Development of Soft Sensor with Time Difference of Process Variables Approach

by

Kala Krissna A/P Balakrishnan

Dissertation submitted in partial fulfilment of

the requirements for the

Bachelor of Engineering (Hons)

(Chemical Engineering)

MAY 2012

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Development of Soft Sensor with Time Difference of Process Variables Approach

by

Kala Krissna A/P Balakrishnan

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,

07/09/2012

(AP Dr. Ramasamy Marappagounder)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
May 2012

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.

KALA KRISSNA A/P BALAKRISHNAN

ABSTRACT

Soft sensors are used to estimate the process variables that are hard to measure online in a process unit but the predictive accuracy of the estimation will deteriorate due to certain reasons. The reasons are usually due to the changes of plant state, catalyst performance loss, sensor or process drift and scale deposition. In order to overcome the degradation of the soft sensors due to process drift, time difference of process variables is proposed to use for the predictive model. The objective of this paper is to develop data-driven soft sensors with time difference of process variables and to evaluate its advantages over traditional static soft sensors. The modeling technique used for this approach is Partial Least Squares (PLS) method. Partial least squares method is a numerical method based on multiple regression. The main purpose of PLS is to predict a set of dependent variables from a set of independent variables or predictors. In this paper, a binary distillation column is selected as a case study and its virtual plant is built in Hysys environment. In the simulation, the input variables such as feed temperature, reflux flow rate, feed flow rate and steam flow rate are varied and the output data are captured with time. In addition, different sets of data were formed with various time differences in the variables. Those data are used to develop the soft sensor model using PLS technique in SIMCA-P software. The performance of the model is evaluated and compared with the conventional soft sensor. Based on the results, the predictive ability of the developed model is higher than the static conventional model.

ACKNOWLEDGEMENT

The past 28 weeks of my enrolment in final year project have been truly valuable experience to me. I have gained new knowledge on soft sensors and the methods to develop the model. I have broadened my knowledge and experiences in these related fields. Hence, I would like to take this opportunity to express my sincerest gratitude to a number of people that have helped me to achieve this.

First of all I would like to express my gratitude to God for letting me to complete this project in good health and well being. Next, I would like to express my deepest appreciation to AP Dr. Ramasamy Marappagounder, my supervisor, who has supervised me throughout my project period. His ever willingness to teach me and guide me has helped me tremendously in achieving my goals on my final year project. On top of that, he was constantly supportive on my decisions and will be always there to share his knowledge and experiences with me.

Secondly, my utmost gratitude goes to my FYP coordinators who guided me throughout my final year project. Without their guidance it is impossible for me to finish the final year project within the time period.

In short, I feel blessed to have done my final year project and for all the help that the aforementioned parties have given me. Only with their help was my final year project becomes a success.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background of study

In the industry, various sensors are used to detect the process variables such as temperature, flow rate, pressure and etc and respond to it. Previously, researches build the predictive models by using the data measured and stored in the process industry (Petr Kadlec, *et al.*, 2009). This predictive model is built because, it will take longer time to get the result of variable which is difficult to measure online. For example, in a process plant, in order to maintain the concentration variable, the measurement of it need to be updated but, it will take longer time to get the result. In order to save the time, this predictive model is created to estimate the process variable which is hard to measure online. This predictive model is called as Soft Sensors. The term soft sensor comes from two word which is software and sensor which shows it is a combination of these both software and sensor. Soft sensors are used to estimate the process variables that are hard to measure online. They use the available input data such as temperature, pressure or flow rate from the process unit to predict the output data. The output data is also known as objective variable such as concentration, density, melt flow rate and etc.

There are two types of soft sensor which are model-driven and data-driven. Model-driven is basically based on First Principle Models (FPM) and it mainly describes the physical and chemical background of the process. For instance, exothermal equation, energy balances, and mass-preservation principles. Apart of that, model-driven soft sensors focus more on the ideal steady-state of the processes and it is not suitable for transient state. Unlike the model-driven model, data-driven model describes the actual process conditions in a proper approach and suitable for the transient state. There are several concepts or approaches developed for this soft sensor to increase the prediction accuracy such as Time Difference models, Just in Time models and Moving Window models. These models are developed using the modelling techniques. For instance, the most famous modelling techniques used are Principle Component Analysis (PCA), Partial Least Squares (PLS), Artificial Neural Networks, Neuro-Fuzzy Systems and Support Vector Machines (SVR). All these concepts enable better sampling, save time and money compare to the expensive sensors which need high maintenance.

1.2 Problem Statement

Modeling a soft sensor which gives the predicted value accurately is not a simple task. This is because, the predicted value will deteriorate when the catalyzing performance loss, state of chemical plant changes, sensor and process drift and etc. The difference between sensor and process drift is sensor drift occur due to variation in the measuring devices while process drift caused by the transformation of the process of external process conditions. For instance, the external process condition is the weather influence where indirectly it might affect the purity of input material and also the catalyst deactivation. This is called as the degradation of soft sensor models. It will be difficult to identify the cause of the abnormal situation in the plant if the degradation is not solved. Furthermore, during the transient periods the conventional soft sensors are not accurate in predicting the quality variable. This is because, the conventional soft sensors predicts more accurate in the steady state condition where it will not be affected by drift in that process. In order to overcome this problem, time difference of process variable approach is developed. Constructing the model using this approach, leads to higher predictive accuracy because the data are represented as the time difference. Eventually the predicted value cannot be affected by the drift if the time difference of process variable approach is used.

1.3 Objectives

The objectives of the project are:

1 To develop data-driven soft sensors with time difference of process variables.

2 To evaluate its advantages over traditional soft sensor models.

1.4 Scope of study

This study involves the time difference of process variables approach for developing soft sensors. Since the soft sensor is data-driven, an appropriate case study will be selected from the literature. This study will be carried out through simulation of the case study.

1.5 Relevancy of Project

Development of soft sensor is one of the active researches in the area of process control. In addition, the author also focuses on the most common computational learning techniques applied for the Soft Sensor modeling such as Least-Square regression.

1.6 Feasibility of Project

Since the scope of the project is limited to simulation studies, the project is feasible. This is because, the simulations that need to be used for this project is SIMCA-P and Hysys, which is available in the UTP lab so there is no wastage of money and time by purchasing the software.

CHAPTER 2

LITERATURE REVIEW

Soft sensors are precious tools in various industrial backgrounds for the application of process plant. For example, oil and gas refineries, chemical plants, food processing industry, power plants, paper industry, nuclear plants, urban and industrial pollution monitoring. They are used to solve a number of different problems such as measuring system back-up, what-if analysis, real-time prediction for plant control, sensor validation and fault diagnosis strategies. (Luigi Fortuna, *et al.*, 2007).

Figure 2.1: Chemical Plants.

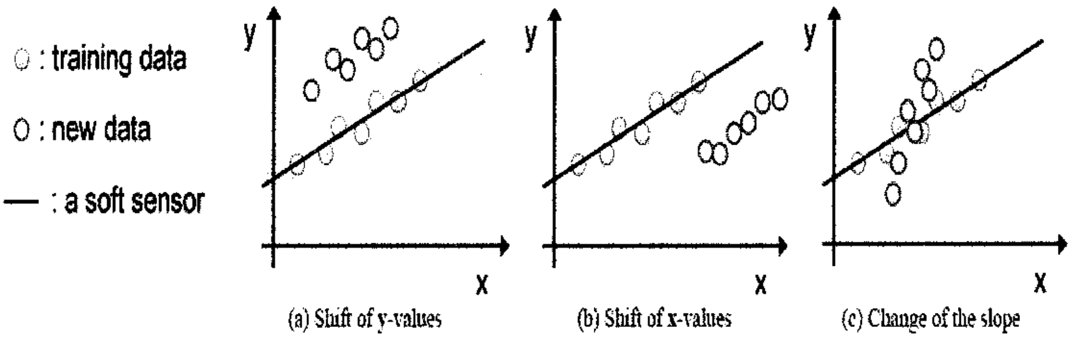
By using soft sensors, y-values, objective variable can be estimated by explanatory variables **X** that can be easily measured online. The explanatory variable is also known as predictor, for instance in a process unit, the variables can be temperature, pressure, flow rate and etc. Meanwhile, the objective variable is the predicted variable which is needed for online measurement such as product concentration, density, melt flow rate and other variable which is hard to measure by hardware instantaneously. Moreover the process can be controlled easily and promptly by using the estimated values. (Okada, *et al.*, 2011).

2.1 Modeling Approach

There are few approaches can be used to develop the soft sensor models such as moving window (MW) model, distance based just-in-time (JIT) model and time difference (TD) model. MW model is constructed with the latest data while JIT model is constructed with data where distances to predict data are smaller than those of other data. According to (Kaneko & Funatsu, 2011), he concluded the characteristic of these approaches as following:

Table 2.1: Summary of characteristic of model approaches.

Type of model approach	Characteristic
Time Difference	Suitable when the shift of **y**-values or **x**-values occurs
Moving Window	Suitable for gradual change of the slope of **x** and **y**.
Just-In-Time	Suitable for instant changes of slope of **x** and **y**.



(a) Shift of y-values (b) Shift of x-values (c) Change of the slope

Figure 2.2: Classification of the degradation of a linear soft sensor model. (Kaneko & Funatsu, 2011).

Unlike those approaches, 'Time difference model' is based on time difference of explanatory variable, **x** and objective variable, **y**. Time difference model can be used when the process unit is in non steady-state condition because during that condition the abnormal data can be detected. Unlike the traditional procedure, the predictive value will be inaccurate during non steady state condition. This is because, in the traditional procedure, it cannot detect the abnormal data accurately since the regressions used are linear regression model. Besides that, a time difference model can adjust shifts of both **y**-values and **x**-values because it attains the same effect as a bias update. (Kaneko & Funatsu, 2011). Furthermore, the parameters of the model, for instance the regression coefficients in linear regression modeling are dramatically changed in some case. Indirectly, this gives low predictive accuracy for traditional procedure.

According to Kaneko (2011), in a traditional procedure, modeling relationship between explanatory variables, **X**(*t*), and an objective variable ,**y**(*t*), is done by regression methods after preparing data, **X**(*t*) and **y**(*t*) related to time *t*. Then, the constructed model predicts the value of **y**(*t′*) with the new data of **x**(*t′*) as shown below:

$$\mathbf{y}(t') = \int[\mathbf{x}(t')] + \mathrm{c} \qquad (1)$$

c = error calculation

Meanwhile for time difference approach, the difference of time for explanatory variables, **ΔX**, and objective variables, Δ**y**, are as shown below:

$$\Delta\mathbf{X}(t) = \mathbf{X}(t) - \mathbf{X}(t\text{-}i) \qquad (2)$$

$$\Delta\mathbf{y}(t) = \mathbf{y}(t) - \mathbf{y}(t\text{-}i) \qquad (3)$$

i = time before the target time

In terms of prediction, the constructed model predicts the time difference of **y**(*t′*), **Δ** **y**(*t′*), with using the time difference of the latest data, **ΔX**(*t′*), the equations are shown below:

$$\Delta\,\mathbf{x}(t') = \mathbf{x}(t') - \mathbf{x}(t'\text{-}i) \qquad (4)$$

$$\Delta\,\mathbf{y}(t') = \mathbf{y}(t') - \mathbf{y}(t'\text{-}i) \qquad (5)$$

y(*t′*) can be calculated as follows because **y**(*t′-i*) is given previously:

$$\mathbf{y}(t') = \Delta\,\mathbf{y}(t') + \mathbf{y}(t'\text{-}i) \qquad (6)$$

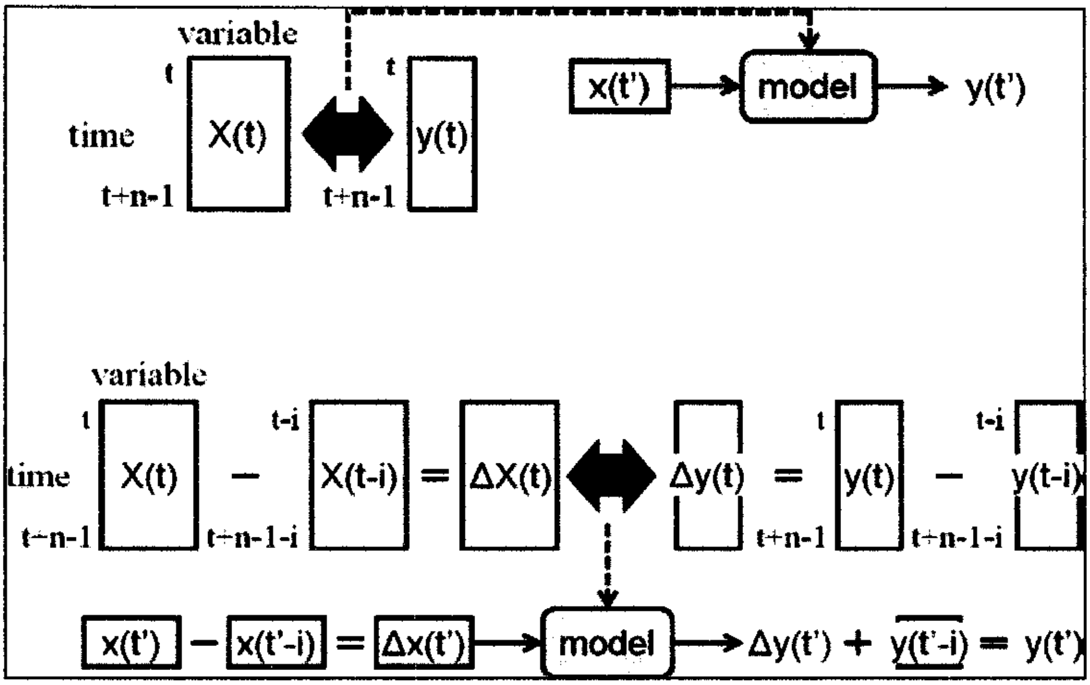


Figure 2.3: Traditional and time difference of process variables procedures.

In the figure above, the difference between a traditional procedure and the proposed, time difference procedure is shown. In order to construct any of the approach or concept, regression method or modeling techniques will be used.

2.2 Modeling Technique

2.2.1 Partial Least Squares (PLS)

Moreover, another commonly used modeling technique is Partial Least Squares or also known as Projection to Latent Structures (PLS) which is the extension of PCA. PLS is a family of multivariate analysis techniques which is used to extract useful information from correlated data. (Samuel Facchin, et al., 2005). The main objective of PLS is to analyze or predict a set of dependent variables from a set of independent variables or predictors. (Abdi, 2010).

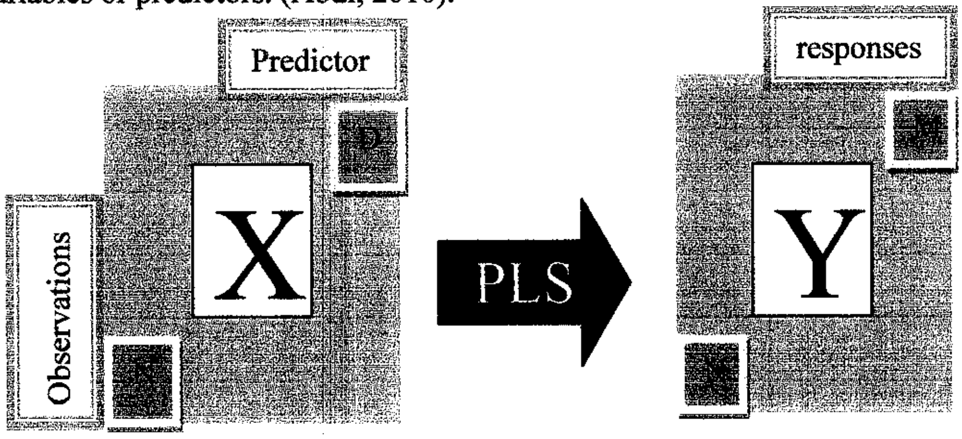


Figure 2.4: PLS Diagram. (Eriksson, et al., 2001).

Based on the figure above, the modeling technique used is Partial Least Square (PLS) and the measurement interval is constant. In X-data set there will be **n** number of rows and **d** number of columns while for Y-data set, there will be **m** number of column and the number of rows is same as the X-data set which is **n**. PLS is a method for relating explanatory variable, X Є $R^{n \times d}$ and an objective variable, y Є $R^{n \times 1}$, (where *n* is the number of sample and *d* is the number of variables). (Hiromasa Kaneko, *et al.*, 2009). In order to check the performance of the model, the error can be calculated by using Mean Squared (MSQ) error equation which is shown as below: (Eliana Zamprogna, 2002)

$$MSQ_i = \frac{(y_i - \hat{y}_i)(y_i - \hat{y}_i)^T}{m_Y} \quad (7)$$

where,

y_i = column vector of measurement of the generic i-th output variable,

$\hat{y}_i$ = estimate obtained from the PLS model,

A data-driven soft sensor derived with PLS deteriorates in the presence of abnormal observations, resulting in model misspecification. Therefore, outlier detection constitutes an essential prerequisite step for design of a data-driven soft sensor (Boa Lin, *et al.*, 2007).

CHAPTER 3

METHODOLOGY

3.1 Introduction

The figure below shows the flowchart for this project.

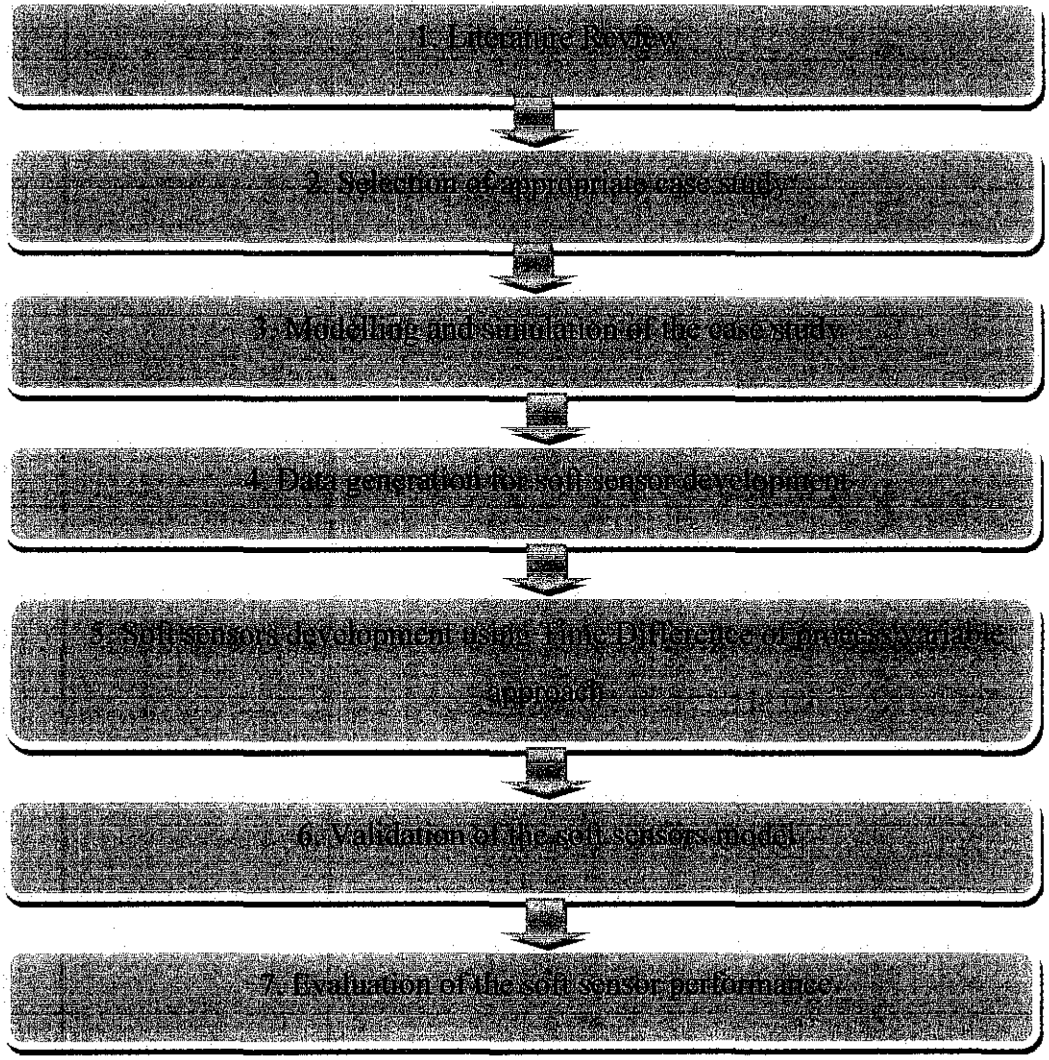

1. Literature Review
2. Selection of appropriate case study
3. Modelling and simulation of the case study
4. Data generation for soft sensor development
5. Soft sensors development using Time Difference of process variable approach
6. Validation of the soft sensors model
7. Evaluation of the soft sensor performance

Figure 3.1: Methodology flowchart.

Firstly, research on the soft sensor models is done using some reliable journals and books. Based on the fundamental knowledge, information on time difference of process variables is gathered and studied. From this information, the modelling techniques of soft sensor development are studied. The modelling techniques are PLS, PCA and SVR. All these modelling techniques are mathematical tools that need to be understood. Then a proper case study is selected for this case to gather the data needed. A proper data is needed for this case to test the efficiency of time difference approach. Once the data is generated from the simulation, the studied approach and modelling technique are applied to develop a soft sensor model. This model will be developed using SIMCA-P software. Then, validation needs to be done for the soft sensor so that it can be implemented in the control system of the chosen case study. Finally the performance of model needs to be evaluated to observe the efficiency of the model.

3.2 Model development

The development of the data-driven soft sensors model will start with the pre-processing of the collected data. The main purpose of pre-processing is to normalize the data to zero mean value and zero standard deviation. In order to normalize the data, outliers need to be removed. Outliers are sensor values which deviate from the normal or typical range of sensor data. Outliers deviate due to the abnormal operating conditions, erroneous measurements, etc., in the data. The identification of stationary state during the data collection period will be performed. There are two type of outliers data exists namely obvious outliers and non-obvious outliers. The difference between the both is the ability to identify the outlier value. This is because, the values of obvious outliers can be easily detected through the violation of the physical or technological limitation. For instance, it is impossible for the value of absolute pressure to be negative value, so it is considered as exceeding the limitation and easily detected. Meanwhile, the values of non-obvious outliers are hard to detect because they do not violate any limitations but deviate from the typical values. Figure below shows the position of the outliers in a set of data.

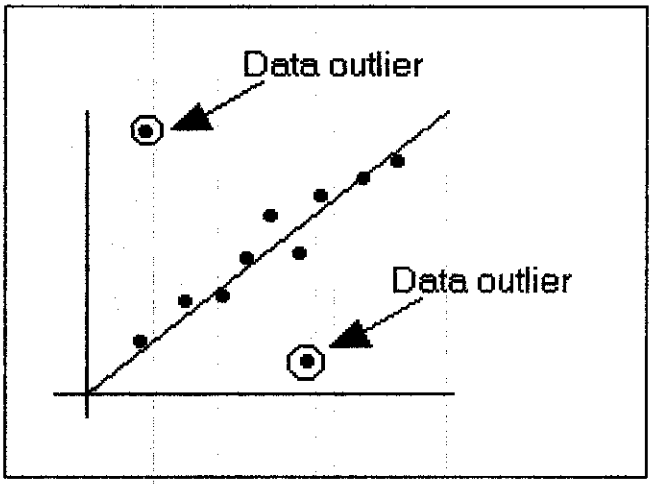


Figure 3.2: Data outliers.

The following step is model structure and regressor selection which is the most crucial part for a soft sensor. Model structure is a set of candidate models among which the model should be searched for. For instance, if the process works close to a steady state condition, a linear model structure can be used, due to the greater simplicity of the design phase. (Luigi Fortuna, *et al.*, 2007). Thus, for regressor selection, it is closely connected with the problem of model structure selection, because it is relevant to the condition of the plant state. For example, in the case of static models, Principle Component Analysis (PCA) and Partial Least Squares (PLS) are valid tools to further simplify the modeling task and avoiding the negative effects of data co-linearity (Luigi Fortuna, *et al.*, 2007). Basically the data measured in the process industry are co-linear due to partial redundancy in the sensor arrangement. For example, two neighboring temperature sensors in a process unit will deliver strongly correlated measurements to the system. Co-linearity can be handled by selecting a subset of the input variables which is less co-linear. Next is model validation, which will verify that, model residuals are not correlated with model inputs and that their autocorrelation function is an impulse function. (Luigi Fortuna, *et al.*, 2007).

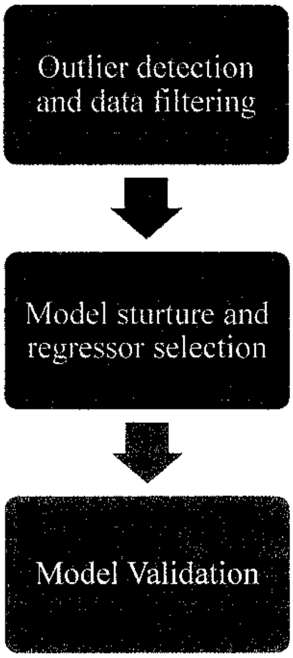

Outlier detection and data filtering
Model sturture and regressor selection
Model Validation

Figure 3.3: Block scheme of soft sensor development.

3.3 Least-Squares Regression

Least-squares regression is a derivation of an approximate function that best fits a given set of data points. (Dechaumphai, 2011). There are two types of regression in Least-Squares regression namely linear regression and multiple regressions.

3.3.1 Linear regression

Linear least-squares regression is a method for fitting a set of data that tends to vary linearly which the coefficients a_1 and a_0 of a linear function as shown below:

$$g(x) = a_0 + a_1 x \tag{8}$$

The main purpose of this method is to minimize the squares of the differences between the data values and the function values. The best fit is the smallest possible total error. The graph below explains more detail about linear regression.

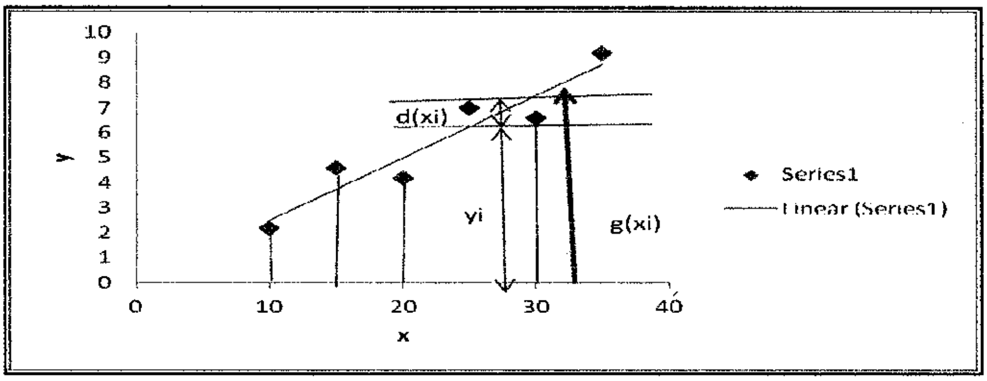


Figure 3.4: Linear regression method for data that tend to vary linearly.

The total error that occurs from all n data is:

$$E = \textstyle\sum_{i=1}^{n} \left[d(x_{i})\right]^2 \tag{9}$$

Where i is data points, the equation can be rearrange as shown below:

$$E = \textstyle\sum_{i=1}^{n} \left[y_i - g(x_{i})\right]^2 \tag{10}$$

By substituting equation 8 into equation 10, the function will be as shown below:

$$E = \textstyle\sum_{i=1}^{n} \left[y_i - (a_0 + a_1 x_{i})\right]^2 \tag{11}$$

The function E has a minimum at the values of a₁ and a₀ where partial derivative of E with respect to each variable is equal to zero.

$$\frac{\partial E}{\partial a_0} = 0 \tag{12}$$

$$\frac{\partial E}{\partial a_1} = 0 \tag{13}$$

From equation 12,

$$2\sum_{i=1}^{n}[y_i - (a_0 + a_1 x_i)](-1) = 0 \tag{14}$$

$$\sum_{i=1}^{n} y_i - \sum_{i=1}^{n} a_0 - \sum_{i=1}^{n} a_1 x_i = 0 \tag{15}$$

$$n a_0 + \left(\sum_{i=1}^{n} x_i\right) a_i = \sum_{i=1}^{n} y_i \tag{16}$$

From equation 13,

$$2\sum_{i=1}^{n}[y_i - (a_0 + a_1 x_i)](-x_i) = 0 \tag{17}$$

$$\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} a_0 x_i - \sum_{i=1}^{n} a_1 {x_i}^2 = 0 \tag{18}$$

$$\left(\sum_{i=1}^{n} x_i\right) a_0 + \left(\sum_{i=1}^{n} x_i^2\right) a_1 = \sum_{i=1}^{n} x_i y_i \tag{19}$$

Combine Equation 16 and 19 in the matrix form as shown below:

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \end{Bmatrix} \tag{20}$$

The solution of the system is:

$$a_0 = \frac{\left(\sum_{i=1}^{n} y_i\right)\left(\sum_{i=1}^{n} x_i^2\right) - \left(\sum_{i=1}^{n} x_i y_i\right)\left(\sum_{i=1}^{n} x_i\right)}{n\left(\sum_{i=1}^{n} x_i^2\right) - \left(\sum_{i=1}^{n} x_i\right)^2} \tag{21}$$

$$a_1 = \frac{n\left(\sum_{i=1}^{n} x_i y_i\right) - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{n\left(\sum_{i=1}^{n} x_i^2\right) - \left(\sum_{i=1}^{n} x_i\right)^2} \tag{22}$$

3.3.2 Case study

A set of data for the wind velocities measured at different elevations of a building is shown in the table below:

Table 3.1: Data of wind velocities at different elevations of the building.

Building elevation, x (m)	Wind velocity, y (m/sec)
10	2.2
15	4.6
20	4.2
25	7.0
30	6.6
35	9.2

To calculate the value of a_0 and a_1, x_i^2 and x_iy_i data for all the elevation needed.

Table 3.2: Values required for linear regression calculation purpose.

x_i	y_i	x_i^2	x_iy_i
10	2.2	100	22
15	4.6	225	69
20	4.2	400	84
25	7.0	625	175
30	6.6	900	198
35	9.2	1,225	322
Σ = 135	Σ = 33.8	Σ = 3,475	Σ = 870

Substitute the values from the table in equation 21 and equation 22 to get the a_0 and a_1 value.

$$a_0 = \frac{(33.8)(3{,}475)-(870)(135)}{6(3{,}475)-(135)^2} = 0.001904 \qquad (23)$$

$$a_1 = \frac{6(870)-(135)(33.8)}{6(3{,}475)-(135)^2} = 0.250286 \qquad (24)$$

So, the fitted value is, $g(x) = 0.001904 + 0.250286x$

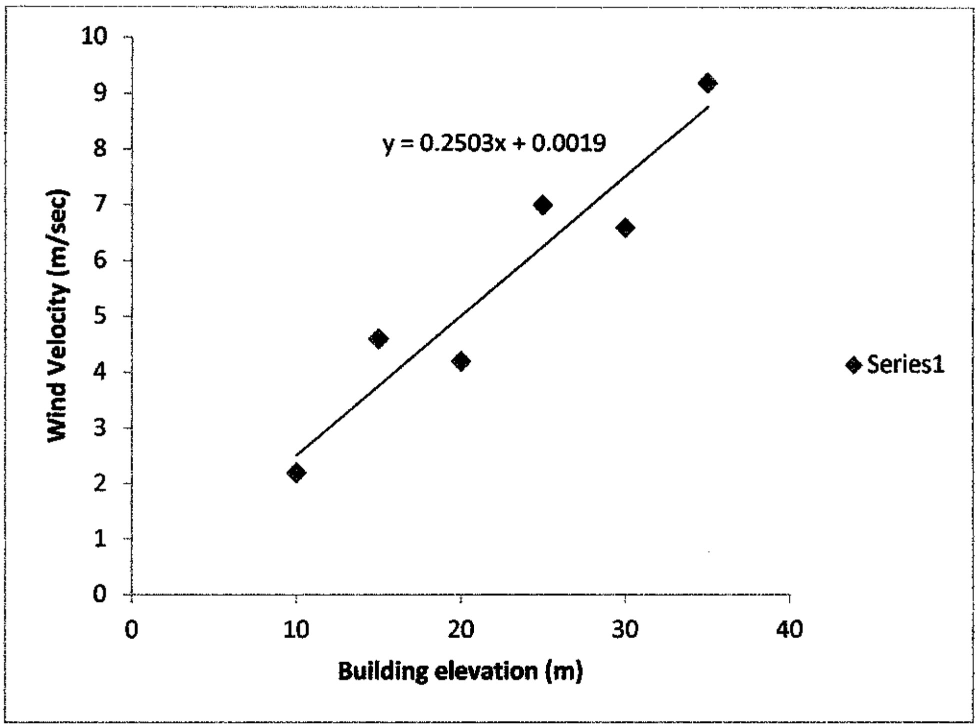


Figure 3.5: Comparison between the fitted function and data.

Refer to appendix 1 for MATLAB coding for this problem solving.

3.3.3 Multiple Regression

The difference in this method is the fitted function y are dependent of many variable of (x_1, x_2, x_3,........ x_k) and this is written as: (Dechaumphai, 2011)

$$y = y(x_1, x_2, x_3, \dots \dots \dots \dots \dots . x_k)$$

where k is the number of the independent variables.

The fitted g function is:

$$g = a_0 + a_1x_1 + a_2x_2 + \dots \dots \dots + a_kx_k \tag{25}$$

The total error E is,

$$E = \sum_{i=1}^{n}[y_i - (a_0 + a_1x_1 + a_2x_2 + \dots \dots \dots + a_kx_k)]^2 \tag{26}$$

The function E has a minimum at the values of a$_0$ until a$_k$ where partial derivative of E with respect to each variable is equal to zero.

$$\left.\begin{aligned}\frac{\partial E}{\partial a_0} &= 0\\ \frac{\partial E}{\partial a_1} &= 0\\ \frac{\partial E}{\partial a_2} &= 0\\ \vdots &= \vdots\\ \frac{\partial E}{\partial a_k} &= 0\end{aligned}\right\} \tag{27}$$

The derivation in equation 27 is solved by using the same method as mentioned in equation 12 and 13. The details from the derivation or the minimization process are written in matrix form as shown below:

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{2i} & \dots & \sum_{i=1}^{n} x_{ki} \\ \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{1i}x_{1i} & \sum_{i=1}^{n} x_{1i}x_{2i} & \cdots & \sum_{i=1}^{n} x_{1i}x_{ki} \\ \sum_{i=1}^{n} x_{2i} & \sum_{i=1}^{n} x_{1i}x_{2i} & \sum_{i=1}^{n} x_{2i}x_{2i} & \cdots & \sum_{i=1}^{n} x_{2i}x_{ki} \\ \vdots & \vdots & \vdots & & \vdots\ddots \\ \sum_{i=1}^{n} x_{ki} & \sum_{i=1}^{n} x_{1i}x_{ki} & \sum_{i=1}^{n} x_{2i}x_{ki} & \cdots & \sum_{i=1}^{n} x_{ki}x_{ki} \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_k \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_{1i}y_i \\ \sum_{i=1}^{n} x_{2i}y_i \\ \vdots \\ \sum_{i=1}^{n} x_{ki}y_i \end{Bmatrix}$$

By using Gauss Elimination method, the matrix form above is solved and can get the coefficient value of a$_0$ until a$_k$.

3.3.4 Case study

Use the multiple linear regression method to fit the data with two independent variables as shown in the table below:

Table 3.3: Data variables of multiple regression.

i	x_{1i}	x_{2i}	y_i
1	0	0	1
2	0	1	4
3	1	0	3
4	1	2	9
5	2	1	8
6	2	2	11

In order to calculate the coefficient values, the data of x_{1i}, x_{2i}, y_i, $x_{1i}x_{1i}$, $x_{1i}x_{2i}$, $x_{2i}x_{2i}$, $x_{1i}y_i$ and $x_{2i}y_i$ are needed and it is tabulated as below:

Table 3.4: Values required for linear regression calculation purpose.

I	x_{1i}	x_{2i}	y_i	$x_{1i}x_{1i}$	$x_{1i}x_{2i}$	$x_{2i}x_{2i}$	$x_{1i}y_i$	$x_{2i}y_i$
1	0	0	1	0	0	0	0	0
2	0	1	4	0	0	1	0	4
3	1	0	3	1	0	0	3	0
4	1	2	9	1	2	4	9	18
5	2	1	8	4	2	1	16	8
6	2	2	11	4	4	4	22	22
Σ	6	6	36	10	8	10	50	52

Substitute the values from the table in equation above to get the a_0, a_1 and a_2 values.

$$\begin{bmatrix} 6 & 6 & 6 \\ 6 & 10 & 8 \\ 6 & 8 & 10 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} 36 \\ 50 \\ 52 \end{Bmatrix}$$

So, after solving the matrix form above the coefficient values are:

$a_0 = 1$, $a_1 = 2$ and $a_3 = 3$

Thus the fitted value function based on equation 25, is $g = 1 + 2x_1 + 3x_2$.

Refer to Appendix 2 for MATLAB coding for this problem solving.

3.3.5 Partial Least Squares regression

3.3.5.1 Pre-processing of data

Before constructing the X and Y co-ordinate systems, the data should be pre-treated through scaling and mean-centering. In a process plant, the value of all the variables varies from very small value to very large value and this affect the result. This is because variable with a large variance is more prone to be expressed in the modeling compare to the low variance. For instance, flow rate variable which usually have large variance will overcome the mole fraction variable which is equal or less than 1.0. Indirectly, this affects the result of estimation. In order to avoid this problem, the data need to be scale it so that the range of all the variables will be equally distributed and once the modeling is done, the data can be de-scaled it to get the original predicted value. The following step after scaling is mean-centering for pre-processing. This step is important because it can minimize the error in the data. Illustration in the figure below shows the method of data pre processing.

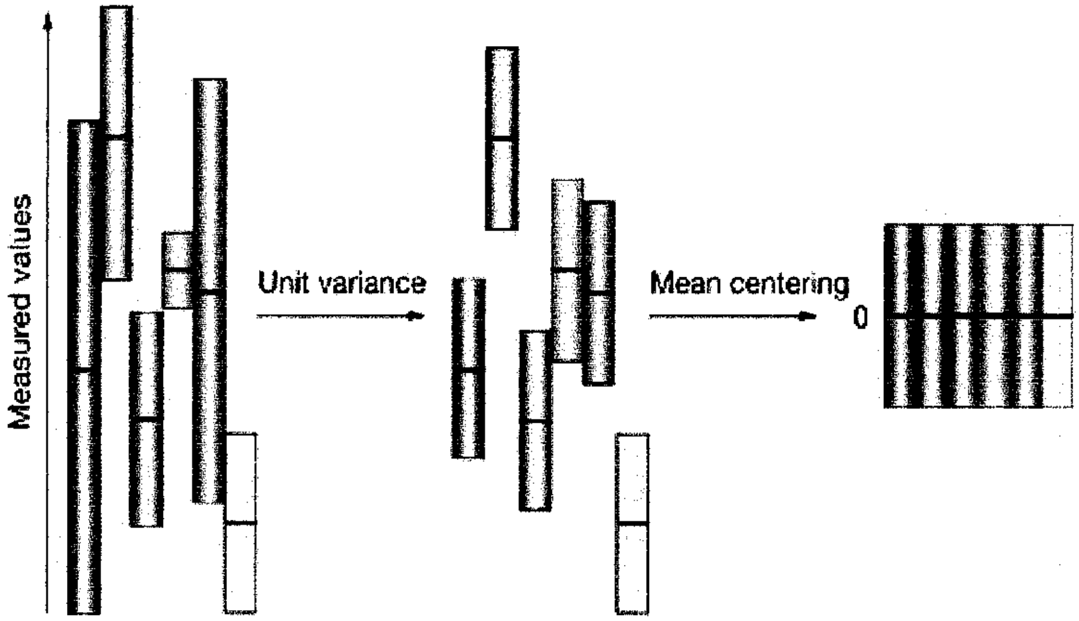


Figure 3.6: Unit variance scaling and mean-centering. (Jacob Bjerrum, 2008)

3.3.5.2 Geometry of PLS

Once the pre processing is done, the data can be used for the computation and modelling. The first step to construct the PLS model is, set up the K-dimensional space with variables where each column of X represents one co-ordinate axis. Next step is plotting the observations (each rows) data in K-dimensional space. (Svante Wold, 2001). Then, the following step is to calculate the first PLS component. At this part, the first component approximates the point-swarm in the X-space and provides a good correlation with the y-vector. The projections of each data towards the line in the X-space give the score of each observation. The score vector mentioned for the first component is t1 and the weight of the y-vector is c1. For the second component, the line will be perpendicular from the first component line and the projections for it give score t2 and the weight of the y-vector is c2. Those two components combines together to define a plane in the X-space. By combining these variables, we can get more accurate results for this predictive model. The illustration of geometric representation of PLS regression is shown as below.

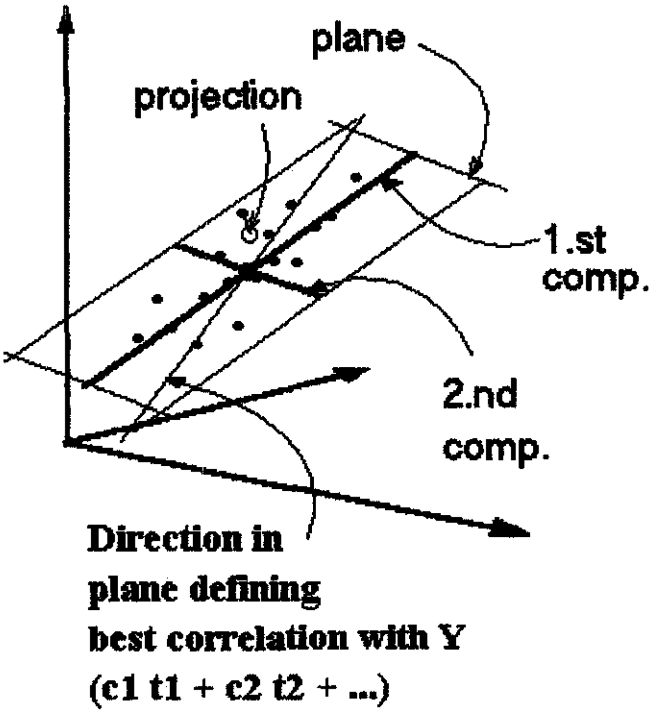


Figure 3.7: The geometric representation of PLS regression.

3.3.5.3 PLS calculation method

With the knowledge of Least squares of regression which is the basic of PLS, the author continue to practice the partial least square regression method to prepare for next step application purpose in the time difference in process variables approach. The calculation methods or steps for PLS are discussed as shown below:

PLS decomposes the (nxN) matrix of zero-mean variables X and (nxM) matrix of zero-mean variables y as shown below:

$\mathbf{X}=\mathbf{TP}^{T}+\mathbf{E}$ (28)

$\mathbf{y}=\mathbf{UQ}^{T}+\mathbf{F}$ (29)

where,

T and **U** = (n x p) of p extracted score vector

P = (N x p) , matrix loading

Q = (M x p), matrix loading

E = (n x N) matrices of residual

F = (n x M) matrices of residual

Then the properties of PLS regression can be calculated by using the NIPALS algorithm. The first step is to form two matrices which is **E** = **X** and **F** = **Y**, where these matrices should be normalized (z-scores). Then, the vector u is assumed with random values and α denotes as 'to normalize the result of the operation'. NIPALS algorithm iteration is as shown below: (Abdi, 2010).

Step 1: $w \alpha E^{T}u$ (to estimate X weights)

Step 2: $t \alpha Ew$ (to estimate X factor scores)

Step 3: $c \alpha F^{T}t$ (to estimate Y weights)

Step 4: $u = Fc$ (estimate Y scores).

Step 1 need to repeat until t has converged. Once it is converged, compute the value of b, $b=t^{T}u$ and compute p value, $p=E^{T}t$. The next step is to deflate the matrices of E and F by subtracting the effect of t. (Abdi, 2010)

$E = E - tp^{T}$ (30)

$F = F - btc^{T}$, scalar b, stored as a diagonal element of B.

If E is a null matrix, then the whole set of latent is correct but if it is not the iteration need to be repeat until E is a null matrix. The dependent variables are predicted using the equation of $\hat{Y} = TBC^T$. By following all the steps as shown above, $\hat{y}$ (predicted value) in matrix form can be obtained.

3.3.6 Case study

The total data set of biochemical oxygen demand is (20 x 6) in the file of moore.mat. The predictor set in matrix form for this case is (20 x 5) while the predicted set is (20 x 1). The predictor for this case is X_0 and X_1 while the objective variable is y only. By using *plsregress* function in MATLAB, the function can be solved easily. The MATLAB coding for this case is attached in Appendix 3. From the MATLAB simulation, the solution for this case is showed in the graph below:

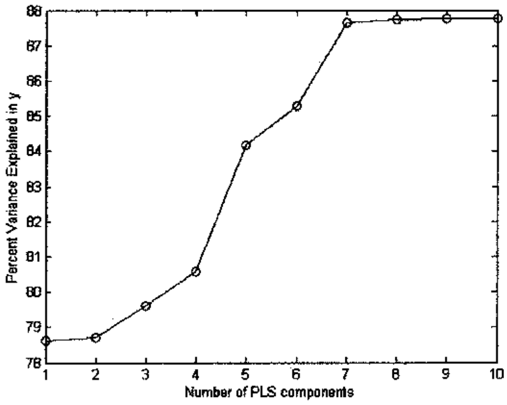


Figure 3.8: Graph of % variance explained in y versus number of PLS component.

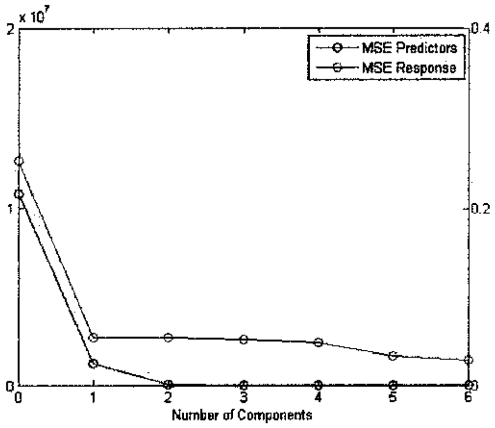


Figure 3.9: Graph of Mean squared errors versus number of components.

From the figure above, it shows that two numbers of components is sufficient for this case study. The root mean square for this case study is 0.8529 which is good result.

3.4 Gantt chart

No	Detail / Week	1	2	3	4	5	6	7		8	9	10	11	12	13	14	15
1	Study on simulation	■	■						Mid Semester Break								
2	Gather and analysis the data		■	■	■	■	■	■									
3	Submission of Progress Report									●							
4	Modeling and validating the time difference of process variable model									■	■	■	■	■			
4	Pre-SEDEX and first draft submission												●				
5	Submission of dissertation(softcopy)													●			
6	Submission of technical paper														●		
7	Oral presentation																●
8	Submission of project dissertation (hard bound)																●

Figure 3.10: Gantt chart for the second semester project implementation.

■ Processes

● Milestones

3.5 Tools required

Since this is a simulation project, software is the basic tool required. The software needed is SIMCA-P which is the standard in multivariate data analysis. By using this software, model development for soft sensor development can be easily done. This software was developed by Umetrics. It is a commercial tool that transform the data into information and provide complete solution for both off-line and on-line data analysis (continuous and batch processes). This software can be used for many purposes; mainly for this project it is useful for the math and computational. Besides that, it is also can be used for PLS modeling technique methods. Apart from that, Hysys is used in this project. Hysys is a simulation based software and commercially used in the industry. The virtual plant from the case study is built in Hysys environment. By using Hysys, data of the process unit can be extracted.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Distillation Column

For this case study, a binary distillation column with dynamic mode simulation is used. This column consists of three main streams, which are feed stream, top product and bottom product stream. The liquid mixture which consists of acetone and 2-propanol is fed to the distillation column through the feed stream. Then, the feed flows down to the column and settles there. However, due to the heat supply from the reboiler, the lower boiling point components in the liquid mixture (acetone) will vaporize. The heat source for the reboiler is the steam. Meanwhile, the remaining liquid (2-propanol) will be removed by the reboiler through the bottom product stream. The vapor released will flow to the top of the column and cooled down by the condenser. The condensed liquid will be discharged through the top product stream. However, there will be some percentage of condensed liquid will be returned to the top column as reflux. Overall, this simulation is about the distillation of acetone and 2-propanol where acetone is the main top product and 2-propanol is the main bottom product of the distillation.

4.1.1 Details of the distillation column

The details for the distillation column are shown as below:

Table 4.1: Specification of the distillation column.

Specification	Description
Height	5.5m
Diameter	150mm
Number of trays	15
Type of trays	Bubble cap
Tray spacing	350mm
Feed tray location	Tray 7

The operating condition of the distillation column used in the simulation is shown as below:

Table 4.2: Operating conditions of the column.

Parameter	Operation data
Feed Flow rate	0.6646 kmol/h
Feed acetone mole fraction	0.3
Feed 2-propanol mole fraction	0.7
Reflux Flow rate	1.051 kmol/h
Distillate Flow rate	0.1974 kmol/h
Top acetone mole fraction	0.9843
Top 2-propanol mole fraction	0.0157
Bottom product flow rate	1.5051 kmol/h
Bottom acetone mole fraction	0.0271
Bottom 2-propanol mole fraction	0.9729
Steam flow rate	18.0285 kg/h
Top temperature	78.60 ℃
Bottom temperature	83.86 ℃
Feed temperature	47°C
Column pressure	1.013 bar

4.1.2 Schematic drawing of distillation column

The schematic drawing of this distillation column is shown as below:

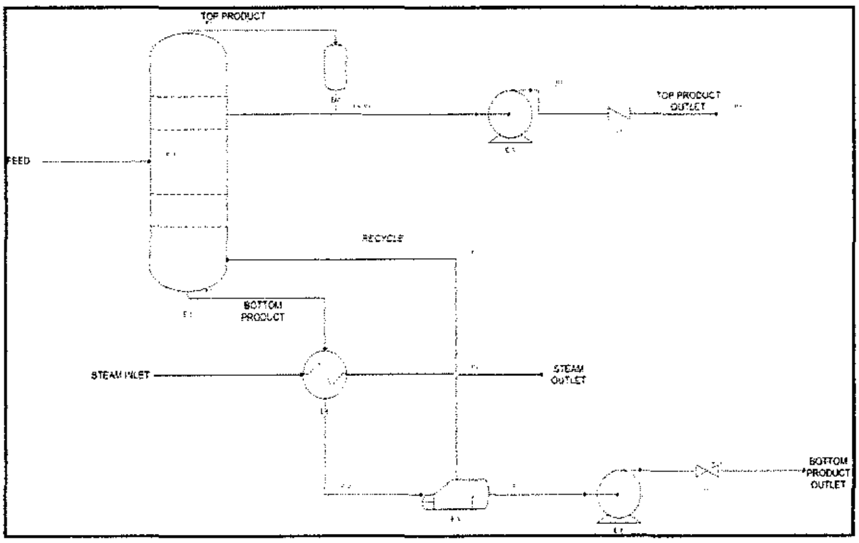


Figure 4.1: Schematic drawing of the distillation column.

The distillation model in the laboratory is shown as below:

Figure 4.2: Distillation column model.

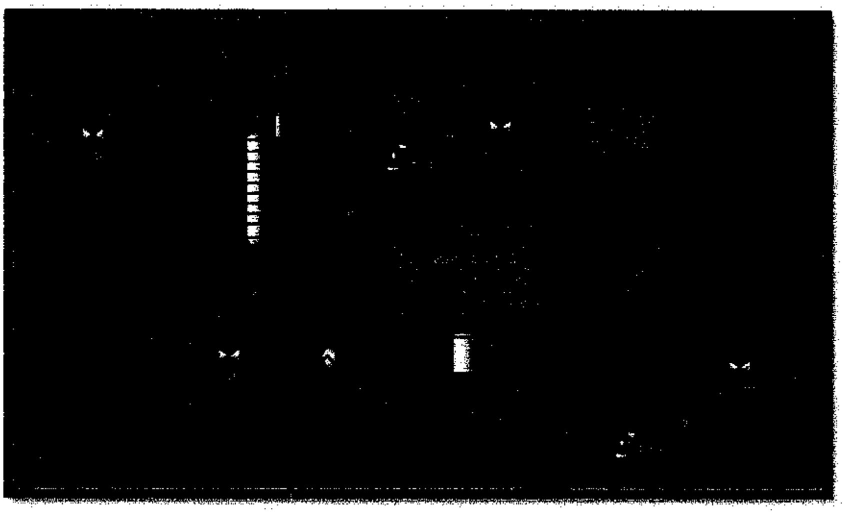


Figure 4.3: Hysys snapshot of the virtual distillation column.

4.2 Simulation studies

Using the distillation column simulation as mentioned above, the input variables are varied to generate a quality data. Those variables are the feed temperature, feed flow rate, steam flow rate and reflux flow rate. Before the input variables are changed, the simulation is modified to fit the case study. There are few problems in the simulation, mainly, the separation is very poor, where the mole fraction of acetone at top column is just 0.4. Moreover, the tray efficiency is about 0.06 merely, and this is very low for an efficient distillation column. Then, the feed temperature does not match with the top and bottom temperature of the column. This is because, the feed temperature is too low for the separation process which is only 28℃ whereas the top and bottom temperatures of the column are 66℃ and 82℃ respectively. In order to solve these problems, few steps are taken and the flow is as shown below:

1) The tray efficiency is increased gradually to 0.85 with the increment of 2%.
2) The feed temperature is increased to 40 ° C with the increment of 2% too.
3) The reflux flow rate is increased to 0.08211 m^3/h with 20% increment. At the same time, the control valve and PV value is changed to 0.0864.
4) The steam flow rate is increased about 6% where the flow rate is 18.0285 kg/h.
5) The feed temperature is increased again until it reaches 47 ° C.

For each step mentioned above, the time is set to 1000s to run the simulation. At the same time, the reflux ratio is observed so that the ratio is maintained below 6. After step 5, the simulation reached the steady-state mode with 0.98 of top acetone mole fraction. By using this modified simulation, the input variables are varied about 2% of step change with the range of ±10% for 1000 seconds. The pattern of step changes is shown clearly in the graph below:

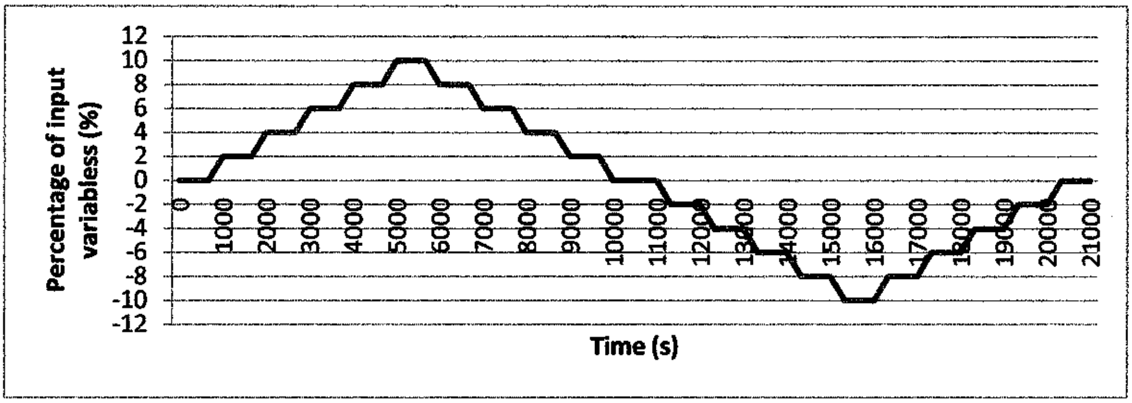


Figure 4.4: Changes of the percentage for all the input variables with time.

4.3 Data analysis and preparation

Based on the step changes of percentage study, the step change at 3000s of the input variables shows the most fluctuation, so the detailed analysis of those input variables, are shown in the graphical method as below:

a)

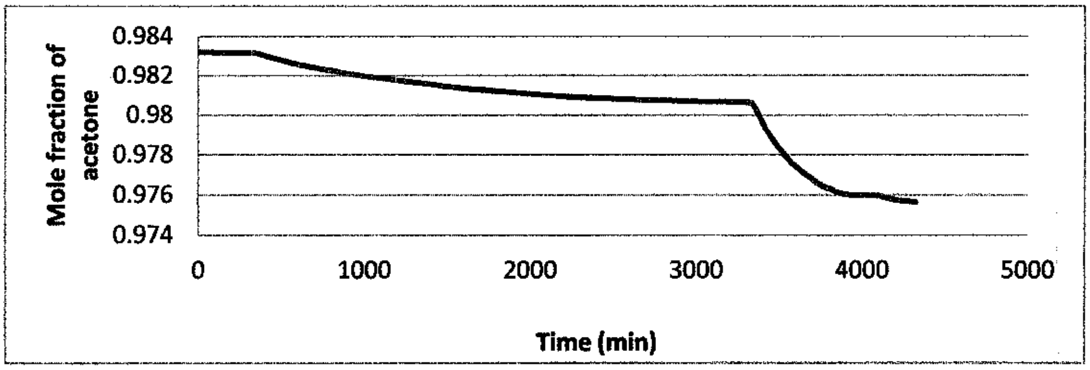


Figure 4.5: Transient response in the top product composition for step change in feed temperature.

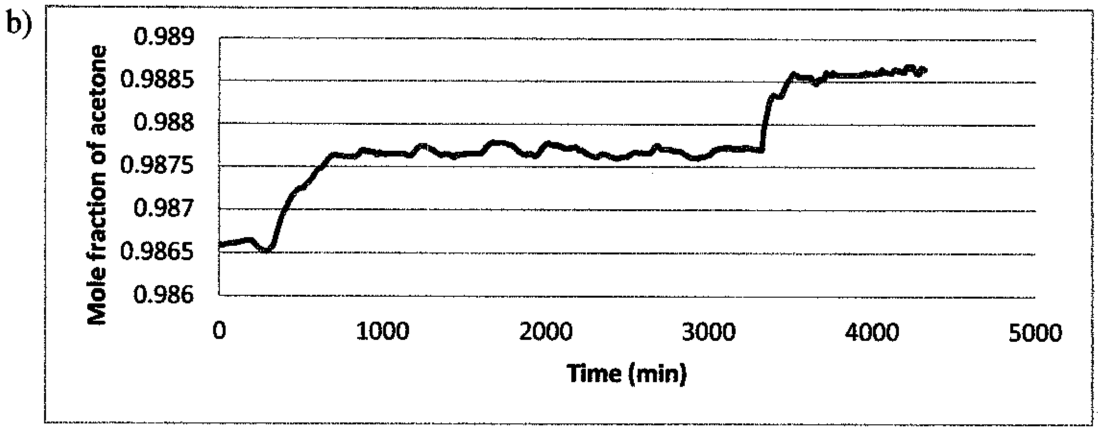


Figure 4.6: Transient response in the top product composition for step change in reflux flow rate.

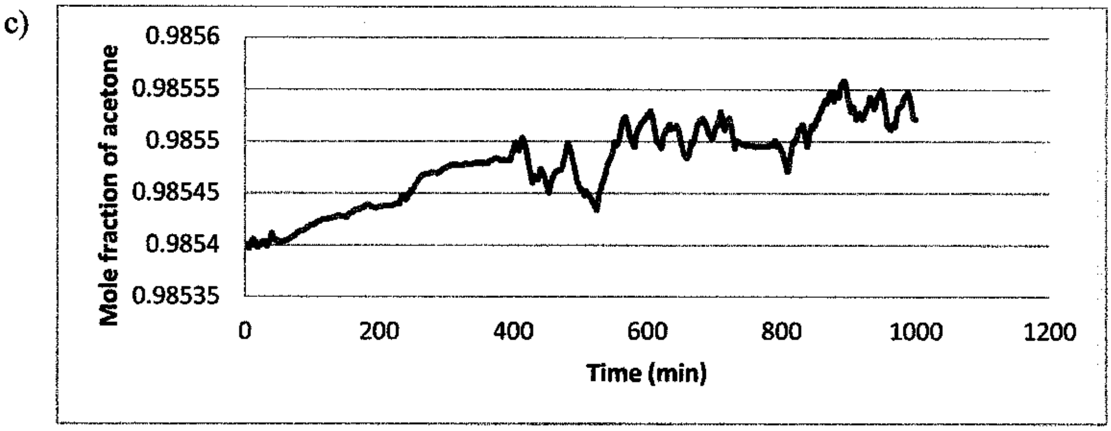


Figure 4.7: Transient response in the top product composition for step change in feed flow rate.

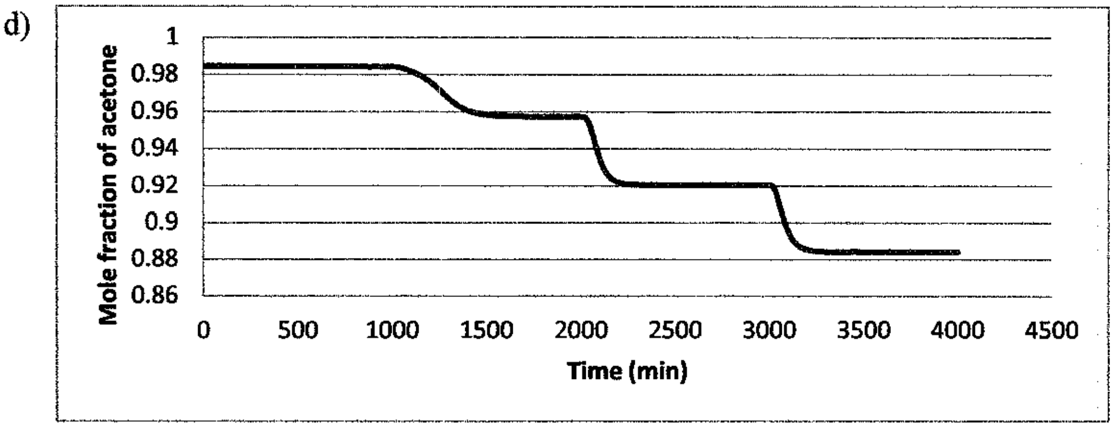


Figure 4.8: Transient response in the top product composition for step change in steam flow rate.

According to the figures above, transient response for feed temperature fluctuate less compare to other input variables changes. Besides that, mole fraction of acetone fluctuates a lot as the time increases compare to other input variables.

Temperature profile of the distillation column for the selected input variables are shown as below:

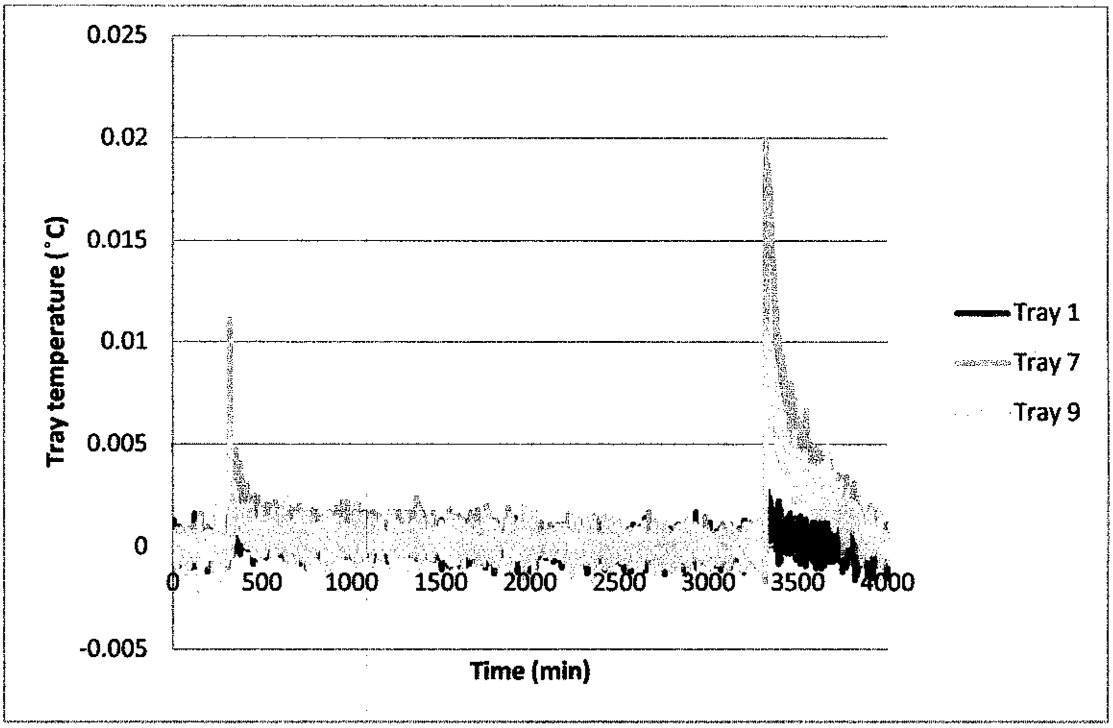


Figure 4.9: Temperature profile for step change in feed temperature.

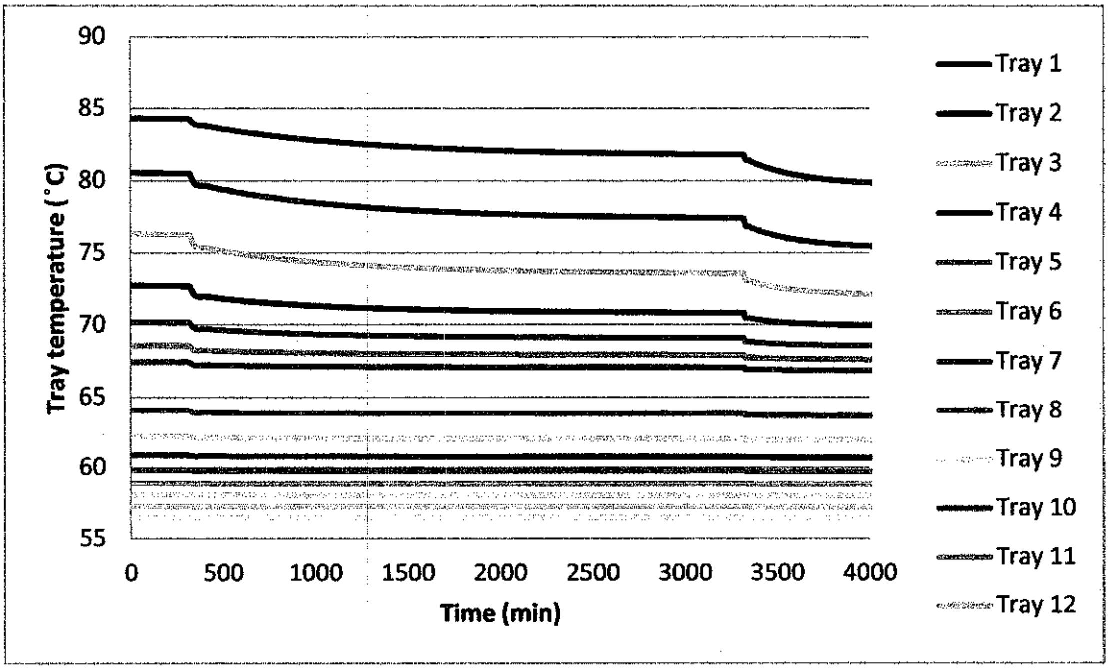


Figure 4.10: Temperature profile for step change in reflux flow rate.

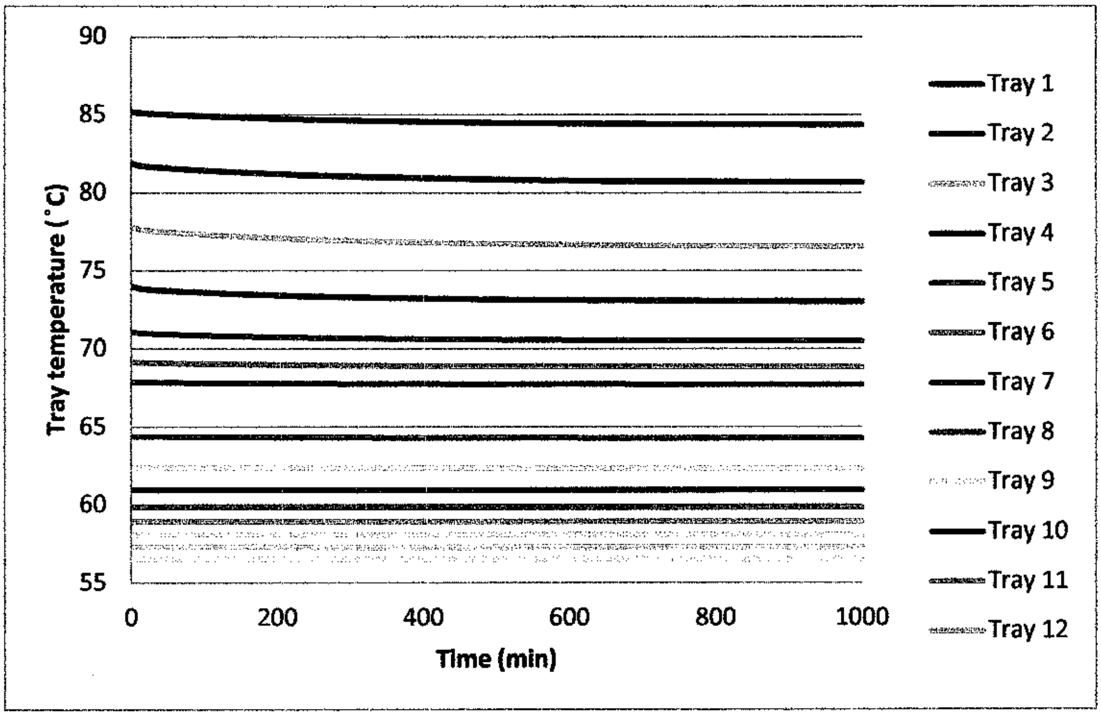


Figure 4.11: Temperature profile for step change in feed flow rate.

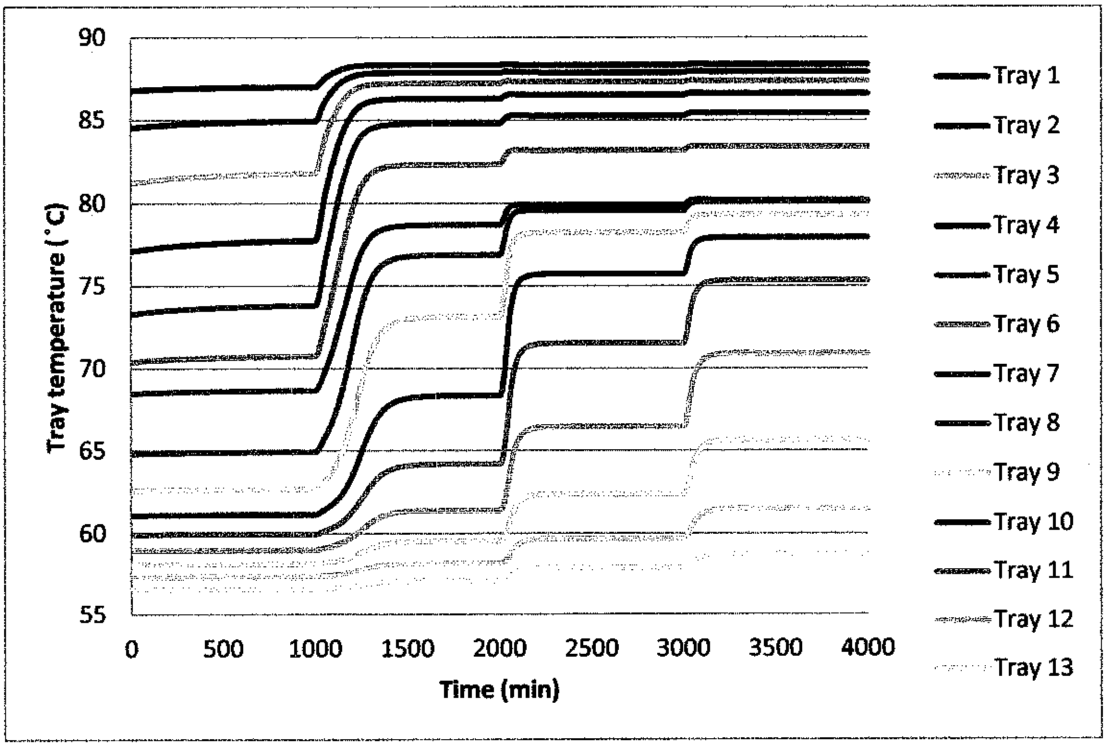


Figure 4.12: Temperature profile for step change in steam flow rate.

According to the temperature profile above, step change in feed temperature affects the most followed by the step change in steam flow rate. Meanwhile, step change in feed flow rate does not affect the tray temperature throughout the process.

Since the response of those input variables meets the criteria for this case study, they are combined in series form as a new data set. The current data set consist of 17,280 samples with specified operating conditions. The operating conditions are feed molar flow, mole fraction of acetone in the top product, all the tray temperatures, steam flow rate and feed temperature. For this case study, about five set of data is prepared to simulate in SIMCA-P software using PLS modeling technique. Those set of data are prepared as below:

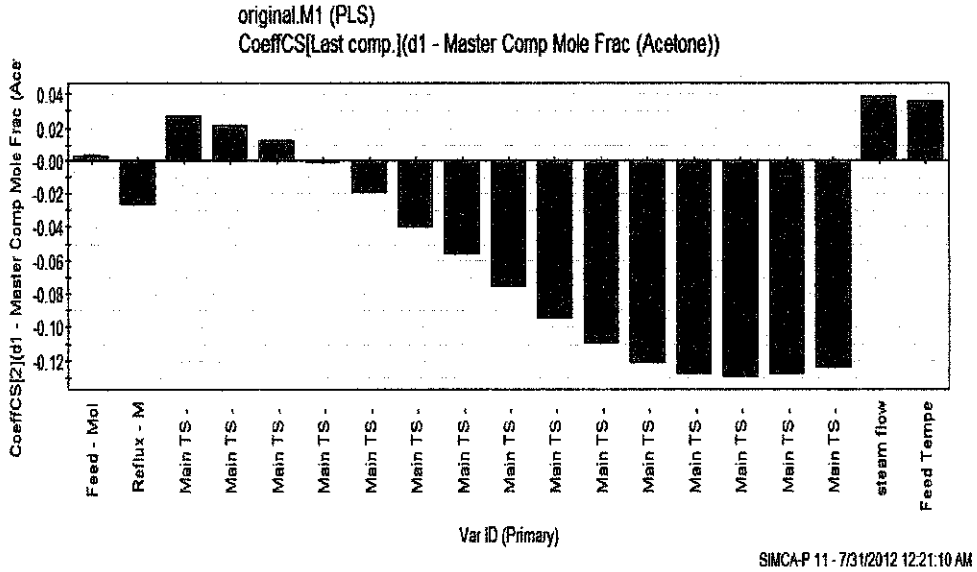
(t-i) , where i= 0, 1, 2, 3& 4 (time before target time) t= time at instantaneous.

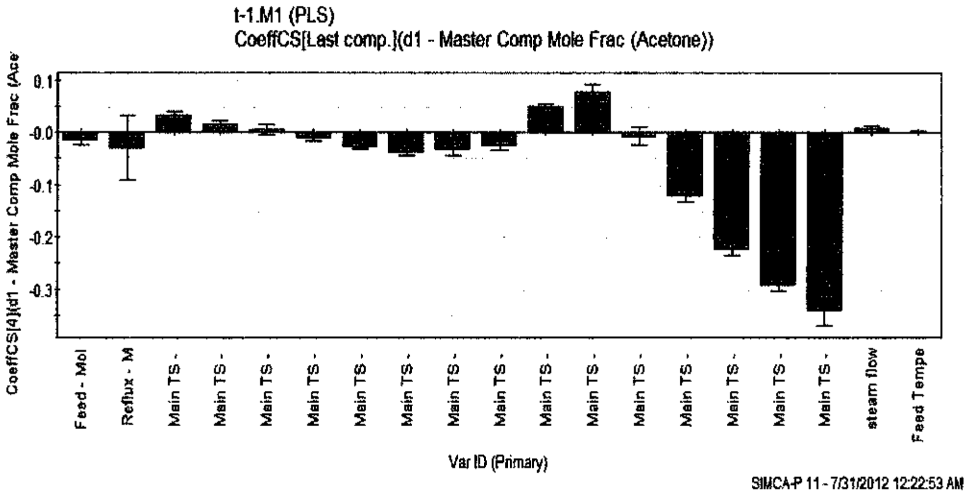
a) (t) data – Current data set without any changes in time.
b) (t-1) data – Data with 1 minute difference.
c) (t-2) data – Data with 2 minute difference.
d) (t-3) data – Data with 3 minute difference.
e) (t-4) data – Data with 4 minute difference.

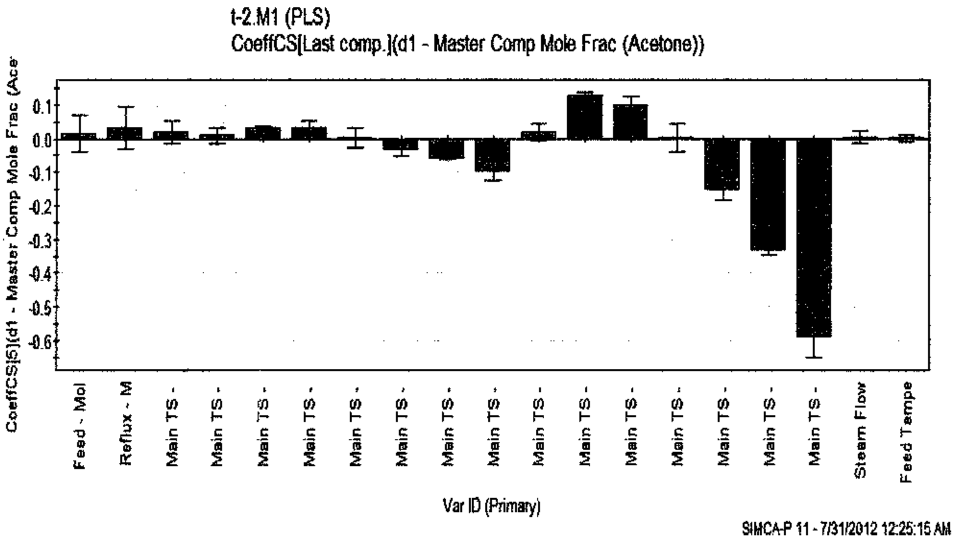
4.4 Development of soft sensors using PLS technique

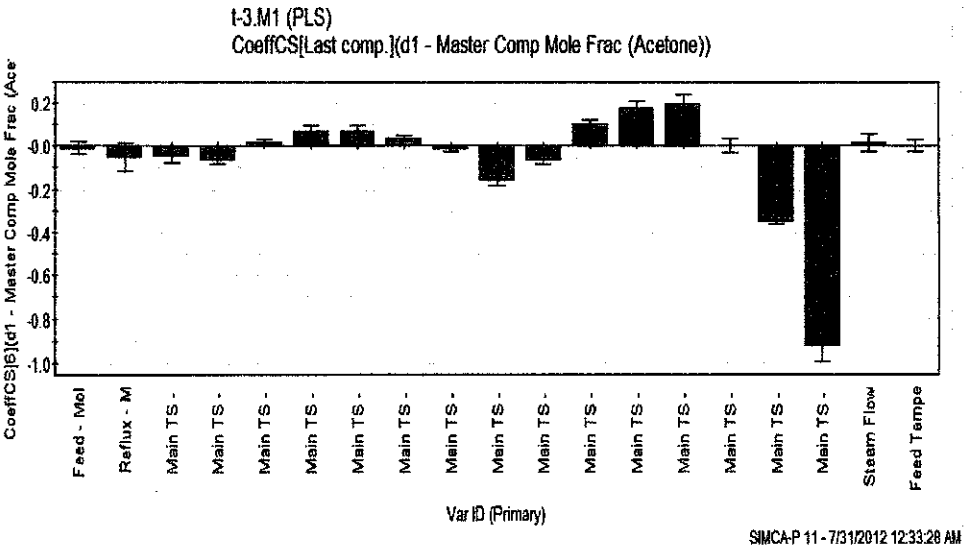
All the data sets as mentioned previously are simulated using SIMCA-P and the coefficient plot is shown as below:

Table 4.3: Coefficient Plot.

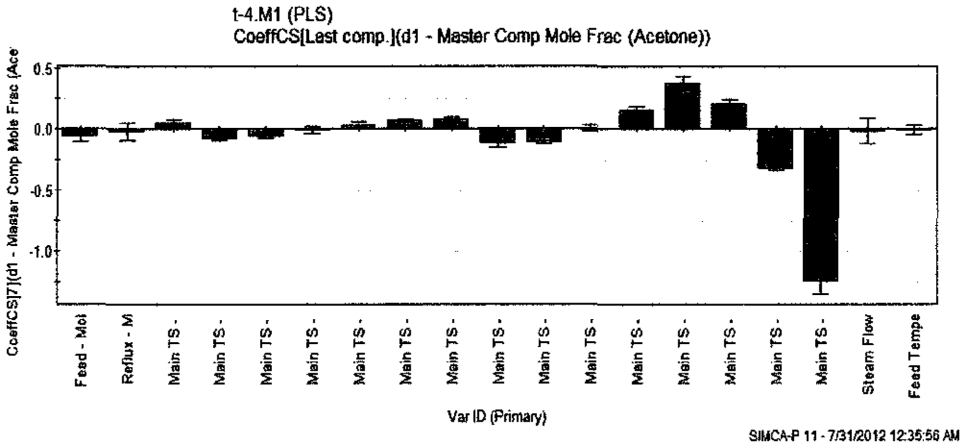
Data set	Coefficient Plot
t	

<table>
<tr><th>Data set</th><th>Coefficient Plot</th></tr>
<tr><td>t-1</td><td>


</td></tr>
<tr><td>t-2</td><td>


</td></tr>
<tr><td>t-3</td><td>


</td></tr>
</table>

Data set	Coefficient Plot
t-4	

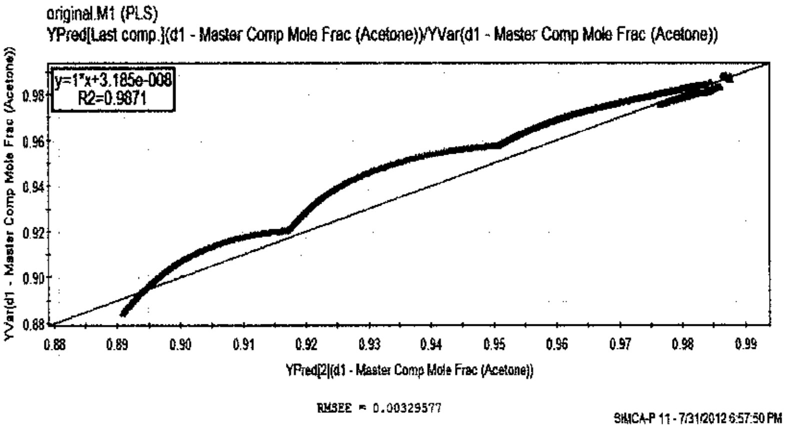
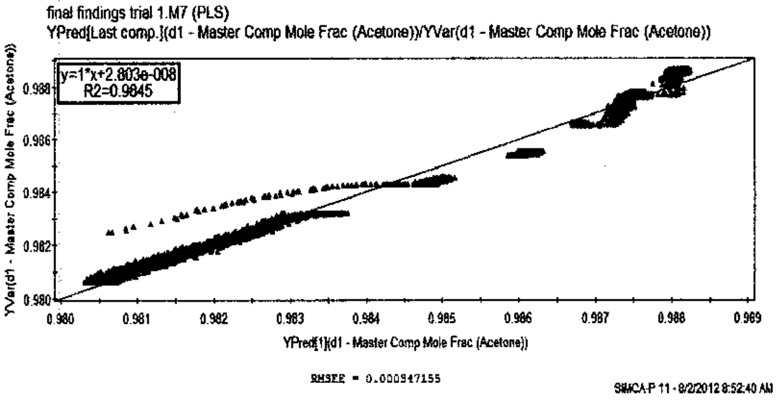
Coefficient plot shows the regression model for each data set and those coefficients refer to scaled and centered X-data, meanwhile the Y-data is scaled but not centered. The scaling technique used for this simulation is unit variance (UV-scaling). The scaled data makes the coefficient more comparable to each other. The bar indicated the confidence level of the coefficients and it is significant if the bar length is small. Moreover the green shaded box represents the average value of the variable. From the observation of table 8, the considerable input variables are selected to generate a new data set. The considerable input variables are selected based on the size of confidence interval and also the average value of input variables. Those selected input variables are tabulated as below:

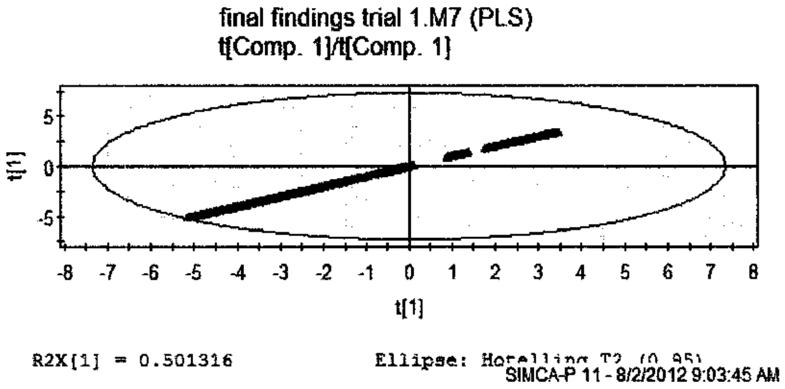
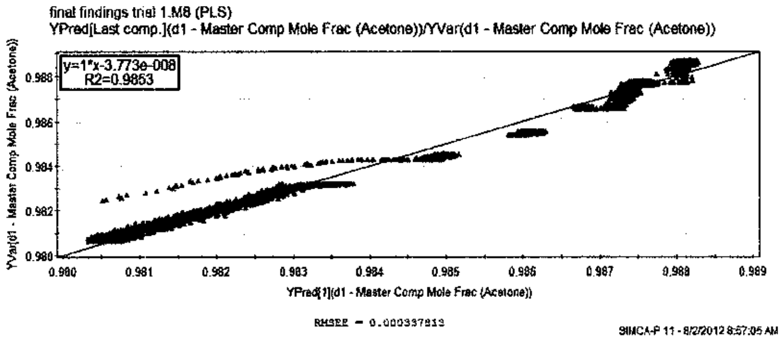
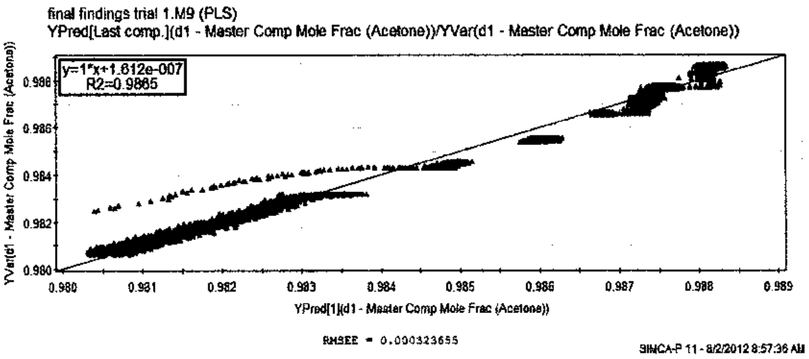
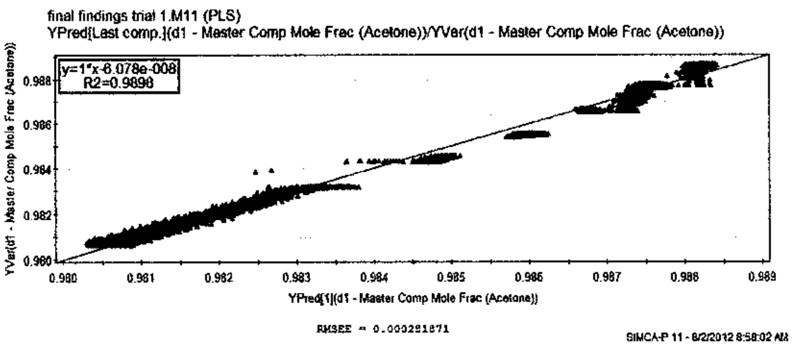
Table 4.4: Selected input variables for new data set.

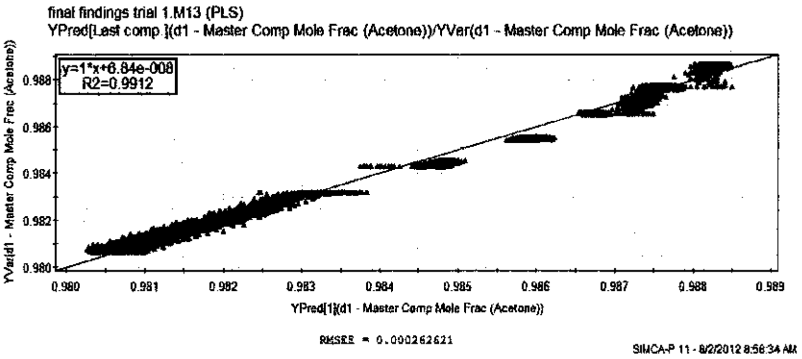
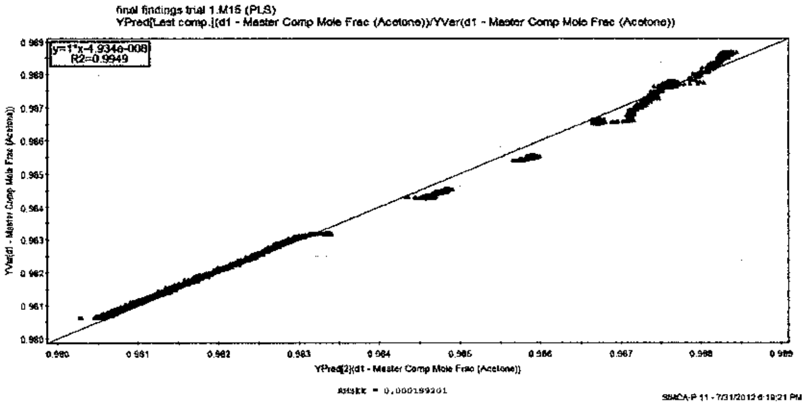
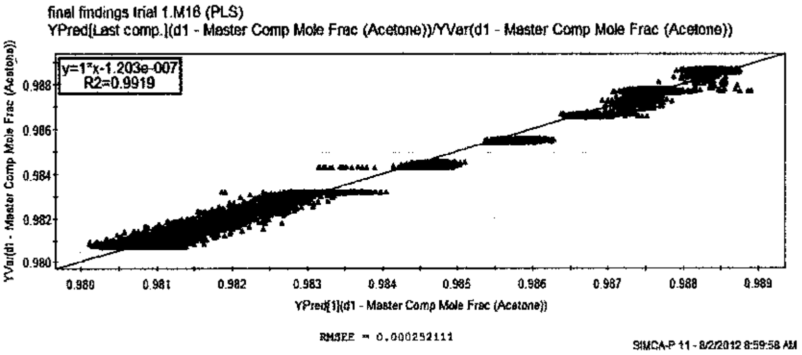
Data set	Selected input variable
Original	Tray temperature 7 until 15
t-1	Tray temperature 12 until 15
t-2	Tray temperature 14 and 15
t-3	Tray temperature 14 and 15
t-4	Tray temperature 15

By using the selected input variables as mentioned above, a new data set is prepared and simulated as a new project. The coefficient plot and Y-observed versus Y-prediction plot is observed in the new data set. In the y-observed and y-prediction plot, it displays the observed versus predicted values of the selected variables. Then, the R^2 of the regression line on the plot indicates the fit. If R^2 value near to 1 shows that the regression line is very fit. The RMSEE on the plot is Root Mean Square Error of the fit for observation in the model where for a good prediction plot, the RMSEE value should be very low. Moreover, if the points on the plot scattered from the regression line, indicated those points are outliers that need to be removed. The input variables in the new data set are reduced until the optimum value of R^2 in the y-observed versus y-predicted plot is reached. The steps of reduction of input variables in the new data set are tabulated as below:

Table 4.5: Steps of input variables reduction in the data.

Step	Description	Plot
1	Outliers are removed in the model.	New data set without removing outliers.  After the outliers are removed. 

Step	Description	Plot
1		Hotelling T^2 figure 
2	Tray temperature 7 is removed	 $R^2 = 0.9853$
3	Tray temperature 8 is removed	 $R^2 = 0.9865$
4	Tray temperature 9 is removed	 $R^2 = 0.9898$

Step	Description	Plot
5	Tray temperature 10 is removed	 $R^2 = 0.9912$
6	Tray temperature 11 is removed	 $R^2 = 0.9949$
7	Tray temperature 12 is removed	 $R^2 = 0.9919$

Based on the table above, after the outliers are removed in the data, the R^2 value in the plot increased. Besides that, when the input variable reduced from tray temperature 7 to 11, the R^2 value of the plot increased. But the reduction of input variable tray temperature 12 shows a low value for R^2 which is 0.9919. So the particular input variable is remained in the model. The other variables are tested based on trial and error method to reduce the number of input variables.

After reducing the input variables in the data, the balance final input variables are shown as below:

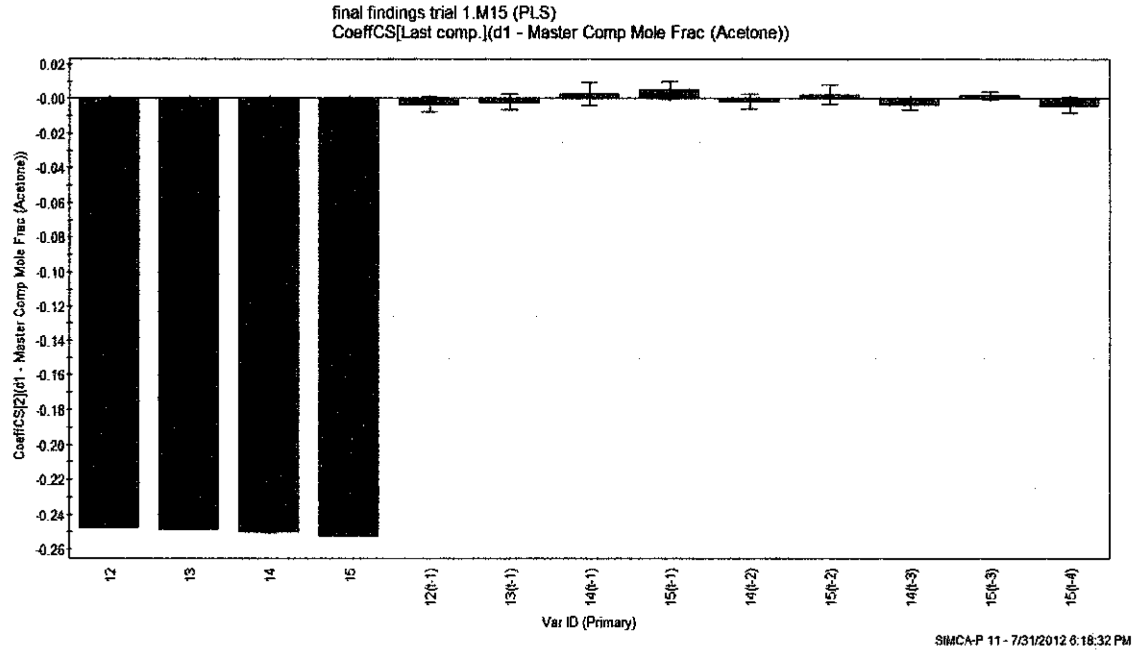


Figure 4.13: Final coefficient plot.

Other than the coefficient plot, VIP plot which stands for variable importance plot explains the correlation of X and Y data. Variable importance plot summarize the importance of the variables in the model. By using this plot, the less important variable can be removed to increase the performance of the model.

The VIP plot for this case is shown as below:

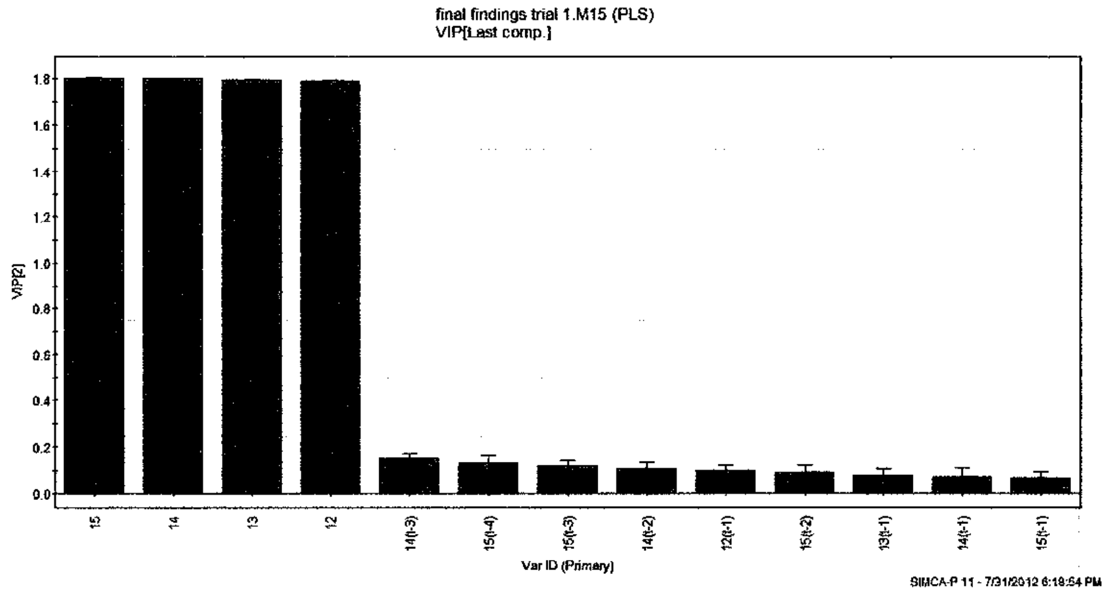


Figure 4.14: Variable importance plot.

About 13 input variables are correlated in the regression coefficient above. The regression model for the coefficient plot above is as following:

Table 4.6: Coefficients of regression model.

Input variable (X)	Description	Regression coefficient
-	Constant	3.54E+02
T_{12}(t)	Tray temperature 12 (original)	-2.47E-01
T_{13}(t)	Tray temperature 13 (original)	-2.48E-01
T_{14}(t)	Tray temperature 14 (original)	-2.50E-01
T_{15}(t)	Tray temperature 15(original)	-2.52E-01
ΔT_{12}(t-1)	Tray temperature 12 (t-1)	-3.73E-03
ΔT_{14}(t-2)	Tray temperature 14 (t-2)	-1.05E-03
ΔT_{15}(t-2)	Tray temperature 15 (t-2)	3.11E-03
ΔT_{14}(t-3)	Tray temperature 14 (t-3)	-2.40E-03
ΔT_{15}(t-3)	Tray temperature 15 (t-3)	3.57E-03
ΔT_{15}(t-4)	Tray temperature 15 (t-4)	-1.99E-03

For this PLS model, the regression model is written as:

Y = Y_{avg} + XB, where B is the regression coefficient.

$$\boxed{\begin{aligned} x_D(t) = 354225 - [247.13\times10^{-3}T_{12}(t)] - [24837\times10^{-3}T_{13}(t)] - [25008\times10^{-3}T_{14}(t)] - \\ [25242\times10^{-3}T_{15}(t)] - [3.73\times10^{-3}\Delta T_{12}(t-1)] - [1.05\times10^{-3}\Delta T_{14}(t-2)] + \\ [3.11\times10^{-3}\Delta T_{15}(t-2)] - [2.40\times10^{-3}\Delta T_{14}(t-3)] + [3.57\times10^{-3}\Delta T_{15}(t-3)] - \\ [1.99\times10^{-3}\Delta T_{15}(t-4)] \end{aligned}}$$

Where Y= X_D (*t*) = mole fraction of acetone in the top product at *t*.

Using this regression model, the output variable which is mole fraction of acetone in the top product is predicted. The efficiency of the model can be seen in the y-observed versus y-predicted plot.

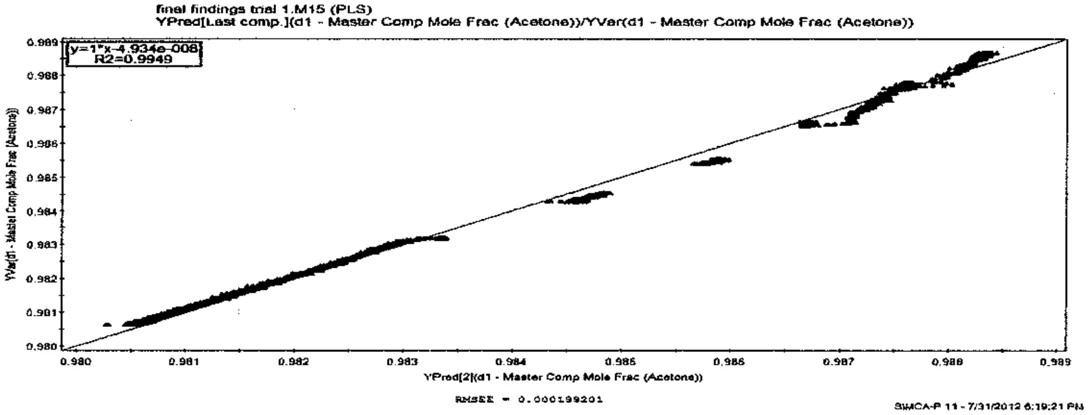


Figure 4.15: Y-observed versus Y-predicted plot.

According to the plot above, the R^2 value is 0.9949 which shows the fit is good enough. In addition, the RMSEE value is 0.000199201, shows that the error is less in this model. Besides that, the regression line is straight about 45 ° and the data points are not scattered far from the regression line. This shows that, there are very less outliers in this model which is good for prediction.

4.5 Evaluation and comparison of soft sensor performance

In order to evaluate the time difference of process variables method against the static conventional method, the original data without time difference approach is simulated to observe the performance of it. The simulation observation of the original data using PLS modeling technique is shown as below:

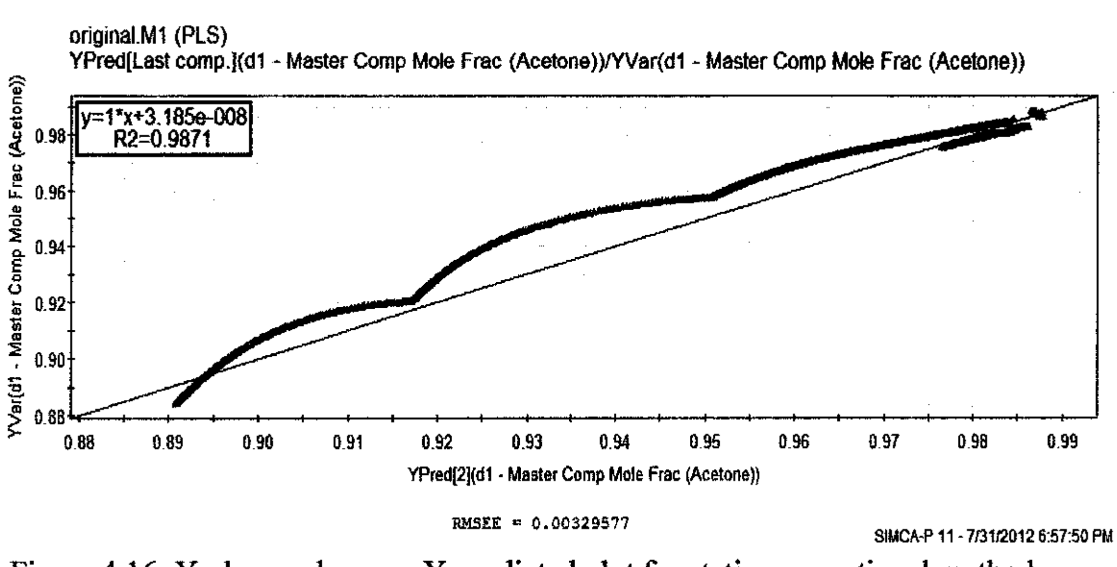


Figure 4.16: Y-observed versus Y-predicted plot for static conventional method.

Based on the plot above, the R^2 value is 0.9871 and the RMSEE value is 0.00329577. This shows that the fitting is good but the data points are scattered far away from the regression line. The comparison of both time difference approach model and static conventional method model is tabulated as below:

Table 4.7: Comparison of conventional method and time difference method model.

Model	R^2	RMSEE	Observation of data point
Static conventional	0.9871	0.0033	Scattered far away from the regression line
Time difference	0.9949	0.0002	Scattered near the regression line

As overall, time difference of process variable approach model gives higher prediction compared to conventional model.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

In this project, data-driven model with time difference of process variables is developed using SIMCA-P software. This model is developed using the PLS modeling technique which is built-in the software. The data to develop this model is generated using the binary distillation column simulation. Moreover, the simulation used to generate the data is in dynamic mode so that the performance of the inferential model can be evaluated. After data pre-processing, the input variables are correlated to reduce the number of variables. By doing this, the components are correlated and the model is developed.

In order to evaluate the time difference of process variable against the static conventional method, the R^2 value of both methods is observed. Based on the observation, the prediction of the inferential model for time difference approach is higher compared to the static conventional method. This is mainly because, the performance of time difference approach is good in non-steady state condition. Meanwhile, the static conventional model performed better in steady-state condition only. As overall, the objective of this project is achieved.

5.2 Recommendation

In order to evaluate the efficiency of soft sensor model, this soft sensor model should be tested with new data set generated by the simulation. If the efficiency is good, this model can be applied to real industrial data. Moreover, apart from PLS modeling technique, this model should be developed using other modeling technique such as SVR (support vector machine). By using this modeling technique, the current model can be evaluated against the model which developed using SVR technique. Besides that, another soft sensor model can be developed using different case study such as reactor unit.

REFERENCES

Abdi, H. 2010, "Partial least squares regression and projection on latent structure regression (PLS Regression)", *Wiley Interdisciplinary Reviews: Computational Statistics,* vol. 2, no. 1, pp. 97-106.

Cheng, C. & Chiu, M. 2004, "A new data-based methodology for nonlinear process modeling", *Chemical Engineering Science,* vol. 59, no. 13, pp. 2801-2810.

Dechaumphai,P N. 2011, "Numerical Methods in Engineering" , UK: Alpha Science International Ltd, pp. 18-32.

Facchin,S., Trierwielder,J.O. & Conz,V. 2005, "Soft Sensor Design: A new approach for variable selection", *Chemical Engineering,* pp. 1-10.

Fortuna, L., Graziani, S., Rizzo, A. & Xibilia, M.G. 2007, "Soft Sensors for Monitoring And Control of Industrial Processes", *Advances in Industrial Control,* pp. 2-30.

Jiangfeng Chen & Baozong Yuan 2010, "Spectral Partial Least Squares Regression", *Signal Processing (ICSP), 2010 IEEE 10th International Conference on,* pp. 1351.

Joe Qin, S. 1998, "Recursive PLS algorithms for adaptive data modeling", *Computers & Chemical Engineering,* vol. 22, no. 4–5, pp. 503-514.

Kadlec, P. & Gabrys, B. 2008, "Adaptive Local Learning Soft Sensor for Inferential Control Support", *Computational Intelligence for Modelling Control & Automation, 2008 International Conference on,* pp. 243.

Kadlec, P., Gabrys, B. & Strandt, S. 2009, "Data-driven Soft Sensors in the process industry", *Computers & Chemical Engineering,* vol. 33, no. 4, pp. 795-814.

Kaneko, H., Arakawa, M. & Funatsu, K. 2011, "Applicability domains and accuracy of prediction of soft sensor models", *AIChE Journal,* vol. 57, no. 6, pp. 1506-1513.

Kaneko, H. & Funatsu, K. 2011, “Classification of the Degradation of Soft Sensor Models and Discussion on Adaptive Models”, *Chemical System Engineering,* pp. 1-5.

Kaneko,H. & Funatsu,K. 2011, “Development of Soft Sensor Models Based on Time Difference of Process Variables with Accounting for Nonlinear Relationship”, *Chemical System Engineering,* vol.50, pp.10643-10651.

Kaneko, H., Arakawa, M. & Funatsu, K. 2011, "Novel soft sensor method for detecting completion of transition in industrial polymer processes", *Computers & Chemical Engineering,* vol. 35, no. 6, pp. 1135-1142.

Kaneko, H., Arakawa, M. & Funatsu, K. 2009, "Development of a new soft sensor method using independent component analysis and partial least squares", *AIChE Journal,* vol. 55, no. 1, pp. 87-98.

Kaneko, H. & Funatsu, K. 2012, "Development of high predictive soft sensor method and the application to industrial polymer processes", *Asia-Pacific Journal of Chemical Engineering,* vol. 7, pp. S39-S47.

Kaneko, H. & Funatsu, K. 2011, *Improvement and Estimation of Prediction Accuracy of Soft Sensor Models Based on Time Difference*, Springer Berlin / Heidelberg.

Kaneko, H. & Funatsu, K. 2011, "Maintenance-free soft sensor models with time difference of process variables", *Chemometrics and Intelligent Laboratory Systems,* vol. 107, no. 2, pp. 312-317.

Kaneko, H. & Funatsu, K. 2011, "A soft sensor method based on values predicted from multiple intervals of time difference for improvement and estimation of prediction accuracy", *Chemometrics and Intelligent Laboratory Systems,* vol. 109, no. 2, pp. 197-206.

L.Eriksson, E. N.-W. 2001, “Multi- and Megavariate Data Analysis”, *Sweden: Umetrics,* pp. 18-34.

Lin, B., Recke, B., Knudsen, J.K.H. & Jørgensen, S.B. 2007, "A systematic approach for soft sensor development", *Computers & Chemical Engineering,* vol. 31, no. 5–6, pp. 419-425.

Liu, J. 2007, "On-line soft sensor for polyethylene process with multiple production grades", *Control Engineering Practice,* vol. 15, no. 7, pp. 769-778.

Maitra,S. & Yan,J. 2008, "Principle Component Analysis and Partial Least Squares: Two Dimension Reduction Techniques for regression", *Casualty Acturial Society Discussion Paper Program*, pp.79-90.

Okada, T., Kaneko,H. & Funatsu,K. 2011, "Development of a Model Selection Method Based on Reliability of a Soft Sensor Model", *TIChe International Conforence*, pp. 1-5.

Park, S. & Han, C. 2000, "A nonlinear soft sensor based on multivariate smoothing procedure for quality estimation in distillation columns", *Computers & Chemical Engineering,* vol. 24, no. 2–7, pp. 871-877.

Phatak, A. & De Jong, S. 1997, "The geometry of partial least squares", *Journal of Chemometrics,* vol. 11, no. 4, pp. 311-338.

Rosipal, R. & Kr\amer, N. 2006, "Overview and recent advances in partial least squares", *Proceedings of the 2005 international conference on Subspace, Latent Structure and Feature Selection*Springer-Verlag, Berlin, Heidelberg, pp. 34.

Sharmin, R., Sundararaj, U., Shah, S., Vande Griend, L. & Sun, Y. 2006, "Inferential sensors for estimation of polymer quality parameters: Industrial application of a PLS-based soft sensor for a LDPE plant", *Chemical Engineering Science,* vol. 61, no. 19, pp. 6372-6384.

Wold, S., Sjöström, M. & Eriksson, L. 2001, "PLS-regression: a basic tool of chemometrics", *Chemometrics and Intelligent Laboratory Systems,* vol. 58, no. 2, pp. 109-130.

Zamprogna, E., Barolo, M. & Seborg, D.E. 2004, "Estimating product composition profiles in batch distillation via partial least squares regression", *Control Engineering Practice,* vol. 12, no. 7, pp. 917-929.

Zamprogna,E., Barolo,M. & Seborg,D.E. 2002, “Development of a Soft Sensor for a batch distillation column using linear and nonlinear PLS regression techniques”, *Chemical Engineering*, pp.45-51.

APPENDICES

Appendix 1

a) MATLAB Coding and command window with results:

```
n = input('\nEnter number of data:');
for irow = 1:n
   x(irow) = input('\nEnter value of x: ');
   y(irow) = input('Enter value of y: ');
end

sumx = 0.0;
sumy = 0.0;
sumx2 = 0.0;
sumxy = 0.0;
for i = 1:n
   sumx = sumx + x(i);
   sumy = sumy + y(i);
   sumx2 = sumx2 + x(i)*x(i);
   sumxy = sumxy + x(i)*y(i);
end
% SOLVE FOR COEEFICIENTS:
det = n*sumx2 - sumx*sumx;
A0 = (sumy*sumx2 - sumxy*sumx)/det;
A1 = (n*sumxy - sumx*sumy)/det;
fprintf('\COEFFICIENT A0 = %14.6e',A0)
fprintf('\nCOEFFICIENT A1 = %14.6e',A1)
```

```
Enter number of data:6

Enter value of x: 10
Enter value of y: 2.2

Enter value of x: 15
Enter value of y: 4.6

Enter value of x: 20
Enter value of y: 4.2

Enter value of x: 25
Enter value of y: 7.0

Enter value of x: 30
Enter value of y: 6.6

Enter value of x: 35
Enter value of y: 9.2

COEFFICIENT A0 =  1.904762e-003
COEFFICIENT A1 =  2.502857e-001
```

Appendix 2

a) Input Data (file name : bros.dat) = 62, 1, 14, 103, 129, 218 & 2211

```
fid=fopen('bros.dat','r');
 n=fscanf(fid,'%f',1);
 k=fscanf(fid,'%f',1);
 x=fscanf(fid,'%f',[3 6]);
 x=x';
 x1=x(:,1:2);
 y=x(:,3);
 b=zeros(k+1,1);
 a=zeros(k+1,k+1);
 for i=1:n
for ir=1:k+1
if ir==1
fr=1.;
end
if ir>1
fr=x1(i,ir-1);
end
for ic = 1:k+1
if ic==1
fc=1.;
end
if ic>1
fc=x1(i,ic-1);
end
a(ir,ic)=a(ir,ic)+fr*fc;
end
b(ir)=b(ir)+fr*y(i);
end
 end
kp1=k+1;
xx=gauss(kp1, a, b);
 fprintf('\ncoefficient of fitted function are:')
 for i = 1:k+1
   im1=i-1;
   fprintf('\n A(%1d) = %13.7e',im1,xx(i));
 end
```

```
coefficient of fitted function are:
 A(0) = 1.0000000e+000
 A(1) = 2.0000000e+000
 A(2) = 3.0000000e+000
```

Appendix 3

```
load moore
y = moore(:,6);                                                    % Response
X0 = moore(:,1:5);                                                % Original predictors
X1 = X0+10*randn(size(X0));                                        % Correlated predictors
X = [X0,X1];[XL,yl,XS,YS,beta,PCTVAR] = plsregress(X,y,10);

plot(1:10,cumsum(100*PCTVAR(2,:)),'-bo');
xlabel('Number of PLS components');
ylabel('Percent Variance Explained in y');
[XL,yl,XS,YS,beta,PCTVAR,MSE,stats] = plsregress(X,y,6);
yfit = [ones(size(X,1),1) X]*beta;

plot(y,yfit,'o')
TSS = sum((y-mean(y)).^2);
RSS = sum((y-yfit).^2);
Rsquared = 1 - RSS/TSS
plot(1:10,stats.W,'o-');
legend({'c1','c2','c3','c4','c5','c6'},'Location','NW')
xlabel('Predictor');
ylabel('Weight');
[axes,h1,h2] = plotyy(0:6,MSE(1,:),0:6,MSE(2,:));
set(h1,'Marker','o')
set(h2,'Marker','o')
legend('MSE Predictors','MSE Response')
xlabel('Number of Components')
```