THE APPLICATION OF MULTIVARIATE STATISTICAL ANALYSIS AND OPTIMIZATION TO BATCH PROCESSES

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN THE FACULTY OF ENGINEERING AND PHYSICAL SCIENCES

2015

Lipeng Yan

School of Electrical and Electronic Engineering

Contents

List of Tables

List of Figures

Abstract

Multivariate statistical process control (MSPC) techniques play an important role in industrial batch process monitoring and control. This research illustrates the capabilities and limitations of existing MSPC technologies, with a particular focus on partial least squares (PLS).

In modern industry, batch processes often operate over relatively large spaces, with many chemical and physical systems displaying nonlinear performance. However, the linear PLS model cannot predict nonlinear systems, and hence non-linear extensions to PLS may be required. The nonlinear PLS model can be divided into Type I and Type II nonlinear PLS models. In the Type I Nonlinear PLS method, the observed variables are appended with nonlinear transformations. In contrast to the Type I nonlinear PLS method, the Type II nonlinear PLS method assumes a nonlinear relationship within the latent variable structure of the model. Type I and Type II nonlinear multi-way PLS (MPLS) models were applied to predict the endpoint value of the product in a benchmark simulation of a penicillin batch fermentation process. By analysing and comparing linear MPLS, and Type I and Type II nonlinear MPLS models, the advantages and limitations of these methods were identified and summarized. Due to the limitations of Type I and II nonlinear PLS models, in this study, Neural Network PLS (NNPLS) was proposed and applied to predict the final product quality in the batch process. The application of the NNPLS method is presented with comparison to the linear PLS method, and to the Type I and Type II nonlinear PLS methods. Multi-way NNPLS was found to produce the most accurate results, having the added advantage that no a-priori information regarding the order of the dynamics was required. The NNPLS model was also able to identify nonlinear system dynamics in the batch process.

Finally, NNPLS was applied to build the controller and the NNPLS method was combined with the endpoint control algorithm. The proposed controller was able to be used to keep the endpoint value of penicillin and biomass concentration at a set-point.

Declaration

No portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright Statement

i. The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the "Copyright") and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.

ii. Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made only in accordance with the Copyright Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.

iii. The ownership of certain Copyright, patents, designs, trademarks and other intellectual property (the "Intellectual Property") and any reproductions of copyright works in the thesis, for example graphs and tables ("Reproductions"), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.

iv. Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=487), in any relevant Thesis restriction declarations deposited in the University Library, The University Library's regulations (see http://www.manchester.ac.uk/library/aboutus/regulations) and in The University's policy on Presentation of Theses.

Acknowledgements

I would like to offer my deepest gratitude to my supervisor Prof. Barry Lennox for his supervision, advice and guidance during my PhD study. Barry led me to the area of multivariate statistical analysis and batch process control. His suggestions always help me to figure out the solution to the difficulties in my PHD project. I attribute this thesis to his encouragement, effort and patience: without him this thesis would not have been completed.

Thanks must go to all control system centre colleagues, especially Dr Jianan Wang, Yin Xin, Linan Tao, Wang Shuai, Zhang Qichun, Xiaomao and Wang Yan who company with me in PHD life. Also many thanks go to my friends in UK and China for their encouragement and support.

My thanks also go to Dr Jiang Yifan who kindly proofread the main chapter of this thesis.

Last but not least, I would like to thank my dear parents who have given me unfailingly support and prayers. Their selfless contributions cannot be expressed in any words.

Abbreviation

ANN	Artificial Neural Network
BPNN	Back Propagation Neural Network
CMR	Conditional Mean Replacement
CUSUM	Cumulative Sum
DPCA	Dynamic Principal Component analysis
EWMA	Exponentially Weighted Moving Average
IMB-PLS	Interconnected Multi-Block Partial Least Squares
LTI	Linear Time Invariant
LVs	Latent Variables
MBPLS	Multi-Block Partial Least Squares
MLR	Multiple Linear Regression
MPC	Model Predictive Control
MPCA	Multi-way Principal Component analysis
MPLS	Multi-way Partial Least Squares
MSPC	Multivariate Statistical Process Control
MSQC	Multivariate Statistical Quality Control
MVTs	Manipulated Variable Trajectories
NIPALS	Non-linear Iterative Partial Least Squares
NIR	Near Infrared
NNPLS	Neural Network Partial Least Squares
OLS	Ordinary Least Squares
RBF–PLS	Radial Basis Functions-Partial Least Squares
PCA	Principal Component analysis
PCR	Principal Component Regression
PCs	Principal Components
PLS	Partial Least Square or Projection to Latent structure
PRBS	Pseudo-Random Binary Signals
PRESS	The Prediction Error Sum of Squares
QP	Quadratic Programming
SCP	Single Component Projection

SISO	Simple-Input-Simple-output
SPC	Statistical Process Control
SSE	The Sum of Square Errors
TQC	Total Quality Control

Notation

C_L	Dissolved oxygen concentration in the broth
CO_2	Carbon dioxide concentration
E	Residual matrix for PCA and PLS (predictor variables) model
F	Residual matrix for PLS prediction of response matrix
f_g	Flow rate of oxygen
H	Hydrogen ion concentration for pH ($[H^+]$)
I	The number of batches
J	The number of measured observations in a complete batch
K	The number of measured variables
P	Penicillin concentration
P_W	Agitator power input P_W
P_1^T	Loading matrices for X_1^T
P_2^T	Loading matrices for X_2^T
p_a	a^{th} loading vector for PCA model / PLS (predictor variables) model
Q_1	Diagonal weighting matrices for the variables in y
Q_2	Diagonal weighting matrices for the variables in Δt
q_a	a^{th} loading vector for PLS (response variables)
S	Substrate concentration
T^2	Hotelling's statistic
t_a	a^{th} score vector for PCA model / PLS (predictor variables) model
$\hat{t}_{present}$	Projection of the X matrix onto the latent variable space
Δt	The value for $t - \hat{t}_{present}$

Δt_{max}	The maximum values for Δt
Δt_{min}	The minimum values for Δt
u_a	a^{th} score vector for PLS (response variables)
u_{MV}^T	Manipulated variables
X^*	The pseudo-inverse of the matrix X
X^T	Data matrix
X_1^T	Known trajectories
X_2^T	Future unknown trajectories
$X_{on-line}^T$	Online process variables
$X_{off-line}^T$	Offline measurements
X_m	Monomer conversion
X_{blo}	Biomass concentration
u	Specific growth rate
u_x	The maximum specific growth rate
u_{pp}	The specific penicillin production rate
θ_i	Decision Points
Δt_{max}	The maximum values for Δt

Publication

Yan, L. and Lennox, B. "The Application of Nonlinear Partial Least Square to Batch Processes", *10th IFAC International Symposium on Dynamics and Control of Process Systems*, Mumbai, India, 2013.

Chapter 1

Introduction

1.1 Background

In industrial processes, to ensure their safe and efficient operation and to improve or maintain product quality, these processes need to be continuously monitored throughout their operation. The monitoring of process systems has been studied extensively over the last few years.

In an attempt to monitor industrial processes, the chemical industry has seen a rapid increase in the number of sensors that have been made commercially available. Unfortunately, because of the large amounts of data available and the highly correlated nature of these measurements, it can be difficult to interpret the data once it has been collected. To help with the interpretation of large quantities of process measurements, Statistical Process Control (SPC) is proposed and applied.

SPC is an approach for process monitoring based on a mathematical statistics method (Wetherill & Brown, 1991), and is used for monitoring and controlling a process. The SPC method is an effective technique for maintaining product quality at a required level and ensuring product consistency. SPC methods have been applied in process industries with great success during past decades. Nevertheless, when the number of the required process variables increases, the number of monitoring charts also needs to be augmented. When the number of required charts becomes too large, traditional SPC methods are often unsuitable.

In industrial process data, the recorded process variables are huge; these variables have collinearity and are highly correlated. In practice, the process variables are recorded irregularly, such as missing value numbers or, a number of the values may be corrupted by process and measurement noise. In these cases, traditional SPC methods are not able to adapt to modern industrial processes. MSPC is as an alternative approach to traditional SPC in the area of process monitoring, and has

overcome the weakness of SPC. MSPC method can reduce some dimensions in the process data, meaning that required univariate control charts are obviously reduced. The MSPC method collects all process data, including past data, and because MSPC explicitly considers the multivariate nature of the data, it identifies the correlations that exist. MSPC is thus suitable for dealing with large and highly correlated data sets. The primary objective of MSPC is to maintain product quality in a desired product specification and to control a process in a desired state. By controlling and monitoring a number of key quality variables of a process, product quality can be as close as possible to the desired value.

MSPC covers a wide range of techniques. In these methods, MSPC is based on the statistic projection techniques of Principal Component Analysis (PCA) and Partial Least Square or Projection to Latent structure (PLS). They are used to analyse process data and to develop predictive models in support of process monitoring and control in real industrial processes. When PLS is applied to batch processes, a technique referred to as Multi-way PLS (MPLS) is frequently used. This technique analyses process behaviour relative to the mean trajectories of the process variables. In doing so, a major nonlinearity in the data is removed.

A major limitation of linear PLS is that industrial processes are always nonlinear to some extent. This is not always a problem as many processes only operate around limited operating regions, where linear PLS techniques tend to provide acceptable accuracy. However, batch processes often operate over relatively large spaces, with many chemical and physical systems displaying nonlinear performance; hence nonlinear extensions to PLS may be required. A number of different methods have therefore been proposed to provide a nonlinear PLS algorithm.

The nonlinear PLS model can be divided into Type I and Type II nonlinear PLS models (Wold, 1989).

In the Type I Nonlinear PLS method, the observed variables are appended with nonlinear transformations. Following this, traditional linear PLS is then applied. In Type I methods, the inputs into the PLS model are specified to be cross and squared terms of the input variables. On the other hand, for Type II methods, nonlinear functions are implemented in the PLS model’s inner structure. In contrast to the

Type I nonlinear PLS method, the Type II nonlinear PLS method assumes a nonlinear relationship within the latent variable structure of the model. Type I and II non-linear structures are integrated within MPLS models to enable them to more accurately approximate nonlinear batch processes. In reality, the exact order of the polynomial of any nonlinear relationship will not be known a-priori; this means that Type I and II nonlinear PLS models cannot determine the particular polynomial expansion to match the non-linearity inner relation of the process. The accuracy of the prediction is hence reduced significantly.

Neural Network PLS (NNPLS) is an alternative method to Linear PLS. For increased functionality, the use of a neural network is proposed in the inner structure of the NNPLS model. The advantage of the application of the neural network in the inner regressors is based on their nonlinear approximation property. In this study, the NNPLS model is used for process monitoring and control.

In recent years, batch processes have gained ever increasing importance in manufacturing industries. In particular, batch processing is frequently used in the manufacture of low volume, high value products, such as pharmaceuticals or specialty chemicals. Unfortunately, batch processing encounters many challenges in continuous production; for instance, there are rarely steady state conditions; process dynamics are typically time-varying and non-linear; and quality measurements are often only available at the end of the batch.

Quality control of batch processes is usually implemented by regulating several process variables, such as temperature and pH. To ensure consistent endpoint product quality, these variables are expected to maintain their set-point value. When variation occurs in the raw material properties, the operation process cannot produce a consistent product. Consequently a number of advanced control methods have been applied to improve product consistency.

Cerrillo and MacGregor (2003) proposed a strategy for controlling end-point quality properties. Endpoint control attempts to optimize the operating conditions during the whole batch to ensure that endpoint product quality satisfies the requirements. This endpoint controller was successfully applied to regulate a simulated batch process.

1.2 Contributions of This Study

The work described in this research focuses on multivariate statistical analysis of an industrial fed-batch fermentation process, used for the production of penicillin and quality control in the industrial polymerization batch process.

Research contributions include:

1) PLS has been successfully applied in the modelling, estimation and control of batch processes. However, the nonlinear nature of many real, complex chemical systems means that traditional linear PLS is not always suitable. Therefore, in this thesis, the use of a nonlinear multi-way PLS is proposed to address the issues of non-linearity in batch processes. Type I and Type II nonlinear multi-way PLS models are used to predict the endpoint value of the product. In the algorithm for the Type II nonlinear PLS model, the inner relation is usually considered to be a quadratic polynomial (2nd order) or 3rd order polynomial. A limitation of this approach is that by choosing a 2nd order polynomial, the type of relationship that can be modelled is restrictive. Therefore, higher order terms for the Type II nonlinear PLS model are considered and applied in this study. By analysing and comparing linear multi-way PLS and Type I and Type II nonlinear multi-way PLS models, the advantages and limitations of these methods are identified and summarized.

As Type I and Type II nonlinear multi-way PLS models have certain limitations, multi-way NNPLS is proposed and applied to predict the quality of the final product in the batch process. Its performance is compared with the performance achieved using the Type II nonlinear multi-way PLS model. The benefits of the multi-way NNPLS is discussed and summarized.

2) Quality control of batch processes is usually implemented by regulating several process variables, such as temperature and pH. Notwithstanding that the process variables are well maintained, the quality of the final product cannot be guaranteed, due to the effects of disturbances. To address this, the endpoint controller based on neural network PLS is proposed and applied to control the endpoint value at a set-point. The Endpoint controller based on NNPLS can be applied to track a changing set-point; the results showed the performance of the NNPLS controller is very accurate. And when some process disturbance and some noises are considered, the

Endpoint controller based on NNPLS has the ability to reject these disturbances and noises.

1.3 Structure of the Thesis

This thesis consists of 7 chapters. Following this introduction, Chapter 2 presents a literature review of the application of multivariate statistical analysis methods, and summarizes the previous work on MSPC methods and applications. Chapter 3 describes some of the basic algorithms of MSPC techniques such as partial least squares (PLS) and also discusses several of the extensions of PLS, including MPLS, nonlinear PLS and neural network PLS (NNPLS).

In Chapter 4, the limitations of linear PLS are discussed and summarised; PLS is then used to predict a linear and nonlinear system. To overcome this deficiency, several nonlinear extensions are proposed to enable it to better handle nonlinear systems. Finally, to illustrate the capabilities of the NNPLS method, NNPLS and the Type II nonlinear PLS model are applied to predict the same testing system. Some conclusions are then provided based on analysis of the results.

In Chapter 5, NNPLS is used to model a benchmark simulation of a penicillin batch fermentation process. The fermentation process is first introduced. The endpoint measurement is used because in most fermentation processes, quality measurements such as penicillin concentration, will only be available at the end of a batch. For this reason, this chapter focuses on the endpoints of the products (Biomass and Penicillin). By analysing and comparing linear multi-way PLS, Neural network multi-way PLS, and Type I and Type II nonlinear multi-way PLS models, the advantages and limitations of these methods are identified and summarized.

In Chapter 6, NNPLS is applied to control Biomass and Penicillin a penicillin batch fermentation process. The basic End-point Control Algorithm is described and discussed. A novel endpoint controller based on NNPLS is proposed, and its performances are compared against other controllers. The benefits of this controller are summarized.

Finally, Chapter 7 provides the conclusions of this study. In addition, suggestions for further work are also included in this chapter.

Chapter 2

Literature Review

This chapter presents an extended overview of multivariate statistical process control (MSPC) methods and techniques. The advantages and disadvantages of some MSPC methods are discussed, specifically those relating to Principal Component Analysis (PCA) and Partial Least Squares (PLS).

The chapter is divided into the following sections:

2.1) presents a background to batch process and process monitoring;

2.2) provides an overview of Statistical Process Control;

2.3) introduces the use of control charts;

2.4) describes MSPC techniques, such as PCA and PLS;

2.5) discusses the use of Multivariate Statistical Projection methods, and summarizes these methods;

2.6) introduces nonlinear PLS and Neural Network PLS; and

2.7) provides a summary of this chapter.

2.1 Background to Batch Processes and Process Monitoring

Batch processes are widely used in industry as they outperform continuous operations in the manufacturing of certain chemicals and materials (Korovessi & Linninger, 2006). The selection of a batch or continuous operation is based on many

factors (Yucai, 2001). A brief comparison of Batch and Continuous Operations is presented in Table 2.1. Batch processing is frequently used in the manufacture of low volume, high value products, such as pharmaceuticals or specialty chemicals. The materials are processed over a finite period of time, where the operational conditions are typically specified to follow a pre-determined recipe. To ensure safe and efficient operation of these processes and to improve or maintain product quality, it is important that these processes are continuously monitored during operation. However, as a result of disturbances to the process, such as changes in the initial conditions of the batch and the frequent absence of on-line quality measurements, this can be challenging (Wetherill & Brown, 1991; Martin & Morris, 1996; Yucai, 2001; Korovessi & Linninger, 2006; Yao & Gao, 2009).

Table 2.1 Reasons for Batch Operations and Continuous Operations

Reason for Batch Operation	Reason for Continuous Operations
Small volume of production (production typically<500,000kg/yr)	Large volume of production
Variability in production rate	Steady production rate
Reuse of equipment (shared equipment)	Dedicated-use equipment (single product use)
Multi-product operation	Single-product operation
Process variables subject to adjustment (uncertainties in the reactivity or potency of raw materials)	Invariable process condition (minor uncertainties in the reactivity or process are sufficiently robust)
Many isolation steps	Lot integrity arbitrary or not required
Lot integrity required	

The monitoring of process systems has been studied extensively in previous years (e.g. Macgregor & Kouriti, 1995; Lennox et al., 2000). The most important aspect of process monitoring is the detection of any abnormal events in the process operation, and the identification of any effects of these abnormalities, such as changes in the product quality and quantity (Jackson, 1991; MacGregor et al., 1991; Iserman, 1997; Camacho et al., 2008). Both industry and academia hope that system performance

and product quality can be increased, and the safety of the operation can also be improved through process monitoring.

A number of approaches have been applied to detect such process abnormalities, such as signal-, knowledge- and model-based techniques (Willsky, 1976; Iserman, 1997). In signal-based techniques, the measured signals are directly analyzed. When the signal exceeds signal tolerances, process abnormalities can be found (Iserman, 1993). In knowledge-based techniques, detection is based upon comparing the difference between the process measured variable and observed variable value, with the heuristic knowledge value and analytical knowledge value (Freyermuth, 1991; Venkatasubramanian et al., 2002; MacGregor & Cinar, 2012). In model-based techniques however, process analysis is undertaken using mathematical process models together with parameter estimation, state estimation and parity equation methods. The detection of abnormal operations is based on the analysis of parameters, state variables and residuals (Iserman & Balle, 1997). According to Frank et al. (2000), the model-based technique is an effective method for detecting process abnormalities, therefore the model-based technique will be focused upon in this thesis.

The most basic and common application of the model-based method is based on the use of linear time invariant (LTI) models. Early work on the use of LTI models examined fault detection in state-space model applications (Beard, 1971; Jones, 1973). In recent years, decoupling techniques have been applied; this approach can make more robust model uncertainties and disturbances, whilst making it easier and more effective when detecting process faults (Frank et al., 2000).

In an attempt to monitor industrial processes, the chemical industry has seen a rapid increase in the number of sensors that have been made commercially available. Unfortunately, due to the large amounts of data available and the highly correlated nature of these measurements, it can be difficult to interpret the data once collected. To help interpret large quantities of process measurements, many researchers have successfully applied data analysis tools, such as those available within the field of Statistical Process Control (SPC) (Martin et al., 1996).

2.2 Statistical Process Control

In the production process, some variables may be subject to some fluctuations caused by, for instance, machine, method, material and environment. These fluctuations can be divided into two types: normal fluctuations and abnormal fluctuation. Normal fluctuations are caused by inevitable factors, which technically are difficult and non-economical to eliminate from the technique. These fluctuations have little effect on product quality. On the other hand, abnormal fluctuations are caused by system reasons (abnormal factors), which can have a serious influence on product quality. These effects however can be avoided and eliminated completely through process control (Barlow & Irony, 1992).

Statistical process control (referred to SPC) is a process monitoring approach based upon the mathematical statistics method (Wetherill & Brown, 1991). The SPC method is applied to monitor and control a process, and is an effective technique for keeping product quality at the required level to ensure product consistency. This technique has been widely applied in design, sales, service, and management processes, and in past decades has been developed through combining it with computer technology.

When SPC involves plotting measurements on a graph, the variances of measurements are plotted on an *x/y* axis with the *x*-axis usually representing time. A number of additional lines representing the average measurement and control limits are drawn across the chart. Control charts compare this variance against upper and lower limits to see if it fits within the expected, specific, predictable and normal variation levels. If it does, the process is considered in control and the variance between measurements is considered normal random variation inherent in the process. If, however, the variance falls outside the limits, the process is considered out of control and action should be taken (Lu et al., 2008).

SPC is applied to quality control, which can be traced back to the 1920s. In 1924, Walter A. Shewhart of Bell Telephone Laboratories originally proposed the concept of the control chart (Shewhart, 1931). A control chart is an important method and tool of statistical quality management (see Section 2.3 for overview of control charts). Shewhart kept working on and improving this scheme, and in 1931, he

published his book '*Economic Control of Quality of Manufactured Product*'. This book provided a good benchmark for the subsequent application of statistical methods to process control. Two of Shewhart's co-workers, Dodge and Romig, initially applied statistical theory to sampling inspection (Dodge, 1955; Dodge & Romig, 1959).

The applications of statistical quality control and SPC have been further improved and developed during the last eighty years. Since SPC was founded, it has been promoted and applied in the area of the industry. During World War II, SPC played an important role in ensuring the quality and timely delivery of military products; for instance, the U.S. Department of Defense decided to use mathematical statistics for the quality management of weapons and ammunition. The rule of the mathematical statistical methods was developed by the Standards Association, and it was used in planning quality management. A special committee was later established, and between 1941 and 1942, a number of initial quality management standards of the United States wartime were published (e.g. wartime standard Z1.1, Z1.2, Z1.3). During 1950-1980, Japan had widely promoted and applied SPC in industry, with Japanese companies creating a Total Quality Control (TQC) approach to quality management. In the 1970s, TQC greatly improved the competitiveness of enterprises in Japan, where cars, household appliances, watches, and electronic products for instance were present in a large number of international markets; this developed the Japanese economy substantially. Japanese product quality and productivity became a world leader, with renowned American management expert Professor Berger also commenting that one of the success cornerstones of SPC was in Japan (Barlow & Irony, 1992). Given the success of Japanese companies, total quality management theory had a huge impact in the world. Since the 1980s, SPC has been revived within industrially developed countries. Many world-class companies have also actively promoted and applied the SPC method in their internal operations (Chiu & Kuo, 2010).

2.3 Control Charts

A control chart is an important tool of statistical quality management; it is a graph whose construction is based on hypothesis testing. Control charts are applied to monitor whether the production process is in control (Bersimis et al., 2007).

In the production process, product quality is affected by random factors and system factors; random factors are composed of a large number of small accidental factors, whereas system factors are caused by some identifiable and obvious reasons. System factors can be found and removed through the appropriate measures. When a process is only affected by random factors, the average and variation quality characteristics of the product are basically stable, and are deemed in control. At this time, product quality characteristics are used to determine the probability distribution of the random variable. The distribution (or one of the unknown parameters) can be estimated based on observation data collated over a long time in steady-state by using statistical method.

When the distribution is determined, the mathematical model of quality characteristics is also determined. For production testing, the quality characteristics need to be tested to see whether they are consistent with this mathematical model. Therefore, at regular intervals, a fixed sample is extracted in the production line, and then a calculation of the quality characteristics is undertaken. If the values match the mathematical model, the production process is normal; otherwise, it means that system changes in production, or the process is out of control. When this occurs, the company needs to consider taking various measures including stopping production or carrying out an inspection. The causes will then need to be found and addressed in order to restore the system to normal production (Lee et al., 2008).

As control charts are easy to build and interpret, and in practice are very effective, they have been widely applied in process control and monitoring (Lee et al., 2009). When a control chart is applied to monitor process performance, it will include a number of points. The plotted points are usually averages of sub-groups (the number of subgroup is n) or ranges of variation between sub-groups; they can also be individual measurements. Control limits are calculated in the control chart, and include an upper and lower control limit. The specification of the control limits is

the most critical decision that has to be made at the design stage of a control chart. Control limits are usually determined from historical data for the statistic being monitored, and they define the boundary between the acceptance and the rejection region. The region on the control chart that the control limits mark out is called the control region. By comparing current data to these lines, conclusions can be drawn as to whether the process variation is consistent (in control) or unpredictable (out of control, affected by special causes of variation). When the process is in control, and the control limits are ensured, the sample statistics will be almost distributed between upper control limits and lower control limit. If some sample statistics distributed outside of the control limits; this showed that the process is out of control.

A typical Shewhart-type control chart showing both upper and lower control limit is shown in Figure 2.1; the blue point represents the target value. Almost all target values are distributed between both control limit lines with only one point falling outside the limit; this point is called the out-of-control point. This event is therefore interpreted as the process being out of control.

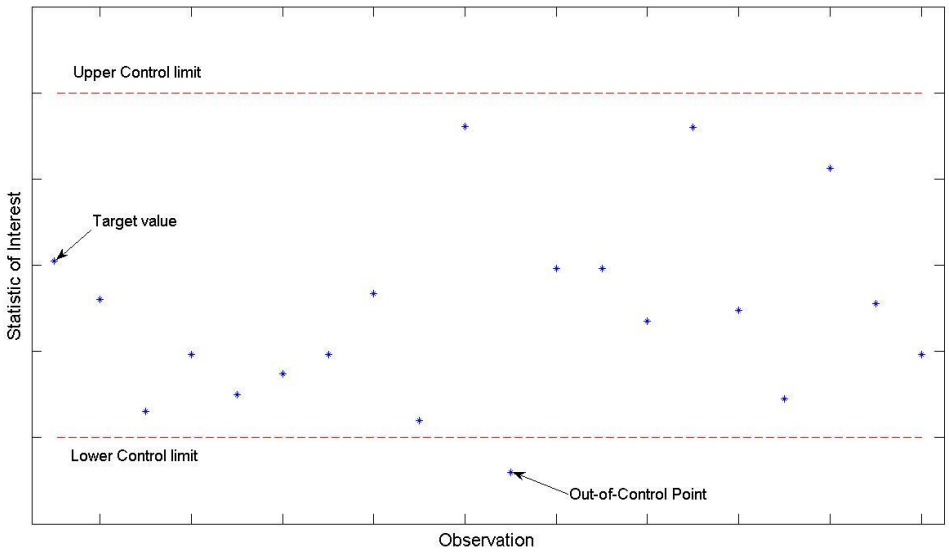

Figure 2.1 A Shewhart-type Control Chart

The application of control charts has been widely researched in process statistical control and statistical quality control. In 1989, Banks published a book '*Principle of Quality Control'* which introduced the application of control charts in statistical quality control. In 1991, control charts were studied as an important part of

Wetherill and Brown's book, *'Statistical Process Control - Theory and Practice'*. Lowery and Montgomery (1995) gave a review of literature on control charts for multivariate quality control. Control charts as a basic method of statistical process control were further explained in *'Introduction to Statistical Quality Control: Part 3'* (Montgomery, 1996).

Control charts can be divided into three common types, namely, Shewhart-type control charts, Cumulative Sum control charts and Exponentially Weighted Moving Average control charts. These will be discussed in the following sub-sections.

2.3.1 Shewhart-type Control Charts

Shewhart created the basis for the control chart and the concept of a state of statistical control through carefully designed experiments. All control charts based on Shewhart's theory and philosophy are called Shewhart-type control charts. It is the most popular SPC method used to detect whether the observed process is under control (Lee et al., 2009).

However in production processes, consecutive observations have a number of certain correlations. In a number of continuous processes, the correlation between the consecutive measurements is difficult to determine. When the control chart is designed, this problem needs to be addressed; a problem that was deeply considered and debated about in the 1980s and 1990s.

There are several solutions suggested. The first approach, which is historically the most classic one, consists of dealing with original data and adjusting the control limits of classical control charts (Vasilopoulos & Stamboulis, 1978; Schmid, 1995, 1997; VanBrackle & Reynolds, 1997; Zhang, 1998; Lu & Reynolds, 1999a, 1999b, 2001). Other approaches are based on the concept of residuals (Alwan & Roberts, 1988; Montgomery & Mastrangelo, 1991) or on monitoring statistics related to autocorrelations (Yourstone & Montgomery, 1991; Jiang et al., 2000).

A statistic of interest is calculated for individual groups of samples randomly collected from a process. A group of random process samples is called a rational

sub-group. The control limits of the Shewhart-type control chart can be expressed mathematically as:

$$CL = \mu \pm L\sigma, \tag{2.1}$$

where μ and σ denote the population mean and the population standard deviation of the statistic, and the value of factor L is selected.

In general, and assuming that the data is normally distributed, then it follows that approximately 95% of the sample will fall within the limits and the control limits can be calculated as:

$$CL_{95\%} = \mu \pm \frac{1.96\sigma}{\sqrt{n}}, \tag{2.2}$$

where n is the sample sub-group.

In practice, it is common to replace the 1.96 with 3 (in order for the interval to include approximately 99% of the sample means), with the control limits being defined as:

$$CL_{99\%} = \mu \pm \frac{3\sigma}{\sqrt{n}}. \tag{2.3}$$

Shewhart control charts have some limitations in the application conditions and principles. As they are based on the theory of mathematical statistics, they are applied to identify system factors in the production process, and then applied to control product quality. There are three application conditions:

(1) There are large quantities of quality characteristic data;

(2) The quality characteristic data are normal distribution, or near normal distribution;

(3) The quality characteristic variables are independent.

When the actual situation comes into conflict with one of the three application conditions, the limitation of Shewhart control charts will be evident. Kim et al. (2007) highlighted the limitation of the principles; that is that Shewhart control charts lack accuracy and are not sensitive to small changes in data.

2.3.2 Cumulative Sum Control Charts

Although, Shewhart control charts are applied to detect large process shifts, the effect is not obvious in detecting small or slow shifts. An alternative control chart was proposed by Page in 1954 called the Cumulative Sum (CUSUM) control chart. Since CUSUM control charts involves the calculation of a cumulative sum, if each point on the chart is the cumulative history (integral) of the process, systematic shifts are easily detected. Large, abrupt shifts are not detected as easily as in a Shewhart chart, so they are more effective than Shewhart-type charts.

CUSUM control charts have been developed and improved upon by many authors. In 1959, Barnad described a V-shaped mask which could be superimposed on the CUSUM chart. Johnson (1961) gave mathematical procedures for constructing CUSUM control charts. Johnson and Leone (1962) constructed CUSUM charts for controlling binomial distribution parameters. Ewan (1963) first applied CUSUM control charts in practice problems. Hawkins (1993) showed that CUSUM control charts were effective in detecting and diagnosing persistent shifts. Woodall and Adams (1993) designed a novel CUSUM control chart based on a fast accurate approximation of Average Run Lengths.

Given a CUSUM control chart contains several samples' information, but a Shewhart control chart plots points based on information only from a single subgroup sample, the CUSUM control chart is more efficient than the Shewhart-type control chart (Kim et al., 2007). When the subgroup of size is n=1, process shifts are between 0.5σ and 1.5σ (σ is population standard deviation of the statistic). The CUSUM control chart and the Shewhart control chart are applied to detect process shifts; however the CUSUM control chart needs only half the time of the Shewhart control chart to do this. The CUSUM Control chart is though slower than the Shewhart control chart in detecting large shifts (i.e. when the process shift is bigger than 3σ). The CUSUM control charts are therefore applied to detect small process shifts (Montgomery et al., 1996).

When the rational subgroups of size $n \geq 1$, the Cumulative Sum control chart is denoted by plotting the statistic:

$$C_i = \sum_{j=1}^{i}(\bar{X}_j - \mu_0), \tag{2.4}$$

where i is the rational subgroup number, μ_0 is the target for the process mean, and $\bar{X}_i$ denotes the average of each rational subgroup.

The control limits are usually calculated by using the V-mask procedure (Barnard, 1959; Johnson, 1961). A typical example of a CUSUM control chart is illustrated in Figure. 2.2, where the red points represent the Cumulative Sum of target value, and the blue line represents the control limit. Only one point is outside the limitation, thus this illustrates the process is out of control in this time.

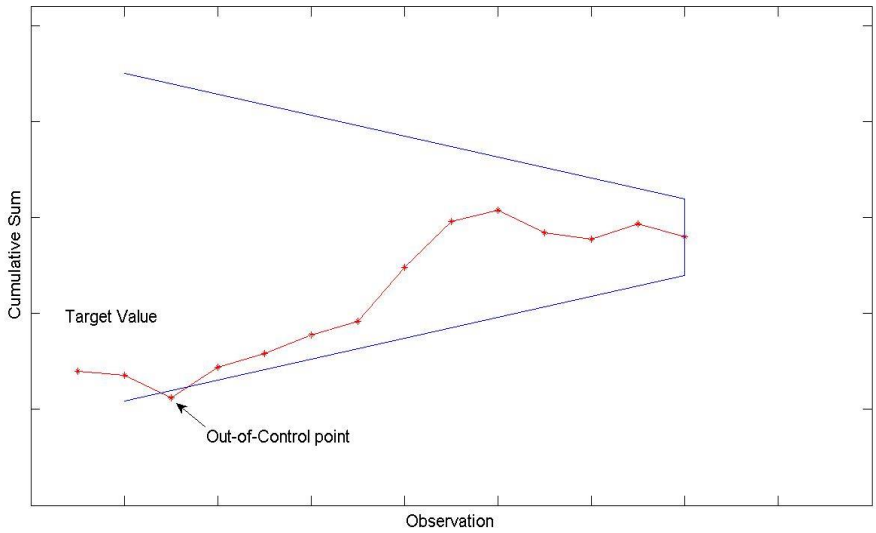

Figure 2.2 A Cumulative Sum Control Chart

2.3.3 Exponentially Weighted Moving Average Control Charts

Exponentially Weighted Moving Average (EWMA) control charts are another alternative to Shewhart control charts. The EWMA is a statistic for monitoring the process that averages the data in a way that gives less and less weight to data, as they are further removed in time. For the EWMA control technique, the decision depends on the EWMA statistic, which is an exponentially weighted average of all prior data, including the most recent measurement (Zhang, 1998; Lee et al., 2008). The EWMA

control chart was proposed by Robert in 1959. A number of EWMA methods and design strategies have been developed to detect shift. Crowder (1987) studied the average run lengths (ARL) properties of the EWMA chart and proposed 4 steps for the application of EWMA chart through computer simulation. Saccucci et al. (1992) introduced the robust EWMA control chart. Holmes and Mergen (1992) proposed the use of parabolic control limits for EWMA control charts; it performs better than EWMA with parallel limits in terms of ARL value. Wardell et al. (1994) explored the application of EWMA charts in the auto-correlated process. Montgomery and Mastrangelo (1995), and Mastrangelo and Brown (2000) researched the application of moving centreline EWMA in the model.

The EWMA statistic is calculated as:

$$EWMA_i = \lambda Y_i + (1-\lambda)EWMA_{i-1}\ ,\ i = 1,2,\cdots n, \tag{2.5}$$

where λ is the weighting factor ($0 < \lambda \leq 1$), and Y_i is the observation at time i. n is the number of the observation. $EWMA_0$ is equal to the population mean of the statistic (μ_0).

The EWMA control chart can be applied to detect a small or gradual drift in the process; it is based on the choice of weighting factor λ, but the Shewhart control chart can only react when out-of-control point occurs.

The parameter λ determines the rate at which past data entered into the calculation of the EWMA statistic. A value of λ=1 implies that only the most recent measurement influences the EWMA. Thus, a large value of λ (closer to 1) gives more weight to recent data and less weight to past data; whereas a small value of λ (closer to 0) gives more weight to past data. Lucas and Saccucci (1990) provide tables that help the user to select λ.

The EWMA control charts are similar to CUSUM control charts, however by comparison, the EWMA procedure is quite competitive in most practical situations (Lucas and Saccucci, 1990) and by the choice of the weighting factor λ, the application of the EWMA control charts can control the target value effectively, when the data included a number of noises (Montgomery et al., 1996).

The estimated variance of the EWMA statistic is:

$$\sigma_{EWMA}^2 = \frac{\lambda}{2-\lambda}\sigma^2, \tag{2.6}$$

where σ is the standard deviation calculated from the historical data.

The centre line for the control chart is the target value of the quality characteristic.

The control limit of the EWMA control chart can be calculated as:

$$CL_{EWMA} = EWMA_0 \pm L\sigma\sqrt{\frac{\lambda}{2-\lambda}[1-(1-\lambda)^{2i}]}, \tag{2.7}$$

where L is a factor defining the width of the control limits, and $EWMA_0$ is equal to μ_0. σ is the standard deviation calculated from the historical data.

A typical example of the Exponentially Weighted Moving Average control chart is illustrated in Figure. 2.3. In the figure, the upper and lower control limits with the centre line are drawn; the blue points represent the EWMA target value. 2 points fall outside the lower control limit, thus indicating that the process is out of control.

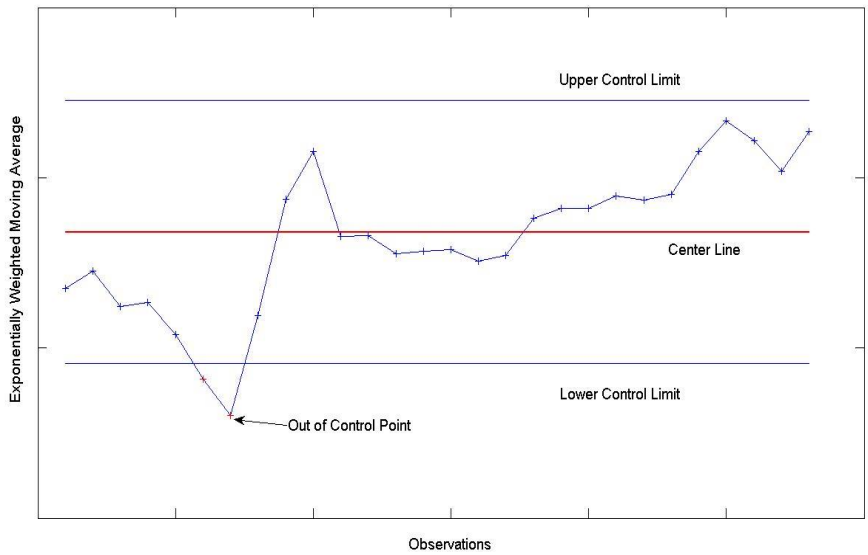

Figure 2.3 An Exponentially Weighted Moving Average Control Chart

SPC is the Simple Statistical technique which was applied and developed for monitoring some important variables (Wetherill and Brown, 1991). Traditional SPC methods are applied to detect and monitor the important quality variable, through the monitoring chart of individual process variable and action. There is therefore a limitation. When the number of required process variables increase, the number of monitoring charts also need to be augmented. As the number of required charts are too large, traditional SPC methods are often unsuitable. Therefore, the limitation of SPC is that it does not consider the multivariable nature of a process.

2.4 Multivariate Statistical Process Control

Multivariate Statistical Process Control (MSPC) comprises a number of modelling techniques that may be seen as a response to the issue of dealing with large, highly correlated data sets. It uses historical data of processes to develop useful process fault diagnosis and control tools. Its name reflects a link with SPC techniques and its application shares many similarities. In this thesis, the benefits of MSPC will be discussed and the limitations in current technology will be identified.

In industrial process data, the recorded process variables are huge, and these variables are collinear and high correlated. In practice, process variables are recorded irregularly, such as missing value numbers, or corrupted value numbers due to process and measurement noise. In these cases, traditional SPC methods are unable to adapt to modern industrial processes. MSPC is an alternative approach to traditional SPC in the area of process monitoring, overcoming SPC's weakness. The advantage of MSPC is that the multivariate method can reduce some dimensions in the process data. The required univariate control charts are reduced clearly, and the MSPC method collects all process data, including past data. As MSPC explicitly considers the multivariate nature of the data, it identifies the correlations that exist; thus MSPC is suitable for dealing with large and highly correlated data sets. The primary target of MSPC is to maintain product quality in a desired product specification and to control a process in a desired state. By controlling and monitoring a number of key quality process variables, product quality can be as close as possible to the desired value (Doan & Srinivasan, 2008).

Hotelling initially proposed Multivariate Statistical Quality Control (MSQC) and Multivariate Statistical Process Control (MSPC) in 1947. In the middle of 1980s, two researchers, Alt and Jackson, applied and developed Hotelling's work. Alt (1985) studied the application of Hotelling's T^2-control chart to monitor the variability of a process. Jackson (1985) discussed some techniques, such as the Hotelling T^2-control chart, and the use of principal components for control charts. The control chart limit of a multivariate process was discussed and constructed, leading to the theory of a multivariate control chart being proposed based on additional information from the recent history of the process (Jackson, 1985). In 1994, Adams introduced a multivariate control web; a graphical approach that offers the simultaneous display of univariate and multivariate summary statistics. In recent years, because the MSPC method can provide on-line monitoring and detection of process faults, it plays an important role in industry (Golshan et al., 2010).

A large number of applications of MSPC techniques have been successful in the area of process monitoring (MacGregor et al., 1991). MSPC can provide on-line monitoring and fault diagnosis of a continuous polymerization process (Nomikos & MacGregor, 1994; Camacho & Pico, 2006). MSPC can not only be applied to monitor a continuous process, but also can monitor batch processes by using monitoring charts (MacGregor & Kourti, 1995; Camacho et al., 2008). For instance, the MSPC technique can be used to predict in Iron Ore for process monitoring (Tano et al., 1993); furthermore MSPC can be applied to analyse and monitor the manufacture of photo-micrographic paper (Miller et al., 1995) and an industrial ceramic melter (Wise et al., 1991a). In relation to its application to general batch processes, MSPC techniques have also been proven to be very successful (MacGregor & Kourti, 1995; Martin & Morris, 1996). MSPC techniques can be extended to monitoring a semi-batch process, such as: a nuclear waste storage tank (Gallagher et al., 1996). In this case, the MSPC method is applied to develop a statistical model of the process. The model is applied to determine some changes in the system, for both on-line and off-line monitoring (Wise & Gallagher, 1996).

2.5 The Use of Multivariate Statistical Projection Methods

Every process has a number of process variables that require observation, such as temperature, feed rate, concentration, flow rate, PH and a number of final products. In a penicillin production process for instance, the final products are Penicillin concentration and Biomass concentration (Briol et al., 2002). Process variables are typically measured on-line, but quality variables are measured off-line. With multivariate data, the major problem that arises is that the measured variables are not independent; rather these variables are auto-correlated in time and are highly correlated with one another at any given time (collinear). This is due to the underlying relationships between the variables, where the measurements were taken or due to the nature of the process; given this, it is important to determine the relationships between the variables. Furthermore, as all past data have contributed to the whole performance of the process, all process data need to be considered. Therefore, some MSPC methods are applied to overcome these difficulties (Bersimis et al., 2007).

MSPC covers a wide range of techniques. In these methods, the bases of MSPC are the statistic projection techniques of Principal Component analysis (PCA) and Partial Least Square or Projection to Latent structure (PLS). They are used to analyse process data and to develop predictive models in support of process monitoring and control in real industrial processes. A mathematical overview of PCA and PLS is described in Chapter 3.

PCA is a well-known multivariate statistical method (Mardia et al., 1989; Jackson, 1991). The main areas of PCA application in process analysis are in data reduction (the dimensionality of data), abnormal operation detection, variable classification, early warning of potential malfunction and fault identification (Martin & Morris, 1996).

Although all process variables can be monitored, in practice there are a small number of underlying characteristics that drive the process. The aim of statistical projection techniques is to create a new set of latent variables, and allow the true dimensionality

of the system to be reflected. PCA is applied to reduce the dimensionality of process data through defining a series of new latent variables (or principle components), which are each linear combinations of the original variables. Within the new latent variables (principal components), the first principal component explains the greatest amount of variation, i.e. it explains most of the information in the data. The second principal component is orthogonal to the first principle component; the information maximizes the remaining variance of the data project on second principle component, and so on. The components are also orthogonal to each other (Piovoso & Kosanovich, 1996), hence the principal components are uncorrelated. The entire sets of scores define the process data, and the loadings are the statistical process model. The sets of scores and the PCA loadings can be applied to determine if the present process operation has changed its behaviour, relative to the data that was used to define the scores and loadings. Typically, the first two principal component scores contain all the important information for early warning of potential malfunction (Martin & Morris, 1996). The number of principle components is a crucial factor in PCA model development. Cross validation is a very useful method for estimating the optimal number of principal components (Wold, 1978). Further aspects of cross validation are explained in Chapter 3.

The initial PCA idea was proposed by Galton (1889). In 1901, the PCA method was introduced as a technique for plane fitting by Pearson (1901). The mathematical concept of PCA was completed initially by MacDonell (1902). Pearson and MacDonell were co-workers. In the 1930s, PCA was officially independently named and developed by Hotelling. Hotelling proposed PCA for analysing the covariance and correlation structures between a number of random variables. In 1949, Burt published a paper, *'Alternative Methods of Factor Analysis and their Relations to Pearson' s Method of Principle Axes',* which provided some proper attributions in the early history of PCA development. Given computational difficulties, the development of PCA was not quick in the mid-20th century. In the 1960s, accompanied by the Quantitative Revolution of social sciences, the applications of computer techniques were extended to data processing and industrial control; therefore the applications of PCA have occurred widely.

A large number of new ideas were considered, and then the interpretations and extensions of PCA were introduced by Rao (1964). Gower (1966) discussed links between PCA and various other statistical techniques, whilst Jeffers (1967) gave two case studies in which the uses of PCA go beyond that of a simple dimension-reducing tool.

Various authors provide a theoretical introduction of PCA (e.g. Mardia et al., 1974). PCA as a tool is applied when more than one variable needs observing, and where there is an inherent interdependence between the variables. In Kendall's book *'Multivariate Analysis'* (1980), PCA is compared to Factor Analysis. Muirhead (1982) explained principal components and other related topics; Seber (1984) stated that the introduction of PCA is based on dimension reduction properties; Krzanowski (1988) focussed upon PCA principles and methodology; and lastly, Johnson & Wichern (1992) studied some basic PCA applications.

Overviews of the concepts, properties and applications of PCA were presented by Wold et al. (1987), Geladi & Kowaski (1986), Mackiewicz & Ratajczak (1993) and Wise &Gallagher (1996). A review of multivariate statistical process control based upon PCA's statistical projection techniques was undertaken by Martin & Morris (1996).

In recent years, some modifications of the PCA technique have been researched. The aim of novel PCA techniques is to extend their application and to improve their capabilities. Nomikos and MacGregor (1994) proposed multiway PCA (MPCA). Typically, batch process is time varying in nature, and all data in the batch will affect the final production. Measurement data from a batch process is stored as a 3-dimensional matrix (X) of size $I \times J \times K$, where I is the number of batches, J is the number of measured observations in a complete batch and K is the number of measured variables. The PCA method has been developed for application in a 2-dimensional matrix; if PCA is to be applied in batch process data, 3-dimensional batch process data must be transformed into a 2-dimensional matrix. The PCA can then be applied to analyse the process data. There are different approaches for rearranging the data sets. The most common approach is batch-wise unfold, which unfolds the matrix in accordance to the direction of batches (Nomikos et al., 1994; see Chapter 3 for further details). Multi-scale PCA (Bakshi, 1998) is another novel

PCA technique, as it combines the ability of PCA with wavelet analysis to extract deterministic features. In this method, the relationship between the variables is determined by PCA, but the relationship between the measurements is determined through wavelet analysis. The benefit of Multi-scale PCA is that this method can detect abnormal operation earlier and more clearly than traditional PCA (Facco et al., 2009).

Traditional PCA defines a linear projection of the data, thus it is not able to consider process dynamics. Dynamic PCA (DPCA), introduced by Ku et al. (1995), tries to use a well-known 'time lag shift' method to include dynamic behaviour in the PCA model. DPCA is applied in the area of statistical process monitoring, by using time-lagged variables. This method has been applied to process monitoring and fault detection in a process simulation of Tennessee Eastman. The Tennessee Eastman process model is a realistic simulation program of a chemical plant, consisting of five major transformation units (a reactor, a condenser, a compressor, a separator, and a stripper), where 41 measurements are generated along with 12 manipulated variables. The results demonstrated that DPCA statistics had higher fault detection rates, presented lower auto-correlation levels, and were able to sustain the out-of-control signals during the whole faults duration; PCA statistics, by comparison, often return to their in-control regions leading to a false sense of normality. The proposed methodology (DPCA) therefore is more effective than the traditional PCA method (Shen et al., 2012).

Although PCA is suitable for process monitoring, some limitations of PCA need to be considered (Piovoso & Kosanovich, 1996; Zhang, 2009). Firstly, it only considers orthogonal transformations (rotations) of the original variables. Secondly, PCA is based only on the mean vector and covariance matrix of the data. Thirdly, dimension reductions can only be achieved if the original variables were correlated; if the original variables were uncorrelated, PCA does nothing, except for ordering them according to their variance. Lastly, PCA method considers all inputs and output at a specific sample instant.

Principal Component Regression (PCR), first proposed by Massy (1965), is an extension of PCA; it is applied in the modelling of Y-data from X-data. Linear regression is an approach for modelling the relationship between a dependent

variable and one or more explanatory variables. The process is called simple linear regression, where one explanatory variable needs consideration; if there is more than one explanatory variable, the process is called multiple linear regression (Warne, 2011).

Linear regression is a classic type of regression analysis. Linear regression has been widely applied within industry. With this method, unknown parameters can be easily fitted; this is because the model is based on linear relation, and furthermore, the statistical properties of the resulting estimators are easier to obtain. But Standard regression methods are based on a number of typical assumptions. In the real world, these assumptions are often unrealistic.

A benefit of PCR is that the multi-collinearity problem can be overcome, when two or more of the explanatory variables are close to being collinear. PCR can also deal with such situations by excluding some of the low-variance principal components in the regression step. In addition, by usually regressing on only a subset of all the principal components, PCR can obtain the result of dimension reduction, through substantially lowering the effective number of parameters characterizing the underlying model. Particularly, PCR is applied in settings with high-dimensional covariates. Through the appropriate selection of principal components to be used for regression, PCR can lead to an efficient prediction of the outcome, based on the assumed model (Dodge, 2003; Bair et al., 2006; Mevik & Wehrens, 2007).

In the PCR technique, the procedure is divided into two steps. Firstly, PCA is applied to the predictor data set (X-data) and then, the response data set (Y-data) is regressed on the scores of predictor data set. In the first step, some factors or information are ignored, because these data do not significantly contribute to the predictor data set; on the other hand, in the second step, these factors or information need to be considered because they are highly correlated with the response data set (Y-data). The PCR method defines a new set of uncorrelated latent vector in the space of X-data; it is applied to minimise the variance covariance matrix $X^T X$ (Geladi & Kowalski, 1986; Jackson, 1991).

PCR has other limitations. The PCR method described above is based on classical PCA and considers a linear regression model for predicting the outcome based on the

covariates. In addition, the principal components are obtained from the Eigen-decomposition of *X*, which involves observing the explanatory variables only. The resulting PCR estimator obtained from using these principal components as covariates therefore, need not necessarily have satisfactorily predicted outcome performance.

PLS can address this issue. Similar to PCR, PLS also uses derived covariates of lower dimensions. However unlike PCR, the PLS algorithm used examines both *X*-data and *Y*-data, and extracts factors (called components or latent variables), which are directly relevant to both sets of variables (Mevik & Wehrens, 2007).

PLS, also called Projection to Latent Structures, has become a popular MSPC technique. PLS is a projection method that models the relationship between a response matrix *Y* and a predictor matrix *X*. PLS is able to define independent latent variables from the covariance structure of given groups of highly correlated, or collinear variables (MacGregor, 1995). Thus, PLS can be used for dimensionality reduction and modelling (Wold, 1966; Wold, 1975).

The basic mathematical and statistical background of PLS can be found in literature such as Manne (1987), Lorber et al. (1987) and Helland (1988). PLS was used to handle collinearities among independent variables in multiple regressions (Wold et al., 1984). In 1986, the Nonlinear Iterative Partial Least Squares (NIPALS) method was applied in the development of PLS model. In this method, the predictor matrix and the response matrix were decomposed to a sum of rank one component matrices (Geladi & Kowaiski, 1986). The detail of the procedure will be introduced in Chapter 3. In 1988, the mathematical and statistical structure of PLS regression had been discussed and proposed by Hoskuldsson (1988). An alternative method for calculating PLS factors was introduced by Jong (1993). In this method, PLS factors are derived from the initial data matrix; this is to say, the deflation of the data matrices is not required.

The objective of Projection to Latent Structures is to construct a linear relationship between the predictor matrix and the response matrix; the observation of highly correlated or collinear variables needs to be included. However, PLS eliminates

redundancies in the original data sets through linear combination and defining a new set of variables; the new set of variables is independent.

PLS is similar to PCR, but it is not the same model as the PCR model. PLS maximizes the covariance of the two data sets (*X*-data and y-data), whilst PCR only maximizes the variance of a single data set (*X*-data).

The regression relationship of PLS needs be built in a stepwise manner. There are several ways, but the most common approach is the Non-linear Iterative Partial Least Squares (NIPALS) algorithm of Wold (1966). It is also the most popular method to calculate the principal components from a multivariate data set (Wold, 1987; Geladi & Kowalski, 1986; Martens & Naes, 1989).

The NIPALS algorithm does not calculate all the Principal Components simultaneously; rather it calculates the first principal component and then, the product of its score and loading is subtracted from the data matrix $\boldsymbol{X}$. The residual matrix is then applied to calculate the second principal component and so on.

NIPALS is a powerful method to calculate the eigenvectors of a matrix (Goldberg, 1991). When the NIPALS algorithm is applied, the score and the loading vectors are the eigenvectors of the $X \cdot X^T$ and $X^T \cdot X$ matrices respectively (Geladi & Kowalski, 1986). In the PLS method, for each step, the NIPALS algorithm calculates two latent vectors, *T* and *U;* these are a linear combination of the predictor data set (*X*-data) and response data set (*Y*-data) data sets respectively.

There are some alternative methods to the NIPALS algorithm, such as the maximum eigenvalue of the residual sample covariance matrix, and the successive Singular Value Decompositions of the cross-covariance matrix of the residual data sets. These methods can also be used to calculate latent dimensions (Hoskuldsson, 1988; Kaspar & Ray, 1993; Lindgren, et al., 1993; Wang et al., 1994).

The number of latent variables is a crucial issue in the application of the PLS model. When the optimal number of latent variables is determined, a satisfactory predictive relationship between the *X*-data and *Y*-data can be obtained. This will avoid over-fitting and under-fitting. Although there are multiple ways to do this, cross validation

is a very useful method to determine the optimal number of latent variables (Stone, 1974; Wold, 1978).

There are certain properties of PLS that have made it so popular in process monitoring (Ferrer et al., 2008). It is a good alternative to classical multiple linear regression and principal component regression methods, as it has been shown to be robust for limited sized data sets and highly collinear data. Furthermore, it has relatively low computational requirements and is efficient in dealing with situations where there are missing measurements.

Some modifications of the PLS technique have been researched recently. For batch process monitoring, MPLS has proven to be useful (Lennox et al., 2001). Traditional PLS is applied to monitor the process based on a fixed model of the system. Given that most industrial processes are time varying, the fixed model can be unsuitable. To address this limitation, a Recursive PLS algorithm was proposed by Helland et al. (1991), where the current measurements and parameters of the PLS model were applied to update the PLS model. A further development from this work by Qin (1998) was applied to a catalytic reformer process. Another Recursive PLS algorithm was introduced by Dayal and MacGregor (1997a). In this method, the covariance matrices of predictor and response matrices (X^TX, X^TY) can be updated. The applications of this method were in a mineral flotation circuit (Dayal & MacGregor, 1997b) and in a fluid catalytic cracking unit (Lennox et al., 2003).

A typical industrial plant usually includes large-scale processes with many process units and measuring devices. For monitoring these processes, a multi-block technique is developed to combine with the PLS method. A multi-block algorithm was originally introduced by Gerlach et al. (1979). Multi-Block PLS (MBPLS) is an extension of PLS, that was originally proposed by Wold et al. (1987b), and further developed by Wangen & Kowalski (1988). They proposed a novel multi-block PLS algorithm, the Interconnected Multi-Block PLS (IMB-PLS). It is especially suited for large complex systems, which consist of many distinct sections that are connected by a few variables. Traditional PLS models the predictive relationship between two blocks of data, whereas the IMB-PLS models the predictive relationship(s) between more than two data blocks (Liu et al., 2011). Wold et al. (1996) proposed a general hierarchical PLS algorithm based on hierarchical Multi-

Block PLS. This method was applied in a residue catalytic cracker unit, where the results showed that the blocked models can provide better model prediction than the standard method.

To enable PLS to track the dynamics of batch processes, multi-way PLS (MPLS) was proposed (Zhang & Lennox, 2003). First introduced by Wold et al. (1987), it has been shown to be particularly useful for monitoring batch processes (Nomikos & MacGregor, 1994; 1994b). MPLS is an extension of PLS that enables the handling of 3-dimensional data arrays (Nomikos & MacGregor, 1994). If MSPC is to be applied, 3-dimensional data must be transformed to a 2-Dimensional matrix; the approaches for rearranging the data sets is the same as for MPCA. PLS is then applied to the unfolded matrix, which has the dimension $I \times JK$ (Wu & Lennox, 2006). More details of unfolding matrix methods can be found in Chapter 3.

Some MPLS applications have been reported. In an industrial batch polymerization reactor, PLS and MPLS applications are provided by analysing historical data from the catalytic cracking section of a large petroleum refinery, when monitoring the industrial batch process (MacGregor & Kourti, 1995). In the semi-batch emulsion polymerization of styrene−butadiene rubber, MPLS is applied to control final product quality, based on using only a few readily available on-line measurements and some off-line measurements (Yabuki & MacGregor, 1997). In a fed-batch fermentation system, MPLS is used to provide long-term predictions of product concentration at the end of the batch, and to determine suitable substrate feed-rates to ensure that batch productivity reaches a required level (Lennox et al., 2001). Lastly, in condensation polymerization and emulsion polymerization systems, MPLS is used as a well-established method for analysing batch process historical data, and for monitoring the progress of new batches (Cerrillo & MacGregor, 2004).

This technique analyses process behaviour relative to the mean trajectories of the process variables. In doing so, a major nonlinearity in the data is removed. However, there remain situations when this approach is insufficient to track nonlinear process behaviour.

A major limitation with PLS is that industrial processes are always nonlinear to some extent. This is not always a problem as many processes only operate around limited

operating regions, where linear PLS techniques tend to provide acceptable accuracy. However, batch processes often operate over relatively large spaces, and many chemical and physical systems display nonlinear performance; hence nonlinear extensions to PLS may be required. Given this, a number of different methods have been proposed to provide a nonlinear PLS algorithm.

2.6 Nonlinear PLS

Given the linear PLS model cannot predict nonlinear systems, several nonlinear PLS methods have been proposed and applied. These tend to be divided into Type I and Type II methods. A detailed overview of these methods is provided by Wold (1989).

In the Type I Nonlinear PLS method, the observed variables are appended with nonlinear transformations. Following this, traditional linear PLS is then applied; for example, the X matrix can be augmented with transformed terms. The addition of transformed terms in X within PLS models was first proposed by Wold (1989), where he proposed the use of quadratic terms in the PLS model. In Type I methods, the inputs into the PLS model are specified to be cross product and squared terms of the input variables. Related works in this area include that of Berglund et al. (1997; 1999), who utilized quadratic and higher order polynomial terms, while ignoring cross-terms. However, for Type II methods, nonlinear functions are implemented in the PLS model's inner structure; for instance, Wold (1989) used quadratic functions of the inner variables. The Type II nonlinear PLS method was extended by Baffi et al. (2000). In contrast to the Type I nonlinear PLS method, the Type II nonlinear PLS method assumes a nonlinear relationship within the latent variable structure of the model. Type I and II non-linear structures are integrated within MPLS models to enable them to more accurately approximate nonlinear batch processes. These are described in Chapter 5.

However, Type I and II nonlinear PLS methods have their limitations. When the order of the nonlinearity does not match that of the process, problems arise and the accuracy of the prediction is reduced significantly. In real studies, the exact order of any nonlinear relationship will not be known a-priori, hence the required expansion

of X will be difficult to determine. Type I nonlinear MPLS therefore is not recommended for the prediction of the nonlinear system. Type II nonlinear PLS algorithms rely upon using polynomial nonlinear mapping, based upon the assumption that the relationship between the predictor and the response latent variables can be modelled by means of that particular polynomial expansion (Baffi et al., 2000). When the inner relation of the system is also similar to this particular polynomial expansion, this Type II nonlinear PLS model can predict the nonlinear system very well, and vice versa.

For increased functionality, the use of a neural network was proposed in the model's inner structure (Qin et al., 1992). An alternative to using polynomials in the inner relationship of the PLS model is to use a neural network to describe this relationship. This method is called Neural Network PLS (NNPLS). The structure of NNPLS will be introduced in Chapter 3. The NNPLS method can be divided into 2 parts: (i) the PLS outer model (the same as in the linear PLS method), which is used to transform the data to score variables; and (ii) inner network train algorithms, which are applied.

Multilayer neural network was proposed by Werbos (1974). This neural network was directly applied to determine the relationship between matrix Y and X by McAvoy et al. (1989) and Aguado et al. (2006). The Neural Network model can be represented as:

$$Y = N(X) + E, \tag{2.8}$$

Where E is the residual matrix after regression; $N(\cdot)$ stands for nonlinear map performed by the network.

Although this method performs better than linear techniques in some case, it suffers from the same problem as the ordinary least-squares method. For example, the number of weights in a multilayer network of m inputs and p outputs could be larger than the number of observations. Therefore, a number of the weights cannot be uniquely determined from the observed data, leading this method to result in over-fitting (Piovoso & Owen, 1991).

The NNPLS approach differs from the direct network method in data application. The data is not directly applied to train neural network in the NNPLS method; rather

it needs to be transformed into PLS outer models. This transformation decomposed a multivariate regression problem into a number of univariate regressors; whilst each regressor is run by a neural network. The major benefit of doing this is that a Single-Input-Single-Output (SISO) network is trained at the same time. The number of weights to be determined is much smaller than that in an *m*-input-*p*-output problem when the direct network method is applied; *m* is the number of causal variables in the *X* data, *p* is different quality indices in the *Y* data (Qin, 1992). When the number of weights reduces to a small number, over-parameterization can be circumvented.

The advantage of applying neural networks in the inner regressor is due to their nonlinear approximation property. It has been proven that a network with only one hidden layer of sigmoidal units is enough to have universal approximation properties (Hornik et al., 1989). Hornik et al. (1990) also proved that its derivative can be approximated by an one-hidden-layer network. Huang (1991) later showed that the number of hidden units, such as one-hidden-layer network, is bounded.

The concept of artificial neural network (ANN) is also applied by a number of other researchers, such as Kramer (1991), Dong and McAvoy (1994) and Saunder et al. (1995). ANN consists of a class of nonlinear models. Back propagation neural network (BPNN) is widely used and is capable of complicated multidimensional mapping (Werbos, 1988; Hecht-Nielsen, 1989; Heermann & Khazenie, 1992). A typical BPNN model is composed of many idealized layers of nodes, specified by node characteristics (weights), the learning rules (transfer or ‘sigmoid’ functions), network interconnection geometry (different layers), and dimensionality (the number of layers and nodes). BPNN resembles the human brain, in that the model learns and stores knowledge (Mehra & Wah, 1992; Werbos, 1994). This learning feeds back into the model to change the weights of nodes between layers, in order to decrease errors between predicted and measured values. Thus, BPNN takes nonlinearity into account using the sigmoid functions that connect the BPNN layers of nodes. The weights of redundant spectral bands (e.g. adjacent spectral bands) are also significantly decreased through the back propagation learning process. After the node weights and sigmoid functions have been determined through the training process, the BPNN model can be used for predictions with new input data (Li et al., 2012).

NNPLS based on BPNN, and Linear PLS, are both applied to the estimation of soil properties (Ramadan et al., 2005). The Back-Propagation Neural Networks method combined with PLS was found to have the most predictive power with the independent test set. NNPLS based on BPNN is also applied for predicting a wide range of soil chemical and physical properties from their mid-infrared (MIR) spectra (Janik et al., 2009). The results demonstrate that NNPLS has the advantages of robustness, and has the qualitative and quantitative features of PLS and the nonlinear capabilities of neural networks.

Qin and McAvoy (2002) proposed a 'generic' nonlinear PLS algorithm; in this method, a feed-forward neural network is applied to robust PLS regression. This was extended further by Baffi et al. (2000), who utilized Radial Basis Functions. Other related works in this area include that of Frank (1990) and Wold (1992), who used smoothing splines to provide the non-linear function within the model, and Hiden et al. (1998), who proposed the use of genetic programming.

Wilson et al. (1997) proposed the radial function network PLS algorithm (PLS-RBF). RBF network training algorithms can be reduced to linear regression problems; though the nonlinear model still can be obtained (Chen et al., 1991; Lennard & Kramer, 1991). PLS-RBF has been applied to detect faults in an industrial overheads condenser and reflux drum plant configuration.

Measurements of process variables often contain outliers, which have a large negative impact on model accuracy and reliability; detecting outliers in sampling data has therefore been given more and more consideration (Chen et al., 1998). The radial basis functions-partial least squares (RBF–PLS) approach is applied to detect such outliers in complex systems (Munoz & Muruzabal, 1998; Zhao et al., 2006). For example, RBF-PLS has been applied in the sulphur recovery process contaminated with natural and synthetic outlier (Garces & Sbarbaro, 2011). The results show that the proposed method is effective and outperforms conventional approaches, by reducing swamping and masking effects. In recent years, RBF-PLS has been widely applied to identify face images (Jiang et al., 2012). RBF-PLS has also been applied to dynamic system identification (Yin et al., 2006). Simulation results of nonlinear dynamic system identification demonstrate the adaptive tracking ability and high learning speed of the proposed algorithm.

In this thesis, the conjugate gradient learning method has been chosen; the reasons behind this decision will be clearly explained in Chapter 3. The Mathematical Overview of Neural Network PLS, Type I and II nonlinear PLS are also described in Chapter 3.

2.7 Summary

This chapter has reviewed MSPC techniques and discussed their application in process monitoring. Several MSPC methods, such as PCA, PCR and PLS, were introduced, and critiqued for advantages and disadvantages. This chapter also discussed control chart applications; many enhancements and extensions to control charts have been proposed. Lastly, nonlinear PLS and Neural network PLS, including their applications, are introduced and described.

Chapter 3

Mathematical Overview

Multivariate statistical process control (MSPC) techniques play an important role in industrial batch process monitoring and control. In this chapter, some basic algorithms of MSPC techniques such as Multiple Linear Regression (MLR), principal component analysis (PCA), principal component regression (PCR) and partial least squares (PLS) are described, followed by some related algorithms, such as unfolding approaches and cross-validation. Their uses are also presented. In addition to outlining the basic PLS algorithm, it also discusses several of its extensions; firstly, it introduces Multi-way PLS (MPLS) analysis of batch data examining both Batch-wise and Variable-wise unfolding; secondly, it discusses Nonlinear PLS and lastly, it provides an overview of Neural Network PLS (NNPLS).

The chapter is divided into the following sections:

3.1) describes some basic algorithms of MSPC techniques, such as Multiple Linear Regression (MLR), principal component analysis (PCA), principal component regression (PCR) and partial least squares (PLS);

3.2) discusses the use of cross-validation;

3.3) introduces Multi-way PLS (MPLS) analysis of batch data, and gives the introduction of Batch-wise unfolding and Variable-wise unfolding;

3.4) discusses the nonlinear PLS model;

3.5) gives an overview of Neural Network PLS; and

3.6) provides a summary of this chapter.

3.1 MSPC Techniques

Multivariate Statistical Process Control (MSPC) was discussed in Chapter 2. It covers a wide range of techniques, but a number of basic MSPC algorithms will be described.

3.1.1 Multiple Linear Regression (MLR)

Multiple Linear Regression is also called Ordinary Least Squares (OLS). This approach is described as follows (Geladi & Kowalski, 1986):

A variable y is the goal and there are m causal variables, x_j ,. MLR is applied to measurements of the variables to build a linear relationship between x_j and y.

This can be represented mathematically as:

$$y = b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_mx_m + e \tag{3.1}$$

$$y = \sum_{j=1}^{m} b_jx_j + e \tag{3.2}$$

$$y = x^TB + e \tag{3.3}$$

In Equation 3.2, x_j are called independent variables and y is a dependent variable, the b_j are sensitivities and e is the error or residual.

The above formulas describe multi-linear dependencies for only one sample. Then, the number of samples can be extended to n.

$$Y = XB + E \tag{3.4}$$

Figure 3.1 shows the structure of MLR. Y is changed to a matrix ($nx1$), X is changed to a matrix (nxm), B remains the same. Y is dependent matrix, and X is independent matrix.

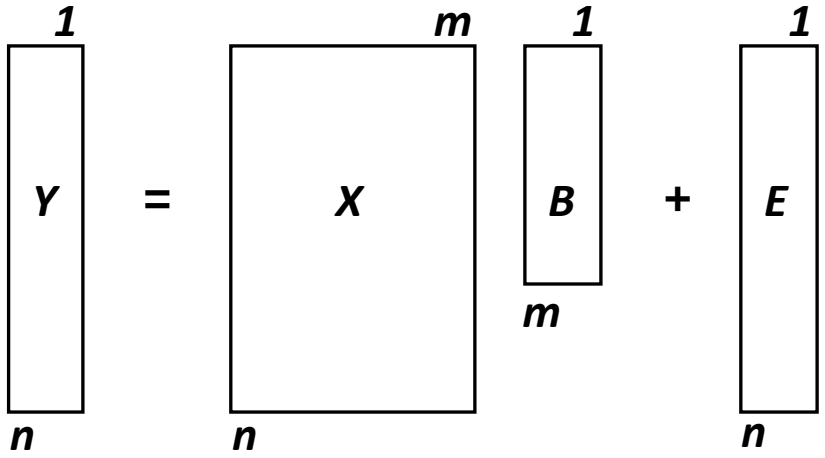

Figure 3.1 The Structure of MLR (Geladi & Kowalski, 1986)

In this case, *n* is the number of samples and *m* is the number of independent variables. There are three possible relationships between *m* and *n*.

(i)When *m* > *n*, there are more variables than samples. In this case, there are an infinite number of solutions for *b*.

(ii)When *m* = *n*, the number of samples is equal to the number of variables. There is one unique solution. In this case, $E = Y - Xb = 0$, where *E* is called the residual vector, and it is equal to 0.

(iii) When *m* < *n*, there are more samples than variables. This case does not allow an exact solution for *b*, but a solution can be found by minimizing the length of the residual vector *E* in the following equation: $E = Y - Xb$

The most common method for identifying *b* is the Least Squares method. The Least Squares solution is obtained as follows: $b = X^{*}Y$

Where X^{*} is the pseudo-inverse of *X*. In mathematics, the pseudo-inverse X^{*} of matrix X is a generalization of the inverse matrix.

Therefore X^{*} is obtained from $X^{*} = (X^{T}X)^{-1}X^{T}$ (3.5)

Hence $b = (X^{T}X)^{-1}X^{T}Y$ (3.6)

Equation 3.6 relates to the most frequent problem in MLR; that is, the inverse of $X^T X$ may not exist. Collinearity, zero determinant and singularity are all names for the same problem. At this point, it might appear that there always has to be at least as many samples as variables, but there are other ways to formulate this problem; for instance, one can delete some variables in the case $m > n$. Many methods exist for choosing which variables to delete, as relevant variables have to be discarded to avoid these problems. It requires X to have more rows (samples) than columns (variables), hence the MLR method fails to give a model which is robust to noise.

3.1.2 Principal Component Analysis (PCA)

PCA is a statistical procedure. In this method, a set of data is assembled in a matrix X, where the rows are sampled process variables at a fixed sampling time, and a column is a uniformly sampled variable. A given matrix X (n rows and m columns) can be decomposed to several sub-matrixes. This is represented mathematically in Figure 3.2.

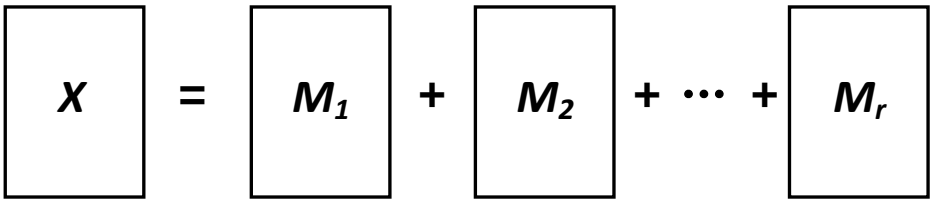

Figure 3.2 The Decomposition of X matrix (Wold, 1987)

In Figure 3.2, r is the rank of X matrix.

M_a ($a = 1 \cdots r$) can be expressed as the outer products of two vectors; i.e.,$M_a = t_a p_a^T$ the column vector is score t_a and the row vector is loading p_a^T.

$$X = t_1 p_1^T + t_2 p_2^T + \cdots + t_r p_r^T. \tag{3.7}$$

Or the equivalent formula:

$$X = T P^T. \tag{3.8}$$

Figure 3.3 presents the structure of PCA.

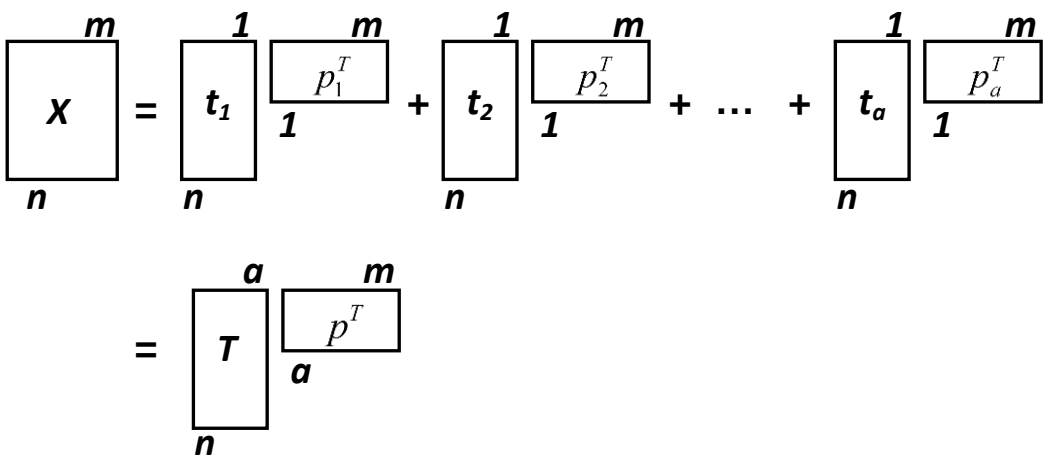

Figure 3.3 The Structure of PCA (Geladi & Kowalski, 1986)

When applying PCA to industrial process data, there is an expectation that since the original variables are highly correlated, the variance of the lower principal components (PCs) or latent variables will be so low as to be negligible. The PCA approach therefore can be described as:

$$X_{n\times m} = T_{n\times A}P^T_{A\times m} + E_{n\times m}. \tag{3.9}$$

This is equivalent to a reduction of the *m*-dimensional variable space to *A*-dimensional space. (*A* is the number of Principal Components). The matrix *T* contains orthogonal column vectors, also called score vectors, which represent the latent variables. The row of the matrix P^T is the loading of these latent variables and can be regarded as the co-variances between the measured variable and a latent variable. The matrix *E* contains the residuals; that is, all the variance in *X* not explained by the retained eigenvectors (Geladi & Kowalski, 1986; Lipp, 1996).

PCA is used as multivariate statistical technique for dimensionality reduction. There are some ways to attain this; the most common approach is the Nonlinear Iterative Partial Least Squares (NIPALS) algorithm (Wold, 1987; Martens & Naes, 1989). The NIPALS algorithm is a fast and effective algorithm to extract the principal components in a sequential manner.

The NIPALS algorithm to perform PCA is as follows:

(1) $a = 0, t_a = 0, P_a^T = 0, E_{a-1} = 0$;

(2) $a = a + 1$;

(3) The column vector X_i with the maximum variance is selected from the E_{a-1} matrix and define to be by t_a;

(4) $P_a^T = t_a^T E_{a-1}/(t_a^T t_a)$;

(5) Normalise P_a^T to length 1: $P_a^T = P_a^T/\|P_a^T\|$;

(6) $t_{a,new} = E_{a-1}P_a/(P_a^T P_a)$;

(7) If the score t_a from step (6) converges, then go to step (8); otherwise return to step (4);

(8) $E_a = E_{a-1} - t_a \cdot P_a^T$;

(9) Go to step (2) until all principal factors are calculated.

As a convergence criterion, in step (7), the sum of squared differences is frequently used:

$\sum_{a=1}^{n}(t_{a,new} - t_a)^2 \leq e$. ($e$ is pre-defined threshold, e.g. 10^{-8})

3.1.3 Principal Component Regression

PCA is suitable for process monitoring, however in Chapter 2, some of its limitations were identified. To overcome these limitations, the PCR method is proposed.

The benefits of PCR is that the multi-collinearity problem can be overcome, when two or more of the explanatory variables are close to being collinear. PCR can deal with such situations by excluding some of the low-variance principal components in the regression step. Additionally, by usually regressing on only a subset of all the

principal components, PCR can obtain dimension reduction through substantially lowering the effective number of parameters characterizing the underlying model.

PCR is an extension of PCA applied to the modelling of *Y* data from *X* data. This method is usually divided into two steps. The first step is to perform a PCA on the *X*-data which builds a set of scores for each measurement vector. If x_j is the j_{th} vector of the *K* measurements at time *j*, t_j is the corresponding j_{th} vector of the *A* scores. *Y* data are regressed on the matrix of scores by:

$$Y = TQ + E_y \tag{3.10}$$

Using the orthogonality of matrix of eigenvectors, *P* and the above equation:

$$X = TP^T \quad \text{(in the PCA parts)} \tag{3.11}$$

T can be associated with the *X* matrix data through Eqn. 3.10 (Figure 3.4).

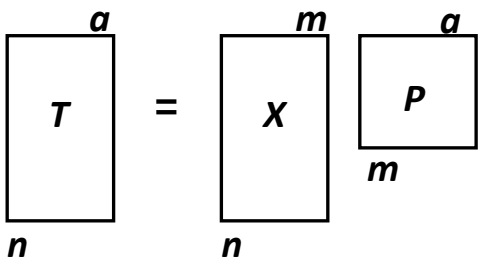

Figure 3.4 The Application of *X* matrix data to Calculate *T* (Geladi & Kowalski, 1986)

Equation 3.11 is substituted into Equation 3.10, then

$$Y = XPQ + E_y \tag{3.12}$$

Or $\quad B = PQ$ (3.13)

The structure of PCR is displayed in the Figure 3.5, where *B* is the principal component regression coefficient of *X* onto *Y*, *Q* is loading matrix, E_y is residual matrix.

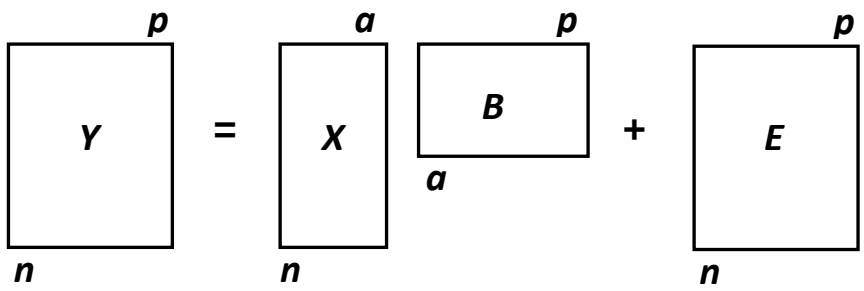

Figure 3.5 The Structure of PCR (Jackson, 1991)

The limitations of PCR were discussed in Chapter 2. In the PCR method, principal components are obtained from the Eigen-decomposition of *X* that involves the observations for the explanatory variables only. The resulting PCR estimator obtained from using these principal components as covariates may not therefore necessarily have a satisfactory predictive performance of the outcome.

Partial Least Squares Regression (PLS) can solve PCR's problem. The algorithm used examines both *X*-data and *Y*-data, and extracts factors (called components or latent variables) which are directly relevant to both sets of variables.

3.1.4 Partial Least Squares Regression

PLS known as Projections onto Latent Structures or Partial Least Squares, was proposed by Wold (Wold et al., 1984) as a regression tool that could be applied to ill-conditioned data sets. It can be considered to be a more robust alternative to classical multiple linear regression. PLS is a projection method that models the relationship between a response matrix, *Y* and a predictor matrix, *X*. These matrices are decomposed as follows:

$$X = \sum_{a=1}^{A} t_a \, p_a^T + E \tag{3.14}$$

$$Y = \sum_{a=1}^{A} u_a \, q_a^T + F \tag{3.15}$$

These equations can also be expressed as:

$$X = TP^T + \mathrm{E} \tag{3.16}$$

$$Y = UQ^T + F \tag{3.17}$$

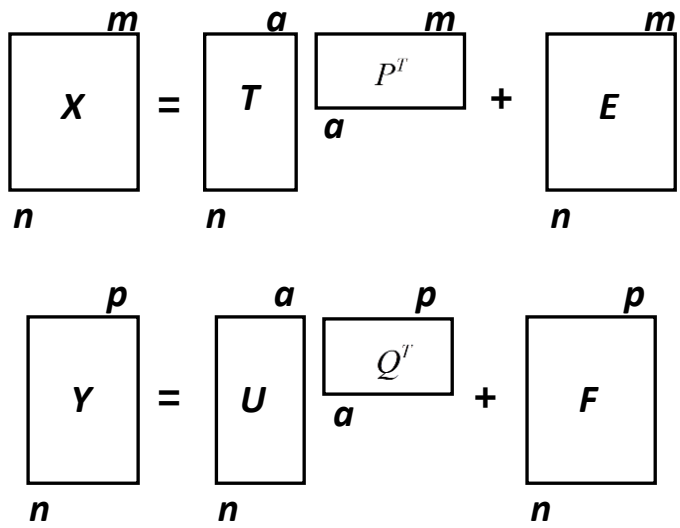

Figure 3.6 The Structure of PLS (Wold, 1987)

In Figure 3.6, the structure of PLS is shown, where *X* is a data matrix of independent variables, *Y* is a data matrix of dependent variables, *T* and *U* are the score matrices, *P* and *Q* are the loading matrices, and *E* and *F* are the residual matrices for *X* and *Y* respectively. In the PLS model, the original descriptors are transformed to a new variable space, based on a small number of orthogonal factors (latent variables). The number of latent variables that are retained in the model, *A*, is determined by cross-validation (cross validation will be introduced in the next part).

PCR is performed by first computing a principal component (factor) analysis, followed by a linear regression of the target value to the factors. In PLS, an iterative approach is used for the determination of as much variance as possible in the target variable by each component (factor) computed (Lipp, 1996).

The x-scores t_a are linear combinations of the independent variables (in the first PLS latent variable) or *X*-residual matrix (X_a) (in the a_{th} latent variable):

$$t_a = X_{a-1} W_a \tag{3.18}$$

$$X_a = X_{a-1} - t_a p_a^t \tag{3.19}$$

$W_a$being the weight vector for the a_{th} latent variable.

PLS is performed in a way to maximize the covariance between *T* and *U*, both related by the inner relationship:

$$U = TB + H \tag{3.20}$$

Where *B* is a diagonal matrix and *H* is a residual matrix. This allows PLS to be expressed as a predictive model:

$$Y = TBQ^T + F \tag{3.21}$$

$$Y = XW(P^TW)^{-1}BQ^T + F \tag{3.22}$$

Where *F* is a residual matrix.

PLS builds the regression relationship in a step wise and sequential manner. The most popular method to calculated the PLS is through the Nonlinear Iterative Partial Least Squares (NIPALS) algorithm (Wold, 1966). For each latent variable, the NIPALS algorithm calculates two latent vectors; $t_i$and u_i , which are a linear combination of the Predictor (*X*) and response (*Y*) data set, respectively.

The NIPALS algorithm to perform PLS is as following:

(1) Mean-centre and scale the *X* and *Y* data sets;

(2) Set *u* equal to any column of *Y* data set;

(3) Regress the columns of *X* on *u*: $w^t = u^TX/(u^Tu)$;

(4) Normalise the *w* vector to unit length;

(5) Calculate the score of *X*:$t = Xw/(w^Tw)$;

(6) Regress the columns of *Y* on *t* : $q^T = t^TY/(t^Tt)$;

(7) Calculate the new score of *Y*: $u_{new} = Yq/(q^Tq)$;

(8) If score *u* in step (7) converges, then go to step (9); otherwise, return to step (3);

(9) Calculate the loading of *X* by regressing columns of *X* on *t*: $p^T = t^TX/(t^Tt)$;

(10) Calculate the residual matrices *E* and *F*: $E = X - tp^T, F = Y - tq^T$;

(11) To calculate an additional latent dimension, replace X and Y by E and F, and repeat steps (2) to (10).

(12) if $X < 10e - 10\ or\ Y < 10e - 10$, the algorithm is stop.

As a convergence criterion, in step (8), the sum squared differences is frequently used:

$\sum_{a=1}^{n}(u_{new} - u)^2 \leq e$ (e is pre-defined threshold).

To compare with the capabilities of PCR and PLS, PLS and PCR are applied to one sample case. In this case, the data set contains near infrared (NIR) spectra of 60 samples of gasoline at 40 wavelengths, and their octane rating. Samples were measured using diffuse reflectance as log $(1/R)$ from 900 to 1700nm in 2nm intervals. More details regarding these data can be found in Kalivas' research (1996). X data is used as NIR spectra and Y data is used as the octane rating. PCR and PLS are applied; the latent variables are selected as 10 in the PLS model, and the PCs are selected as 10 in the PCR model. The results are shown in Figure 3.7.

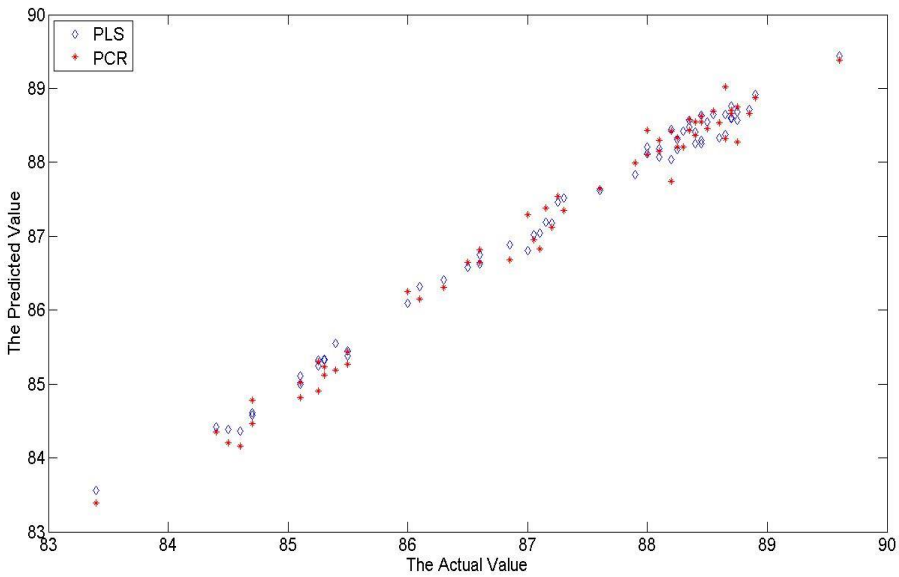

Figure 3.7 The Predicted Value of the Test System by PCR and PLS Model

In Figure 3.7, blue diamonds are the predicted value by PLS, and the red stars are the predicted value by PCR. The sum of square error (SSE) is calculated over the testing

data sets. The SSE for PCR is 2.8748 and the SSE for PLS is 1.0464. The results showed that PLS can provide better accuracy in predictions than PCR and MLR.

3.2 Cross Validation

If the underlying model for the relationship between X and Y is a linear model, the number of components needed to describe this model is equal to the model dimensionality. Nonlinear models require extra components to describe nonlinearities. The number of components to be used is a very important property of a PLS or PCA model (Geladi & Kowalski, 1986).

Although it is possible to calculate as many PLS components as the rank of the X block matrix, not all of them are normally used. The main reasons for this are that the measured data are never noise-free, and some of the smaller components will only describe noise (Geladi & Kowalski, 1986).

PLS, like any data modelling paradigm, may under-fit or over-fit the data. By under-fitting, not enough loadings are used, and the model fails to capture some of the information and dynamics. By over-fitting, too many loadings are used, and the model tends to fit some of the noise. This would cause a decrease in the precision in prediction. Both cases produce sub-optimal models, thus it is necessary to determine the number of components which fit the model best (MacGregor et al., 1999). Cross-validation is an effective and popular approach used to determine the number of latent variables or PCs in the PLS or PCA model (Piovoso & Kosanovich, 1996; Kjeldahl et al., 2008).

The basic principal of cross-validation is to leave out part of the data, build a model, and then predict the left-out samples. The concept of cross-validation was initially proposed by Mosier (1951), as a ‘design’ for assessing the effectiveness of model weights. In 1956, Wold laid the foundations for principal component analysis (PCA) cross-validation, a method used to identify the dimensions that best describe the systematic variations in the data. The cross-validation method for PCA proposed by Wold (1978) relies on the special property of the NIPALS algorithm to cope with a moderate amount of randomly missing data. Wold’s cross-validation scheme

provides both a way to calculate prediction error sum of squares (PRESS), by a specific leave-out pattern and a criterion for the selection of a number of components.

In 1982, Eastment and Krzanowski suggested an alternative approach that could be used to choose a feasible number of components in PCA; however their cross-validation scheme is very complex. In this method, 2 PCA models need to be combined. The mean PRESS difference between the actual and the predicted value is calculated, and then the number of components is determined by comparing the mean PRESS of the 2 PCA models. The Eigenvector approach is another alternative method (Wise et al., 1991), where PCA models are calculated with one or several samples left out, and then the model is used to predict estimates of the left-out samples. In contrast to the other methods, the PRESS values estimated with Eigenvector's method are actually independent from the predicted elements.

These cross-validation techniques mentioned however have two significant problems; either over-fitting is introduced as the model with which left-out elements are predicted is not independent of the left-out elements, or an unintended additional error is introduced because the rationale behind the method is not correct.

Given this, an alternative method has been proposed - cross-validation based on an improved Wold procedure (Wise et al., 2003). This method can be briefly described as follows: when cross-validation is applied to the latent variable of the PLS model, X data and Y data are divided into several groups. Using one of the groups, the PLS model is generated as the numbers of latent variables varies from 1 to A (A is the number of latent variable). Each of these models is used to predict the Y data in the group withhold. The prediction error sum of squares (PRESS) is computed for each model. This routine is repeated until each group is withheld once and only once. The overall PRESS is then generated for a given number of loadings (from 1 to A), by summing the prediction errors for all withheld data. A plot of the PRESS vs. loading number (latent variables) will typically reach a minimum and then start to increase again. The value corresponding to the minimum PRESS is taken as the number of loadings required. Having fewer than this number tends to under-fit the data, whereas having more, it begins to over-fit the data (Piovoso & Kosanovich, 1996).

To demonstrate its capabilities, cross-validation is applied to one sample case. This sample case was introduced in Section 3.1.4. The data set contains near infrared (NIR) spectra and their octane rating. *X* data is used as NIR spectra and *Y* data is used as the octane rating. From 1 latent variable to 20 latent variables, the prediction error sum of squares (PRESS) are calculated and presented in Figure 3.8. Cross validation was used to determine the number of latent variables, which was found to be 7 in this example.

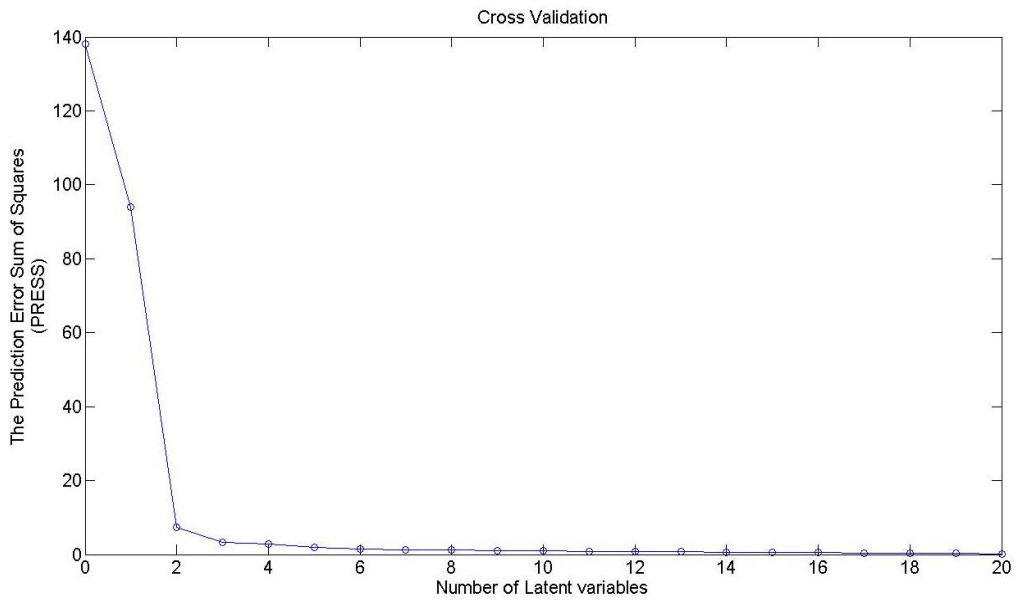

Figure 3.8 The Application of Cross-Validation

3.3 Multi-way PLS (MPLS) Analysis of Batch Data

To enable PLS to track the dynamics of batch processes, multi-way PLS has been proposed. MPLS is an extension of PLS that enables it to handle 3-dimensional data arrays (Nomikos. & MacGregor, 1994). Measurement data from a batch process is typically stored as a 3-dimensional matrix (X) of size $I \times J \times K$, where I is the number of batches, J is the number of measured observations in a complete batch, and K is the number of measured variables. The structure of the data collected from a batch process is a 3-dimensional cube. If MSPC is to be applied, 3-dimensional data must be transformed into a 2-dimensional matrix. There are different approaches for rearranging the data sets (Golshan & MacGregor, 2010).

The relationship between MPLS and PLS is that MPLS is equivalent to performing ordinary PLS on a 2-dimensional matrix *X*', formed by unfolding the 3-dimensional array *X* (Nomikos & MacGregor, 1994; Louwerse & Smilde, 2000). MPLS is a method successfully applied to batch-process monitoring and endpoint quality prediction.

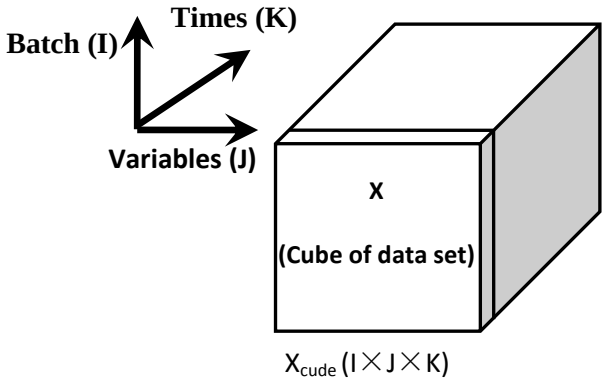

Figure 3.9 The Structure of the Data Collected from a Batch Process (Nomikos & Macgregor, 1994)

MSPC algorithms have been developed for application in a 2-dimensional matrix; therefore these algorithms want to be applied in the batch data, and the batch data needs to be transformed into a 2-dimensional matrix by using some unfolding techniques. There are various ways to unfold a 3-dimensional matrix. Batch-wise unfolding and variable-wise unfolding are two alternative methods (Nomikos & MacGregor, 1994; Wold et al., 1998).

3.3.1. Batch-wise Unfolding

Batch-wise unfolding, or the B-approach, unfolds the matrix in accordance to the direction of the batches. Measurement data from a batch process is usually 3-dimensional. Assuming this, this 3-dimensional matrix is $I \times J \times K$.

The batch-wise unfolding method can be divided into two steps. The first step is to transform the 3-dimensional matrix ($I \times J \times K$) into a two-dimensional matrix ($I \times JK$).

Each row of the new matrix represents a batch, which is inclusive of all measurement data. The second step is to scale this 2-dimensional matrix, with the aim being to remove the non-stationary trajectories from the process data. In this way, the mean is 0 and the variance is a unit in each column of the matrix. This procedure is explained in Figure 3.10. The method of scaling removes the non-stationary trajectories from the process data.

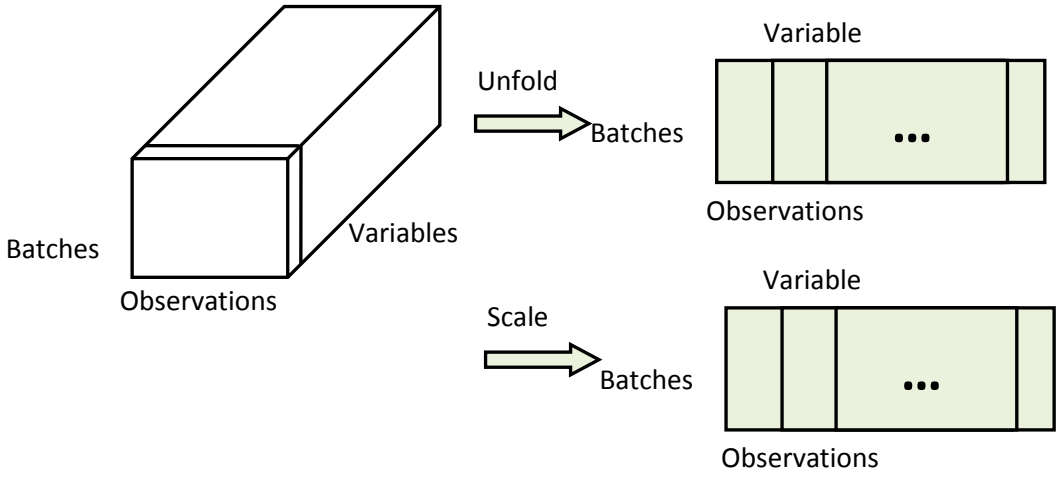

Figure 3.10 Procedure of B-approach (Wu and Lennox, 2006)

The application of linear MSPC methods to batch processes faces a number of problems. The most significant problem is that process data tends to be highly nonlinear and non-stationary. By unfolding in the B-approach, the batch data is transformed into a 2-dimensional matrix; through scaling in the B-approach, the major non-stationary dynamics are removed. Standard linear MSPC methods can be applied to this 2-dimensional transformed data.

3.3.2 Variable-wise Unfolding

Unlike batch-wise unfolding, variable–wise unfolding or the V-approach is a method that unfolds the matrix in accordance with the direction of variables. The variable-wise unfolding method can be divided into four steps; the first two steps are optional, and are identical to the batch-wise unfolding method. The third step is to refold the

scaled 2-dimensional matrix back into the original 3-dimensional matrix (i.e. the 3-dimensional matrix of data has become the scaled data). Finally, the 3-dimensional matrix ($I \times J \times K$) is unfolded into a 2-dimensional matrix ($IJ \times K$) along the direction of variables. Figure 3.11 provides a description of variable-wise unfolding method (Wu & Lennox, 2006).

The principal difference between the V-approach and B-approach is in the unfolding process of batch data matrix before MSPC algorithms are applied.

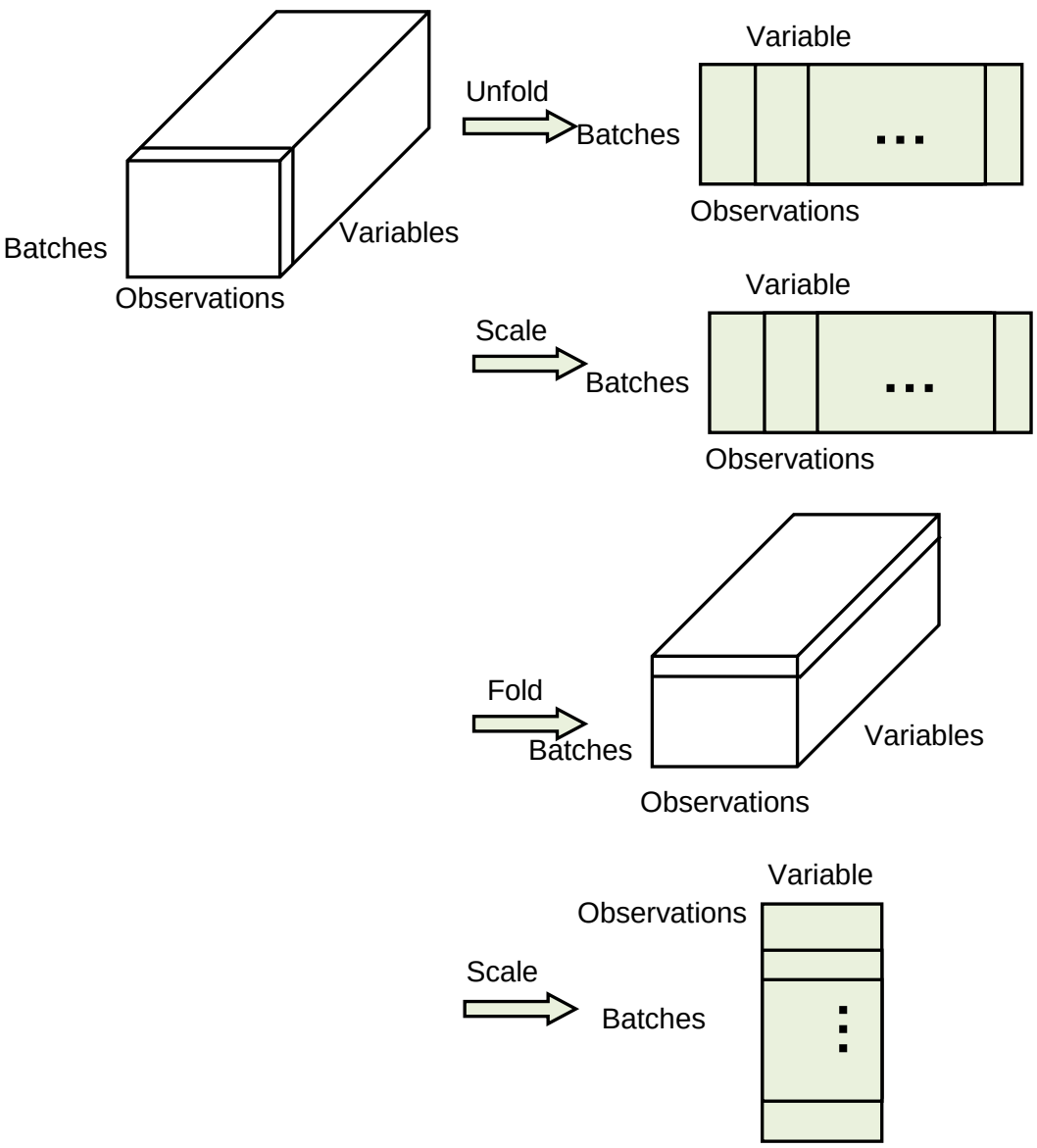

Figure 3.11 Procedure of V-approach (Wu & Lennox, 2006)

To compare the capabilities of the B-approach and the V-approach, MPLS based on these two unfolding methods are applied to detect the abnormal conditions by Wu & Lennox (2006). Both MPLS models are able to detect the abnormal condition and identify the cause of the abnormality; however the B-approach MPLS model has greater sensitivity than the V-approach, when the models are applied to detect small deviations (Wu & Lennox, 2006). The B-approach model though can capture more of the information in the normal operation data (Westerhuis et al., 1999).

In recent years, the B-approach MPLS could be integrated within a model predictive controller (MPC) and applied to a fed-batch formation process (Zhang & Lennox, 2003). The B-approach has been widely applied to process monitoring of fed batch formation processes, whilst it has been proven to be very successful by Lennox et al. (2001).

When the unfold method is applied to the batchwise unfold method. Performing PLS on the batchwise unfolded data of these two sets also results in a reduced dimension latent variable model of the form:

$$X = TP^T + E \tag{3.16}$$

$$Y = UQ^T + F \tag{3.17}$$

where X is the batchwise unfolded matrix of $I \times JK$ for cause variables, Y is the batchwise unfolded matrix of $I \times L$ for effect variables, P of $JK \times A$ and Q of $L \times A$ are the loading matrices for X and Y, respectively. The scores T and U are related by a diagonal matrix B of proper dimensions with $U = TB, T = XW$, where W is the weight matrix. Finally, E and F are residual matrices.

3.4 Nonlinear PLS model

To improve the modelling capabilities of PLS, several nonlinear extensions have been proposed to enable it to better handle nonlinear systems. These methods can be divided into two categories: Type I and Type II Nonlinear PLS methods.

3.4.1 Type I Nonlinear PLS

In the Type I Nonlinear PLS method, the observed variables are appended with nonlinear transformations, such as the X matrix is transformed into the X^2 matrix. Following this, traditional linear PLS is then applied. For example, the X matrix can be augmented with transformed terms. The addition of transformed terms in X within PLS models was firstly proposed by Wold (1989), which proposed the use of quadratic terms in the PLS model. Other studies involving this technique have utilised quadratic and higher order polynomial terms, while ignoring cross-terms (Berglund et al., 1997; 1999).

3.4.2 Type II Nonlinear PLS

In contrast to the Type I nonlinear PLS method, the Type II nonlinear PLS method assumes a nonlinear relationship within the latent variable structure of the model. The Type II Nonlinear PLS model was first proposed by Wold et al. (1989) and has been shown to be able to provide an accurate fit to more complex nonlinear relationships, in comparison to Type I Nonlinear PLS (Hiden et al., 1998).

Type II nonlinear PLS models were constructed using the technique proposed by Baffi et al. (2000). The principle of this algorithm is as follows:

In traditional PLS, the inner relation between t and u is defined as follows, where t and u are score vectors and h denotes residuals:

$$U = bt + h \tag{3.23}$$

In the algorithm proposed by Baffi et al. (2000), the inner relation is replaced by a quadratic polynomial (2nd order):

$$u = c_0 + c_1 t + c_2 t^2 + h \tag{3.24}$$

A limitation with this approach is that by choosing a second order polynomial, the type of relationship that can be modelled is restrictive; therefore in the applications described in Chapter 5 of this thesis, higher order terms are also included. The

relationships are provided in Equation 3.25 and Equation 3.26 for 4th order and 6th order polynomials respectively.

$$u = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + h \tag{3.25}$$

$$u = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + c_5 t^5 + c_6 t^6 + h \tag{3.26}$$

3.5 Neural Network PLS (NNPLS)

An alternative to using polynomials in the inner relationship of the PLS model is to use a neural network to describe this relationship. In this case, the PLS inner model can be represented as follows:

$$U_a = N(t_a) + r_a \tag{3.27}$$

Where $N(\cdot)$ denotes the nonlinear relation represented by a neural network, which is determined by minimizing the residual r_a (Qin et al., 1992).

The structure of the NNPLS method is showed in Figure 3.12. In the NNPLS approach, the PLS outer model is used to transform the data to score variable (u_a and t_a). The neural networks are then applied to learn the score. These networks can be recurrent networks (William & Zipser, 1989), Error Back Propagation network (Rumelhart et al., 1986), radial basis functions (Poggio & Girosi, 1990a; 1990b) or multilayer feed-forward network (Svozil et al., 1997) and so on.

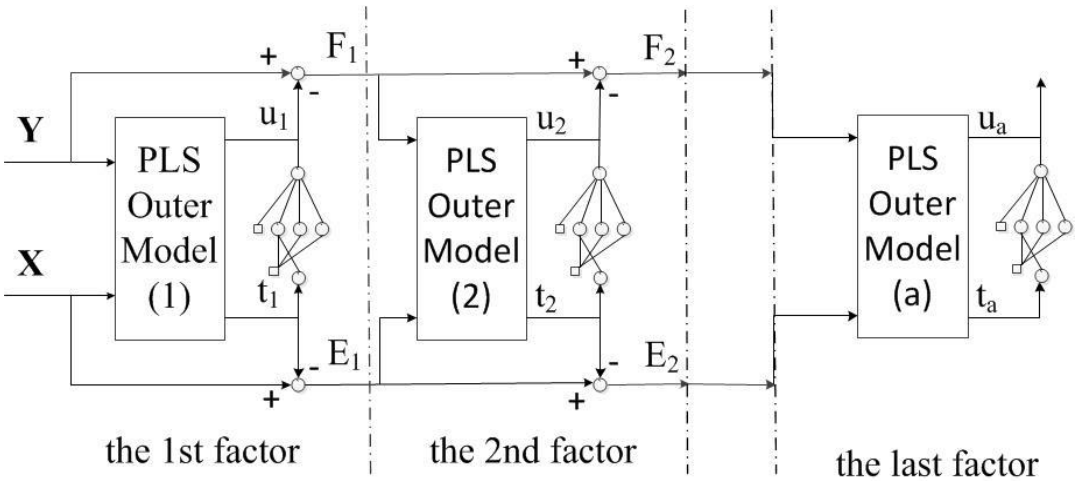

Figure 3.12 The Structure of NNPLS (Qin et al., 1992)

The NNPLS approach differs from the direct network method. The differences were introduced in Chapter 2. The advantage of the application of neural network in the inner regressors is due to their nonlinear approximation property. As the simplicity and universal approximation property of the neural network would be considered, when a nonlinear PLS method wants to be applied, a nonlinear inner model needs to be built. The type of network has one sigmoidal hidden-layer and one linear output-layer.

The NNPLS method can be implemented based on the structure of the NNPLS method (Figure 3.12). It has 2 parts; one is the PLS outer transform model, and the other is an inner network training algorithm. The PLS outer transform model is the same as the linear PLS method.

The NNPLS method can be formulated as follows (Qin et al., 1992):

(1) Scale *X* and *Y* data, let $E_0 = X, F_0 = Y \; and \; a = 1$.

(2) For each factor a, take $u_a = y_p$

(3) PLS outer transform model:

In matrix *X*:

$w_a^T = arg_{w_a} min \|E_{a-1} - u_a w_a^T\|^2 = u_a^T E_{a-1} / u_a^T u_a$, normalized $w_a$to norm 1.

$t_a = arg_{t_a} min \|E_{a-1} - t_a w_a^T\|^2 = E_{a-1} w_a$;

In matrix *Y*:

$q_a^T = arg_{q_a} min \|F_{a-1} - t_a q_a^T\|^2 = t_a^T F_{a-1} / t_a^T t_a$, normalized $q_a$to norm 1.

$u_a = arg_{u_a} min \|F_{a-1} - u_a q_a^T\|^2 = F_{a-1} q_a$.

Iterate this step until it converges.

(4) Calculate the X loading and rescale the variable:

$p_a^T = arg_{p_a} min \|E_{a-1} - t_a p_a^T\|^2 = t_a^T E_{a-1} / t_a^T t_a$,

Normalized $p_a = p_a / \|p_a\|$,

$t_a = t_a \|p_a\|$,

$w_a = w_a \|p_a\|$.

(5) The inner neural network model:

The rule of training the inner neural network model is that the following error function is minimized.

$J_a = \|u_a - N(t_a)\|^2$.

A conjugate gradient training method is applied in this algorithm. The detail of the conjugate gradient training method will be described in section 3.5.2.

(6) Calculate the residual for factor a:

$E_a = E_{a-1} - t_a p_a^T$,

$F_a = F_{a-1} - \hat{u}_a q_a^T$,

Where

$\hat{u}_a = N(t_a)$.

(7) Let $a = a + 1$, return to step (2) until all principal factors are calculated.

3.5.1 Choosing Neuron Activation Function

When the network wants to be a universal approximator, many kinds of nonlinear functions can be applied in the hidden layer (Stinchombe & White, 1989). Normally, a sigmoidal function valued from 0 to 1 is selected. In this application, the relation between u_a and t_a is modelled by the neural network (Eqn. 3.26). Both u_a and t_a have the following property (Geladi et al., 1986):

$$\sum_{i=1}^{n} u_{ai} = 0, \tag{3.28}$$

$$\sum_{i=1}^{n} t_{ai} = 0, \tag{3.29}$$

where u_{ai} and t_{ai} are the ith elements of u_a and t_a. Therefore, the following centred sigmoid is chosen to model the inner relation:

$$\sigma(z) = \frac{1-e^{-z}}{1+e^{-z}}, \tag{3.30}$$

Note that the derivative of the centred sigmoid is:

$$\sigma'(z) = \frac{1}{2}[1 - \sigma^2(z)]. \tag{3.31}$$

These relations are useful in deriving the learning rules for neural networks with centred sigmoidal functions.

3.5.2 Choosing a Learning Algorithm

The neural network used in the NNPLS method can be trained by the generalized delta learning rule (Rumehart et al., 1986). Therefore, if the generalized learning method is applied, the learn rate for each network needs to be appointed. In this thesis, the conjugate gradient learning method (Lasdon et al., 1967; Leonard & Kramer, 1990) is used to train the network. The conjugate gradient method was firstly introduced by Fletcher and Powell (1963), and applied to train a feed-forward network by Leonard and Kramer (1990).

There are two reasons for the selection of the conjugate gradient learning method (Qin, 1992). Firstly, the learn speed of the conjugate gradient learn method is faster than back-propagation. Secondly, the learning rate constants are calculated automatically and adaptively in the conjugate gradient learning method, so that they do not need to be specified before training. However, the conjugate gradient learning method is the most convenient for the NNPLS approach.

3.5.3 Determining the Number of Hidden Neurons

The number of hidden layers and hidden units are important factors when designing a neural network. In general, a more complex input-output relation would require

more hidden units, however this is not true. When too many hidden units are used, the over-parameterize problem will occur; whilst too few hidden units would result in an under-parameterized model. In the NNPLS model, cross-validation was used to determine the size of the hidden units of the neural networks.

The data have been divided into a training set and a testing set (Figure 3.13). The number of factors is chosen such that the model gives the minimum prediction error for the testing data. It is similar to when cross validation is applied in the linear PLS model.

$$u_a^{test} = F_{a-1}^{test} w_a \,, \tag{3.32}$$

$$t_h^{test} = E_{a-1}^{test} q_a \,, \tag{3.33}$$

and

$$E_a^{test} = E_{a-1}^{test} - t_a^{test} p_a^T, E_0^{test} = X^{test}, \tag{3.34}$$

$$F_a^{test} = F_{a-1}^{test} - N(t_a^{test}) q_a^T, F_0^{test} = Y^{test}, \tag{3.35}$$

where w_a,p_a and q_a have been determined in the NNPLS method.

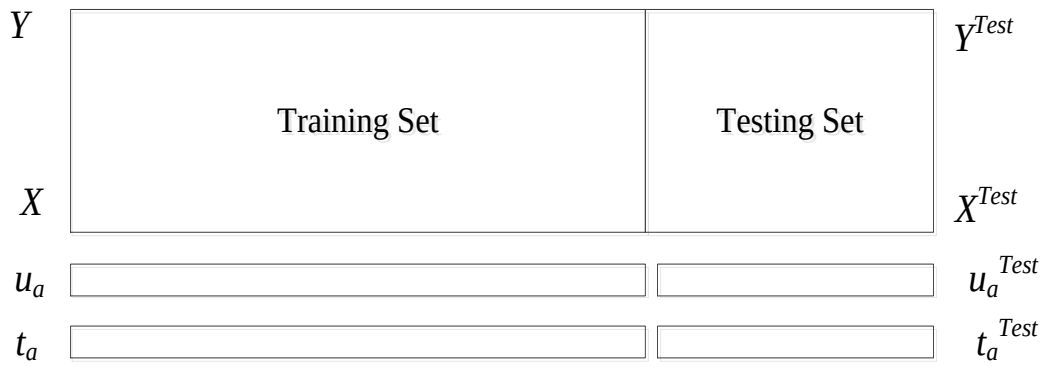

Figure 3.13 The Application of Crossing Validation in the Training Set and Testing Set of NNPLS Method (Qin et al., 1992)

The score for the training set , u_a and t_a , has already been generated in the NNPLS method. The inner neural network starts to train the testing data from one hidden unit.

The optimal number of hidden units can be found, which gives the best prediction error for the test pairs: $u_a^{test} and\ t_a^{test}$. The method used is cross validation.

3.6 Summary

This chapter has provided a mathematical overview of some MSPC methods, such as MLR, PCA, PCR and PLS. Some related algorithms, such as unfolding approaches and cross-validation, have been discussed in this chapter. Additionally, introductions to Nonlinear PLS and NNPLS have been given, with the property of NNPLS being discussed. In the next chapter, to compare the capabilities and limitations of linear PLS, Nonlinear PLS and NNPLS, these methods will be applied to a number of sample testing systems.

Chapter 4

Application of Linear and Nonlinear PLS Modelling Techniques to Numerical System

A number of MSPC methods were introduced in the previous chapter. In this chapter, linear MPLS, Type I and II nonlinear PLS and NNPLS will be applied to a test focused upon batch application. Linear MPLS is first applied to predict linear and nonlinear systems; the limitations and capabilities of linear MPLS are then discussed, based on the analysis of the predicted results. The results show that linear MPLS is suitable for modelling linear systems, but it cannot predict nonlinear systems accurately. In real industry, the industrial process is almost nonlinear.

To overcome this deficiency, several nonlinear extensions have been proposed to enable it to better handle nonlinear systems. The application of Type I and II nonlinear PLS models is discussed with their limitations considered. The inner relation of the Type II nonlinear PLS model is commonly a quadratic polynomial (2nd order). To highlight the limitation of Type II nonlinear PLS model, the higher order terms (4th and 6th) of Type II nonlinear PLS model are used in this chapter. Finally, to illustrate the capabilities of the NNPLS method, NNPLS and the Type II nonlinear PLS model are applied to predict the same testing system. The Chapter concludes with a discussion of results analysis.

The chapter is divided into the following sections:

4.1) introduces the application of linear MPLS to simple linear and nonlinear systems, and the limitations and capabilities of linear MPLS illustrates;

4.2) describes the application of the nonlinear PLS model, and discusses the benefits and limitations of Type I and Type II nonlinear PLS models;

4.3) discusses the application of the NNPLS method; and

4.4) summarizes and concludes this chapter.

4.1 Application of Linear MPLS to Example Systems

Linear PLS has been widely applied to solve many practical problems. The major restriction of linear PLS is that only linear information can be extracted from data; however, many industrial data are inherently nonlinear. To discuss the limitations of linear PLS, PLS is applied to predict both linear and nonlinear systems. The limitation of Linear PLS model is summarized. Then a number of alternative methods are applied, such as Type I and II nonlinear PLS and NNPLS.

In this part, the testing system is composed of 4 simple systems, to which linear MPLS was applied to demonstrate its capabilities. These systems were Linear time invariant (system 1); Nonlinear time invariant (system 2); Linear time varying (system 3); and Nonlinear time varying (system 4).

The linear time invariant system was defined as:

$$y(t) = 0.3x_1(t) - 0.2x_2(t) + 0.1y(t-1) \tag{4.1}$$

The linear time varying system was defined as:

$$y(t) = 0.5t \cdot x_1(t) + 0.7t^{0.6} \cdot x_2(t) + 0.1y(t-1) \tag{4.2}$$

The nonlinear time invariant system was defined as:

$$y(t) = 0.9x_1^2(t) - 0.6x_1(t) + 0.2x_2^2(t) - 0.4x_2(t) + 0.1y(t-1) \tag{4.3}$$

The nonlinear time varying system was defined as:

$$y(t) = 1.2t \cdot x_1^2(t) - 0.8t^{0.3} \cdot x_1(t) + 0.9t^{0.5}x_2^2(t) - 0.7t^{0.6} \cdot x_2(t) + 0.1y(t-1) \tag{4.4}$$

In each system, x_1 and x_2 were specified to be equal to a PRBS signal with an amplitude between -1 and 1, and switching time of 1 sample. The initial value of y

was 0. White noise with a standard deviation of 0.08 and 0.06, was added to the measurements of x_1 and x_2 respectively. Each of the systems was considered to operate as a batch, with each batch containing 50 samples. For each system, 100 batches of data were collected for training the models, and 20 batches were collected for testing purposes. Cross validation was used to determine the number of latent variables, which in each case was found to be 40. The model is applied to predict the trajectory values during batch progression. The model input is the entire batch of x_1 and x_2. The accuracy of the models was measured using the sum of square error (SSE); this was calculated over 20 batches testing data sets. Linear PLS model is applied to the linear system and nonlinear system. In Figure 4.1, 4.1(a) is presented the linear time invariant system, 4.1(b) is presented linear time varying system. The prediction results of one testing data set are displayed in Figures 4.1 and 4.2. In Figure 4.1, 4.1(a) is presented that linear PLS model is applied to predict linear time invariant system, 4.1(b) is presented that linear PLS model is applied to predict linear time varying system. In Figure 4.2, 4.1(a) is presented that linear PLS model is applied to predict nonlinear time invariant system, 4.1(b) is presented that linear PLS model is applied to predict nonlinear time varying system. In these figures, the red line is the predicted value for the output, *y*, and the blue line is the actual value.

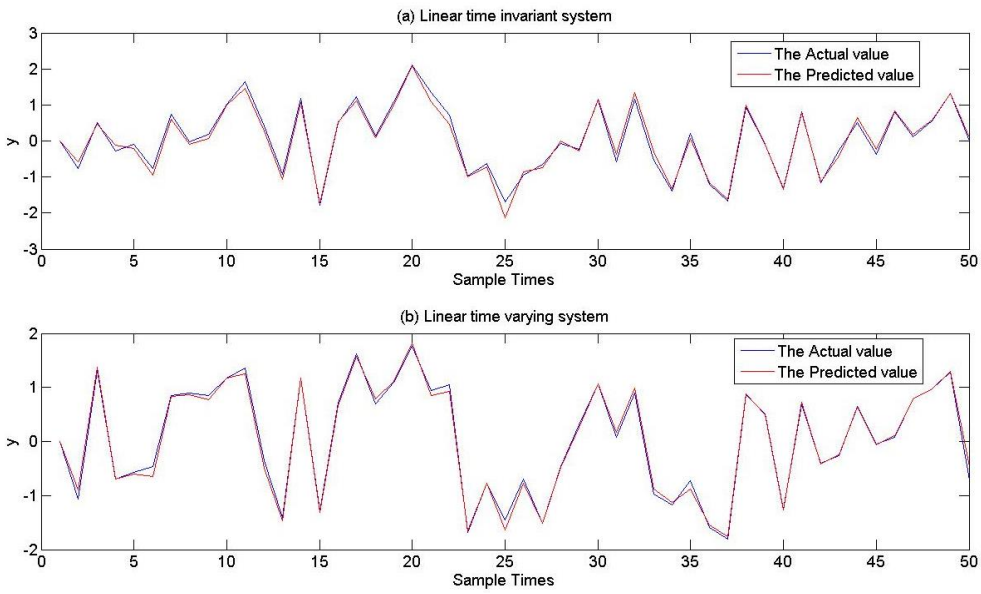

Figure 4.1 Linear MPLS Model Prediction in the Linear System

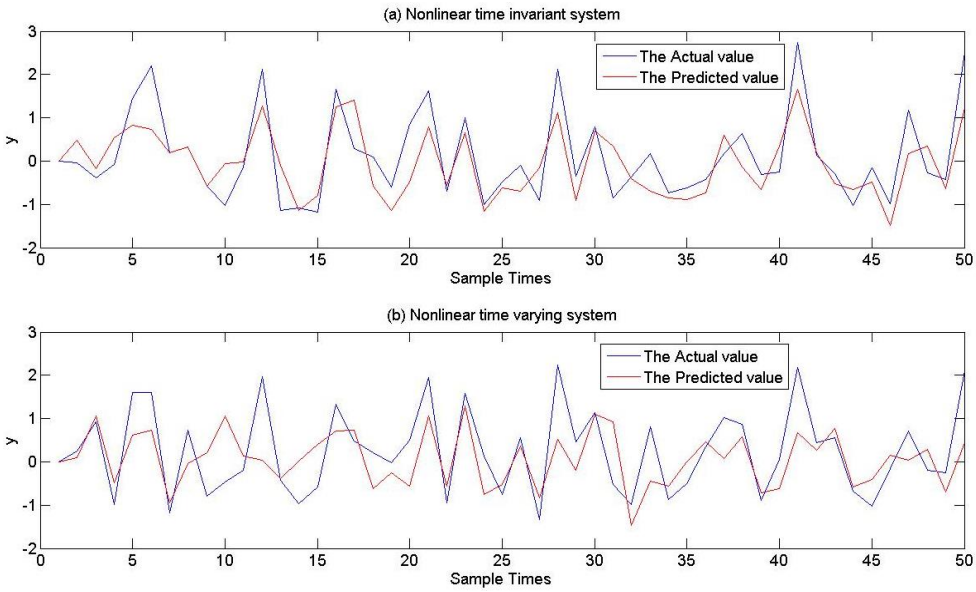

Figure 4.2 Linear MPLS Model Prediction in the Nonlinear System

In Figure 4.1, the linear MPLS appears able to approximate the output of both the linear time invariant and the linear time varying systems with high accuracy. The average SSE of 20 batches testing data set for these two systems was 0.8533 and 0.3221, respectively. However, Figure 4.2 shows that the MPLS model was not as accurate when used to predict the output of the nonlinear time invariant and nonlinear time varying systems. The average SSE in these two cases was 23.6 and 31.8, respectively.

The results show that as might be expected, linear MPLS can predict the linear systems very well. However, this algorithm was not able to track the dynamics contained in the two non-linear systems. These systems demonstrate that linear MPLS is suitable for modelling linear, time-varying systems.

4.2 Nonlinear Multiway PLS Model

Linear MPLS can predict linear systems very well; however problems may be encountered when this algorithm is used to model non-linear systems. To illustrate the capabilities of the non-linear extensions to MPLS, these algorithms were applied to the nonlinear systems, defined by Equation 4.5 to Equation 4.9.

4.2.1 Application of Type I Nonlinear MPLS to Simulated Systems

Before investigating the simulated systems introduced in Section 4.1, the limitation of using Type 1 Nonlinear MPLS is first illustrated through its application to two simple non-linear systems. These are defined as follows:

System 5: $y_1(t) = 2.5x^2(t) + 1.5x(t) + 3 + 0.1y_1(t-1)$ (4.5)

System 6: $y_2(t) = 2.5x^3(t) + 1.5x(t) + 3 + 0.1y_2(t-1)$ (4.6)

x was specified to be white noise with a mean of 0 and a standard deviation of 1. For each system, 50 batches of data were collected for training the models, and 20 batches were collected for testing purposes. Each batch contained 20 samples. For each system, a Type I Nonlinear MPLS model using second order polynomials only was used to predict the endpoint. The latent variable is selected as 20. The model input is the entire batch of x.

The accuracy of the models over the testing data is shown in Figure 4.3(a). The predicted endpoint value is seen to be very close to the actual endpoint value suggesting high accuracy. However, Figure 4.3(b) shows that in the situation where the order of the non-linearity does not match that of the process, problems are introduced and the prediction accuracy is reduced significantly. In real studies, the exact order of any nonlinear relationship will not be known a-priori, hence the required expansion of *x* will be difficult to determine. Type I MPLS is therefore not recommended for the prediction of the nonlinear system.

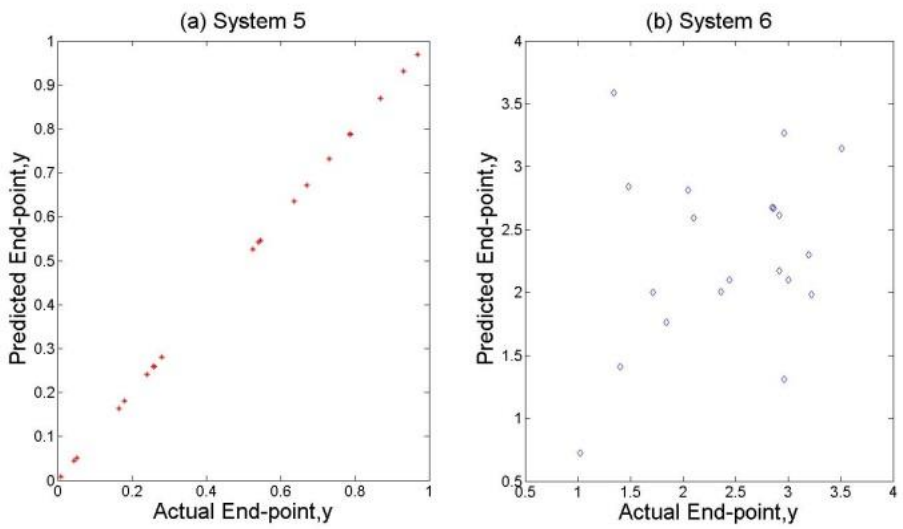

Figure 4.3 The Predicted Endpoint of the Simple Nonlinear System by Type I Nonlinear MPLS Model

4.2.2 Application of Type II Nonlinear MPLS Model to Simulated Systems

In this section, 4th order Type II Nonlinear MPLS is used to approximate *system 6*, as defined in Section 4.2.1. To provide a comparison, linear MPLS is also applied. In this part, the latent variable is selected as 20 by cross validation.

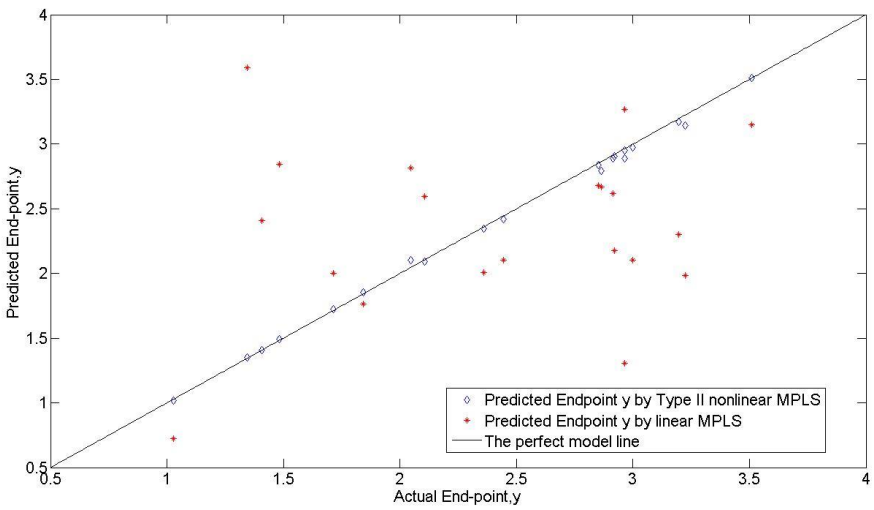

Figure 4.4 The Predicted Endpoint of the Simple Nonlinear System by Type II Nonlinear MPLS Model

In Figure 4.4, the red dots are the endpoints predicted by the Type II nonlinear MPLS model and the diamonds are endpoints predicted by linear MPLS. The accuracy of the Type II model is significantly greater than the accuracy of the linear model. In the Type II nonlinear MPLS model, the predicted endpoint value is very close to the real endpoint value. This shows that the Type II Model can be used to predict this simple nonlinear system.

To illustrate the capabilities and limitations of Type II nonlinear MPLS, the ability of this model to approximate three different modifications to Eqn.4.3 is now presented. In this section, 2nd, 4th and 6th order Type II nonlinear MPLS models are applied. To compare the capabilities and limitations of these models, the testing systems selected are 4th, 5th and 6th nonlinear systems. The results show the influence of the order selection in Type II nonlinear PLS model. The systems used for this test are defined as follows:

4th order nonlinear system:

$$y(t) = 2.2x_1^4(t) + 1.4x_1^3(t) - 1.8x_1(t) + 1.4x_2^2(t) - 0.9x_2(t) + 0.05y(t-1) \tag{4.7}$$

5th order nonlinear system:

$$y(t) = 5.4x_2^5(t) + 2.2x_1^4(t) + 1.4x_1^2(t) - 1.8x_1(t) + 1.4x_2^2(t) - 0.9x_2(t) + 0.05y(t-1) \tag{4.8}$$

6th order nonlinear system:

$$y(t) = 6.1x_1^6(t) + 5.4x_2^5(t) + 2.2x_1^4(t) + 1.4x_1^2(t) - 1.8x_1(t) + 1.4x_2^2(t) - 0.9x_2(t) + 0.05y(t-1) \tag{4.9}$$

In each system, x_1 and x_2 were specified to be equal to a PRBS signal with amplitude between -1 and 1, and switching time of 1 sample. The initial value of y was 0. White noise with a standard deviation of 0.4 and 0.2 was added to the measurements of x_1 and x_2 respectively. Each of the systems was considered to operate as a batch, with each batch containing 50 samples. For each system, 50 batches of data were collected for training the models, and 20 batches were collected

for testing purposes. The model inputs are x_1 and x_2 over the whole batch. The Type II nonlinear PLS model was applied to predict the endpoint of these testing systems. The results are shown in Figures 4.5, 4.6 and 4.7.

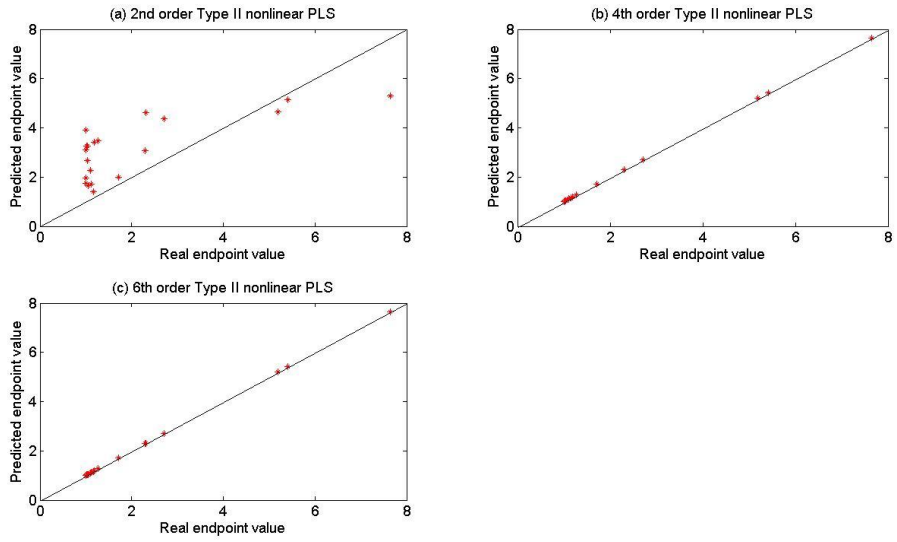

Figure 4.5 The Application of Type II nonlinear PLS Model to Test 4th Order Nonlinear System (Equation 4.7)

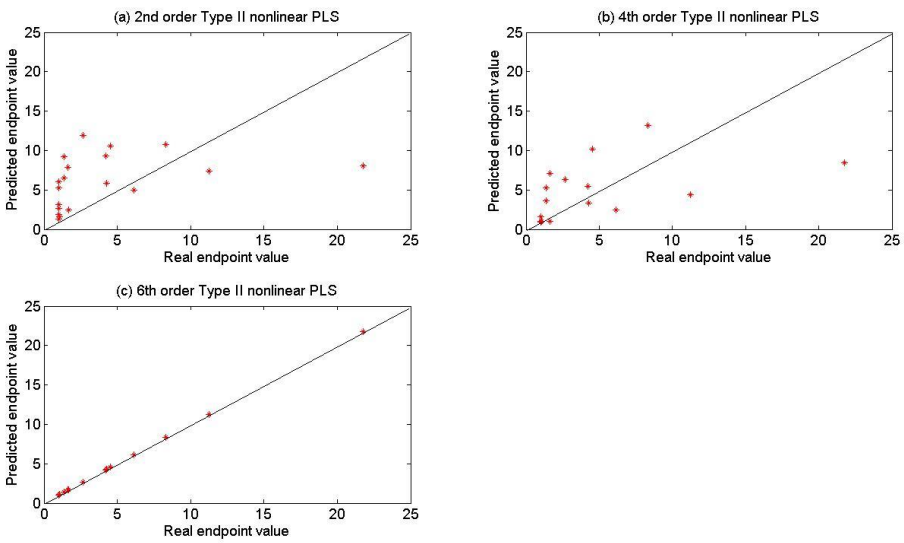

Figure 4.6 The Application of Type II nonlinear PLS Model to Test 5th Order Nonlinear System (Equation 4.8)

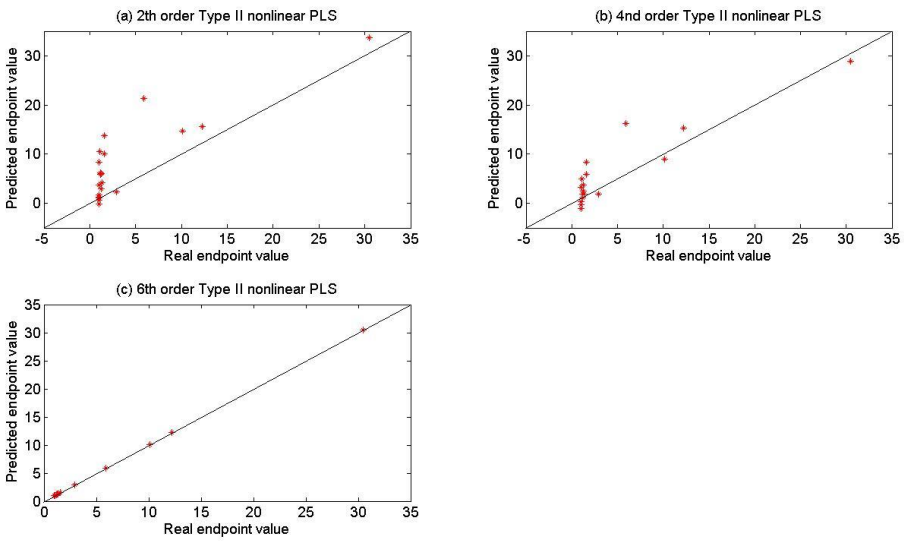

Figure 4.7 The Application of Type II Nonlinear PLS Model to Test 6th Order Nonlinear System (Equation 4.9)

Table 4.1 shows the errors that result when Type II (defined with 2nd, 4th and 6th order polynomials) nonlinear MPLS models are used to approximate each of these systems. The sum square error over the testing data sets is provided in Table 4.1.

Table 4.1 The List of SSE (when Type II nonlinear MPLS is applied in the nonlinear testing systems)

Type II nonlinear MPLS	Testing system		
	4th order nonlinear system	5th order nonlinear system	6th order nonlinear system
2nd order	254.7	541.3	1.1e+003
4th order	0.1329	314.5	363.2
6th order	0.1497	0.1789	0.1826

These results show that when the system is known, the order of the Type II nonlinear MPLS model can be precisely determined. For example, when the testing system is a 4th order nonlinear system, 4th and 6th order Type II nonlinear MPLS can predict the

end-point values very well. However, there were large errors when the order of the Type II nonlinear MPLS model was less than the order of the nonlinear system in the process. Although the Type II nonlinear PLS model can predict some nonlinear systems, the model's limitation is that the testing system needs to be known a-priori. However, most industrial systems can be pre-determined, so the NNPLS model is applied.

4.3 Application of the NNPLS Model

To illustrate its capabilities, the NNPLS method is applied to the nonlinear testing systems, defined by Equation 4.9 and Equation 4.10. To provide a comparison, 6th Type II nonlinear PLS is also applied.

6th order nonlinear system is defined as Equation 4.9 in section 4.2.2.

7th order nonlinear system is defined:

$$y(t) = 3.5x_1^7(t) + 6.1x_1^6(t) + 5.4x_2^5(t) + 2.2x_1^4(t) + 1.4x_1^2(t) - 1.8x_1(t) + 1.4x_2^2(t) - 0.9x_2(t) + 0.05y(t-1) \tag{4.10}$$

In each system, x_1 and x_2 were specified to be equal to a PRBS signal with an amplitude between -1 and 1, and switching time of 1 sample. The initial value of y was 0. White noise with a standard deviation of 0.4 and 0.2 was added to the measurements of x_1 and x_2 respectively. Each of the systems was considered to operate as a batch, with each batch containing 50 samples. For each system, 50 batches of data were collected for training the models, and 20 batches were collected for testing purposes. 6th order Type II nonlinear PLS model and NNPLS model were applied to predict the endpoint of these testing systems. In the NNPLS model, the number of LVs is selected as 20, the number of hidden layers is selected as 1. Cross validation was used to determine the number of hidden units, which in each case was found to be 5. See Figures 4.8 and 4.9 for the results.

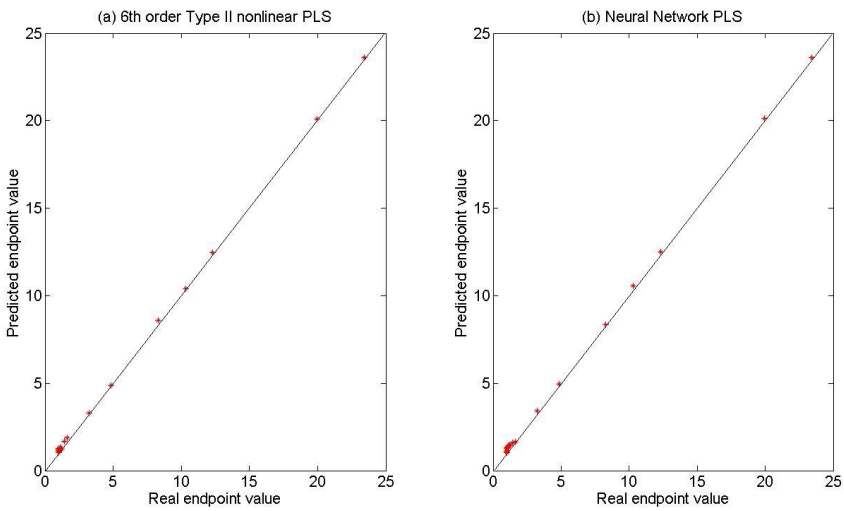

Figure 4.8 The Application of 6th Type II Nonlinear PLS and NNPLS Model to Test 6th Order Nonlinear System

The accuracy of the models over the testing data is shown in Figure 4.8; the predicted endpoint value is seen to be very close to the actual endpoint value suggesting high accuracy. The SSE of the NNPLS model is 0.5678, and the SSE of the 6th Type II nonlinear PLS model is 0.5917. The NNPLS model and the 6th order Type II nonlinear PLS model can both predict the endpoint of the 6th order testing nonlinear system, and the accuracy of the Type II nonlinear PLS model is similar as NNPLS in this test system.

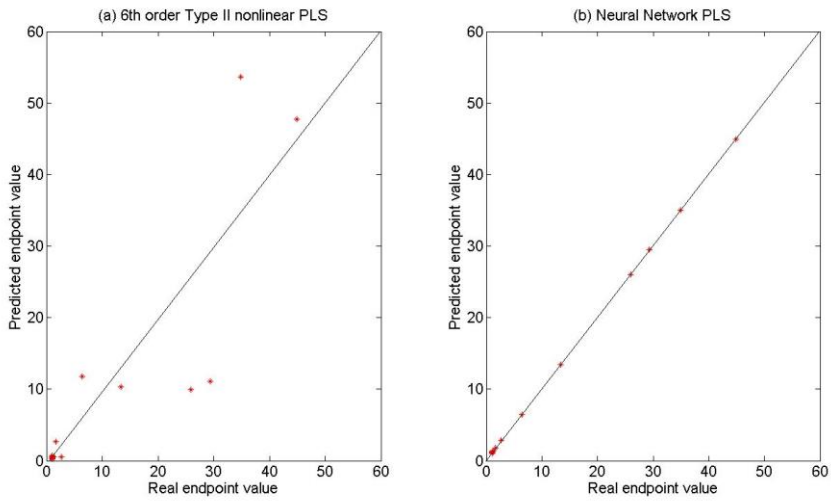

Figure 4.9 The Application of 6th Type II Nonlinear PLS and NNPLS Model to Test 7th Order Nonlinear System

In Figure 4.9 (b), the predicted endpoint value is also very close to the actual endpoint value for NNPLS. The SSE is 0.4803. However, Figure 4.9 (a) shows that where the order of the nonlinearity does not match that of the process, prediction accuracy is reduced significantly in the Type II nonlinear PLS model. The SSE is 993.6. In general, the exact order of any nonlinear relationship will not be known a-priori, so the inner relation of the Type II nonlinear PLS model is difficult to determine. This is a major limitation of the Type II nonlinear PLS model; the NNPLS does not have this limitation. In the NNPLS, the advantage of the application of neural network in the inner regressors is due to their nonlinear approximation property. So Neural Network-based PLS can immune to polynomial order change of the nonlinear model.

4.4 Summary

In this chapter, the MPLS model was applied to four example systems. The results demonstrated that the linear MPLS model was suitable for linear and time varying or time-invariant systems. When the system was nonlinear, the MPLS model cannot provide the accuracy of the predictive value. To overcome the limitation of linear MPLS, Type I and Type II nonlinear PLS model were proposed and applied in this chapter.

In the Type I Nonlinear PLS method, the observed variables were appended with nonlinear transformations. As mentioned, in practice it might be very difficult to find such a simple nonlinear transformation. This implies that the exact expansion of the X matrix is very important. If the exact expansion of the X matrix cannot be determined, the accuracy of the Type I nonlinear PLS model is unsatisfactory. In this chapter, the Type II nonlinear PLS model was applied, and it showed that it can predict the nonlinear system very accurately. In comparison to the linear PLS model, the predicted effect is very obvious in the Type II nonlinear PLS model, because the inner relation of this Type II nonlinear PLS model is a quadratic polynomial. The application of this Type II nonlinear PLS model has though a certain limitation. The inner relation of the system had expended to 4th order and 6th order polynomials.

The results showed that when the order of the Type II nonlinear MPLS model could be precisely determined, this model can be applied to predict the end-point value.

In real industry, the exact order of any nonlinear relationship is not easy to determine, hence the Type II nonlinear PLS model is not commonly utilised to predict the nonlinear system. To overcome this problem, NNPLS is applied to the prediction of the nonlinear system. The results showed that NNPLS is a better method to model the endpoint value of the nonlinear system, than Type II nonlinear PLS. The results in this chapter have been published by Yan and Lennox, (2013).

To illustrate the capabilities of the nonlinear extensions to MPLS, the Type I and II algorithms and NNPLS will be applied to a benchmark simulation of a penicillin batch fermentation process. This is discussed in the next chapter.

Chapter 5

Application of Linear and Nonlinear PLS Modelling Techniques to Fermentation Process Simulator

In Chapter 4, Linear PLS, Type I and II nonlinear PLS, and NNPLS were applied to some simple simulations systems. In this chapter, to illustrate NNPLS' capabilities, it is applied to a benchmark simulation of a penicillin batch fermentation process. The fermentation process investigated is the Pensim simulator (Birol et al., 2002). To provide some comparisons, Linear PLS, the Type I nonlinear model and the Type II nonlinear model are also applied to test the Pensim data. There are two primary quality output variables in this process, biomass and penicillin; these are each affected by the primary manipulated variable, the substrate feed-rate. By analysing the response of this system, it can be determined that the relationship between the substrate and biomass is linear and time invariant, and for penicillin, the relationship is nonlinear and time varying.

The end-point measurement is used because in most fermentation processes, quality measurements such as penicillin concentration will only be available at the end of a batch. For this reason, this chapter will focus upon the endpoints of the products (Biomass and Penicillin). Linear MPLS has been shown to be able to be applied to predict linear systems, particularly linear time varying systems. The Linear MPLS model therefore is used to predict the endpoint of Biomass and Penicillin. The results show that linear MPLS is not able to accurately predict the endpoint of penicillin, hence the nonlinear PLS model is applied. This chapter will also summarise the limitations of Type I and II nonlinear models, in addition to analysing the results when Multi-way NNPLS is applied to predict the endpoint of penicillin concentration. By analysing and comparing linear multi-way PLS, Neural network multi-way PLS, and Type I and Type II nonlinear multi-way PLS models, the

advantages and limitations of these methods are identified and summarized.The chapter is divided into the following sections:

5.1) introduces the benchmark simulation (Pensim);

5.2) describes the application of MPLS model, and discusses the predicted endpoint value of Biomass and Penicillin;

5.3) discusses the application of Type I and II Nonlinear MPLS model to the Estimation of Penicillin, and identifies the limitations of Type I and II nonlinear MPLS;

5.4) describes the application of the Neural network MPLS model to the Estimation of ; Penicillin, and explains the benefits of Neural network MPLS; and

5.5) summarizes and concludes this chapter.

5.1 Introduction of the Benchmark Simulation (Pensim)

Pensim is a benchmark simulation of a fed-batch fermentation system. The simulator is based on a series of detailed mechanistic models that describe an industrial fed-batch fermentation process, used for the production of penicillin. The original models were proposed by the Control Group at Illinois Institute of Technology (Briol et al., 2002).

A basic flow chart of the process is presented in Figure. 5.1.

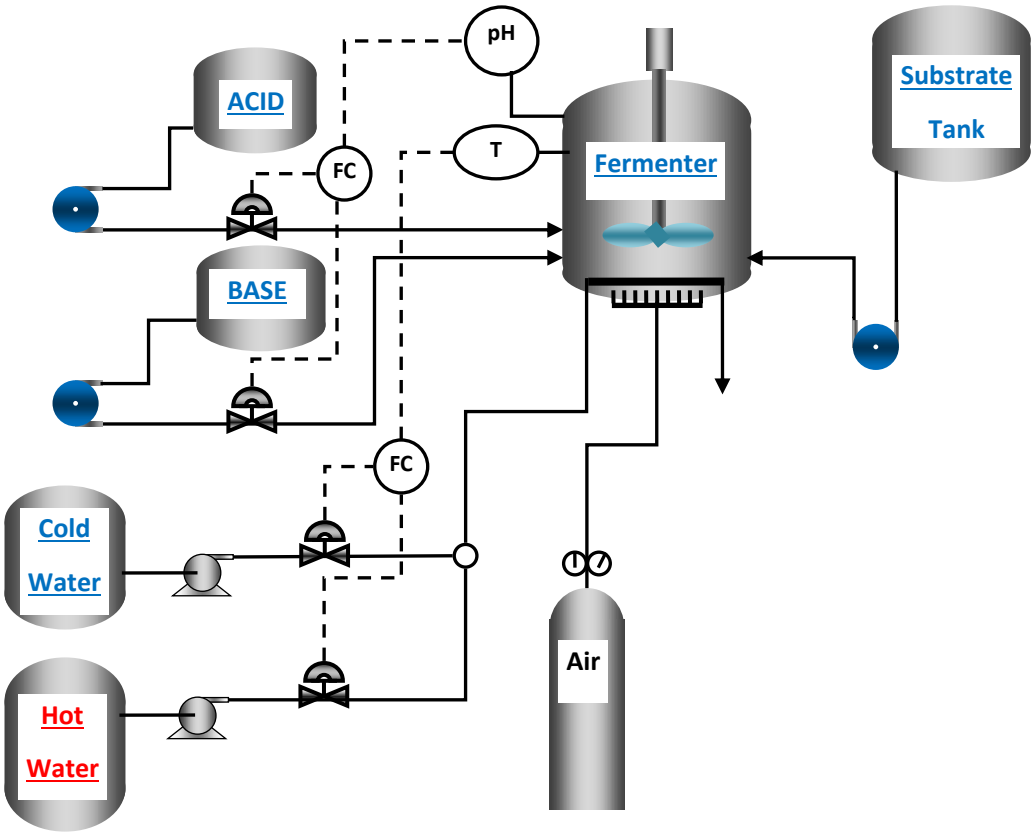

Figure 5.1 The Basic Flow Chart of an Industrial Fed-batch Fermentation Process in the Production of Penicillin (Briol et al., 2002)

Initially, a small amount of biomass, substrate and water is loaded into the fermenter. The substrate tank provides the reaction material (substrate). In the operation process, air and substrate is fed into the fermenter. Hot and cold water are used to heat up and cool down the reactor temperature. Acid and Base are applied to regulate the pH value.

The process consists of two phases. During the initial batch phase, no substrate is fed and the microorganisms grow on glucose (the main substrate) initially available in the broth. The reactor is switched from batch to fed-batch mode, once the glucose concentration drops below 0.3 g/L. At this phase, a continuous stream with additional substrate is fed. Due to the low substrate concentration in the reactor, the microorganisms produce penicillin as a secondary metabolite. Fermentation is stopped when a total of 25 L of substrate feed has been added to the reactor. Process input and output variables are listed in Table 5.1.

Table 5.1 Process Input/Output Structure (Briol et al., 2002)

Input variables	Output variables
Glucose Feed Temperature	Culture Volume
Glucose Feed Flow Rate	Fermenter Temperature
Aeration Rate	Generated Heat
Agitator Power Input	pH
Coolant Flow Rate	Concentrations of Glucose
Acid/ Base Flow Rate	Concentrations of Biomass
	Concentrations of Penicillin
	Concentrations of Dissolved Oxygen
	Concentrations of Carbon Dioxide

The initial conditions of the various states in the model are listed in Table 5.2, as well as the set points for the process inputs. The initial substrate concentration, biomass concentration, and culture volume are subject to random variations for each batch to represent changing initial conditions. They are sampled from a normal distribution with 95% confidence intervals indicated in Table 5.2. Additionally, small low frequency fluctuations are added to several process inputs to represent a real process environment. Reactor temperature and pH are controlled at their respective set points by standard PID controllers during both phases. The parameter of PID controller can be found in the original Pensim paper (Briol et al., 2002).

Table 5.2 Initial Conditions of the State Variables and Set Point of the Process Inputs in Pensim

Initial conditions	Value
Substrate concentration [g/L]	15 ± 2
Biomass concentration [g/L]	0.1+0.05
Culture Volume [L]	100 ± 10
Dissolved oxygen concentration [g/L]	1.16
Penicillin concentration [g/L]	0
CO2 concentration [g/L]	0.5
pH [–]	5
Reactor temperature [K]	298
Reaction heat [cal]	0
Process inputs	Set point
Substrate feed rate [L/h]	0.05
Aeration rate [L/h]	8
Agitator power [W]	30
Feed temperature [K]	296
Controlled variables	Set point
Reactor temperature [K]	298
pH [–]	5

Pensim provides measurements of 17 process variables. Table 5.3 provides an overview of these measurements. If total production hours to be simulated are 200, and the sampling interval used in this work was 1 hour, the data of each batch contained 200 observations and 17 variables under normal operation conditions. Figure 5.2 provides an example of the Pensim data.

Table 5.3 Process Variables in Pensim

Number	Process variables
1	Aeration Rate (L/h)
2	Agitator Power (W)
3	Substrate Feed Rate (L/h)
4	Substrate Feed Temperature (K)
5	Substrate Concentration (g/L)
6	Dissolved Oxygen Concentration (g/L)
7	Biomass Concentration (g/L)
8	Penicillin Concentration (g/L)
9	Culture Volume (L)
10	Carbon Dioxide Concentration (g/L)
11	pH
12	Reactor Temperature (K)
13	Generated Heat (kcal)
14	Acid Flow Rate (L/h)
15	Base Flow Rate (L/h)
16	Cooling Water Flow Rate (L/h)
17	Hot Water Flow Rate (L/h)

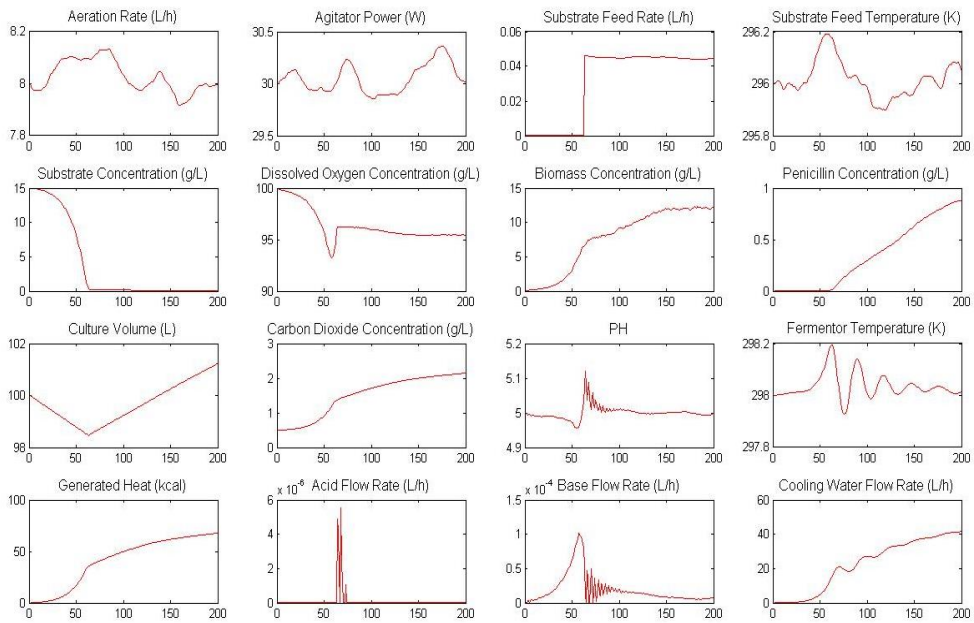

Figure 5.2 Example of the Pensim Data

The functional relationships among the process variables are summarized in Table 5.4.

Table 5.4 Functional Relationship among the Process Variables (Briol et al., 2002)

Model Structure
$X_{bio} = f(X_{bio}, S, C_L, H, T)$
$S = f(X_{bio}, S, C_L, H, T)$
$C_L = f(X_{bio}, S, C_L, H, T)$
$P = f(X_{bio}, S, C_L, H, T, P)$
$CO_2 = f(X_{bio}, H, T)$
$H = f(X_{bio}, H, T)$

In Table 5.4, X_{bio} is biomass concentration; S is substrate concentration; C_L is dissolved oxygen concentration in the broth; P is penicillin concentration; CO_2 is carbon dioxide concentration; H is hydrogen ion concentration for pH ($[H^+]$); and T is reactor temperature.

The model equations are introduced as the following parts (Briol et al., 2002): there are a total of 15 differential equations which are solved simultaneously. All the parameters are taken from literature or assigned values (Briol et al., 2002). The parts of the equations are displayed in the following:

Biomass Growth:

$$\frac{dX_{bio}}{dt} = \mu X_{bio} - \frac{X_{bio}}{V}\frac{dV}{dt}, \tag{5.1}$$

where μ is specific growth rate. It contains the effects of environmental variables (pH and Temperature). Glucose (S) and oxygen (C_L) are considered in its kinetic expression.

It is defined:

$$\mu = \mu_x \frac{S}{(K_x X_{bio} + S)} \frac{C_L}{(K_{ox} X_{bio} + C_L)}, \tag{5.2}$$

μ_x is maximum specific growth rate.

Penicillin Production:

$$\frac{dP}{dt} = \mu_{pp} X_{bio} - KP - \frac{P}{V}\frac{dV}{dt}, \tag{5.3}$$

where μ_{pp} is the specific penicillin production rate. It contains biomass(X_{bio}), glucose(S) and oxygen (C_L) in its kinetic expression.

It is defined:

$$\mu_{pp} = \mu_p \frac{S}{(K_p + S + S^2/K_I)} \frac{C_L^P}{(K_{op} X_{bio} + C_L^P)}. \tag{5.4}$$

Substrate utilization

Glucose:

$$\frac{dS}{dt} = -\frac{\mu}{Y_{x/s}} X_{bio} - \frac{\mu_{pp}}{Y_{p/s}} X_{bio} - \mu_x X_{bio} + F - \frac{S}{V}\frac{dV}{dt}. \tag{5.5}$$

Oxygen:

$$\frac{dC_L}{dt} = -\frac{\mu}{Y_{x/o}} X_{bio} - \frac{\mu_{pp}}{Y_{p/o}} X_{bio} - \mu_o X_{bio} + K_{1a}(C_L^* - C_L) - \frac{C_L}{V}\frac{dV}{dt}, \tag{5.6}$$

where K_{1a} is taken to be a function of agitator power input P_W and flow rate of oxygen f_g as suggested by Bailey and Ollis (1986).

It is defined:

$$K_{1a} = \alpha\sqrt{f_g}\left(\frac{P_W}{V}\right)^{\beta}. \tag{5.7}$$

Carbon dioxide production

$$\frac{dCO_2}{dt} = \alpha_1 \frac{dX_{bio}}{dt} + \alpha_2 X_{bio} + \alpha_3, \tag{5.8}$$

here, the values of α_1, α_2 and α_3 have been taken from Montague et al. (1986).

5.1.1 The Relationship between the Quality Output Variables and the Manipulated Variable

In Pensim data, substrate feed-rate was the primary manipulated variable; it affected two primary quality out variables: Biomass and Penicillin. The relationship between the quality out variables and the manipulated variable were tested in this section.

3 batches of Pensim data were collected for testing. Every batch data contains 200 sample times. One simple time is one hour. In order to better display the results, Pseudo-Random Binary Signals (PRBS) and white noise are not added into these batches.

Batch 1 was collected under normal operation condition.

In Batch 2, a step signal was added in the manipulated variable-substrate feed-rate from 100 to 200 sample times (magnitude = 20%).

In Batch 3, another step signal was added in the substrate feed-rate from 100 to 200 sample times (magnitude = 40%).

Afterwards, 3 batches of the response of biomass and penicillin value are presented in Figure 5.3 and 5.4.

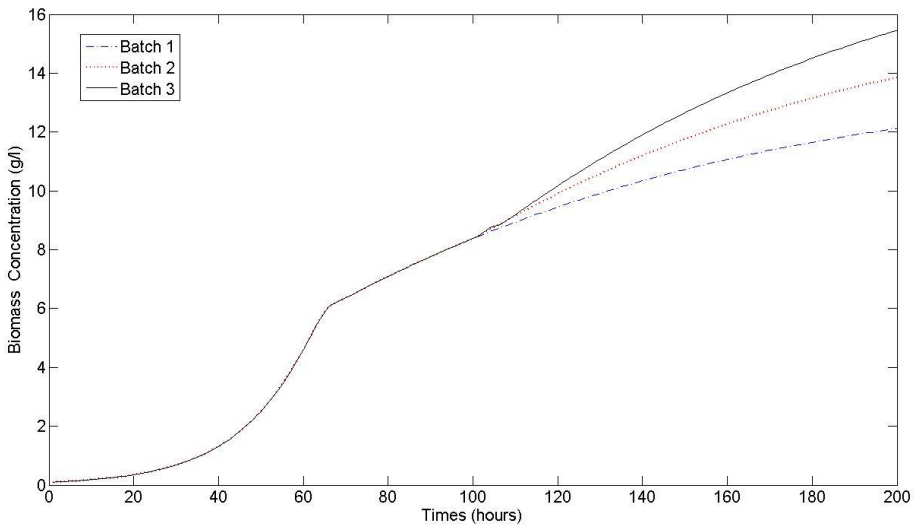

Figure 5.3 Biomass Response Results

In Figure 5.3, the blue dashdot line is Batch 1, the red dashed line is Batch 2, and the black line is Batch 3. If the system is linear, the system needs to satisfy the following conditions:

$$y = f(x + \Delta x) = f(x) + f(\Delta x) \tag{5.9}$$

In this case, x is the substrate feed-rate, and y is the biomass concentration. Δx is the 20% and 40% magnitude of step signal.

The results show that the difference between Batch 1 and Batch 2 is approximately equal to the difference between Batch 2 and Batch 3. This demonstrates that the relationship between biomass and substrate is linear.

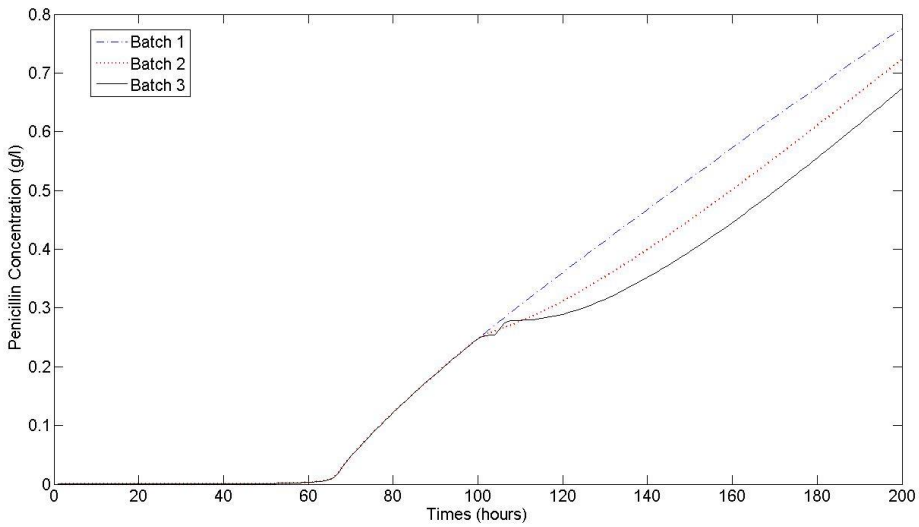

Figure 5.4 Penicillin Response Results

In Figure 5.4, the results show that the relationship between Penicillin and substrate does not meet the linear condition, thus it is nonlinear.

5.2 The Application of MPLS Model

In the Pensim data, there are two primary quality output variables in this process, biomass and penicillin. In this section, MPLS is applied to predict the endpoint

values of Biomass and Penicillin. In the MPLS model, 20 batches of Pensim data were collected for training the models, and 100 batches were collected for testing purposes. All batches of Pensim data contain 200 sample times. Pensim provides measurment of 17 variables. In this thesis, for building PLS model, Biomass Concentration, Penicillin Concentration and Carbon Dioxide Concentration are not considered, other rest process variables are the model inputs (the cause variables).

Firstly, the MPLS model is used to predict the final productivity of the Batch (Biomass and Penicillin). In the following data, Pseudo-Random Binary Signals (PRBS) with high/low values of -0.008 and 0.008 were applied to the nominal feed-rate of substrate (0.05 L/h) in order to excite process dynamics. In this thesis, to better simulate an industrial fermenter, random disturbance and noise were added in the simulation; these disturbances were introduced as white noise sequences, with a standard deviation of 0.1, 0.2, 0.05, 0.05, and were applied to the biomass growth constant, to the carbon dioxide evolution rate, and to the feed-rates of the base and cooling water respectively.

5.2.1. The Application of MPLS to Predict the Final Productivity of the Batch (Biomass)

When MPLS is applied to real batch process, the number of batches used to identify the model is important to consider. Therefore, in this section, the training data is selected as 5 batches, 10 batches, and 20 batches in the MPLS model. The influence of the number of batches used in modelling was then tested. *Y*-data used the biomass value at the end-point of each batch. PRBS and random white noise have been added in the process, because they have a certain degree of randomness. When the testing data is bigger, the result can be more accurate. In order to ensure the experiments' accuracy, testing data was collected 10 times, and every time, the testing data included 100 batches. The average of the 10 times' predicted results were analysed. The number of latent variable was selected by using crossing validation; the results of which are presented in Figures 5.5, 5.6 and 5.7.

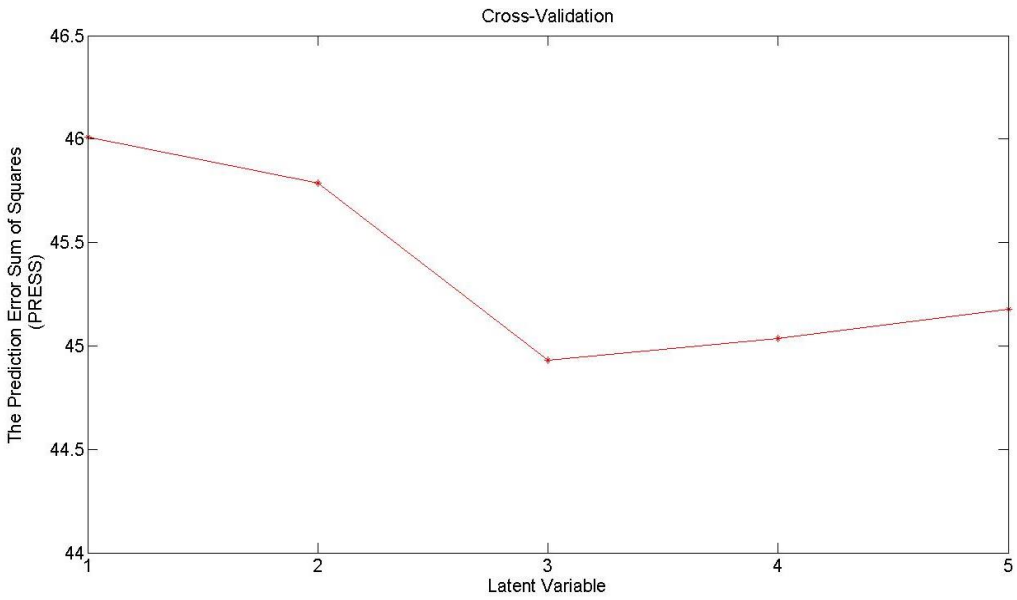

Figure 5.5 Cross-Validation of MPLS Model (5 batches training data)

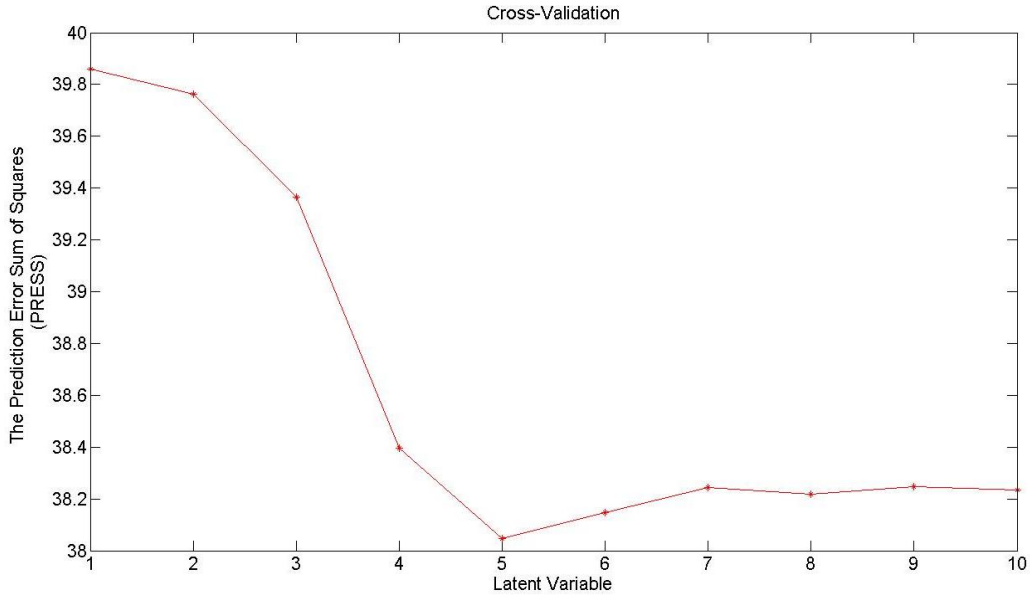

Figure 5.6 Cross-Validation of MPLS Model (10 batches training data)

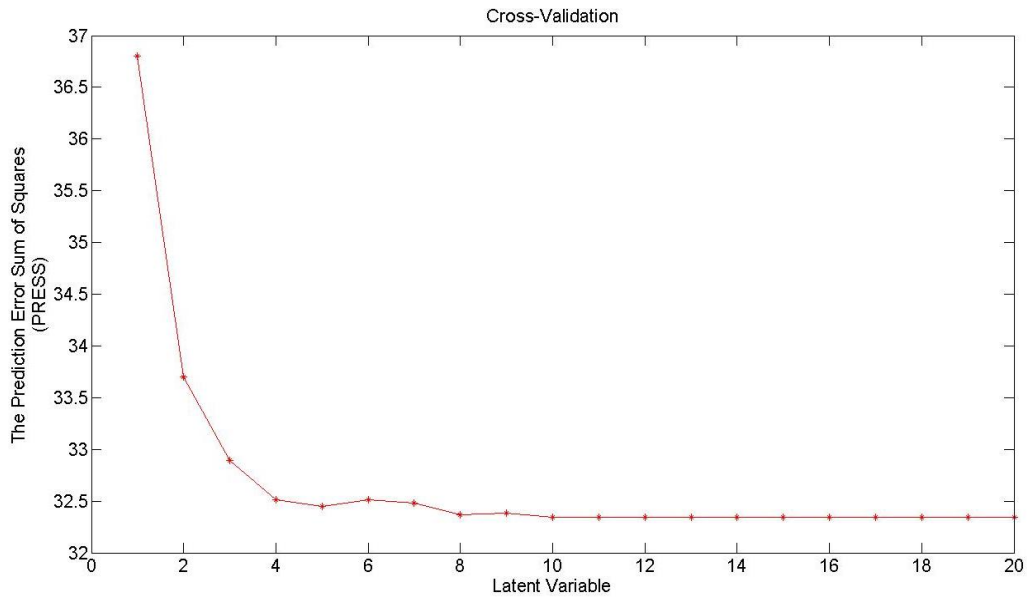

Figure 5.7 Cross-Validation of MPLS Model (20 batches training data)

In Figures 5.5, 5.6 and 5.7, there are minimum values of the Prediction Error Sum of Squares (PRESS) when latent variable is selected as 3 (5 batches training data), 5 (10 batches training data), and 10 (20 batches training data). Therefore, in the MPLS model, the number of latent variables is respectively found to be 3, 5 and 10.

The predictive data are shown in Figure 5.8 (5 batches training data), in Figure 5.9 (10 batches training data) and in Figure 5.10 (20 batches training data). In these figures, the *Y*-axis represents the actual data, and the *X*-axis represents the predictive data. The blue diamond points are training data and the red star points are testing data (Note: The predictive data are the same, so one time is displayed in here).

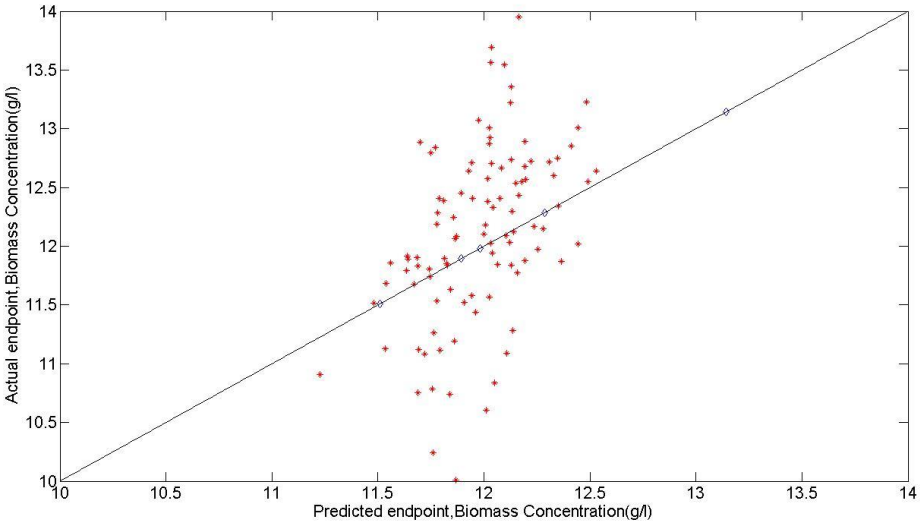

Figure 5.8 The End Point Prediction of the Biomass (5 batches training data)

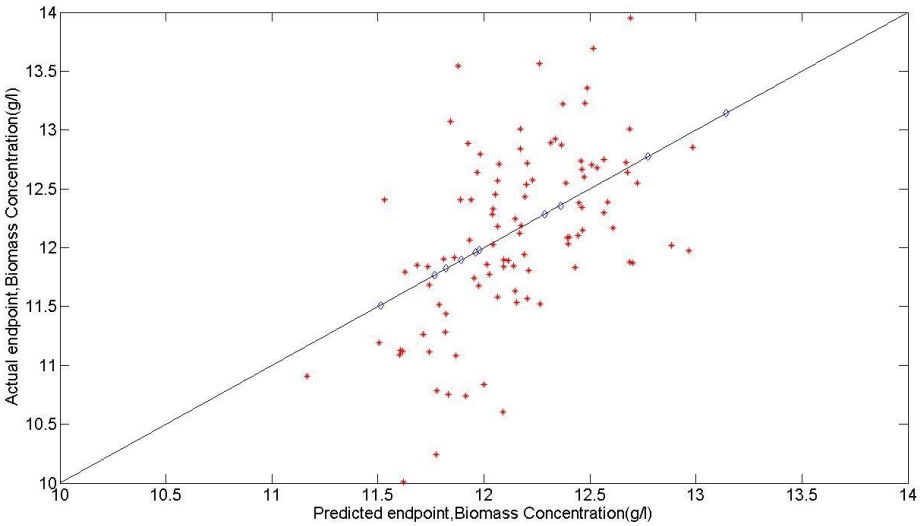

Figure 5.9 The End Point Prediction of the Biomass (10 batches training data)

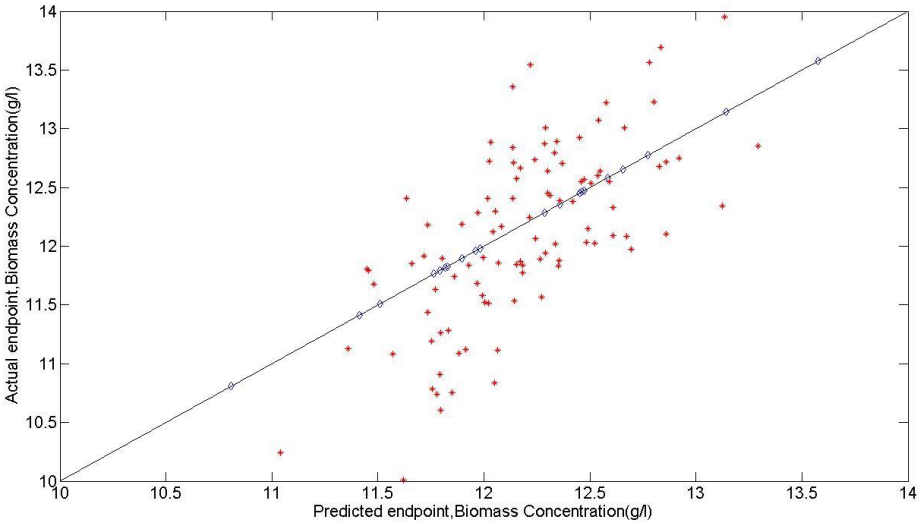

Figure 5.10 The End Point Prediction of the Biomass (20 batches training data)

When the PLS model is built, the testing data is used to assess the model's accuracy. The average error of the testing batch and training batch are calculated. The results are recorded in Table 5.5. In order to directly compare the data, the results are depicted in a bar chart (Figure 5.11).

Table 5.5 The Average Error of Testing Batch and Training Batch (Biomass)

Time No.	5 training	10 training	20 training
1st	0.3168	0.2217	0.2131
2nd	0.3054	0.2094	0.1989
3rd	0.3249	0.2283	0.2159
4th	0.2942	0.1963	0.1849
5th	0.2896	0.1846	0.1795
6th	0.3369	0.2317	0.2198
7th	0.3146	0.2168	0.2031
8th	0.2848	0.1811	0.1789
9th	0.3165	0.2031	0.1913
10th	0.3067	0.2129	0.2064

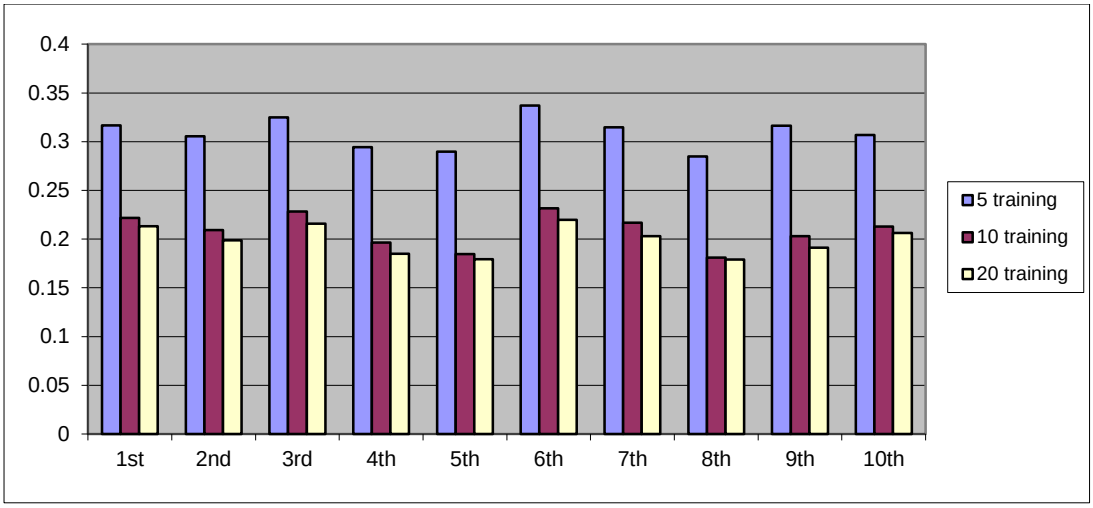

Figure 5.11 The Average Error of Testing Batch (Biomass)

In order to ensure the experiments' accuracy, testing data was collected 10 times, and every time, the testing data included 100 batches. In Figure 5.11, X-axis is time number. Some conclusions can be drawn from Figure 5.11. When the training batch number increases from 5 batches to 10 batches, the average error decreases evidently (the same test data); thus when the testing data are the same, with the increase in the number of training data, the accuracy of the model would also be improved.

However the error reduces only very slightly after 10 training batches. This demonstrates that when training data is enough to identify the MPLS model, providing more training data is not useful for improving the MPLS model's accuracy.

5.2.2 The Application of MPLS to the Final Productivity of the Batch (Penicillin)

Training data are selected as 5 batches, 10 batches, and 20 batches. *Y*-data used the penicillin value in the endpoint of each batch. The predictive data are shown in Figures 5.12, 5.13 and 5.14. In these figures, the *Y*-axis represents the actual data, and the *X*-axis represents the predictive data. The blue diamond points are training data and the red star points are testing data.

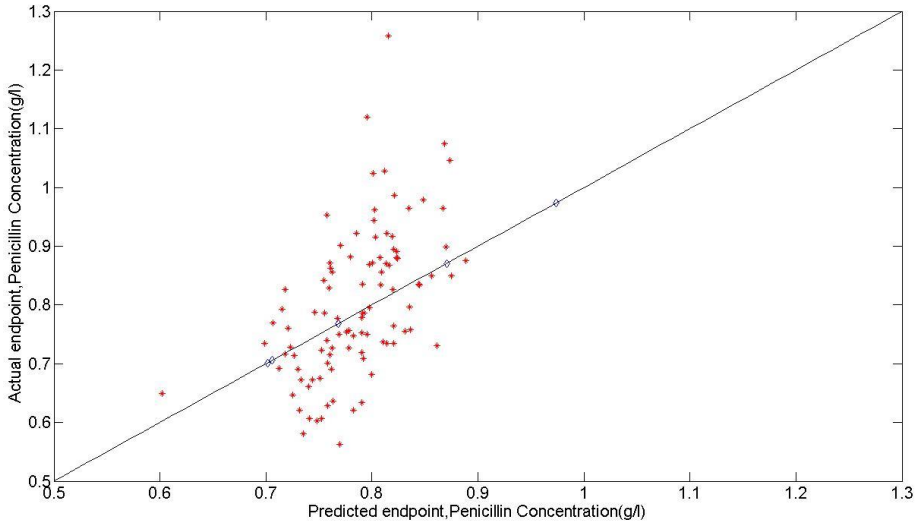

Figure 5.12 The End Point Prediction of the Penicillin (5 batches training data)

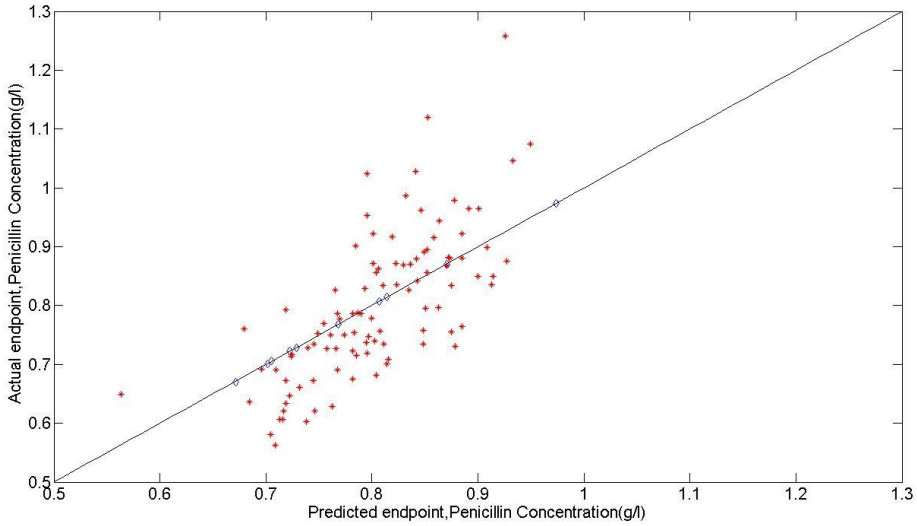

Figure 5.13 The End Point Prediction of the Penicillin (10 batches training data)

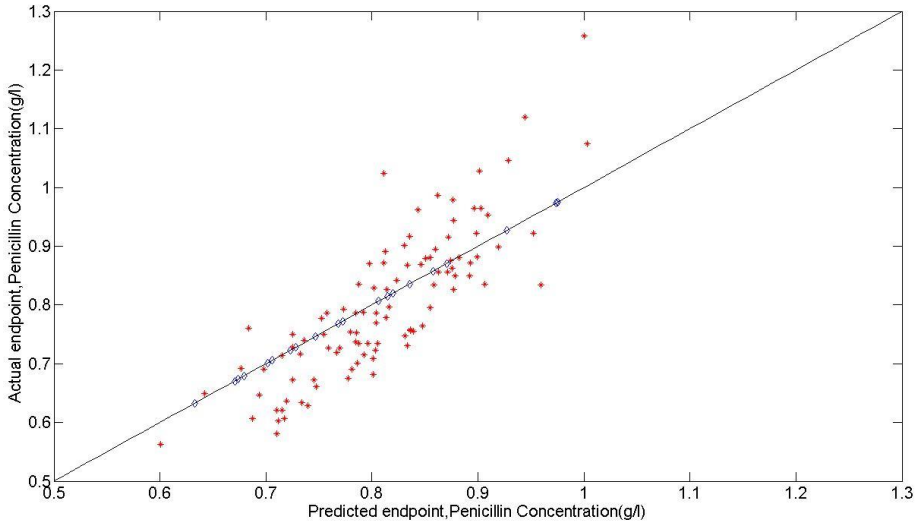

Figure 5.14 The End Point Prediction of the Penicillin (20 batches training data)

In Figures 5.12, 5.13 and 5.14, the results show that the error is very large. This is because the relationship between Penicillin and Substrate is nonlinear, therefore the linear MPLS model cannot be used to predict the endpoint value of Penicillin.

The average error of the testing batch and of the training batch are calculated (Table 5.6). In order to directly compare data, the results are presented in Figure 5.15.

Table 5.6 The Average Error of Testing Batch (Penicillin)

time	5 training	10 training	20 training
1st	0.039	0.026	0.0198
2nd	0.0292	0.0214	0.017
3rd	0.0364	0.0258	0.0182
4th	0.0383	0.0271	0.0168
5th	0.0327	0.0263	0.0167
6th	0.0247	0.0218	0.0161
7th	0.028	0.022	0.0164
8th	0.0278	0.0234	0.014
9th	0.022	0.016	0.0139
10th	0.027	0.0208	0.0171

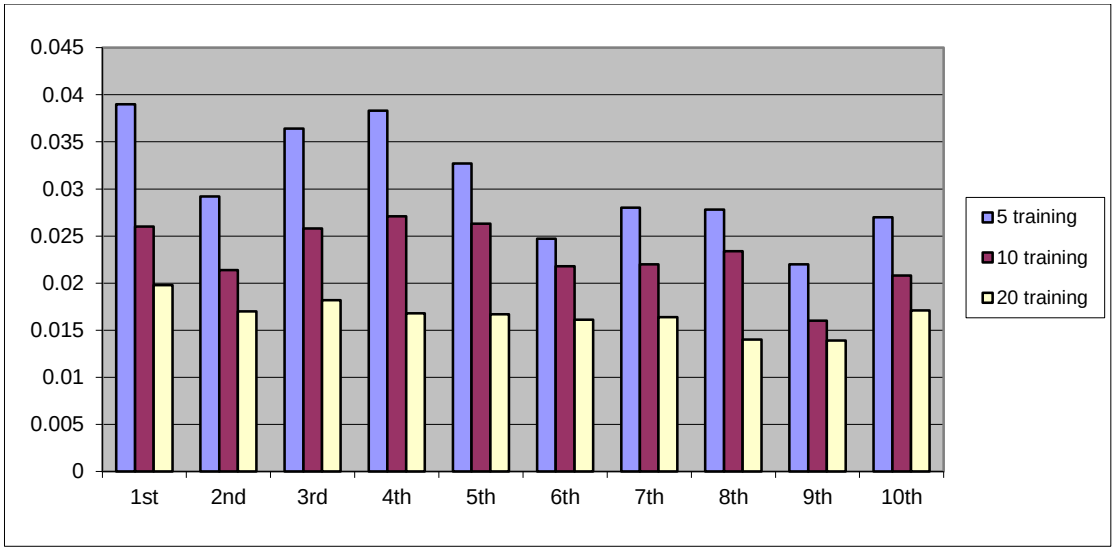

Figure 5.15 The Average Error of Testing Batch (Penicillin)

When comparing the Biomass and Penicillin results, the accuracy of the PLS model is better when it is used to predict biomass in the end point, than when it is used to predict penicillin.

Note: When the training data selected was 10 batches, the Average Error of Testing Batch (biomass) is about 0.2. The average error of the penicillin is 0.02. The average of biomass's endpoint value is 12. The average of Penicillin's endpoint value is 0.65. The error rates are 0.0167 (Biomass) and 0.0308 (Penicillin).

$$The\ error\ rate = \frac{The\ average\ predicted\ error}{the\ normal\ value\ of\ the\ endpoints} \tag{5.10}$$

5.2.3 Application of MPLS to Track the Trajectories of the Batch

In order to highlight the capability and limitations of MPLS, the MPLS model was applied to estimate the biomass and penicillin trajectories. The predicted error of the trajectories is used to analyse the accuracy of the MPLS model.

In Pensim data, the following process variables were collected hourly from simulated runs of 40 batches with a duration time of 200 hours for the fed-batch operation: aeration rate, agitator power, substrate feed temperature, substrate concentration,

dissolved oxygen concentration, culture volume, pH, fermenter temperature and generated heat, substrate feed rate and biomass concentration. The substrate feed rate is the manipulated process variable while the biomass concentration and the Penicillin concentration are the process variable to be tracked. In the following data, Pseudo-Random Binary Signals (PRBS) with high/low values of -0.008 and 0.008 were applied to the nominal feed-rate of substrate (0.05 L/h), in order to excite process dynamics.

The training data consisted of 20 batches, with 20 batches used for testing each model. The number of latent variables, selected using crossing validation, was found to be 10. The results are in Figure 5.16 and 5.17. These figures display that MPLS was applied in the 4 batches of testing data (penicillin and biomass respectively). The predicted curve is red solid line and the actual curve is blue dashed line.

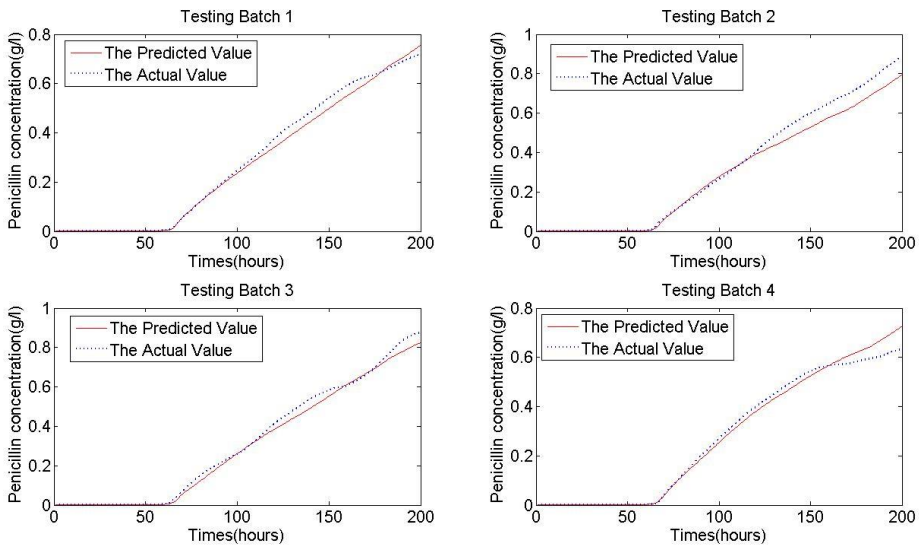

Figure 5.16 The Predicted Trajectory of the Penicillin

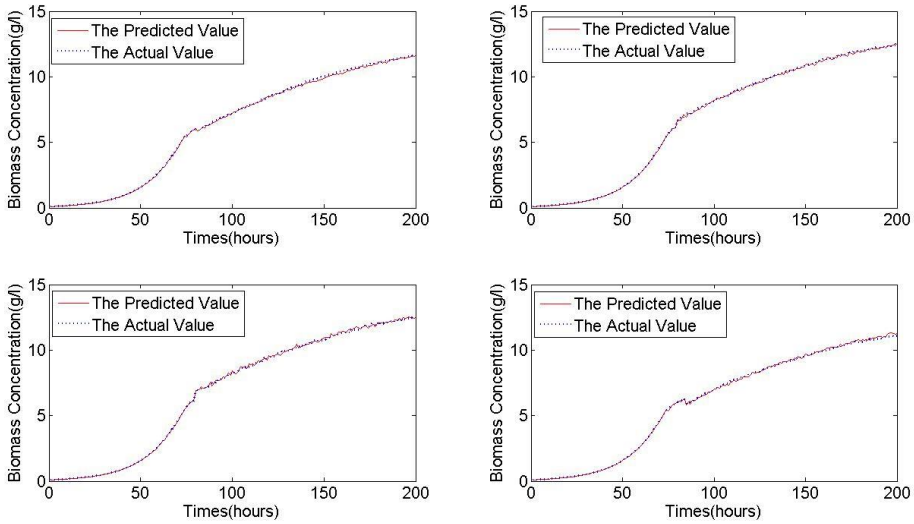

Figure 5.17 The Predicted Trajectory of the Biomass

The sum of square error (SSE) was calculated in every testing batch. The results are listed in Tables 5.7 and 5.8.

Table 5.7 The SSE of Penicillin in MPLS Model

Testing Batch	The value of SSE (Penicillin)
Number 1	0.0926
Number 2	0.3177
Number 3	0.1138
Number 4	0.1415

Table 5.8 The SSE of Biomass in MPLS Model

Testing Batch	The value of SSE (Biomass)
Number 1	1.0904
Number 2	1.1567
Number 3	1.6877
Number 4	0.9812

The average SSE value of 20 batches testing data are calculated; they are 0.2147 (Penicillin) and 1.2157 (Biomass). The results show that the predicted error is very

bad when MPLS is applied to predict penicillin. The data supports that the PLS model is more accurate when it is used to predict biomass within batch, than when it is used to predict penicillin within batch. Penicillin is nonlinear, thus MPLS would be unable to model penicillin very well, therefore Type I and Type II nonlinear MPLS models were proposed and applied.

5.3 Application of Nonlinear PLS to Pensim

5.3.1 Application of Type I Nonlinear MPLS Model to Estimation of Penicillin

Linear MPLS can capture the relationship between substrate and biomass. The Linear MPLS model can be applied to predict the endpoint value of biomass, but it cannot provide the accuracy prediction of penicillin. Type I nonlinear MPLS was thus applied. Type I Nonlinear MPLS model using second order polynomials only was applied. In the Type I nonlinear MPLS, the expansion of the X matrix was still considered with the quadratic term x^2 only.

The ability of linear MPLS and Type I nonlinear MPLS to predict penicillin for 20 testing whole batches was applied; one testing whole batch is shown in Figure 5.18. The predicted curve is red solid line and the actual curve is blue dashed line. The predicted endpoints of Penicillin in 20 batches testing data are presented in Figure 5.19.

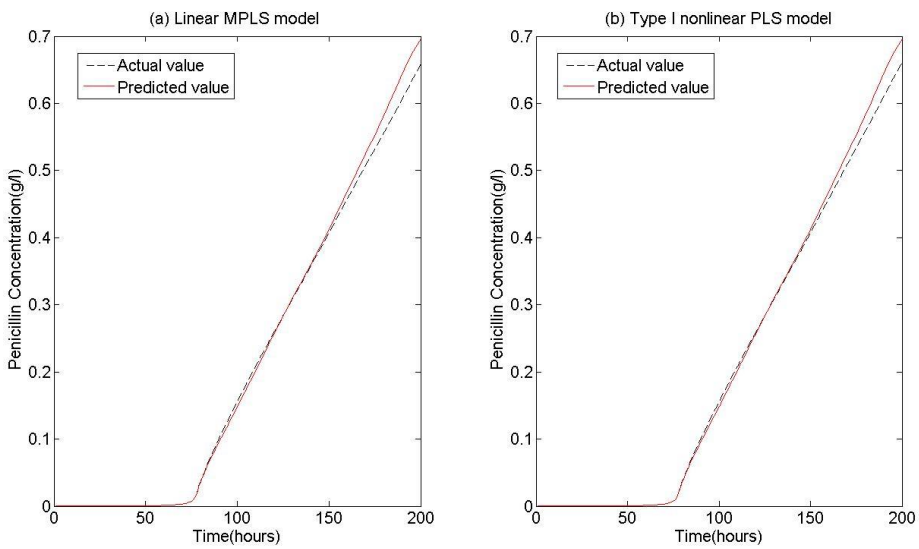

Figure 5.18 Penicillin Prediction Using Linear MPLS and Type I Nonlinear MPLS

In Figure 5.18, the SSE of the one whole batch is 0.1416 in the linear MPLS model; the SSE is 0.1414 in the Type I nonlinear MPLS model. The average SSE value of 20 batches testing data were calculated; they are 0.2147 (MPLS) and 0.2124 (Type I nonlinear PLS). The results illustrate that the Type I nonlinear MPLS model did not significantly improve the accuracy of the prediction.

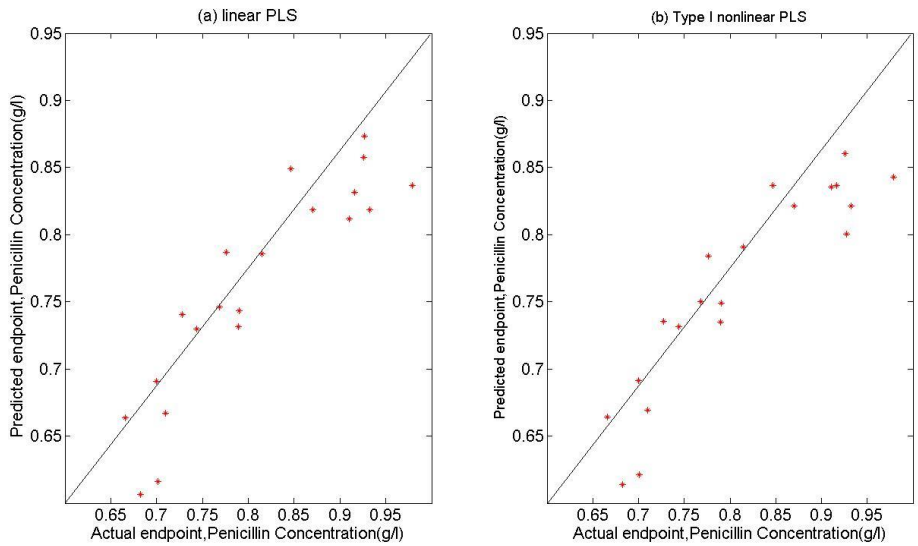

Figure 5.19 The Endpoints of Penicillin Prediction Using Linear MPLS and Type I Nonlinear MPLS

In Figure 5.19, 20 batches testing data were applied to test the linear MPLS model and the Type I nonlinear MPLS model. The predicted endpoints of penicillin were saved, and the SSEs were calculated in these two models. Linear MPLS, as shown in Figure 5.19(a), produced a SSE of 0.0826. The Type I nonlinear MPLS, shown in Figure 5.19(b), produced a SSE of 0.0788. The results illustrate that the Type I nonlinear MPLS model has a slightly improved accuracy over the Linear MPLS model. The accuracy of the Type I nonlinear MPLS model is though still not enough. The reason is that in the Type I nonlinear model, the expansion of the X matrix is only considered with the quadratic term x^2. Therefore, the Type II nonlinear MPLS model was applied to Pensim data.

5.3.2 Application of Type II Nonlinear MPLS Model to Pensim

In this section, the ability of Linear MPLS and the 2nd, 4th and 6th order Type II nonlinear MPLS to estimate the final endpoint concentration of penicillin and biomass are illustrated.

In the following data, Pseudo-Random Binary Signals (PRBS) with high/low values of -0.008 and 0.008 were applied to the nominal feed-rate of substrate (0.05 L/h), in order to excite process dynamics. The training data consisted of 20 batches, with 20 batches used for testing each model. Each batch was allowed to run for 200 samples, with a sample time of 1 hour. The number of latent variables, selected using crossing validation, was found to be 10. The results are shown in Figures 5.20 and 5.21.

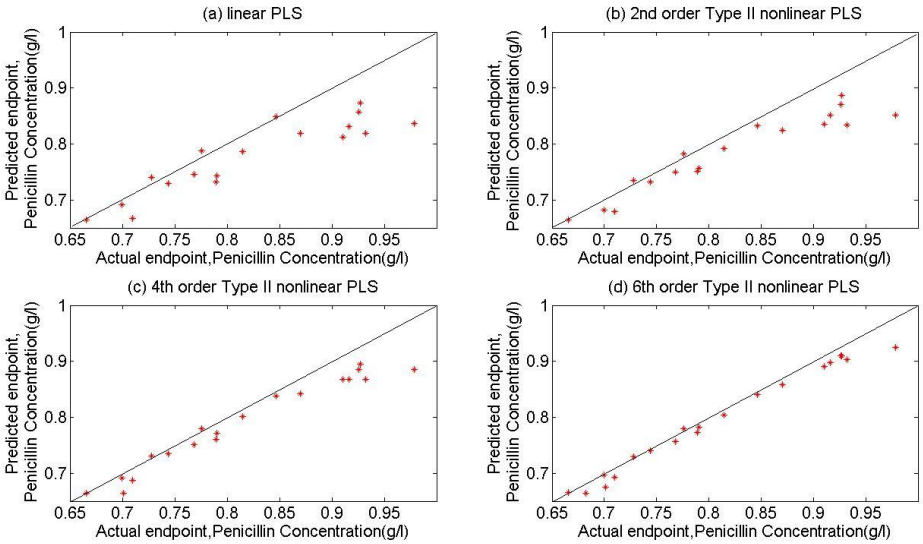

Figure 5.20 The Application of Type II Nonlinear MPLS (Penicillin)

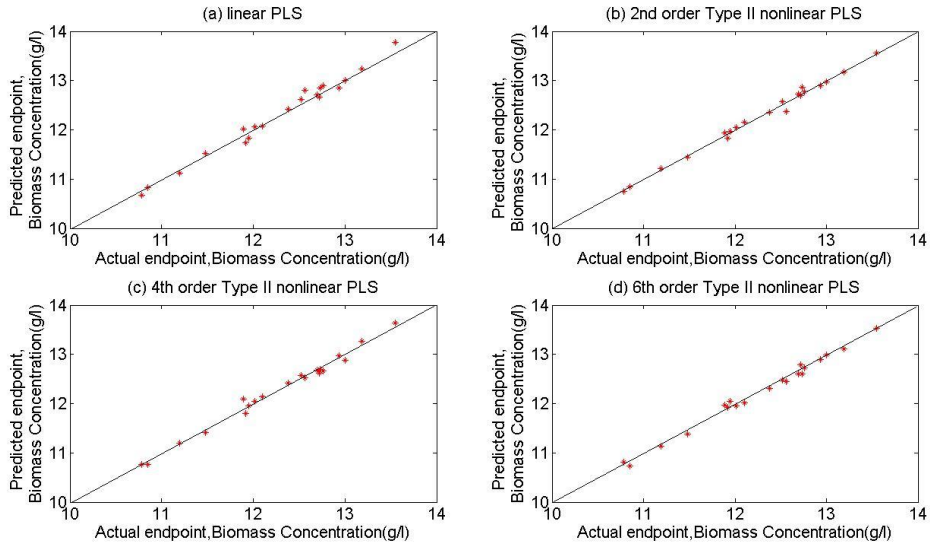

Figure 5.21 The Application of Type II Nonlinear MPLS (Biomass)

The SSE values of these models are calculated, and are listed in Table 5.9.

Table 5.9 The SSE of Type II Nonlinear MPLS

	The value of SSE(Penicillin)	The value of SSE(Biomass)
Linear MPLS	0.0826	0.1615
The 2nd order Type II nonlinear MPLS	0.0557	0.1289
The 4th order Type II nonlinear MPLS	0.0257	0.1246
The 6th order Type II nonlinear MPLS	0.0072	0.1247

When these models are applied to predict the endpoint concentration of penicillin, the result shows that the Type II nonlinear MPLS model can provide a more accurate prediction than Linear PLS, and as the order of the model improves, so too does the model's accuracy.

Because biomass is linear, linear MPLS and Type II nonlinear MPLS therefore can predict the endpoint concentration of biomass very accurately.

5.4 Application of Multi-way Neural Network PLS to Pensim

To illustrate the benefit of using multi-way NNPLS, the ability of this model to predict the endpoint of penicillin concentration is presented. In the following data, Pseudo-Random Binary Signals (PRBS) with high/low values of -0.008 and 0.008 were applied to the nominal feed-rate of substrate (0.05 L/h), in order to excite process dynamics. The training data consisted of 20 batches, with 20 batches used for testing each model. Each batch was allowed to run for 200 samples, with a sample time of 1 hour. The number of hidden layers was selected as 1. Cross validation was used to determine the number of hidden units, which in each case, was found to be 8. The number of latent variable is found be 10. Figure 5.22 shows a

comparison of the predictions made using Type II non-linear MPLS with that obtained using the multi-way NNPLS. In Figure 5.22, the diamonds represent predicted endpoint by the multi-way NNPLS; the red dots represent predicted endpoint by the 6th order Type II nonlinear MPLS.

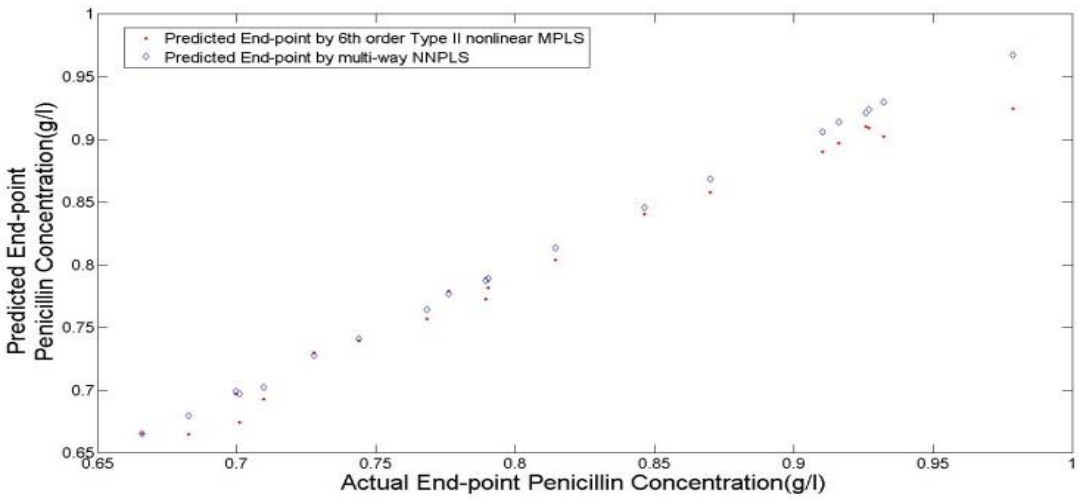

Figure 5.22 The 6th Order Type II nonlinear MPLS Model and Multi-way NNPLS Used to Predict the Endpoint Value of Penicillin

In Figure 5.22, the SSE of 6th order Type II nonlinear MPLS was calculated to be 0.0072; this was higher than the SSE for the multi-way NNPLS, which was 2.94e-04.

The primary advantage of using Multi-way NNPLS is that it provides improved accuracy without the need to determine the order for the model, which can be a critical parameter with Type II MPLS models.

5.5 Summary

In this chapter, MPLS was applied to a benchmark simulation: Pensim - a penicillin batch fermentation process. There are 17 variables in the Pensim data. Substrate feed-rate is the primary manipulated variable, which affected two primary quality out variables: Biomass and Penicillin.

MPLS was applied to predict the whole batch of product quality (Biomass and Penicillin concentration) and the final productivity of the batch. Comparing the results of the average error of testing batches and training batches, MPLS can predict the biomass very well. The reason is that the relationship between the substrate and biomass is linear and time invariant. However, Linear MPLS cannot provide a suitable prediction for penicillin. The reason is the relationship between the substrate and penicillin is nonlinear and time varying.

Type I and Type II nonlinear MPLS models were therefore applied in the Pensim. The results showed that the Type I nonlinear MPLS model did not significantly improve the accuracy of the prediction. The reason is that in the Type I nonlinear model, the expansion of the X matrix is only considered with the quadratic term x^2. However, the actual relationship of the Penicillin system is of a higher order. Compared to Linear MPLS, the 2nd, 4th, and 6th order Type II nonlinear MPLS models have significant improvements when these methods were applied to predict the end value of the penicillin. However, when the Multi-way NNPLS model is used to compare with the 6th Type II nonlinear MPLS model, the results showed that Multi-way NNPLS can provide a better prediction than Type II nonlinear MPLS. The results in this chapter have been published by Yan and Lennox, (2013).

In the next chapter, NNPLS will be applied to control the end point value of the biomass and penicillin concentration.

Chapter 6

Nonlinear PLS Control

NNPLS has been applied to process monitoring. In this chapter, NNPLS will be applied to control the end-point of Biomass and Penicillin. After discussing the end-point control method, the NNPLS model is used to build the controller, which is applied in the Pensim simulation. By analysing the results, the controller's performance is discussed and summarized.

The chapter is divided into the following sections:

6.1) describes the End-point Control Algorithm;

6.2) describes the handling of missing data through single component projection;

6.3) describes the application of a nonlinear controller to control the end-point value of both the biomass and Penicillin concentrations in the Pensim simulation; and

6.4) summarizes and concludes this chapter.

6.1 End-point Control Algorithm

Quality control of batch processes is usually implemented by regulating several process variables, such as temperature and pH. The process variables are well maintained, though the quality of the final product cannot be guaranteed, due to the effects of disturbances. To address this, the endpoint controller is proposed and applied to control the endpoint value at set-point. Cerrillo and MacGregor (2003) proposed a strategy for controlling end-point quality properties. In this controller, the quality of the end-product was regulated by adjusting the trajectory of the manipulated variables in a reduced space (scores) of a latent variable model. Model inversion and trajectory reconstruction is obtained by using the correlation structure in the PLS model. This controller was referred to as the end-point control. In this

thesis, the technique is going to replace the PLS model with a NNPLS model in the controller. More details on the endpoint control methodology are described in this section.

6.1.1 Model building

In this method, the model used in end-point control is obtained by using PLS, and the dataset used to identify the model must contain data that is representative of the process being studied. That is to say, the dataset should contain data collected while the process is being excited. In this thesis, PRBS was applied and added. The detail of the PRBS will be described later. The dataset used for identification should be arranged into a 3-dimensional original array X, and a 2-dimensional response matrix Y ($K \times N$), as shown in Figure 6.1.

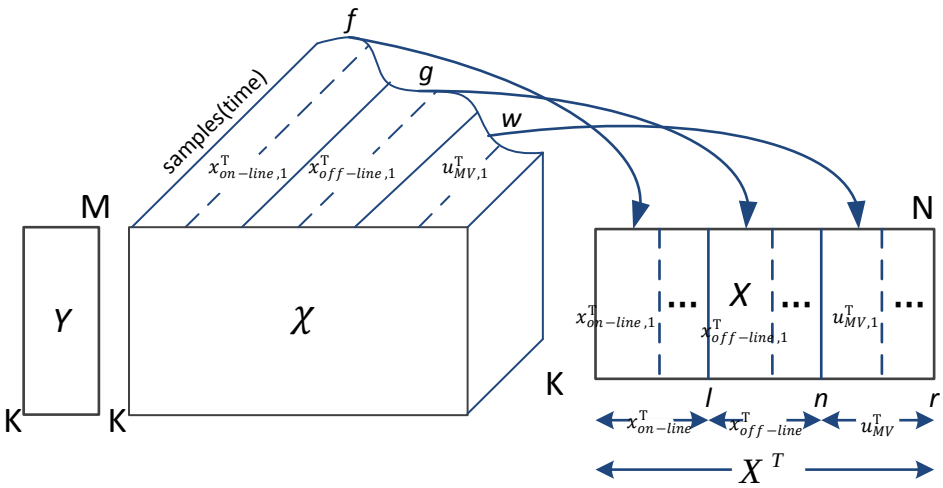

Figure 6.1 Unfolding of Database for Model Building (Cerrillo & MacGregor, 2003)

In Figure 6.1, K is the number of batches, and M are quality properties measured at the end of each batch.

Each row vector of X, denoted as X^T, is composed of:

$$X^T = \left[x_{on-line}^T \quad x_{off-line}^T \quad u_{MV}^T\right], \tag{6.1}$$

where $X_{on-line}^T = [x_{\mathrm{on-line},1}^{\mathrm{T}} \quad x_{\mathrm{on-line},2}^{\mathrm{T}} \quad \cdots \quad x_{\mathrm{on-line},l}^{\mathrm{T}}]$ is a vector of the trajectories of l on-line process variables; $X_{off-line}^T = [\, x_{\mathrm{off-line},1}^{\mathrm{T}} \quad x_{\mathrm{off-line},2}^{\mathrm{T}} \quad \cdots \quad x_{\mathrm{off-line,r}}^{\mathrm{T}}]$ is the set of any off-line measurements collected occasionally on r variables during the batch, and $u_{MV}^T = [u_{\mathrm{MV},1}^{\mathrm{T}} \quad u_{\mathrm{MV},2}^{\mathrm{T}} \quad \cdots \quad u_{\mathrm{MV,n}}^{\mathrm{T}}]$ is a vector of the trajectories of n manipulated variables.

In Figure 6.1, $X_{on-line,j}^T = [x_{on-line,1}, \cdots, x_{online,f}]_j$ and $X_{off,s}^T = [x_{off-line,1}, \cdots, x_{off-line,g}]_s$ represents, respectively, f on-line measurements for the j th variable, and g off-line measurements for the s th variable, while $u_{MV,m}^T = [u_{\mathrm{MV},1}, \cdots, u_{\mathrm{MV},w}]_{\mathrm{m}}$ denotes that w manipulated variables for the m th variable.

As mentioned in Chapter 3, a number of statistical methods are only suitable for 2-dimensional datasets, therefore, a 3-dimensional original array X needs to be transformed into a 2-dimensional dataset. In this work, the X matrix is unfolded as shown in Figure 6.1, where $N = f \times l + g \times r + w \times n$. In the following text, $X_{on-line}^T$ and $X_{off-line}^T$ are combined into a new vector: $X_{Line}^T = [X_{on-line}^T \quad X_{off-line}^T\,]$, and then $X^T = [X_{Line}^T \quad u_{MV}^T]$.

6.1.2 Control

Following the development of the model, the control system can be conceived. Full manipulated variable trajectories (MVTs) can be divided into a number of intervals and control decision points. At each decision point, endpoint quality is predicted. When the predicted final quality deviates from the desired value, the remaining MVTs (after this decision point) are computed to adjust the predicted endpoint quality. In this controller, a number of decision points are applied during the batch, with control action being taken at every decision point. The selection of decision points is arbitrary. Usually, a low number of decision points are adequate. Cerrillo and MacGregor (2003) proposed this form of controller to control product properties in the condensation polymerisation process.

For on-line end-quality estimation ($\hat{y}$), when a new batch *k* is being processed, every decision point (θ_i, i = 1, 2,...) $0 \leq \theta_i \leq \theta_f$, X^T is composed of:

$$X^T = [X_{Line}^T \quad u_{MV}^T] =$$
$$[X_{Line,measured,\theta_i^T} \quad X_{Line,future}^T \quad u_{MV,implemented,\theta_i^T} \quad u_{MV,future}^T] \tag{6.2}$$

X^T consists of: all measured variables ($X_{Line,measured}$) available up to time θ_i ($0 \leq \theta \leq \theta_i$); unmeasured variables ($x_{Line,future}$) not available at θ_i, but will be available in the future ($\theta_{i+1} \leq \theta \leq \theta_f$); implemented control actions $u_{MV,implemented}$ ($0 \leq \theta \leq \theta_{i-1}$); and future control actions $u_{MV,future}$ ($\theta_i \leq \theta \leq \theta_f$) which will be determined through the control algorithm.

The prediction is performed considering $u_{MV,future} = u_{MV,nominal}$(i.e. assuming that the remaining MVTs will be kept at their nominal conditions) using the PLS model:

$$\hat{t}_{present}^T = [X_{Line}^T \quad u_{MV}^T]W$$

$$= [X_{Line,measured,\theta_i}^T \quad X_{Line,future}^T \quad u_{MV,implemented,\theta_i}^T \quad u_{MV,nominal}^T]W \tag{6.3}$$

$$\hat{y}^T = \hat{t}_{present}^T Q^T \tag{6.4}$$

W and *Q* are projection matrices obtained from the PLS model building stage. The vector of scores, $\hat{t}_{present}$, for the new batch is the projection of the *x* vector onto the reduced dimension space of the latent variable model at time θ_i ; $\hat{y}$ is the vector of predicted end-quality properties. $x_{m,future}^T$ is obtained using the PLS model and the missing data algorithm available in the paper (Nelson & MacGregor, 1998; Arteaga & Ferrer, 2002). In this work, Single Component Projection (SCP) is applied; it is the simplest method for missing data prediction. Arteaga suggested using an alternative technique, such as the conditional mean replacement method (CMR). When the alternative techniques are tested, there is little difference between the methods proposed by Arteaga and since SCP is found, therefore in this thesis, the simplest method-SCP is applied. The SCP method will be introduced in next part (Section 6.2).

When the quality prediction need to be controlled, model inversion to obtain MVTs for the remainder of the batch $u_{MV,future}^T$ is needed. Firstly, to compute the

adjustment of the MVTs, the scores (Δt) are required. After that, to obtain the real MVTs for the remainder of the batch, the inversion of the PLS model needs to be considered.

Following the prediction procedure, necessary changes in the scores (Δt) are identified, which will ensure that the predicted endpoint measurement will match the set-point, (y_{sp}). To identify the value of Δt which minimizes the following cost function:

$$\underbrace{min}_{\Delta t(\theta_i)} \left(\hat{y} - y_{sp}\right)^T Q_1 \left(\hat{y} - y_{sp}\right) + \Delta t^T Q_2 \Delta t + \lambda T^2$$

st $$\hat{y}^T = \left(\Delta t + \hat{t}_{present}\right)^T Q^T \tag{6.5}$$

$$T^2 = \sum_{a=1}^{A} \frac{\left(\Delta t + \hat{t}_{present}\right)_a^2}{s_a^2}$$

$$\Delta t_{min} \leq \Delta t \leq \Delta t_{max}$$

where$\Delta \boldsymbol{t}^T = \boldsymbol{t}^T - \hat{\boldsymbol{t}}_{present}^T$, Q_1 is a diagonal weighting matrix defining the relative importance of the variables y's; Q_2 is a diagonal movement suppression matrix that is used as a tuning matrix to moderate the aggressiveness of the control; T^2 is the Hotelling's statistic; s_a^2 is the variance of the score t_a; and λ is a weighting factor which determines how tightly the solution is to be constrained to the region of the score space defined by past operation. $\Delta \boldsymbol{t}_{min}$ and $\Delta \boldsymbol{t}_{max}$ are the constraints which define the minimum and maximum values for $\Delta \boldsymbol{t}$. This final constraint is included to limit the action of the control system.

To identify the value of Δt which minimizes Eqn. 6.5, the vector X is considered to be made up of a series of known trajectories, X_1, and future trajectories, X_2. For control intervals at times θ_i > 0, the $\boldsymbol{X}$ vector trajectory ($X^T = [x_{Line,measured,(0:\theta_i)}^T \quad u_{\mathrm{MV,implemented,(0:\theta_i)}}^{\mathrm{T}} \quad X_{Line,future,(\theta_i:\theta_f)}^T \quad u_{MV,future,(\theta_i:\theta_f)}^T]$) is composed of measured process variables ($x_{Line,measured,(0:\theta_i)}^T$) for the interval $0 \leq \theta < \theta_i$, and for the already implemented manipulated variables ($u_{MV,\mathrm{implemented},(0:\theta_i)}^{\mathrm{T}}$) that must be respected when computing the trajectories for

the remainder of the batch ($\theta_i \leq \theta < \theta_f$). X^T can be divided to X_1^T (known trajectories) and X_2^T (future trajectories). $X_1^T = [X_{Line,measured,(0:\theta_i)}^T \quad u_{MV,implemented,(0:\theta_i)}^T]$ is the known trajectories over time interval $(0:\theta_i)$; $X_2^T = [X_{Line,future,(\theta_i:\theta_f)}^T \quad u_{MV,future,(\theta_i:\theta_f)}^T]$ is the future trajectories over time interval $(\theta_i:\theta_f)$.

At times $\theta_i > 0$, if x is directly reconstructed using as $X^T = t^T P^T$ then:

$$[X_1^T \quad X_2^T] = [t^T P_1^T \quad t^T P_2^T], \tag{6.6}$$

where P_1^T and P_2^T are corresponding loading matrices for X_1^T and X_2^T respectively.

However, the computed $t^T P_1^T$ will not be equal to the actually observed trajectories at time θ_i $X_1^T = [X_{Line,measured,(0:\theta_i)}^T \quad u_{MV,implemented,(0:\theta_i)}^T]$. Therefore, simply selecting $X_2^T = t^T P_2^T$ would not be correct, as it does not account for what has actually been observed for x_1^T in the first part of the batch.

Given this, assume that the remaining trajectories (future manipulated variables and measurements) are:

$$X_2^T = (t^T + \alpha^T) P_2^T, \tag{6.7}$$

where $\alpha^T P_2^T$ is an adjustment to X_2^T that accounts for the effects of discrepancy between $t^T P_1^T$ and x_1^T during the first part of the batch. Therefore:

$$t^T = [X_1^T \quad X_2^T] \begin{bmatrix} W_1 \\ W_2 \end{bmatrix} = X_1^T W_1 + X_2^T W_2, \tag{6.8}$$

where W_1 and W_2 are the weight matrices for X_1^T and X_2^T respectively.
Then

$$X_2^T W_2 = t^T - X_1^T W_1. \tag{6.9}$$

Substituting $X_2^T = (t^T + \alpha^T) P_2^T$ in Equation 6.9:

$$(t^T + \alpha^T) P_2^T W_2 = t^T - X_1^T W_1. \tag{6.10}$$

Therefore,

$$(t^T + \alpha^T) = (t^T - X_1^T W_1)(P_2^T W_2)^{-1}. \tag{6.11}$$

The change in the future process measurements ($X^T_{Line,future,(\theta_i:\theta_f)}$), and the manipulated variables ($u^T_{MV,future,(\theta_i:\theta_f)}$), can be estimated by inverting the PLS model.

$$X_2^T = (t^T - x_1^T W_1)(P_2^T W_2)^{-1} P_2^T, \tag{6.12}$$

where $\Delta t^T = t^T - \hat{t}^T_{present}$, so $t^T = \Delta t^T + \hat{t}^T_{present}$. This inferential control algorithm is then repeated at every decision point (θ_i) until completion of the batch.

6.2 Single Component Projection

There are some methods for dealing with missing data in MSPC, such as: Trimmed score method (TRI), Projection to the model plane (PMP), Conditional mean replacement method (CMR), and Trimmed score regression method (TSR) (Arteaga & Ferrer, 2002). Single Component Projection (SCP) is the simplest method for missing data prediction, although Arteaga (2002) suggested using an alternative technique, such as the CMR and the PMP methods. When the alternative techniques are tested, there is little difference between the methods proposed by Arteaga and since SCP is found, this thesis therefore applied the simplest method – SCP.

The SCP method is proposed by Nelson et al. (1996) and based on the NIPALS algorithm. When MSPC models have been built and the loading vectors are fixed, the non-iterative approach can be applied to handle missing data in new multivariate observation. The score calculation step of the NIPALS missing data model-building algorithm is applied to each dimension sequentially. This method can be briefly introduced by the following (Arteaga & Ferrer, 2002):

If a data matrix X is considered, then the structure can be expressed as

$$X = TP^T \tag{6.13}$$

where T is a $N \times K$ matrix of scores and P is a $K \times K$ matrix of loadings.

These matrices have the standard PCA or PLS properties of the model, of which they are a part. Since the data may not have full rank, some of the columns of T may be composed entirely of zeros. This allows the true dimensionality of the underlying system to differ from that of any model of it. The number of dimensions of any PCA or PLS model in this work is A.

The data X can be considered as a collection of row vectors z_i^T (observation) or column vectors x_j (variables). The K columns of loading matrix P are the loading vectors P_j . The score matrix T can be considered as a collection of row vector τ_i^T (scores of the *i*th observation) or column vectors t_j(latent variables).

For the new object z, the score vector τ can be calculated as

$$\tau = P^T z \tag{6.14}$$

Then it can be expressed as

$$z = P\tau \tag{6.15}$$

When the new observation z has some unmeasured variables, the vector can be partitioned as

$$z = \begin{bmatrix} z^{\#} \\ z^{*} \end{bmatrix} \tag{6.16}$$

where $z^{\#}$ denotes the missing measurements and z^{*} denotes the observed variables.

Matrix X can then be partitioned as

$$X = [X^{\#} \quad X^{*}] \tag{6.17}$$

where $X^{\#}$ is the sub-matrix containing the first R columns of X, with X^{*} accommodating the remaining K-R columns.

Correspondingly, the P matrix can be partitioned as

$$P = \begin{bmatrix} P^{\#} \\ P^{*} \end{bmatrix} \tag{6.18}$$

where $P^{\#}$ is the sub-matrix made up of the first R rows of P, and matrix P^{*} contains the remaining K-R rows.

6.2.1 Handling Missing Data through Single Component Projection

The standard procedure for handling missing data in PCA during model-building is based on the NIPALS algorithm (Nelson et al., 1996). In PCA model-building, one iteration of the NIPALS algorithm consists of a linear regression of the columns of X on a score vector t to obtain a loading vector p, followed by a linear regression of the rows of X on the loading vector to obtain a new estimate of t. Convergence is reached when the mean square change in the scores falls below a threshold. When data in any column or row of X are missing, the iterative regressions are performed using the data that is present, with the missing points ignored. This procedure can be interpreted in different ways. It is equivalent to setting the residuals for all missing elements in the least squares objective function to zero, in each iteration. It can also be interpreted as replacing the missing values by their minimum distance projections onto the current estimate of the loading or score vector at each iteration (Martens & Naes, 1989).

As long as the number of variables present in any row or column is greater than or equal to the number of scores to be calculated, then the NIPALS algorithm can obtain a solution. However, in practice one should have many more observations than the scores or loadings being estimated to obtain reliable results. The NIPALS algorithm is usually recommended only when the missing data pattern is random rather than structured.

6.2.2 Single component projection algorithm for missing data in PLS

In the SCP method, let z be a new incomplete individual with only the last K-R variables measured, z^{*}. Consider z(0)=z and let $z^{*}(i-1)$ be the portion of $z^{*}(0)$ not explained by the first a-1 larger components. To estimate the ath element of the

vector score, τ_a (co-ordinate of the new observation in the a th component), the SCP method is based on the simple regression model.

$$z^*(i-1) = \tau_i P_i^* + e^*(i) \tag{6.19}$$

The SCP algorithm minimizes the sum of the squared prediction errors $e^{*T}(i)e^*(i)$, which yields

$$\hat{\tau}_i = \frac{P_i^{*T} z^*(i-1)}{P_i^{*T} P_i^*} \tag{6.20}$$

as the least square estimate of τ_i based on the observed variables. The portion of $z^*(i-1)$ explained by the ath component is then subtracted to yield the deflated object, $e^*(i) = z^*(i)$ and the next component $\hat{\tau}_{i+1}$ is then calculated analogously.

The expression for estimation error in the first score can be written as

$$\tau_1 - \hat{\tau}_1 = -(P_1^{*T} P_1^*)^{-1} P_1^{*T} \sum_{j=2}^{K} P_j^* \tau_j \tag{6.21}$$

In general, for the i th component (i=2,3…..A) as

$$\tau_i - \hat{\tau}_i = -(P_i^{*T} P_i^*)^{-1} P_1^{*T} \left(\sum_{j=1}^{i-1} P_j^* \left(\tau_j - \hat{\tau}_j\right) + \sum_{j=i+1}^{K} P_j^* \tau_j \right) \tag{6.22}$$

6.2.3 Error analysis for PLS

The analysis of score estimation error using an existing PLS model can be obtained by replacing the loading vector P_i^*, onto which the data is projected by w_i^*. The structure of the deflated data vector at the i th stage of the single component projection algorithm for PLS is:

$$z^*(i) = P^*\tau + e^* + d^* - \sum_{j=1}^{i-1} \hat{\tau}_j P_j^* \tag{6.23}$$

since PLS deflates using the p vectors. If there is no score estimation error then $\hat{\tau}_j = \tau_j$ and the first and last terms cancel after all A scores have been used in deflation. In this case, the residual data vector is composed of the sum of a vector of random error variables e^* and a deterministic remainder d^*. The d^* term arises because PLS does not necessarily use all the non-random information in the

independent data block that is of greater magnitude than the noise. Substituting this expression for $z^*(i)$ into the expression for $\hat{\tau}_i$, results in:

$$\tau_i - \hat{\tau}_i = \tau_i - (w_i^{*T} w_i^*)^{-1} w_i^{*T} z^*(i)$$

$$= \tau_i - (w_i^{*T} w_i^*)^{-1} w_i^{*T} \left(P^* \tau + e^* + d^* - \sum_{j=1}^{i-1} \hat{\tau}_j P_j^* \right)$$

$$= \tau_i - (w_i^{*T} w_i^*)^{-1} w_i^{*T} \left([P_1^*, P_2^*, \dots, P_K^*] \tau + e^* + d^* - \sum_{j=1}^{i-1} \hat{\tau}_j P_j^* \right)$$

$$= \tau_i (1 - (w_i^{*T} w_i^*)^{-1}) w_i^{*T} P_i^* -$$

$$\sum_{j=i+1}^{K} (w_i^{*T} w_i^*)^{-1} w_i^{*T} P_j^* \tau_j - (w_i^{*T} w_i^*)^{-1} w_i^{*T} e^* - (w_i^{*T} w_i^*)^{-1} w_i^{*T} d^*$$

$$- \sum_{j=1}^{i-1} (w_i^{*T} w_i^*)^{-1} w_i^{*T} P_j (\tau_j - \hat{\tau}_j) \tag{6.24}$$

When there are no missing measurements, the score estimation error reduces to

$$\tau_j - \hat{\tau}_j = -(w_i^T w_i)^{-1} w_i^T e - \sum_{j=1}^{i-1} (w_i^T w_i)^{-1} w_i^T P_j (\tau_j - \hat{\tau}_j) \tag{6.25}$$

the PLS score estimation error with no missing data has an error propagation term. This means that score estimation errors originating in measurement noise are transmitted to later scores. Errors propagate because the loading vector w_i is not required to be orthogonal to P_j when j is less than i, although deviations from orthogonality are penalized (Nelson et al., 1996).

6.3 Case study

6.3.1 Control Methodology

The end point control algorithm proposed by Cerrillo and MacGregor (2003) is an effective method for controlling product quality in a batch process, however the approach is based on linear PLS. In Chapter 5, the results showed that Linear PLS cannot provide a suitable prediction for penicillin. The reason is the relationship

between the substrate and penicillin is nonlinear and time varying. So in this thesis, the endpoint controller based on NNPLS is proposed.

In this chapter, the end-point controller is applied to regulate the biomass and penicillin in the Pensim simulation. The data requirements and model-building procedure are the same as those described in Section 6.1 for Cerrillo's end-point controller. The end-point controller based on NNPLS is applied. In order to compare the performance of the proposed controller, the endpoint controller based on PLS is also applied.

In proposed Endpoint controller, linear PLS is replaced by NNPLS. Before decision points, NNPLS is applied to build model. Because NNPLS is better than linear PLS, when the model is applied to predict the endpoint value, therefore the proposed Endpoint controller based on NNPLS is used to improve the control effect of the Endpoint value.

Following the development of the NNPLS models, the control methodology was considered in two stages. Firstly, the online and offline process measurements and MVTs available (before this decision point) are applied to predict the values of the future outputs at each decision point. To predict outputs $\hat{y}_{NN}^{T}$, the future measurements $X_{m,future}^{T}$, needed to be estimated.

The prediction is performed considering $u_{MV,future} = u_{MV,nominal}$(i.e. assuming that the remaining MVTs will be kept at their nominal conditions) using the NNPLS model:

$$\hat{t}_{NN}^{T} = [X_{Line}^{T} \quad u_{MV}^{T}]W_{NN}$$

$$= \left[X_{Line,measured,\theta_i}^{T} \quad X_{Line,future}^{T} \quad u_{MV,implemented,\theta_i}^{T} \quad u_{MV,nominal}^{T}\right]W_{NN} \quad (6.26)$$

$$u_{NN} = N(\hat{t}_{NN}^{T}) + r_{NN}$$

$$\hat{y}_{NN}^{T} = u_{NN} \cdot Q_{NN}^{T} \quad (6.27)$$

W_{NN} and Q_{NN} are projection matrices obtained from the NNPLS model building stage. The vector of scores, $\hat{t}_{NN}$, for the new batch is the projection of the x vector onto the reduced dimension space of the latent variable model at time θ_i and $\hat{y}_{NN}^{T}$ is

the vector of predicted end-quality properties. *N(·)* denote the nonlinear relation represented by a neural network, which is determined by minimizing the resideual r_{NN}. u_{NN} are score variable, they are obtained from the PLS outer model in the NNPLS approach.

The second stage was to regulate the end-point value by determining the necessary control action. Following this, the model was inverted to generate the required MTVs. The control algorithm was repeated at every decision point until the batch terminated.

Following the prediction procedure, the necessary changes in the scores (Δt_{NN}) were identified, that ensured that the predicted endpoint measurement matched the set-point, (y_{sp}). The value of Δt which minimized the following cost function was then determined:

$$\underbrace{min}_{\Delta t(\theta_i)} \left(\hat{y}_{NN}^T - y_{sp}\right)^T Q_{1N}\left(\hat{y}_{NN}^T - y_{sp}\right) + \Delta r_{NN}{}^T Q_{2N} \Delta r_{NN}$$

st

$$\Delta u_{NN} = N(\Delta t_{NN} + \hat{t}_{NN}) + \Delta r_{NN}$$

$$\hat{y}_{NN}^T = \Delta u_{NN} \cdot Q_{NN}^T \qquad (6.5)$$

$$\Delta t_{NNmin} \leq \Delta t_{NN} \leq \Delta t_{NNmax}$$

where $\Delta t_{NN} = \boldsymbol{t}^T - \hat{t}_{NN}^T$, Q_{1N} is a diagonal weighting matrix defining the relative importance of the variables *y*'s; Q_{2N} is a diagonal movement suppression matrix that is used as a tuning matrix to moderate the aggressiveness of the control. $\Delta \boldsymbol{t}_{NNmin}$ and $\Delta \boldsymbol{t}_{NNmax}$ are the constraints which define the minimum and maximum values for Δt_{NN}. Δr_{NN} is applied to determine the neural network. This final constraint is included to limit the action of the control system.

6.3.2 Application of the End-point Controller in Pensim Simulation

In order to assess and compare the performance of the addressed control approaches, the end-point controller is applied in a benchmark (Pensim). More detail can be found in Chapter 5. The objective for each of the considered approaches is to control the end-point biomass and penicillin concentration through manipulating substrate feed rates. The training data consisted of 20 batches, with 20 batches used for testing each model. Each batch was allowed to run for 200 samples, with a sample time of 1 hour.

For the end-point controller, 20 batches of data were used to build the PLS model and NNPLS mode, and another 20 batches were applied to test the model. Cross validation method was used to choose the latent variables. The predicted and actual outputs (the biomass or Penicillin) are compared in Chapter 5. The results show that the NNPLS model and PLS model can be applied to predict the end-point Biomass very well. When the end-points of the Penicillin are predicted, the NNPLS model can provide more accurate predicted results than the linear PLS model.

The end-point controller is applied to control the end-point value of the quality variables (biomass and penicillin). Three statements were considered to evaluate their performance for controlling the value of end-point: control the end-point value in nominal target, control the end-point value in modified target, and control the end-point value under additional disturbance and measurement noises. The ability of tracking a changing set-point target at the end-point and rejecting disturbance is important for a controller in practice, since changing demands and varying disturbances often happen among/within batch runs. The end-point controller is firstly applied to control the end-point value of the biomass.

6.3.2.1 The End-Point Control of Biomass Concentration to the Nominal Target Value

20 testing batches were collected. Pseudo-Random Binary Signals (PRBS) were applied to the nominal feed-rate of substrate, in order to excite process dynamics. The more detail of the PRBS can be found in Chapter 5. The biomass trajectories of these nominal testing data are presented in Figure 6.2.

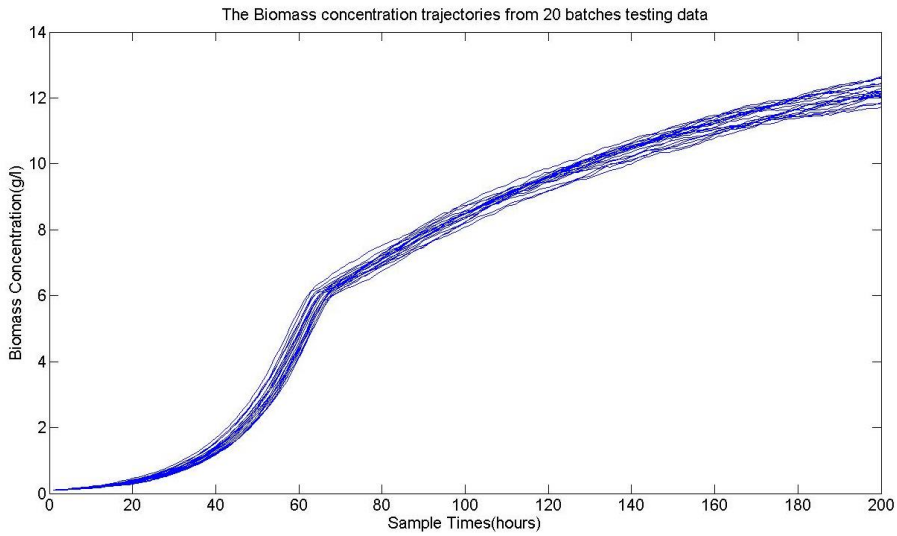

Figure 6.2 The Biomass Concentration Trajectories from 20 testing batches

The end-point controllers based on NNPLS and PLS are applied to control the end-point value of the biomass concentration. In the end-point controller, the control decision points are at 70, 100, 130 and 160 sample times. The effect of selecting the decision point is discussed in the following part. The purposed value of the end-point value of the biomass concentration is set to 12 g/l. For the control results for controlling the end-point value of 20 testing batches, see Figure 6.3. In Figure 6.3, the red star points represent the Target value (12 g/l). The blue circle points represent the end-point value in open loop. The Black diamond points highlight the controlled end-point value in the end-point controller based on NNPLS. The magenta plus points highlight the controlled end-point value in the end-point controller based on PLS. Controllers are applied to control the end-point value of the biomass concentration. In order to compare the accuracy of the controllers, the sum squared of error in 20 batch end-point value of the biomass (SSE_{Bio}) are calculated and listed in Table 6.1.

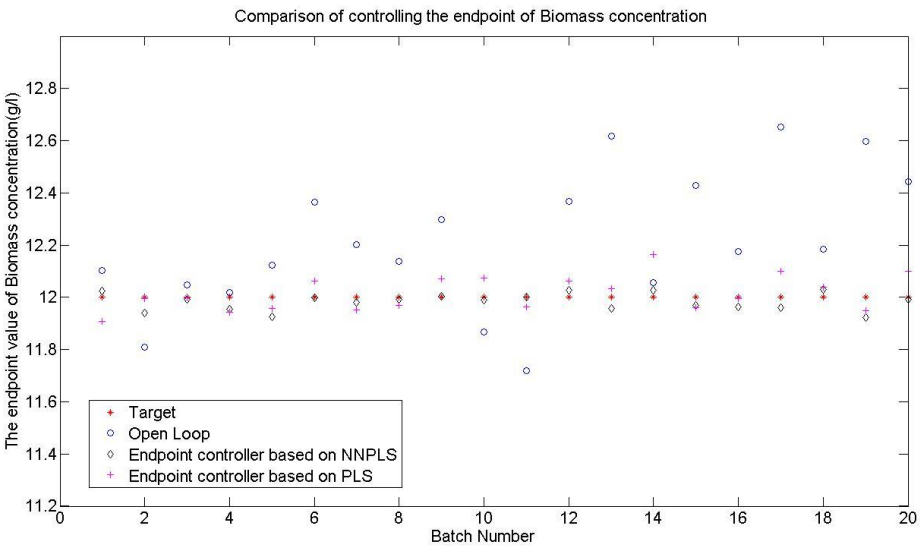

Figure 6.3 Controlling the End-point value of Biomass Concentration in Nominal Target

Table 6.1 The SSE of the End-point value of the Biomass Concentration in Nominal Target (20 testing batches)

Controller	SSE_{Bio}
Open Loop	2.1886
The end-point controller based on PLS	0.0899
The end-point controller based on NNPLS	0.0263

20 testing batches were applied to calculate the standard deviation of biomass end-point measurement; the standard deviation is 0.3308 under open-loop control and

0.0363 and 0.0670 under the end-point controller based on NNPLS and PLS. The results show that both end-point controllers can control the nominal target end-point value of the biomass concentration. These results therefore indicate that the NNPLS-based end-point controller has slightly less variation in the end-point of the biomass, than the PLS-based end-point controller.

To compare with open loop and different controller, the corresponding trajectories for the manipulated substrate feed rate are shown in Figure 6.4. In Figure 6.4, one testing batch is shown. The open-loop substrate feeding rate is kept constant at 0.0045. The blue dashed line represents the manipulated substrate feed rate in open loop. The Black line highlights the manipulated substrate feed rate value in the end-point controller based on NNPLS. The green line highlights the manipulated substrate feed rate in the PLS-based end-point controller

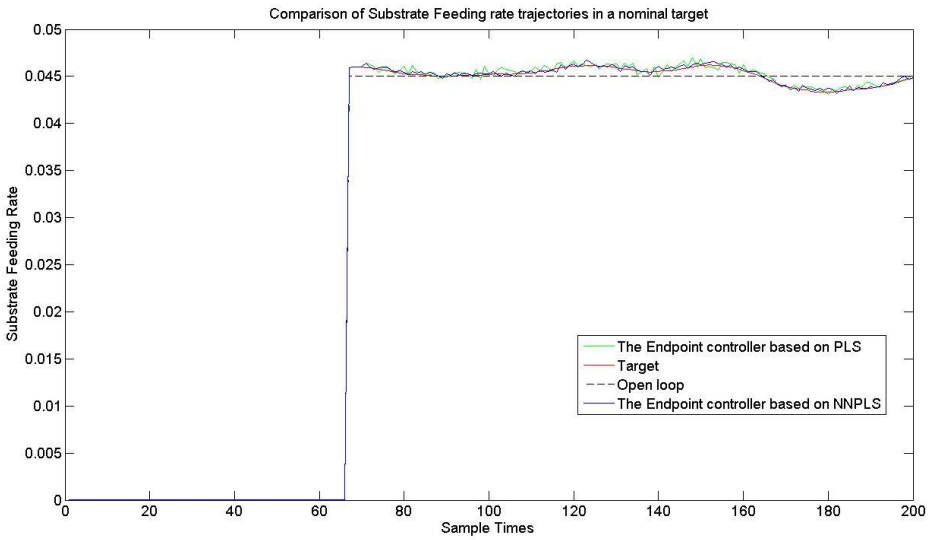

Figure 6.4 The Corresponding Trajectories for the Manipulated Substrate Feed Rate in Nominal Target (Biomass)

The sum squared of error in substrate trajectory ($SSE_{Substrate}$) are calculated. The values are 2.7526e-05 in the endpoint controller based on PLS and 8.7448e-06 in the endpoint controller based on NNPLS. In this case, the results are shown that the endpoint controller based on NNPLS is better than the endpoint controller based on PLS.

The selection of different decision points is also very important and this too was tested. Various decision points were used in the end-point controller and the performance compared (Table 6.2).

Table 6.2 Comparison of Control Performance with Different Decision Points (End-point controller based on NNPLS)

Decision Points	The sum squared of error in 20 batch End-point Value of the biomass
50	0.0748
100	0.0679
150	0.0723
200	0.1524
70, 100	0.0598
100, 130	0.0574
130, 160	0.0613
70, 100, 130	0.0394
100, 130, 160	0.0416
70, 100, 130, 160	0.0263

Table 6.2 shows that when a simple decision point is applied, the controller performances were similar, until the decision point was increased to 200. At this point, the consistency and performance of the controller reduced. The reason for this is that selecting a decision point too close to the end of the batch will mean there is insufficient time for the process variables to adjust. When a simple decision point is applied, it is shown that the decision point of 100 samples times provides the most accurate results; therefore, a number of decision points are added near this sample's time. There are no general guidelines for the selection procedure of the decision point. In this thesis, 4 decision points were applied. The decision points were selected at sample number 70, 100, 130 and 160. This is because when selecting these decisions points, the controller can provide the most accurate results.

6.3.2.2. The End-Point Control of Biomass Concentration to the Modified Target Value

In this part, 20 testing batches were collected and applied to test controller performance. The Biomass Concentration Trajectories 20 testing batches are shown in Figure 6.5. PRBS were applied to the nominal feed-rate of substrate, in order to excite process dynamics. The end-point controllers based on NNPLS and PLS are applied to control the end-point value of the biomass concentration. In the end-point controller, the control decision points are used in 70, 100, 130 and 160 sample times. The target end-point value of the biomass concentration is changed from 12 g/l to 12.8g/l. The control results for controlling the end-point value of 20 testing batches are presented in Figure 6.6. In Figure 6.6, the red star points represent the Target value (12.8 g/l). The blue circle points represent the end-point value in open loop. The Black diamond points highlight the controlled end-point value in the End-point controller based on NNPLS. The magenta plus points highlight the controlled end-point value in the end-point controller based on PLS. Controllers are applied to control the end-point value of the biomass concentration. In order to compare the accuracy of the controllers, SSE_{Bio} are calculated and listed in Table 6.3.

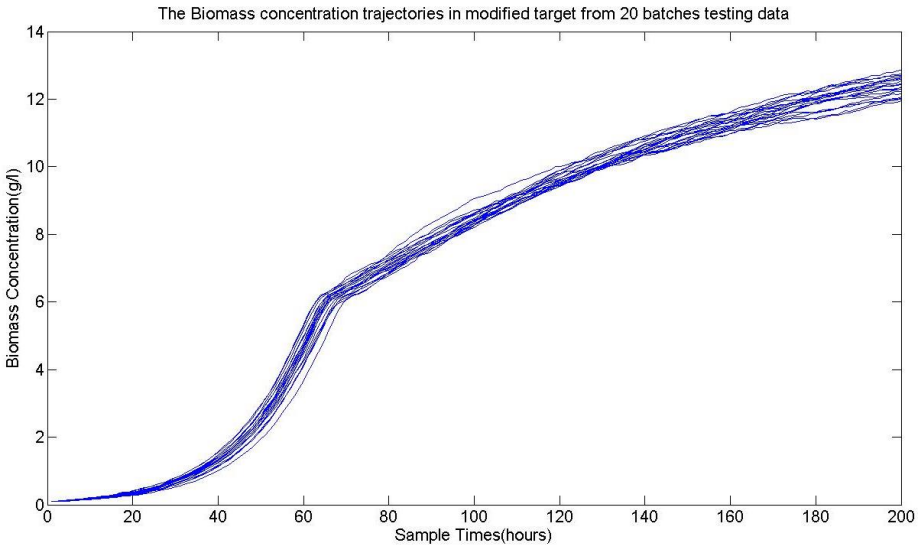

Figure 6.5 The Biomass Concentration Trajectories in Modified target from 20 testing batches

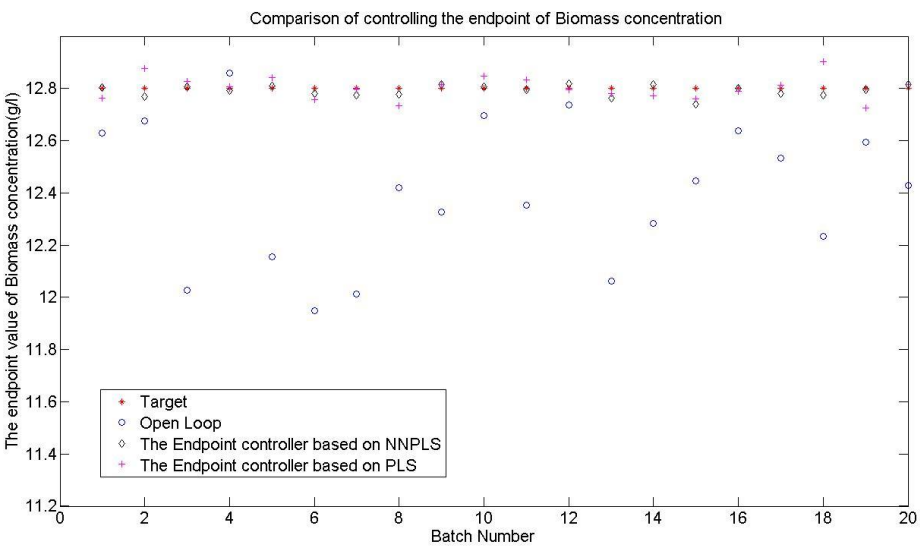

Figure 6.6 Control Results for 20 testing batches end-point value of the Biomass Concentration in Modified Target

Table 6.3 The SSE of the End-point value of the Biomass Concentration in Modified Target (20 testing batches)

Controller	SSE_{Bio}
Open Loop	4.2561
The end-point controller based on PLS	0.0383
The end-point controller based on NNPLS	0.0097

20 testing batches were applied to calculate the standard deviation of biomass end-point measurement; the standard deviation is 0.4613 under open-loop control and 0.0220 and 0.0438 under the end-point controller based on NNPLS and PLS.

These results show that the NNPLS-based end-point controller can be applied to track a changing set-point; the NNPLS-based end-point controllers can also provide better performance than the PLS-based end-point controller.

6.3.2.3. The End-Point Control of Biomass Concentration in the Presence of Noise and Disturbance

In this part, the end-point controller is applied to control the end-point value of the biomass concentration under additional disturbance on Substrate Concentration, and measurement noises on biomass concentration, Carbon dioxide and Dissolved Oxygen concentration. Normally distributed disturbances and noise were added to the simulation; these disturbances were introduced as white noise sequences with a standard deviation of 0.1, 0.2, 0.05, 0.05, and were applied to the biomass growth constant, to the carbon dioxide evolution rate, and to the feed-rates of the base and cooling water respectively.

In the Pensim simulation, the initial value of the substrate concentration is 15g/l. However, as the properties of raw materials are not always kept the same, the substrate concentration can be different from the nominal value; therefore the initial value of substrate concentration is replaced from 15g/l to 13g/l. The biomass concentration trajectory will be changed; one batch comparison results are seen in Figure 6.7. The end-point value of the biomass changed from 12.3652 to 11.4109.

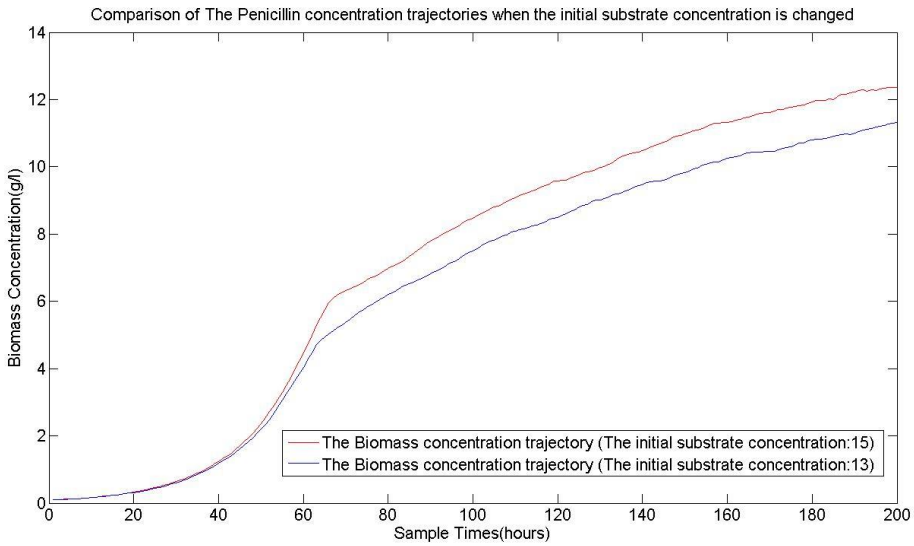

Figure 6.7 Comparison of The Biomass Concentration Trajectories when the Initial Substrate Concentration is Changed

20 testing batches are collected and applied to test the performance of the controller. The biomass trajectories of these testing data are shown in the Figure 6.8. PRBS

were applied to the nominal feed-rate of substrate, in order to excite process dynamics.

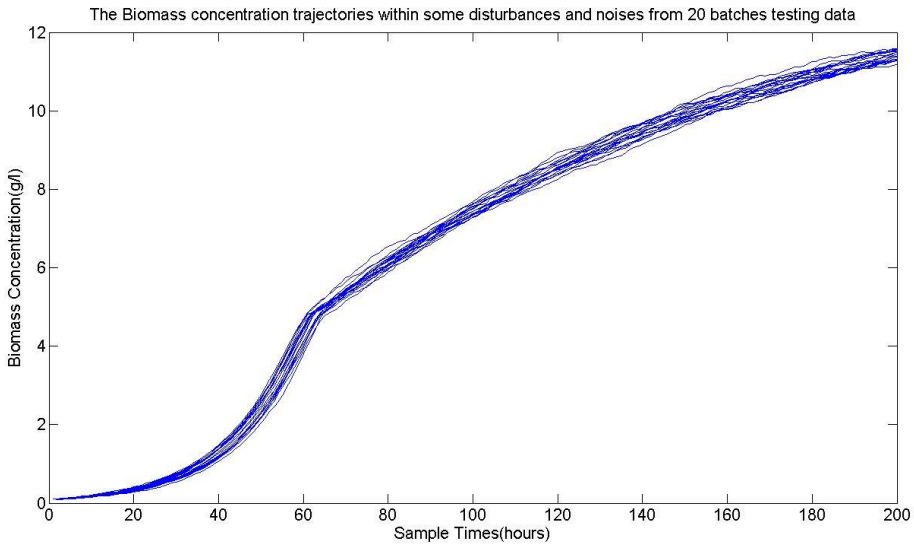

Figure 6.8 The Biomass Concentration Trajectories within Some Disturbances and Noises form 20 testing batches

In the end-point controller, the control decision points are at 70, 100, 130 and 160 sample times. The target end-point value of the biomass concentration is set to 12 g/l. The control results for controlling the end-point value of 20 testing batches are presented in Figure 6.9. In Figure 6.9, the red star points represent the Target value (12 g/l). The blue circle points represent the end-point value in open loop. The Black diamond points highlight the controlled end-point value in the end-point controller based on NNPLS. The magenta plus points highlight the controlled end-point value in the end-point controller based on PLS. Controllers are applied to control the end-point value of the biomass concentration. In order to compare the accuracy of the controllers, SSE_{Bio} are calculated and listed in Table 6.4.

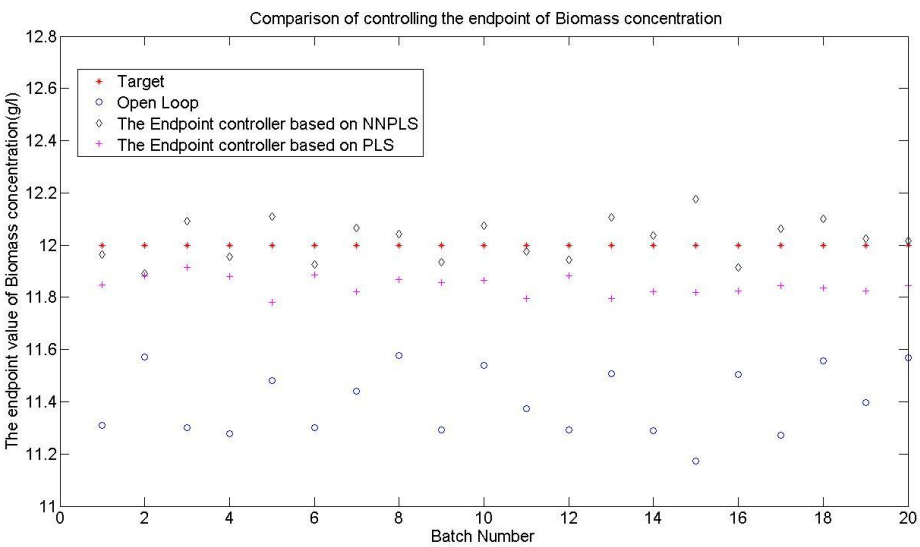

Figure 6.9 Control Results for 20 testing batches end-point value of the Biomass Concentration within Some Disturbances and Noises

Table 6.4 The SSE of the End-point value of the Biomass Concentration within some Disturbances and Noises (20 testing batches)

Controller	SSE_{Bio}
Open Loop	7.4857
The end-point controller based on PLS	0.5086
The end-point controller based on NNPLS	0.1265

20 testing batches were applied to calculate the standard deviation of biomass end-point measurement; the standard deviation is 0.6118 under open-loop control and 0.0825 and 0.1595 under the end-point controller based on NNPLS and PLS.

The results show that the NNPLS-based end-point controller has the ability to reject these disturbances and noises. Furthermore, the NNPLS-based end-point controller

performs slightly better than the PLS-based end-point controller in terms of tracking errors.

In the Pensim simulation, the penicillin is another quality variable. In the following part, the end-point controller is applied to control the end-point of the Penicillin. Three statements were similar to consider: control the end-point value of the penicillin concentration in nominal target, control the end-point value of the penicillin concentration in modified target, and control the end-point value of the Penicillin concentration under additional disturbance and measurement noises.

6.3.2.4. The End-Point Control of Penicillin Concentration to the Nominal Target Value

The Penicillin trajectories of 20 testing batches are shown in Figure 6.10. Pseudo-Random Binary Signals (PRBS) were applied to the nominal feed-rate of substrate, in order to excite process dynamics.

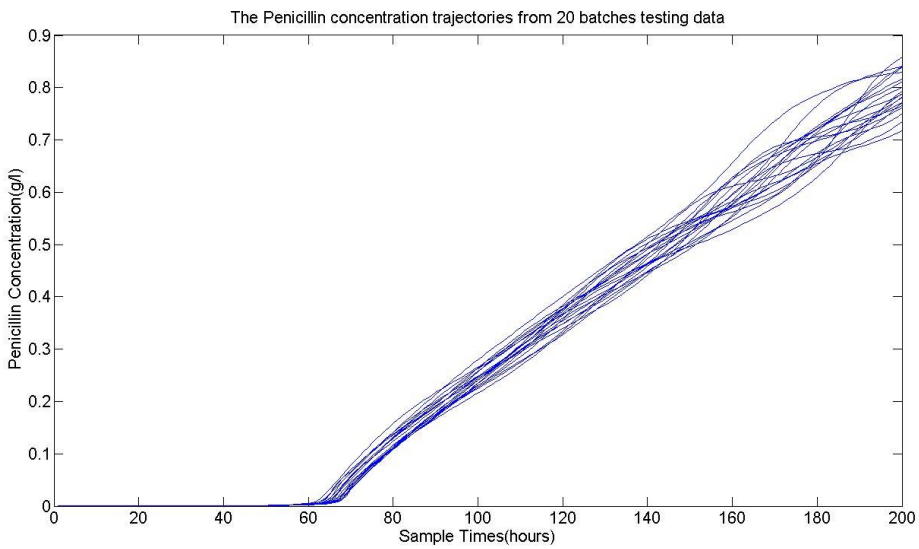

Figure 6.10 The Penicillin Concentration Trajectories from 20 testing batches

Both NNPLS-based and PLS-based end-point controllers were applied to control the end-point value of the Penicillin concentration. In the end-point controller, the control decision points are at 70, 100, 130 and 160 sample times. The target value of the end-point value of the penicillin concentration is set to 0.75 g/l. The control results for controlling the end-point value of 20 testing batches are presented in

Figure 6.11. In Figure 6.11, the red star points represent the Target value (0.75 g/l). The blue circle points represent the end-point value in open loop. The Black diamond points highlight the controlled end-point value in the end-point controller based on NNPLS. The magenta plus points highlight the controlled end-point value in the PLS-based end-point controller. Controllers are applied to control the end-point value of the penicillin concentration. In order to compare the accuracy of the controllers, the sum squared of error in 20 batches end-point value of the Penicillin (SSE_{Pen}) are calculated and listed in Table 6.5.

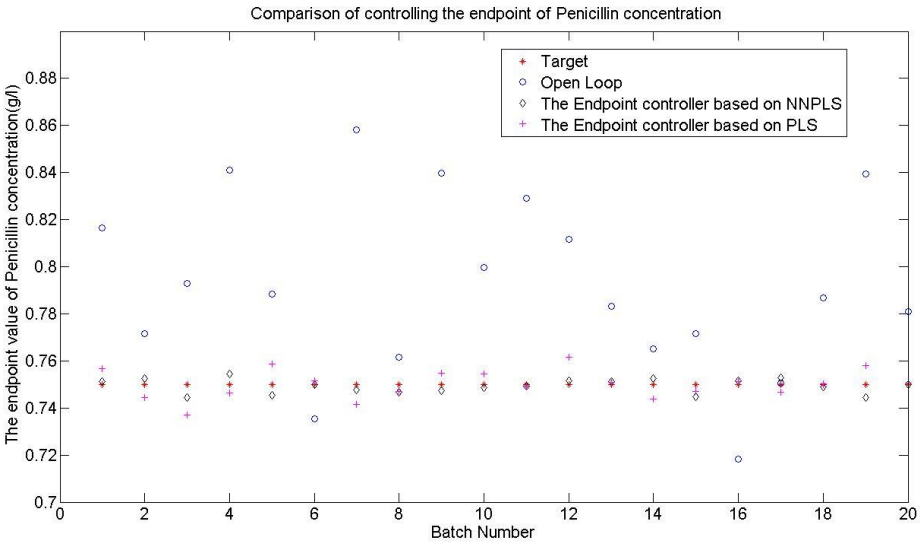

Figure 6.4 Controlling the End-point value of Penicillin Concentration in Nominal Target

Table 6.5 The SSE of the End-point value of the Penicillin Concentration in Nominal Target (20 testing batches)

Controller	SSE_{Pen}
Open Loop	0.0622
The end-point controller based on PLS	0.0012
The end-point controller based on NNPLS	1.8753e-04

20 testing batches were applied to calculate the standard deviation of Penicillin end-point measurement; the standard deviation is 0.0558 under open-loop control and 0.0031 and 0.0077 under the end-point controller based on NNPLS and PLS. These results therefore indicate that the NNPLS-based end-point controller can control the endpoint value of the Penicillin, and that it has slightly less variation in the end-point of the Penicillin than the PLS-based end-point controller.

To compare with open loop and different controller, the corresponding trajectories for the manipulated substrate feed rate are shown in Figure 6.12. In Figure 6.12, one testing batch is shown. The open-loop substrate feeding rate is kept constant at 0.0045. The blue dashed line represents the manipulated substrate feed rate in open loop. The Black line highlights the manipulated substrate feed rate value in the end-point controller based on NNPLS. The green line highlights the manipulated substrate feed rate in the PLS-based end-point controller.

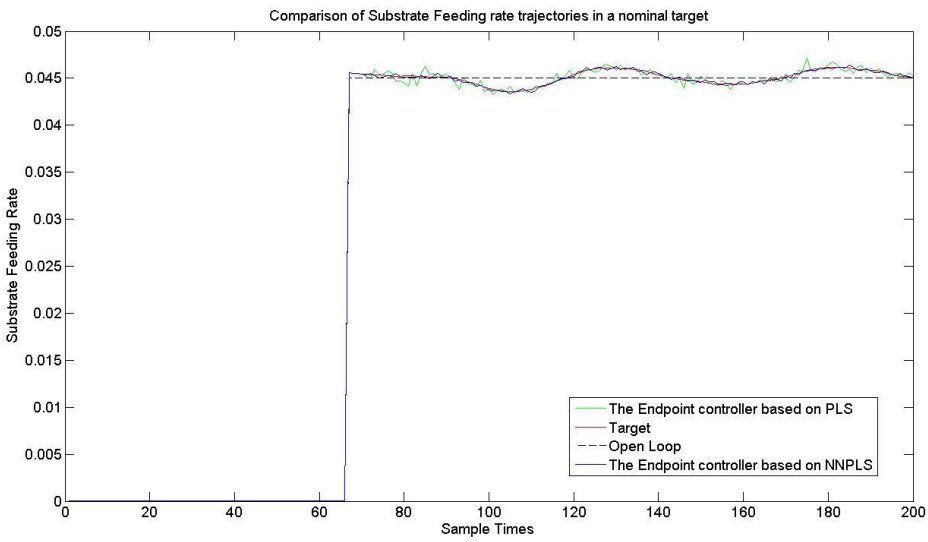

Figure 6.12 The Corresponding Trajectories for the Manipulated Substrate Feed Rate in Nominal Target (Penicillin)

The $SSE_{Substrate}$ are calculated. The values are 2.9781e-04 in the endpoint controller based on PLS and 1.0697e-04 in the endpoint controller based on NNPLS. In this case, the results are shown that the endpoint controller based on NNPLS is better than the endpoint controller based on PLS.

6.3.2.5. The End-Point Control of Penicillin Concentration to the Modified Target Value

In this part, 20 testing batches were collected and applied to test controller performance. The Penicillin Concentration Trajectories 20 testing batches are shown in Figure 6.13. PRBS were applied to the nominal feed-rate of substrate, in order to excite process dynamics. The end-point controllers based on NNPLS and PLS were applied to control the end-point value of the penicillin concentration. In the end-point controller, the control decision points are at 70, 100, 130 and 160 sample times. The target end-point value of the penicillin concentration was changed from 0.75 g/l to 0.82 g/l. The control results for controlling the end-point value of 20 testing batches are presented in Figure 6.14. In Figure 6.14, the red star points represent the Target value (0.82 g/l). The blue circle points represent the end-point value in open

loop. The Black diamond points highlight the controlled end-point value in the end-point controller based on NNPLS. The magenta plus points highlight the controlled end-point value in the End-point controller based on PLS. Controllers are applied to control the end-point value of the penicillin concentration. In order to compare the accuracy of the controllers, SSE_{Pen} are calculated and listed in Table 6.6.

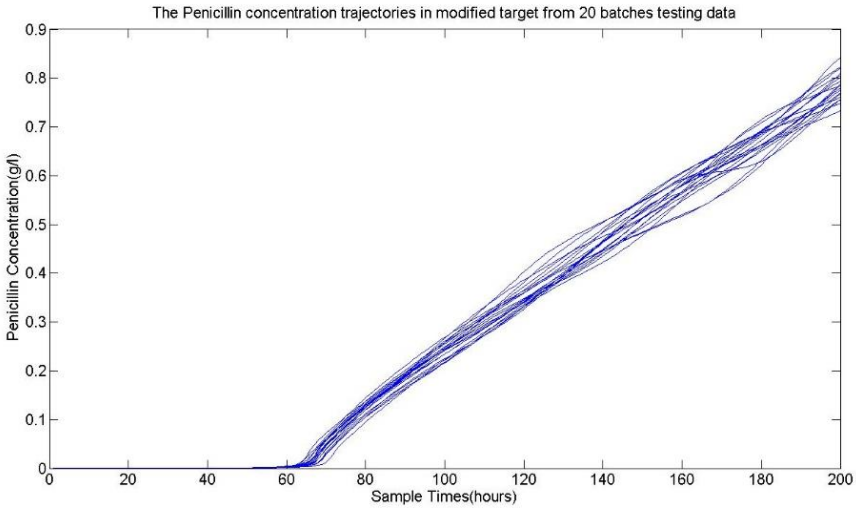

Figure 6.5 The Penicillin Concentration Trajectories in Modified target from 20 testing batches

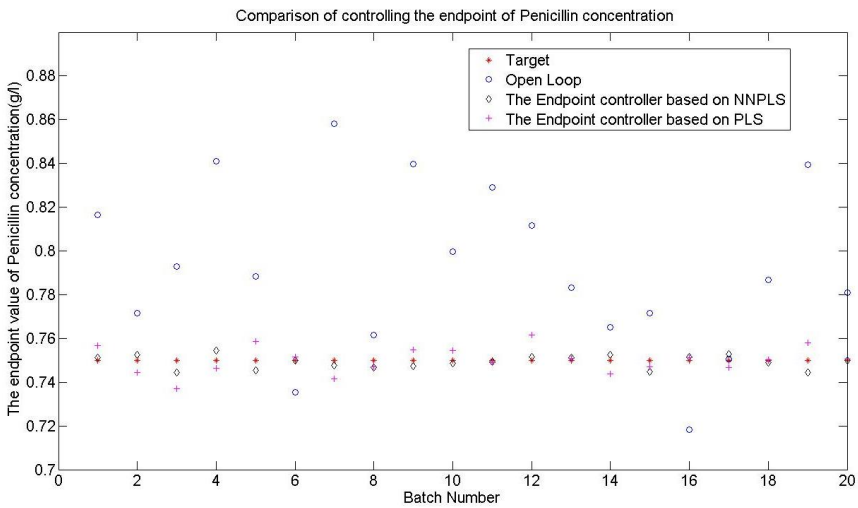

Figure 6.6 Control Results for 20 testing batches end-point value of the Penicillin Concentration in Modified Target

Table 6.6 The SSE of the End-point value of the Penicillin Concentration in Modified Target (20 testing batches)

Controller	SSE_{Pen}
Open Loop	0.0766
The end-point controller based on PLS	0.0030
The end-point controller based on NNPLS	6.1360e-04

20 testing batches were applied to calculate the standard deviation of Penicillin end-point measurement; the standard deviation is 0.0619 under open-loop control and 0.0055 and 0.0122 under the end-point controller based on NNPLS and PLS. The results are obvious that the ability to adapt to the modified end-point target of the penicillin concentration has changed, where the end-point controller based on NNPLS can provides the better control in that case.

6.3.2.6. The End-Point Control of Penicillin Concentration in the Presence of Noise and Disturbance

In this part, the end-point controller is applied to control the end-point value of the penicillin concentration under additional disturbance on Substrate Concentration, and measurement noises on biomass concentration, Carbon dioxide and Dissolved Oxygen concentration. Normally distributed disturbances and noise were added to the simulation; these disturbances were introduced as white noise sequences, with a standard deviation of 0.1, 0.2, 0.05, 0.05, and were applied to the biomass growth constant, to the carbon dioxide evolution rate, and to the feed-rates of the base and cooling water respectively.

In the Pensim simulation, the initial value of the substrate concentration is 15g/l. However, as the properties of raw materials are not always kept the same, the substrate concentration can be different from the nominal value; therefore the initial value of substrate concentration is replaced from 15g/l to 13g/l. The penicillin concentration trajectory was changed; one batch comparison results are shown in Figure 6.15. The end-point value of the penicillin concentration changed from 0.7998 to 0.6858.

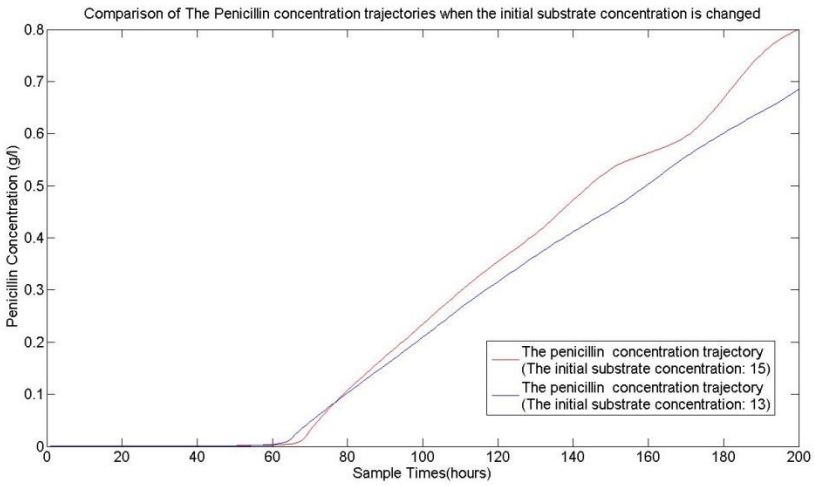

Figure 6.7 Comparison of The Penicillin Concentration Trajectories when the Initial Substrate Concentration is changed

20 testing batches were collected and applied to test the performance of the controller. PRBS were applied to the nominal feed-rate of substrate, in order to excite process dynamics. The penicillin trajectories of these testing data are shown in Figure 6.16.

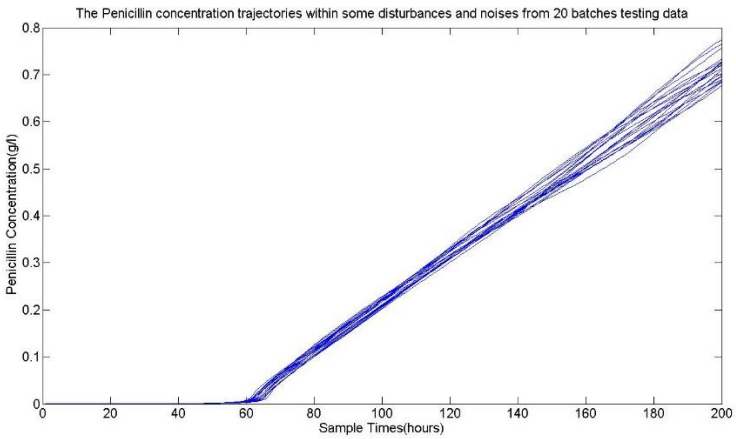

Figure 6.8 The Penicillin Concentration Trajectories within Some Disturbances and Noises form 20 testing batches

In the end-point controller, the control decision points are at 70, 100, 130 and 160 sample times. The target end-point value of the penicillin concentration is set to 0.75 g/l. The control results for controlling the end-point value of 20 testing batches are presented in Figure 6.17. In Figure 6.17, the red star points represent the Target value (0.75 g/l). The blue circle points represent the end-point value in open loop. The Black diamond points highlight the controlled end-point value in the End-point controller based on NNPLS. The magenta plus points highlight the controlled end-point value in the End-point controller based on PLS. Controllers are applied to control the end-point value of the penicillin concentration. In order to compare the accuracy of the controllers, SSE_{Pen} are calculated and listed in Table 6.7.

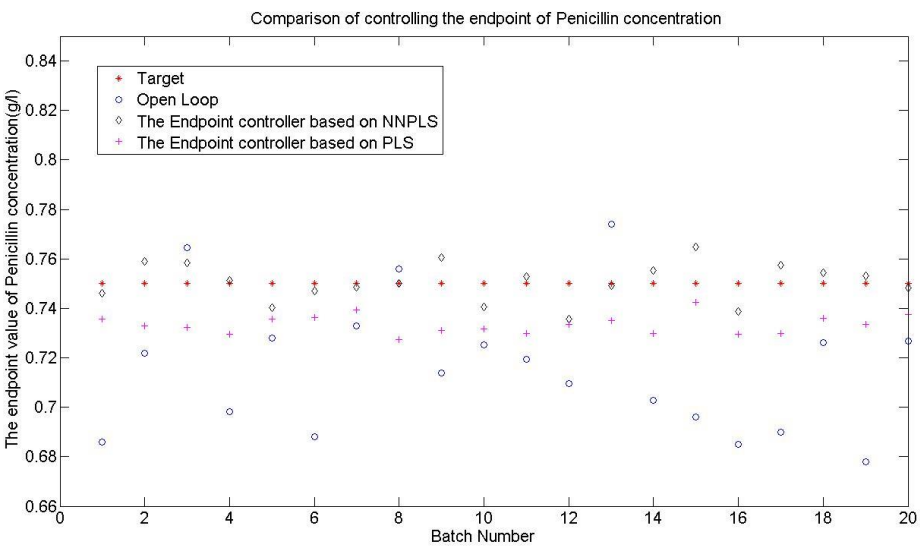

Figure 6.9 Control Results for 20 testing batches End-point value of the Penicillin Concentration within Some Disturbances and Noises

Table 6.7 The SSE of the End-point value of the Penicillin Concentration within Some Disturbances and Noises (20 testing batches)

Controller	SSE_{Pen}
Open Loop	0.0368
The end-point controller based on PLS	0.0058
The end-point controller based on NNPLS	0.0012

20 testing batches were applied to calculate the standard deviation of Penicillin end-point measurement; the standard deviation is 0.0429 under open-loop control and 0.0077 and 0.0172 under the end-point controller based on NNPLS and PLS.

The results show that the NNPLS-based end-point controller has the ability to reject these disturbances and noises in this case. For quality variable (Penicillin

concentration), the performance of the NNPLS-based end-point controller is better than the PLS-based end-point controller at the end-point sample time.

6.4 Summary

In this chapter, Nonlinear PLS was applied to a benchmark simulation: Pensim simulation. The end-point control algorithm was then introduced. Based on this method, PLS and NNPLS were used to build the controller. The controller is applied to control the end-point value of the quality variables (Biomass concentration and penicillin concentration). The results show that this proposed NNPLS controller is able to be applied in the batch process.

The NNPLS-based end-point controller can be applied to track a changing set-point at the end-point; the results showed the performance of the NNPLS controller is very accurate. When some process disturbance and some noises are considered, the NNPLS-based end-point controller has the ability to reject these disturbances and noises. For quality variable (Penicillin or Biomass concentration), the performance of the NNPLS-based end-point controller is better than the performance of the PLS-based end-point controller at the end-point sample time.

Chapter 7

Conclusion and Future Work

This chapter firstly provides conclusions of the work presented in this thesis and then suggests lines for future research in this field.

7.1 Summary and Conclusions

Chapter 2 presented a literature review on the application of multivariate statistical analysis methods, and summarized the previous work on multivariate statistical process control (MSPC) methods and applications. Chapter 3 described some basic algorithms of MSPC techniques, such as Partial Least Square (PLS) and discussed several of PLS's extensions, including multi-way PLS (MPLS), nonlinear PLS and neural network PLS (NNPLS).

In Chapter 4, the limitations of linear PLS are discussed, with PLS being applied to predict linear and nonlinear systems. When PLS is applied to batch processes, a technique referred to as multi-way PLS (MPLS) is frequently applied. This technique analyses process behaviour relative to the mean trajectories of the process variables. In doing so, a major nonlinearity in the data is removed. The results showed, as expected, that linear MPLS can predict linear systems very well; however, this algorithm was not able to track the dynamics contained in the non-linear systems. To overcome this deficiency, several nonlinear extensions were proposed to enable it to better handle nonlinear systems. The nonlinear PLS model can be divided into Type I and Type II nonlinear PLS model. In the Type I Nonlinear PLS method, the observed variables were appended with nonlinear transformations. In contrast to the Type I nonlinear PLS method, the Type II nonlinear PLS method assumes a nonlinear relationship within the model's latent variable structure. Type I and II non-linear structures are integrated within MPLS models to enable them to more accurately approximate nonlinear batch processes. In this thesis, higher order terms of Type II nonlinear MPLS model were considered and applied. 2nd, 4th and 6th order

Type II nonlinear MPLS were applied to test the 4th, 5th and 6th order nonlinear systems. The results showed that, although the Type II nonlinear MPLS model can predict some nonlinear systems, the limitation of this model is that the testing system needs to be known a-priori. Most industrial system can be pre-determined; therefore the NNPLS model is applied. To illustrate the capabilities of NNPLS, multi-way NNPLS and 6th Type II nonlinear MPLS were applied to predict the 7th order nonlinear testing systems. The results showed that multi-way NNPLS is a better method to use to model the endpoint value of the nonlinear system, in comparison to the Type II nonlinear MPLS. MPLS do not have a number of limitations.

In Chapter 5, the linear MPLS model, the Type I and Type II nonlinear MPLS models and the multi-way NNPLS model were applied in a benchmark simulation of the penicillin batch fermentation process. In this batch process, the substrate feed-rate was the primary manipulated variable; it affected two primary quality output variables: Biomass and Penicillin. The relationships between the quality output variables and the manipulated variable were tested. The results showed that the relationship between the substrate feed-rate and the Biomass concentration is linear; and between the substrate feed-rate and Penicillin concentration, the relationship is linear.

Linear MPLS is firstly applied to predict the endpoint value of the biomass and penicillin. Comparing the results of the average error of testing batches and training batches, MPLS can predict the biomass very well. It proved that linear MPLS can model linear system very well. On the other hand, linear MPLS cannot provide a suitable prediction of the endpoint of penicillin, as the relationship between the substrate and penicillin is nonlinear.

To predict the endpoint value of Penicillin, some nonlinear MPLS model were proposed and applied. In comparison to linear MPLS, the results illustrated that the Type I nonlinear MPLS model did not significantly improve prediction accuracy. This is because in the Type I nonlinear model, the expansion of the X matrix is only considered with the quadratic term x^2. Therefore, the Type II nonlinear MPLS model was applied to predict the endpoint value of Penicillin.

To show the limitations and capabilities of the Type II nonlinear MPLS model, linear MPLS and Type II nonlinear MPLS were applied to estimate the final end-point concentration of penicillin. In this thesis, higher order terms of the Type II nonlinear MPLS model were considered and applied. 2nd, 4th and 6th order Type II nonlinear MPLS were applied to predict the endpoint value of penicillin, respectively. This result showed that the Type II nonlinear MPLS model can provide a more accurate prediction than linear PLS, and as the order of the model improves, so too does the accuracy of the model.

To illustrate the benefit of using multi-way NNPLS, the multi-way NNPLS model was also applied to predict the same testing data. The results showed the Multi-way NNPLS model can provide a better prediction than the 6th order Type II nonlinear MPLS. The primary advantage in using the multi-way NNPLS is that it provides improved accuracy without the need to determine the order for the model; this can be a critical parameter with Type II MPLS models.

In Chapter 6, a novel endpoint control based on the NNPLS method was proposed, and applied to batch process. The proposed controller was used to control the endpoint biomass and penicillin. The results showed that the proposed controller can precisely control the endpoint value at a set-point. And the proposed controller can be applied to track a changing set-point value. And when some process disturbance and some noises are considered, the Endpoint controller based on NNPLS has the ability to reject these disturbances and noises. The proposed NNPLS controller was presented in comparison to the endpoint controller based on PLS; the performance of the proposed NNPLS controller was better at the endpoint of the batch.

7.2 Recommendations for Future Work

(1) In the end-point controller, an important aspect is the selection of the decision points, as they will directly affect the controller performance. Interestingly though, there are no general guidelines for the selection procedure; therefore, this issue needs to be further researched.

(2) In Chapter 6, the endpoint controller based on NNPLS was applied in the Pensim simulation. The major purpose of this proposed controller is to maintain the endpoint value at a set-point. This proposed controller probably will not regulate the within batch quality variable very well. The next step for future study will therefore be to apply NNPLS to design the controller; the purpose of a new controller would be applied to control the whole batch of quality variables.

Reference

1. Aguado, D., Ferrer, A. and Seco, J. " Comparison of different predictive models for nutrient estimation in sequencing batch reactor for wastewater treatment". *Chemometrics and Intelligent Laboratory Systems,* 84:75–81, 2006.
2. Alt, F. B., Multivariate Quality Control. In Johnson, S. (eds) Encyclopaedia of Statistical Science. John Wiley, 110-122, New York, 1985.
3. Alwan, L. C. and Roberts, H. V. "The problem of misplaced control limits". *Journal of the Royal Statistical Society,* 44(3):268-278, 1995.
4. Alwan, L. C. and Roberts, H. V. "Time-series modeling for statistical process control". *Journal of Business and Economic Statistics,* 6: 87–95, 1988.
5. Arteaga, F. and Ferrer, A. "Dealing with missing data in MSPC: several methods, different interpretation, some examples". *Journal of Chemometrics,* 16: 408–418, 2002.
6. Baffi,G., Martin, E. and Morris, A. "Non-linear dynamic projection to latent structures modeling". *Chemometrics and Intelligent Laboratory Systems,* 52: 5-22,2000.
7. Baffi,G., Martin, E. and Morris, A. "Non-linear projection to latent structures revisited (the neural network PLS algorithm)". *Computers and Chemical Engineering,* 23:1293–1307, 1999.
8. Bailey, J. E. and Oillis. D. F., Biochemical Engineering Fundamentals. McGraw Hill, New York, 1986.
9. Bair, E., Hasie, T., Paul, D. and Tibshirani, R. "Prediction by supervised principal components". *Journal of the American Statistical Association,* 101:119-137, 2006.
10. Bakshi, B. R. "Multiscale PCA with application to multivariate statistical process monitoring". *AIChE Journal,* 44: 1596–1610, 1998.
11. Banks, J., Principles of Quality Control. John Wiley, New York, 1989.
12. Barlow, R. E. and Irony, Z. "Foundations of statistical quality control." *Lecture Notes-Monograph Series,* 99-112, 1992.
13. Barnard, G. A. "Control charts and stochastic processes". *Journal of the Royal Statistical Society*. Series B (Methodological), 239-271,1959.

14. Berglund, A. and Wold, S. *"*A serial extension of Multiblock PLS*".Journal of Chemometrics,* 13: 461-471, 1999.

15. Berglund, A. and Wold, S. "INLR implicit non-linear latent variable regression". *Journal of Chemometrics*, 11:141-156, 1997.

16. Bersimis, S., Psarakis, S. and Panaretos, J. "Multivariate statistical process control charts: An overview". *Quality and Reliability Engineering International,* 23:517–543, 2007.

17. Billings, S.A. and Fadzil, M. B. "The practical identification of systems with nonlinearities". *Proceedings of IFAC System Identification and Parameter Estimation*, 155-160, York, 1985.

18. Billings, S. A. and Voon, W. S. F. "A prediction error and stepwise regression algorithm for nonlinear system". *International Journal of Control*, 44 (3):803-822, 1983.

19. Billings, S. A. and Voon, W. S. F. "Correlation based model validity tests for nonlinear models". *International Journal of Control*, 44 (1):235-244, 1986.

20. Birol, G., Undey, C. and Cinar, A. "A modular simulation package for fed-batch fermentation: penicillin production". *Computers and Chemical Engineering,* 26: 1553-1565, 2002.

21. Bread, R.V. "Failure Accommodation in Linear Systems through Self-Reorganization". *Cambridge MA: Department MVT-71-1,* Man Vechnicle Laboratory, 1971.

22. Bro, R. "PAPAFAC: Tutorial and applications". *Chemometrics and Intelligent Laboratory System,* 38:149-171, 1997.

23. Bro, R., Kjeldahl, K., Smilde, A. K. and Kiers, H. A. L. "Cross-validation of component models: A critical look at current methods". *Analytical and bioanalytical chemistry,* 390:1241-1251, 2008.

24. Burt, C. "Alternative methods of factor analysis and their relations to pearson's method of Principle Axes". *British Journal of Psychology, Statistical Section,* 2:98–121, 1949.

25. Camacho, J., Pico,J. and Ferrer, A. "Multi-phase analysis framework for handling batch process data".*Journal of Chemometrics,* 10:632–643, 2008.

26. Camacho, J. and Pico, J. "Multi-phase principal component analysis for batchprocesses modeling". *Chemometrics and Intelligent Laborator Syetems*, 81:127–136, 2006.

27. Chen, G, Cheng, S.W. and Xie, H. “Monitoring process mean and variability with one EWMA chart”. *Journal of Quality Technology*, 33:223–233, 2001.

28. Chen, S., Cowan, C. F. N. and Grant, P. M. “Orthogonal least squares algorithms for radial basis function networks”. *IEEE Transactions on Neural Networks*, 2 (2) :02–309, 1991.

29. Chen, X. and Yan, X. “Using improved self-organizing map for fault diagnosis in chemical industry process”. *Chemical Engineering Research and Design*, 90 (12): 2262– 2277, 2012.

30. Chiu, J.E. and Kuo, T. “Control charts for fraction nonconforming in a bivariate binomial process”. *Journal of Applied Statistics.* 37:717-1728, 2010.

31. Chylla, R. W. and D.R. Haase. “Temperature control of semi-batch polymerization reactors*”. Computers & Chemical Engineering*. 17:257–264, 1993.

32. Crowder, S. Y. “Design of exponentially weighted moving average schemes”. *Journal of Quality Technology*, 21(3), 155-162, 1989.

33. Dayal, B. and MacGregor, J. “Improved PLS algorithms”. *Journal of Chemometrics*, 11(1): 73–85, 1997a.

34. Dayal, B. S. and MacGregor, J. F. “Recursive exponentially weighted PLS and its applications to adaptive control and prediction”. *Journal of Process Control*, 7(3):169–179, 1997b.

35. De Jong, S. “SIMPLS: an alternative approach to partial least squares regression”. *Chemometrics and Intelligent Laboratory Systems*, 18 (3): 251–263, 1993.

36. Doan, X. T., & Srinivasan, R. “Online monitoring of multi-phase batch processes using phase-based multivariate statistical process control”. *Computers and Chemical Engineering*, 32: 230–243.2008.

37. Dodge, H.F. and Romig, H. G., Sampling Inspection Tables, Single and Double Sampling, 2nd ed. John Wiley & Sons, New York, 1959.

38. Dodge, H.F. “Chain sampling inspection plans”. *Industrial Quality Control*, 11(4): 10-13, 1955.

39. Dong, D. and McAvoy, T. “Nonlinear principal component analysis-based on principal curves and neural networks”. *Computers & Chemical Engineering*, 20(1):65-78, 1996.

40. Eastment, H. T. and Krzanowski, W. J. “Cross-validatory choice of the number of components from a Principal Component Analysis”. *Technometrics*, 24: 73-77, 1982.

41. Ewan, W. D. "When and how to use cu-sum charts." *Technometrics*, 5(1): 1-22, 1963.

42. Facco, P., Doplicher, F., Bezzo, F. and Barolo, M. “Moving average PLS soft sensor for online product quality estimation in an industrial batch polymerization process”. *Journal of Process Control*, 19(3):520–529, 2009.

43. Ferrer, A., Aguado, D., Vidal-Puig, S., Prats, J. M. and Zarzo, M. “PLS: a versatile tool for industrial process improvement, and optimizztion”. *Applied Stochastic Models in Business and Industry*, 24: 551-567, 2008.

44. Fletcher, R. and M. J.D. “Powell, a Rapidly Convergent Descent Method for Minimization”. *The Computer Journal*, 6:163–168, 1963/1964.

45. Flores-Cerrillo, J. and MacGregor, J. F. “Control of batch product quality by trajectory manipulation using latent variable models”. *Journal of Process Control*, 14(5), 539-553, 2004.

46. Flores-Cerrillo, J. and MacGregor, J. F. “Multivariate monitoring of batch processes using batch to batch information”. *AIChE Journal*, 50(6), 1219-1228, 2004.

47. Frank, I. “A nonlinear PLS model”. *Chemometrics and Intelligent Laboratory Systems*, 8:109-119, 1990.

48. Frank, P. M. Ding, S.X. and Marcu, T. “Model-based fault diagnosis in technical processes”. *Transactions of the Institute of Measurement and Control*, 22(1): 57–101, 2000.

49. Freyermuth, B. "An approach to model based fault diagnosis of industrial robots." *Robotics and Automation*, 1991. *Proceedings.*, 1991 *IEEE International Conference on. IEEE*, 1991.

50. Gallagher, N. B., Wise, B. M. and Stewart, C. W. “Application of Multi-way principal components analysis to nuclear waste Storage tank monitoring”. *Computers & chemical engineering*, 20:739:744, 1996.

51. Galton, F., Natural Inheritance. London: Macmillan, 1889.

52. Garces, H.and Sbarbaro, D. “Outliers detection in environmental monitoring databases”. *Engineering Applications of Artificial Intelligence*, 24:41–349, 2011.

53. Geladi, P. and Kowaiski, B. R. "Partial Least-Squares regression: A Tutorial". *Analytica Chimica Acta*, 185:1-17, 1986.

54. Gerlach, R. W., Kowalski, B. R. and Wold, H. "Partial least squares path modeling with latent variables". *Analytica Chimica Acta,* 112: 417–421, 1979.

55. Goldberg, J. L., Matrix Theory and Applications. McGraw-Hill, New York, 1991.

56. Golshan, M. and MacGregor, J. F. "Latent variable modeling of batch processes for trajectory tracking control". *International Federation of Automatic Control,* Leuven, Belgium, 2010.

57. Golshan, M., MacGregor, J. F., Bruwer, M. J. and Mhaskar, P. "Latent variable model predictive control (LV-MPC) for trajectory tracking in batch processes". *Journal of Process Control*, 20 (4) : 538–550, 2010.

58. Gower, J. C. "Some distance properties of latent root and vector methods used in multivariate data analysis". *Biometrika*, 53:315–328, 1966.

59. Graichen, K., Hagenmeyer, V. and Zeitz, M. "Feedforward control with online parameter estimation applied to the Chylla–Haase reactor benchmark". *Journal of Process Control*, 16:733–745, 2006.

60. Han, S. W., Tsui, K. L., Ariyajunya, B. and Kim, S. B. "A comparison of CUSUM, EWMA, and temporal scan statistics for detection of increases in Poisson rates". *Quality and Reliability Engineering International*, 26(3): 279:289, 2009.

61. Hawkins, D. M. "Cumulative sum control charting: An underutilized SPC tool". *Quality Engineering,* 5(3):463:477, 1993.

62. Hecht-Nielsen, R. "Theory of the back propagation neural network". *International Joint Conference on Neural Networks*, 593–605, 1989.

63. Heermann, P.D. and Khazenie, N. "Classification of multispectral remote-sensing data using a back-propagation neural network". *Geoscience and Remote Sensing, IEEE Transactions on* 30(1): 81–88, 1992.

64. Helland, I. S. "On the structure of partial least squares". *Communications in Statistics – Simulations,* 17(2):581-607, 1988.

65. Helland, K., Berntsen, H., Borgen, O.and Martens, H. "Recursive algorithm for partial least squares regression". *Chemometrics and Intelligent Laboratory Systems*, 14: 129–137, 1991.

66. Hiden, H., Mckay, B., Willis, M. and Montague, G. "Non-linear partial least squares using genetic programming". *In J. Koza (Ed.), Genetic Programming: Proceedings of the Third Annual Conference*, Morgan Kaufmann, 1998.

67. Holcomb, T. R. and Morari, M. "PLS/Neural networks". *Computers & Chemical Engineering*, 16:393-411, 1992.

68. Hornik K., Stinchcombe, M. and White, H. "Multilayer feedforward neural networks are universal approximates". *Neural Networks*, 2:359-366, 1989.

69. Homik K., Stinchcombe, M. and White, H. "Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks". *Neural Network*, 3:551-560, 1990.

70. Hoskuldsson,A. "PLS regression methods". *Journal of Chemometrics*, 2:211-228, 1988.

71. Hotelling, H. "Analysis of a complex of statistical variables into principal components". *Journal of Educational Psychology*, 24:417-441, 1933.

72. Hotelling, H. Multivariate Quality Control, Techniques of Statistical Analysis. In Eisenhart, Hastay and Wallis (eds). McGraw-Hill, New York, 1947.

73. Huang S.C. and Huang, Y. F. "Bounds on the number of hidden neurons in multilayer perceptrons". *Neural Networks, IEEE Transactions on*, 2(1): 47-55,1991.

74. Isermann, R. and Balle, P. "Trends in the Application of Model Based Fault Detection and Diagnosis of Technical Processes". *Proceedings of the 13th IFAC World congress*, N: 1–12, 1996.

75. Isermann, R. "Fault diagnosis of machines via parameter estimation and knowledge processing". *Automatica*, 29(4):815-835, 1993.

76. Isermann, R. "Supervision, fault-detection and fault-diagnosis methods—an introduction". *Control Engineering Practice*, 5(5): 639 – 652, 1997.

77. Jackson, J. E., A User's Guide to Principal Components. John Wiley & Sons, New York, 1991.

78. Jackson, J. E. "Multivariate quality control".*Communications in Statistics - Theory and Methods*, 14(11): 2657-2688, 1985.

79. Janik, L.J., Forrester, S.T. and Rawson, A. "The prediction of soil chemical and physical properties from mid-infrared spectroscopy and combined partial least-squares regression and neural networks (PLS-NN) analysis". *Chemometrics and Intelligent Laboratory Systems*, 97 (2):179–188, 2009.

80. Jeffers, J. N. R. “Two case studies in the application of principal component analysis”. *Applied Statistics.*, 16: 225-236, 1967.

81. Jiang, W., Tsui, K. L. and Woodall, H. W. “A new SPC monitoring method: The ARMA chart”. *Technometrics*, 42, 399–410, 2000.

82. Jiang, J., Hu, R., Han, Z., Lu, T. and Huang, K. “A super-resolution method for low-quality face image through RBF-PLS Regression and Neighbor Embedding”. *In ICASSP*, 1253--1256, 2012.

83. Johnson, R. A.and Bagshaw, M. “The effect of serial correlation on the performance of CUSUM tests”. *Technometrics*, 16:103–112, 1974.

84. Johnson, R.A. and Wichern, D.W., Applied Multivariate Statistical Analysis. Third Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1992.

85. Johnson, N. L. “A simple theoretical approach to cumulative sum control charts”.*Journal of the American Statistical Association*, 54, 1961.

86. Jolliffe I.T., Principal Component Analysis. Series: Springer Series in Statistics, 2nd ed., Springer, New York, 2002.

87. Jones, L., Failure Detection in Linear Systems. Ph.D. Dissertation, MIT, Cambridge, MA, 1973.

88. Kaspar, M. H., and Ray, W. H. “Partial least squares modelling as successive singular value decompositions”.*Computers and Chemical Engineering*, 17(10):985-989, 1993.

89. Kendall, M., Multivariate analysis. Second edition. Charles Griffin, London, UK, 1980.

90. Kim, S., Alexopoulos, C., Tsui, K. and Wilson, J. R. “A distribution-free tabular CUSUM chart for autocorrelated data”. *IIE Transactions*, 39(3): 317–330, 2007.

91. Korovessi, E. and Linninger, A., Batch Processes. CRC Press, Taylor & Francis Group. Boca Raton, FL, 2006.

92. Kramer, M. A. “Nonlinear principal component analysis using autoassociative neural networks”. *AIChE Journal*, 37(2):233-243, 1991.

93. Krzanowski, W. J. “Selection of variables to preserve multivariate structure, using principal components”. *Applied Statistics*, 36:22–33, 1987.

94. Ku, W., Storer, R.H. and Georgakis, C. “Disturbance detection and isolation by dynamic principal component analysis”. *Chemometrics and Intelligent Laboratory Systems*, 30: 179–196, 1995.

95. Lee, J.M., Yoo, C.K. and Lee, I.B. “Statistical process monitoring with independent component analysis”. *Journal of Process Control*, 14 (5):467–485, 2003a.

96. Lee, J.M., Yoo, C.K. and Lee, I.B. “Statistical process monitoring with multivariate exponentially weighted moving average and independent component analysis”. *Journal of Chemical Engineering of Japan*, 36 (5):563–577, 2003b.

97. Lee, M.W., Hong, S. H., Choi, H., Kim,J. M., Lee, D.S. and Park, J. M. “Real-time remote monitoring of small-scaled biological wastewater treatment plants by a multivariate statistical process control and neural network-based software sensors”. *Process Biochemistry*, 43 (10): 1107–1113, 2008.

98. Lennard, J. A. and Kramer, M. A. “Radial basis function networks for classifying process faults”. *IEEE Transactions Control System Magazine*, 31–38, 1991.

99. Lennox, B., Hiden, H.G., Montague, G.A., Kornfeld, G. and Goulding, P.R. “Application of multivariate statistical process control to batch operations”. *Computers and Chemical Engineering*, 24:291-296, 2000.

100. Lennox, B., Hiden, H.G., Montague, G. A., Kornfeld, G. and Goulding, P. R. “Process monitoring of an industrial fed-batch fermentation”. *Biotechnology and Bioengineering*, 74: 125-135, 2001.

101. Lennox, B., Montague, G., Hiden, H. and Kornfeld, G. “Moving window MSPC and its application to batch processes”. *Proceedings of Computer Applications in Biotechnology (CAB 8)*, Quebec City, Canada, 2001.

102. Leonard J. and Kramer M. A. “Improvement of the backpropagation algorithm for training neural networks”. *Computers & Chemical Engineering*, 14(3):337-341, 1990.

103. Lindgren, F., Geladi, P. and Wold, S. “The kernel algorithm for PLS”. *Journal of Chemometrics*. 7: 45- 59, 1993.

104. Lipp, M. “Comparison of PLS, PCR and MLR for quantitative determination of foreign oils and fats in butter fats of several European countries by their triglyceride composition”. *Zeitschrift für Lebensmittel-Untersuchung und Forschung*, 202:193-198, 1996.

105. Li, S., Li, L, Milliken, R., and Song, K. “Hybridization of partial least squares and neural network models for quantifying lunar surface minerals”. *Icarus.* 221(1): 208–225, 2012.

106. Lorber, A., Wangen, L. E. and Kowalski, B. R. “A Theoretical Foundation for the PLS Algorithm”.*Journal of Chemometrics*, 1:19-31, 1987.

107. Louwerse, D. J. and Smilde, A. K. “Multivariate statistical process control of batch processes based on three-way models”. *Chemical Engineering Science,* 55:1225-1235, 2000.

108. Lowry, C. A., Champ, C. W. and Woodall, W. H. “The performance of control charts for monitoring process variation”. *Communications in Statistics - Simulation*, 24:409-437, 1995.

109. Lowry, C. A. and Montgomery, D. C. “A review of multivariate control charts”.*IIE Transactions*, 27(6):800-810, 1995.

110. Lowry, C. A., Woodall, W. H and Champ, C. W. “Multivariate exponentially weighted moving average control chart”. *Technometrics*, 32:1-12, 1992.

111. Lu, C. W. and Reynolds, JR. M. R. “CUSUM charts for monitoring an autocorrelated process”. *Journal of Quality Technology*, 33:316-334, 2001.

112. Lu, C. W. and Reynolds, JR. M. R. “Control charts for monitoring the mean and variance of autocorrelated processes”. *Journal of Quality Technology* 31:259-274,1999b.

113. Lu, C. W. and Reynolds, M. R. “EWMA control charts for monitoring the mean of autocorrelated processes”. *Journal of Quality Technology*, 31:166–188,1999a.

114. Lucas, J. M. and Saccucci, M.S. “Exponentially weighted moving average control schemes properties and enhancements”. *Technometrics,* 32(1):1-12, 1990.

115. Lu, C.J., Wu, C. M., Keng, C. J. and Chiu, C.C. “Integrated application of SPC/EPC/ICA and neural networks”. *International Journal of Production Research*, 46 (4): 873–893, 2008.

116. MacDonell, W.R. “On criminal anthropometry and the identification of criminals”. *Biometrika*, 1:177–227, 1902.

117. MacGregor, J. F., Marlin, T. E., Kresta, J. V. and Skagerberg, B. “Multivariate statistical methods in process analysis and control”. *Proceeding*

of Fourth International Conference On Chemical Process Control, 70-99, 1991a.

118. MacGregor, J. F.and Cinar, A. "Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods". *Computers and Chemical Engineering*, 47:111–120, 2012.

119. MacGregor, J. F., Jaeckle, C., Kiparissides, C. and Koutoudi, M. "Process monitoring and diagnosis by Multi-Block PLS methods". *Journal of the America Institution of Chemical Engineers*, 40(5): 826-838,1994.

120. MacGregor, J. F. and T. Kourti. "Statistical process control of multivariate processes". *Control Engineering Practice*, 3: 403-414, 1995.

121. Mackiewicz, A. and Ratajczak, W. "*Principal Components Analysis (PCA)".Computers & Geosciences*, 19(3):303-342, 1993.

122. Malthous, E. C., Tamhane, A. C. and Mah, R. S. H. "Nonlinear partial least squares". *Computers & Chemical Engineering*, 21:875-890, 1997.

123. Manne, R. "Analysis of two partial-least-squares algorithms for multivariate calibration". *Chemometrics and Intelligent Laboratory Systems*, 2(1):187-197, 1987.

124. Mardia, K. V., Kent, J. T. and Bibby, J. M., Multivariate Analysis. Academic Press, London, 1989.

125. Martens, H. and Naes, T., Multivariate Calibration. John Wiley and Sons., New York, 1989.

126. Martin, E. B. and Morris, A. J. "An overview of multivariate statistical process control in continuous and batch process performance monitoring". *Transaction Instrument Measurement and Control*, 18:51-60, 1996.

127. Massy, W. F. "Principal components regression in explanatory statistical research". *Journal of the American Statistical Association*, 60**,** 234-246, 1965.

128. Mastrangelo, C.M. and Brown, E.C. "Shift detection properties of moving centerline control chart schemes". *Journal of Quality Control*, 32(1):67–74, 2000.

129. Mastrangelo, C.M. and Montgomery, D.C. "SPC with correlated observations for the chemical and process industries". *Quality and Reliability Engineering International*, 11:79–89, 1995.

130. McAvoy T. J.,Wang, N. S., Naidu,N., Bhat,N. V., Gunter, J. and Simmons, M. "Interpreting biosensor data via back-propagation". *Proceedings IEEE*

International Joint Conference on Neural Networks, I: 227~233, Washington, DC, 1989.

131. Mehra, P. and Wah, B.W. Artificial Neural Networks: Concepts and Theory. IEEE Computer Society Press, Los Alamitos, Calif, 1992.
132. Mevik, B.H. and Wehrens, R. "The PLS package: principal component and partial least squares regression in R. J. Stat". *Journal of Statistical Software* , 18 (2):1–24, 2007.
133. Miller, P., Swanson, R. E. and Heckler, C. F. "Contribution plots: The missing link in multivariate quality control". *Applied mathematics and computer science*, 8: 775-7921998.
134. Montague, G., Morris, A., Wright, A., Aynsley, M. and Ward, A. "Growth monitoring and control through computer-aided on-line mass balancing in fed-batch penicillin fermentation". *Canadian Journal of Chemical Engineering*, 64: 567-580, 1986.
135. Montgomery, D. C., Introduction to Statistical Quality Control. John Wiley,New York,1996.
136. Montgomery, D. C. and Mastrangelo, C. M. "Some statistical process control charts methods for autocorrelated data". *Journal of Quality Technology*. 23: 179-193, 1991.
137. Muñoz, A. and Muruzábal, J. "Self-organizing maps for outlier detection". *Neurocomputing*, 18:33–60, 1998.
138. Muirhead, R.J., Aspects of Multivariate Statistical Theory. Wiley, New York, 1982.
139. Nelson, P.R.C., Taylor, P.A. and MacGregor, J.F. "Missing data methods in PCA and PLS: Score calculations with incomplete observations". *Chemometrics and Intelligent Laboratory Systems*, 35(1): 45-65, 1996.
140. Nomikos, P.and MacGregor, J. F. "Multi-way partial least squares in monitoring of batch processes". *Chemometrics and Intelligent Laboratory Systems*, 30(1): 97-108, 1995.
141. Nomikos, P. and MacGregor, J. F. "Monitoring batch processes using Multi-way principal component analysis". *AIChE*, 40:1361-1375,1994.
142. Nomikos, P. and MacGregor, J. F. "Multivariate SPC charts for monitoring batch processes". *Technometrics*, 37(1):41-59, 1995.

143. Nomikos, P. and MacGregor, J. F. "Multi-way partial least squares in monitoring batch processes". *Chemometrics and Intelligent Laboratory System,* 30:97-108, 1995.

144. Pearson, K. "On Lines and planes of closest fit to systems of points in space". *Philosophical Magazine,* 2 (11): 559–572, 1901.

145. Piovoso, M. J. and Kosanovich K. A., The Use of Multivariate Statistics in Process Control, in Control Handbook. (Ed. Levine, W. C. and Dorf, J. C.) 561-573, CRS Press and IEEE Press, Boca Raton, 1996.

146. Piovoso M. J. and Owens, A. J. "Sensor data analysis using artificial neural networks". *International Conference on Chemistry Process Control,* Texas, 1991.

147. Poggio, T. and Girosi, F. "Networks for approximation and learning". *Proceedings of the IEEE,* 78(10):1481–1497, 1990a.

148. Poggio, T. and Girosi, F. "Regularization algorithms for learning that are equivalent to multilayer networks". *Science,* 247:978–982,1990b.

149. Qin, S.,and McAvoy, T. "Non-linear PLS model using neural networks". *Computers & Chemical Engineering,* 16(4):379-391, 1992.

150. Qin S.J. "A statistical perspective of neural networks for process modelling and control". *In: Proceedings of international symposium on intelligent control,* Chicago, 559–604, 1993.

151. Qin, S. J. "Recursive PLS algorithms for adaptive data modeling". *Computers and Chemical Engineering,* 22(4–5):503–514, 1998.

152. Rao, C.R. "The use and interpretation of principal component analysis in Applied Research". *Sankhyā: The Indian Journal of Statistics, Series A.* 26:329-358, 1964.

153. Ramadan, Z., Hopke, P. K., Johnson, M. J. and Scow, K. M. "Application of PLS and back-propagation neural networks for the estimation of soil properties". *Chemometrics and Intelligent Laboratory Systems,* 75: 23–30, 2005.

154. Roberts, S. W. "Control charts tests based on geometric moving averages". *Technometrics,* 1, 1959.

155. Rosipal, R. "Nonlinear partial least squares: An overview". *Chemoinformatics and Advanced Machine Learning Perspectives: Complex Computational Methods and Collaborative Techniques,* 169-189, 2011.

156. Rumelhart, D. E., Hinton, G. E. and Williams, R. J.,Learning internal representations by error propagation. in Rumelhart, D. and McClelland, J.(eds), Parallel Distributed Processing: Explorations in the Microstructure of Cognition, The MIT Press, Cambridge, MA. , 1986.

157. Saccucci, M.S., Amin, R.W, and Lucas, J.M. "Exponentially weighted moving average control schemes with variable sampling intervals". *Communication in Statistics—Simulation and Computation*, 21:627–657, 1992.

158. Schmid, W. "On the run length of a Shewhart control chart for correlated data". *Statistical Papers, 36*:111–130, 1995.

159. Schmid,W., On EWMA Charts for Time Series. Physica-Verlag HD, 1997.

160. Seber, G. A. F., Multivariate Observations. John Wiley & Sons, Hoboken, New Jersey. 1984.

161. Seber, G. A. F., Linear Regression Analysis. John Wiley & Sons, Hoboken, New Jersey, 2003.

162. Shen, D., Shen, H. and Marron. J. "Consistency of sparse PCA in high dimension, low sample size contexts". *Journal of Multivariate Analysis*, 115: 317-333, 2012.

163. Shewhart, W. A., Economic Control of Quality of Manufactured Product. MacMillan and Co., London, 1931.

164. Stinchcombe, M. and White, H. "Universal approximation using feedforward networks with non-sigmoid hidden layer activation functions". *In Proceedings of the International Joint Conference on Neural Networks*, 1:613-618, San Diego:SOS Printing,1989.

165. Sun, W., Meng, Y., Palazoglu, A., Zhao, J., Zhang, H. and Zhang, J. "A method for multiphase batch process monitoring based on auto phase identification". *Journal of Process Control*, 21: 627–638, 2011.

166. Svozil, D., Kvasnicka, V., and Pospichal, J. "Introduction to multi-layer feedforward neural networks". *Chemometrics and Intelligent Laboratory Systems.* 39(1): 43–62, 1997.

167. Tano, K., Samskog P. O. and Andreasson, B. "Multivariate modelling and on-line data presentation for process monitoring at LKAB". *8th IFAC Symposium on Automation in Mining Mineral and Metal Processing*, Sun City, South Africa, 1995.

168. VanBrackle III, L. N. and Reynolds, JR. M. R. “EWMA and CUSUM ControlCharts in the Presence of Correlation”. *Communications in Statistics – Simulation*, 26:979-1008, 1997.

169. Vasilopoulos, A. V. and Stamboulis, A. P. “Modification of control limits in the presence of correlation”. *Journal of Quality Technology,* 10:20–30, 1978.

170. Venkatasubramanian, V., Rengaswamy, R., Yin, K. and Kavuri, S. N. “A review of process fault detection and diagnosis. Part I: Quantitative model-based methods”. *Computer and Chemical Engineering,* 27(3): 293–311, 2003.

171. Venkatasubramanian, V., Rengaswamy, R. and Kavuri, S. N. “A review of process fault detection and diagnosis. Part II: Qualitative model and search strategies”. *Computer and Chemical Engineering,* 27(3):313–326, 2003.

172. Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N. and Yin, K. “A review of process fault detection and diagnosis. Part III: Process history based methods”. *Computer and Chemical Engineering,* 27(3): 327–346, 2003.

173. Wangen, L. E. and Kowalski, B. R. “A multiblock partial least squares algorithm for investigating complex chemical systems”. *Journal of Chemometrics.*3: 3–20, 1988.

174. Wang, X.; Kruger, U. and Lennox, B. “Recursive partial least squares algorithms for monitoring complex industrial processes”. *Control Engineering Practice.* 11 (6):613, 2003.

175. Wardell, D. G., Moskowitz, H. and Plante, R. D. “Control charts in the presence of data correlation”. *Management Science*, 38:1084-1105,1992.

176. Warne, R. T. “Beyond multiple regression: Using commonality analysis to better understand R2 results”. *Gifted Child Quarterly, 55,* 313-318, 2011.

177. Werbos, P. J., Beyond regression: new tools for prediction and analysis in the behavioral science. Ph.D. Dissertation, Harvard University, Cambridge, *MA*, 1974.

178. Werbos, P.J., The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting. John Wiley & Sons, New York, 1994.

179. Werbos, P.J. “Generalization of back propagation with application to a recurrent gas market model”. *Neural Networks,* 1:339–356 , 1988.

180. Westerhuis, J. A., Kourti, T. and MacGregor, J. F. "Compare alternative approaches in batch process data analysis". *Journal of Chemometrics*, 13:397-413, 1999.

181. Wetherill, G. B. and Brown, D. W., Statistical Process Control - Theory and Practice. Chapman and Hall, London, 1991.

182. Williams, R.J. and Zipser, D. "A learning algorithm for continually running fully recurrent neural networks". *Neural Computation*, 1(2):270–280, 1989.

183. Willsky A.S. "A survey of design methods for failure detection in dynamic systems", *Automatica*, 12: 601 – 611, 1976.

184. Wilson, D., Irwin, G. and Lightbody, G. "Nonlinear PLS modelling using radial basis functions". *American Control Conference, 1997. Proceedings of the 1997*, 5:3275-3276, 1997.

185. Wise, B., Gallagher, N., Bro, R. and Shaver, J. "PLS Toolbox 3.0". *Manson, WA: Eigenvector Research Inc*.171, 2003.

186. Wise, B. and Ricker, N. "Recent Advances in Multivariate Statistical Process Control: Improving Robustness and Sensitivity". *IFAC Symposium on Advanced Control of Chemical Processed*, 125-130, Toulouse, France, 1991.

187. Wise, B. M. and Gallagher, N. B. "The process chemometrics approach to process monitoring and fault detection". *Journal of Process Control*, 6(6): 329-348, 1996.

188. Wise, B. M., Veltkamp, D. J., Ricker, N. L., Kowalski, B. R., Barnes, S. M. and Arakali, V. "Application of multivariate statistical process control to the west valley slurry-red ceramic melter process". *Waste Management Proc*, Tuscon, 1991a.

189. Wold, H., Nonlinear Estimation by Iterative Least Squares Procedures: in David, F. N. (eds). John Wiley& Sons, New York, 1966.

190. Wold, H., Soft Modelling by Latent Variables : The Nonlinear Iterative Partial Least Squares Approach: in Gani, J. (eds) ,Perspectives in Probability and Statistics, Papers in Honour of M.S. Bartlett, Academic Press, London,1975.

191. Wold, H. "Soft Modelling, the basic design and some extensions". *System under Indirect Observation*, 2: 1–54, North-Holland, Amsterdam, 1982.

192. Wold, H., Ruhe, A., Wold, H. and Dumn, W. "The collinearity problem in linear regression The Partial least squares (PLS) approach to generalized

inverses". *SIAM Journal on Scientific and Statistical Computing*, 5: 735-743, 1984.

193. Wold, S. "Cross-Validatory estimation of number of components in factor and principal components models". *Technometrics*. 20: 397-405, 1978.

194. Wold, S. "Principal component analysis". *Chemometrics and Intelligent Laboratory Systems*, 2:37-52, 1987.

195. Wold, S. "Nonlinear partial least squares modelling. II. Spline inner relation". *Chemometrics and Intelligent Laboratory Systems*, 14(1):71-84, 1992.

196. Wold, S., Geladi, P., Esbensen, K. and Ohman, J. "Multi-Way principal components and PLS analysis". *Journal of Chemometrics*, 1(4): 1-56, 1987.

197. Wold, S., Hellberg, S., Lundstedt, T., Sjostrom, M. and Wold, H. "PLS modeling with latent variables in two or more dimensions". *Version 2.1. Presented at Frankfurt PLS Meeting*, Frankfurt, Germany, 1987.

198. Wold, S., Kettaneh, N., Friden, H. and Holmberg, A. "Modelling and diagnositics of batch processes and analogous kinetic experiments". *Chemometrics and Intelligent Laboratory*, 44:331-340, 1998.

199. Wold, S., Kettaneh-Wold, N. and Skagerberg, B. "Nonlinear PLS modelling." *Chemometrics and Intelligent Laboratory Systems*. 7: 53-65, 1989.

200. Wold, S., Kettaneh-Wold, N. and Tjessem, K. "Hierarchical multiblock PLS and PC models, for easier model interpretation, and as an alternative to variable selection". *Journal of Chemometrics*, 10:463–482, 1996.

201. Woodall, W. H., and Benjamin M. A. "The statistical design of CUSUM charts." *Quality Engineering*, 5(4):559-570, 1993.

202. Wu, L. and Lennox, B. "Batch monitoring and control: A comparison of two approaches". *International Control Conference*, Glasgow, United Kingdom, 2006.

203. Yabuki, Y. and MacGregor, J. F. "Product quality control in semibatch reactors using midcourse correction policies". *Industrial & Engineering Chemistry Research*, 36(4):1268-1275, 1997.

204. Yan, L. and Lennox, B. (2013) "The application of nonlinear partial least square to batch processes", *10th IFAC International Symposium on Dynamics and Control of Process Systems*, Mumbai, India, 2013.

205. Yao, Y. and Gao, F. "A survey on multistage/multiphase statistical modelling methods for batch processes". *Annual Reviews in Control*, 33:172-183, 2009.

206. Yao, Y. and Gao, F. “Phase and transition based batch process modeling and online monitoring”. *Journal of Process Control,* 19:816–826, 2009.

207. Yin, L., Chen, X., Sun, Y., Worm, T. and Reale, M. “A high resolution 3D dynamic facial expression database”. *Automatic Face & Gesture Recognition,* 2008. *FG'08. 8th IEEE International Conference On. IEEE,* 2008.

208. Yourstone, S. A. and Montgomery, D. C. “Detection of process upsets sample autocorrelation control chart and group autocorrelation control chart applications”. *Quality and Reliability Engineering International,* 7:133–140, 1991.

209. Yucai, Z., Multivariable System Identification for Process Control. Elsevier Science Ltd. Kidlington, UK, 2001.

210. Zhang, G. and Chang, S. “Multivariate EWMA control charts using individual observations for process mean and variance monitoring and diagnosis”. *International Journal of Production Research,* 46(24):6855-6881, 2008.

211. Zhang, H. and Lennox, B. “Integrated condition monitoring and control of fed-batch fermentation processes”. *Journal of Process Control,* 14: 41-50, 2003.

212. Zhang, N. F. “Statistical control chart for stationary process data”. *Technometrics,* 40: 24–38, 1998.

213. Zhang, Y. “Enhanced statistical analysis of nonlinear process using KPCA, KICA and SVM”. *Chemical Engineering Science,* 64:801–811, 2009.

214. Zhang, Y. X. “Artificial neural networks based on principal component analysis input selection for clinical pattern recognition analysis”. *Talanta,* 73(1):68-75, 2007.

215. Zhao, C. Y., Zhang, H. X., Zhang, X. Y., Liu, M. C., Hu, Z. D.and Fan, B. T. “Application of support vector machine (SVM) for prediction toxic activity of different data sets”. *Toxicology,* 217(2):105-119, 2006.

216. Zhao, W., Chen, D. and Hu, S. “Detection of outlier and a robust BP algorithm against outlier”. *Computers & Chemical Engineering,* 28(8): 1403–1408, 2004.