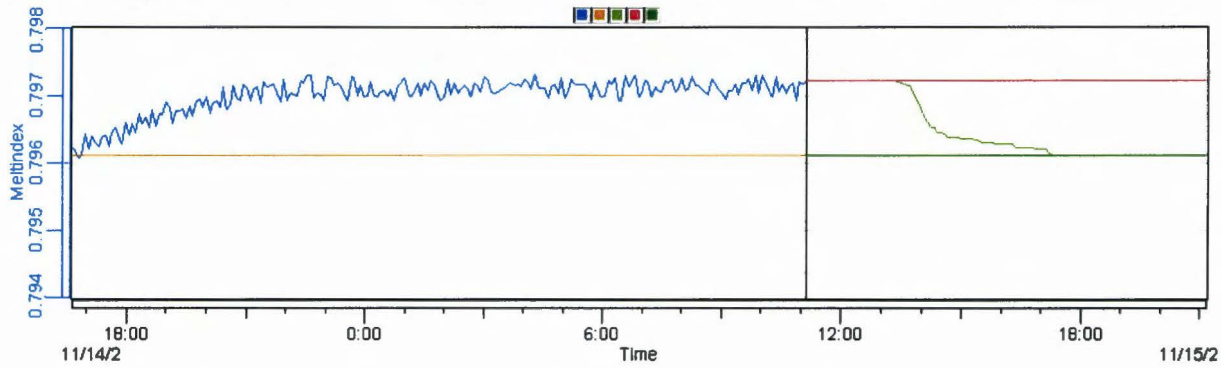

Figure 4.20: *Open loop meltindex with partly periodic disturbance and noise*

As can be seen, the meltindex and density have for 6 hours (a quarter of 24 hours) a sinusoidal behaviour and after that the values stay the same. Since the controller is switched off, the ideal value can not be reached again.

Figure 4.21 shows the product density with the same disturbance but now in closed loop. It can be seen that the density's value decreases after 4 hours back to the ideal value. This is due to the controller which changes the setpoints of the inputs of the process.

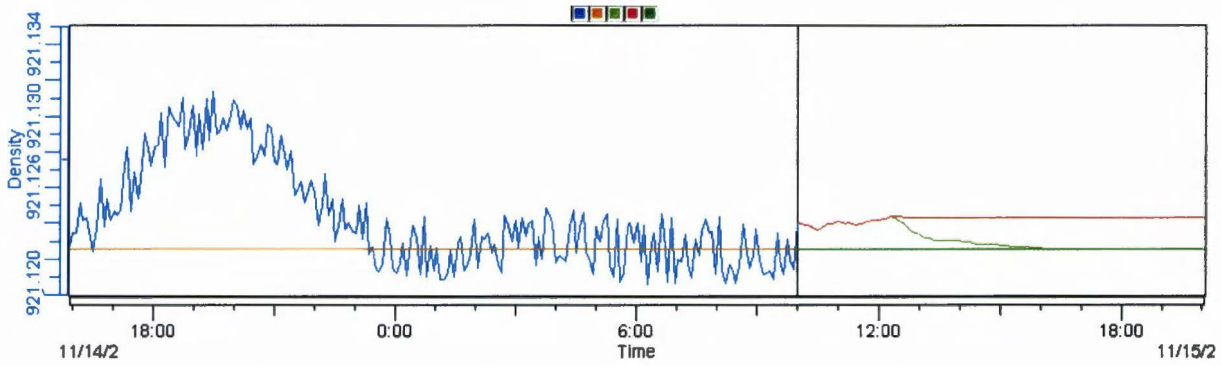

Figure 4.21: *Closed loop density with partly periodic disturbance and noise*

Not only the density is controlled, also the value of the meltindex is brought back to the ideal value, as can be seen in figure 4.22. After the start of the disturbance, it takes approximately 7 hours before the ideal value is reached again.

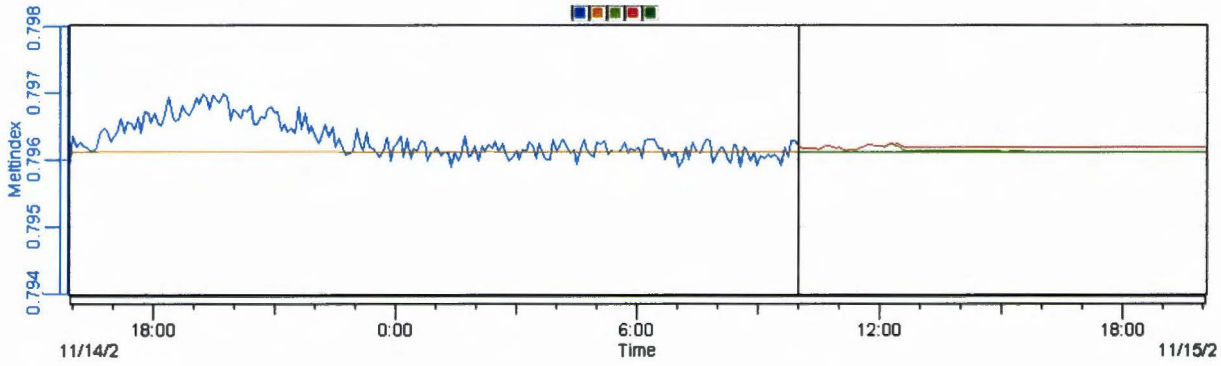

Figure 4.22: *Closed loop meltindex with partly periodic disturbance and noise*

In order to bring back the quality outputs to their ideal value, the controller changes the setpoints of several inputs. Figure 4.23 shows the change of the jacket temperature in zone 1.

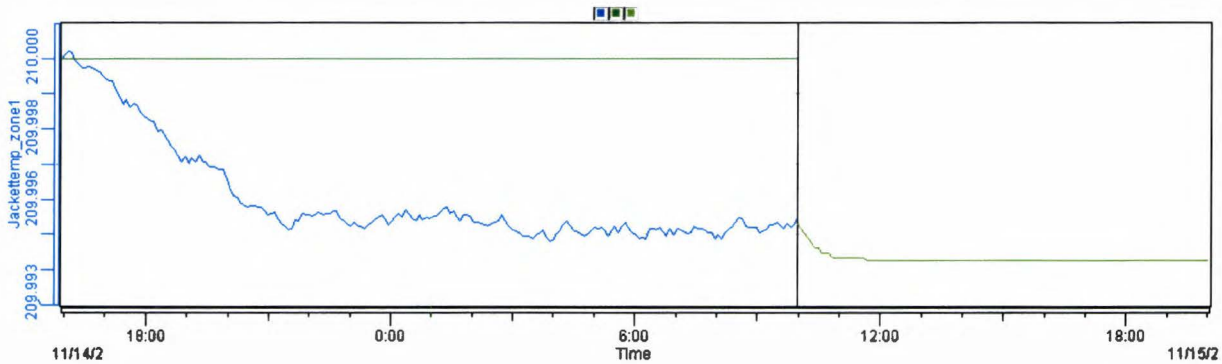

Figure 4.23: *The input jacket temperature zone 1 is changed in order to correct the quality outputs*

4.3.5 An Offset in Output Temperature

Also a little offset in temperature of the produced product is made. This output disturbance can occur in the summer where at noon the outside temperature is much higher than at night. This higher outside temperature influences the process in several ways (for instance the jacket temperatures of each zone). However for simplicity reasons only the influence at the output temperature is tested. The controller should be able to compensate the changes of the density and meltindex of the produced product due to this extra offset in output temperature. A value of 2 degrees Celsius is added to the output temperature (see figure 4.24). The controller is able to compensate this disturbance within a few hours (see figure 4.25).

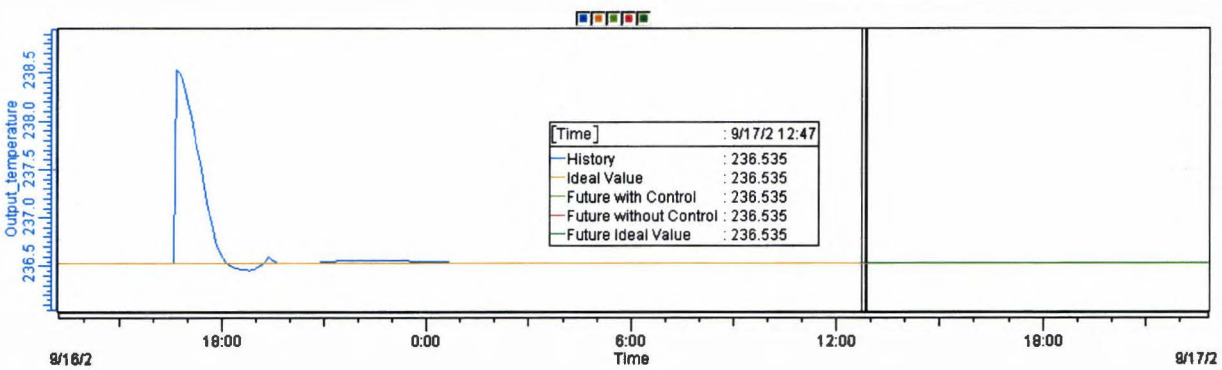

Figure 4.24: *Change of output temperature due to an offset at the output temperature*

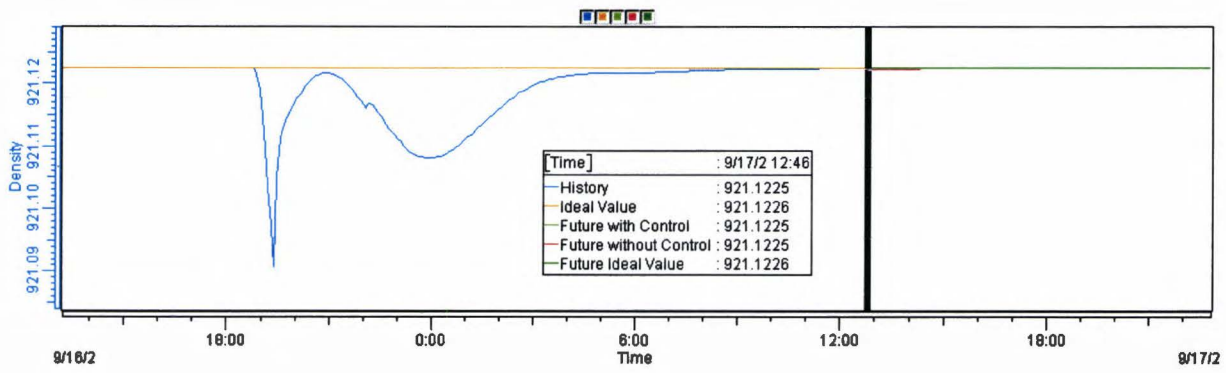

Figure 4.25: *Change of density due to an offset at the output temperature*

The first figure shows the extra offset at the output temperature. It also shows that this offset becomes smaller as time runs. It becomes smaller due to the controller which tries to reach the ideal value of the output temperature. The ideal value is reached again within 2 till 3 hours (in spite off some undershoot).

Unfortunately also other output values change due to this offset. One of those outputs is the density. In the second figure the change of the density can be seen. It takes a while (approximately 14 hours) before the old ideal value is reached again. To improve the performance for this extremely unusual case, the tuning strategy should be changed. Giving the output temperature a higher rank (i.e. lower priority in reaching the ideal value) will give the controller more freedom to (if necessary) correct the quality outputs.

4.3.6 Choosing a wrong Model

As the name already assumes, the performance of Model Predictive Control is very dependent of the used model. When the behaviour of the used model does not accurately describes the real process, controlling the process is difficult or even impossible. For example: when the sign of a certain transfer (between an input and an output of the process) is different in the model than in practice, the closed loop system could be instable.

To show how important the model actually is, the model for grade A is replaced with the model for grade G. These models differ a lot; their transfers have other delays, other dynamics and in most cases other gains. In figure 4.26 two transfers of both models can be seen.

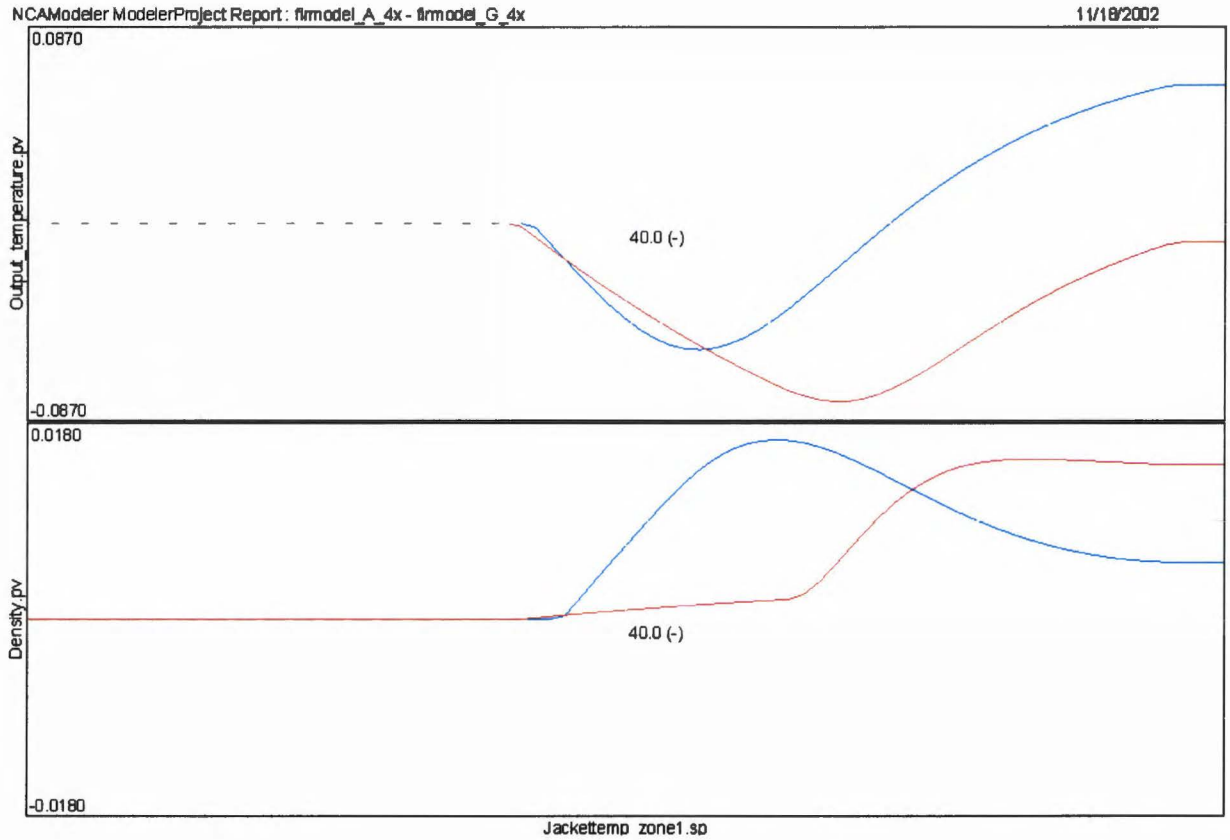

Figure 4.26: *Two transfers of the different models A and G*

The same step is made as in the first part of figure 4.3 in section 4.3.1 This step can also bee seen in figure 4.27. Figure 4.28 shows the response of the density.

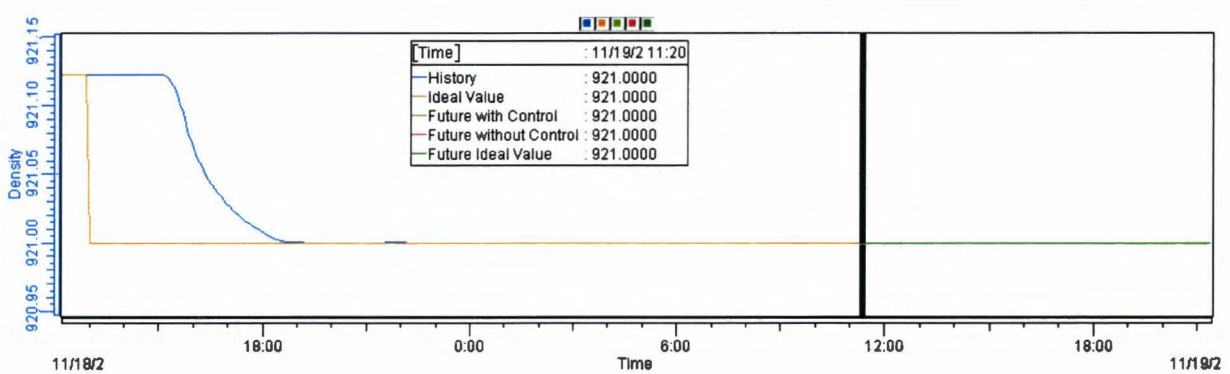

Figure 4.27: *Change of density using the correct model A to control the process*

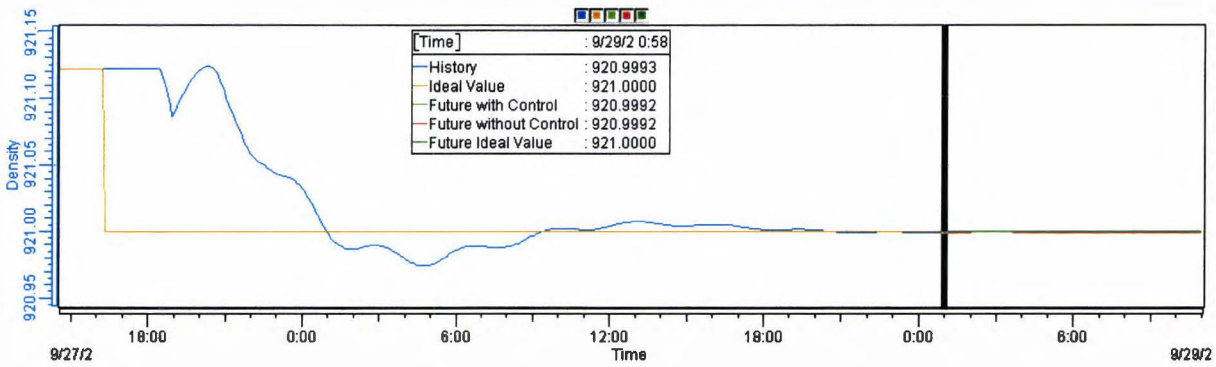

Figure 4.28: *Change of density using wrong model in the controller*

Figure 4.29 shows the response of the meltindex. The process value oscillates and deviates from the ideal value. After a while the ideal value is almost reached again.

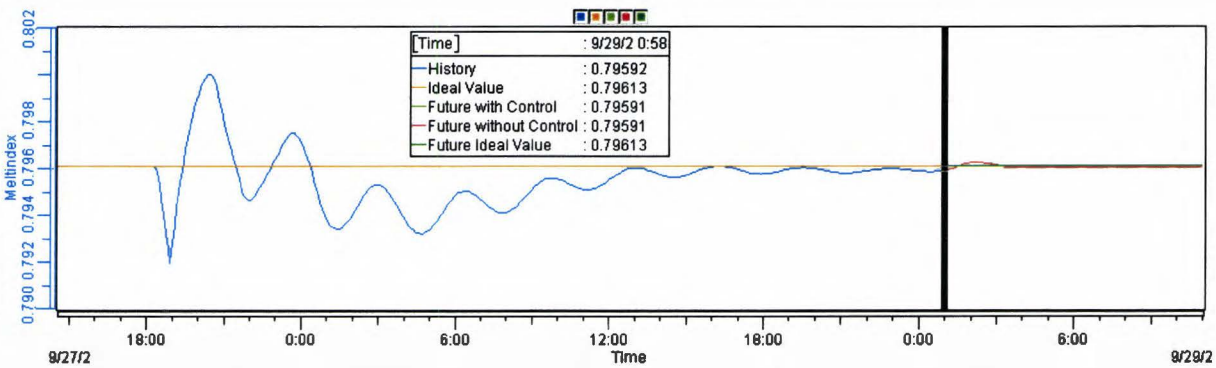

Figure 4.29: *Change of meltindex due to step at density using wrong model in the controller*

The oscillation of the outputs is due to oscillation at several inputs. Figure 4.30 shows the input 'Jackettemp_zone3' which is oscillating too. The reason for this oscillating are wrong gains in the model used to control the process (the gains in the model are too small with the effect that the controller actuates the inputs too much).

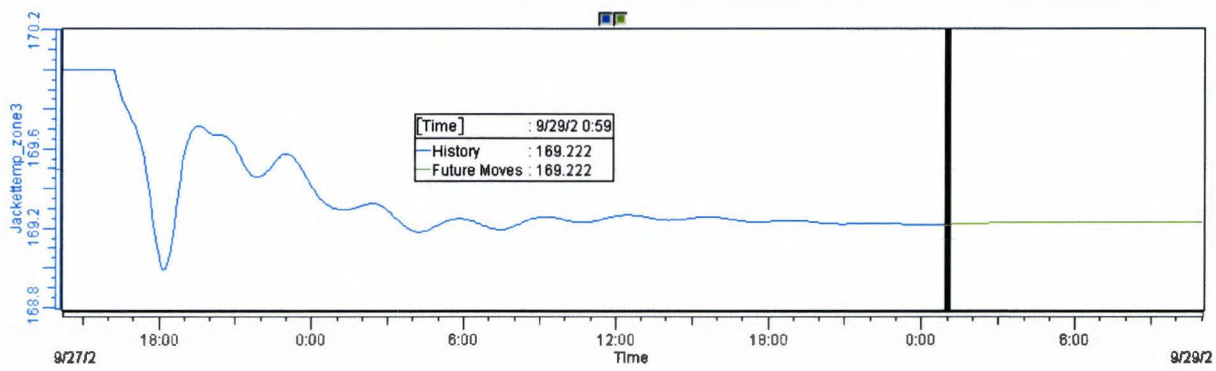

Figure 4.30: *Change of the input Jackettemp_zone3 due to change of the density's ideal value*

When the model is bad and there is no possibility to log new data in order to identify a new model, one could try to improve the performance of the controller by changing the tuning parameters. Figure 4.31 shows the same step as in figure 4.28. The controller using a wrong model is now better tuned: the ideal weightings are changed. It is clear that the performance has increased, but is still not as good as possible (compare this figure with figure 4.27).

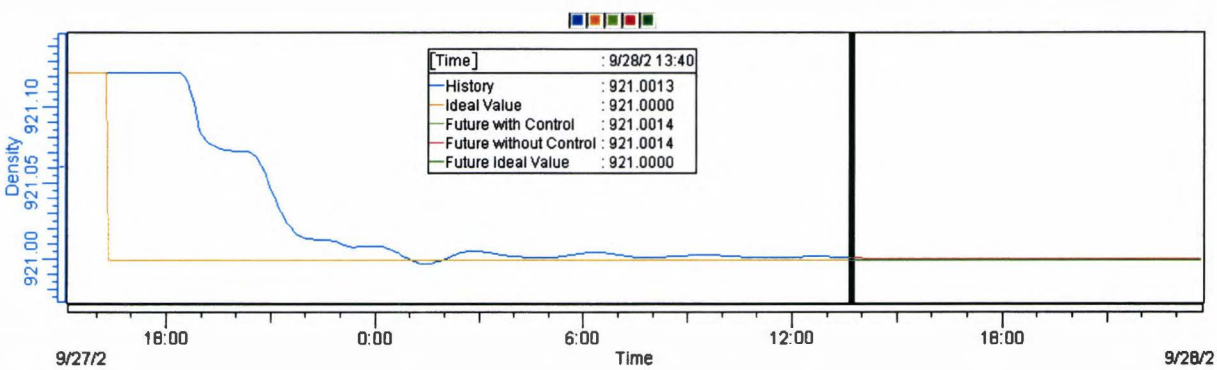

Figure 4.31: *Change of density using wrong model in the controller (a little bit tuned)*

Note that not only the model is important. Using a good model with bad tuning parameters will also lead to bad performance of the MPC controller.

4.3.7 Conclusion

In this chapter the controller setup, strategy and achieved performance is discussed. With a good strategy the product quality of polymer can be controlled well. Not only the right tuning parameters are necessary, also a good model which fits well on the process is essential. This is made clear in section 4.3.6.

In practice noise or other disturbances are present at inputs, outputs and states. In this chapter the behaviour of the outputs responsible for the product quality is tested with noise on it. Also noise is added to the inputs of the process. In the first case the performance of the controller is good enough.

In the second case, the performance is still reasonable; some overshoot occurs and the settling time is a little bit larger. It is shown that periodic disturbances can not completely be suppressed. The reason for this are the physical delays of the process. When only a part of the periodic disturbance is added to the process, the controller is able to correct the quality determining outputs to their ideal values.

In this chapter only the performance of the controller for product grade A is discussed. The performance check of the controllers of all other grades can be done equally and is therefore further not discussed in this report.

Chapter 5

Multi-Modelling

It may already be clear that the simulation of the tubular LDPE reactor is very non-linear. When identifying a linear model, little model mismatches are made automatically. An identified linear model is therefore valid for a limited operating range (see section 3.2 for more information). When a larger operating range is needed, because of the production of polymer with several product qualities and corresponding operating ranges, more than one linear model or a non-linear model is needed. However, the *INCAEngine* is not able yet to handle non-linear models. Therefore several linear models will be used to describe a larger range.
In this chapter, the clustering of several linear models, the implementation of multi-modelling and the switching between product grades will be treated.

5.1 Clustering the Grades

A first idea to simplify the problem is to cluster several grades. This has the advantage that the number of models necessary to switch between grades, can be reduced. For example: when two grades lay far apart, the two models for those grades are not enough to switch between the two grades (i.e the two working areas do not overlap). Therefore one or more intermediate models between two grade-models, concerning the behaviour, are needed. When for all combinations of two grade-models one or more extra models should be defined, the number of models increases quickly.

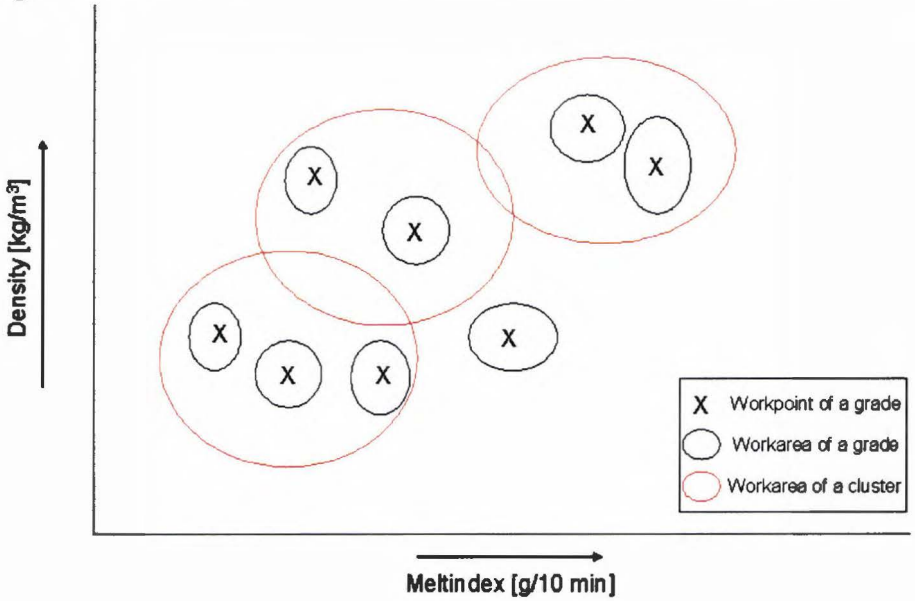

Figure 5.1: *Example of how grades could be clustered and how operating ranges lap each other*

In practice the management of a factory for which models should be made, will not accept when many models should be identified. The identification process takes a lot of time in which the process is in someway disturbed (e.g. making little steps at the inputs, directly interferes the produced polymer). So it is desirable to use as little as possible models to control the complete process with all its products and qualities. Therefore the grade-models are being clustered based on their transfers. Figure 5.1 shows an example of a clustering procedure. Note that this example is chosen completely random and has no connection to the introduced grades in this report.

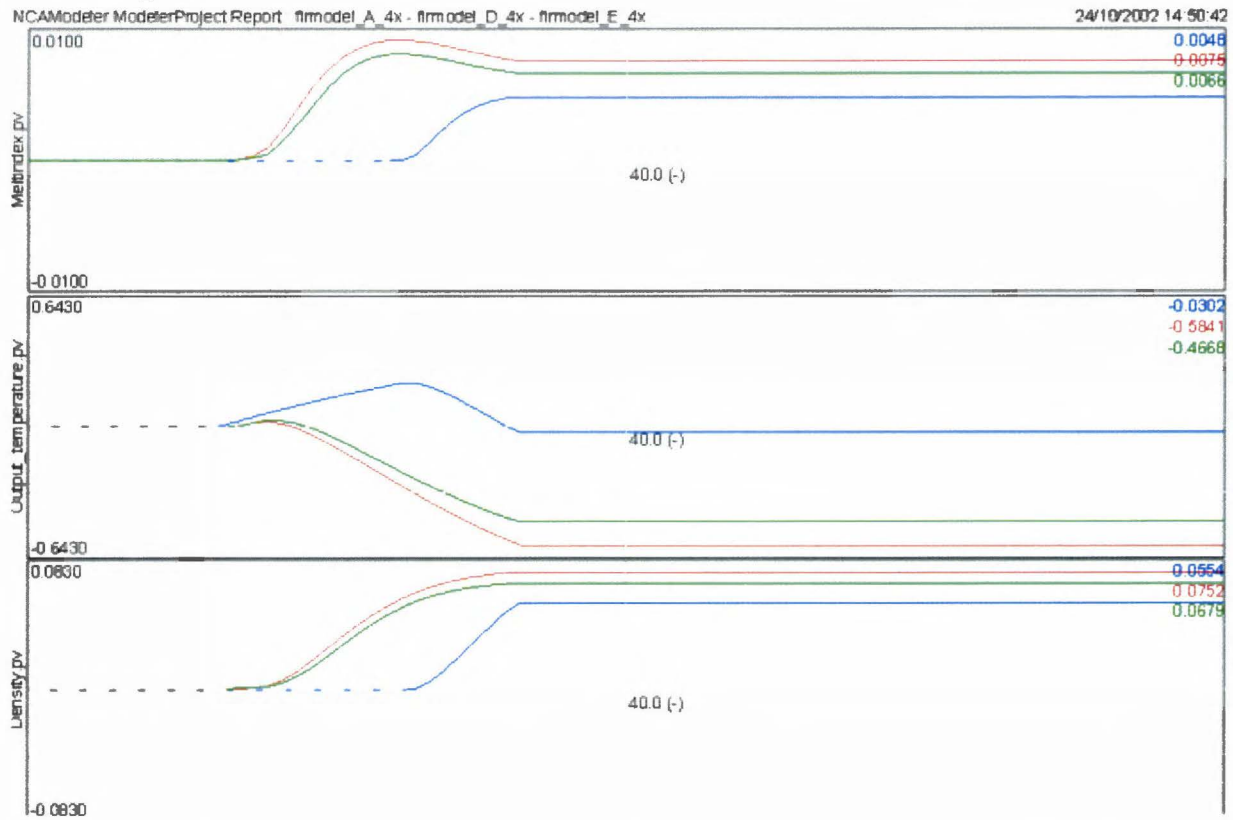

Figure 5.2: *Based on the step responses of each transfer, several grades are put in a cluster. Blue = grade A, red = grade D and green = grade E*

In figure 5.2 three step responses of models of different product grades are shown. From this figure can be concluded that grade D and E can possibly be put in a cluster: the dynamics and gains of the showed step responses equal very much. Grade A differs too much (different delays, different dynamics and different gain) from the other two grades and therefore it is most likely that it can not be put in the same cluster. Of course not only the transfers from input 'Jacket temperature zone 2' to the three outputs output temperature, density and meltindex should be watched. All different transfers must be considered when building a cluster.

All grade-models have different transfers. This can be seen in plots like the one for grade B in appendix D. The step responses differ per model. Not only the delays of a single step response are different, also the gain and dynamics differ from model to model. Therefore the models that behave approximately the same (i.e. have almost the same delay, dynamics and if possible gain), are put together in a cluster. The clusters definition can be seen in table 5.1. As you can see, two grades are not clustered. Grade C differs to much from the other grades (and clusters). Grade F is not used at all because of its very small operating range.

Table 5.1: *Definition of the used clusters*

Cluster	Contain grade
1	D and E
2	B and G
3	A and H

For each cluster, a model to control the process must be identified. A cluster model is also identified round a operating point. The operating point of a cluster is an average of the operating points of the

grades which lay in that cluster. The operating range of the cluster is at least so large that it contains the operating ranges of each included grade. It is wishful for the cluster model to describe even a larger operating range. Finally, the complete work area of the process must be described with as few models as possible (identifying extra models costs time and money). The model for a certain cluster is identified out of data received from logging data when a testsignal is put on the simulated process. For more information about the identification-process, see chapter 3. The operating point of a cluster is determined averaging the values of the inputs that determine the operating point of a grade (Initiator feeds and the feed temperature). These values for each cluster can be seen in appendix E in table E.1 till table E.3. Note that a cluster model will be less accurate than a grade model, since it describes a larger (non-linear) operating range and in fact is a average of several grade models.

Another way to define a cluster model, is to average the grades models that are included in the cluster model. The gain and delays can be averaged easily. However, the dynamics are more difficult to average. It can be more accurate to identify a cluster model from scratch. In practice this will cost too much time, but since a simulation of the process is used, this method is chosen to obtain the cluster models.

After obtaining cluster models, the following strategy is tested. When switching to another product grade, the cluster model containing the current grade is loaded. When also a new set of tuning parameters (belonging to the just loaded cluster model) is loaded, a larger operating range is created and the controller gets more freedom to change the output values in the right direction. Note that loading a less accurate model will lead to less accurate control. When the operating range of the model of the new (wanted) grade is reached, the cluster model is replaced by the model belonging to that new grade.

5.2 Performance of a Cluster Model

It is already said that a cluster model will perform different than a grade model. It is even logical since the the cluster model describes a larger operating range in a non-linear process. In this section will be shown that the performance (e.g. settling time, overshoot or not) differs from the performance when using a grade model. This is done based on the same step responses using different models (cluster or grade).

5.2.1 Model Validation

First a model of a cluster is tested with validation data. As already told in chapter 3.6, an identified model can be verified using logged data which is not used to identify a model. The independent data of the inputs are used to predict the outputs. When comparing the logged output with the predicted outputs, a judgement of the models performance can be made. Figure 5.3 shows the process value and the predicted value of the outputs density and meltindex.

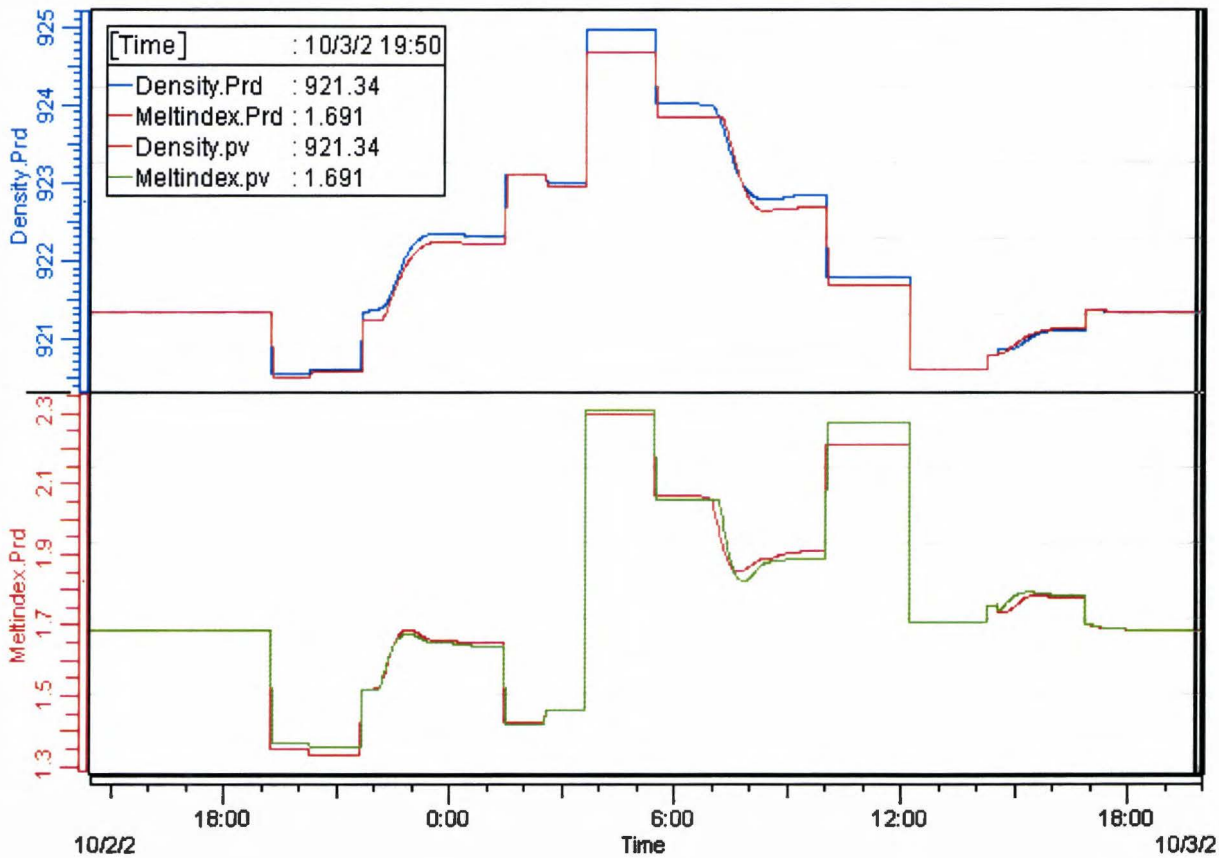

Figure 5.3: *Process value (.PV) and the predicted value (.PRD) of the outputs density and meltindex using cluster model 1 to predict the outputs*

The same can be done for the model of cluster 2. Figure 5.4 shows the process value and the predicted value of the outputs density and meltindex when predicting the outputs using cluster model 2. Note that the used verification data for cluster model 2 differs from the data used to verify cluster model 1. It differs because each cluster model describes a different operating point and range.

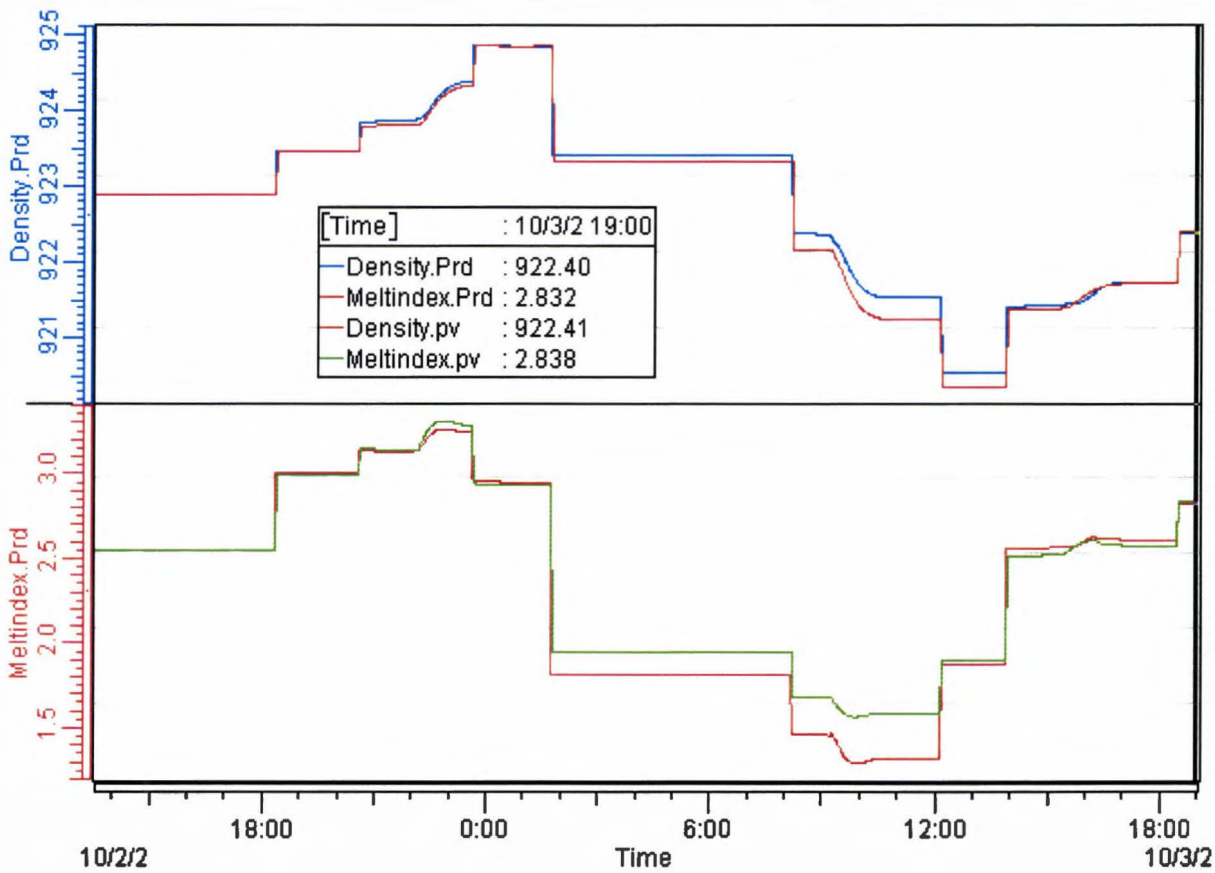

Figure 5.4: *Process value (.PV) and the predicted value (.PRD) of the outputs density and meltindex using cluster model 2 to predict the outputs*

It can be seen in both figures of this subsection that the models behave well enough. Sometimes the gain is not exactly as it should be, but the global behaviour of the models is alright.

5.2.2 Change of Ideal Value

Now the performance of a cluster model, based on its behaviour for some step responses (e.g. short settling time, overshoot or not), will be discussed. First cluster model 1 will come across. A step at the ideal of both the density and the meltindex of the product is made. This change of ideals is not chosen randomly but actually is the switching from product grade E to grade D. Both grades lay in cluster 1. The simulation is started using the model of cluster 1 to control the process. Also the tuning parameters belonging to this cluster are loaded. After the process is brought in grade E and is in steady state, the actual performance simulation starts.
Figure 5.5 shows the density when changing the ideal values several times. First the density's ideal value is increased to 920.5 $\left[\frac{kg}{m^3}\right]$ and the meltindex value is increased to 1.3 $\left[\frac{g}{10\ min}\right]$. The values of these outputs are changed at the same time because both values should be changed in order to reach the new grade. It is also possible to change the values after each other. Unfortunately, it is not researched yet if this is better for the performance during the switching. After reaching steady state of the new grade, the ideal values are changed back to the values belonging to grade E. This changing of ideals is done three times: once with cluster model 1 loaded, once with the model of grade E and once with the model belonging to grade D.

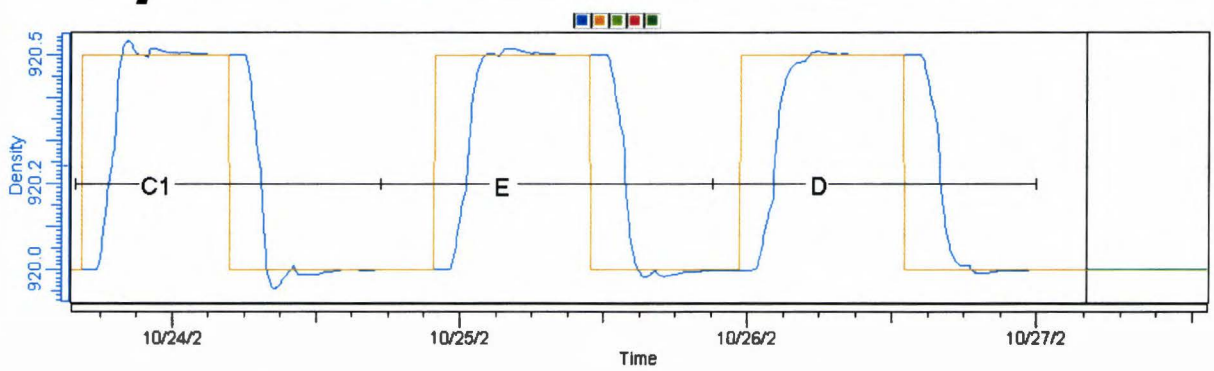

Figure 5.5: *Performance of the models C1, E and D with the same tuning parameters after changing the product, viewing the change of the density*

From figure 5.5 it may be clear that the model of cluster 1 (C1) performs worse than the models for grade D and E. It has the most overshoot and needs the most time when following the change of the ideal value. The main reason for this is the fact that the cluster model is made for a larger operating range and is therefore not as accurate as a grade model. However, the cluster model should be able to make larger changes of ideal values than the grade models. This will be shown in subsection 5.2.3. Also can be seen that the change of ideal value is done fastest with grade model D: it takes approximately 8 hours to reach product grade D and approximately 9 hours to go back to grade E.

Figure 5.6 shows the meltindex when changing the product grades as described above. It becomes directly clear that model D is not as good as thought after watching the previous figure. In this case, it is slower than the models E and C1. Besides it has some overshoot when following the ideal value. Model E performs best because of the less fluctuations it has compared to model C1. Note the sharp peaks in figure below. They are probably caused by sudden discontinuities in the simulation.

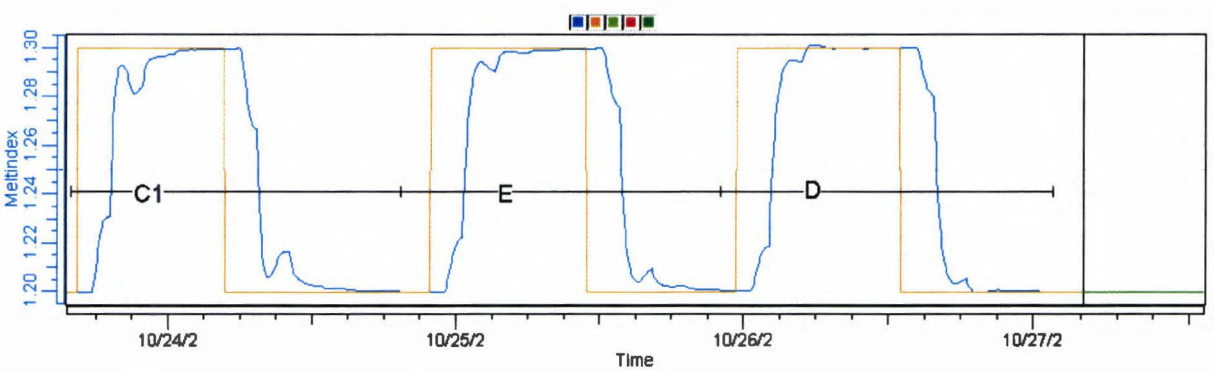

Figure 5.6: *Performance of the models C1, E and D with the same tuning parameters after changing the product, viewing the change of the meltindex*

In figure 5.5 grade model D and E performed approximately the same, in figure 5.6 model E performs better because of less fluctuations and less overshoot. Therefore it can be concluded that model E has the best performance if only this performance check is done. However, in the next subsection another test will be done to test the performance when making larger steps.

Now the performance of cluster model 2 will be discussed. In order to judge the performance differences between models, first a step at the density's and meltindex ideal value is made. Figure 5.7 shows the density of the product after changing the ideal values. Note that the change of these ideal values is not chosen at random, but actually is a product change from grade B to G. In the first part of this

screenshot, the model belonging to Cluster 2 (C2) is used. In the second part the model belonging to grade G is used. In the last part, model B was loaded. In all cases, the tuning parameters of model C2 are used.

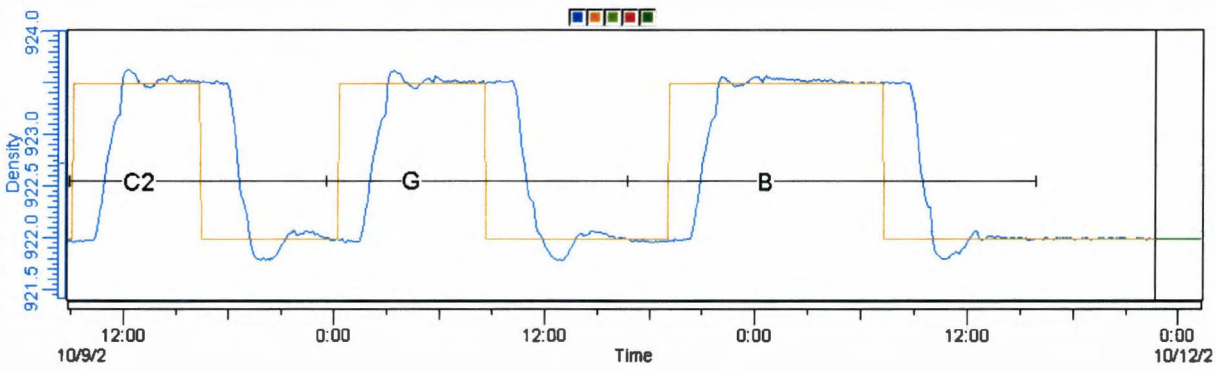

Figure 5.7: *Performance of the models C2, G and B with the same tuning parameters after changing the product, viewing the change of the density*

As can be seen, there is almost no difference between the behaviour of model C2 and model G when looking at the change of density. The behaviour when using model B is a little bit better: there is less overshoot when following the step at the ideal value. However, not only the output density should be observed when judging the performance of the models. Of equal importance is the output meltindex since this one determines the product quality as well. Figure 5.8 shows the change of the meltindex when changing the product from grade B to G.

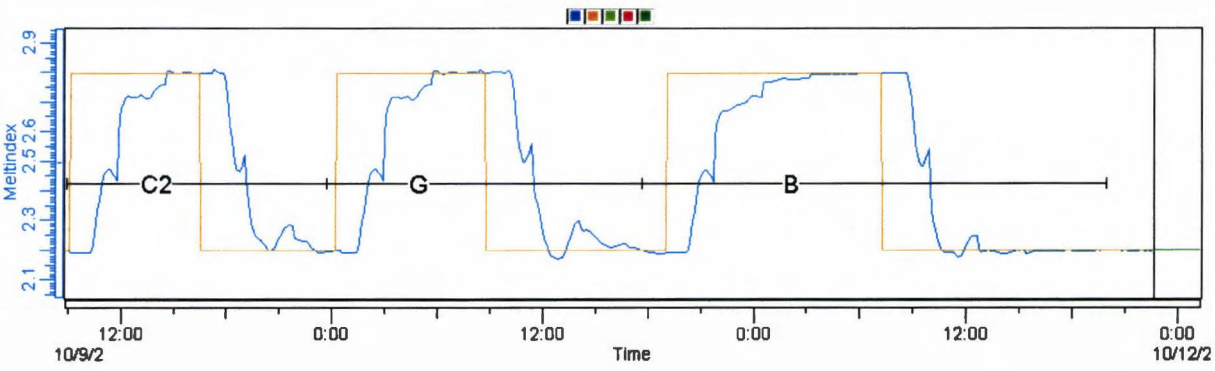

Figure 5.8: *Performance of the models C2, G and B with the same tuning parameters after changing the product, viewing the change of the meltindex*

Now the model of grade B performs a worse (it is much slower) than the models G and C2. When taking an exact look at the performance of model C2 and G, one can conclude that model C2 performs a little bit better. When changing the ideal meltindex value from 2.8 back to 2.2 $\left[\frac{g}{10\ min}\right]$ some undershoot occurs when using model G. That undershoot does not occur when using model C2.
Again, the sharp peak are probably caused by sudden discontinuities in the simulation.

5.2.3 Larger Steps at Ideal Values

It is already said that a cluster model should perform better when larger steps are made since it has been developed for that purpose. Therefore a large step at both the density and the meltindex of the

product is done using model C1, E and D again. The simulation is started with model C1 in grade E. A step is made to grade B (density 922 $\left[\frac{kg}{m^3}\right]$ and meltindex 2.2 $\left[\frac{g}{10\ min}\right]$). Figure 5.9 shows how all used models perform. It is directly clear that model D is not suitable, since it is slower than the others and a product change should go as fast as possible. The fastest change of product grade is made with the cluster model C1. In approximately 10 hours the new grade is reached, where using the other models leads to a change time of approximately 14 and 16 hours.

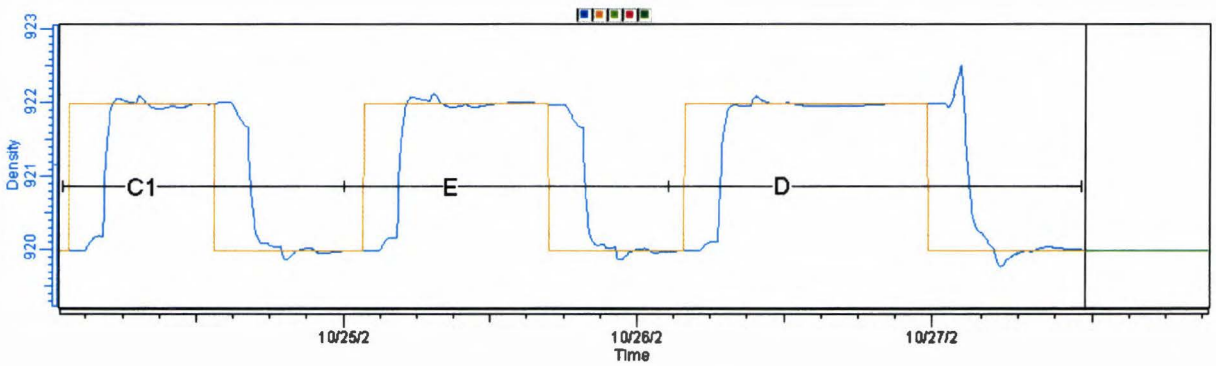

Figure 5.9: *Performance of the models C1, E and D with the same tuning parameters after changing the product with a large step, viewing the change of the density*

Not only the change of density is important, also the change of the meltindex must be considered. Figure 5.10 shows that change of meltindex. Also in this case the cluster model C1 performs the best. It has a little less overshoot than the other models and besides is much faster in reaching the ideal value. In approximately 11 hours that new value of the meltindex is reached.

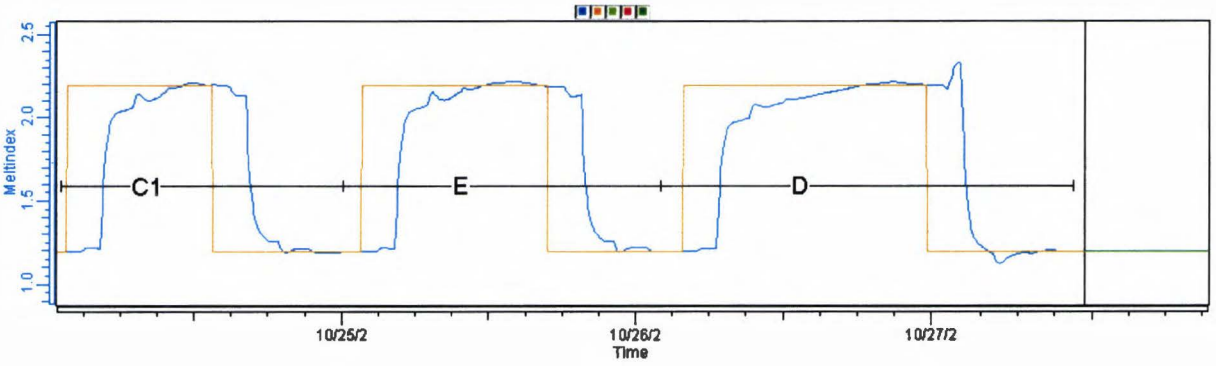

Figure 5.10: *Performance of the models C1, E and D with the same tuning parameters after changing the product with a large step, viewing the change of the meltindex*

Overall it can be concluded that the cluster model C1 performs better than a grade model when large steps have to be made.

5.3 Multi-Modelling and INCAEngine

Since version 2.1, the *INCAEngine* is capable to use multiple models to control the process. However, there is the restriction that if initially a FIR model is used, the other models to load must also be of

FIR type (the same for State Space models). In future releases of the Engine, this condition will be removed and both FIR and State Space models can be loaded randomly.

When a new model is loaded, it should first be initialised. There are two kinds of initialisation possible: cold and hot initialisation. When cold initialisation is chosen, the inputs and outputs of the model are made equal to the current inputs and outputs of the process. The other model states are made zero. The process is assumed to be in steady-state.

There are two ways to initialise a model hot: based on simulation and observer based. When the model is initialised hot, based on simulation, the state of the model is simulated using historical data. Only the input states for the new model are obtained this way. The output states are corrected in such a way that they become equal to the most recent sample of the historian. When a FIR model is used always this kind of hot initialisation is used.

In the observer based hot initialisation, a finite time observer estimates the best model states using the historical data for both the inputs and the outputs of the model. The hot initialisation of the new loaded model is done by an external program called *INCAStateEstimator*. The StateEstimator directly communicates with the Engine and gives the Engine a sign when the loaded model is initialised and the actual switching can be realised.

5.4 Switching between Product Grades

In subsection 5.2.2 already the change between two product grades is discussed. In that case, the two grades between which was changed, lay both in the same cluster. In this section the situation is discussed where two product grades lay far away from each other, but the operator of the plant wants to switch between them.

5.4.1 Two complete different Grades

When switching between two completely different grades (i.e. their transfers look complete different), it could be possible that more than one cluster model needs to be loaded. In this subsection the results of switching between grade E and G will be discussed. The switching process will be treated in several parts. In the next subsection a switching as fast possible will come across.

After starting the simulation in grade E with model E to control the process, the density's value is 919.92 $\left[\frac{kg}{m^3}\right]$ and the meltindex is 1.23 $\left[\frac{g}{10\ min}\right]$. This is exactly the ideal value at that moment. The ideal value is on purpose equal to the process value because in this case the process is in steady state when starting a simulation. The first thing to do is to change the ideal value of the density and the meltindex to 920 $\left[\frac{kg}{m^3}\right]$ and 1.20 $\left[\frac{g}{10\ min}\right]$ respectively. As can be seen in figure 5.11 overshoot occurs at the output density. Also the response of the output meltindex has some overshoot.

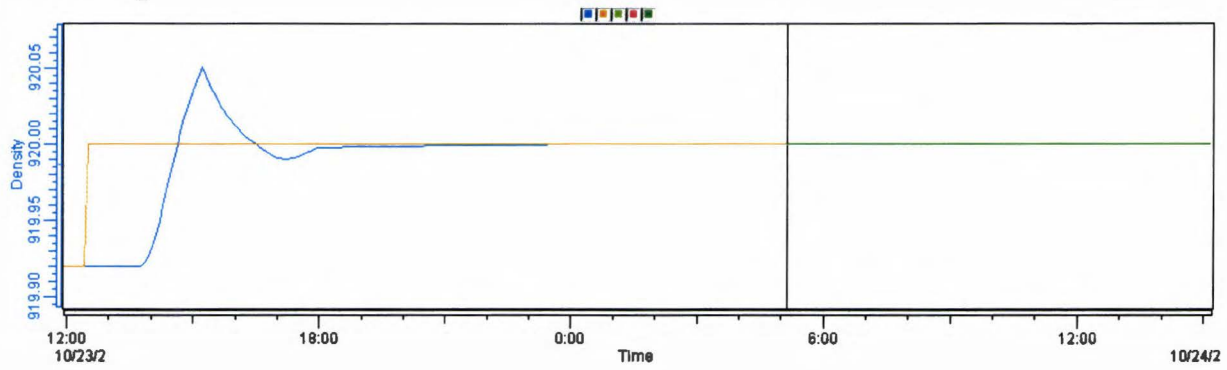

Figure 5.11: *Density when switching from grade E to G: reaching the exact ideal values of grade E*

The sharp peak of the overshoot is probably due to sudden discontinuities in the simulation. The delay in the beginning of the figure is a physical delay: when an input is changed, it takes a while before the influence of this change is noticeable at the end of the reactor where the density is determined. As already told before, the inputs of the process are being changed by the MPC controller directly after the change of the density's ideal value.

After the value of the density and meltindex are settled at the new ideal values, the model used to control the process is changed. The new model loaded is the model for cluster 1. After the model is initialised and the controller is automatically switched on again, new tuning parameters are loaded. This is done because the tuning parameters used for grade E are not good enough. Think for example of the boundaries for each MV and CV. After the model and tuning parameters for cluster 1 are loaded, the ideal values of the density and meltindex are changed again. The new ideal value of the density is 922 $\left[\frac{kg}{m^3}\right]$, and the new value for the meltindex is 2.2 $\left[\frac{g}{10\ min}\right]$. These values are not the values belonging to the target grade G, but it is a step in the right direction.

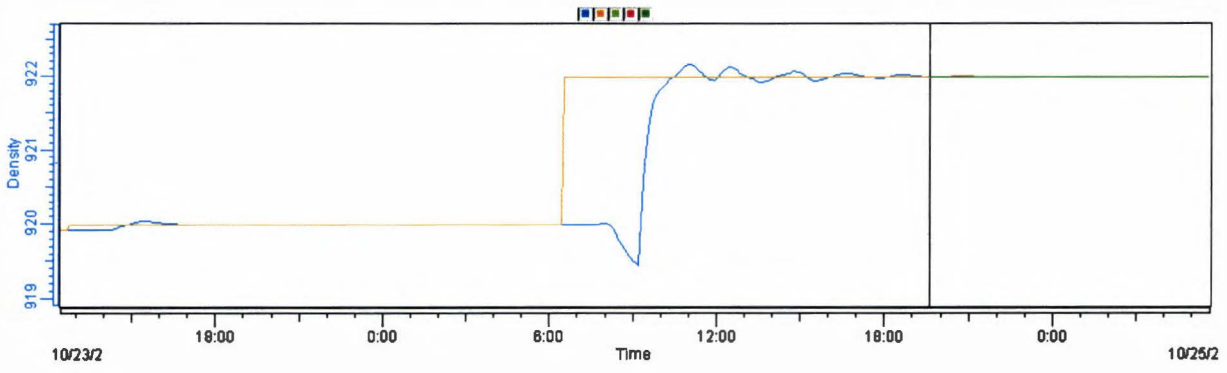

Figure 5.12: *Density when switching from grade E to G: reaching grade B using cluster model 1*

In figure 5.12 can be seen that the new ideal value of the density is reached in 3 to 4 hours. At that moment, the value oscillates round the ideal value. This is not bad because it is not the end ideal value of the density (Note that normally oscillations should be prevented as good as possible). Again, the sharp peak is caused by discontinuities in solving the differential equations of the simulation. In figure 5.13 the meltindex can be seen as it reaches the new ideal value. Also this new ideal is reached in 3 to 4 hours and afterwards oscillates a little bit.

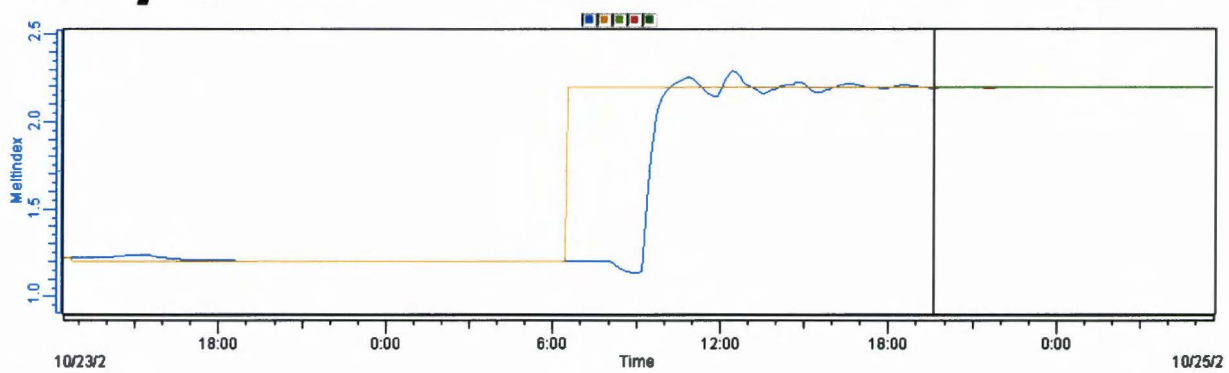

Figure 5.13: *Meltindex when switching from grade E to G: reaching grade B using cluster model 1*

As soon as the density is 922 $\left[\frac{kg}{m^3}\right]$ and the meltindex is 2.2 $\left[\frac{g}{10\ min}\right]$, grade B is reached. Grade B lays in cluster 2 together with grade G. Because we finally want to reach grade G, the cluster model 2 is loaded. After the model is initialised and the controller is switched on again, the matching tuning parameters are loaded. With the model cluster 2 to control the process, the ideal values are changed. The density's value should increase to 923.5 $\left[\frac{kg}{m^3}\right]$ and the meltindex should increase to 2.8 $\left[\frac{g}{10\ min}\right]$ to reach product grade G. Figure 5.14 shows the density during this last part of the way to grade G.

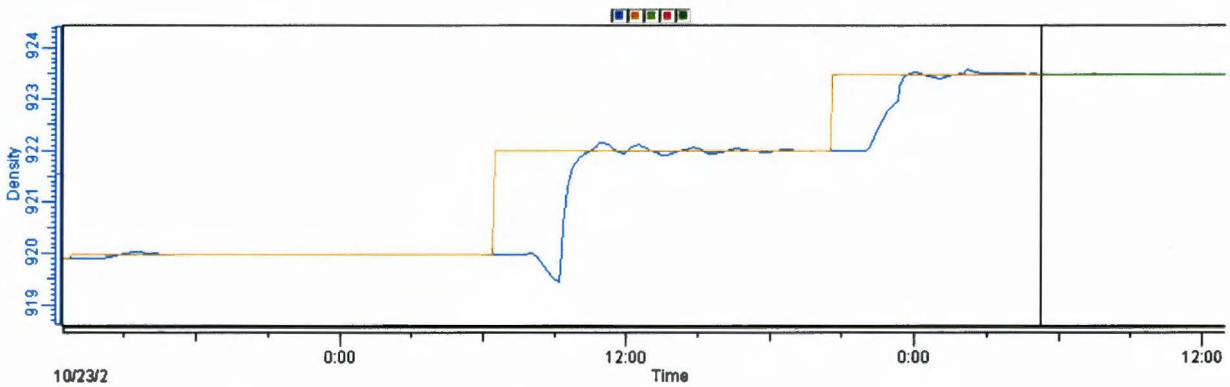

Figure 5.14: *Density when switching from grade E to G: reaching grade G using cluster model 2*

It can be seen that the value of the density belonging to grade G is reached in approximately 3 hours. Figure 5.15 shows the meltindex during the last part of reaching grade G. It takes approximately 5 to 6 hours before the right value is reached.

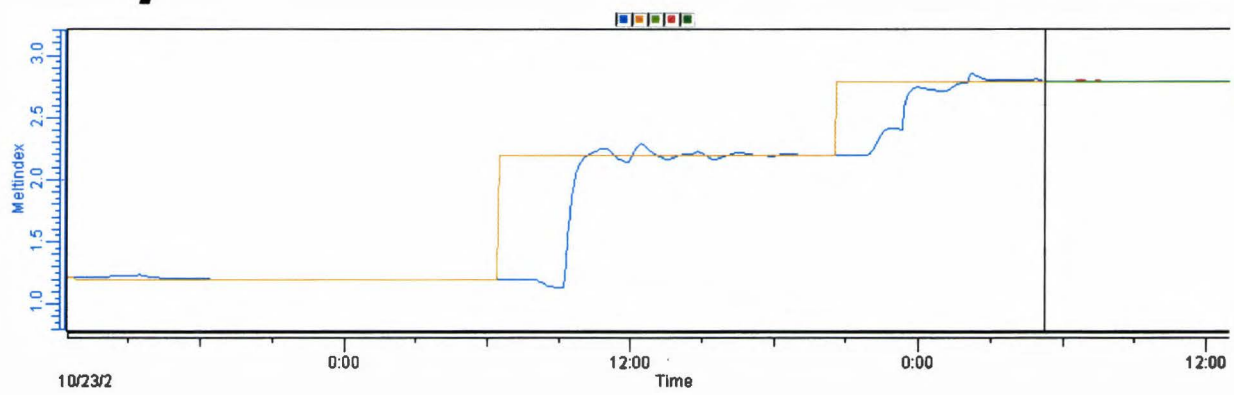

Figure 5.15: *Meltindex when switching from grade E to G: reaching grade G using cluster model 2*

5.4.2 Fast Switching between Grades

When looking at the figures 5.14 and 5.15, it seems to take 1 day and 17 hours to reach the new product grade. However, these figures may not be used for determining the switching time. Switching to another model and entering new ideal values can be done much earlier and through that much faster. The process must not necessary be in steady state before another model with accompanying tuning parameters are loaded. This fast-switching-procedure must be done automatically. An operator could possibly be too late to switch to another model. It can be done automatically by the usage of a switching configuration file which contains values of for example the density and meltindex at which another model must be loaded. Such a configuration file must contain the variables that are being used as switching-base. A lower and an upper switching-boundary should be defined (i.e. a hysteresis should be build in). Of course the file must contain the name of the model and belonging tuning parameters-file to switch to.

Unfortunately the INCA environment is not able yet to load new models and tuning parameters by itself. The fastest change of product grade (from E to G), when changing models and tuning parameters by hand, is done in a bit more than 16 hours. This can be seen in figure 5.16 and 5.17.

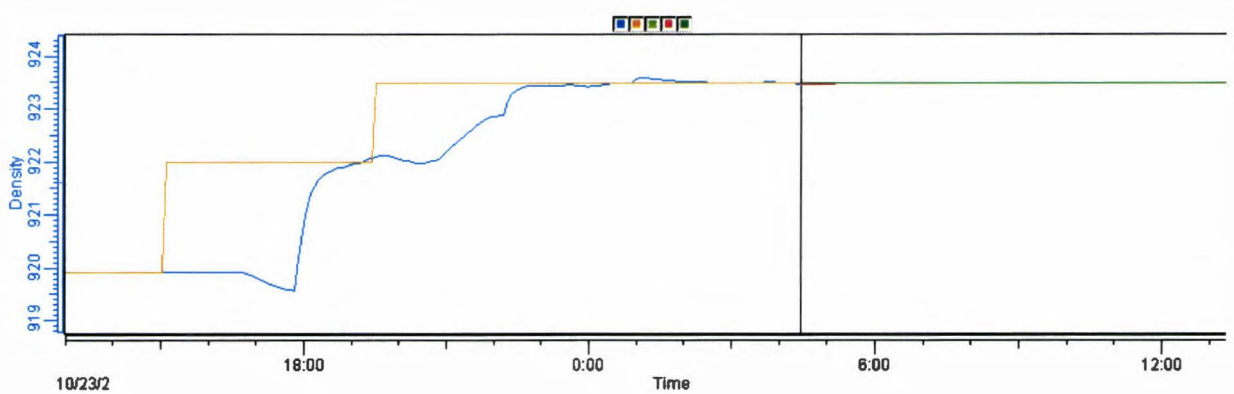

Figure 5.16: *Density when switching from grade E to grade G as fast as possible with multiple models*

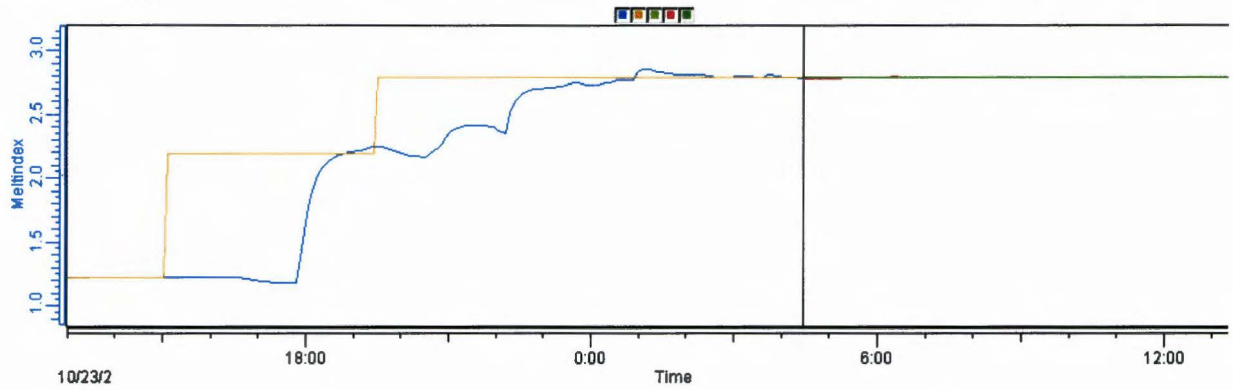

Figure 5.17: *Meltindex when switching from grade E to grade G as fast as possible with multiple models*

Of course in practice there will not only be switched from grade E to G, but also back from G to E. As can be seen in figure 5.18 and 5.19 the time needed for this switching is approximately the same as needed in figure 5.16 and 5.17 where the switching from E to G can be seen. In eight hours the new ideal value for both the meltindex and the density is reached, but it takes a couple hours more to settle these values. In approximately 15 to 16 hours, the product belonging to grade E is produced again with almost no off-spec.

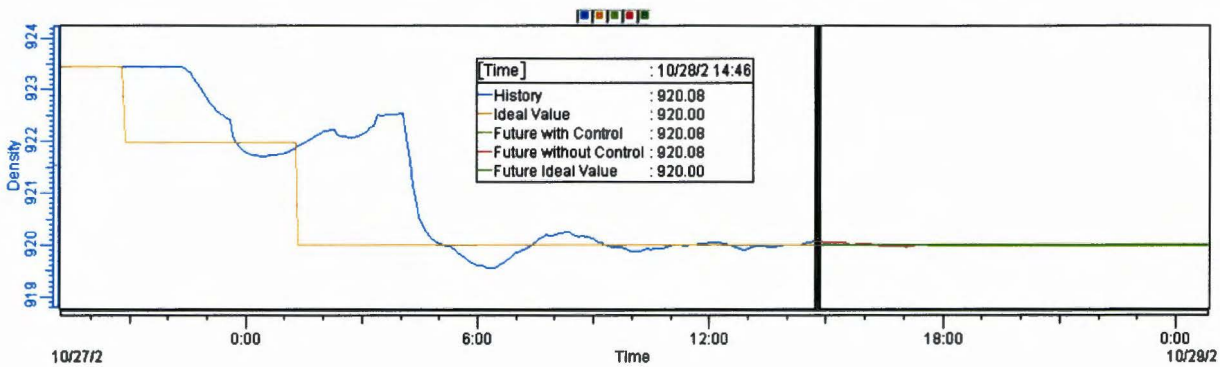

Figure 5.18: *Density when switching from grade G to grade E as fast as possible with multiple models*

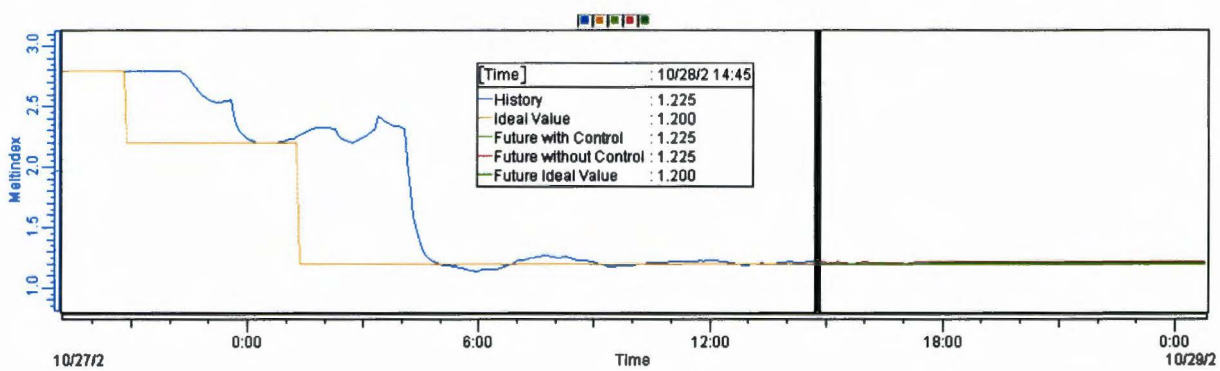

Figure 5.19: *Meltindex when switching from grade G to grade E as fast as possible with multiple models*

5.4.3 Why Multi-Modelling?

When reading this, the reader could think why use multiple models, can it not be done with only one global model and what is the performance when doing so? It is decided to use cluster model 2 to test this situation. Therefore the tuning parameters (especially the boundaries) of the CV's should be edited. Else the density and the meltindex are being clipped at their upper or lower boundaries. Figure 5.20 shows the result of switching from grade E to G with only one model.

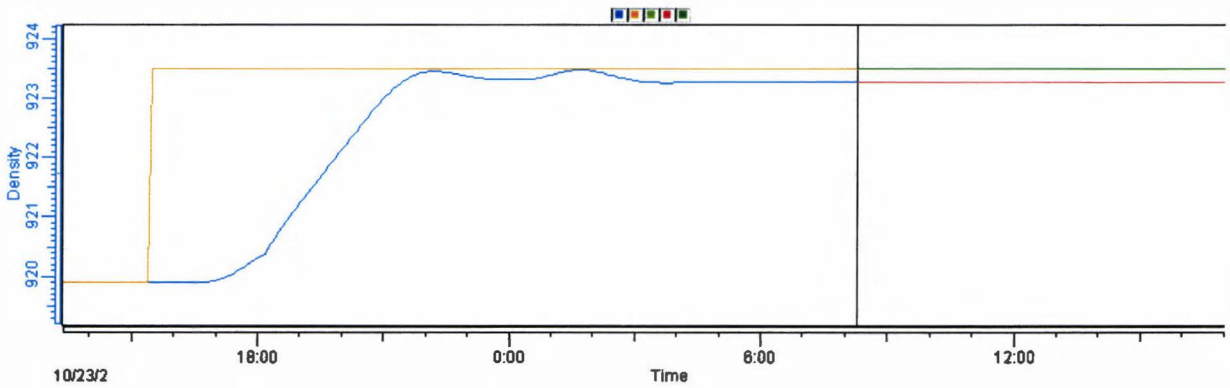

Figure 5.20: *Density when switching from grade E to grade G as fast as possible using only model C2*

As can be seen in the above figure, the new ideal value of the density is not reached. Also the new ideal value of the meltindex is not reached. The cause of this is the lack of freedom of the controller. Several MV's are clipped at their operator boundaries. Figure 5.21 shows the MV jacket temperature zone 2 which is clipped at its upper boundary.

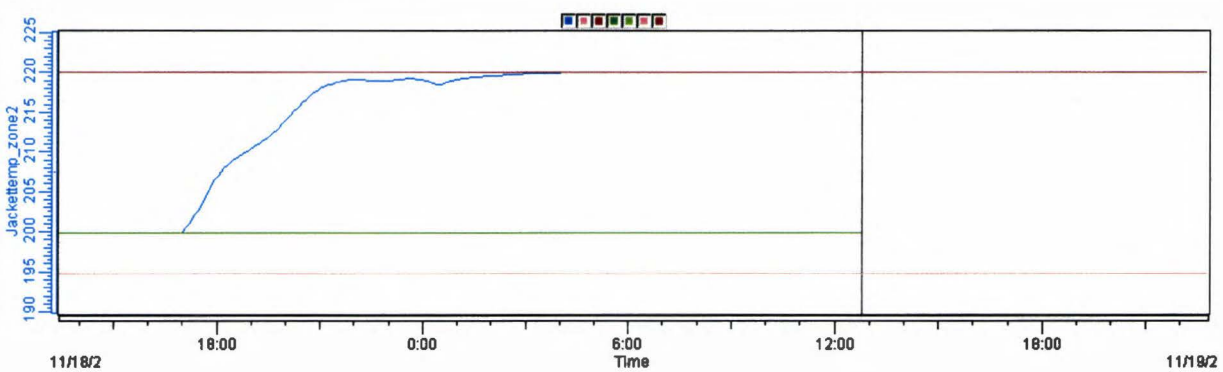

Figure 5.21: *The MV jacket temperature zone 2 is clipped when switching from grade E to G with only one model*

When giving the controller more freedom by changing these MV boundaries, grade G can be reached. However, giving a controller more freedom is in practice not always possible (think of the MV boundaries which cannot always be made larger and larger because of safety reasons for example). Besides, the output density is not the only output with a steady state failure. Also the output meltindex has not reached the new ideal value. This can be seen in figure 5.22.

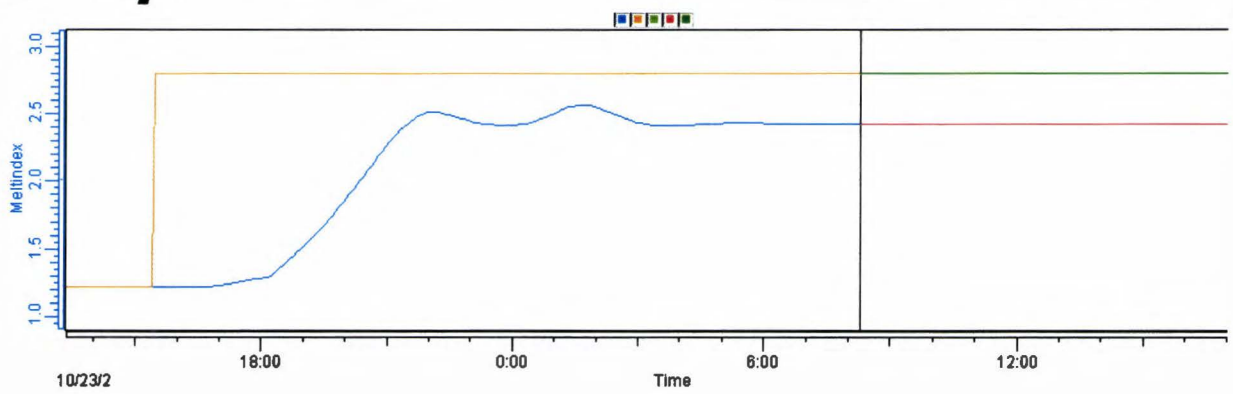

Figure 5.22: *Meltindex when switching from grade E to grade G as fast as possible using only model C2*

5.4.4 Other Grade Changes

In this master thesis, only the switching (using multiple models) between product grade E and G is discussed. Of course also other product changes are possible and a possible change strategy for it can be developed. This will not be completely worked out in this report, but the method to do so will be described now.

When the process is in a certain grade and is controlled by the controller belonging to that grade, one can switch to a grade in the same cluster or to a grade in another cluster. When switching to a grade in the same cluster, it is not always necessary to load the cluster model. The cluster model and the models of the grades in that cluster are almost equal concerning their global behaviour. It is also not always necessary to load (or change by hand) other tuning parameters. However, the performance of the controller, during and after the switching, can be improved when changing tuning parameters. Some tests need to be done in order to determine which model should be chosen and how tuning parameters should be changed.

When switching to a grade in another cluster, it is almost impossible to do so without changing the current model. The current model, used to control the process in the current grade, is changed to the model belonging to the cluster that encloses the current grade. That cluster model describes a larger operating range and when loading the belonging tuning parameters, the controller is more robust. Now should be tried if that controller is able to bring the process to the wanted product grade. When the wanted grade can be reached with the single cluster model, the switching strategy is found. As soon as the wanted grade is reached, the model and tuning parameters belonging to that grade can be loaded in order to improve the performance of the controller.

When a cluster model is not able to reach the wanted grade (in the case described in section 5.4), more than one cluster model is needed. In that case several intermediate points should defined. Each point once becomes the new setpoint of the controller during the changing procedure. In the case described before, the point (density, meltindex) = (922, 2.2) is such an intermediate point. This point is reached with the first cluster model and from thereon another intermediate point should be reached with another cluster model and tuning parameters. Each intermediate point lays closer to the point of the wanted product grade. Finally the wanted product grade is reached and the model and tuning parameters belonging to that grade, can be loaded.

It is preferable to change a model (and tuning parameters) as less as possible; the less models needed, the better. So an intermediate point should lay at far as possible (expressed in points of density, meltindex) from the current grade or previous intermediate point.

Chapter 6

Conclusions and Recommendations

6.1 Conclusions

Controlling a non-linear model with a model based controller, is not very easy. A non-linear model or several linear models are needed for the controller. In this report the control of a tubular Low-Density Polyethylene reactor, using Model Predictive Control, is discussed. Unfortunately is was not possible to use a real plant to design and test the controller. Therefore a simulation of a reactor in such a plant is used. The simulator has some limitations because of the assumptions that are made. However, the main process behaviour is described by this simulator.

Several kind of polyethylene products can be produced in the reactor. The kind of product is basically determined with two quality parameters: density and meltindex. Both parameters are outputs of the simulator. In this simulation eight grades (a combination of a certain product density and meltindex) for eight different products, are introduced. Per grade a single controller is designed, tested and one of the eight controllers is discussed in this report. The controllers perform well when little steps at the ideal value of the density or meltindex are made. When both quality outputs change at the same time, the performance reduces: overshoot occurs and the settling time becomes larger. In spite of the fact that the outputs are coupled, the controller makes the outputs follow their ideal values.

When noise is added to the process outputs, the controller is not influenced by this kind of disturbance. When also noise is added at some inputs of the process, the performance of the controller decreases. When adding a periodic disturbance with noise, the controller is not able to suppress the periodic part due to the physical delays in the process. When only a part of the periodic disturbance is added to the quality-outputs, the controller is able to correct the deviated values.
Another kind of disturbance is an offset at the output temperature of the produced polymer. This influence from outside has influence on the product quality since the controller will try to bring the temperature back to the ideal value. After a while the process is in steady state again and the process value of the density and meltindex equals the ideal value.

When changing from one product grade to another, one model is not always enough. When two grades lay close to each other (i.e. the behaviour of the process in those grades is not very different), one of the grade models is good enough to reach the other product grade. However, to reach the wanted grade, in some cases the degree of freedom for the controller should be increased or a so called cluster model with belonging tuning parameters should be loaded. This should be researched and tried case by case.

When two product grades lay far away from each other, it is almost impossible to reach the wanted grade without loading one or more other models. Multi-modelling techniques are in that case very useful. By defining intermediate points where the used model should be switched, the wanted grade can be reached. However, if one can do with a single model and maybe different sets of tuning param-

eters, this would be preferable since identifying a model takes some time and will cost the factory a lot of money.

6.2 Recommendations

- A simulator with some extra's could be developed. Influences from outside the reactor could be taken into account. For instance: in summer at noon it is warmer than in the evening and that temperature swing has some influence at the process.
  Also the fact that some kind of backward mixing takes place, can be simulated. In this simulation the chemicals flow in one direction and it is assumed that mixing takes place only in that direction. In practice, mixing will take place in more directions; backward is one of them.
  In the current simulation, the jacket of a zone has a constant temperature over the whole zone. Some kind of first order change of jacket temperature is more realistic.
  As could be read in the report, sometimes sharp peaks occur in several trends. These peaks are probably caused by discontinuities while solving the differential equations of the simulation. Status-signals of the equations solvation can be useful to prevent those discontinuities and sharp peaks.
- Changing a model by hand needs some experience of an operator. The automatic change to another model is preferable. This could take place in two ways: by a configuration file written by an engineer or by automatic recognition that the current model is no longer good enough.
- When changing from a certain product grade to another product grade, an optimal trajectory can be followed. Such a trajectory can be optimal because of as little as possible off-spec product while switching between two products, or by optimal operating conditions. Optimal trajectories could be developed and being loaded in *INCAEngine* when switching to another product.

Bibliography

[Bac92] Backx, A.C.P.M. and A.J.W. van den Boom, *Model Predictive Control (Toegepaste Systeemanalyse dictaat)*. Technical University of Eindhoven, 1992.

[Fal94] Falkus, H.M., *Parametric uncertainty in system identification*. Technical University of Eindhoven, 1994 (dissertation).

[Gup85] Gupta, S.K. and A. Kumar, M.V.G. Krishnamurthy, *"Simulation of tubular low-density polyethylene reactors"*. Polymer Engineering and Science, Volume 25, 1985, Issue 1, page 37 - 47.

[Hen02a] Hennissen, J.J.M., *Analysis Fortran code Tubular LDPE reactor*. Boxtel: IPCOS Technology BV, 2002.

[Hen02b] Hennissen, J.J.M., *Handleiding simulator Tubular LDPE reactor*. Boxtel: IPCOS Technology BV, 2002.

[IPC01] IPCOS Technology BV, *User Manual*, Release 2.1, Boxtel: IPCOS Technology BV, 2001.

[Kip94] Kiparissides, C. and G. Verros, A. Pertsinidis, I. Goossens *On-line parameter estimation n a high-pressure low-density polyethylene tubular reactor.*Process Systems Engineering, AIChE Journal, Volume 42, 1996, Issue 2, page 440 - 454.

[Kip93] Kiparissides, C. and G. Verros, J.F. MacGregor, *Mathematical modeling, optimization, and quality control of high-pressure ethylene polymerization reactors*. Journal of Macromolecular science-reviews in macromolecular chemistry and physics, Volume 33, 1993, Issue 4, page 437 - 527.

Appendix A

Literature Search Report

This appendix contains the literature search report written by me in order to end the Documentary Information Course (5J053) successfully.

1. Assignment Description

1.1 Formulation of the problem

IPCOS Technology provides model predictive control technology with its INCA software package and engineering services for the process industry, with the emphasis on the glass and chemical industry. Within the ongoing research projects for the polymer industry, IPCOS is developing a rigorous non-linear simulator for the free radical tubular LDPE (Low Density Polyethylene) process to show the advantages and benefits of multi-model and trajectory control with INCA. For the tubular process an initial simulation model in Fortran sourcecode is available.

The master thesis consists of the following phases:

- Implementing the tubular simulation model in C++ using the Fortran code as a basis. The conversion is required to optimize simulation speed and to benefit of already existing C++ communication and numerical libraries.
- Developing a control strategy with INCA around a fixed operating condition of the tubular process. Using a single linear model for control of the non-linear tubular simulation will result in a trade-off between robustness and accurancy.
- To increase the control performance several linear models will be needed to operate the non-linear tubular process over a broad working range. A strategy to switch between linear models depending on the operating conditions must be developed.
- Main benefits can be achieved in practice when switching between operating conditions is realized via optimal trajectories. These trajectories can be calculated by optimizing a process cost function. Combining trajectory control which typically covers a broad working range and using time variant models in the model predictive controller should realize optimal process control.

1.2 Nature and scope of the literature search

In my master thesis the scope of the literature search will be to find out how to model a free radical tubular LDPE process. This search needs to be done in order to understand the main literature search better and in order to find out if the model IPCOS already has (used in the seventies) is still valid and useful nowadays.

However, the main part is to find (several) way(s) to control a LDPE process using Model Predictive Control (MPC) techniques. For instance: which input(s) are more sensitive to the process than others and how controlling should lead to the optimal output of the plant.

2. Concept Index of the Graduation Report

In this chapter the concept index of the master thesis report (which will be written in the near future) can be found. Only the (temporary) main parts of the index are listed below (of course the thesis will also contain a preface, a summary and so on).

- Implementation of the simulation model
- Control strategy for the tubular process
- Increasing control performance due to several models
- Introducing trajectories for controlling the tubular reactor

3. List of search Terms

For the literature search the search terms listed below were used. These search terms are all in English, because they all are names of techniques or applications, which are difficult to translate in Dutch and are not often used in Dutch.

	English	Dutch
1.	Tubular	
2.	(Free) Radical	
3.	Polyethylene	Polyethyleen
4.	LDPE	
5.	Reactor	Reaktor
6.	Plant	
7.	Control	Regel(techniek)
8.	Model	Model
9.	Predictive	Voorspellend
10.	Trajectory	

It must be noted that some search terms are (almost) the same, such as number 3 and 4. Unfortunately the usage of these terms can lead to different search results and therefore both terms are used in my literature search. Also some search terms (model, predictive, control) could be packed together, but are used alone to generate more search results (hopefully!).

4. Literature Sources

4.1 Used sources

Table A.1: *Used sources and their searched period*

Soure name	Source medium	Searched period
Library Catalog TU/e	Online (VubisWEB)	All data available
INSPEC	Online (INSPEC via WebSPIRS)	1985 — april 2002
IEE Library	Online (IEEE Xplore)	1988 - april 2002
EI Compendex	Online (via Elsevier ScienceDirect)	1967 - april 2002

The reason for using these sources is the following:

- for Vubis(Web): this source medium contains references to all literature available at the TU/e, both physical and electronical.
- for INSPEC and IEEE Xplore: these source media are both intended to be used for users in the field of electrical and information technological engineering and to cover publications on these fields of interest. Another advantage of these sources is the fact that they cover recent publications as well.
- for Compendex: this source medium is not only intended to be used for users in the field of electrical and information technological engineering, but is also very usefull in the field of chemical engineering where for instance a LDPE reactor is part of. Both INSPEC and (even more) Compendex do cover control theory and engineering very well.

4.2 Search results

The sources mentioned in paragraph 4.1 above were used with the search terms mentioned earlier in paragraphs 3.1 and 3.2. This gave the results listed in Table A.2.

Table A.2: *Number of found results per source and search term*

Search term(s)	VUBIS	INSPEC	IEEE Xplore	Compendex
1. Tubular	62	3305	285	Too many
2. Free Radical	11	3111	133	Too many
3. Polyethylene	93	12429	1664	Too many
4. LDPE	2	993	369	Too many
5. Reactor	479	106031	5107	Too many
6. Plant	512	64624	9449	Too many
7. Control	5297	578066	138425	Too many
8. Model	2689	867453	116804	Too many
9. Predictive	60	18268	5315	Too many
10. Trajectory	34	22380	4805	Too many
1 + 3	0	37	2	210
1 + 5	3	462	23	1787
1 + 6	0	90	10	526
7 + 8	22	108261	36386	Too many
7, 8 + 9	19	3651	3589 1)	3223
3, 7, 8 + 9	0	17	2	9
4, 7, 8 + 9	0	3	0	2
7, 8, 9 +10	0	160	157	137
5, 7, 8 + 9	0	300	112	236
6, 7, 8 + 9	0	665	348	598
3, 5, 7, 8, 9 + 10	0	4	0	1
1, 3, 5, 7, 8, 9 + 10	0	0	0	0

It must be noted that the source Picarta is also used, with the regrettable result of 1 when searching for "Model Predictive Control" and "Polyethylene". Also other search terms did not lead to a satisfied result and therefore this source is not taken down in Table 1 and Table 2. I also tried to use the internet sources Engineering Electronic Library Sweden (EELS) and Edinburgh Engineering Virtual Library (EEVL). Unfortunately, the results of these sources couldn't satisfy me.

Only a few books were found using these search terms. Therefore a more general search term was used in VUBIS to find some books that contain general, but also specific information, that I was looking for. The Dutch keyword used is "Procesregeling" (Process Control). This led to 119 search results from which 2 were useful. The Dutch keyword "Procesindustrie" (Process Industry) led to 47 results, but none of them were new for me or interesting enough.

4.3 Selection of search results

Table A.2 shows many search results and even in some cases too many results (especially Compendex is willing to generate too many results). Therefore the search results need to be filtered carefully. I did this by reading the abstracts of each found document and judged it on base of its relevance to my scope of literature search (chapter 1.2). All found literature was written in and after 2001. The found literature is recent enough so no extra filtering, based on year of publication, is needed.

The documents found searching VUBIS as searching source seemed not good enough. Some of the results were already found by another source (for instance INSPEC). Using the links in INSPEC led to references in VUBIS which I didn't find using the search engine of VUBIS. A possible explanation for this is the fact that I searched VUBIS in English (some search terms can not be translated) and VUBIS is not so strong at that point yet.

In the previous chapter was already mentioned that using Dutch search terms in VUBIS led to some interesting books. The selected books were all written in English and selected on their content using the index of the books.

The documents found with IEEE Xplore are unfortunately not very interesting. Searching for search term 3, 7, 8 and 9 gave 2 results and only one was interesting (but not very useful). The 10 search results when searching for "Tubular" and "Plant" were all not useful.

The best (and by coincidence also the most) search results were found using INSPEC. Searching for term 4, 7, 8 and 9 gave three useful results. Searching for 3, 7, 8 and 9 gave 17 results, but only 10 are unique and useful.

Also good results came from searching with Compendex. The useful results were all already found by INSPEC, so no extra literature was found using Compendex.

5. Snowball Method

The following publications were selected initially.

1. Cervantes, A., et al., *Large-scale dynamic optimization of a low density polyethylene plant* (6)
2. Xiao, K., et al., *Control of coating properties of LDPE through melt strength measurements* (42)

The reason for selecting these publications is that these publications have a recent publication date (which means these are probably based on most of the publications in this area). Of course the selected publications have a great relevance to the subject of the literature search. In this snowball method, found literature with strong emphasis to the chemical reaction in the reactor, are omitted. Some of that literature is not unimportant, so it can be found in the reference list in chapter 9.

6. Citation Method

The following publications were selected initially.

1. Chen, C. H, et al., *Computer model for tubular high pressure polyethylene reactors* (7)
2. Donati, G, et al., *Mathematical model of low density polyethylene tubular reactor* (10)
3. Gupta, Santosh K, et al., *Simulation of tubular low density polyethylene reactors* (15)

There are two reasons for choosing these documents. The first one is the fact that these documents (which were already found) are one of the oldest documents found. The second reason is the fact that they describe the modelling of a tubular LDPE reactor. The citation method is a perfect way to find out how important a particular document was and maybe still is.

In Figure 2 the results of the citation method can be found. It is clear that used literature for this search was and is still very useful since, it is cited so often.

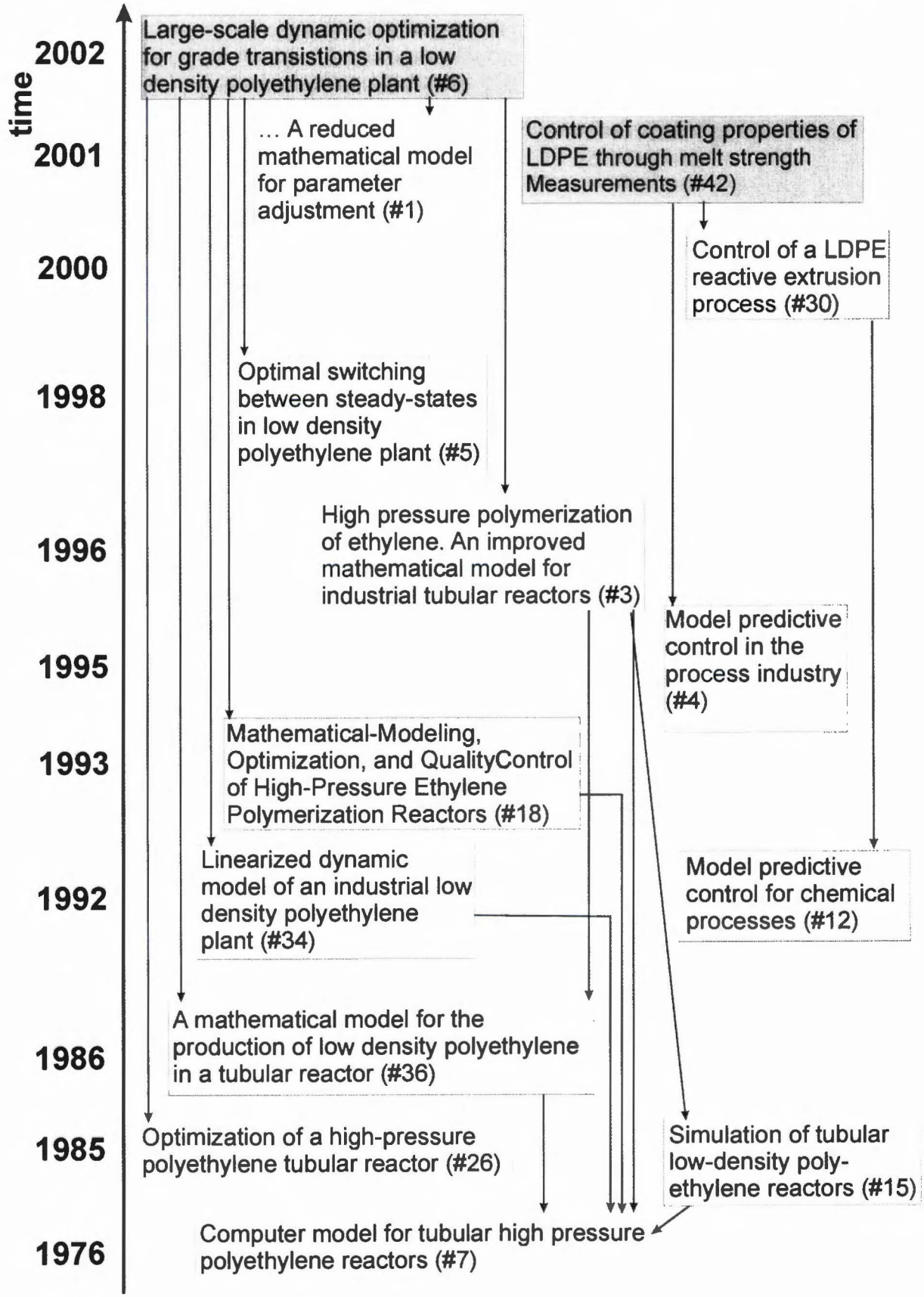

Figure A.1: *Results of snowball method*

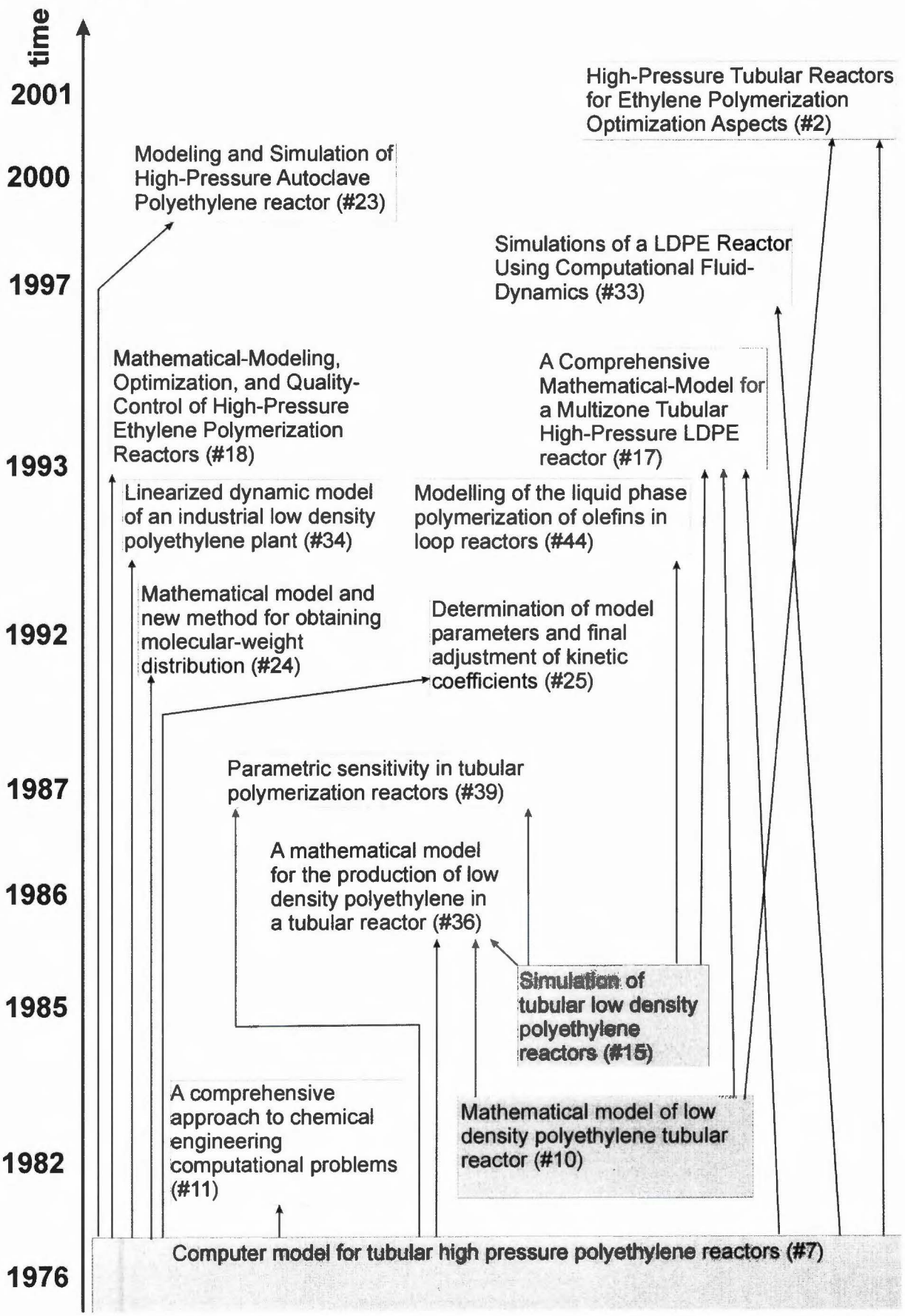

Figure A.2: *Results of citation method*

7. Relation between found Literature and Final Report

In this chapter the relations between the found literature and the concept index will be made clear. This is done by filling a matrix. The rows will be filled with the concept index and the columns will contain the reference numbers.

Table A.3: *Relation between found literature and concept index*

Subject	R	e	f	e	r	e	n	c	e	s	:							
Implementation of the simulation model	7	15	36	13		24	25	34	42	9		3	32					
Control strategy the tubular process					14	12					4		32			30		42
Increasing control performance due to models		26								19		20				6		
Introducing trajectories for controlling the tubular reactor						27				9				38	5	40	41	
Year	76	85	86	88	89	92	92	93	93	94	95	96	96	97	98	00	00	01

8. Conclusions and Recommendations

At the end of this literature search report, some conclusions can be drawn. The first conclusion is that the model described in literature number 7 is this referenced in today's literature. This can easily be seen when looking at the results of the citation method shown in Figure 2. Using that computer model for a simulation of a tubular Low Density Polyethylene (LDPE) reactor, gives me a good feeling because this model is still referenced today. Of some modifications are done in the past years which should be taken into account when implementing a simulation environment.

Also many literature about the control of a tubular LDPE reactor is found. Most of this literature describes the control method and the results when other (or none) control techniques. Up till now no document was found which described precisely how to control such a reactor (i.e. which inputs to change when a certain change of output(s) is desired). When looking for literature describing the control of the reactor, also literature about trajectories and grade transitions (changing the circumstances of the reactor in such a way that another wanted output is received) is found. That literature can be very useful for me when implementing grade transitions during my thesis.

Most found literature was found using INSPEC. Most of the found publications were part of American Institute of Chemical Engineering (AIChE) publications. Books were found using VUBIS as search engine. From all books found, only 2 were good enough and not too general.

An author which was (and still is) very important in this kind of research is C. Kaprissides. In the bibliography of this literature search (see chapter 9) this author is referenced 4 times with publications of his own and once as a co-author (reference number 26). Other authors that were (and are) important are S.A. Nield, A. Cervantes and A. Brandolin.

9. References

1. Asteasuain, M. and S. Pereda, M.H. Lacunza, P.E. Ugrin, A. Brandolin, *Industrial High-Pressure*

Ethylene Polymerization Initiated by Peroxide Mixtures - A Reduced Mathematical-Model for Parameter Adjustment. Polymer engineering and science, Volume 41, 2001, Issue 5, page 711 - 726.

2. Brandolin, A. and E.M. Valles, J.N. Farber, *High-Pressure Tubular Reactors for Ethylene Polymerization Optimization Aspects.* Polymer engineering and science, Volume 31, 1991, Issue 5, page 381 - 390.
3. Brandolin, A. and M.H. Lacunza, P.E. Ugrin, N.J. Capiati, *High pressure polymerization of ethylene. An improved mathematical model for industrial tubular reactors.* Polymer Reaction Engineering, Volume 4, 1996, page 193 - 241.
4. Camacho,E.F. and. C. Bordons, *Model predictive control in the process industry.* New York: Springer, 1995.
5. Cervantes, A. and S. Tonelli, A. Bandoni, L. Biegler, *Optimal switching between steady-states in low density polyethylene plant.* Miami: American Institute of Chemical Engineering (AIChE), meeting, 1998, page 263.
6. Cervantes, A. and S. Tonelli, A. Brandolin, A. Bandoni, L. Biegler, *Large-scale dynamic optimization for grade transistions in a low density polyethylene plant.* Polymer Engineering ans Science, Volume 5, 2002, page 711 - 726.
7. Chen, C. H. and J.G. Vermeychuk, J.A. Howell, P. Ehrlich, *Computer model for tubular high pressure polyethylene reactors.* AIChE Journal, Volume 22, 1976, Issue 3, page 463 - 469.
8. Curry, J., S. Jackson, B. Stöhrer, A. van der Veen, *Free radical degradation of polypropylene.* Chemical Engineering Progress, 1988, page 43 - 46.
9. Debling, J. A. and G.C. Han, F. Kuijpers, J. Verburg, J. Zacca, W.H. Ray, *Dynamic modeling of product grade transitions for olefin polymerization processes.* American Institute of Chemical Engineering Journal, Volume 40, 1994, 506 - 520.
10. Donati, G. and L. Marini, G. Marziano, C. Mazzaferri, M. Spampinato, E. Langianni, *Mathematical model of low density polyethylene tubular reactor.* ACS Symposium Series, 1982, page 579 - 590.
11. Donati, G. and L. Marini, G. L. Marziano, *A comprehensive approach to chemical engineering computational problems.* Chemical Engineering Science, Volume 37, 1982, Issue 8, page 1252 - 1282.
12. Eaton, J. W. and J.B. Rawlings, *Model predictive control of chemical processes.* Chemical Engineering Science, Volume 47, 1992, page 705 - 720.
13. Feyo De Azevedo, S. and J.A. Howell, *Second order model for low density polyethylene pipeline reactors.* Chemical Engineering Research and Design, Volume 66, 1988, Issue 2, page 128 - 137.
14. Garcia, C. E. and D.M. Prett, M. Morari, *Model predictive control: Theory and practice.* Automatica, Volume 25, 1989, page 335 - 348.
15. Gupta, S. K. and A. Kumar, M.V.G. Krishnamurthy, *Simulation of tubular low-density polyethylene reactors.* Polymer Engineering and Science, Volume 25, 1985, Issue 1, page 37 - 47.
16. Kalyon, D.M. and Y.N. Chiou, S. Kovenklioglu, A. Bouaffar, *High-Pressure Polymerization of Ethylene and Rheological Behavior of Polyethylene Product.* Polymer engineering and science, Volume 34, 1994, Issue 10, page 804 - 814.
17. Kiparissides, C. and G. Verros, G. Kalfas, M. Koutoudi, C. Kantzia, *A Comprehensive Mathematical-Model for a Multizone Tubular High-Pressure LDPE Reactor.* Chemical Engineering Communications, Volume 121, 1993, page 193 - 217.
18. Kiparissides, C. and G. Verros, J.F. Macgregor, *Mathematical-Modeling, Optimization, and Quality-Control of High-Pressure Ethylene Polymerization Reactors.* Journal of Macromolecular science-reviews in macromolecular chemistry and physics, Volume 33, 1993, Issue 4, page 437 - 527.

19. Kiparissides, C. and G. Verros, A. Pertsinidis, *On-line optimization of a high-pressure low-density polyethylene tubular reactor.* Chemical Engineering Science, Volume 49, 1994, Issue 24, page 5011 - 5024.

20. Kiparissides, C. and G. Verros, A. Pertsinidis, I. Goossens, *On-line parameter estimation in a high-pressure low-density polyethylene tubular reactor.* AIChE Journal, Volume 42, 1996, Issue 2, page 440 - 454.

21. Kwag, B.G. and K.Y. Choi, *Effect of Initiator Characteristics on High-Pressure Ethylene Polymerization in Autoclave Reactors.* Industrial and engineering chemistry research, Volume 33, 1994, Issue 2, page 211 - 217.

22. Lacunza, M.H. and P.E. Ugrin, A. Brandolin, N.J. Capiati, *Heat-Transfer Coefficient in a High-Pressure Tubular Reactor for Ethylene Polymerization.* Polymer engineering and science, Volume 38, 1998, Issue 6, page 992 - 1013.

23. Lee, H.J. and Y.K. Yeo, *Modeling and Simulation of High-Pressure Autoclave Polyethylene Reactor Including Decomposition Phenomena.* Journal of Engineering of Japan, Volume 33, 2000, Issue 2, page 323 - 329.

24. Lorenzini P. and M. Pons, J. Villermaux, *Free-radical polymerization engineering III. Modelling homogeneous polymerization of ethylene: mathematical model and new method for obtaining molecular-weight distribution.* Chemical Engineering Science, Volume 47, 1992, Issue 15-16, page 3969 - 3980.

25. Lorenzini, P. and M. Pons, J. Villermaux, *Free-radical polymerization engineering IV. Modelling homogeneous polymerization of ethylene: determination of model parameters and final adjustment of kinetic coefficients.* Chemical Engineering Science, Volume 47, 1992, Issue 15-16, page 3981 - 3988.

26. Mavridis, H. and C. Kiparissides, *Optimization of a high-pressure polyethylene tubular reactor.* Polymer Process Engineering, Volume 3, 1985, Issue 3, page 263 - 290.

27. McAuley, K. B. and J.F. MacGregor, *Optimal grade transitions in a gas phase polyethylene reactors.* American Institute of Chemical Engineering (AIChE) Journal, Volume 38, 1992, page 1564 - 1576.

28. Nield, S.A., *Peroxide initiated modification of low density polyethylene.* Waterloo: University of Waterloo, Ontario, Canada, 1997 (Master Thesis).

29. Nield, S.A. and C. Tzoganakis, H. Budman, *Chemical modification of LDPE through reactive extrusion: Part I:Process development and product characterization.* Advances in Polymer Technology, Volume 19, 2000, page 237 - 248.

30. Nield, S.A. and C. Tzoganakis, H. Budman, *Control of a LDPE reactive extrusion process.* Control Engineering Practice, Volume 8, 2000, page 911 - 920.

31. Ogunnaike, B.A. and W.H. Ray, *Process dynamics, modeling, and control.* Oxford : Oxford University Press, 1994.

32. Ray, W. H., *Modeling and control of polymerization processes.* Chicago: American Institute of Chemical Engineers, meeting, 1996, page 176.

33. Read, N.K. and S.X. Zhang-SX, W.H. Ray, *Simulations of a LDPE Reactor Using Computational Fluid-Dynamics.* LES, Volume 43, 1997, Issue 1, page 104 - 117.

34. Schbib, S. and S. Tonelli, E. Brignole, J. Romagnoli, *Linearized dynamic model of an industrial low density polyethylene plant.* Computers and Chemical Engineering, Volume 17, 1993, page 323 - 328.

35. Scholsky, K. M., *Polymerization reactions at high pressure and supercritical conditions.* Journal of Supercritical Fluids, Volume 6, 1993, page 103 - 127.

36. Shirodkar, P.P. and G. O. Tsien, A mathematical model for the production of low density polyethylene in a tubular reactor. Chemical Engineering Science, Volume 41, 1986, Issue 4, page 1031 - 1038.

37. Stephanopoulos, G., Chemical process control : an introduction to theory and practice. Englewood Cliffs : Prentice-Hall, 1984.

38. Takeda, M., *Optimal grade transition strategies for multi-stage olefin polymerization reactors with bimodal products.* Wisconsin: University of Wisconsin, 1997 (Master Thesis).

39. Tjahjadi, M. and S.K. Gupta, M. Morbidelli, A. Varma, *Parametric sensitivity in tubular polymerization reactors.* Chemical Engineering Science, Volume 42, 1987, Issue 10, page 2385 - 2394.

40. Wang, Y. and H. Seki, S. Ohyama, K. Akamatsu, M. Ogawa, M. Ohshima, *A nonlinear predictive control for optimal grade transition of polymerization reactors.* Pisa: AD-ChEM, 2000.

41. Wang, Y. and H. Seki, S. Ohyama, K. Akamatsu, M. Ogawa, M. Ohshima, *Optimal grade transition control for polymerization reactors.* Computers and Chemical Engineering, Volume 24, 2000, page 1555 - 1561.

42. Xiao, K. and C. Tzoganakis, H. Budman, *Control of coating properties of LDPE through melt strength Measurements.* Control Engineering Practice, Volume 9, 2001, page 357 - 366.

43. Zabisky, R.C.M. and W.M. Chan, P.E. Gloor, A.E. Hamielec, *A Kinetic-Model for Olefin Polymerization in High-Pressure Tubular Reactors - A Review and Update.* Polymer, Volume 33, 1992, Issue 11, page 2243 - 2262.

44. Zacca, J.J. and W. Harmon Ray, *Modelling of the liquid phase polymerization of olefins in loop reactors.* Chemical Engineering Science, Volume 48, 1993, Issue 22, page 3743 - 3765.

Appendix B

Definition of Operating Points and Ranges

In this chapter the definition of the operating points and belonging ranges per product grade will be given.

Table B.1: *Definition of operating point A*

Inputs	Lower boundary	Ideal value	Upper boundary	Unit
Initiator feed zone 1	0.095	0.1	0.12	$\frac{l}{hour}$
Initiator feed zone 2	1.8	2.1	2.5	$\frac{l}{hour}$
Feed temperature zone 1	20	24	65	°C
Feed temperature zone 2	112	120	123	°C
Jacket temperature zone 1	206	210	220	°C
Jacket temperature zone 2	190	200	210	°C
Jacket temperature zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	182.5	232.6	248	meter
Temperature peak zone 1	165.3	267.6	272.7	°C
Place peak zone 2	373.4	368.6	419.7	meter
Temperature peak zone 2	302.7	305.8	309.3	°C
Output temperature	230.3	236.5	242.9	°C
Density polymer	919.5	921.1	922.2	$\frac{gram}{10\ min}$
Meltindex polymer	0.652	0.796	1.014	$\frac{kg}{m^3}$

Table B.2: *Definition of operating point B*

Inputs	Lower boundary	Ideal value	Upper boundary	Unit
Initiator feed zone 1	0.15	0.2	0.26	$\frac{l}{hour}$
Initiator feed zone 2	2.8	3.5	4.6	$\frac{l}{hour}$
Feed temperature zone 1	30	60	99	°C
Feed temperature zone 2	55	60	65	°C
Jacket temperature zone 1	208	210	212	°C
Jacket temperature zone 2	190	200	220	°C
Jacket temperature zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	122.5	167.1	200.7	meter
Temperature peak zone 1	279.7	285.2	291.1	°C
Place peak zone 2	494.6	559.1	608.1	meter
Temperature peak zone 2	312.8	316.7	322.2	°C
Output temperature	254	262.4	270	°C
Density polymer	920.6	921.8	923.4	$\frac{gram}{10\ min}$
Meltindex polymer	1.576	2.204	3.461	$\frac{kg}{m^3}$

Table B.3: *Definition of operating point C*

Inputs	Lower boundary	Ideal value	Upper boundary	Unit
Initiator feed zone 1	0.24	0.3	0.34	$\frac{l}{hour}$
Initiator feed zone 2	3.5	4.8	5.1	$\frac{l}{hour}$
Feed temperature zone 1	60	85	105	°C
Feed temperature zone 2	101	105	109	°C
Jacket temperature zone 1	208	210	212	°C
Jacket temperature zone 2	190	200	210	°C
Jacket temperature zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	106.9	129.2	156.6	meter
Temperature peak zone 1	292.3	296.7	300	°C
Place peak zone 2	394.9	417.1	439.4	meter
Temperature peak zone 2	317.9	324.4	325.7	°C
Output temperature	242.5	248.7	254.9	°C
Density polymer	922.3	924.2	925.1	$\frac{gram}{10\ min}$
Meltindex polymer	2.371	4.004	4.524	$\frac{kg}{m^3}$

Table B.4: *Definition of operating point D*

Inputs	Lower boundary	Ideal value	Upper boundary	Unit
Initiator feed zone 1	0.34	0.6	0.71	$\frac{l}{hour}$
Initiator feed zone 2	1.9	2.1	2.8	$\frac{l}{hour}$
Feed temperature zone 1	35	60	80	°C
Feed temperature zone 2	55	60	65	°C
Jacket temperature zone 1	208	210	212	°C
Jacket temperature zone 2	194	200	210	°C
Jacket temperature zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	120.7	141.3	166.8	meter
Temperature peak zone 1	295.3	306.3	309.6	°C
Place peak zone 2	543.1	584.3	615.9	meter
Temperature peak zone 2	303.8	305.7	311.5	°C
Output temperature	254	259.9	265.9	°C
Density polymer	919.6	920.5	921.9	$\frac{gram}{10\ min}$
Meltindex polymer	1.150	1.318	1.892	$\frac{kg}{m^3}$

Table B.5: *Definition of operating point E*

Inputs	Lower boundary	Ideal value	Upper boundary	Unit
Initiator feed zone 1	0.21	0.35	0.5	$\frac{l}{hour}$
Initiator feed zone 2	2	2.2	3.5	$\frac{l}{hour}$
Feed temperature zone 1	25	60	80	°C
Feed temperature zone 2	51	60	70	°C
Jacket temperature zone 1	207	210	213	°C
Jacket temperature zone 2	195	200	220	°C
Jacket temperature zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	131.4	152.9	190.3	meter
Temperature peak zone 1	286.2	295.9	302.74	°C
Place peak zone 2	514.9	588.4	614.4	meter
Temperature peak zone 2	305.1	307	316.3	°C
Output temperature	250.8	260.6	266.7	°C
Density polymer	919.3	919.9	922.4	$\frac{gram}{10\ min}$
Meltindex polymer	1.059	1.223	2.333	$\frac{kg}{m^3}$

Table B.6: *Definition of operating point F*

Inputs	Lower boundary	Ideal value	Upper boundary	Unit
Initiator feed zone 1	0.095	0.1	0.12	$\frac{l}{hour}$
Initiator feed zone 2	1.8	2.1	2.5	$\frac{l}{hour}$
Feed temperature zone 1	6	12	16	°C
Feed temperature zone 2	97	100	113	°C
Jacket temperature zone 1	206	210	220	°C
Jacket temperature zone 2	190	200	210	°C
Jacket temperature zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	219.8	242.5	256.2	meter
Temperature peak zone 1	231.6	264.4	269.7	°C
Place peak zone 2	409.4	455.1	527	meter
Temperature peak zone 2	301.9	305	308.5	°C
Output temperature	238.3	244.3	250.3	°C
Density polymer	918.4	920.1	921.1	$\frac{gram}{10\ min}$
Meltindex polymer	0.647	0.787	0.996	$\frac{kg}{m^3}$

Table B.7: *Definition of operating point G*

Inputs	Lower boundary	Ideal value	Upper boundary	Unit
Initiator feed zone 1	0.55	0.7	0.8	$\frac{l}{hour}$
Initiator feed zone 2	3.4	3.6	4.9	$\frac{l}{hour}$
Feed temperature zone 1	30	60	80	°C
Feed temperature zone 2	55	60	65	°C
Jacket temperature zone 1	208	210	212	°C
Jacket temperature zone 2	195	200	205	°C
Jacket temperature zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	117.9	138.3	168.5	meter
Temperature peak zone 1	304.6	309.3	312.5	°C
Place peak zone 2	511.4	538.9	559.7	meter
Temperature peak zone 2	315	316.4	322.7	°C
Output temperature	256.2	262	267.9	°C
Density polymer	922.4	923.5	925.2	$\frac{gram}{10\ min}$
Meltindex polymer	2.392	2.797	4.511	$\frac{kg}{m^3}$

Table B.8: *Definition of operating point H*

Inputs	Lower boundary	Ideal value	Upper boundary	Unit
Initiator feed zone 1	0.11	0.14	0.17	$\frac{l}{hour}$
Initiator feed zone 2	2.7	3	3.5	$\frac{l}{hour}$
Feed temperature zone 1	10	22	38	°C
Feed temperature zone 2	69	84	98	°C
Jacket temperature zone 1	208	210	212	°C
Jacket temperature zone 2	195	200	205	°C
Jacket temperature zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	203.3	222.1	236.6	meter
Temperature peak zone 1	269.2	273.7	277.3	°C
Place peak zone 2	428.4	475.6	523.9	meter
Temperature peak zone 2	310	312.1	315.2	°C
Output temperature	245.6	251.4	257.3	°C
Density polymer	921.3	921.9	922.8	$\frac{gram}{10\ min}$
Meltindex polymer	1.194	1.389	1.751	$\frac{kg}{m^3}$

Appendix C

Parameters of the used Grade Models

In table C.1 the number of FIR parameters per transfer of the model for grade A can be found. In table C.2 the delays expressed in sample numbers can be found. The '-' in the tables indicates that there is no transfer simulated between that input and output.

For all other grades (B till H), equivalent tables (C.3 till C.16) can be found in this appendix.

Table C.1: *Number of FIR parameters per transfer of the model in grade A*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	10	-	10	-	180	-	-	10	-	5	-
Temperature peak zone 1	10	-	10	-	180	-	-	10	-	5	-
Place peak zone 2	5	10	10	5	180	40	-	5	5	5	5
Temperature peak zone 2	5	10	10	5	180	40	-	5	5	5	5
Output temperature	5	5	5	5	180	85	50	5	5	5	5
Meltindex polymer	5	5	5	5	180	85	50	5	5	5	5
Density Polymer	5	5	5	5	180	85	50	5	5	5	5

Table C.2: *Known delays per transfer of the model in grade A*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	163	-	133	-	0	-	-	166	-	169	-
Temperature peak zone 1	163	-	133	-	3	-	-	166	-	169	-
Place peak zone 2	205	28	205	28	37	0	-	205	28	205	30
Temperature peak zone 2	205	28	205	28	30	0	-	205	28	205	30
Output temperature	303	128	303	128	132	48	0	303	128	303	128
Meltindex polymer	303	128	303	128	139	48	0	303	128	303	128
Density Polymer	303	128	303	128	136	48	0	303	128	303	128

TU/e technische universiteit eindhoven
IPCOS

Table C.3: *Number of FIR parameters per transfer of the model in grade B*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	10	-	20	-	135	-	-	5	-	5	-
Temperature peak zone 1	10	-	10	-	135	-	-	5	-	5	-
Place peak zone 2	5	5	5	5	180	85	-	5	5	5	5
Temperature peak zone 2	5	5	5	5	180	85	-	5	5	5	5
Output temperature	5	5	5	5	180	85	50	5	5	5	5
Meltindex polymer	5	5	5	5	180	85	50	5	5	5	5
Density Polymer	5	5	5	5	180	85	50	5	5	5	5

Table C.4: *Known delays per transfer of the model in grade B*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	118	-	92	-	0	-	-	122	-	125	-
Temperature peak zone 1	118	-	92	-	0	-	-	122	-	125	-
Place peak zone 2	243	64	240	66	69	0	-	243	67	244	68
Temperature peak zone 2	243	64	240	66	69	0	-	243	67	244	68
Output temperature	303	128	303	128	128	52	0	303	128	303	128
Meltindex polymer	303	128	303	128	128	52	0	303	128	303	128
Density Polymer	303	128	303	128	128	52	0	303	128	303	128

TU/e technische universiteit eindhoven
IPCOS

Table C.5: *Number of FIR parameters per transfer of the model in grade C*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	10	-	15	-	105	-	-	5	-	5	-
Temperature peak zone 1	10	-	10	-	105	-	-	5	-	5	-
Place peak zone 2	5	5	5	5	180	45	-	5	5	5	5
Temperature peak zone 2	5	5	5	5	180	45	-	5	5	5	5
Output temperature	5	5	5	5	180	85	50	5	5	5	5
Meltindex polymer	5	5	5	5	180	85	50	5	5	5	5
Density Polymer	5	5	5	5	180	85	50	5	5	5	5

Table C.6: *Known delays per transfer of the model in grade C*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	95	-	81	-	0	-	-	97	-	98	-
Temperature peak zone 1	96	-	81	-	0	-	-	97	-	98	-
Place peak zone 2	213	37	211	36	38	0	-	213	37	213	38
Temperature peak zone 2	213	37	211	36	38	0	-	213	37	213	38
Output temperature	303	128	303	128	128	48	0	303	128	303	128
Meltindex polymer	303	128	303	128	128	48	0	303	128	303	128
Density Polymer	303	128	303	128	128	48	0	303	128	303	128

TU/e technische universiteit eindhoven IPCOS

Table C.7: *Number of FIR parameters per transfer of the model in grade D*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	10	-	10	-	115	-	-	5	-	5	-
Temperature peak zone 1	10	-	10	-	115	-	-	5	-	5	-
Place peak zone 2	5	5	5	10	180	85	-	5	5	5	5
Temperature peak zone 2	5	5	5	10	180	85	-	5	5	5	5
Output temperature	5	5	5	5	180	85	50	5	5	5	5
Meltindex polymer	5	5	5	5	180	85	50	5	5	5	5
Density Polymer	5	5	5	5	180	85	50	5	5	5	5

Table C.8: *Known delays per transfer of the model in grade D*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	104	-	91	-	0	-	-	105	-	106	-
Temperature peak zone 1	105	-	91	-	0	-	-	105	-	106	-
Place peak zone 2	248	68	246	70	74	0	-	248	71	249	73
Temperature peak zone 2	248	68	246	70	74	0	-	248	71	249	73
Output temperature	303	128	303	128	128	51	0	303	128	303	128
Meltindex polymer	303	128	303	128	128	51	0	303	128	303	128
Density Polymer	303	128	303	128	128	51	0	303	128	303	128

TU/e technische universiteit eindhoven IPCOS

Table C.9: *Number of FIR parameters per transfer of the model in grade E*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	10	-	10	-	125	-	-	5	-	5	-
Temperature peak zone 1	10	-	10	-	125	-	-	5	-	5	-
Place peak zone 2	10	10	5	10	180	85	-	5	5	5	5
Temperature peak zone 2	10	10	5	10	180	85	-	5	5	5	5
Output temperature	5	5	5	5	180	85	50	5	5	5	5
Meltindex polymer	5	5	5	5	180	85	50	5	5	5	5
Density Polymer	5	5	5	5	180	85	50	5	5	5	5

Table C.10: *Known delays per transfer of the model in grade E*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	108	-	99	-	0	-	-	113	-	114	-
Temperature peak zone 1	108	-	99	-	0	-	-	113	-	114	-
Place peak zone 2	249	67	246	68	75	0	-	250	72	250	74
Temperature peak zone 2	249	67	246	68	75	0	-	250	72	250	74
Output temperature	303	128	303	128	128	52	0	303	128	303	128
Meltindex polymer	303	128	303	128	128	52	0	303	128	303	128
Density Polymer	303	128	303	128	128	52	0	303	128	303	128

TU/e technische universiteit eindhoven
IPCOS

Table C.11: *Number of FIR parameters per transfer of the model in grade F*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	10	-	10	-	180	-	-	10	-	5	-
Temperature peak zone 1	10	-	10	-	180	-	-	10	-	5	-
Place peak zone 2	5	10	10	10	180	60	-	5	5	5	5
Temperature peak zone 2	5	10	10	10	180	60	-	5	5	5	5
Output temperature	5	5	5	5	180	85	50	5	5	5	5
Meltindex polymer	5	5	5	5	180	85	50	5	5	5	5
Density Polymer	5	5	5	5	180	85	50	5	5	5	5

Table C.12: *Known delays per transfer of the model in grade F*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	158	-	172	-	0	-	-	160	-	165	-
Temperature peak zone 1	158	-	173	-	0	-	-	160	-	164	-
Place peak zone 2	221	43	219	36	46	0	-	221	44	221	46
Temperature peak zone 2	221	43	224	36	46	0	-	221	44	221	46
Output temperature	303	128	303	128	128	48	0	303	128	303	128
Meltindex polymer	303	128	303	128	128	48	0	303	128	303	128
Density Polymer	303	128	303	128	128	48	0	303	128	303	128

TU/e technische universiteit eindhoven
IPCOS

Table C.13: *Number of FIR parameters per transfer of the model in grade G*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	10	-	20	-	110	-	-	5	-	5	-
Temperature peak zone 1	10	-	10	-	110	-	-	5	-	5	-
Place peak zone 2	5	5	5	5	180	75	-	5	5	5	5
Temperature peak zone 2	5	5	5	5	180	75	-	5	5	5	5
Output temperature	5	5	5	5	180	85	50	5	5	5	5
Meltindex polymer	5	5	5	5	180	85	50	5	5	5	5
Density Polymer	5	5	5	5	180	85	50	5	5	5	5

Table C.14: *Known delays per transfer of the model in grade G*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	102	-	89	-	0	-	-	103	-	104	-
Temperature peak zone 1	102	-	89	-	0	-	-	103	-	104	-
Place peak zone 2	239	59	236	61	64	0	-	239	62	239	64
Temperature peak zone 2	239	59	236	61	64	0	-	239	62	239	64
Output temperature	303	128	303	128	128	48	0	303	128	303	128
Meltindex polymer	303	128	303	128	128	48	0	303	128	303	128
Density Polymer	303	128	303	128	128	48	0	303	128	303	128

TU/e technische universiteit eindhoven IPCOS

Table C.15: *Number of FIR parameters per transfer of the model in grade H*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	10	-	10	-	170	-	-	5	-	5	-
Temperature peak zone 1	10	-	10	-	170	-	-	5	-	5	-
Place peak zone 2	5	5	5	10	180	60	-	5	5	5	5
Temperature peak zone 2	5	5	5	5	180	60	-	5	5	5	5
Output temperature	5	5	5	5	180	85	50	5	5	5	5
Meltindex polymer	5	5	5	5	180	85	50	5	5	5	5
Density Polymer	5	5	5	5	180	85	50	5	5	5	5

Table C.16: *Known delays per transfer of the model in grade H*

	Initiator feed zone 1	Initiator feed zone 2	Feed temperature zone 1	Feed temperature zone 2	Jacket temperature zone 1	Jacket temperature zone 2	Jacket temperature zone 3	Monomer feed zone 1	Monomer feed zone 2	Solvent feed zone 1	Solvent feed zone 2
Place peak zone 1	155	-	148	-	0	-	-	159	-	162	-
Temperature peak zone 1	155	-	148	-	0	-	-	159	-	162	-
Place peak zone 2	224	47	224	39	49	0	-	224	47	224	49
Temperature peak zone 2	224	47	224	39	49	0	-	224	47	224	49
Output temperature	303	128	303	128	128	48	0	303	128	303	128
Meltindex polymer	303	128	303	128	128	48	0	303	128	303	128
Density Polymer	303	128	303	128	128	48	0	303	128	303	128

TU/e technische universiteit eindhoven
IPCOS

Appendix D

Step Responses of Grade B Model

In this chapter the step response of each transfer function of the grade B model will be shown. This will be done in 4 parts. Some cells in the matrix of plots are empty. This means there is no transfer between that input and output. These plots can be made of all models discussed in this document. Because of the huge size each plot needs and the fact that they do not add such relevant information to this report, the step responses of other models are not included.

As can be seen, the most transfers have no dynamics. Some transfers with dynamics have a strange response: one could think the number of FIR parameters used is not enough. However, using more FIR parameters in those cases is tested and gave the same result. Those strange responses are probably caused by shortages of the simulator.

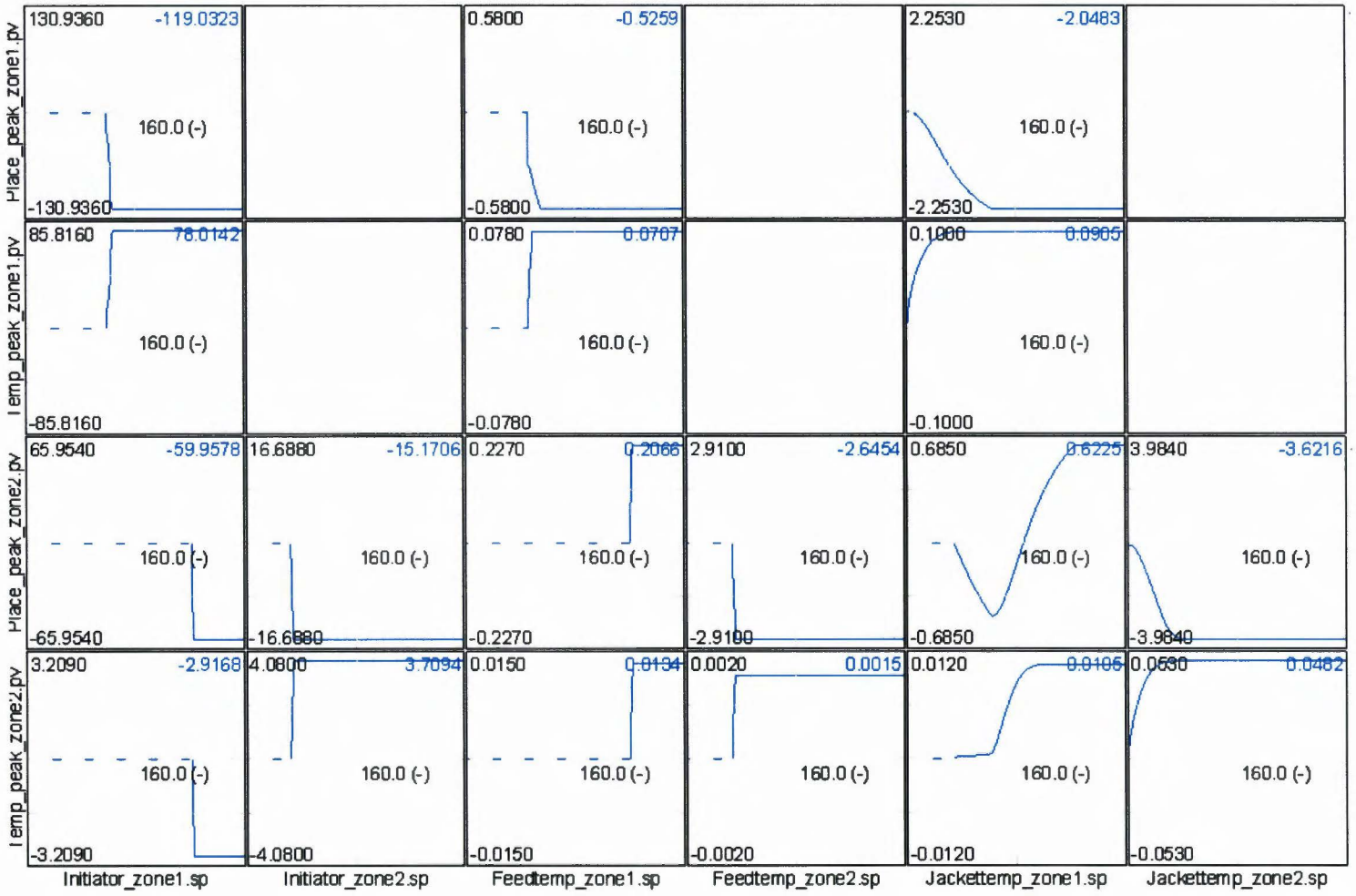

Figure D.1: *Step response of model for grade B, part 1/4*

TU/e technische universiteit eindhoven
IPCOS

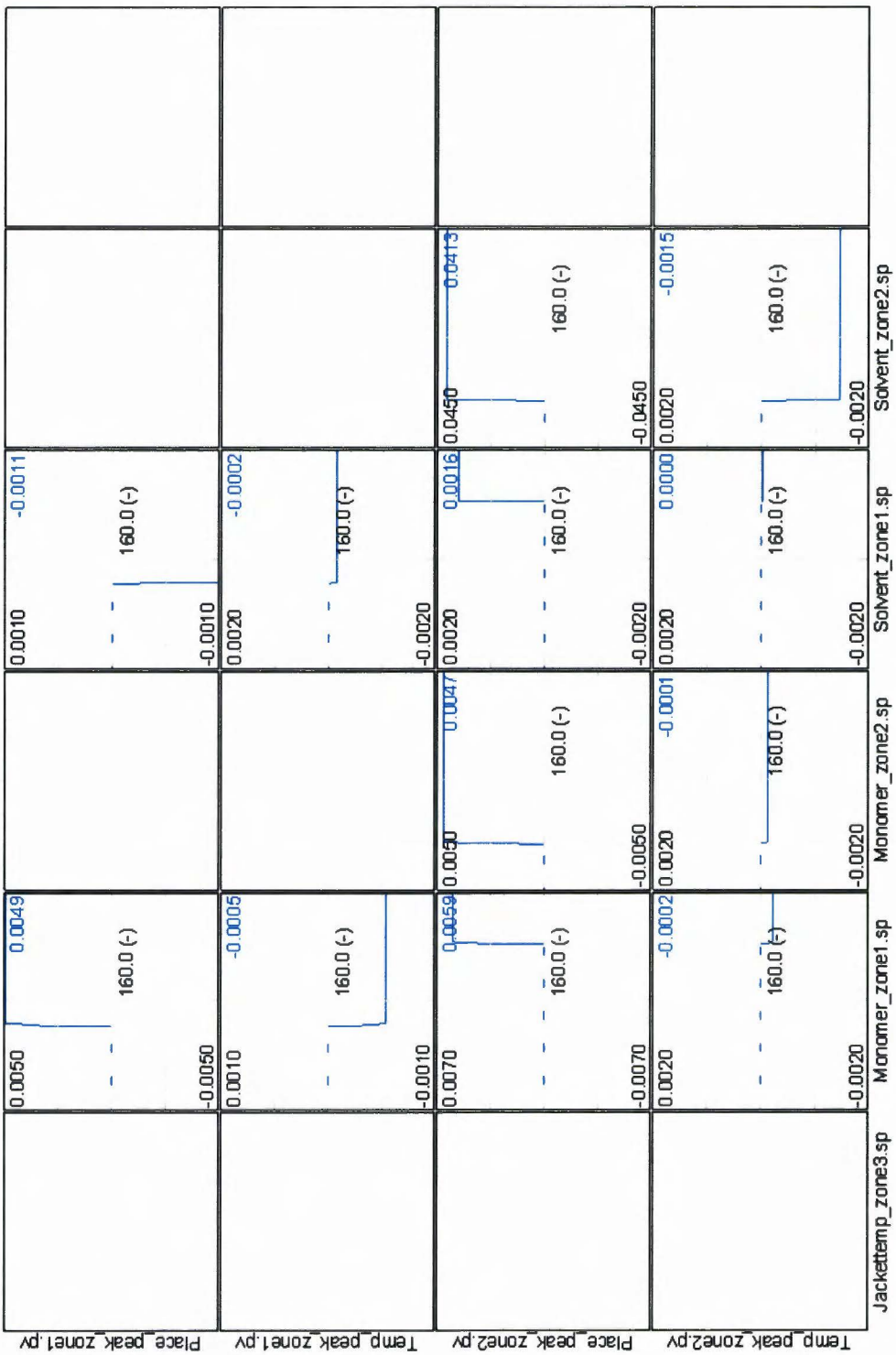

Figure D.2: *Step response of model for grade B, part 2/4*

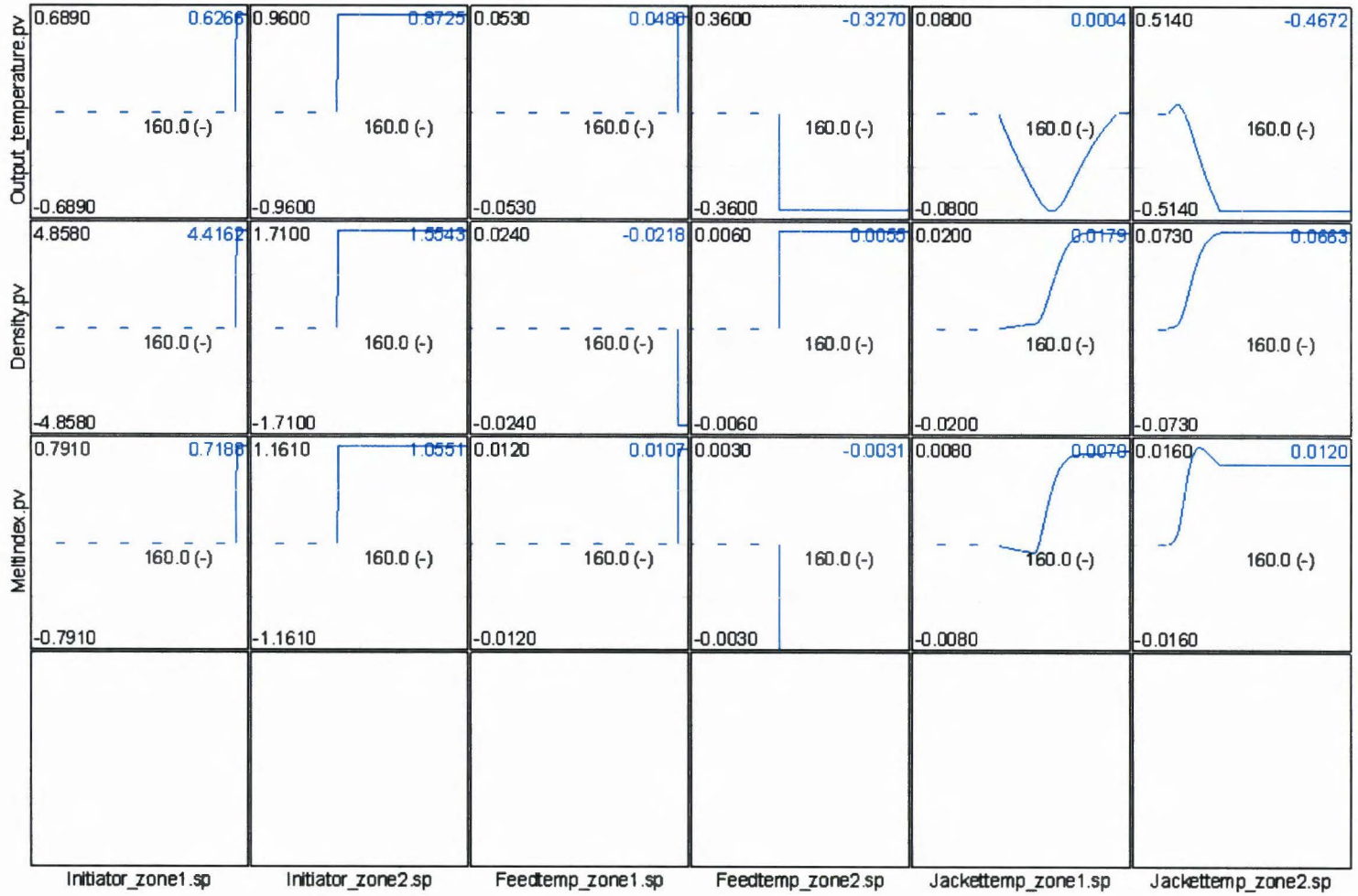

Figure D.3: *Step response of model for grade B, part 3/4*

TU/e technische universiteit eindhoven
IPCOS

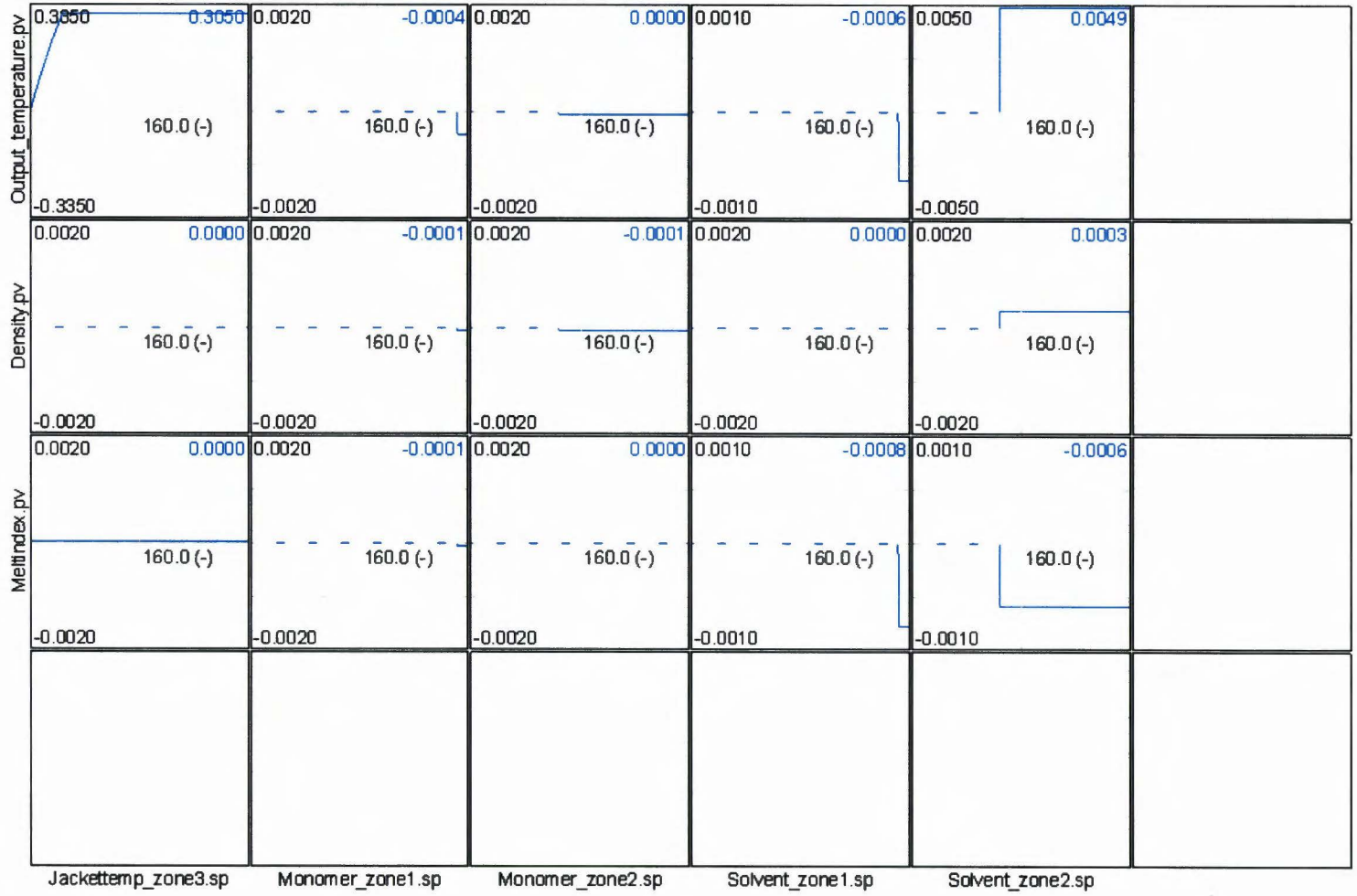

Figure D.4: *Step response of model for grade B, part 4/4*

TU/e technische universiteit eindhoven
IPCOS

Appendix E

Operating Points and Areas of the used Clusters

In this appendix, the operating point and range of each cluster can be found. For each MV the ideal value and the boundaries are given. The ideal values of the outputs are the output values when the simulator is initialised with the MV's ideal values.

Table E.1: *The operating point and area of cluster 1*

Input	Lower boundary	Ideal value	Upper boundary	unit
Initiator Feed Zone 1	0.21	0.46	0.71	$\frac{l}{hour}$
Initiator Feed Zone 2	1.9	2.7	3.5	$\frac{l}{hour}$
Feed Temperature Zone 1	25	60	80	°C
Feed Temperature Zone 2	51	60	70	°C
Jacket Temperature Zone 1	207	210	213	°C
Jacket Temperature Zone 2	194	200	220	°C
Jacket Temperature Zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	125.8	146.8	183.3	meter
Temperature peak zone 1	286.1	301.1	309.6	°C
Place peak zone 2	500.1	567.7	596.7	meter
Temperature peak zone 2	303.9	310.9	316.2	°C
Output temperature	252.7	261.2	267.2	°C
Density polymer	919.6	921.3	922.7	$\frac{gram}{10\ min}$
Meltindex polymer	1.09	1.69	2.44	$\frac{kg}{m^3}$

Table E.2: *The operating point and area of cluster 2*

Input	Lower boundary	Ideal value	Upper boundary	unit
Initiator Feed Zone 1	0.15	0.47	0.8	$\frac{l}{hour}$
Initiator Feed Zone 2	2.8	3.6	4.9	$\frac{l}{hour}$
Feed Temperature Zone 1	30	60	99	°C
Feed Temperature Zone 2	55	60	65	°C
Jacket Temperature Zone 1	208	210	212	°C
Jacket Temperature Zone 2	190	200	220	°C
Jacket Temperature Zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	105	146.4	177.6	meter
Temperature peak zone 1	280.3	301.5	311.9	°C
Place peak zone 2	484.1	545.3	591.6	meter
Temperature peak zone 2	311.6	316.7	323	°C
Output temperature	254	262.3	269.2	°C
Density polymer	921.5	922.9	924.6	$\frac{gram}{10\ min}$
Meltindex polymer	1.78	2.56	4.18	$\frac{kg}{m^3}$

Table E.3: *The operating point and area of cluster 3*

Input	Lower boundary	Ideal value	Upper boundary	unit
Initiator Feed Zone 1	0.095	0.13	0.17	$\frac{l}{hour}$
Initiator Feed Zone 2	1.8	2.6	3.5	$\frac{l}{hour}$
Feed Temperature Zone 1	10	33	65	°C
Feed Temperature Zone 2	69	96	123	°C
Jacket Temperature Zone 1	206	210	220	°C
Jacket Temperature Zone 2	190	200	210	°C
Jacket Temperature Zone 3	150	170	190	°C
Monomer feed zone 1	33407	35407	37407	$\frac{l}{hour}$
Monomer feed zone 2	86000	90000	94000	$\frac{l}{hour}$
Solvent feed zone 1	353	393	433	$\frac{l}{hour}$
Solvent feed zone 2	348	388	428	$\frac{l}{hour}$
Outputs				
Place peak zone 1	182.5	211.8	248	meter
Temperature peak zone 1	265.3	273.6	277.3	°C
Place peak zone 2	373.4	456.6	523.9	meter
Temperature peak zone 2	302.7	309.8	315.2	°C
Output temperature	230.3	247.2	257.3	°C
Density polymer	919.5	921.0	922.8	$\frac{gram}{10\ min}$
Meltindex polymer	0.65	1.15	1.75	$\frac{kg}{m^3}$