Design of Simplified Model Predictive Controller

By

Mohd Ikram Solehuddin Bin Borhan

Dissertation submitted in partial fulfillment

of the requirements for the

Bachelor of Engineering (Hons)

(Chemical Engineering)

May 2013

Universiti Teknologi PETRONAS

Bandar Seri Iskandar

31750 Tronoh

Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Design of Simplified Model Predictive Controller

by

Mohd Ikram Solehuddin B Borhan

A project dissertation submitted to the

Chemical Engineering Programme

Universiti Teknologi PETRONAS

In partial fulfillment of the requirement for the

BACHELOR OF ENGINEERING (Hons)

(CHEMICAL ENGINEERING)

Approved by,

____

(AP Dr. M Ramasamy)

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

MAY 2013

CERTIFICATION OF ORIGINALITY

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.

MOHD IKRAM SOLEHUDDIN BIN BORHAN

ABSTRACT

There are a lot of control strategies being derived from model predictive control (MPC) concept such as Internal Model Control (IMC). In MPC, inverse of process transfer functions is required in obtaining the control law. However, an exact inverse transfer function can never be obtained due to certain conditions that lead to physically unrealizable processes such as dead time, numerator dynamics, constraints and model mismatch. New technique known as Simplified Model Predictive Control (SMPC) was developed which solves the problem of acquiring exact inverse response of a model to predict the future outputs of the corresponding inputs. SMPC control algorithm is an efficient and simple method for multivariable control. SMPC algorithm for 2 x 2 system of distillation column model developed by Saniye and Suleiman (2011) is designed using MATLAB simulation. This SMPC only concerns for set point tracking of the model.

ACKNOWLEDGEMENT

I wish to express my sincere and deepest gratitude to my final year project supervisor, **AP Dr M Ramasamy** for his prized supervision and advices throughout the entire project. His provisions, tolerance, and willingness to assist me in the problem or complications that I confronted in my project have contributed tremendously to my project. His ideas and moral supports that make the project a success to achieve its objectives. Thanks for his valuable advice, encouragement and patience, I am able to improve in so many ways especially in the research field.

Last but not least, I would like to give my utmost gratitude and appreciation to my family and fellow friends who provide a lot of helps during conducting experiments, and other essential thing. With their provision and inspiration, this project can be accomplished successfully.

Thank you.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

There are difficulties in obtaining the inverse of a process model due to the presence of characteristics that make an exact inverse process is physically unrealizable process. A control strategy that does not need an inverse of a process is required in controlling packed distillation column in separating mixture of methanol-ethanol-*n* butanol-isoamine alcohol-anisol.

Furthermore, distillation consumes the largest energy consumption, which is about 30% to 50% and will be reduced until 15% if using an appropriate control (Riggs, 2000). A conventional control strategy, PI (Proportional Integral) controller, is so far used in distillation control which has satisfied results in distillation control, but in a large multi input multi output configuration, the performance control is very poor (Wahid and Ahmad, 2008). Therefore, application of advanced control, such as SMPC is the best option.

1.2 Objective

The objective of this research is to design SMPC that can be implemented in a distillation column.MPC is designed using MATLAB as it involves transfer functions that have to be considered and computing time can be reduced as in designing SMPC addresses the input-output model, constraints, disturbances prediction and sampling period. This design helps to minimize the energy consumption issue caused by operating a distillation column.

1.3 Scope of Study

In this study, the main subjects under investigation are:

i. Set point changes
ii. Process model sensitivity and robustness

The details of the scope of study will be discussed in Chapter 3.

1.4 Project Background

MPC applications first utilization was recorded in late 1950s. Based on Åström and Witten mark (1984, p. 3), it was cited that March 12, 1959 as the first day when a computer control system went online at Texaco Refinery in Port Arthur, Texas. This computer control system was employed in calculating optimal operating point for a process unit.

In MPC, there are lots of control strategies being derived based on MPC strategy. In MPC concept, the inverse of process transfer function is obtained in predicting the future output of an input into particular process. However, according to Seborg et al. (2004) an exact inverse transfer function can never be obtained due to certain conditions such as:

i. Dead time
ii. Numerator dynamics
iii. Constraints
iv. Model Mismatch

These conditions make control strategies become physically unrealizable control strategies. Thus, the inverse of a process model is approximated numerically. Figure 1 shows how prediction of an output of a process model is applied. $G_{cp}(s)$ is the transfer function of a controller, $G_p(s)$ is the transfer function of the process, $G_m(s)$ is the transfer function of the inverse process model, $G_d(s)$ is the transfer function representing the disturbance, D(s) is the disturbance disturbing the process, SP(s) is the set point, CV(s) is the controlled variable and $E_m(s)$ is the error between actual controlled variable values and inverse model $G_m(s)$ output.

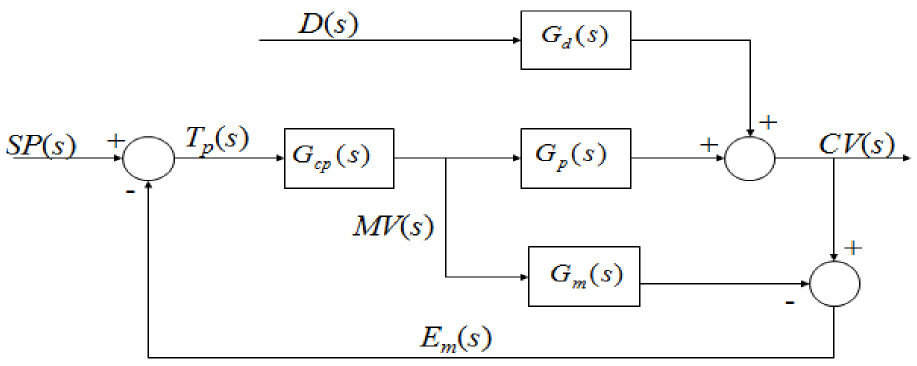

Figure 1: Block diagram of model based predictive control strategy

Ways of implementing MPC is described as below:

1) An appropriate model is used to predict the output behavior of a plant over a future time interval or normally known as the prediction horizon (P). For a discrete time model this means it predicts the plant output from $\hat{y}(k+1)$ to $\hat{y}(k+H_p)$based on all actual past control inputs $u(k)$,$u(k-1)$,..., $u(k-j)$and the available current information $y(k)$.

2) A sequence of control actions adjustments $\Delta u(k|k-1)\ldots$ $\Delta u(k+m|k-1)$ to be implemented over a specified future time interval, which is known as the control horizon (m) is calculated by minimizing some specified objectives such as the deviation of predicted output from setpoint over the prediction horizon and the size of control action adjustments in driving the process output to target plus some operating constraints. However, only the first move of computed control action sequence is implemented while the other moves are discarded. The entire process step is repeated at the subsequent sampling time. This theory is known as the receding horizon theory.

3) A nominal MPC is impossible, or in other words that no model can constitute a perfect representation of the real plant. Thus, the prediction error, ε(k)between the plant measurement $y_m(k)$and the model prediction $\hat{y}(k)$will

always occur. The ε(k)obtained is normally used to update the future prediction.

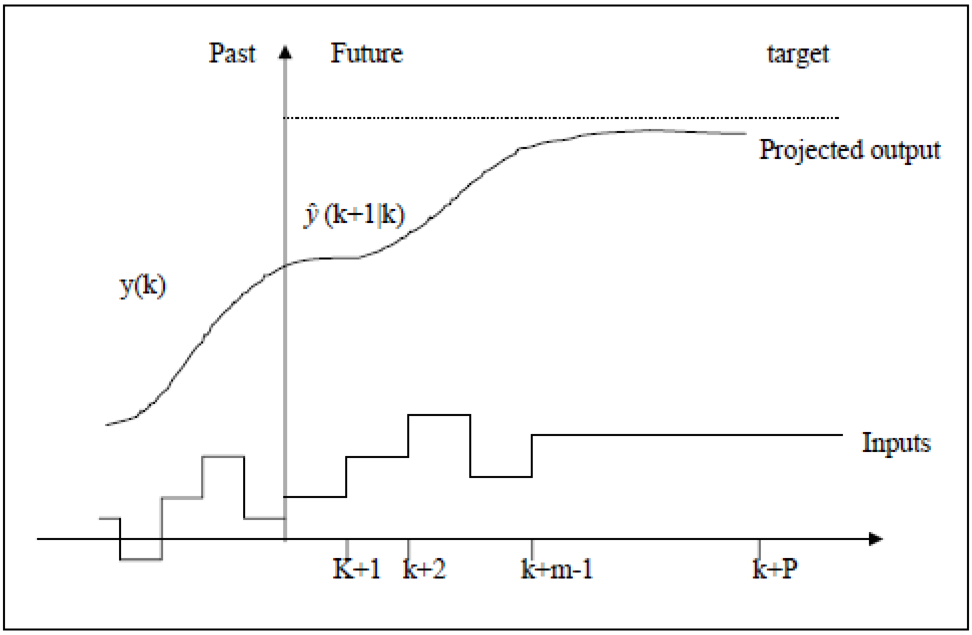

Figure 2: Graph showing the projected output and inputs into the process

Internal Model Control (IMC) strategy is derived due to the conditions that restrict in obtaining an exact inverse of a model. According to Brosilow (1979) and Garcia et al (1983), IMC approach segregates and eliminates properties of model that make an inverse process model to become physically unrealizable process. The process model which is represented as $G_m(s)$ is separated into two parts; each has the invertible and non invertible part. Invertible part is represented by $G_m^-(s)$ meanwhile the non invertible part is represented by $G_m^+(s)$.The process model Gm(s) is factored into these two factors as shown below:

$$G_m(s) = G_m^+(s)G_m^-(s) \qquad \textbf{(1.1)}$$

As the non invertible part of the process model is removed in obtaining the transfer function of a controller, this controller design is then physically realizable and internally stable.

According to Seborg et al (2004), however, the drawbacks of the $G_{cp}(s)$ are the controller involves first, second and third order derivatives of the feedback signal. Next, these derivatives cannot be calculated exactly but can be approximated numerically. Then, controller cannot be used without modifications to make it a proper controller. Thus, in IMC, a filter is required in order to make the design of a controller to be a proper or semi proper controller. A filter transfer function model is represented by $G_f(s)$. The block diagram of a proper or semi proper controller is shown below:

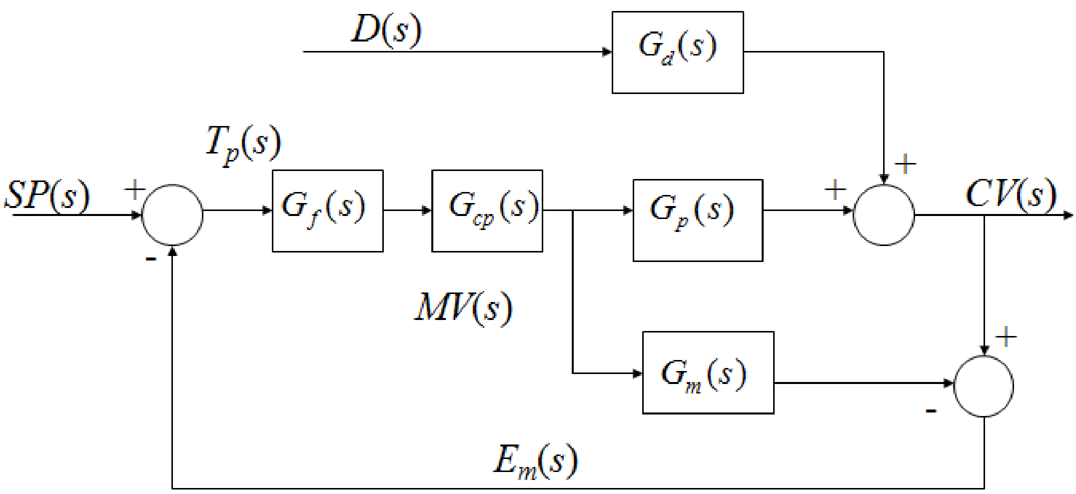

Figure 3: Filter is added to model based predictive control

A proper or semi proper controller design is obtained by equation as shown below:

$$G_{cp}(s) = \tilde{G}_{cp}(s) = [G_m^-(s)]^{-1} G_f(s) \tag{1.2}$$

For set point tracking changes, a filter transfer function that is applied in the in the controller design as shown as below:

$$G_f(s) = \frac{1}{[\lambda s+1]^n} \tag{1.3}$$

Semi proper controller is described as controller having similar order of s for numerator and denominator, meanwhile for proper controller design is described as controller having higher order of s for denominator compared to numerator.

SMPC has the benefits of model predictive control but it does not require inverse of a process in obtaining the future output of that particular process.

CHAPTER 2

LITERATURE REVIEW

2.1 Simplified Model Predictive Control

SMPC control algorithm is developed based on below block diagram. In this block diagram R (z) is the input set point, meanwhile E (z) is the error between actual controlled variable value and the set point, D (z) is the control algorithm may be of the PID type or it may be one of the z transform based control algorithm (Deshpande and Ash, 1981), G(z) is the process transfer function.

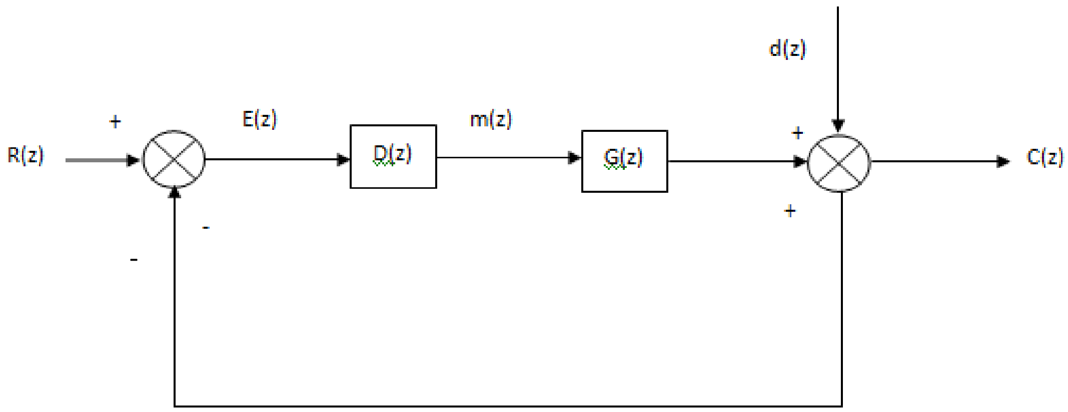

Figure 4: Block diagram of multivariable system

Derivation as shown below is based on Arulalan and Deshpande (1985), vector and matrices are indicated by boldface letters. The method is illustrated for 2x2 system. Assumption is made the process is an open loop stable. The normalized open loop response of the multivariable system is

$$\boldsymbol{C} = \boldsymbol{G}(\boldsymbol{K})^{-1}\boldsymbol{M} \qquad \textbf{(1.4)}$$

Where $\mathbf{(K)}^{-1}$ is inverse of process gain matrix

The closed loop pulse transfer function of the system is

$$\boldsymbol{C} = ((\boldsymbol{I} + \boldsymbol{GD})^{-\mathbf{1}})\boldsymbol{GDR} \tag{2}$$

Ratio $\boldsymbol{C}/\boldsymbol{R}$does not exist for multivariable systems since **C** and **R** are matrices. However, it is still possible to define the closed loop transfer function matrix as (Kuo, 1983)

$$\boldsymbol{P} = ((\boldsymbol{I} + \boldsymbol{GD})^{-\mathbf{1}})\boldsymbol{GD} \tag{3}$$

The closed loop response may be evaluated by

$$\boldsymbol{C} = \boldsymbol{PR} \tag{4}$$

The normalized open loop transfer function matrix is defined as

$$\boldsymbol{Q} = (\boldsymbol{G}(\boldsymbol{K}^{-\mathbf{1}})) \tag{5}$$

Then the normalized open loop response may be obtained by

$$\boldsymbol{C} = \boldsymbol{QM} \tag{6}$$

It should always be possible to design a control algorithm which will give a set point response having the same dynamics as open loop response. Thus, equation 29 and 31 are equated to give

$$(\boldsymbol{I} + \boldsymbol{GD})^{-\mathbf{1}}\,\boldsymbol{GDR} = \boldsymbol{G}(\boldsymbol{K})^{-\mathbf{1}} \tag{7}$$

Premultiplying each side of equation 7 by (I+GD) and then post multiplying by K yields

$$\boldsymbol{GDK} = \boldsymbol{G} + \boldsymbol{GDG} \tag{8}$$

Or

$$\boldsymbol{DK} = \boldsymbol{I} + \boldsymbol{DG} \tag{9}$$

Solution of equation 9 for D is

$$\boldsymbol{D} = (\boldsymbol{K} - \boldsymbol{G})^{-\mathbf{1}} \tag{10}$$

From Figure 4

$$\boldsymbol{M} = \boldsymbol{DE} \tag{11}$$

Combining equation 10 and 11 gives

$$\boldsymbol{M} = (\boldsymbol{K} - \boldsymbol{G})^{-\mathbf{1}}\boldsymbol{E} \tag{11a}$$

$$\boldsymbol{KM} = \boldsymbol{E} + \boldsymbol{GM} \tag{12}$$

Or

$$\boldsymbol{M} = (\boldsymbol{K})^{-\mathbf{1}}\boldsymbol{E} + (\boldsymbol{K})^{-\mathbf{1}}\,\boldsymbol{GM} \tag{13}$$

$$(\boldsymbol{K})^{-\mathbf{1}} = \mathbf{k} \tag{14}$$

Equation 13 becomes

$$\boldsymbol{M} = \boldsymbol{kE} + \boldsymbol{kGM} \tag{15}$$

Z transform operator is introduced and system of equations in equation 15 is expanded to give

$$M_1(z) = k_{11}E_1(z) + k_{12}E_1(z) + k_{11}[G_{11}(z)M_1(z) + G_{12}M_2(z)] + k_{12}[G_{21}(z)M_1(z) + G_{22}M_2(z)] \tag{16}$$

And

$$M_2(z) = k_{21}E_2(z) + k_{22}E_2(z) + k_{21}[G_{11}(z)M_1(z) + G_{12}M_2(z)] + k_{22}[G_{21}(z)M_1(z) + G_{22}M_2(z)] \tag{17}$$

Process transfer function in equation 42 can be represented with the aid of impulse coefficients as (Despande, 1985)

$$G_{ij}(z) = h_{ij}^1 Z^{-1} + h_{ij}^2 Z^{-2} + \cdots + h_{ij}^N Z^{-N} \tag{18}$$

With i and j = 1 and 2. Then equation 17 and 18 become

$$M_1(z) =$$
$$k_{11}E_1(z) + k_{12}E_1(z) + k_{11}[(h_{11}^1 Z^{-1} + h_{11}^2 Z^{-2} + \cdots + h_{11}^N Z^{-N})M_1(z) +$$

$(h_{12}^{1}Z^{-1} + h_{12}^{2}Z^{-2} + \cdots + h_{12}^{N}Z^{-N})M_2(z)] + k_{12}[(h_{21}^{1}Z^{-1} + h_{21}^{2}Z^{-2} + \cdots + h_{21}^{N}Z^{-N}(z))M_1(z) + (h_{22}^{1}Z^{-1} + h_{22}^{2}Z^{-2} + \cdots + h_{22}^{N}Z^{-N})M_2(z)]$ **(19)**

And

$$M_2(z) = k_{21}E_1(z) + k_{22}E_1(z) + k_{21}[(h_{11}^{1}Z^{-1} + h_{11}^{2}Z^{-2} + \cdots + h_{11}^{N}Z^{-N})M_1(z) + (h_{12}^{1}Z^{-1} + h_{12}^{2}Z^{-2} + \cdots + h_{12}^{N}Z^{-N})M_2(z)] + k_{22}[(h_{21}^{1}Z^{-1} + h_{21}^{2}Z^{-2} + \cdots + h_{21}^{N}Z^{-N}(z))M_1(z) + (h_{22}^{1}Z^{-1} + h_{22}^{2}Z^{-2} + \cdots + h_{22}^{N}Z^{-N})M_2(z)] \quad \textbf{(20)}$$

Equation 19 and 20 can be inverted into time domain to give

$$M_1^n = k_{11}\ E_1^n(z) + k_{12}\ E_1^n(z) + k_{11}[(h_{11}^{1}M^{N-1} + h_{11}^{2}M^{N-2} + \cdots + h_{11}^{N}M^{N-N}) + (h_{12}^{1}M^{N-1} + h_{12}^{2}M^{N-2} + \cdots + h_{12}^{N}M^{N-N})] + k_{12}[(h_{21}^{1}M^{N-1} + h_{21}^{2}M^{N-2} + \cdots + h_{21}^{N}M^{N-N}(z)) + (h_{22}^{1}M^{N-1} + h_{22}^{2}M^{N-2} + \cdots + h_{22}^{N}M^{N-N})] \quad \textbf{(21)}$$

And

$$M_2^n = k_{21}\ E_1^n(z) + k_{22}\ E_1^n(z) + k_{21}[(h_{11}^{1}M^{N-1} + h_{11}^{2}M^{N-2} + \cdots + h_{11}^{N}M^{N-N}) + (h_{12}^{1}M^{N-1} + h_{12}^{2}M^{N-2} + \cdots + h_{12}^{N}M^{N-N})] + k_{22}[(h_{21}^{1}M^{N-1} + h_{21}^{2}M^{N-2} + \cdots + h_{21}^{N}M^{N-N}(z)) + (h_{22}^{1}M^{N-1} + h_{22}^{2}M^{N-2} + \cdots + h_{22}^{N}M^{N-N})] \quad \textbf{(22)}$$

Equations 21 and 22 if implemented will yield closed loop responses having open loop dynamics. Algorithm can be speeded up by introducing a matrix of gains α in equation 21 and 22 to give

$$M_1^n = k_{11}\ \alpha E_1^n(z) + k_{12}\ \alpha E_2^n(z) + k_{11}[(h_{11}^{1}M^{N-1} + h_{11}^{2}M^{N-2} + \cdots + h_{11}^{N}M^{N-N}) + (h_{12}^{1}M^{N-1} + h_{12}^{2}M^{N-2} + \cdots + h_{12}^{N}M^{N-N})] + k_{12}[(h_{21}^{1}M^{N-1} + h_{21}^{2}M^{N-2} + \cdots + h_{21}^{N}M^{N-N}(z)) + (h_{22}^{1}M^{N-1} + h_{22}^{2}M^{N-2} + \cdots + h_{22}^{N}M^{N-N})] \quad \textbf{(23)}$$

And

$$M_2^n = k_{21}\ \alpha E_1^n(z) + k_{22}\ \alpha E_2^n(z) + k_{21}[(h_{11}^{1}M^{N-1} + h_{11}^{2}M^{N-2} + \cdots + h_{11}^{N}M^{N-N}) + (h_{12}^{1}M^{N-1} + h_{12}^{2}M^{N-2} + \cdots + h_{12}^{N}M^{N-N})] + k_{22}[(h_{21}^{1}M^{N-1} + h_{21}^{2}M^{N-2} + \cdots + h_{21}^{N}M^{N-N}(z)) + (h_{22}^{1}M^{N-1} + h_{22}^{2}M^{N-2} + \cdots + h_{22}^{N}M^{N-N})] \quad \textbf{(24)}$$

Equation 23 and 24 is the final form of the algorithm. The constants α_{11}, α_{12}, α_{21}, α_{22} are tuning constants of the algorithm.

For set point changes, goal of optimization effort would be to ensure good transient response for one variable, consistent with zero offset and zero steady state error for the other.

According to Arulalan and Deshpande (1985) the robustness of the algorithm can be enhanced by adding a first order filter in the feedback path. Excessive ringing of manipulated variables is to be avoided.

For simplification, equation 24 can be written in form as shown below:

$$\begin{bmatrix} M_1 \\ M_2 \end{bmatrix} = \begin{bmatrix} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{bmatrix} \begin{bmatrix} E_1 \\ E_2 \end{bmatrix} + \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} M_{1o} \\ M_{2o} \end{bmatrix} \quad \textbf{(25)}$$

This equation shows that the latest value of M1 and M2 are calculated based on the past values of M1 and M2. α denotes the tuning parameters required in order to obtain the most appropriate response of the model. E denotes the error of actual value and the true value of the system. Next, k denote the inverse gain of a particular transfer function. h denotes the impulse response coefficient of each transfer function in the 2 x 2 system.

2.2 Stability Properties of SMPC Algorithm

Control algorithm, D(z) is given by,

$$D(z) = \frac{\alpha Kp}{Kp + (\alpha Kp - 1)G(z)} \quad \textbf{(26)}$$

There is no requirement, as far as loop stability is concerned, that D(z) must be open loop stable. Indeed the commonly used PI controller is open loop unstable. The open loop response of the SMPC algorithm is similar to that of a PI algorithm.

2.3 Deviation Variables

Most real process variables are function of time. Typically, values fluctuate around a normal value, sometimes slightly higher, sometimes lower. This long time value is one aspect of steady state. Deviation variable is the difference of a particular variable from its steady state value. A dynamic model is converted to its equivalent deviation variable form by subtracting the steady state equation from the linearised dynamic equation as shown below:

$$Deviation\ variable\ = Linearized\ dynamic - Steady\ state \qquad \textbf{(27)}$$

These steady state values are used in finding the deviation of a particular parameter with respect to changes induce into the system. It helps us tracking that particular parameter of interest as it move away from the steady value. A set of deviation variables provides an intuitive basis for explaining this dynamic behavior with the appropriate reference to the desired operating point. (Jose Alberto Romahnoli, Ahmet Alazoglu, 2012).

2.4 Finite Impulse Response

In this type of test, a unit pulse is applied to the manipulated input and the model coefficients are simply the values of the outputs at each time strep after the pulse input is applied (Bequette, 2003).

There is direct relationship between step and impulse response models as shown below:

$$s_i = \sum_{j=1}^{i} h_j \qquad \textbf{(28)}$$

s denotes the step test coefficient meanwhile h denotes the impulse test coefficient. There are limitations to impulse response models. It can be only used to represent open loop stable processes and requires large number of parameters compared to state space and transfer function models (Bequette, 2003).

2.5 Integral Square Error (ISE)

$$ISE = \int_0^\infty [e(t)]^2 dt \qquad \textbf{(29)}$$

A performance index such as ISE would be used in this paper for tuning parameter selection. Tuning parameters that have non zero offset and low ISE are selected. The error signal is e(t) which is the difference between the set point and the measurement.

2.6 Process Description

Distillation columns, which are widely used for separation and refining operations, require phenomenal amount energy for its operation. Nevertheless, minimization of energy usage is possible if the compositions of both the top and bottom product streams are controlled to their design values, i.e deal temperature control.

A common scheme is to use reflux flow to control top product temperature whilst heat input is used to control bottom product temperature. However, changes in reflux also affect bottom product temperature and component fractions in the top product steam are also affected by changes in heat input. Several loop interactions can therefore occur in the dual temperature control of distillation columns.

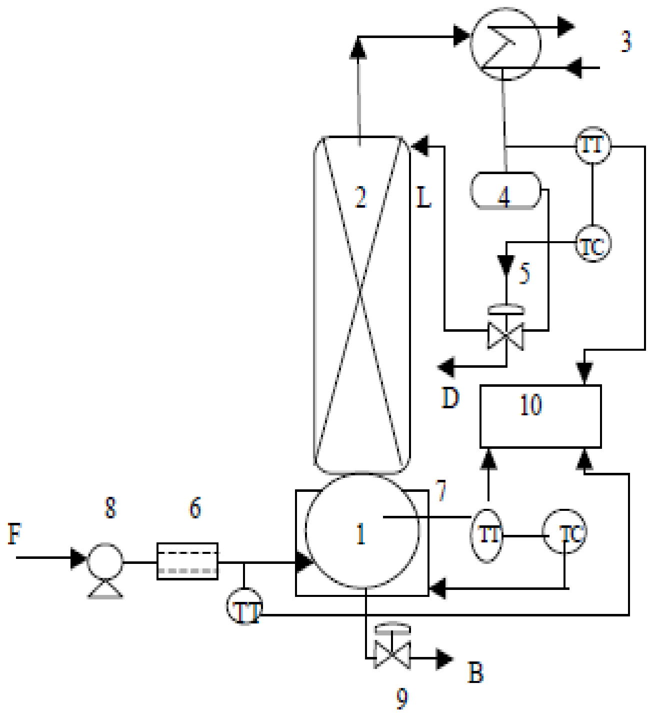

Figure 5: Distillation column and equipments involve

Loop interactions may also arise as a consequence of process design: typically the use of recycle streams for heat recovery purposes. An example is where the hot bottom product stream of distillation column is used as the heating medium to heat the reboiler as shown in Figure 5 (Tham, 1999). Suppose heat input to the reboiler is used to control the temperature of bottom product stream. If for some reason, the composition of this stream changes, then heat input will change in an attempt to maintain the composition at its desired level. However changes in heat input will alter the temperature of the bottom product stream, which will affect the temperature of the feed stream (Ay and Karacan, 2011). Changes in feed temperature will in turn influence bottom product temperature. Equipments involve in operating a distillation column as shown below:

Table 1: Equipments involve in operating a distillation column

Equipment Number	Equipment Name
1	Reboiler
2	Packed Column

3	Condenser
4	Accumulator
5	Reflux Valve
6	Heat Exchanger
7	Jacket Exchanger
8	Pump
9	Bottom product valve
10	Computer

Saniye and Suleiman distillation column model has transfer functions as shown below:

$$\begin{bmatrix} T_d \\ T_b \end{bmatrix} = \begin{bmatrix} \frac{1.84e^{-11.7s}}{56.1s+1} & \frac{1.04e^{-4.64s}}{16.55s+1} \\ \frac{2.88\,e^{-8.15s}}{74.6s+1} & \frac{-2.39e^{-3s}}{9.94s+1} \end{bmatrix} \begin{bmatrix} R \\ Q_r \end{bmatrix}$$

From the transfer functions as shown above it clearly shows that the controlled variables of the distillation column model are distillate and bottom temperature which are $T_d(s)$ and $T_b(s)$ respectively. The manipulated variables of the system are reflux ratio, R and reboiler heat duty, Q_r.

It is a 2x2 system in which it has four transfer functions constituting of G_{p11}, G_{p12}, G_{p21} and G_{p22}. The steady state value of this model is shown as below:

Table 2: Steady state values of distillation column distilling mixture of methanol-ethanol-*n* butanol-isoamine alcohol-anisol

Distillation Column Parameters	Reflux Ratio	Reboiler Heat Duty (cal/min)	Top Product Temperature (^{O}C)	Bottom Product Temperature (^{O}C)
	1.5	700	70.5	77

CHAPTER 3

METHODOLOGY

3.1 Designing Procedures/Approach

Figure 6 below shows the flowchart diagram depicting the general approach in this project. There are few steps required in obtaining the SMPC algorithm. Such steps are conducting impulse test in determining the impulse test coefficients. Next, is determining the tuning parameters α_{11}, α_{12}, α_{21}, α_{22}. These tuning parameter values are obtained by trial and error method.

Then, gain matrix of the transfer function is determined. This gain matrix is obtained at G_p (0) for four transfer functions utilized in this paper. Inverse gain matrix is obtained from these gains.

Later, set points for each controlled variable are entered. The loads also are then entered. Errors which are difference between actual controller variable values and the set points are obtained. Predicted process outputs are then obtained. These process outputs are used in computing the errors. These errors are then used in equations 23 and 24 to obtain the desired controller outputs. These processes are repeated unless steady state is reached.

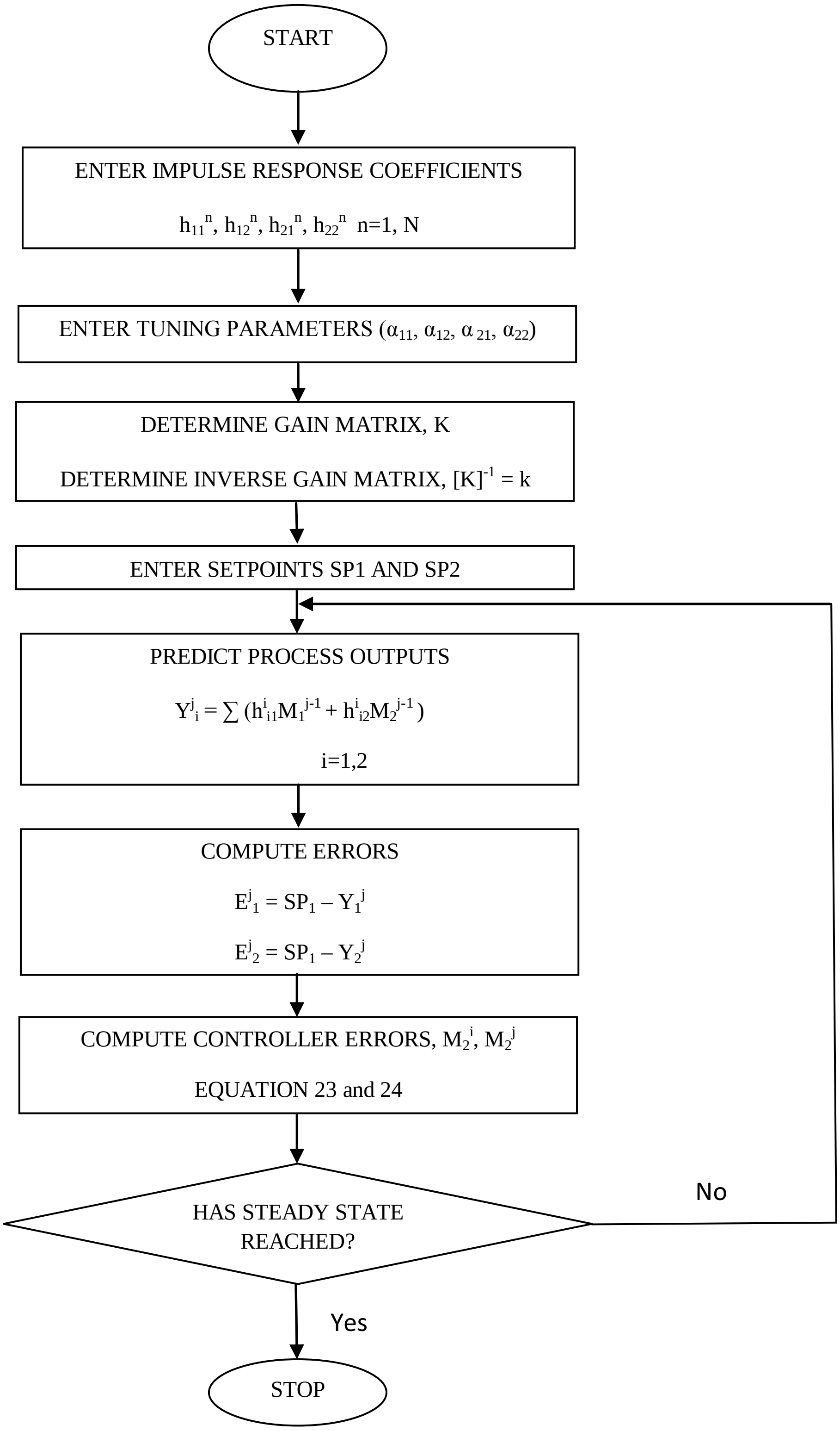

Figure 6: Flowchart of SMPC strategies

3.2 Key Milestones

Several key milestones for this research project must be achieved in order to meet the objective of this project:

Problem Statement and Objective of the project

Identifying the purpose of this research project

Literature Review

Gathering as much information as possible from various sources such as journals and websites

Simulation Design

Identifying the subjects that need to be investigated and the designing procedures, as well as the process model and the input output relationships

Data Analysis and Interpretation

The findings obtained are analyzed and interpreted critically. Comparison with other literature readings will also be done.

Documentation and Reporting

The whole research project will be documented and reported in detail. Recommendations or aspects that can be further improved in the future will also be discussed.

Figure 7: Project key milestones

3.3 FYP I Gantt Chart

Table 3: FYP I Gantt Chart

No	Detail Week	1	2	3	4	5	6	7		8	9	10	11	12	13	14
1	**Selection of Project Title**															
2	Preliminary Research Work and Literature Review															
3	Submission of Extended Proposal Defense						●									
4	Preparation for Oral Proposal Defense															
5	Oral Proposal Defense Presentation															
6	Detailed Literature Review															
7	Preparation of Interim Report															
8	Submission of Interim Draft Report														●	
9	Submission of Interim Final Report															●

3.4 FYP II Gantt Chart

Table 4: FYP II Gantt chart

No	Detail Week	1	2	3	4	5	6	7		8	9	10	11	12	13	14	15
1	Project Work Continues																
2	Submission of Progress Report									●	8th July						
3	Project Work Continues																
4	Pre EDX									25th July			●				
5	Submission of Draft Report													●	5th August		
6	Submission of Dissertation (soft bound)										15th August				●		
7	Submission of Technical Paper										15th August				●		
8	Oral Presentation										26th - 30th August					●	
9	Submission of Project Dissertation (hard bound)											30th September					●

CHAPTER 4

RESULTS & DISCUSSION

As per design procedures mentioned in **CHAPTER 3** finite impulse response test has been conducted and the impulse response coefficients are obtained through Matlab simulation. These finite impulse response coefficients are used in order to predict the process output of the 2 x 2 system.

These coefficients are then plotted against with time. The suitable time is chosen as that all responses stop at that time for each of the transfer function.

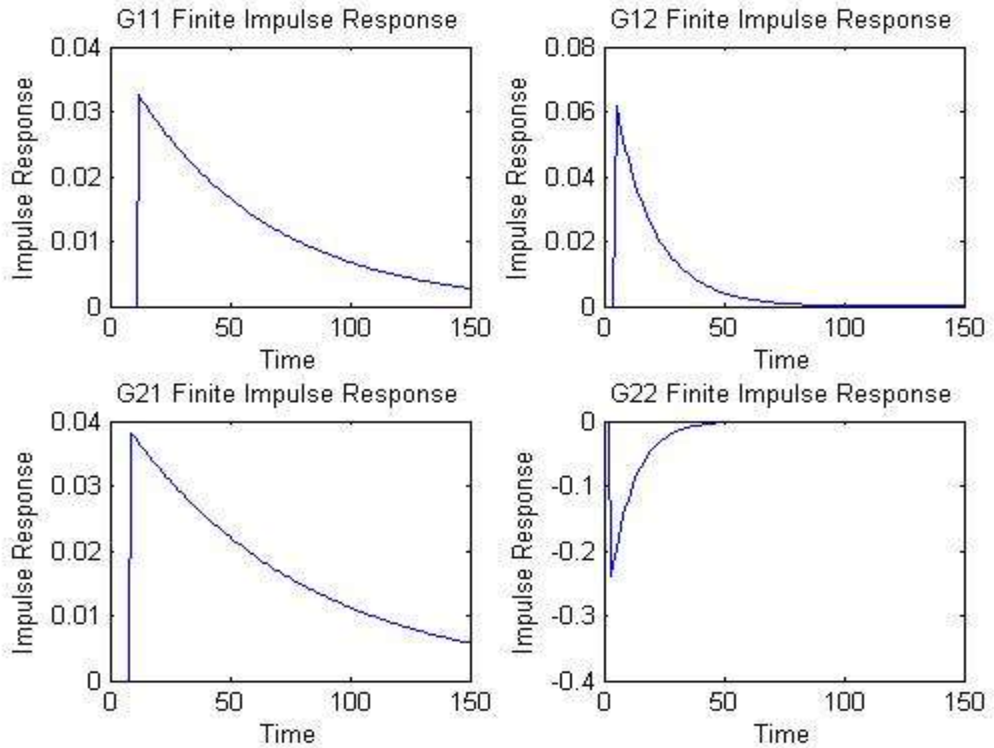

Figure 8: Impulse response of the 2 x 2 system utilized in the project

Above plots shows the coefficients obtained against time for each of the transfer function in the 2 x 2 system. In this plot also shows that the G22 requires the shortest time taken for bringing the coefficients to 0. Meanwhile, G12 has the second shortest time taken to bring the coefficient to 0. G21 has the longest time taken to bring the impulse response coefficients to 0. Lastly, G11 has the second shortest time taken to bring the coefficients to 0.

This SMPC algorithm only deals with set point tracking of the system. If there is any changes in the set point there are two variables that will be manipulated to compensate the changes. First, is the reflux ratio, R and secondly, is the reboiler heat duty, Q_r. These set point changes are conducted per controlled variable in the system. Firstly, the set point changes in the top product temperature, Td is increased by 1^{o}C from its steady state value which is 70.5 ^{o}C, meanwhile the bottom product temperature, Tb is held constant at its steady state value which is 77 ^{o}C. Thus, the increment in set point of the top product temperature is from 70.5 ^{o}C to 71.5 ^{o}C.

Meanwhile for set point change in bottom product temperature, Tb is conducted by increasing the set point by 1^{o}C from its steady state value which is 77 ^{o}C. Thus, the set point is increased from 77 ^{o}C to 78 ^{o}C. For this test, the set point for the top product temperature, Td is held constant at its steady state value which is 70.5 ^{o}C.

4.1 Top product temperature, T_d set point changed from 70.5 ^{o}C to 71.5 ^{o}C, bottom product temperature, T_b set point held at steady state value 77 ^{o}C

For the set point change in the top product temperature, T_d the sampling instant is at 1 second in which this instant is repeated for 150 times. The tuning parameters used in this change as shown below:

Table 5: Tuning Parameters for top product temperature, T_d set point change

Tuning Parameter	
α_{11}	1.471
α_{12}	3.855
α_{21}	0.698
α_{22}	-2.524

The ISE for these tuning parameters is 15.9731. A lower ISE can be obtained but the offset of the final value would be large. These, tuning parameters is suitable though it has quite large ISE. This is due to the slow response of the process to either to reach to its new steady state value or to achieve to its former value.

The response of the top product temperature, T_d due to change in set point as shown below:

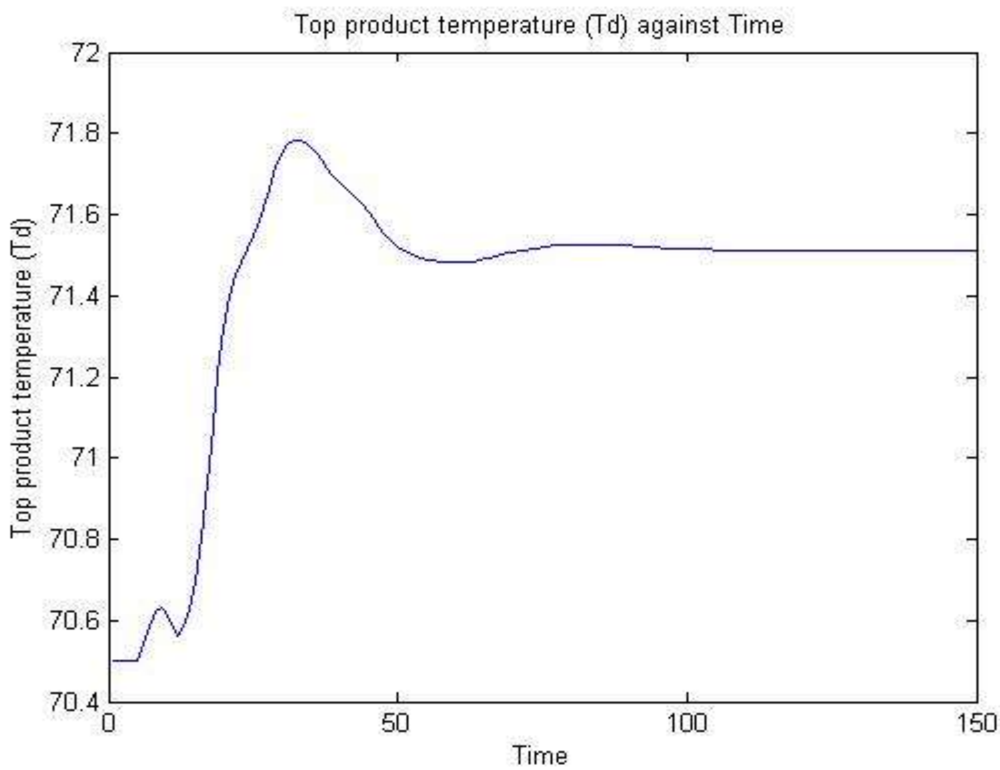

Figure 9: Top product temperature, T_d response to its set point change

Figure 9 shows that the after set point change the system requires some time to reach to new steady state which is 70.5°C. There is some deviation in the response of top product temperature, Td compare to its desired value. After nearly, 80 seconds top product temperature response goes to steady state.

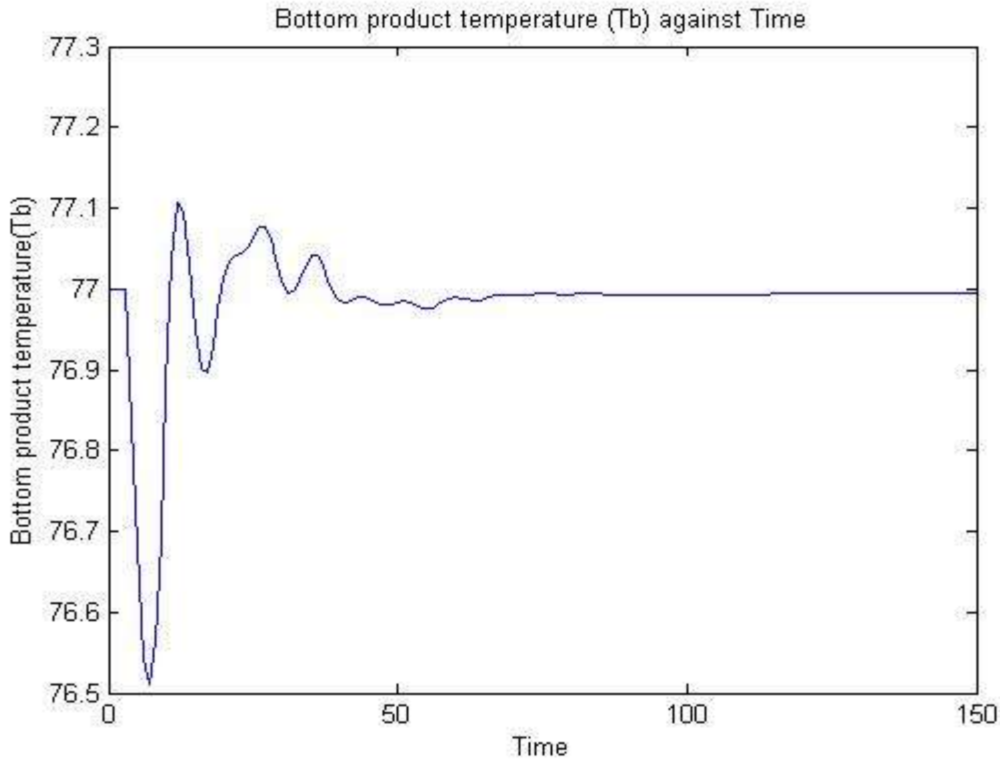

Figure 10: Bottom product temperature, T_b response to T_d set point change

Figure 10 shows the response of bottom product temperature, T_b to the set point change in top product temperature T_d. There is an offset from the desired values of Tb to its steady state value. This offset can be minimized by tuning the system with the most appropriate tuning parameters. The response of T_b is quite slow, in which it takes longer time to reach to its steady state value compared to T_d response.

. The final values for errors are shown in the table below:

Table 6: Errors in top product temperature, T_d set point change

Offset (%)	
Top product temperature, T_d	0.0098
Bottom product temperature, T_b	0.0071

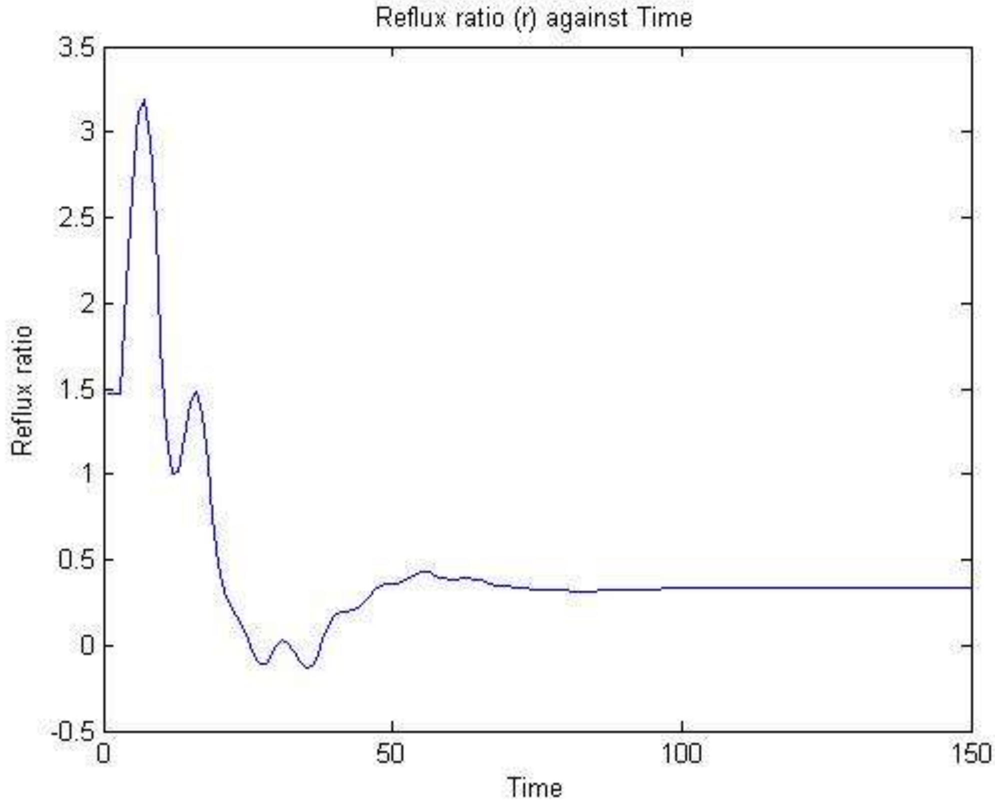

Figure 11: Reflux ratio in deviation variable required for T_d set point change

Above figure shows the required reflux ratio to be manipulated in order to compensate the change in T_d set point. As show in the plot, reflux ratio goes to steady state value after some time. The final value of reflux ratio required is 1.810

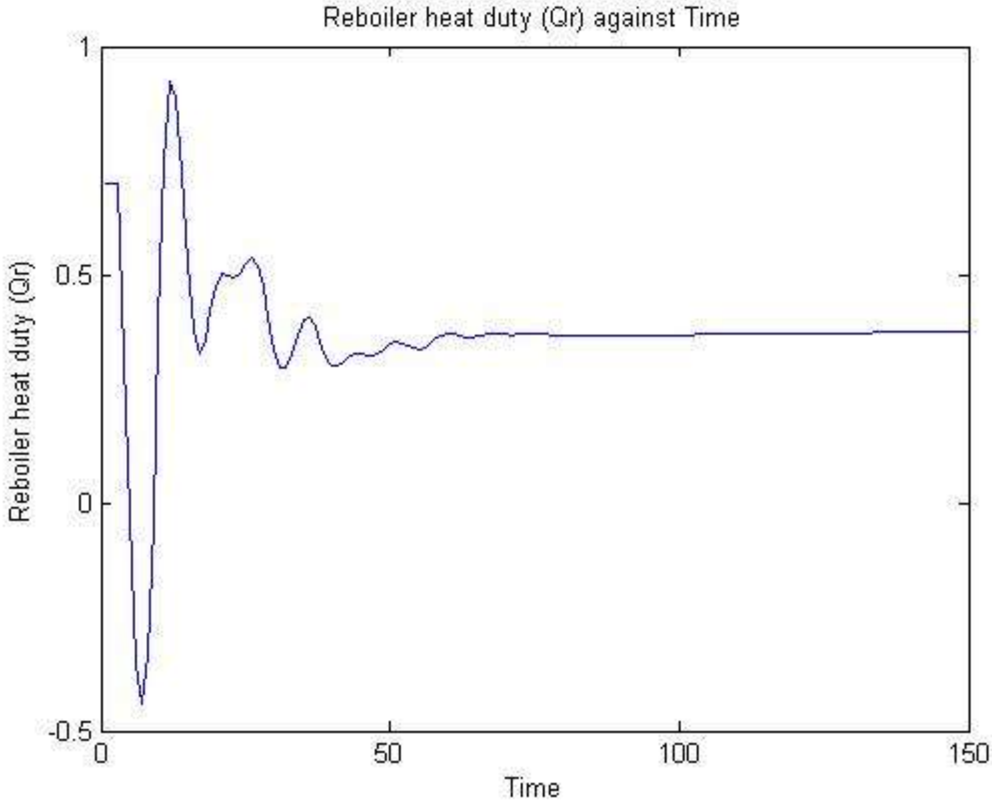

Figure 12: Reboiler heat duty in deviation variable required for T_d set point change

Meanwhile, it is similar for reboiler heat duty, in which the system requires about 700.3740 cal/min of heat duty to compensate the set point change in top product temperature, T_d. The final values of the reflux ratio and reboiler heat duty are shown in the table below:

Table 7: Manipulated variables final values for top product temperature T_d, set point change

New Steady State Values at Td = 71.5°C and Tb = 77 °C	
Reflux ratio	1.810
Reboiler heat duty (cal/min)	700.3740

4.2 Bottom product temperature, T_b set point changed from 77 ^oC to 78 ^oC, top product temperature, T_d set point held at steady state value 70.5 ^oC

For the set point change in the bottom product temperature, T_d the sampling instant is at 1 second in which this instant is repeated for 150 times. The prediction horizon of this case is 150 seconds. The tuning parameters used in this change as shown below:

Table 8: Tuning Parameters for bottom product temperature, T_b set point change

Tuning Parameters	
α_{11}	1.406
α_{12}	1.545
α_{21}	0.6426
α_{22}	-1.4111

The ISE of these tuning parameters is 4.2436.

The response of the top product temperature, T_d due to change in set point of bottom product temperature, T_b as shown below:

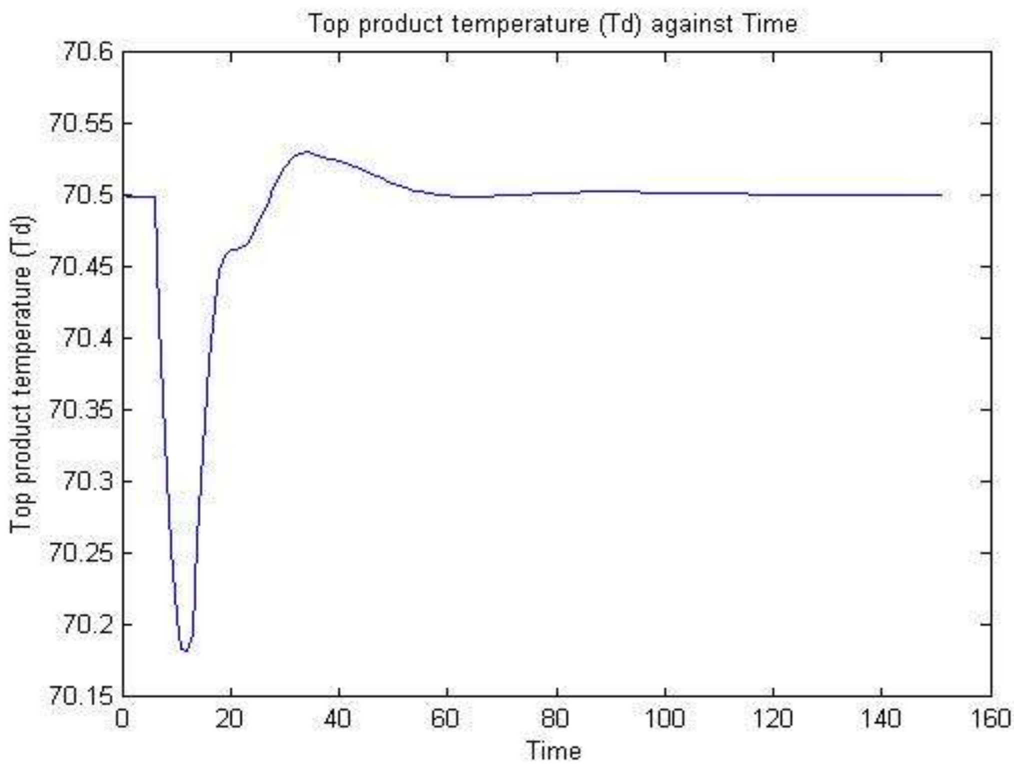

Figure 13: Top product temperature, T_d response to its set point change

Figure 13 shows what is happening for top product temperature of the distillation column when the bottom product temperature set point is raised to 1^oC from its initial value.

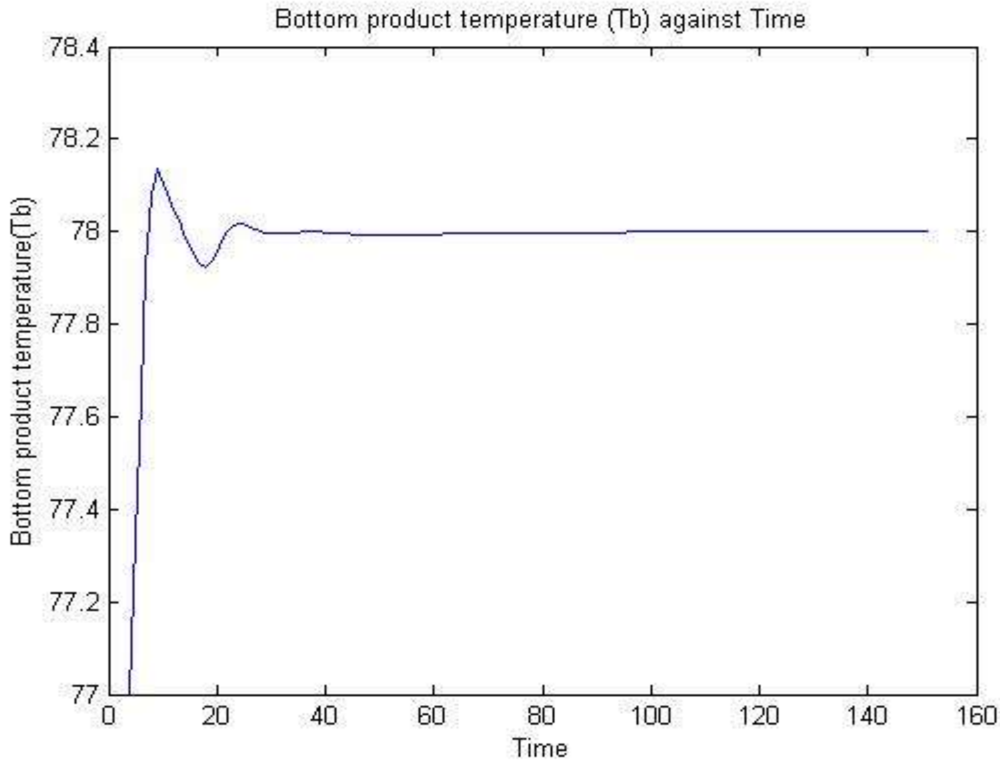

Figure 14: Bottom product temperature, T_b response to its set point change

Figure 14 shows the response of bottom product temperature, T_b to its set point change. It is a fast response as the controller is properly tuned. Furthermore, the control algorithm only has very small offset from its target value.

The final values of the errors are shown in the table below:

Table 9: Errors on bottom product temperature, T_b set point change

Offset (%)	
Top product temperature, T_d	0.0004
Bottom product temperature, T_b	0.0077

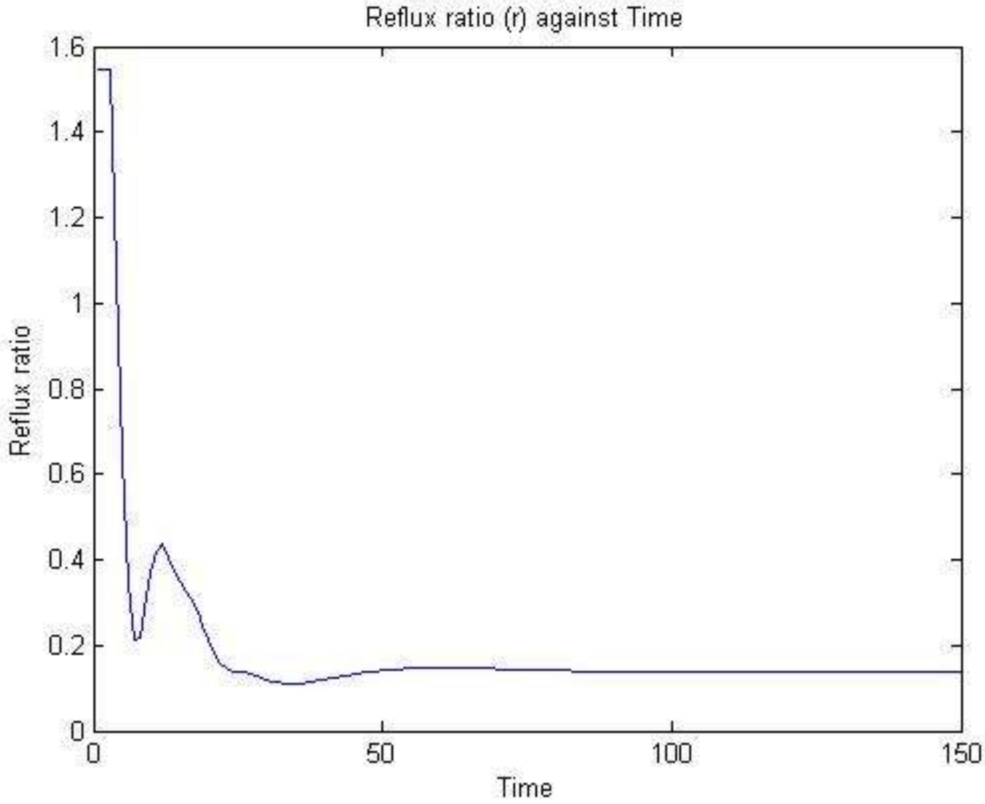

Figure 15: Reflux ratio in deviation variable required for T_b set point change

Above figure shows the required reflux ratio to be manipulated in order to compensate the change in T_b set point. As show in the plot, the final value of reflux ratio required is 1.6395.

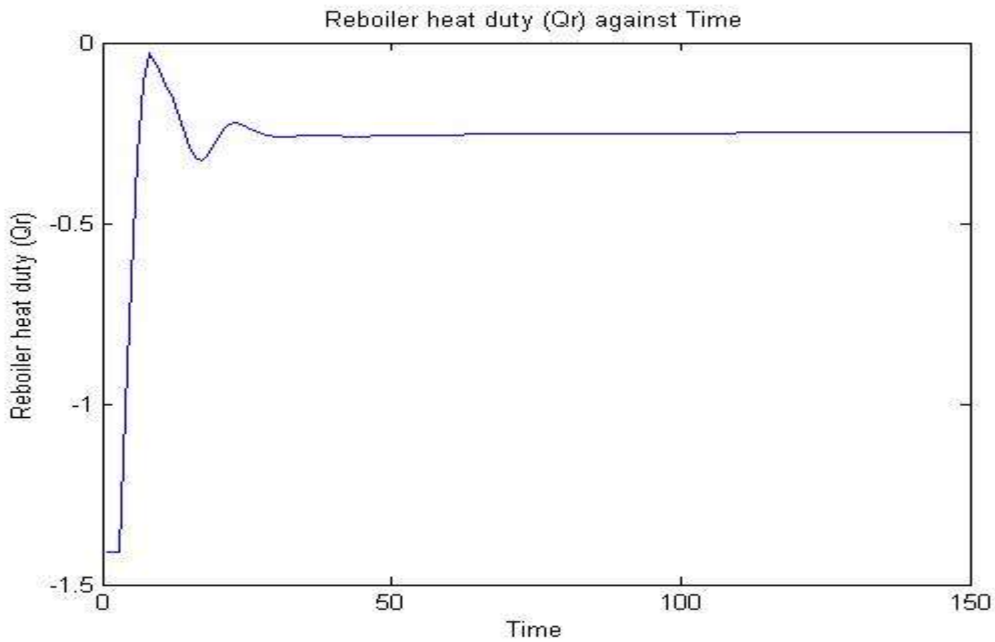

Figure 16: Reboiler heat duty in deviation variable required for T_d set point change

Above figure shows the final value of reboiler heat duty required in order to compensate the set point change in bottom product temperature, T_b. The final values of the reflux ratio and reboiler heat duty are shown in the table below:

Table 10: Manipulated variables final values for bottom product temperature, T_b set point change

New Steady State Values at Td = 70.5^{o}C and Tb = 78 ^{o}C	
Reflux ratio	1.6395
Reboiler heat duty (cal/min)	699.7517

4.3 Controller Robustness Test

This robustness test is done by changing gain value of certain transfer functions in the system. The gain is changed up to the value until the controller could not be able to control the system. There is range of a controller can sustain new system to be operated. The higher the range controller can sustain, the more robust the controller it is.

This test is done in the middle of the program. The impulse response coefficients are obtained based on the initial value of the process gain. In addition there is no change in the control algorithm of the controller including the tuning parameters. This test indicates that the controller is being operated under new system. Such condition in real industry can take plant when the controller is being designed based on specific properties of raw materials. However, that particular raw material may not be able due to certain issues such market condition and demands. Thus, new raw materials with slightly different properties are brought in. Hence, the difference in property may cause deviations from the initially designed plant.

4.3.1 Top product temperature set point change from 70.5^{o}C to 71.5^{o}C, bottom product temperature set point is at 77^{o}C,

<u>Process Gains are Randomly Increased and Decreased</u>

The control algorithm for top product temperature set point change is robust when K21 is reduced to 7% of its initial value and increased to 10% of its initial value. In addition, this controller is still capable to control the system when K11 is increased to 8% of its initial value and reduced to 13% of its initial value.

To conclude, the controller for top product temperature set point change is robust when the system is changed to 10% less from its initial system. This controller would also be robust and sensitive when the system is changed about 7% more than its initial value.

4.3.2 Bottom product temperature set point change from 77^{o}C to 78^{o}C, top product temperature set point is at 70.5^{o}C

<u>Process Gain are Randomly Increased and Decreased</u>

The control algorithm for bottom product temperature set point change is robust when K21 is reduced to 15% of its initial value and increased to 13% of its initial value. In addition, this controller is still capable to control the system when K22 is increased to 14% of its initial value and reduced to 13% of its initial value.

To conclude, the controller for bottom product temperature set point change is robust when the system is changed to 15% less from its initial system. This controller would also be robust and sensitive when the system is changed about 14% more than its initial value.

CHAPTER 5

CONCLUSION& RECOMMENDATIONS

5.1 Relevancy to Objectives

The objectives of this project have been achieved. In which control algorithm based on SMPC is design for multi component packed distillation column. Set point tracking for this system has also been conducted. The best responses are obtained by manipulating tuning parameter values based on minimum offset and ISE. This project also shows that SMPC can also be implemented in any kind of multi variable systems.

5.2 Recommendations

For future study, it is recommended to include load change in designing SMPC control algorithm. Verification method such as an experimental system to test this SMPC algorithm is also suggested.

REFERENCES

Qin, S.J. and T.A. Badge well (1997). An overview of industrial model predictivecontrol technology. In: Chemical Process Control - V. Vol. 93, no. 316. pp. 232:256. AIChe Symposium Series - American Institute of Chemical Engineers.

Lee, J.H. and B. Cooley (1997). Recent advances in model predictive control.In: Chemical Process Control - V. Vol. 93, no. 316. pp. 201{216b. AICheSymposium Series - American Institute of Chemical Engineers.

Garcia, C.E., D.M. Prett and M. Morari (1989). Model predictive control: Theory and practice

Nikolaou, M. (2001). *Model Predictive Controllers: A Critical Synthesis of Theory and Industrial Needs.* Houston, USA: Author

Åström, K. J., and B. Witten mark, *Adaptive Control*, Addison-Wesley (1989).

Baxley, R. A., and J. Bradshaw, *Personal communication*, Texas A&M University (1998).

Riggs, James B. (2000). "Comparison of advanced distillationcontrol methods". *Final Technical Report April 1994 – March1999*

Wahid, A., & Ahmad, A. (2008). *Representative Model Predictive Control*. Depok, Johor, Indonesia, Malaysia: Author

Ay. S., & Karacan, S. (2011). *Decoupling Constrained Model Predictive Control Of Multi Component Packed Distillation Column.* Ankara, Turkey: IDOSI Publications.

Tham, M.T. (1999). *Multivariable Control: An Introduction to Decoupling Control.* Newcastle, United Kingdom

Qin, S.J and TA, Badgwell, 2003. Soft computing in model based predictive control. Control Engineering Practice, 11: 733-764.

Wills, A.G., D. Bates,A.J Fleming, B. Ninness, S.O.R Moheimani, 2008. Predictive Control Applied to Constraint Handling in Active Noise and Vibration Control. IEEE Transactions on Control Systems Technol., 16:3-12

Perez, T., G.C Goodwin, 2005. Constrained predictive control of ship fin stabilizers to prevent dynamic stall. Control Engineering. Practice, 16: 482-494.

Lam, D.K., R. Baker and C.L.E Swartz, 2007. Reference trajectory optimization under constrained predictive control. Canadian J. Chemical Engineering, 85: 454-464.

Gomatam R. Arulalan and Pradeep B. Deshpande, 1985. Simplified Model Predictive Control. Chemical Engineering Department, University of Louisville, Louisville, Kentucky.

APPENDIX

Controller Robustness Test Result

A. Top product temperature set point change from 70.5°C to 71.5°C, bottom product temperature set point is at 77°C,

<u>Process Gains are Randomly Increased and Decreased</u>

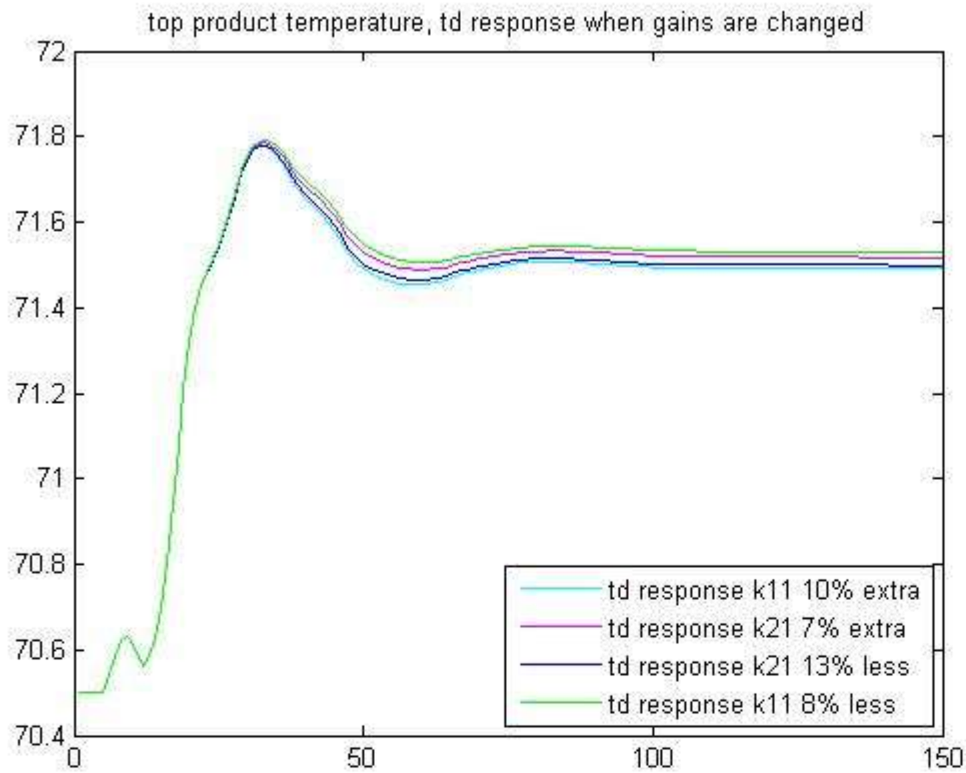

Figure 17: T_d response when process gains are increased and decreased

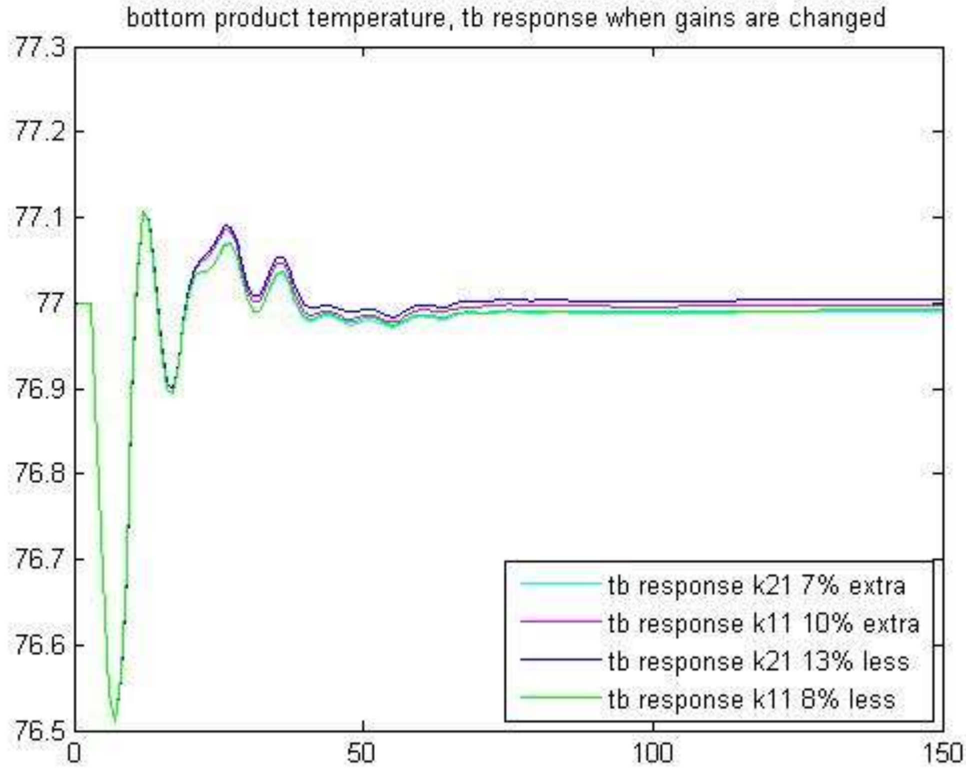

Figure 18: T_b response when process gains are increased and decreased

B. Bottom product temperature set point change from 77^{o}C to 78^{o}C, top product temperature set point is at 70.5^{o}C

<u>Process Gains are Randomly Increased and Decreased</u>

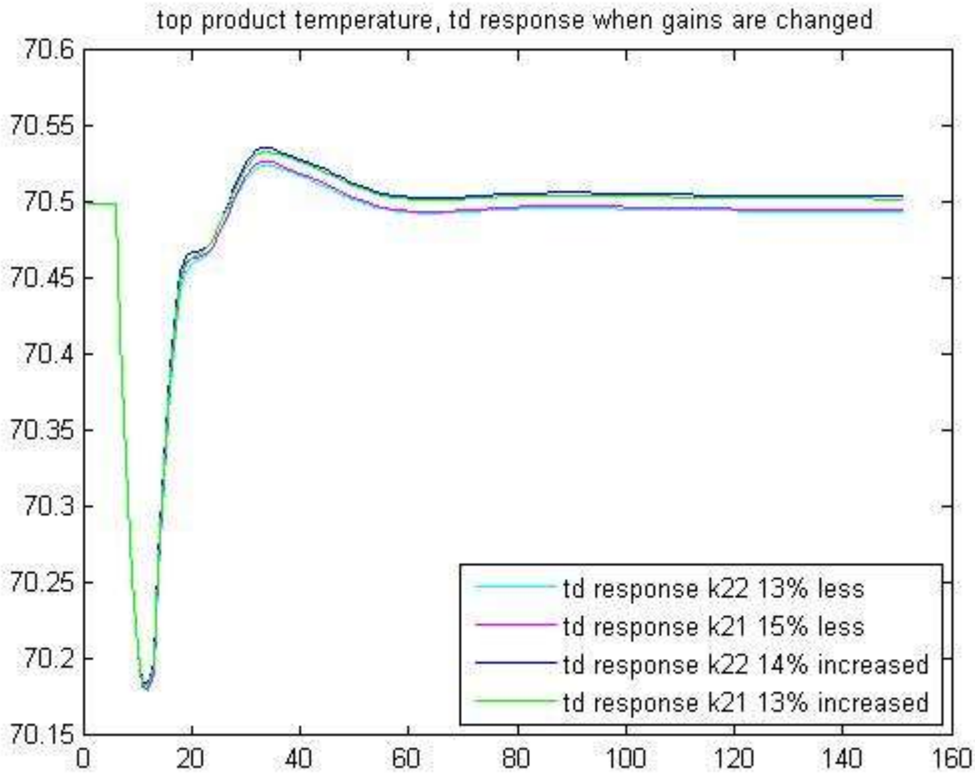

Figure 19: T_d response when process gains increased and decreased

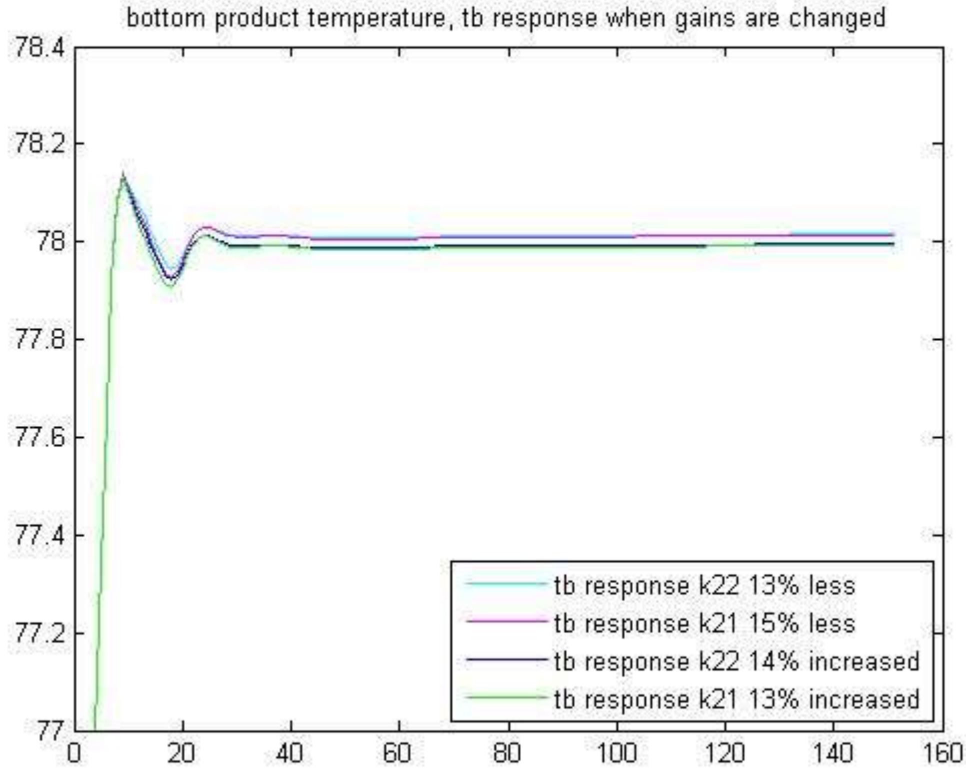

Figure 20: T_b response when process gains increased and decreased

C. Control Algorithm Bottom Product Set Point Change

```
%This program is dedicated to  predict future output of multi component
%distillation column to distillate mixture of  methanol-ethanol-n
%butanol-isoamine alcohol-anisol.

%Manipulated variables of this MIMO system are R(reflux ratio) and Qr
%(reboiler heat duty). Meanwhile, controlled variables of the system are Td
%(top product temperature) and Tb (bottom product temperature).

%num = TF numerator in function of s
%den = TF denominator in function of s

%MIMO consist of 4 TFs (2x2 system)

%Suppose at R =1.5, Qr = 700, the output responses are Tb=77 and Td=70.5
%This program deals with only set point tracking of the 2x2 system

%This program focuses on the deviation values of each variable sp1, sp2, M1
%and M2

clc
clear
%------------------------Impulse response coefficient----------------------

%G11 = (1.84*exp(-11.7*s))/(56.1*s+1)          % 1st TF
num1 = [0 1.84];
den1 = [56.1 1];
delay1 = 11.7;
TFinal = 1:150;
G11 = tf(num1, den1,'InputDelay', 11.7);
[y1, t1]= impulse (G11,TFinal);
h1(:,1)=y1;

%G12 = (1.04*exp(-4.64*s))/(16.55*s+1)         %2nd TF
```

```
num2 = [0 1.04];
den2 = [16.55 1];
delay2 = 4.64;
G12 = tf(num2, den2,'InputDelay', 4.64);
[y2, t2]= impulse (G12,TFinal);
h2(:,1)=y2;

%G21 = (2.88*exp(-8.15*s))/(74.6*s+1)          %3rd TF
num3 = [0 2.7648];
den3 = [74.6 1];
delay3 = 8.15;
G21 = tf(num3, den3,'InputDelay',8.15);
[y3, t3]= impulse (G21,TFinal);
h3(:,1)=y3;

%G22 = (-2.39*exp(-3*s))/(9.94*s+1)            %4th TF
num4 = [0 -2.39];
den4 = [9.94 1];
delay4 = 3;
G22 = tf(num4, den4,'InputDelay', 3);
[y4, t4]= impulse (G22,TFinal);
h4(:,1)=y4;

G=[G11 G12;
  G21 G22];
H11=h1';                 %impulse response coefficient of G11
H12=h2';                 %impulse response coefficient of G12
H21=h3';                 %impulse response coefficient of G21
H22 =h4';                %impulse response coefficient of G22

H = [H11 H12; H21 H22];     %matrix of impulse response coefficients

%------------------Plotting impulse response coefficient-------------------
```

```
figure(1)
subplot(2,2,1), plot (t1,y1,'b');
ylabel ('Impulse Response');
xlabel ('Time');
title ('G11 Finite Impulse Response')

subplot (2,2,2), plot (t2,y2,'b');
ylabel ('Impulse Response');
xlabel ('Time');
title ('G12 Finite Impulse Response')

subplot (2,2,3), plot (t3,y3,'b');
ylabel ('Impulse Response');
xlabel ('Time');
title ('G21 Finite Impulse Response')

subplot (2,2,4), plot (t4,y4,'b');
ylabel ('Impulse Response');
xlabel ('Time');
title ('G22 Finite Impulse Response');

%-----------------tuning parameters a1,a2,a3,a4----------------------------

a11 =1.406;
a12 =1.545;
a21 =0.6426;
a22 =-1.4111;
a = [a11 a12;
    a21 a22];

%-------------------------Inverse gain matrix-----------------------------

K = [1.84 1.04;
    2.88 -2.39];
```

```
k = inv (K);

%-------------------------------------------------------------------------
Td=70.5;                    %New set point Td ()
Tb=78;                      %New set point Tb ()

sp1 = Td-70.5;              %Td set point change
sp2 = Tb-77;                %Tb set point change
sp = [sp1; sp2];
E = [0;0];                  %Steady state error

% E_1 = (ones(150,1))*0;
% E_2 = (ones(150,1))*0;

% Td_1 = (ones(150,1))*Td;
% Tb_1 = (ones(150,1))*Tb;

M = ones((300),1);          %Matrix of M1 and M2
M1 = (ones(150,1))*0;       %M1 is the reflux ratio
M2 = (ones(150,1))*0;       %M2 is the reboiler heat duty
M = [M1;M2];

Ct = [70.5;77];
Mt = [];
Et = [];
C0=[70.5;77];

Mn = [M(1); M(151)];        %Latest values of M1 and M2

for i = 1:150
   C = H*M;
   E = sp - C;
   Ct = [Ct C+C0];
   Et = [Et E];
```

```
    Mn = k*H*M+a*E;
    M1=[Mn(1);M1];
    M1(end)=[];
    M2=[Mn(2);M2];
    M2(end)=[];
    M=[M1;M2];

end

%------------------------Plotting Td and Tb------------------------------

figure(2);
plot (Ct(1,:),'b');
title('Top product temperature (Td) against Time');
xlabel ('Time');
ylabel ('Top product temperature (Td)');
% hold on
% plot (Td_1, '--r')
% hold off
% legend ('Top product temperature, Td Prediction','Top product temperature, Td
Setpoint')

figure(3);
plot (Ct(2,:),'b');
title('Bottom product temperature (Tb) against Time');
xlabel('Time');
ylabel ('Bottom product temperature(Tb)');
% hold on
% plot (Tb_1,'--r')
% hold off
% legend ('Bottom product temperature, Tb Prediction','Bottom product temperature, Tb
Setpoint')

%------------------Plotting Td error and Tb error-----------------------
```

```
figure(4);
plot (Et(1,:),'b');
title ('Error on Top product temperature (Td) against Time');
xlabel('Time');
ylabel('Error Top Product Temperature');
% hold on
% plot (E_1,'--r')
% hold off
% legend ('Td Error',' Td Desired Error ')

figure(5);
plot (Et(2,:),'b');
title ('Error on Bottom product temperature (Tb) against Time');
xlabel('Time');
ylabel('Error Bottom Product Temperature');
% hold on
% plot (E_2,'--r')
% hold off
% legend ('Tb Error',' Tb Desired Error ')
%
% %-------------------plotting reflux and reboiler heat duty---------------
%
r = flipud(M1+1.5);
Qr = flipud(M2 + 700);

figure(6);
plot (r,'b');
title ('Reflux ratio (r) against Time');
xlabel('Time');
ylabel('Reflux ratio');

figure(7);
plot (Qr,'b');
```

```
title ('Reboiler heat duty (Qr) against Time');
xlabel('Time');
ylabel('Reboiler heat duty (Qr)');

figure (8)
plot (r)
hold on
plot (Qr)
hold off

%--------------------------------ISE CALCULATION---------------------------

%square of errors
b1  = Et(1,:).^2;
b2  = Et(2,:).^2;
%sum of square of errors
B1 = sum(b1)
B2= sum(b2)
%percentage of ISE

ISE = B1+B2
```