SIMULATION OF DEBUTANIZER USING MATLAB

By

Emira Farzana binti Ellias
15025
Chemical Engineering

Supervisor:
Dr Nasser bin Ramli

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Engineering (Hons)
(Chemical)

JANUARY 2015

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan
CERTIFICATION OF APPROVAL

SIMULATION OF DEBUTANIZER USING MATLAB

by

Emira Farzana binti Ellias

A project dissertation submitted to the
Chemical Engineering Programme
Universiti Teknologi PETRONAS
In partial fulfilment of the requirement for the
BACHELOR OF ENGINEERING (Hons)
(CHEMICAL ENGINEERING)

Approved by,

_____________________
(Dr Nasser Bin Ramli)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
January 2015
CERTIFICATION OF ORGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

______________________________
EMIRA FARZANA ELLIAS
ACKNOWLEDGEMENT

Above all, I would like to express my gratitude to my supervisor, Dr. Nasser Bin Ramli for his supervision and guidance throughout this project. I appreciate his support and advice that enable this project to achieve its objectives and be completed successfully within the time frame given. I would also like to thank Department of Chemical Engineering of Universiti Teknologi PETRONAS for allocating this project title for me. With this opportunity, I am able to increase my knowledge, both in theory and practical and most importantly, exploring and learning to appreciate the importance of computer programming in enhancing chemical processes.

Last but not the least, I would like to acknowledge my family and friends and those who has directly or indirectly involved in this project, for their tremendous support and motivation during undertaking this project.
ABSTRACT

A debutanizer is a fractional distillation column used to separate butane from gas during the refining process. In this project a model of a debutanizer is developed for simulation purposes. The model can be used to study the steady state and dynamic behavior of the column. The main tool used for modeling and simulation of the debutanizer is MATLAB. Steady state model involves algebraic equations that will be solved by MATLAB functions while MATLAB toolbox ode45 is applied in dynamic model to obtain the solutions of ordinary differential equations. Simulation study was performed on steady state and dynamic state model in order to understand the principle and behaviour of the debutanizer.
# TABLE OF CONTENTS

CERTIFICATION OF APPROVAL ........................................................................ i  
CERTIFICATION OF ORIGINALITY ............................................................... ii  
ACKNOWLEDGEMENT ................................................................................... iii  
ABSTRACT ........................................................................................................ iv  
LIST OF FIGURES ............................................................................................. vii  

| CHAPTER 1 | : | INTRODUCTION ......................................................... 1  
1.1 Background of Study .............................................................. 1  
1.2 Problem Statement .............................................................. 2  
1.3 Objectives and Scope of Study ............................................. 3  

| CHAPTER 2 | : | LITERATURE REVIEW ............................................... 4  
2.0.1 Modeling and Simulation for Distillation Column of Steady State Behaviour ............................................. 4  
2.0.2 A Simulated Debutanizer Column .................................... 5  
2.1 Dynamic Modelling .............................................................. 6  
2.2 PID Controller ................................................................. 7  
2.3 Internal Model Controller (IMC) ......................................... 12  
2.4 Overview of LPG .............................................................. 14  
2.5 Production of LPG in PP(T)SB ........................................... 16  
2.6 LPG Production from Crude Distillation Unit (CDU) .......... 17  

iii
CHAPTER 3 : METHODOLOGY ........................................... 19
3.1 Project Flow Chart ............................................. 19
3.2 Data Collection ................................................. 19
3.3 On-line Controller Tuning ................................. 25
3.4 PID Controller Settings using various tuning
equations .............................................................. 28

CHAPTER 4 : RESULTS AND DISCUSSION ............... 29
4.1 The model of a Debutanizer Column ................. 29
4.2 Material and Energy Balance ............................. 30
4.3 Tray Holdup Dynamics ...................................... 35
4.4 PID Controller Settings using Various Tuning
Equations ............................................................ 36

CHAPTER 5 : CONCLUSION ............................................. 40

REFERENCES .......................................................... 37
APPENDICES ............................................................ 38
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 2.1</td>
<td>Process Flow Diagram of Debutanizer</td>
<td>5</td>
</tr>
<tr>
<td>Figure 2.2</td>
<td>A block diagram of a PID controller</td>
<td>8</td>
</tr>
<tr>
<td>Figure 2.3</td>
<td>Internal Model Control Scheme</td>
<td>12</td>
</tr>
<tr>
<td>Figure 2.4</td>
<td>Molecule of Propane and Butane</td>
<td>14</td>
</tr>
<tr>
<td>Figure 2.5</td>
<td>Chemical Composition of LPG</td>
<td>16</td>
</tr>
<tr>
<td>Figure 4.6</td>
<td>Composition of Butane, Error and Manipulated Variable value after</td>
<td>38</td>
</tr>
<tr>
<td></td>
<td>Closed Loop Cohen Coon Response-Setpoint</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 1
INTRODUCTION

1.0 Background of Study

A debutanizer is a certain type of fractional distillation column. It is used to separate butane from gas during the refining process. In order to separate or purify the liquid, distillation is used as a process of heating a liquid to vapor and condensing the vapors back to liquid. Some examples would be to include water that is distilled to purify it and also distill liquor to make it stronger. As occurs in a debutanizer, fractional distillation is the separation of a fraction which is a set of compounds that have a boiling point within a range from the rest of the mixture.

From the petroleum distribution, raw natural gas can be drilled from natural deposits or released as a byproduct. In either case, it is not the nearly pure methane used by consumers. About 20 percent of raw natural gas is various heavier hydrocarbons which are chemical compounds made up of hydrogen and carbon, such as butane, propane and ethane. Oxygen, nitrogen, hydrogen sulfide and trace amounts of the noble gases might be mixed within the gas.

As the natural gas is refined, all these other substances are removed in the production phase. Huge fraction columns which are the industrial towers that are about 2-20 feet across and 20-200 feet high or sometimes more will vaporize the gas in an expansion turbine and later condense it with multiple valve trays. The column might be debutanizer, a depropanizer or a deethanizer, depending on which hydrocarbon is being removed from the natural gas liquids (NGL). The natural gas is pure methane when the process is finished. To make leaks more detectible, traces amount of mercaptan which is the source of the rotten egg smell associated with the natural gas are actually added to the methane.
Different refineries might refine the gas in different orders. However, it is very common that the NGL will first flow through a depropanizer to remove heavier propane from the mix and through the debutanizer to siphon off the butane. There are other refineries that use a debutanizer to remove a mix of butane and propane (C3/C4 mix) from the NGL and then use a depropanizer to separate butane and propane from each other.

Updating debutanizers for greater efficiency would be both profitable and eco-friendly. Normal petroleum refinery operation allows the light hydrocarbon gasses to be dissolved into the petroleum. Siphoning off these gases gives the oil company another energy source to market but makes the petroleum more efficient by decreasing plugging and fouling of burner tips. (Hubbard, 2014)

1.1 Problem Statement

The main task of this project is to model and simulate a debutanizer where butane is separated from natural gas using MATLAB software. A MATLAB program will be developed to model and simulate a debutanizer. The project attempts to expand and modify the program in order to broaden its application.
1.2 Objective & Scope of Study

The main objective of the research is to develop a model of a debutanizer using MATLAB program. The program is then used to study its steady state and dynamic behavior of for further understanding. The characteristics and performance of the column can be revealed by observing the steady state and dynamics of the system.

This project provides an opportunity to explore the features of MATLAB software and its importance in engineering study apart from deriving the mathematical model of the column especially in modeling and simulation of process equipment.

This project will be based on data collected from a chosen refinery in Malaysia which is PETRONAS Penapisan (Terengganu) Sdn Bhd: PP(T)SB. LPG is one of the light and top products yield from the refinery’s distillation tower. The focus of this project is the debutanizer column in the Crude Distillation Unit (CDU) of Kerteh Refinery 1.

The task of modeling and simulation is given individually and to be completed within 28 weeks (2 semesters). Therefore, the scope of this project has to be clearly defined and should be feasible within the time period. The task is concentrated on observing the behavior of the system using MATLAB software.

For this project, there are main aims in modeling and simulation using MATLAB were identified as below:

1. Mathematical modeling of a debutanizer distillation column.
2. Programming models into MATLAB with appropriate functions.
3. Gaining optimum performance of debutanizer column and maximize LPG production from Crude Distillation Unit (CDU)
CHAPTER 2
LITREATURE REVIEW AND THEORY

2.0 LITREATURE REVIEW

In the recent years, debutanizer has been gaining a growing attention in both industrial and a research sector as the process is economically favourable in many cases. Debutanizer, a fractional distillation equipment, is used to separate butane and lighter components from natural gas liquids. There were not many published on experimentally validated models for debutanizer. The research on this area is still quite new compare to other separation process.

Some of the literature I found is a rigorous model for the simulation of the steady state behavior of the distillation column by (Sohail Rasool Lone 2013). A simulated case study on the debutanizer for advance process control by Morten Bakke (2008) also explains more on the dynamic state of debutanizer. These journal topics are discussed more below.

2.0.1 Modeling and Simulation for Distillation Column of Steady State Behaviour

In this journal by Sohail Rasool Lone (2013), MESH equations which represent the behaviour of the distillation column is used in MATLAB. Methanol is used as feed. The effect of feed condition and the feed composition on the steady state behavior is studied. The mathematical model presented consists of a set of algebraic equations for the column. Assumptions made comply well with the mathematical models. ‘Fsolve’ solver was used in MATLAB to solve the non-linear equations.
2.0.2 A Simulated Debutanizer Case Study

The main objective of this journal is to control a distillation column (debutanizer). This journal only shows the simulation of the control but does not model the debutanizer. MATLAB is not used in this journal. Instead it uses a Honeywell simulation tool called UniSim. However, this journal stress the focus more on the control theory instead of the tools used for simulation. (Bakke & Skogestad)

![Process Flow Diagram of Debutanizer](image)

*Figure 2.1: Process Flow Diagram of Debutanizer*
2.1 Dynamic Modelling

Dynamic models are used to predict how a process and its controls respond to various upsets as a function of time. They can be used to evaluate equipment conditions and control schemes and to determine the reliability and safety of a design before capital is committed to the project. For grassroots and revamp projects, dynamic simulation can be used to accurately assess transient conditions that determine process design temperatures and pressures. In many cases, unnecessary capital expenditures can be avoided using dynamic simulation.

Dynamic simulation during process design leads to benefits during plant start-up. Expensive field changes, which impact schedule, can often be minimized if the equipment and control system is validated using dynamic simulation. Start-up and shutdown sequences can be tested using dynamic simulation.

Dynamic simulation also provides controller-tuning parameters for use during start-up. In many cases, accurate controller settings can prevent expensive shutdowns and accelerate plant start-up. Dynamic simulation models used for process design are not based on transfer functions as normally found in operator training simulators, but on fundamental engineering principles and actual physical equations governing the process.

When used for process design, dynamic simulation models include:

- Equipment models that include mass and energy inventory from differential balances
- Rigorous thermodynamics based on property correlations, equations of state, and steam tables
- Actual piping, valve, distillation tray, and equipment hydraulics for incompressible, compressible, and critical flow.
These models are so detailed that the results can influence engineering design decisions and ensure a realistic prediction of the process and the control system’s interaction to assess control system stability.

2.2 PID CONTROLLER

A proportional-integral-derivative controller (PID controller) is a generic control loop feedback mechanism (controller) widely used in industrial control systems. A PID controller attempts to correct the error between a measured process variable and a desired setpoint by calculating and then outputting a corrective action that can adjust the process accordingly and rapidly, to keep the error minimal.

The PID controller calculation (algorithm) involves three separate parameters; the proportional, the integral and derivative values. The proportional value determines the reaction to the error, the integral value determines the reaction based on the sum of recent errors, and the derivative value determines the reaction based on the rate at which the error has been changing. The weighted sum of these three actions is used to adjust the process via a control element such as the position of a control valve or the power supply of a heating element.
Figure 2.2: A block diagram of a PID controller

By tuning the three constants in the PID controller algorithm, the controller can provide control action designed for specific process requirements. The response of the controller can be described in terms of the responsiveness of the controller to an error, the degree to which the controller overshoots the setpoint and the degree of system oscillation. Note that the use of the PID algorithm for control does not guarantee optimal control of the system or system stability.

Some applications may require using only one or two modes to provide the appropriate system control. This is achieved by setting the gain of undesired control outputs to zero. A PID controller will be called a PI, PD, P or I controller in the absence of the respective control actions. PI controllers are particularly common, since derivative action is very sensitive to measurement noise, and the absence of an integral value may prevent the system from reaching its target value due to the control action.
The PID control scheme is named after its three correcting terms, whose sum constitutes the manipulated variable (MV). Hence:

\[ MV(t) = P_{out} + I_{out} + D_{out} \]

where \( P_{out}, I_{out}, \) and \( D_{out} \) are the contributions to the output from the PID controller from each of the three terms, as defined below.

\subsection{2.2.1 Proportional term}

The proportional term (sometimes called gain) makes a change to the output that is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a constant \( K_p \), called the proportional gain.

The proportional term is given by:

\[ P_{out} = K_p e(t) \]

Where

- \( P_{out} \): Proportional term of output
- \( K_p \): Proportional gain, a tuning parameter
- \( e \): Error = SP - PV
- \( t \): Time of instantaneous time (the present)

A high proportional gain results in a large change in the output for a given change in the error. If the proportional gain is too high, the system can become unstable. In contrast, a small gain results in a small output response to a large input error, and a less responsive (or sensitive)
controller. If the proportional gain is too low, the control action may be too small when responding to system disturbances.

In the absence of disturbances, pure proportional control will not settle at its target value, but will retain a steady state error that is a function of the proportional gain and the process gain. Despite the steady-state offset, both tuning theory and industrial practice indicate that it is the proportional term that should contribute the bulk of the output change.

### 2.2.2 Integral term

The contribution from the integral term (sometimes called reset) is proportional to both the magnitude of the error and the duration of the error. Summing the instantaneous error over time (integrating the error) gives the accumulated offset that should have been corrected previously. The accumulated error is then multiplied by the integral gain and added to the controller output. The magnitude of the contribution of the integral term to the overall control action is determined by the integral gain, Ki.

The integral term is given by:

\[ I_{out} = K_i \int_0^t e(\tau) d\tau \]

Where

- Iout : Integral term of output
- Ki : Integral gain, a tuning parameter
- e : Error = SP - PV
- t : Time or instantaneous time (the present)
- \( \tau \) : A dummy integration variable
The integral term (when added to the proportional term) accelerates the movement of the process towards set point and eliminates the residual steady-state error that occurs with a proportional only controller. However, since the integral term is responding to accumulated errors from the past, it can cause the present value to overshoot the set point value (cross over the set point and then create a deviation in the other direction). For further notes regarding integral gain tuning and controller stability, see the section on loop tuning.

2.3.3 Derivative terms

The rate of change of the process error is calculated by determining the slope of the error over time (i.e., its first derivative with respect to time) and multiplying this rate of change by the derivative gain \( Ka \). The magnitude of the contribution of the derivative term (sometimes called rate) to the overall control action is termed the derivative gain,

The derivative term is given by:

\[
D_{out} = K_d \frac{de}{dt}(t)
\]

Where
- \( D_{out} \): Derivative term of output
- \( K_d \): Derivative gain, a tuning parameter
- \( e \): Error = SP- PV
- \( t \): Time or instantaneous time (the present)

The derivative term slows the rate of change of the controller output and this effect is most noticeable close to the controller set point. Hence, derivative control is used to reduce the magnitude of the overshoot produced by the integral component and improve the
combined controller-process stability. However, differentiation of a signal amplifies noise and thus this term in the controller is highly sensitive to noise in the error term, and can cause a process to become unstable if the noise and the derivative gain are sufficiently large.

2.3 INTERNAL MODEL CONTROL (IMC)

Internal Model Control (IMC) was developed by Morari and coworkers (Garcia and Morari, 1982; Rivera et al., 1986). The IMC method is based on an assumed process model and leads to analytical expressions for the controller settings. The IMC approach has the advantage that it allows model uncertainty and tradeoff between performance and robustness to be considered in a more systematic fashion.

The IMC method is based on the simplified block diagram shown in Figure 2.2:

![Internal Model Control Scheme](image)

**Figure 2.3 : Internal Model Control Scheme**

From Figure 2.3, below are the following conventions to describe the blocks in the system:

- controller - $G_c(s)$
- process - $G_p(s)$
- internal model - $G_{pm}(s)$
- disturbance - $d(s)$
- disturbance transfer function - $D(s)$
The Figure 2.3 shows the standard linear IMC scheme where the process model Gpm(s) plays an explicit role in the control structure. This structure has some advantages over convectional feedback loop structures. For the nominal case Gp(s) = Gpm(s), for instance, the feedback is only affected by the disturbance D(s) such that the system is effectively open loop and hence no stability problems can arise. This control structure also depicts that if the process Gp(s) is stable, which is true for most industrial processes, the closed loop will be stable for any stable controller Gc(s). Thus, the controller Gc(s) can simply be designed as a feedforward controller in the IMC scheme.

From the IMC scheme depicted in Figure 2.2 above, the feedback signal is represented as follows:

\[ d'(s) = [Gp(s) - Gpm(s)]U(s) + D(s) \]

As said above, if the model is an exact representation of the process then d'(s) is simply a measure of the disturbance. If there exist no disturbance, then d'(s) is simply a measure in difference in behaviour between the process and its model.

The closed loop transfer function of the IMC scheme can be seen modelled as below:

\[ Y(s) = \frac{R(s)G_{c}(s)G_{p}(s)H[1\delta G_{c}(s)G_{pm}(s)]D(s)}{1H[G_{p}(s)\delta G_{pm}(s)]G_{c}(s)} \]

The above transfer function will be shown to exist in the next section on transfer functions. From this closed loop analysis, we can see that if we design a controller such that Gc(s) = Gpm(s)^{-1} where the process model is an exact
representation of the process, then the design will yield good set point tracking and disturbance rejection.

The controller is then detuned for robustness to account for a possible plant model mismatch. This is done by augmenting the controller with a low-pass filter to reduce the loop gain for high frequencies. This idea also counteracts the effects of model inversion, as the pure inverse of the model is not physically realizable. The inversion often process model may also lead to unstable controllers in case of unstable zeros in the model.

(Source: Donald A. Mohutsiwa (2006), PID controller tuning using Internal Model Control)

2.4 OVERVIEW OF LPG

Liquefied Petroleum Gas (LPG) is a non-renewable source of energy. It is a byproduct which extracted from crude oil and natural gas. The main composition of LPG is a mixture of hydrocarbon gases containing three or four carbon atoms. The normal components of LPG are propane (C\textsubscript{3}H\textsubscript{8}) and butane (C\textsubscript{4}H\textsubscript{10}). Small concentrations of other hydrocarbons may also be present like propylene (C\textsubscript{3}H\textsubscript{6}) and butylenes (C\textsubscript{4}H\textsubscript{8}).

LPG is used as a fuel in heating appliances and vehicles. It replaces chlorofluorocarbons, as an aerosol propellant to reduce ozone layer damaged.

\textbf{Figure 2.4} : \textit{Molecule of Propane and Butane}
LPG is a vapor at atmospheric pressure and normal ambient temperatures which can be liquefied by compression at ambient temperature to enhance the molecular distribution. It is one of the cleanest fuels available and can be easily condensed, packaged, stored and utilized, which makes it an ideal energy source for a wide range of applications. In Malaysia, it is used extensively as fuel for domestic cooking, agricultural, heating and drying process and vehicle as well as for industrial applications.

In fact, LPG can be liquefied at relatively low pressure approximately at 2.2 bar which approaching 2.17 atm pressure; it facilitates the storage of large amount in spherical tank or cylindrical tank. This is because LPG in liquid state is 250 times denser than when in the gas state. For an ideal gas, the general equation of state is as the following:

\[ PV = RT \]

Where, P is absolute pressure; V is molal volume; T is temperature and R is gas constant.

In low pressure case, gases will depart slightly from ideality, a compressibility factor, \( z \) is then introduced into the equation of state. At high pressure, the gas has low molal volume which slightly less than the expected volume for ideal gas. Therefore, to liquefy the gas, increment of pressure is required. The LPG density will be higher at liquid state since it's volume reduced and mass remains constant.

LPG is highly flammable and must therefore be stored away from sources of ignition and in a well-ventilated area, so that any leak can disperse safely. A special chemical; mercaptan, is added to give distinctive to LPG; an unpleasant smell so that a leak can be detected. The concentration of the
chemical is such that an LPG leak can be smelled when the concentration is well below the lower limit of flammability.

The composition chart in Figure 2.5 below outlines the typical components of LPG as supplied via Gas Malaysia's distribution system.

![Chemical Composition of LPG Supplied by Gas Malaysia Sdn. Bhd](image)

Figure 2.5: Chemical Composition of LPG Supplied by Gas Malaysia Sdn. Bhd

2.6 PRODUCTION OF LPG IN PETRONAS PENAPIS~J~I (TERENGGANU) SDN. BHD.

LPG Recovery Unit is designed to process the unstabilized product from Crude Tower together with tmstabilized LPG from Reformer Unit. The designed capacity of mixed LPG is 1180 BPSD. Crude Distillation Unit (CDU), Catalytic Reforming Unit (CRU) and Condensate Fractionation Unit (CFU) are the sources of throughput for LPG Recovery Unit. The fractionation section of LPG Recovery Unit consists of a Deethanizer column and Debutanizer column. The function of this fractionation section is to recover light gaseous and LPG from the overhead distillate before producing Light Naphtha. The light gases mainly ethane (C2) from Deethanizer is muted to Refinery Fuel Gas System. Mixed LPG which is a mixture of propane and butane are sent to LPG storage.
The throughput of Debutanizer column is the bottom product of Deethanizer. The Debutanizer column is equipped with 35 valve type trays and one liquid pass. The low boiling point components rise up the tower in contact with the internal reflux.

In contrast, the high boiling point components, which are the heavier components, flow down in contact with the vapor produced in the Debutanizer reboiler which provided to the reboiler bottom level. Thus, hydrocarbon feed to the Debutanizer will be fractionated to the lighter components; mixture of C3 and C4 as the overhead product and heavier components as the bottom product of the column.

The Debutanizer condenser condensed the overhead vapor. Part of the stream may by pass from the condenser in order to control the overhead pressure. The condensed overhead is collected inside the Debutanizer receiver drum. The Debutanizer overhead system is set and controlled at 8.0 kg/cm2 (7.85 bars) by the Debutanizer overhead pressure control valve which has two split range controllers. While if the pressure low, the other valve will be opened and by passed part of the overhead gas from the Debutanizer condenser. Meanwhile, if the pressure of the system increases too much, manual valve should be opened and routed to the flare. In case of emergency loss of fuel gas system, the off gas will be routed to the Refinery Fuel Gas System by opening the tie in block valves.

Part of the condensed hydrocarbon collected is pumped by the Debutanizer reflux pump to the Debutanizer top as reflux. The reflux flow rate is controlled by the Debutanizer overhead reflux control valve. The balanced can be routed to flare the stock if the LPG product is off specification.
The Debutanizer reboiler is equipped to the Debutanizer bottom section which is heated by the Kerosene pump around system. The Debutanizer reboiler temperature control valve controls the reboiler temperature. While the Debutanizer bottom level controller controls the bottom product level.

Bottom product is then muted via Light Naphtha rundown cooler to Light Straight Run Naphtha rundown tank or to slop if off specification. Light Straight Run Naphtha (LSRN) rundown rate is measured by LSRN flow meter.
CHAPTER 3
METHODOLOGY

3.1 Project Flow Chart

- Preliminary study on past researched based on related topic and issue
- Identify the variables/parameters of the project
- Definition of problem

3.2 Data Collection

After the problem is clearly defined, the relevant data is identified and gathered. Simulation using MATLAB will be performed in order to know which data is needed. Data is collected from supervisor with the help of an engineer working at PETRONAS Penapisan (Terengganu) Sdn. Bhd. (PP(T)SB).
3.2.1 Debutanizer Column (C-110)

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of trays of the column</td>
<td>35</td>
</tr>
<tr>
<td>Feed tray – stage number</td>
<td>23</td>
</tr>
<tr>
<td>Type of tray used</td>
<td>Valve</td>
</tr>
<tr>
<td>Column diameter</td>
<td>1.3m</td>
</tr>
<tr>
<td>Column length</td>
<td>23.95m</td>
</tr>
<tr>
<td>Type of Condenser</td>
<td>Partial</td>
</tr>
<tr>
<td>Feed mass flow rate</td>
<td>44 106 kg/hr</td>
</tr>
<tr>
<td>Feed temperature</td>
<td>113°C</td>
</tr>
<tr>
<td>Feed pressure</td>
<td>823.8 kPa</td>
</tr>
<tr>
<td>Overhead vapor mass flow rate</td>
<td>11 286 kg/hr</td>
</tr>
<tr>
<td>Overhead liquid mass flow rate</td>
<td>5040 kg/hr</td>
</tr>
<tr>
<td>Pressure condenser</td>
<td>823.8 kPa</td>
</tr>
<tr>
<td>Pressure Reboiler</td>
<td>853.2 kPa</td>
</tr>
</tbody>
</table>

*Table 3.1: Debutanizer column plant data*

3.2.2 Composition in the feed in mass fraction including components in the feed

*Liquefied Petroleum Gas (LPG) and LSRN*

<table>
<thead>
<tr>
<th>Composition</th>
<th>Mass Fraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Propane</td>
<td>0,037</td>
</tr>
<tr>
<td>i-Butane</td>
<td>0,093</td>
</tr>
<tr>
<td>n-Butane</td>
<td>0,062</td>
</tr>
<tr>
<td>i-Pentane</td>
<td>0,082</td>
</tr>
<tr>
<td>n-Pentane</td>
<td>0,110</td>
</tr>
</tbody>
</table>
3.2.3 Operational Parameters

3.2.3.1 TC-110

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>Auto</td>
</tr>
<tr>
<td>Action</td>
<td>Reverse</td>
</tr>
<tr>
<td>SP</td>
<td>140.7°C</td>
</tr>
<tr>
<td>OP</td>
<td>52.00%</td>
</tr>
<tr>
<td>Kc</td>
<td>250</td>
</tr>
<tr>
<td>Ti</td>
<td>1.33 minutes</td>
</tr>
<tr>
<td>Td</td>
<td>0.333 minutes</td>
</tr>
<tr>
<td>PV Minimum</td>
<td>125.15°C</td>
</tr>
<tr>
<td>PV Maximum</td>
<td>145.55°C</td>
</tr>
</tbody>
</table>

3.2.3.2 FIC-123

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>Auto</td>
</tr>
<tr>
<td>Action</td>
<td>Reverse</td>
</tr>
<tr>
<td>SP</td>
<td>19.37 m³/h</td>
</tr>
<tr>
<td>OP</td>
<td>74.2%</td>
</tr>
<tr>
<td>Kc</td>
<td>0.1</td>
</tr>
<tr>
<td>Ti</td>
<td>0.5 minutes</td>
</tr>
<tr>
<td>Td</td>
<td>-</td>
</tr>
<tr>
<td>PV Minimum</td>
<td>19.37 m³/h</td>
</tr>
<tr>
<td>PV Maximum</td>
<td>56.40 m³/h</td>
</tr>
</tbody>
</table>
### 3.2.3.3 PC-109

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>Auto</td>
</tr>
<tr>
<td>Action</td>
<td>Reverse</td>
</tr>
<tr>
<td>SP</td>
<td>823.8 kPa</td>
</tr>
<tr>
<td>OP</td>
<td>25.30</td>
</tr>
<tr>
<td>Kc</td>
<td>0.5</td>
</tr>
<tr>
<td>Ti</td>
<td>0.7 minutes</td>
</tr>
<tr>
<td>Td</td>
<td>-</td>
</tr>
<tr>
<td>PV Minimum</td>
<td>552.6 kPa</td>
</tr>
<tr>
<td>PV Maximum</td>
<td>903.58 kPa</td>
</tr>
</tbody>
</table>

### 3.2.3.4 LC-111

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>Manual</td>
</tr>
<tr>
<td>Action</td>
<td>Reverse</td>
</tr>
<tr>
<td>SP</td>
<td>60.65 %</td>
</tr>
<tr>
<td>OP</td>
<td>60.00 %</td>
</tr>
<tr>
<td>Kc</td>
<td>0.45</td>
</tr>
<tr>
<td>Ti</td>
<td>11 minutes</td>
</tr>
<tr>
<td>Td</td>
<td>-</td>
</tr>
<tr>
<td>PV Minimum</td>
<td>45.4 %</td>
</tr>
<tr>
<td>PV Maximum</td>
<td>75.76 %</td>
</tr>
</tbody>
</table>
### 3.2.3.5 LC-112

<table>
<thead>
<tr>
<th>Mode</th>
<th>Manual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Reverse</td>
</tr>
<tr>
<td>SP</td>
<td>50.00%</td>
</tr>
<tr>
<td>OP</td>
<td>60.00%</td>
</tr>
<tr>
<td>Kc</td>
<td>0.25</td>
</tr>
<tr>
<td>Ti</td>
<td>15 minutes</td>
</tr>
<tr>
<td>Td</td>
<td>-</td>
</tr>
<tr>
<td>PV Minimum</td>
<td>25%</td>
</tr>
<tr>
<td>PV Maximum</td>
<td>73%</td>
</tr>
</tbody>
</table>

### 3.2.3.6 FIC-126

<table>
<thead>
<tr>
<th>Mode</th>
<th>Cascade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Reverse</td>
</tr>
<tr>
<td>SP</td>
<td>8.8206 m3/h</td>
</tr>
<tr>
<td>OP</td>
<td>54.30%</td>
</tr>
<tr>
<td>Kc</td>
<td>0.2</td>
</tr>
<tr>
<td>Ti</td>
<td>0.2 minutes</td>
</tr>
<tr>
<td>Td</td>
<td>-</td>
</tr>
<tr>
<td>PV Minimum</td>
<td>0.00 m</td>
</tr>
<tr>
<td>PV Maximum</td>
<td>15.8</td>
</tr>
</tbody>
</table>
### 3.2.3.7 FC-121

<table>
<thead>
<tr>
<th>Mode</th>
<th>Manual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Reverse</td>
</tr>
<tr>
<td>SP</td>
<td>515.9 m3/h</td>
</tr>
<tr>
<td>OP</td>
<td>50.00</td>
</tr>
<tr>
<td>Kc</td>
<td>0.028</td>
</tr>
<tr>
<td>Ti</td>
<td>0.417 minutes</td>
</tr>
<tr>
<td>Td</td>
<td>-</td>
</tr>
<tr>
<td>PV Minimum</td>
<td>379 m3/h</td>
</tr>
<tr>
<td>PV Maximum</td>
<td>721 m3/h</td>
</tr>
</tbody>
</table>
3.3 On-Line Controller Tuning

3.3.1 Step Test Method

After the process has reached steady state (at least approximately), the controller is placed in the manual mode. Then a step change in the controller output (e.g. 52 to 82%) is introduced. The controller settings are based on the closed loop response.

3.3.1.1 Process model parameter estimation using MATLAB

1. Insert the step change data (process variables, PV and controller output, OP) in M-File.
2. Open Ident

3. Set time domain data
4. Estimate process models
3.4 PID Controller Settings using various tuning equations

<table>
<thead>
<tr>
<th>Tuning Equation</th>
<th>$K_c$</th>
<th>$T_i$</th>
<th>$T_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ziegler and Nichol's</td>
<td>$\frac{2\tau_i}{K_p\tau_d}$</td>
<td>$2\tau_d$</td>
<td>$0.5\tau_d$</td>
</tr>
<tr>
<td>Parr</td>
<td>$\frac{1.25\tau_i}{K_p\tau_d}$</td>
<td>$2.5\tau_d$</td>
<td>$0.4\tau_d$</td>
</tr>
<tr>
<td>Chien et al.</td>
<td>$\frac{0.95\tau_i}{K_p\tau_d}$</td>
<td>$2.38\tau_d$</td>
<td>$0.42\tau_d$</td>
</tr>
<tr>
<td>Murrill</td>
<td>$\frac{1.370}{K_p}\left(\frac{\tau_i}{\tau_d}\right)^{0.950}$</td>
<td>$\frac{\tau_i}{1.351}\left(\frac{\tau_d}{\tau_i}\right)^{0.738}$</td>
<td>$0.365\tau_i\left(\frac{\tau_d}{\tau_i}\right)^{0.950}$</td>
</tr>
<tr>
<td>Cohen and Coon</td>
<td>$\frac{1}{K_p}\left(1.35\frac{\tau_i}{\tau_d}+0.25\right)$</td>
<td>$\frac{2.5\tau_d + 0.46\left(\frac{\tau_d}{\tau_i}\right)^2}{1+0.61\frac{\tau_d}{\tau_i}}$</td>
<td>$0.37\tau_i\frac{1}{1+0.19\frac{\tau_d}{\tau_i}}$</td>
</tr>
<tr>
<td>Saiz and Ozgen</td>
<td>$\frac{1}{K_p}\left(0.6939\frac{\tau_i}{\tau_d}+0.1814\right)$</td>
<td>$\frac{0.8647\tau_i+0.226\tau_d}{\tau_d}$</td>
<td>$0.0565\tau_i\frac{0.8647\tau_i+0.226}{\tau_d}$</td>
</tr>
</tbody>
</table>

[Reference: Aidan O'Dwyer (2003), Handbook of PI and PID Controller Tuning Rules]
CHAPTER 4

RESULTS AND DISCUSSION

4.1. The Model of Debutanizer Column

The debutanizer column will be employed for the separation of a four-component hydrocarbon mixture. The rigorous process model consists of a set of ordinary differential equations (ODEs) coupled with algebraic equations/correlations. The ODEs are obtained from mass and energy balances around each plate of the distillation column. The algebraic equations/correlations are used to predict the thermodynamic and physical properties, plate hydraulics, and actual vapour-phase compositions.

The following assumptions have been made in the development of the process model:

- The molar vapour holdup is negligible compared to the molar liquid holdup.
- The liquid and vapour leaving each plate are in thermal equilibrium.
- The definition of Murphree plate efficiency applies for each plate.
- The liquid is perfectly mixed on each plate.
- No subcooling is considered in the total condenser.
- Coolant and steam dynamics in the condenser and reboiler respectively are neglected.
- Molal liquid holdup varies in each tray (including reflux drum and column base) but the holdups in reflux drum and column base are constant in volume.
- The column operates with the top (Pt) and base (PB) pressures of 102.98 and 122.70 psia, respectively. The tray-to-tray pressure varies linearly according to the following form of equation:

\[
P_n = P_B - \frac{n(P_B - P_T)}{n_T + 1}
\]
Where $P_n$ is the pressure in the $n$th tray and $nT$ represents the total number of trays (here 35).

- Liquid hydraulics are calculated from the Francis weir formula.
- Vapour-liquid equilibria (VLE) and enthalpy are calculated based on the Soave-Redlich-Kwong (SRK) equation of state.
- The Muller iteration method based on second degree equation is used in the bubble-point calculations.

4.2 Material and Energy Balance Equations

For all column trays including a reboiler and a reflux drum, and for a component $i=1,..., N_c$, the model ordinary differential equations representing total continuity (one per tray), component continuity ($N_c-1$ per tray) and energy balance (one per tray) can be obtained. However for this simulation we will focus only on:

Reboiler-Column Base System

Total Continuity: 

\[ m_B = L_1 - V_B - B \]

Component Continuity: 

\[ m_B x_{B,i} = L_1 x_{1,i} - V_B y_{B,i} - B x_{B,i} \]

Dynamic State:

\[ \frac{d x_{B,i}}{dt} = \frac{1}{m} [L_1 x_{1,i} + V_B y_{B,i} - B x_{B,i}] \]

Bottom Tray (subscript ‘1’)

Total Continuity: 

\[ m_1 = L_2 + V_B - L_1 - V_1 \]

Component Continuity: 

\[ m_1 x_{1,i} = L_2 x_{2,i} + V_B y_{B,i} - L_1 x_{1,i} - V_1 y_{1,i} \]

Dynamic State:

\[ \frac{d x_{1,i}}{dt} = \frac{1}{m} [L_2 x_{2,i} + V_B y_{B,i} - L_1 x_{1,i} - V_1 y_{1,i}] \]
Nth Tray (subscript ‘n’ where to and to )

Total Continuity :

\[ m_n = L_{n+1} + V_{n-1} - L_n - V_n \]

Component Continuity :

\[ m_n x_{n,i} = L_{n+1} x_{n+1,i} + V_{n-1} y_{n-1,i} - L_n x_{n,i} - V_n y_{n,i} \]

Dynamic State :

\[ \frac{dx_{n,i}}{dt} = \frac{1}{m} \left[ L_{n+1} x_{n+1,i} + V_{n-1} y_{n-1,i} - L_n x_{n,i} - V_n y_{n,i} \right] \]

Feed Tray (subscript ‘nf’)

Total Continuity:

\[ m_{nf} = L_{nf+1} + F^L + V_{nf+1} - L_{nf} - V_{nf} \]

Component Continuity:

\[ m_{nf} x_{nf,i} = L_{nf+1} x_{nf+1,i} + F^L x_{F,i} + V_{nf} y_{nf-1,i} - L_{nf} x_{nf,i} - V_{nf} y_{nf,i} \]

Dynamic State :

\[ \frac{dx_{nf,i}}{dt} = \frac{1}{M} \left[ L_{nf+1} x_{nf+1,i} + F^L x_{F,i} + V_{nf} y_{nf-1,i} - L_{nf} x_{nf,i} \right] \]

Above Feed Tray (subscript ‘nF+1’)

Total Continuity:

\[ m_{nF+1} = L_{nF+2} + F^V + V_{nF} - L_{nF+1} - V_{nF+1} \]

Component Continuity:

\[ m_{nF+1} x_{nF+1,i} = L_{nF+2} x_{nF+2,i} + F^V y_{F,i} + V_{nF} y_{nF,i} - L_{nF+1} x_{nF+1,i} - V_{nF+1} y_{nF+1,i} \]

Dynamic State :

\[ \frac{dx_{nF+1,i}}{dt} = \frac{1}{m} \left[ L_{nF+2} x_{nF+2} + V_{nF} y_{nF,i} - L_{nF+1} x_{nF+1,i} - V_{nF+1} y_{nF+1,i} \right] \]
Top Tray (subscript ’nT’)

Total Continuity:
\[ m_{nT} = R + V_{nT-1} - L_{nT} - V_{nT} \]

Component Continuity:
\[ m_{nT} x_{nT,i} = R x_{D,i} + V_{nT-1} y_{nT-1,i} - L_{nT} x_{nT,i} - V_{nT} y_{nT,i} \]

Dynamic State:
\[ \frac{dx_{nT,i}}{dt} = \frac{1}{m} [R x_{D,i} + V_{nT-1} y_{nT-1,i} - L_{nT} x_{nT,i} - V_{nT} y_{nT,i}] \]

Condenser-Reflux Drum System (subscript ‘D’)

Total Continuity:
\[ m_D = V_{nT} - R - D \]

Component Continuity:
\[ m_D x_{D,i} = V_{nT} y_{nT,i} - (R + D) x_{D,i} \]

Dynamic State:
\[ \frac{dx_{D,i}}{dt} = \frac{1}{m} [V_{nT} y_{nT,i} - (R + D) x_{D,i}] \]

In the above distillation modeling equations:

\[ x_{n,i} = \text{the mole fraction of component i in liquid stream leaving nth tray} \]

\[ y_{n,i} = \text{the mole fraction of component i in a vapour stream leaving nth tray} \]

\[ x_{F,i} = \text{the mole fraction of component i in the liquid feed,} \]

\[ y_{F,i} = \text{the mole fraction of component i in the vapour feed,} \]

\[ x_{D,i} = \text{the mole fraction of component i in the distillate.} \]
\( x_{B,i} \) = the mole fraction of component \( i \) in the bottom product,

\( L_n \) = the liquid flow rate leaving \( n \)th tray (lbmol/h),

\( V_n \) = the vapour flow rate leaving \( n \)th tray (lbmol/h),

\( x_{D,i} \) = the mole fraction of component \( i \) in the distillate.

\( V_n \) = the vapour flow rate leaving \( n \)th tray

\( R \) = the reflux flow rate (lbmol/h),

\( D \) = the distillate flow rate (lbmol/h),

\( V_B \) = the vapour boil-up rate (lbmol/h),

\( B \) = the bottom product flow rate (lbmol/h),

\( F^L \) = the flow rate of the liquid feed (lbmol/h),

\( F^V \) = the flow rate of the vapour feed (lbmol/h),

\( m_n \) = the liquid holdup on the \( n \)th tray (lbmol),

\( m_D \) = the liquid holdup in the reflux drum (lbmol),

\( m_B \) = the liquid holdup in the reboiler-column base system (lbmol)

\( Q_R \) = the heat input to the reboiler (Btu/h),

\( N_c \) = the number of components (here 4) and

\( n_F \) = the feed tray.
4.3 Tray Holdup Dynamics

Distefano (1968) has reported a procedure for tray holdup calculations in distillation columns. This approach formulates based on the assumption of constant volume holdup on the plates. Therefore, the molal holdup on nth plate is given by:

\[ m_D = \frac{m_{vD} \rho_{LD}}{MW_D} \]

where, \( m_{vD} \) is the constant volumetric liquid holdup (volume) on any plate and \( \rho_{LD} \) represent the average density (mass/volume) and average molecular weight, respectively, of the liquid stream on nth plate.

In real-time distillation processes, the volumetric liquid holdups (or heights) in the reflux drum and column base are held almost constant by implementing conventional level controllers (proportional) to the manipulation of distillate and bottom product flow rates, respectively. But this is not the ease for the internal trays. Distefano (1968) has suggested to use the Equation for all trays of a batch distillation column except the still-pot. Again Luyben (1990) preferred to simulate the total continuity equations for the calculation of liquid holdups on the internal trays of a multicomponent continuous distillation column. But he considered constant volumetric liquid holdups in the condenser-accumulator as well as in the reboiler - column base system. In order to predict the holdup dynamics in the condenser-reflux drum and reboiler column base systems, the molal holdups may be represented according to the approach explained above as:

For Condenser-Reflux Drum System

\[ m_D = \frac{m_{vD} \rho_{LD}}{MW_D} \]

For Reboiler-Column Base System

\[ m_B = \frac{m_{vB} \rho_{LB}}{MW_D} \]
Where, subscripts 'D' and 'B' refer, respectively to the distillate and bottoms. For this project, debutanizer column Equations will be used for the computations respectively and internal tray holdups will be calculated simulating the total continuity equations.

4.4 PID Controller Settings using Various Tuning Equations

Error is then calculated for each tuning to choose the best performance tuning. The error data is tabulated below:

<table>
<thead>
<tr>
<th>No</th>
<th>Type of Tuning</th>
<th>Error Value Calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IAE</td>
<td>7.645 x 10^-6</td>
</tr>
<tr>
<td>2</td>
<td>ITAE</td>
<td>3.112 x 10^-5</td>
</tr>
<tr>
<td>3</td>
<td>IMC</td>
<td>0.0018</td>
</tr>
<tr>
<td>4</td>
<td>HA</td>
<td>1.2775 x 10^-5</td>
</tr>
<tr>
<td>5</td>
<td>ISE</td>
<td>6.5215 x 10^-6</td>
</tr>
<tr>
<td>6</td>
<td>Cohen Coon</td>
<td>2.4481 x 10^-7</td>
</tr>
<tr>
<td>7</td>
<td>Ziegler and Nichols (1942) Model Method 2</td>
<td>9.9733 x 10^-7</td>
</tr>
<tr>
<td>8</td>
<td>Hazebroek and Van der Waerden(1950),Model Method 2</td>
<td>4.5997 x 10^-7</td>
</tr>
<tr>
<td>9</td>
<td>Chien(1952), Servo, Model: Method 2, 0% overshoot</td>
<td>3.6639 x 10^-6</td>
</tr>
<tr>
<td>10</td>
<td>Chien(1952), Servo, Model: Method 2, 20% overshoot</td>
<td>7.3022 x 10^-6</td>
</tr>
<tr>
<td>11</td>
<td>Cohen and Coon (1953), Model: Method 2</td>
<td>8.2070 x 10^-6</td>
</tr>
<tr>
<td>12</td>
<td>Two Constraints Method- Wolfe (1951), Model: Method 3</td>
<td>4.6099 x 10^-6</td>
</tr>
<tr>
<td>13</td>
<td>Two Constraints Criterion- Murrill (1967), Model: Method 4</td>
<td>5.2716 x 10^-6</td>
</tr>
<tr>
<td>15</td>
<td>St. Clair (1997), Model: Method 4</td>
<td>4.0526 x 10^-6</td>
</tr>
<tr>
<td>17</td>
<td>Hay (1998) Servo Tuning 1, Model: Method 2</td>
<td>2.2114 x 10^-6</td>
</tr>
</tbody>
</table>
Table 4.1: Error for each tuning done

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Error Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>Minimum IAE - Smith and Corripio (1997), Model: Method 1</td>
<td>7.3022 x 10^6</td>
</tr>
<tr>
<td>19</td>
<td>Minimum IAE - Rovira et al. (1969), Model: Method 4</td>
<td>7.6450 x 10^6</td>
</tr>
<tr>
<td>20</td>
<td>Minimum IAE - Marlin (1995), Model: Method 1</td>
<td>2.6216 x 10^5</td>
</tr>
<tr>
<td>21</td>
<td>Minimum IAE - Hwang (1995), Model: Method 26</td>
<td>8.182x10^-4</td>
</tr>
<tr>
<td>22</td>
<td>Minimum ISE - Zhuang and Atherton (1993), Model: Method 1</td>
<td>0.0030</td>
</tr>
<tr>
<td>23</td>
<td>Minimum ISE - Khan and Lehman (1996), Model: Method 1</td>
<td>7.3155 x 10^6</td>
</tr>
<tr>
<td>24</td>
<td>Minimum ITAE - Rovira et al. (1969), Model: Method 4</td>
<td>1.4230 x 10^6</td>
</tr>
<tr>
<td>26</td>
<td>Minimum ISTES - Zhuang and Atherton (1993), Model: Method 1</td>
<td>7.0588 x 10^-7</td>
</tr>
</tbody>
</table>

The tuning with the least error value is the best tuning method for debutanizer. From the table above, we can see that tuning with smallest value of error is the Cohen Coon Tuning. Therefore we can say that Cohen Coon tuning is the best tuning method for the debutanizer column.
4.4.1 Close Loop Cohen Coon Response-Setpoint

*Figure 4.1: Composition of Butane, Error and Manipulated Variable value after Closed Loop Cohen Coon Response-Setpoint*
Figure 4.1 represent the composition of Butane, Error and Manipulated Variable value after Closed Loop Cohen Coon Response-Setpoint tuning. The composition increases proportionally with time when using the closed loop Cohen Coon Tuning. Error increases to a point after a while and maintains the same value throughout the time run. The manipulated variable has a value after a few minits and increases proportionally after.
CHAPTER 5
CONCLUSION

This project is mainly about modelling a debutanizer column in order to optimize the performance of the column and to identify the best tuning method for the column. Debutanizer Column in Crude Distillation unit (CDU) of Kerteh Refinery-1 (KR-1) has been chosen as a model for this project.

From the results and discussion, it is concluded that different tuning methods would give different results on the behavior of the response. The optimum response has been chosen considering the behaviour of the response and the value of error calculation.

All the research and findings obtained will be used to improve the overall performance of the plant as well as to improve the quality of the product and maximize profitability. The successful outcome of this project will be a great helping hand for industrial application.
REFERENCES

Bakke, M., & Skogestad, S. A simulated debutanizer case study for teaching advanced process control.


Donald A. Mohutsiwa (2006), PID controller tuning using Internal Model Control

APPENDICES

Other Tuning Graphs

4.4.2 Closed Loop IAE Response-Setpoint

Figure 4.1: Composition of Butane, Error and Manipulated Variable value after Closed Loop IAE Response-Setpoint
4.4.3 Close Loop ITAE Response-Setpoint

![Graph 1: Close Loop ITAE Response-Setpoint]

![Graph 2: Error]

![Graph 3: Manipulated Variable]
Figure 4.2: Composition of Butane, Error and Manipulated Variable value after Closed Loop ITAE Response-Setpoint

4.4.4 Closed Loop IMCResponse-Setpoint
Figure 4.3: Composition of Butane, Error and Manipulated Variable value after Closed Loop IMC Response-Setpoint

4.4.5 Close Loop HA Response-Setpoint
Figure 4.4: Composition of Butane, Error and Manipulated Variable value after Closed Loop HA Response-Setpoint

4.4.6 Close Loop ISE Response-Setpoint
Figure 4.5: Composition of Butane, Error and Manipulated Variable value after Closed Loop ISE Response-Setpoint

4.4.7 Close Loop Ziegler and Nichols Response
Figure 4.7: Composition of Butane, Error and Manipulated Variable value after Close Loop Ziegler and Nichols Response.

4.4.8 Close Loop Hazerbroek and Van der Waarden Response
Figure 4.8: Composition of Butane, Error and Manipulated Variable value after Closed Loop Hazerbroek and Van der Waarden Response

4.4.9 Close Loop Chien (1952)- Servo, Model: Method 2, 0% overshoot
Figure 4.9: Composition of Butane, Error and Manipulated Variable value after Closed Loop Chien (1952) - Servo, Model: Method 2, 0% overshoot

4.4.10 Close Loop Chien (1952) Servo, Model 2, 20% Overshoot
Figure 4.10: Composition of Butane, Error and Manipulated Variable value after Closed Loop Chien (1952) Servo, Model 2, 20% Overshoot

4.4.11 Close Loop Cohen and Coon (1953), Model Method 2
4.4.12 Close Loop Two Constraints Method – Wolfe (1951), Model : Method 3
Figure 4.12: Composition of Butane, Error and Manipulated Variable value after Closed Loop Two Constraints Method – Wolfe (1951), Model : Method 3

Figure 4.13: Composition of Butane, Error and Manipulated Variable value after Closed Loop Two Constraints Criterion-Murill (1967) Method : Method 4

Figure 4.14: Composition of Butane, Error and Manipulated Variable value after Closed Loop McMillan (1994) Model: Method 4

4.4.15 Close Loop St. Clair (1997), Model: Method 4
Figure 4.15: Composition of Butane, Error and Manipulated Variable value after Closed Loop St. Clair (1997), Model: Method 4

Figure 4.16: Composition of Butane, Error and Manipulated Variable value after Closed Loop Shinskey (2000), (2001) Model: Method 2

4.4.17 Close Loop Hay (1998), Servo Tuning 1, Model: Method 2
Figure 4.17: Composition of Butane, Error and Manipulated Variable value after Closed Loop Hay (1998), Servo Tuning 1, Model: Method 2

4.4.18 Close Loop Hay (1998) Servo Tuning 2, Model: Method 2
Figure 4.18: Composition of Butane, Error and Manipulated Variable value after Closed Loop Hay (1998) Servo Tuning 2, Model: Method 2

4.4.19 Close Loop Minimum IAE-Rovira et. Al. (1969), Model: Method 4
Figure 4.19: Composition of Butane, Error and Manipulated Variable value after Closed Loop Minimum IAE-Rovira et. Al.(1969), Model : Method 4

4.4.20 Close Loop Minimum IAE-Marlin(1995)-Model: Method 1
Figure 4.20: Composition of Butane, Error and Manipulated Variable value after Closed Loop Minimum IAE-Marlin(1995)-Model: Method 1

4.4.21 Close Loop Minimum IAE- Smith and Corripio, Model : Method 1
Figure 4.21: Composition of Butane, Error and Manipulated Variable value after Closed Loop Minimum IAE- Smith and Corripio, Model: Method 1

M-Files Codings

1. DynamicState.m

   % A program developed by
   % Emira Farzana Ellias
   % Universiti Teknologi PETRONAS
   % Bandar Seri Iskandar
   % Perak

   % Declaration of some variables

   global k1 k1
   global k2
   global DIST_PAR
   global stepk1
   global tstepk1
   global xA xB xC xD
   global yA yB yC yD

   % General Data Input for Column

   disp('Please Enter Distillation Column Parameters');

   D=input('Distillate withdrawal rate (mol/s):');
   DIST_PAR(1)=input('Number of stages :');
   DIST_PAR(2)=input('Reactant A Feed stage:');
   DIST_PAR(3)=input('Reactant B Feed stage: ');
   DIST_PAR(4)=input('Feed A flowmte (tool/s):');
   DIST_PAR(5)=input('Feed B flowrate (mol/s):');
   DIST_PAR(6)=input('Feed A purity (mole traction):');
   DIST_PAR(7)=input('Feed B purity (mole fraction):');
   DIST_PAR(8)=input('Feed A quality:');
   DIST_PAR(9)=input('Feed B quality:');
   DIST_PAR(10)=input('Reflux flowrate (mol/s):');
   DIST_PAR(11)=(DIST_PAR(10)+DIST_PAR(4)*DIST_PAR(8)+DIST_PAR(5)*DIST_PAR(9))- (DIST_PAR(4)+DIST_PAR(5)-D);
   DIST_PAR(24)=input ('Overall liquid hold-up in column (mol.s/m3):');
   DIST_PAR(25)=input('Distillate molar hold-up (mol):');
   DIST_PAR(26)=input('Bottom molar hold-up (mol):');
   DIST_PAR(35)=input('Condensor Pressure (bar):');
DIST_PAR(36)=input('Reboiler Pressure (bar):');

%Antomds Equation

disp('Please, inSet Ant~in~ Equatio~ Con~anU for Reactant A, Reactant B, Product C and Product D');

disp('For Reactant A');
DIST_PAR(12)=input('Constant A:');
DIST_PAR(13)=input('Constant B:');
DIST_PAR(14)=input('Constant C:');

disp('For Reactant B');
DIST_PAR(15)=input('Constant A:');
DIST_PAR(16)=input('Constant B:');
DIST_PAR(17)=input('Constant C:');

disp('For Product C');
DIST_PAR(18)=input('Constant A:');
DIST_PAR(19)=input('Constant B:');
DIST_PAR(20)=input('Constant C:');

disp('For Product D');
DIST_PAR(21)=input('Constant A:');
DIST_PAR(22)=input('Constant B:');
DIST_PAR(23)=input('Constant C:');

% number of stages % teed A stage % feed B stage % teed A 11oxvrate % teed B flow~ate % teed A purity % iced B purity % teed A qua~t3' % teed B qua~t3' % reflux t, ervao,ete % vapor flowrate % liquid holdup % condenser pressure % rebokler pressure

dist = input('Do you want to introduce a disturbance into the process? Please Enter 1 for YES or 2 for NO:');

% Assignment of Variables

ns = DIST_PAR(1);
nfA = DIST_PAR(2);
fB = DIST_PAR(3);
feedAi = DIST_PAR(4);
feedBi = DIST_PAR(5);
zfeedAi = DIST_PAR(6);
zfeedBi = DIST_PAR(7);

qfA = DIST_PAR(8);
qfB = DIST_PAR(9);
refluxi = DIST_PAR(10);
vapori = DIST_PAR(11);
mf = DIST_PAR(24);
md = DIST_PAR(25);
mb = DIST_PAR(26);
PC = DIST_PAR(35);
PR = DIST_PAR(36);

AA = DIST_PAR(12)
BA = DIST_PAR(13)
CA = DIST_PAR(14)

AB = DIST_PAR(15)
BB = DIST_PAR(16)
CB = DIST_PAR(17)

AC = DIST_PAR(18)
BC = DIST_PAR(19)
CC = DIST_PAR(20)

AD = DIST_PAR(21)
BD = DIST_PAR(22)
CD = DIST_PAR(23)

% To Introduce disturbance into the system

if dist==1;
    disp('Please insert magnitude step disturbance and the time of step change for desired parameter:');
    disp('For other parameters, please enter 0 to indicate no step change for the particular parameter:');
    DIST_PAR(27) = input('Magnitude of step in reflux:');
    DIST_PAR(28) = input('Time of reflux step change:');
    DIST_PAR(29) = input('Magnitude of step in vapor:');
    DIST_PAR(30) = input('Time of vapor step change:');
    DIST_PAR(31) = input('Magnitude of feed A composition change:');
    DIST_PAR(32) = input('Time of feed A composition change:');
    DIST_PAR(33) = input('Magnitude of feed flow A change:');
    DIST_PAR(34) = input('Time of feed flow A change:');
    DIST_PAR(37) = input('Magnitude of feed B composition change:');
    DIST_PAR(38) = input('Time of feed B composition change:');
    DIST_PAR(39) = input('Magnitude of feed B flow change:');
    DIST_PAR(40) = input('Time of feed flow B change:');

    stepk1 = input('Magnitude of change of forward reaction rate:');
    tstepk1 = input('Time of forward reaction rate change:');

    disp('Please wait, solving for the dynamic-state stage composition');
    disp('of reactant A, reactant B, product C and product D');

% Zero the function vector
xA=zeros(ns,1);
xB=zeros(ns,1);
xC=zeros(ns,1);
xD=zeros(ns,1);

% Linear Variation of Pressure at Each Column Stage

P(1)=Pc;
for i=2:ns-1;
P(i)=P(i-1)+(Pr-Pc)/(ns-1);
end
P(ns)=Pr;

% Subroutine for Column Tray Temperature
% To calculate vapor composition from Raoult’s Law
% Assume initial guess temperature of 50°C

T=50;

while abs(net)>0.02;
pA(i)=10^((Aa-(Ba/(T+Ca)))/760);
pB(i)=10^((Ab-(Bb/(T+Cb)))/760);
pC(i)=10^((Ac-(Bc/(T(i)+Cc)))/760);
pD(i)=10^((Ad-(Bd/(T(i)+Cd)))/760);

yA1(i)=pA(i)*xA(i)/P(i);
yB1(i)=pB(i)*xB(i)/P(i);
yC1(i)=pC(i)*xC(i)/P(i);

xD(i)=1-xA(i)-xB(i)-xC(i);

yD1(i)=pD(i)*xD(i)/P(i);

sum_y=yA1(i)+yB1(i)+yC1(i)+yD1(i);
net=1-sum_y;
T(i)=T(i)+0.001;
end

yA(i)=yA1(i);
yB(i)=yB1(i);
yC(i)=yC1(i);
yD(i)=yD1(i);
% Solving ordinary differential equation by Runge-Kutta 4th and 5th order method
% Initial condition of component composition

for i=1:ns;
    xA0(i)=xA(i)
    xB0(i)=xB(i)
    xC0(i)=xC(i)
    xD0(i)=xD(i)
end

% Time range of change

tspan = [0 100];

[t, xA]=ode45('dist_dynA',tspan,xA0);
[t, xB]=ode45('dist_dynB',tspan,xB0);
[t, xC]=ode45('dist_dynC',tspan,xC0);
[t, xD]=ode45('dist_dynD',tspan,xD0);

% No disturbance in system
else
    DIST_PAR(27)=0;
    DIST_PAR(28)=0;
    DIST_PAR(29)=0;
    DIST_PAR(30)=0;
    DIST_PAR(31)=0;
    DIST_PAR(32)=0;
    DIST_PAR(33)=0;
    DIST_PAR(34)=0;
end

% Zero the function vector

xA=zeros(ns, 1);
xB=zeros(ns, 1);
xC=zeros(ns, 1);
xD=zeros(ns, 1);

% Linear Variation of Pressure at Each Column Stage

P(1)=Pc;
for i=2:ns-1;
    P(i)=P(i-1)+(Pr-Pc)/(ns-1)
end

% Subroutine for Column Tray Temperature
% To calculate vapor composition from Raoult's law

T=50;

net=1;
while abs(net)>0.02;
    pA(i) = 10^*(Aa - (Ba/(T + Ca)))/760;
    pB(i) = 10^*(Ab - (Bb/(T + Cb)))/760;
    pC(i) = 10^*(Ac - (Bc/(T + Cc)))/760;
    pD(i) = 10^*(Ad - (Bd/(T + Cd)))/760;

    yA1(i) = pA(i)*xA(i)/P(i);
    yB1(i) = pB(i)*xB(i)/P(i);
    yC1(i) = pC(i)*xC(i)/P(i);
    yD(i) = 1-xA(i)-xB(i)-xC(i);
    yD1(i) = pD(i)*xD(i)/P(i);

    sum_y = yA1(i)+yB1(i)+yC1(i)+yD1(i);
    net = 1- sum_y;
end

yA(i) = yA1(i);
yB(i) = yB1(i);
yC(i) = yC1(i);
yD(i) = yD1(i);

% Solving ordinary differential equation by Runge-Kutta 4th and 5th order method
% Initial condition of composition range
for i=1:ns;
    xA0(i)=xA(i);
    xB0(i)=xB(i);
    xC0(i)=xC(i);
    xD0(i)=xD(i);
end

tspan = [0 100];

[t, xA]=ode45('dist_dynA',tspan,xA0);
[t, xB]=ode45('dist_dynB',tspan,xB0);
[t, xC]=ode45('dist_dynC',tspan,xC0);
[t, xD]=ode45('dist__dynD',tspan,xD0);
% Graph Plotting of Result

plot(t,x(:,4)), xlabel('Time'), ylabel('Composition of Methyl Acetate'),
title('Disturbance'), axis([0 100 0.5 1.0]);

% Function file for Reactant A

function xAdot = dist_dynA(t,xA)

% Declaration of some variables

global DIST_PAR
global k1i k1l
global k2
global stepk1
global tstepk1
global xB xC xD
global yA yB yC yD

% Assignment of Values to parameters

ns = DIST_PAR(1);
nfA = DIST_PAR(2);
nfB = DIST_PAR(3);
feedAi = DIST_PAR(4);
feedBi = DIST_PAR(5);
zfeedAi = DIST_PAR(6);
zfeedBi = DIST_PAR(7);
qfA = DIST_PAR(8);
qfB = DIST_PAR(9);
refluxi = DIST_PAR(10);
vapori = DIST_PAR(11);

mt = DIST_PAR(24);
md = DIST_PAR(25);
mb = DIST_PAR(26);
Pc = DIST_PAR(35);
Pr = DIST_PAR(36);

if length(DIST_PAR)==40;
    stepr = DIST_PAR(27);
end

if length(DIST_PAR)==40;
    stepr = DIST_PAR(27);
end
tstepr = DIST_PAR(28);
stepv = DIST_PAR(29);
tstepv = DIST_PAR(30);
stepzfA = DIST_PAR(31);
tstepzfA = DIST_PAR(32);
stepfA = DIST_PAR(33);
tstepfA = DIST_PAR(34);
stepzfB = DIST_PAR(37);
tstepzfB = DIST_PAR(38);
stepfl3 = DIST_PAR(39);
tstepfB = DIST_PAR(40);
else
stepr = 0;
tstepr = 0;
stepv = 0;
tstepv = 0;
stepzfA = 0;
tstepzfA = 0;
stepfA = 0;
tstepfA = 0;
stepzfB = 0;
tstepzfB = 0;
stepfB = 0;
tstepfB = 0;
end

% Check for disturbances in Reflux

if t<tstepr;
    reflux = refluxi;
else
    reflux = refluxi + stepr;
end

% Check the disturbances vapor boil-up

if t<tstepv;
    vapor = vapori;
else
    vapor = vapori + stepv;
end
% Check for disturbances in Feed A Composition
if t<tstepzfA;
    zfeedA = zfeedAi;
else
    zfeedA = zfeedAi + stepzfA;
end

% Check for disturbances in Feed B Composition
if t<tstepzfB;
    zfeedB = zfeedBi;
else
    zfeedB = zfeedBi + stepzfB;
end

% Check for disturbances in Feed A Flowrate
if t<tstepfA;
    feedA=feedAi;
else
    feedA=feedAi+stepfA;
end

% Check for disturbances in Feed B Flowrate
if t<tstepfB;
    feedB=feedBi;
else
    feedB=feedBi+stepfB;
end

% Check the disturbances in Rate of Forward Reaction
if t<tstepkl;
    kl=kli;
else
    kl=kli+stepkl;
end

% Rectifying and stripping section liquid flowrates
lr = reflux;
lsl = reflux+feedA*qfA;
is = reflux+feedA*qfA+feedB*qtB;
% Rectifying and stripping section vapor flowrates

vs = vapor;  
vr=vs+feedA*(1-qfA)+feedB*(1-qfB);

% Distillate. and bottoms rates

dist = vr - reflux;  
lbot = ls - vs;

% Zeros the function vector

xAdot=zeros(ns,1);

% Material balances

% Overhead receiver

xAdot(1)=(l/md)*(vr*yA(2)-(dist+reflux)*xA(1));

% Rectifying (top)section

for i=2:nfA-1;  
    xAdot(i)=(1/mt)*(lr*xA(i-1)+vr*yA(i+l)-lr*xA(i)-vr*yA(i));
end

% Feed A

xAdot(nfA)=(1/mt)*(lr*xA(nfA-1)+vr*yA(nfA+l)-ls1*xA(nfA)-vr*yA(nfA)+feedA);

% Ratio of fo~'ard reaction rate to reverse reaction rate

K=k1/k2;

% extractive section

for i=nfA+1:nfA+4;  
    xAdot(i)=(1/mt)*(ls1*xA(i-1)+vr*yA(i+l)-ls1*xA(i)-vr*yA(i));
end

% Intermediate Section

for i=nfA+5:nfB-1;  
    xAdot(i)=(1/mt)*(ls1*xA(i-1)-ls1*xA(i)+vr*yA(i+l)-vr*yA(i))+(k1*(xC(i)*xD(i)/K-xA*xB(i)));
end

% Feed B stage reaction
\[
x_{A\dot}(n_fB) = \left(\frac{1}{m_t}\right) \left( l_s x_A(n_fB - l) - l_s x_A(n_fB) + v_s y_A(n_fB + 1) - v_r y_A(n_fB) \right) + (k_1 \times (x_C(n_fB) \times x_D(n_fB) / K - x_A(n_fB) \times x_B(n_fB)))
\]

\% Stripping (bottom) section

\% Reaction in the Stripping Section

\[
x_{A\dot}(n_fB + 1) = \left(\frac{1}{m_t}\right) \left( l_s x_A(n_fB) - l_s x_A(n_fB + 1) + v_s y_A(n_fB + 2) - v_s y_A(n_fB + 1) \right) + (k_1 \times (x_C(n_fB + 1) \times x_D(n_fB + 1) / K - x_A(n_fB + 1) \times x_B(n_fB + 1)))
\]

\% Normal stripping section

for \(i = n_fB + 2 : n_s - 1\):
\[
x_{A\dot}(i) = \left(\frac{1}{m_t}\right) \left( l_s x_A(i - 1) + v_s y_A(i + 1) - l_s x_A(i) - v_s y_A(i) \right)
\]
end

\% Reboiler

\[
x_{A\dot}(n_s) = \left(\frac{1}{m_b}\right) \left( l_s x_A(n_s - l) - l_{bot} x_A(n_s) - v_s y_A(n_s) \right)
\]

\% Function file of Reactant B

function \(x_{B\dot} = dist\_dynB(t, x_B)\)

\% Declaration of some variables

global \(DIST\_PAR\)

global \(k_{li} \ k_l \ k_2 \ 
\text{step}k_l \ 
\text{step}k_1 \ x_A \ x_C \ x_D \ y_A \ y_B \ y_C \ y_D\)

\% Assignment of values to parameters

\(n_s = DIST\_PAR(1)\);
\(n_fA = DIST\_PAR(2)\);
\(n_fB = DIST\_PAR(3)\);
\(feedAi = DIST\_PAR(4)\);
\(feedBi = DIST\_PAR(5)\);
\(zfeedAi = DIST\_PAR(6)\);
\(zfeedBi = DIST\_PAR(7)\);
\(qfA = DIST\_PAR(8)\);
\[
qfB = \text{DIST\_PAR}(9);
\]
\[
\text{refluxi} = \text{DIST\_PAR}(10);
\]
\[
\text{vapori} = \text{DIST\_PAR}(11);
\]
\[
\text{mt} = \text{DIST\_PAR}(24);
\]
\[
\text{md} = \text{DIST\_PAR}(25);
\]
\[
\text{mb} = \text{DIST\_PAR}(26);
\]
\[
\text{Pc} = \text{DIST\_PAR}(35);
\]
\[
\text{Pr} = \text{DIST\_PAR}(36);
\]

if length(DIST\_PAR)==40;
    \[
    \text{stepr} = \text{DIST\_PAR}(27);
    \]
    \[
    \text{tstepr} = \text{DIST\_PAR}(28);
    \]
    \[
    \text{stepv} = \text{DIST\_PAR}(29);
    \]
    \[
    \text{tstepv} = \text{DIST\_PAR}(30);
    \]
    \[
    \text{stepzfA} = \text{DIST\_PAR}(31);
    \]
    \[
    \text{tstepzfA} = \text{DIST\_PAR}(32);
    \]
    \[
    \text{stepfA} = \text{DIST\_PAR}(33);
    \]
    \[
    \text{tstepfA} = \text{DIST\_PAR}(34);
    \]
    \[
    \text{stepzfB} = \text{DIST\_PAR}(37);
    \]
    \[
    \text{tstepzfB} = \text{DIST\_PAR}(38);
    \]
    \[
    \text{stepfB} = \text{DIST\_PAR}(39);
    \]
    \[
    \text{tstepfB} = \text{DIST\_PAR}(40);
    \]
else
    \[
    \text{stepr} = 0; \text{tstepr} = 0; \text{stepv} = 0; \text{tstepv} = 0;
    \]
    \[
    \text{stepzfA} = 0; \text{tstepzfA} = 0; \text{stepfA} = 0; \text{tstepfA} = 0;
    \]
    \[
    \text{stepzfB} = 0; \text{tstepzfB} = 0; \text{stepfB} = 0; \text{tstepfB} = 0;
    \]
end

% Check the disturbances in Reflux

if t<tstepr;
    \[
    \text{reflux} = \text{refluxi};
    \]
else
    \[
    \text{reflux} = \text{refluxi} + \text{stepr};
    \]
end

% Check the disturbances Vapor boil-up

if t<tstepv;
    \[
    \text{vapor} = \text{vapori};
    \]
else
    \[
    \text{vapor} = \text{vapori} + \text{stepv};
    \]
end

% Check for disturbances in Feed A Composition

if t<tstepzfA;
\[ \text{zfeed}_A = \text{zfeed}_{Ai}; \]
\[
\text{else} \\
\text{zfeed}_A = \text{zfeed}_{Ai} + \text{stepzf}_A; \\
\text{end}
\]

\text{% Check for disturbances in Feed B Composition}

\[
\text{if } t < \text{tstepzf}_B; \\
\text{zfeed}_B = \text{zfeed}_{Bi}; \\
\text{else} \\
\text{zfeed}_B = \text{zfeed}_{Bi} + \text{stepzf}_B; \\
\text{end}
\]

\text{% Check for disturbances in Feed A Flowrate}

\[
\text{if } t < \text{tstepf}_A; \\
\text{feed}_A = \text{feed}_{Ai}; \\
\text{else} \\
\text{feed}_A = \text{feed}_{Ai} + \text{stepf}_A; \\
\text{end}
\]

\text{% Check for disturbances in Feed B Flowrate}

\[
\text{if } t < \text{tstepf}_B; \\
\text{feed}_B = \text{feed}_{Bi}; \\
\text{else} \\
\text{feed}_B = \text{feed}_{Bi} + \text{stepf}_B; \\
\text{end}
\]

\text{% Check the disturbances in Rate of Forward Reaction}

\[
\text{if } t < \text{tstepk}_1; \\
\text{k}_1 = \text{k}_{li}; \\
\text{else} \\
\text{k}_1 = \text{k}_{li} + \text{stepk}_1; \\
\text{end}
\]

\text{% Rectifying and stripping section liquid flowrates}

\[
\text{lr} = \text{reflux}; \\
\text{ls}_1 = \text{reflux} + \text{feed}_A * \text{qf}_A; \\
\text{ls} = \text{reflux} + \text{feed}_A * \text{qf}_A + \text{feed}_B * \text{qf}_B; \\
\text{% Rectifying and stripping section vapor flowrates}
\]

\[
\text{vs} = \text{vapor}; \\
\text{vr} = \text{vs} + \text{feed}_A * (1 - \text{qf}_A) + \text{feed}_B * (1 - \text{qf}_B); \\
\]
% Distillate and bottoms rates

dist = vr - reflux;
lbot = ls - vs;

% Zeros the function vector

xBdot=zeros(ns,1);

% Material balances

% Overhead receiver

xBdot(1)=(1/md)*(vr*yB(2)-(dist+reflux)*xB(1));

% Rectifying (top) section

for i=2:nfA-1;
    xBdot(i)=(1/mt)*(lr*xB(i-1)+vr*yB(i+l)-lr*xB(i)-vr*yB(i));
end

% Feed A

xBdot(nfA)=(1/mt)*(lr*xB(nfA-1)+vr*yB(nfA+1)-ls1*xB(nfA)-vr*yB(nfA));

% Ratio of forward reaction rate to reverse reaction rate

K=kl/k2;

% Extractive section

for i=nfA+1 :nfA+4;
    xBdot(i)=(1/mt)*(ls1*xB(i-1)+vr*yB(i+l)-ls1*xB(i)-vr*yB(i));
end

% Ratio of forward reaction rate to reverse reaction rate

K=kl/k2;

% Intermediate Section

for i=nfA+5 :nfB-1;
    xBdot(i)=(1/mt)*(ls1*xB(i-1)-ls1*xB(i)+vr*yB(i+1)-vr*yB(i)+(kl*(xC(i)*xD(i)/K-xA(i)*xB(i)));
end

% Feed B stage reaction
\[
x_{Bdot}(n_B) = \frac{l}{m_t} \left( \text{feed}_B \cdot x_B(n_B-1) + l_s \cdot x_B(n_B) \right) + \frac{(k_l \cdot x_C(n_B) \cdot x_D(n_B) - K \cdot x_A(n_B) \cdot x_B(n_B))}{m_t};
\]

% Stripping (bottom) section

% Normal stripping section

for \( i = n_B + 2 : n_s - 1 \):

\[
x_{Bdot}(i) = \frac{l}{m_t} \left( l_s \cdot x_B(i-1) + v_s \cdot y_B(i+1) - l_s \cdot x_B(i) - v_s \cdot y_B(i) \right);
\]

end

% Reboiler

\[
x_{Bdot}(n_s) = \frac{1}{m_b} \left( l_s \cdot x_B(n_s - 1) - l_{bot} \cdot x_B(n_s) - v_s \cdot y_B(n_s) \right);
\]

% Function file for Product C

function \( x_{Cd}(t, x_C) \)

% Declaration of some variables

global DIST_PAR

global kli k_l

global k2

global stepkl

global tstepkl

global xA xB xD

global yA yB yC yD

% Assignment of values to parameters

\( n_s = \text{DIST}_\text{PAR}(1); \)
\( n_fA = \text{DIST}_\text{PAR}(2); \)
\( n_fB = \text{DIST}_\text{PAR}(3); \)
\( \text{feed}_A = \text{DIST}_\text{PAR}(4); \)
\( \text{feed}_B = \text{DIST}_\text{PAR}(5); \)
\( z_{feed}_A = \text{DIST}_\text{PAR}(6); \)
\( z_{feed}_B = \text{DIST}_\text{PAR}(7); \)
\( q_fA = \text{DIST}_\text{PAR}(8); \)
\( q_fB = \text{DIST}_\text{PAR}(9); \)
\( \text{reflux}_1 = \text{DIST}_\text{PAR}(10); \)
\( \text{vapori} = \text{DIST}_\text{PAR}(11); \)
\( m_t = \text{DIST}_\text{PAR}(24); \)
\( m_d = \text{DIST}_\text{PAR}(25); \)
\( m_b = \text{DIST}_\text{PAR}(26); \)
Pc = DIST_PAR(35);
Pr = DIST_PAR(36);

mt = DIST_PAR(24);
md = DIST_PAR(25);
mb = DIST_PAR(26);

if length(DIST_PAR)==40;

stepr = DIST_PAR(27);
tstepr = DIST_PAR(28);
stepv = DIST_PAR(29);
tstepv = DIST_PAR(30);
stepzfA = DIST_PAR(31);
tstepzfA = DIST_PAR(32);
stepfA = DIST_PAR(33);
tstepfA = DIST_PAR(34);

stepzfB = DIST_PAR(37);
tstepzfB = DIST_PAR(38);
stepm = DIST_PAR(39);
tstepm = DIST_PAR(40);
else
    stepr=0; tstepr=0; stepv=0; tstepv=0;
    stepzfA=0; tstepzfA=0; stepfA=0; tstepfA=0;
    stepzfB=0; tstepzfB=0; stepfB=0; tstepfB=0;
end

% Check for disturbances in. Reflux

if t<tstepr;
    reflux=refluxi;
else
    reflux=refluxi+stepr;
end

% Check for disturbances Vapor boil-up

if t<tstepv;
    vapor=vapori;
else
    vapor=vapori+stepv;
end

% Check for disturbances in Feed A Composition
if \( t < \text{stepzfA} \);
    \( z\text{feedA} = z\text{feedAi} \);
else
    \( z\text{feedA} = z\text{feedAi} + \text{stepzfA} \);
end

% Check for disturbances in Feed B Composition

if \( t < \text{stepzfB} \);
    \( z\text{feedB} = z\text{feedBi} \);
else
    \( z\text{feedB} = z\text{feedBi} + \text{stepzfB} \);
end

% Check for disturbances in Feed A Flowrate

if \( t < \text{stepfA} \);
    \( \text{feedA} = \text{feedAi} \);
else
    \( \text{feedA} = \text{feedAi} + \text{stepfA} \);
end

% Check for disturbances in Feed B Flowrate

if \( t < \text{stepfB} \);
    \( \text{feedB} = \text{feedBi} \);
else
    \( \text{feedB} = \text{feedBi} + \text{stepfB} \);
end

% Check for disturbances in Rate of Forward Reaction

if \( t < \text{stepkl} \);
    \( \text{kl} = \text{kli} \);
else
    \( \text{kl} = \text{kli} + \text{stepkl} \);
end

% Rectifying and stripping section liquid flowrates

\( \text{lr} = \text{reflux} \);
\( \text{lsl} = \text{reflux} + \text{feedA} \times qfA \);
\( \text{ls} = \text{reflux} + \text{feedA} \times qfA + \text{feedB} \times qfB \);

% Rectifying and stripping section vapor flowrates
\( \text{vs} = \text{vapor}; \)

\( \text{vr} = \text{vs} + \text{feedA}*(1-\text{qfA}) + \text{feedB}*(1-\text{qfB}); \)

\% Distillate and bottoms rates

\( \text{dist} = \text{vr} - \text{reflux}; \)
\( \text{lbot} = \text{Is} - \text{vs}; \)

\% Zeros the function vector

\( \text{xCdot}=\text{zeros}(\text{ns}, 1); \)

\% Material balances

\( \text{xCdot}(1)=(1/\text{md})*(\text{vr}^{*}\text{yC}(2) - (\text{dist} + \text{reflux}^{*}\text{xC}(1)); \)

\% Rectifying section

\text{for i=2 :nfA - 1;}
\( \text{xCdot}(i)=(1/\text{mt})*(\text{lr}^{*}\text{xC}(i-1) + \text{vr}^{*}\text{yC}(i+1) - \text{lr}^{*}\text{xC}(i) - \text{vr}^{*}\text{yC}(i)); \)
\text{end}

\% Feed A

\( \text{xCdot}(\text{nfA})=(1/\text{mt})*(\text{lr}^{*}\text{xC}(\text{nfA}) + \text{vr}^{*}\text{yC}(\text{nfA}-1) - \text{ls1}^{*}\text{xC}(\text{nfA}) - \text{vr}^{*}\text{yC}(\text{nfA}); \)

\% Ratio of forward reaction rate to reverse reaction rate

\( \text{K}=\text{k1}/\text{k2} \)

\% Extractive section

\text{for i=nfA+ 1 :nfA+4;}
\( \text{xCdot}(i)=(1/\text{mt})*(\text{ls1}^{*}\text{xC}(i-1) + \text{vr}^{*}\text{yC}(i+1) - \text{ls1}^{*}\text{xC}(i) - \text{vr}^{*}\text{yC}(i)); \)
\text{end}

\% Ratio of forward reaction rate to reverse reaction rate

\( \text{K}=\text{k1}/\text{k2}; \)

\% Intermediate Section

\text{for i=nfA+ 1 :nB- 1;}
\( \text{xCdot}(i)=(1/\text{mt})*(\text{ls1}^{*}\text{xC}(i-1) - \text{ls1}^{*}\text{xC}(i) + \text{vr}^{*}\text{yC}(i+1) - \text{vr}^{*}\text{yC}(i)) + \text{feedA}^{*}\text{xA}(i) + (\text{kl}(\text{xA}(i) - \text{xB}(i)^{*}\text{xC}(i) - \text{xD}(i)/\text{k}); \)
\text{end}
% Feed B stage reaction

\[
x_{\text{Cdot}}(n_f B) = \frac{1}{r_{\text{nt}}} (\text{feedB} \cdot z_{\text{feedB}} + \text{ls1} \cdot x_{\text{C}}(n_f B - 1) - \text{ls} \cdot x_{\text{C}}(n_f B) + \text{vs} \cdot y_{\text{C}}(n_f B + 1) - \text{vr} \cdot y_{\text{C}}(n_f B)) + \text{kl}^* (x_A(n_f B) \cdot x_B(n_f B) - x_{\text{C}}(n_f B) \cdot x_D(n_f B)/K);
\]

% Stripping (bottom) section

% Reaction ill the Stripping Section

\[
x_{\text{Cdot}}(n_f B + 1) = \frac{1}{m_t} \cdot (\text{ls} \cdot x_{\text{C}}(n_f B) - \text{ls} \cdot x_{\text{C}}(n_f B + 1) + \text{vr} \cdot y_{\text{C}}(n_f B + 2) - \text{vr} \cdot y_{\text{C}}(n_f B + 1)) + (1:1 \cdot (x_A(n_f B + 1) \cdot x_B(n_f B + 1) - x_{\text{C}}(n_f B + 1) \cdot x_D(n_f B + 1)/K));
\]

% Normal stripping section

for i=n_f B+2:n_s-1;
    x_{\text{Cdot}}(i) = \frac{1}{m_t} \cdot (\text{ls} \cdot x_{\text{C}}(i-1) + \text{vs} \cdot y_{\text{C}}(i+1) - \text{ls} \cdot x_{\text{C}}(i) - \text{vs} \cdot y_{\text{C}}(i));
end

% Reboiler

\[
x_{\text{CAot}(m)} = \frac{1}{r_{\text{nb}}} (\text{ls} \cdot x_{\text{C}}(n_s - 1) - \text{lb}_\text{ot} \cdot x_{\text{C}}(n_s) - \text{vs} \cdot y_{\text{C}}(n_s));
\]

% Function file the Product D

function x_{Ddot} = dist_dynD(t, x_D)

% Declaration of some variables

global DIST_PAR
global kl kl
global k2
global stepkl
global tstepkl
global x_A x_B x_C
global y_A y_B y_C y_D

% Assignment of values to variables

ns = DIST_PAR(1);
nf_A = DIST_PAR(2);
nf_B = DIST_PAR(3);
feedAi = DIST_PAR(4);
feedBi = DIST_PAR(5);
zfeedAi = DIST_PAR(6);
zfeedBi = DIST_PAR(7);
qf_A = DIST_PAR(8);
qf_B = DIST_PAR(9);
refluxi = DIST_PAR(10);
vapori = DIST_PAR(11);
mt = DIST_PAR(24);
md = DIST_PAR(25);
mb = DIST_PAR(26);

Pc = DIST_PAR(35);
Pr = DIST_PAR(36);

if length(DIST_PAR)==40;
    stepr = DIST_PAR(27);
tstepr = DIST_PAR(28);
stepv = DIST_PAR(29);
tstepv = DIST_PAR(30);
stepzfA = DIST_PAR(31);
tstepzfA = DIST_PAR(32);
stepfA = DIST_PAR(33);
tstepfA = DIST_PAR(34);
stepzfB = DIST_PAR(37);
tstepzfB = DIST_PAR(38);
stepfB = DIST_PAR(39);
tstepfB = DIST_PAR(40);
else
    stepr=0; tstepr=0; stepv=0; tstepv=0;
    stepzfA=0; tstepzfA=0; stepfA=0; tstepfA=0;
    stepzfB=0; tstepzfB=0; stepfB=0; tstepfB=0;
end

% Check for disturbances in Reflux

if t<tstepr;
    reflux=refluxi;
else
    reflux=refluxi+stepr;
end

% Check for disturbances Vapor boil-up

if t<tstepv;
    vapor=vapori;
else
    vapor=vapori+stepv;
end

% Check for disturbances in Feed A Composmon

if t<tstepzfA;
    zfeedA=zfeedAi;
else
    zfeedA=zfeedA\text{i}+stepzfA;
end

% Check for disturbances in Feed B Composition
if t<tstepzfB;
    zfeedB=zfeedB\text{i};
else
    zfeedB=zfeedB\text{i}+stepzfB;
end

% Check for disturbances in Feed A Flowrate
if t<tstepfA;
    feedA=feedA\text{i};
else
    feedA=feedA\text{i}+stepfA;
end

% Check for disturbances in Feed B Flowrate
if t<tstepfB;
    feedB=feedB\text{i};
else
    feedB=feedB\text{i}+stepfB;
end

% Check for disturbances in Rate of Forward Reaction
if t<tstepkl;
    kl=kl\text{i};
else
    kl=kl\text{i}+stepkl;
end

% Rectifying and stripping section liquid flowrates
lr = reflux;
ls1 = reflux+feedA*qfA;
ls = reflux+feedA*qfA+feedB*qfl3;

% Rectifying and stripping section vapor flowrates
vs = vapor;
vr = vs + feedA*(1-qfA) + feedB*(1-qtB);
% Distillate and bottoms rates

dist = vr - reflux;
lbot = ls - vs;

% Zeros the function vector

xDdot=zeros(ns, 1);

% Material balances

% Overhead r-ceiver

xDdot(1)=(1/md)*(vr*yD(2)-(dist+reflux)*xD(1));

% Rectifying (top)section

for i=2:nfA-1;
    xDdot(i)=(1/mt)*(lr*xD(i-1)+vr*yD(i+1)-lr*xD(i)-vr*yD(i));
end

% Feed A

xDdot(nfA)=(1/mt)*(lr*xD(nfA-1)+vr*yD(nfA+1)-ls1*xD(nfA)-vr*yD(nfA));

% Ratio of forward reaction rate to reverse reaction rate

K=kl/k2;

% Extractive section

for i=nfA+1 :nfA+4;
    xDdot(i)=(1/mt)*(ls1*xD(i-1)+vr*yD(i+1)-ls1*xD(i)-vr*yD(i));
end

% Ratio of forward reaction rate to reverse reaction rate

K=k1/k2

% Intermediate Section

for i=nfA+1 :nfB-1;
    xDdot(i)=(1/mt)*(ls1*xD(i-1)-ls1*xD(i)+vr*yD(i+1)-vr*yD(i))+feedA*xA(i)+(kl*(xA(i))^B(i)-xC(i)^xD(i)/K);
end

% Feed B stage reaction
\[ \dot{x}_{D}(n_{B}) = \left( \frac{l}{m_{t}} \right) \left( \text{feed}_{D} \cdot x_{D}(n_{B}) \right) + \left( \frac{l_{s}}{m_{t}} \right) \left( \text{feed}_{D} \cdot x_{D}(n_{B} - 1) \right) + \left( \frac{v_{s}}{m_{t}} \right) \left( y_{D}(n_{B} + 1) \right) + \left( \frac{k_{l}}{m_{t}} \right) \left( x_{A}(n_{B}) \cdot x_{B}(n_{B}) \right) \cdot \left( x_{C}(n_{B}) \cdot x_{D}(n_{B}) \right) \] 

% Stripping (bottom) section

% Reaction in the Stripping Section

\[ \dot{x}_{D}(n_{B} + 1) = \left( \frac{l}{m_{t}} \right) \left( l_{s} \cdot x_{D}(n_{B}) - l_{s} \cdot x_{D}(n_{B} + 1) + v_{s} \cdot y_{D}(n_{B} + 2) - v_{s} \cdot y_{D}(n_{B} + 1) \right) + \left( \frac{k_{l}}{m_{t}} \right) \left( x_{A}(n_{B} + 1) \cdot x_{B}(n_{B} + 1) \right) \] 

% Normal stripping section

for \( i = n_{B} + 2 : n_{s} - 1 \) 
\[ \dot{x}_{D}(i) = \left( \frac{l}{m_{t}} \right) \left( l_{s} \cdot x_{D}(i) + v_{s} \cdot y_{D}(i + 1) - l_{s} \cdot x_{D}(i) - v_{s} \cdot y_{D}(i) \right) \] 
end

% Reboiler
\[ \dot{x}_{D} = \left( \frac{l}{m_{b}} \right) \left( l_{s} \cdot x_{D}(n_{s}) + l_{b} \cdot x_{D}(n_{s}) + v_{s} \cdot y_{D}(n_{s}) \right) \]

2. Feedbacksetpoint.m

% For Setpoint
% Step
\[ K_{1} = 2.45; T_{1} = 4.9755; d_{1} = 26.805; \]

% IAE
\[ K_{c} = \left( \frac{0.758}{K_{1}} \right) \left( (d_{1}/T_{1})^{0.861} \right); T_{i} = T_{1} / (1.02 - 0.323*(d_{1}/T_{1})); T_{d} = 0; \]
\[ \text{sim}('feedback'); \]
\[ \text{figure(1)}; \]
\[ \text{plot(time,T3)}; \]
\[ \text{title('Close Loop IAE Response-Setpoint')}; \]
\[ \text{xlabel('Time(min)')}; \]
\[ \text{ylabel('Composition i-butane (mass fraction)')}; \]
\[ \text{grid on}; \]
\[ \text{figure(2)}; \]
\[ \text{plot(time,Error)}; \]
\[ \text{title('Error')}; \]
\[ \text{xlabel('Time(min)')}; \]
\[ \text{ylabel('Error')}; \]
\[ \text{grid on}; \]
\[ \text{figure(3)}; \]
\[ \text{plot(time,Mv)}; \]
\[ \text{title('Manipulated variable')}; \]
\[ \text{xlabel('Time(min)')}; \]
\[ \text{ylabel('mv')}; \]
\[ \text{grid on}; \]
% ITAE
K_c=(0.586/K1)^((d1/T1)^(0.916)); T_i=T_1/(1.03-0.165*(d1/T_1)); T_d=0;
sim('feedback');
figure(4);
plot(time,T_3);
title('Close Loop ITAE Response-Setpoint');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(5);
plot(time,Error);
title('Error')
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(6);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('mv');
grid on;

% IMC
K_c=(1/K1)^((T_1/T_1+d1)); T_i=T_1; T_d=0;
sim('feedback');
figure(7);
plot(time,T_3);
title('Close Loop IMC Response-Setpoint');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(8);
plot(time,Error);
title('Error')
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(9);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('mv');
grid on;
%HA
Kc=(0.14/K1)+(0.28*T1/d1*K1);Ti=0.33*d1+(6.8*d1*T1/(10*d1+T1));Td=0;
sim('feedback');
figure(10);
plot(time,T3);
title('Close Loop HA Response-Setpoint');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(11);
plot(time,Error);
title('Error')
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(12);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('mv');
grid on;

%ISE
Kc=(1.495/K1)*(d1/T1)^(-0.945);Ti=(T1/(1.101)*(d1/T1)^0.771);Td=0;
sim('feedback');
figure(13);
plot(time,T3);
title('Close Loop ISE Response');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(14);
plot(time,Error);
title('Error')
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(15);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('mv');
grid on;
% Cohen Coon
\[ K_c = \frac{T_1}{(K_1 \cdot d_1)} \times (0.9 + d_1)/(12 \cdot T_1); T_i = d_1 \times (30 + (3 \cdot d_1/T_1)/(9 + 20 \cdot d_1)/T_1); T_d = 0; \]

sim('feedback');
figure(16);
plot(time, T3);
title('Close Loop Cohen Coon Response - Setpoint');
xlabel('Time (min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(17);
plot(time, Error);
title('Error');
xlabel('Time (min)');
ylabel('Error');
grid on;
figure(18);
plot(time, Mv);
title('Manipulated variable');
xlabel('Time (min)');
ylabel('Mv');
grid on;

% Ziegler and Nichols (1942), Model Method 2
\[ K_c = \frac{(0.9 \cdot T_1)/(K_1 \cdot d_1)}{T_i = d_1 \times 3.33; T_d = 0}; \]

sim('feedback');
figure(19);
plot(time, T3);
title('Close Loop Ziegler and Nichols Response - Model Method 2');
xlabel('Time (min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(20);
plot(time, Error);
title('Error');
xlabel('Time (min)');
ylabel('Error');
grid on;
figure(21);
plot(time, Mv);
title('Manipulated variable');
xlabel('Time (min)');
ylabel('Mv');
grid on;
% Hazebroek and Van der Waerden (1950), Model Method 2
Kc = (0.621386 * T1) / (K1 * d1); Ti = d1 * 7.62715178; Td = 0;
sim('feedback');
figure(22);
plot(time, T3);
title('Close Loop Hazebroek and Van der Waerden Response - Model Method 2');
xlabel('Time (min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(23);
plot(time, Error);
title('Error');
xlabel('Time (min)');
ylabel('Error');
grid on;
figure(24);
plot(time, Mv);
title('Manipulated variable');
xlabel('Time (min)');
ylabel('mv');
grid on;

% Chien (1952), Servo, Model: Method 2, 0% overshoot
Kc = (0.35 * T1) / (K1 * d1); Ti = 1.17 * T1; Td = 0;
sim('feedback');
figure(25);
plot(time, T3);
title('Close Loop Chien (1952), Servo, Model: Method 2, 0% Overshoot');
xlabel('Time (min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(26);
plot(time, Error);
title('Error');
xlabel('Time (min)');
ylabel('Error');
grid on;
figure(27);
plot(time, Mv);
title('Manipulated variable');
xlabel('Time (min)');
ylabel('mv');
grid on;
% Chien (1952), Servo, Model: Method 2, 20% overshoot
Kc=(0.6*T1)/(K1*d1); Ti=T1; Td=0;
sim('feedback');
figure(28);
plot(time,T3);
title('Close Loop Chien (1952), Servo, Model: Method 2, 20% overshoot');
xlabel('Time (min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(29);
plot(time,Error);
title('Error');
xlabel('Time (min)');
ylabel('Error');
grid on;
figure(30);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time (min)');
ylabel('mv');
grid on;

%Cohen and Coon (1953), Model: Method 2
Kc=(1/K1)*((0.9*(T1/d1)+0.083); Ti=T1*((3.33*(d1/T1)+0.31*(d1/T1)^2)/(1+2.22*(d1/T1))); Td=0;
sim('feedback');
figure(31);
plot(time,T3);
title('Close Loop Cohen and Coon (1953), Model: Method 2');
xlabel('Time (min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(32);
plot(time,Error);
title('Error');
xlabel('Time (min)');
ylabel('Error');
grid on;
figure(33);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time (min)');
ylabel('mv');
grid on;
Two Constraints Method - Wolfe (1951), Model: Method 3

\[ K_c = \frac{4.3740102 \times T_1}{K_1 \times d_1}; \quad T_i = 3.63221264 \times d_1; \quad T_d = 0; \]

```matlab
sim('feedback');
figure(34);
plot(time,T3);
title('Close Loop Two Constraints Method - Wolfe (1951), Model: Method 3');
xlabel('Time (min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(35);
plot(time,Error);
title('Error');
xlabel('Time (min)');
ylabel('Error');
grid on;
figure(36);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time (min)');
ylabel('mv');
grid on;
```

Two Constraints Criterion - Murrill (1967), Model: Method 4

\[ K_c = \frac{0.928}{K_1} \times \frac{(T_1/d_1)^{0.946}}{T_i = (T_1/1.078)^{0.583} \times (d_1/T_1)^{0.583}; \quad T_d = 0; \]

```matlab
sim('feedback');
figure(37);
plot(time,T3);
title('Close Loop Two Constraints Criterion - Murrill (1967), Method: Method 4');
xlabel('Time (min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(38);
plot(time,Error);
title('Error');
xlabel('Time (min)');
ylabel('Error');
grid on;
figure(39);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time (min)');
ylabel('mv');
```
grid on;

% McMillan (1994), Model: Method 4
Kc=K1/3; Ti=d1; Td=0;
sim('feedback');
figure(40);
plot(time,T3);
title('Close Loop McMillan (1994), Model: Method 4');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(41);
plot(time,Error);
title('Error')
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(42);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('Mv');
grid on;

% St. Clair (1997), Model: Method 4
Kc=(0.333*T1)/(K1*d1); Ti=T1; Td=0;
sim('feedback');
figure(43);
plot(time,T3);
title('Close Loop St. Clair (1997), Model: Method 4');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(44);
plot(time,Error);
title('Error')
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(45);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('mv');
grid on;

Kc=(0.667*T1)/(K1*d1);Ti=3.78*T1;Td=0;
sim('feedback');
figure(46);
plot(time,T3);
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(47);
plot(time,Error);
title('Error')
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(48);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('mv');
grid on;

% Hay (1998) Servo Tuning 1, Model: Method 2
Kc=(6/K1);Ti=9.5*d1;Td=0;
sim('feedback');
figure(49);
plot(time,T3);
title('Close Loop Hay (1998), Servo Tuning 1, Model: Method 2');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(50);
plot(time,Error);
title('Error')
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(51);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('mv');
grid on;

% Hay (1998) Servo Tuning 2, Model: Method 2
Kc=(4/K1);Ti=10*d1;Td=0;
sim('feedback');
figure(52);
plot(time,T3);
title('Close Loop Hay (1998) Servo Tuning 2, Model : Method 2');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(53);
plot(time,Error);
title('Error');
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(54);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('mv');
grid on;

% Servo Tuning
% Minimum IAE- Rovira et al. (1969), Model: Method 4
Kc=(0.758/K1)*((T1/d1)^0.861);Ti=T1/(1.020-0.323*(d1/T1));Td=0;
sim('feedback');
figure(55);
plot(time,T3);
title('Close Loop Minimum IAE-Rovira et al. (1969), Model : Method 4');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(56);
plot(time,Error);
title('Error');
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(57);
plot(time,Mv);
% Minimum IAE - Marlin (1995), Model: Method 1
Kc = 1.4/K1; Ti = 0.72*d1; Td = 0;
sim('feedback');
figure(58);
plot(time,T3);
title('Close Loop minimum IAE - Marlin (1995), Model: Method 1');
xlabel('Time(min)');
ylabel('Composition i-butane (mass fraction)');
grid on;
figure(59);
plot(time,Error);
title('Error');
xlabel('Time(min)');
ylabel('Error');
grid on;
figure(60);
plot(time,Mv);
title('Manipulated variable');
xlabel('Time(min)');
ylabel('mv');
grid on;