This work is specially dedicated to

my mother,

Anjale Sinnasamy Thanimalai;

&

my father,

Jewaratnam Narayanasamy

&

my siblings

Jai Sri Shahila Jewaratnam

Jaiswar Jewaratnam

Jaiganth Jewaratnam

my pillars of strength.

Batch-to-batch Iterative Learning Control of a Fed-Batch Fermentation Process

By

Jegalakshimi Jewaratnam

A Thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

School of Chemical Engineering and Advanced Materials
Newcastle University
UK

April, 2013

ABSTRACT

Recently, iterative learning control (ILC) has been used in the run-to-run control of batch processes to directly update the control trajectory. The basic idea of ILC is to update the control trajectory for a new batch run using the information from previous batch runs so that the output trajectory converges asymptotically to the desired reference trajectory. The control policy updating is calculated using linearised models around the nominal reference process input and output trajectories. The linearised models are typically identified using multiple linear regression (MLR), partial least squares (PLS) regression, or principal component regression (PCR). ILC has been shown to be a promising method to address model-plant mismatches and unknown disturbances. This work presents several improvements of batch to batch ILC strategy with applications to a simulated fed-batch fermentation process. In order to enhance the reliability of ILC, model prediction confidence is incorporated in the ILC optimization objective function. As a result of the incorporation, wide model prediction confidence bounds are penalized in order to avoid unreliable control policy updating. This method has been proven to be very effective for selected model prediction confidence bounds penalty factors. In the attempt to further improve the performance of ILC, averaged reference trajectories and sliding window techniques were introduced. To reduce the influence of measurement noise, control policy is updated on the average input and output trajectories of the past a few batches instead of just the immediate previous batch. The linearised models are re-identified using a sliding window of past batches in that the earliest batch is removed with the newest batch added to the model identification data set. The effects of various parameters were investigated for MLR, PCR and PLS method. The technique significantly improves the control performance. In model based ILC the weighting matrices, Q and R, in the objective function have a significant impact on the control performance. Therefore, in the quest to exploit the potential of objective function, adaptive weighting parameters were attempted to study the performance of batch to batch ILC with updated models. Significant improvements in the stability of the performance for all the three methods were noticed. All the three techniques suggested have established improvements either in stability, reliability and/or convergence speed. To further investigate the versatility of ILC, the above mentioned techniques were combined and the results are discussed in this thesis.

ACKNOWLEDGEMENTS

I would like to thank the following:

My supervisors, Dr. Jie Zhang, Prof. Julian Morris and Prof. Mohd. Azlan Hussain for the advice and guidance throughout the preparation of this thesis. My special thanks to Dr Jie Zhang who has been guiding me since day one and having weekly meetings to guide me throughout the PhD study. He has been very understanding of my circumstances and patiently motivated me to complete my work.

My sponsors, Ministry of Higher Education Malaysia (MOHE) and University Malaya (UM) for providing me sufficient financial support.

My CPACT colleagues, Dr.Shallon, Dr.Jeong Jin, Dr.Yusri, Dr.Gang, Carlo and Dr.Cheng for constantly guiding me throughout my work. They have been a joy to work with. Special thanks to Shallon for helping me on the matlab codings in my first year.

My family, Mr. Jewaratnam, Mrs. Anjale, Jai Sri Shahila, Jaiswar and Dr.Jaiganth, for providing moral support throughout the course of my studies. Special thanks to my mother, Mrs Anjale Jewaratnam, for being my strength.

My friends back in Malaysia, Komathi D., Dr.Vasanthi S. and Parthiban R. for their constant support. My friends and housemates in Newcastle for being my moral support while making my stay in Newcastle meaningful. Special thanks to Shengnan Yu, Aishath, Misgun, Khar Mun Looi, Low Hock Soon, Dr.Cumanan K. and Arun George K.

The admin and IT support staffs of CEAM who has been very helpful and efficient in providing support as and when it was needed. Special thanks to Justine, Vince and Daniel.

Finally, to all the research and postgraduate students, past and present, of the School of Chemical Engineering and Advanced Materials, for making my period spent here an enjoyable one.

Jegalakshimi Jewaratnam , 12-12-12.

LIST OF FIGURES

LIST OF TABLES

LIST OF NOMENCLATURE

C_e	ethanol concentration (gL^{-1})
Ce_o	initial ethanol concentration
C_o	oxygen concentration (gL^{-1})
C_o^*	interface concentration of oxygen (gL^{-1})
Co_o	initial oxygen concentration
C_s	substrate (glucose) concentration (gL^{-1})
Cs_o	initial substrate concentration
C_x	biomass concentration (gL^{-1})
Cx_o	initial biomass concentration
C_2H_6O	ethanol
$C_6H_{12}O_6$	sucrose
$CH_{HX}O_{OX}N_{NX}$	molecular formula of yeast biomass
CO_2	carbon dioxide
F	feed flow rate (Lh^{-1})
F_o	initial feed
Gs	linearised model
h	data record interval
H_2O	water
H*X*, O*X*, N*X*	elemental analyses of biomass
K_e	saturation constant for ethanol (gL^{-1})
K_i	inhibition constant (gL^{-1})
k_La_o	total volumetric oxygen transfer coefficient (h^{-1})
K_o	saturation constant for oxygen (gL^{-1})
K_s	saturation constant for substrate (glucose) (gL^{-1})

NH_3	ammonia
O_2	oxygen
$Q_{e,max}$	maximum specific ethanol consumption rate ($gg^{-1}h^{-1}$)
$Q_{e,ox}$	ethanol oxidation rate
$Q_{e,\,pr}$	specific ethanol production rate ($gg^{-1}h^{-1}$)
Q_m	glucose consumption rate for maintenance energy ($gg^{-1}h^{-1}$)
$Q_{o,max}$	maximum specific oxygen consumption rate ($gg^{-1}h^{-1}$)
$Q_{s,max}$	maximum specific substrate consumption rate ($gg^{-1}h^{-1}$)
Q and R	weighting matrices for objective function
RQ	respiratory quotient
S_o	substrate concentration of feed (gL^{-1})
t	time (h)
t_d	time delay (h)
t_f	total fermentation time
Us	nominal substrate feed profile
V	volume (L)
V_{fer}	fermentor volume
V_o	initial reactor volume
$Y_{c/e}$	yield of carbon dioxide on ethanol (gg^{-1})
$Y_{c/s}^{ox}$	yield of carbon dioxide on substrate oxidative metabolism (gg^{-1})
$Y_{c/s}^{red}$	yield of carbon dioxide on substrate reductive metabolism (gg^{-1})
$Y_{e/o}$	yield of ethanol on oxygen (gg^{-1})
$Y_{e/s}$	yield of ethanol on substrate (gg^{-1})
$Y_{o/s}$	yield of oxygen on substrate (gg^{-1})
Ys	nominal biomass output

$Y_{x/e}$	yield of biomass on ethanol (gg^{-1})
$Y_{x/s}^{ox}$	biomass yield on substrate in oxidative metabolism (gg^{-1})
$Y_{x/s}^{red}$	biomass yield on substrate in reductive metabolism (gg^{-1})

Italic fonts

e_k	error from previous run
k	batch numbers
L	ILC learning gain
t	time for kth batch run
u_k	control input of the previous batch
u_{k+1}	control input for the current batch
X	biomass
y_d	desired output trajectory
y_k	previous batch output

Greek Fonts

α	time constant
λ	weighting parameter for model prediction confidence bounds
μ	specific growth rate (h^{-1})
μ_{cr}	critical specific growth rate (h^{-1})
ρ	weighted parameter

LIST OF ACRONYMS

CCL	current cycle learning
ILC	iterative learning control
MLR	multiple linear regression
MPC	model predictive control
MPCB	model prediction confidence bounds
ODE	ordinary differential equation
PCCL	previous and current cycle learning
PCL	previous cycle learning
PCR	principal component regression
PID	proportional–integral–derivative
PLC	programmable logic controllers
PLS	partial least square

CHAPTER 1: INTRODUCTION

1.1 Introduction

Fermentation applications stretch to wide industrial sectors. Beginning with mere domestic household practice, mainly for producing bread and wine in much earlier days, today fermentation have gained wide acceptance in industries such as pharmaceutical, food, household products, textiles, agriculture, chemicals, environment and medicine. Fermentation aim to operate by manipulating and controlling cell lines to produce maximum product yield with best possible quality, at lowest variability, reasonable productivity cost and in the most simple and efficient way. These processes are carried out in either batch, fed-batch or continuous reactors depending on the demand and production cost of the product. Typically batch or fed batch operations are used as a scale up process from the successful laboratory experiment procedures. A classic feature of batch processes such as fermentation is that the same process unit is used in the manufacturing of different products and each product is of relatively small amount and high value added.

The optimal operation of batch processes is very important in the face of increasing competition and stringent regulations on product quality and waste disposal. Optimal control can be used to improve the profit of batch process manufacturing (Bonvin and Srinivasan, 2003). In classic batch process operation, a fixed recipe is applied to the batch process with a PID controller to control reactor temperature (Lee et al., 1999). Based on the feedback and visual inspection on the reactor content, the recipe is adjusted manually. A knowledgeable operator is needed to adjust the control parameters based on intuition and heuristics. At this point, negligence of the operator is a feared risk. Despite that, after long period of continuous iterative amendments, an improved but not optimal product quality is obtained. It is notable that such improvement is possible because the batch operation is repetitive in nature (Lee and Lee, 2003). In this practice, repeatedly occurring measured error from batch to batch can be addressed. However, many other unknown parametric and process disturbances that occur cannot be apprehended via this method. There are sophisticated optimal control algorithms that can be used but are not favoured in the industry due to setting up cost and complexity. An effective yet simple automated process control and optimization algorithm is necessary (Alford, 2006).

Based on the current industrial practice, the iterative improvement has been identified to have similarity with the iterative learning control (ILC) which has been successfully used in the robotic industry for over 30 years (Bristow et al., 2006). In recent years, ILC have been applied in the run-to-run control of batch processes to directly update input trajectory (Zhang et al., 2008). The basic idea of ILC is to update the control trajectory for a new batch run using the information from previous batch runs so that the output trajectory converges asymptotically to the desired reference trajectory. Refinement of control signals based on ILC can significantly enhance the performance of tracking control systems (Xiong and Zhang, 2003). ILC does not involve very complex process modelling and is rather a simple control system.

1.2 Motivation

Although ILC has been actively studied for batch process control, there are several practical issues that have not been solved. Firstly, ILC is derived based on linear models, but most batch processes are typically highly nonlinear. Efficient representation of a highly nonlinear batch process by a linearised model is a challenging problem. Secondly, the practical difficulties faced by the industrial implementation of optimal control strategy include the unavoidable model plant mismatches and the presence of unknown disturbances (Xiong and Zhang, 2005a). Linearising around the nominal control trajectory is the common approach used in batch process ILC. This is acceptable when batch to batch variations are not significant. When significant difference in batch initial condition exist, then the batch to batch variations can be very large and linearising around a fixed nominal control trajectory can introduce significant errors. In such conditions, it would be desirable to incrementally update the linearised model as the batch to batch operation progresses. In a recent development, batch to batch ILC based on linearized perturbation model identified using multiple linear regressions (MLR) is reported by Xiong and Zhang (2003). In that work, the perturbation model is obtained using deviations of process input and output from their nominal trajectories and is updated after every batch by using the immediate previous batch as the nominal batch. This way, the unexpected process and parametric disturbances is expected to be captured and removed to render a more precise model prediction.

1.3 Aims and Objectives

The aims of the study are

- to investigate an ILC strategy for a fed-batch fermentation process to address model-plant mismatches
- to investigate the performance of different linearised models used in the batch to batch iterative learning control strategy
- to improve the performance of simple iterative learning control method

In order to cope with model plant mismatches, the batch wise linearised model will be re-identified after each batch run. Multiple linear regressions (MLR), partial least squares (PLS) regression and principal component regression (PCR) is used in identifying the model parameters. A reliable ILC techniques incorporating model prediction confidence is developed. Model prediction confidence bounds are incorporated into the optimization objective function and a reliable ILC law is developed and analyzed. Batch to batch control techniques capable of coping with significant batch to batch initial condition variations is developed. The developed techniques are tested on a simulated fed-batch fermentation process.

The objectives of the study, which have been discussed above, are listed below:

1. To identify batch wise linearised models for highly non-linear fed-batch fermentation processes from process operational data and develop batch to batch iterative learning control based on the identified models. The model parameters will be generated using linear regressions, which includes multiple linear regression (MLR), partial least square (PLS) and principal component regression (PCR).
2. To develop reliable batch to batch iterative learning control algorithm incorporating model predictive confidence that addresses significant parametric and process uncertainty.
3. To develop a simplified and efficient batch to batch ILC by using sliding window of process data in identifying the process model. The process model developed from smaller amount of most recent process data should be able to capture the relevant dynamics of the original systems at a particular process environment to improve and sustain optimal productivity.

4. To investigate, compare and further improve the performance of MLR, PCR and PLS model based ILC algorithms.
5. To exploit the ILC algorithm to improve batch to batch updated linearised model based ILC method performance. The suggested techniques should improve the convergence rate and stability of the process system.

1.4 Contributions

This thesis investigates an ILC strategy for a fed-batch fermentation process using linearised models identified from process operational data. The control policy updating is calculated using a model linearised around a reference batch. In order to cope with process variations and disturbances, the reference batch can be taken as the immediate previous batch. In such a way, the model is a batch wise linearised model and is updated after each batch. The newly obtained process operation data after each batch is added to the historical data base and an updated linearised model is re-identified. In order to overcome the colinearity among the predictor variables, this study proposes that the linearised model can be identified using principal component regression (PCR) and partial least squares (PLS). The linearised model using multiple linear regressions (MLR) is also studied and the performances for all the regression models were evaluated.

As many batch processes are highly nonlinear, the batch-wise linearised model may only be valid over a small operating range. Thus control actions from the ILC strategy may not be reliable if the mismatch between the linearised model and the plant becomes large. In order to overcome this problem, a reliable model based ILC strategy is proposed in this work. Model prediction confidence bound for future predictions can be obtained from historical process operation data used for model identification. The model prediction confidence bound is incorporated into the model based ILC optimisation objective function and wide model prediction confidence bounds are penalised.

In order for the updated model to capture the process behaviour in the face of process variations, a new technique using a moving window of the historical batches to update batch-wise linearised models is developed in this work. The historical batches were updated after every batch run but using only the M recent number of batches. In other words, after every run the “oldest” batch is forgotten and the new batch is included into the sliding “window” of historical batches.

In some cases the stability of the control system is affected in the presence of disturbance. In order to have more control on the future control input, slight adjustment is made to the learning rate of the system. Instead of using a constant weighting ratio, an adaptive weightings ratio method is introduced to the optimization objective function. This method is meant to adjust the control input of future batch in accordance to the error magnitude.

In accordance to the repetitive nature of fed-batch fermentation processes, ILC applied to these processes is meant to improve the control action progressively and iteratively. The existing feed-back controllers only account for within batch input actions based on batch run time dimension. In general, ILC works on 2 dimensions, which is both intra and inter-batch. Since there are a lot of controllers already in existence and used in manufacturing processes, it not may not be necessary to make a total change on the current control system. A controller that adds to the performance of the existing system will be more desirable because it is economical and easier to be implemented. The basic objective of this work is to improve on the existing control systems. The feed-forward action of this proposed ILC will give the input for batch to batch control actions to improve and sustain product yield, which is in addition to the existing time-dimensional control system.

1.5 Publications

1. Jegalakshimi Jewaratnam, Jie Zhang, Mohd Azlan Hussain, Julian Morris, "Batch-To-Batch Iterative Learning Control of a Fed-Batch Fermentation Process Using Incrementally Updated Models", 11th IFAC Symposium on Computer Applications in Biotechnology, CAB2010, 5^{th}-9^{th} July, 2010, Leuven, Belgium.

   Publication: Ifac-papersonline, Elsevier. Computer Applications in Biotechnology,2010, Volume 11,pp.78-83.

2. Jegalakshimi Jewaratnam, Jie Zhang, Mohd Azlan Hussain, Julian Morris, “Reliable Batch-to-Batch Iterative Learning Control of a Fed-batch Fermentation Process”, 22nd European Symposium on Computer Aided Process Engineering – ESCAPE22 ,17-20 June 2012, London, UK

   Publication: Elsevier BV: Computer-Aided Chemical Engineering, Volume 30,2012, pp. 802-806.

3. Jewaratnam J., Zhang J., Morris J. and Hussain A., "Batch-to-batch iterative learning control using linearised models with adaptive model updating," *Control 2012 UKACC International Conference,* 3-5 Sept. 2012, Cardiff, UK

   Publication: IEEExplore; CONTROL 2012, pp.271-276.

4. J. Jewaratnam, J. Zhang, A. Hussain and J. Morris, “Batch-to-batch iterative learning control using updated models based on a moving window of historical data.” CHISA, 25-29th Aug. 2012, Prague.

   Publication: Elsevier Ltd: Procedia Engineering, 2012, Volume 42, pp.206-213.

5. Jegalakshimi Jewaratnam , Jie Zhang, Mohd. Azlan Hussain and Julian Morris, “Run-To-Run Iterative Learning Control With Updated Models Applied In Baker’s Yeast Fed-Batch Fermentation Process,” 4th AUN/SEED-Net Regional conference on Chemical Engineering, 9-10th Feb 2012, Kuala Lumpur, Malaysia. University of Malaya Chemical Engineering Department

6. J. Jewaratnam, J. Zhang, A. Hussain and J. Morris, “Batch-to-batch Iterative Learning Control of Fed-Batch Fermentation Process,” Computer Aided Process Engineering Subject Group (CAPE) Ph.D. Poster Day 12th May 2010 School of Process, Environmental and Materials Engineering, University of Leeds, UK

1.6 Thesis Layout

The thesis is structured as follows. Chapter 2 provides a literature review of batch to batch iterative learning control of a fed batch fermentation process. An introduction to fed batch fermentation and iterative learning control concept are presented. Then, a review on the past works on batch to batch iterative learning control method is presented. Finally, the mathematical representation of the control method and regression models used in this work is laid out. Chapter 3 presents the dynamic and kinetic models of Baker’s yeast fed-batch fermentation process. The models were coded in MATLAB and the simulation results were verified with reference paper. The simulation assumptions were also outlined. The simulation of this process is used as a case study in the research. In Chapter 4, the parameter selection outcome for the simulation work is presented. The outcome of batch to batch ILC control with updated linearised models applied in Baker's yeast fed-batch fermentation simulations is presented and discussed. Following that, in Chapter 5, the development of reliable batch to batch ILC method is illustrated. The improvement by incorporating model prediction confidence bounds into MLR, PLS and PCR models were recorded and analysed. Chapter 6 points out the possibility of using reduced number of historical batches to develop the linearised process model for batch to batch ILC. The predefined number of batches is updated after every batch run to represent the current system condition. This technique proves to further improve the stability and convergence of batch to batch updated model ILC performance. Chapter 7 is on a study about feasible manipulation of the weighting parameters ratio in the objective function to improve the control performance in the presence of error. The adaptive weighting ratio method gave more control to the system and enhances system stability. Finally, Chapter 8 concludes the whole report and provides some suggestions for future works.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

Bioprocess is used in wide spectrum of production line these days such as to produce biofuels, renewable chemicals, biotechnology, aquaculture, poultry production, pharmaceuticals, enzymes and food processing. Bioprocesses literally use living cells to produce desired products. There are different dynamics involved in the production systems. The types of bioprocess dynamics include fermentations, anaerobic digestions, biomass cultivations, animal and fish production systems, and targeted substance extractions from biological materials. One of the popular bioprocess mechanisms is fermentation. Depending on the characteristics of the fermentation and its goals, fermentations are carried out in batch, fed-batch or continuous process. Fed-batch reactor is a more attractive choice in fermentation to produce high cell density than batch fermenter (Lee et al., 1999; Ashoori et al., 2009). About 50% of the fermentation processes are carried out in fed-batch mode (Givens, 2009) and therefore, this work is focused on fed-batch fermentation.

Fed-batch reactors can be easily modified for changing product specifications where one vessel can be used for several products and can be used to produce in small volumes (Morris and Zhang, 2009). Therefore, fed-batch bioreactors are usually used in bioprocesses involving agile manufacturing of high value added products such as specialty polymers (Zhang, 2005), pharmaceuticals (Hong et al., 2011), fine chemicals, and bio-products (Bonvin 1998; Zhang, 2005). Agile manufacturing in this context is referred to the biochemical processes developed to respond quickly to evolving customer needs and market changes while still controlling costs and rendering quality. Product quality is an important specification for fermentation production line. To date, there are no sensors to directly measure product quality for fermentation yields (Muske et al., 2004).

In a complete fed-batch bioreactor setup in industry today, there are a few sensors used to control the fermentation environment that gives indirect measurements of the product quality. In the current industrial practice, samples are withdrawn at predetermined time intervals and at the end of a batch. The samples are tested in a lab and the performance of the reactor at the sampling time is evaluated. This creates a lag time to determine the process performance. Based on the lab results, the feed rate for the new batch run is calculated offline using a mathematical model that is deemed best representing a particular fermentation process. The offline calculated control policy may not be optimal when used on the real process mainly because the mathematical models used in the calculation may not wholly represent the real plant. Therefore, there exist model-plant mismatches. In addition to that, presence of unknown disturbances is literally unavoidable and unpredictable for fed-batch fermentation process. These uncertainties cannot be anticipated and therefore, cannot be addressed in the offline calculations. Furthermore, there is high possibility that variation in the initialization parameter such as due to reactor fouling or raw material specification variation could occur in a batch run. The initialization parameter uncertainties may vary from one batch to another batch which makes offline calculation not fit to account for these unexpected disturbances (Xiong and Zhang, 2003; Lei et al., 2001). Inefficiency of offline calculated control policy is an issue in current fed-batch fermentation control and optimization.

The other issue in current fed-batch fermentation control and optimization in the industry is dependency on human operators to monitor and optimize on-line batch performance manually. The offline calculated control policy is introduced to the next batch run. The new control trajectory will be periodically monitored by plant personnel based on the available records of sensor readings. Experienced personnel are needed to make the necessary amendments in heuristic manner in the event there is a deviation from the control policy trajectory (Yuzgec et al., 2009). No mechanisms are 100% efficient and so are human. There could be human error or carelessness where important changes could be missed and cause wastage due to faulty production. Besides that, the experienced personnel may leave the company for a better job. Then, the company has to spend some time and money to train new personnel and risk losing them too.

Both these issues call for the need of an advanced control system which will either replace the present control system or act as a supervisory control system for the existing control systems. The advanced control system should be able to automatically calculate the control policy for the next batch based on the immediate previous batch performance and self-learn the system to improve product quality performance from one batch to another. There are two ways this can be done (Camacho et al., 2007). One way is to combine the fundamental process knowledge with data-based models to complement the inefficiency of the mechanistic model (Lee et al., 1999; Ng and Hussain, 2004; Rotem et al., 2000). The other method is to completely depend on the input and output data and developing a process model by exploiting the available data (Camacho et al., 2007; Alford, 2006; Cueli and Bordons, 2008). The data-based model is becoming more and more favourable these days due to the transformation of the bioprocess industry into agile manufacturing environment. Furthermore, the repetitive nature of batch process serves as an advantage for application of data-based model (Vlassides et al., 2001). A collection of input and output data from repeated batch trials is expected to have some sort of correlation. This correlation can be exploited to to develop an intelligent and efficient control method for fed-batch fermentation systems. One of the popular data-based control methods is iterative learning control (ILC) (Lee and Lee, 2003; Gao et al., 2001; Mezghani et al., 2001; Fu and Barford, 1992). ILC is literally the best choice of control method when it comes to control of repetitive system. Therefore, ILC is the most appropriate choice of control technique for fed-batch fermentation system due to its repetitive nature.

The chapter is organised as follows. Section 2.2 gives a brief history of control and optimization for fermentation process. Control methods used in the past and present is briefly discussed. Section 2.3 presents an overview of the fed batch fermentation control and optimization. The fed- batch fermentation system operation is explained. Then, current issues with batch to batch control and optimization are discussed. Section 2.4 presents a complete introduction to iterative learning control. Section 2.5 presents the issues pertaining batch to batch model predictive iterative learning control. Section 2.6 presents issues on fed batch fermentation product quality control and optimization. Section 2.7 present the mathematical representation of the batch to batch iterative learning control. Finally, Section 2.8 concludes the chapter.

2.2 Fermentation control

The history of fermentation control and optimization started way back in 1940's during the first industrial production of antibiotics (Beyeler et al., 2000). Beyeler et al. (2000) has given an interesting chronological account on the development of automated control system from 1940's to the 21st century. It started off with proportional–integral–derivative (PID) controllers and then programmable logic controllers (PLC) and then picked up swiftly with the advancement of computers in the 1970's. The current interest in advance control has been there for the last 25 years. In theory, automation is only possible if the process behaviour is known and predictable at any time.

In the past till the early 1980's, researchers were focused in developing comprehensive mathematical models to understand, control and optimize a bioprocess. The accuracy of the models is determined by the model parameters. Model parameters are determined through considerable amount of experiments. The more complex a system is, more parameters are required and therefore more experiments will have to be conducted. Ideally, with a comprehensive and accurate mathematical model of a particular bioprocess, the operation parameters can be varied to obtain optimal biomass growth and/or product formation (Muske et al., 2004). It is as easy as that only and only if such mathematical models are present or easily attainable. Unfortunately, in reality obtaining an accurate and comprehensive mechanistic model requires extensive amount of time, effort and resources. In the ever changing market environment, the product demands and recipes change quite drastically. The time to market literally decides the competency and sustainability of the industrial players. In addition to that, bio-products are usually high value added and only produced in small amounts. Some products are produced only when there is a current need, in other words custom made orders. Therefore, developing a mechanistic model that covers all the possible processing condition is very laborious and less attractive because it is resource consuming (Vlassides et al., 2001; Xiong and Zhang, 2005b).

In the urge to better understand the cell mechanisms, various studies have been conducted to develop on-line sensors and chemical analysers. These are used to estimate different phases and aspects of the bioprocess to supply data to better understand the insights of the processes which are then expected to aid in the bioprocess control and optimization. It is understood that process control and optimization is specific for a particular process and vessel. It is therefore important to identify process-specific cell stress factors, to understand the physiological responses to the vessel specific physical conditions. The mutual influences and interactions of the various physical and physiological parameters have to be analysed in detail. Thanks to the advanced development of genome sequencing, now it is possible to sequence almost any organisms of industrial interest rapidly (Wang et al., 2009). Wang et al. (2009) have given an account of recent development in the industrial bioprocess control and optimization in the context of system biotechnology emphasizing on strain developments to understand the microbial cell technology. Although knowledge about biological reactions has increased immensely during the last decades, whether the advanced technologies will allow accurate prediction or interpretation of the complex behaviour of biological systems completely is still uncertain. With its huge variability, a biological process is not completely predictable. The on-line measurements do not contribute much to overcome this lack of knowledge as there are still only a few exceptions known where biological quantities such as biomass, products, intermediates or substrate can be measured on-line in an industrial environment (Beyeler et al., 2000). Till today, reliable and economical on-line measure of key parameters such as product concentration, quality and feed surplus in the reactor is almost nil (Muske et al., 2004; Karakuzu et al., 2006; Renard and Wouver, 2008).

A large amount of process data are collected in bioprocess operations with the aid of the existing sensors and analysers and stored for almost no apparent use of it. These data remain underutilised in most of the bioprocess industries due to inability to capture and apply the knowledge represented by the data (Vlassides et al., 2001; Schugerl, 2001; Karakuzu et. al., 2006). The problem is not actually collecting more data but to effectively extract the knowledge in the already present database (Alford, 2006). Since the reactions at each phase in the process are interconnected, the collected data on the various parameters will exhibit exploitable correlation (Cueli and Bordons, 2008). Due to the repetitive nature of the biological process, the variation in the good and bad production condition of the past runs can be analysed and used to optimize future production (Vlassides et al., 2001).

In the recent advanced control method development, intelligent controllers can be used to do pattern recognition based on the collection of data from past batch runs without any information on the real process system. Empirical models can be identified from the correlated data and used in the development of intelligent controllers. The empirical model based intelligent controllers are becoming popular simply because it is much easier and faster to be developed. In addition to that, data based models portray promising potential (Chachuat, 2007) to be used in fed batch fermentation process control.

2.3 Fed-batch fermentation control and optimization

2.3.1 Overview

In the past, batch process control and optimization was mainly based on mechanistic models (Zhang, 2005). Mechanistic models are difficult and time consuming to develop. Therefore, mechanistic model based optimal control methods have become less attractive in the agile manufacturing environment. To overcome this problem, data based control models have become a popular choice of control method. Neural networks (Xiong et al.,2008; Zhang, 2008b;Tian et al.,2002) and fuzzy logic (Caramihai and Severin, 2009) have been reigning as a widely accepted data based control method. Neural networks are capable of approximating non-linear functions appreciably and have been used in batch process control and optimization. Neural networks are able to predict one-step-ahead or multi-step-ahead prediction are relative easy to build models (Zhang, 2005). The accuracy and robustness of neural network are depending on the quantity of training data and training method. Ideally, large training data will give a more accurate and robust neural network model. The issue with quality measurement is limited number of data available because it is dependent on sampling time and lab analysis. Therefore, it is not as abundant as the sensor measurement data collection. Therefore, modification has to be made to neural network to cope with insufficient data which makes it complicated again.

Fed-batch fermentation control and optimization is a challenging task mainly because batch processes are highly non-linear and operates in transient state (Zhang, 2005). The prime objective of fed-batch process is to produce maximum product with high quality in a safe and economical process operation. The product quality during a batch run is estimated via indirect measurements such as pH and temperature. The actual product quality data is obtained from lab analysis from the sample taken at the end of a batch run (Srinivasan et al., 2002). The challenge in control and optimization of fed-batch fermentation is to provide an accurate, long range prediction model to predict product quality at the end of a batch run (Zhang, 2005). These complexities in fed-batch fermentations call for an advanced supervisory control system (Bonvin, 1998).

2.3.2 Fed-batch fermentation system

Fermentation is usually carried out in a batch or fed-batch mode mainly because fermentation is a highly nonlinear process. Since there is no steady state for fermentation, continuous batch process mode is less attractive. Typical batch fermentation is equivalent to a cake making process, where a lump sum of substrate needed for the total microorganism growth is fed at once at the beginning of the process. The fermentation process then runs for a fixed period of time until the whole substrate is used up for maximum growth for the allocated reactor size. Then, the biomass is harvested. Following that, the batch reactor is cleaned and sterilised for a fresh new start. The new batch starts with the exact same initial parameters and reactor condition. The process repeats until the quantity is achieved.

Fed-batch is an evolution from a batch system. The fed-batch fermenter only differs in the feeding technique compared to the batch system. Instead of feeding all the substrate at once, it is divided into a few intervals. For example as shown in Figure 2.1, if the fermentation run time is 100hrs, the feeding can be divided into 5 intervals and the feeding rates can be increased or decreased at every interval in accordance to the process need.

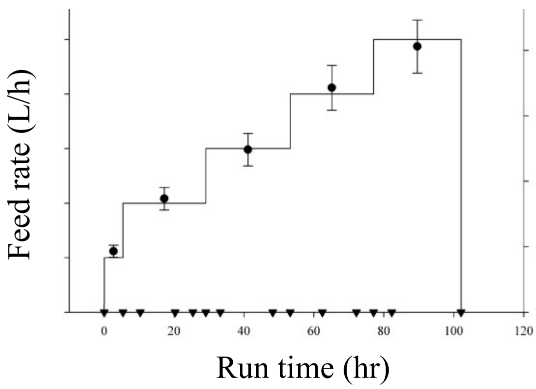


Figure 2.1: Sample feed rate profile

In a typical fed-batch fermentation process, the bioreactor starts with initial volume (V_0), biomass concentration (X_0) and substrate concentration (S_0). The substrate/feed is added over the finite reaction time or till the desired reactor volume is reached. Then, the reaction is allowed to proceed and complete at final reaction time, t_f. After the reaction is completed, the reactor is drained and cleaned for use in the next run (Shukla and Pushpavanam, 1998). This is slightly different in a sequencing batch reactor where the reactor is drained till it reaches V_0. In this case, the final condition of the first run is the initial condition of the second run.

Ideally, the objective of using fed-batch in fermentation process is to feed the substrate at the same rate that the organism utilizes it. In other words, the feeding should be optimized in accordance to the process performance. This explains why fed-batch is more attractive choice in fermentation to produce high cell density. Unfortunately, this objective is difficult to achieve. Although every single repeat runs start with the exact same initial parameters, somehow the results are always different. It is desirable that the outcome varies in an acceptable range assuring consistency to a high percentage. However, it is often the case that something goes awfully wrong in the system and could not be detected till it is too late to make any changes to save it. Eventually, at end of the fermentation run time, it is realised that the product is out of specifications and had to be discarded. That accounts for a significant waste of resources. This could be largely due to the fact that bioprocesses involve living organisms, which means expression of the microbial activities can be very complex and unpredictable. This forms the challenge in control and optimization of fed-batch fermentation process. Provided that the growth medium is in optimal condition, it is essential that the input is controlled and optimized effectively to ensure desired productivity (Jiang et al., 2012).

Fermentation biochemical pathways are usually characterised by substrate inhibition and/or product inhibition. The system controlling factors could be feed rates, product formation rate, surrounding parameters such as pH and temperature or process run times. In the past, motive of control and optimization in fed-batch fermentation were primarily on maximizing the biomass and/or product quantity (Shukla and Pushpavanam, 1998). In the recent developments, despite quantity, quality is also of importance (Xiong and Zhang, 2003; Flores-Cerrillo and MacGregor, 2005;Wang and Srinivasan, 2009; Reiss et al., 2010). A very common approach for control and optimization of fed-batch fermentation is to optimize the feed rate (Shukla and Pushpavanam, 1998). Optimal feeding policies are often affected by batch run time which is often directly related to biomass concentration (Weigand, 1981), cell growth profile (San and Stephanopoulos, 1986) and initial bioreactor input conditions (Cazzador, 1988). There are several types of feeding policies in a fed-batch reactor. They are:

i) Pulse feed: In this case the feed flow rate can be large and added to the reactor at various discrete instants. This feed profile raises the reactor volume through different stages. The concern to this type of reactor is how much to feed, which time to feed, how much time for each feeding and how much reactor volume change.

ii) Continuous feeding for a finite time: In this feeding, the feeding starts at t=0 and continues till t=tf but the feeding rate will be changing at designated sub-intervals. At certain sub-intervals it could be zero if necessary. The sub-intervals could be equally or unequally divided (Shukla and Pushpavanam, 1998).

2.3.3 Batch to batch control and optimization

Batch-to-batch or run-to-run optimization refers to the manipulation of the repetitive nature of batch processes, whereby the previous batch information can be used to improve the process recipe in order to stay closer to the desired output target for the future batches iteratively (Lee et al., 1999; Campbell et al., 2002; Xiong and Zhang, 2005a). The idea of batch to batch control existed as a solution for issues related to the absence of equilibrium. In such cases, optimization works are performed around a trajectory from previous and current batch to improve the successive batches (Srinivasan and Bonvin, 2007). The key to the successful batch to batch optimization is in determining which and how information could be extracted from previous batches and use it to improve the next batch (Bonvin, 1998; Srinivasan et al., 2001). Srinivasan et al. (2002) pointed out that batch to batch optimization is appropriate to be used when off-line product quality measurements are available. The study by Srinivasan et al. (2002) suggests that a relationship between the product quality measurements and key process variables can be developed to generate a batch to batch optimization strategy. For fermentation processes, feed rate is the main reactant and usually the deciding factor of the end-batch quality. Therefore, a relationship between product quality and feed rate can be developed. The feed rate can be used as control signals. The idea is to manipulate the feed rate effectively to achieve maximum end-batch desired product characteristics (Xiong et al., 2009).

Srinivasan et al. (2001) reported that batch to batch optimization method will result in improved yield as the batches progresses. However, batch to batch control is only efficient in controlling future batch runs thus has no control on the current batch run (Xiong et al., 2005). The batch to batch control only acts on repetitive disturbances and is not efficient for non-repetitive error (Srinivasan and Bonvin, 2007). In the event the disturbances randomly changes from one batch to another, batch to batch control would become incompetent and may amplify the repercussion of the disturbances (Lee et al., 2002). Therefore, previous studies have concluded that batch to batch control approach without integration of within batch feed-back is susceptible to perturbation thus slower in convergence (Bien and Xu, 1998; Tousain et al., 2001).

Xiong et al. (2005) conducted a study on integrated strategy by combining batch to batch ILC control and on-line control within current batch using shrinking horizon model predictive control. The idea is that the on-line control would immediately respond to disturbances within batch while the batch-to-batch ILC would correct deviation of the end-point from the desired output trajectory which is left uncorrected by the on-line controller. It was interesting to note that the on-line batch control was established in a similar formulation to the batch-to-batch control. The on-line predictive control was updated to the future control profile which was already calculated by the batch-to-batch ILC formulation. To be precise, the necessary deviation was added to the future interval control policy calculated using batch to batch control. There was no redundancy in directly calculating the future control action using on-line control. The integrated control strategy does complement each other (Lee and Lee, 2003; Flores-Cerillo and MacGregor, 2003). It eliminates disturbance in the batch process faster compared to the simple batch to batch ILC control strategy (Xiong et al., 2005).

Though the integrated control strategy does have its advantage in terms of rapid disturbance elimination, the system also has its drawback. If the predictive errors calculated from the batch-to-batch controller are not added to the predictive model within a batch, on-line control calculation ends up ‘undoing’ the correction made by the batch-to-batch controller (Lee et al., 2002). It is essential to ensure that the on-line control method is reliable and so is able to amend the future control policy while the batch is progressing (Lee and Lee, 2003). It is certainly possible but tedious.

It is true that in an end-point batch to batch control system, there is no information of the within batch quality evolution. If the desired quality trajectory is known and product quality at an intermediate phase may have significant impact on the final product quality, then it is absolutely necessary to have within batch control to closely track the desired quality trajectory. It is also essential if the particular batch process shows signs of varying, frequent disturbances that significantly affects the end-batch product quality. Otherwise, the within batch control may not be necessary.

Unless the disturbance elimination that enhances product quality within lesser amount of batches by using integrated control strategy contributes significantly in terms of overall operation cost, a simple near optimal control system is still a favourable option. In this agile manufacturing era, expensive and complex control strategies are becoming less favourable. It is deemed not necessary to closely track the desired trajectory if it does not have significant contribution to the cost effectiveness of the overall process/plant operation. A simple and reliable control system that achieves optimal product quality is more favourable when efficient cost operation and resource management are essential.

There are two types of batch to batch dynamic system controller design: the first one is to deal with the regulation problem and the second one is to deal with trajectory tracking problem. In the former problem, the controller is designed to manipulate the input variable to attain desired output even in the presence of unknown disturbances. As for the latter problem, the controller is designed to closely track desired trajectory to attain desired results (Bouakrif, 2010). Various studies have been done to design controllers to address these issues separately or simultaneously using single method or combined method. Some of the controllers used for trajectory tracking are proportional-integral-derivative (PID), adaptive control and fuzzy control. In a batch process, both the regulation and tracking trajectory problems can arise within a batch and in a batch-to-batch control (Bouakrif, 2010).

2.4 Iterative Learning Control

Iterative learning control (ILC) as the name suggest, is a self-tuning learning control technique. ILC is a favourable control technique for repetitive control systems. It is designed to exploit repetitiveness and enhance performance pattern via trial and error method (Verwoerd, 2005). Repetitive control systems do the same action over and over again. These systems usually have a finite run time and are expected to begin each trial with exact same initial conditions. When a system is repeated several times, a set of correlated and exploitable historical data is accumulated. Using the information from the repeated runs, ILC is designed to improve the control performance iteratively by gradually reducing the control error with increasing trial numbers.

A standard iterative learning control scheme is represented in Figure 2.2 below.

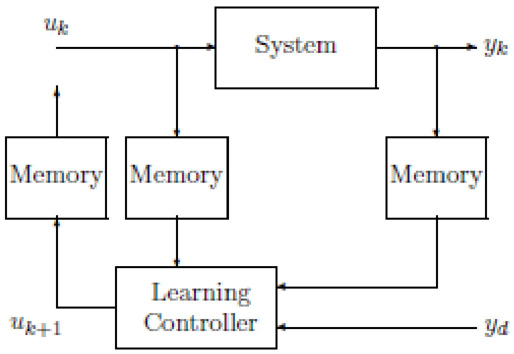


Figure 2.2: A standard iterative learning control scheme (Moore,2006)

In Figure 2.2, assuming an ILC scheme for a simple repetitive dynamic system, the input and output variables are defined as u_k and y_k respectively. The desired reference trajectory is defined as y_d .The control policy for the new trial (next batch) is defined as u_{k+1}. The control objective of ILC is to eventually find a control input, u_k that produces the corresponding output, y_k that tracks the desired trajectory, y_d as closely as possible (Owens and Daley, 2008). A precise tracking is expected as the trial number reaches infinity. Therefore, an asymptotic convergence is desired for an ILC system. As such, ILC simply means updating the control trajectory for the next batch using information from the immediate previous batch to produce an output trajectory that converges asymptotically to the desired reference trajectory (Zhang et al., 2008; Xiong and Zhang, 2003).

As seen in Figure 2.2, there are a few memory boxes which defines the distinct characteristics of the ILC. In an ILC scheme, both the input and output data of the previous trials are stored in a memory. The information in the memory database, in other words, the historical batch data and its deviation from the desired output trajectory, are exploited by the learning controller to develop a new control policy for the next batch run. The new control policy is then stored into the memory database and the cycle repeats. The learning of the controller takes place via the memory. The memory base learning allows flexibility to understand and upgrade the performance of a system more effectively.

The ILC can do both previous and current trial learning. The ILC learning scheme can be generally categorized as previous cycle learning (PCL), current cycle learning (CCL) and combination of both previous and current cycle learning (PCCL) (Xu et al., 2004). An in depth analysis and comparison of these schemes was done by Xu et al. (2004). The author concluded that PCCL provides better performance compared to the PCL and CCL. Definitely more updated information will provide better understanding of the system hence allow better control of the system.

2.4.1 ILC versus conventional feedback control

Learning about ILC and its memory based learning characteristics; it would be interesting to note how ILC differs from the conventional feedback control system. The conventional feedback control system has been there in the industrial application for a very long time. However, the performance of the system was very limited and significant manual adjustment was still necessary to achieve satisfactory performances which in many cases are not optimal. The limitation caused by the need for *a priori* knowledge of the control system inhibits competitive controller design hence the average performance (Verwoerd, 2005). This calls for advanced control systems such as ILC.

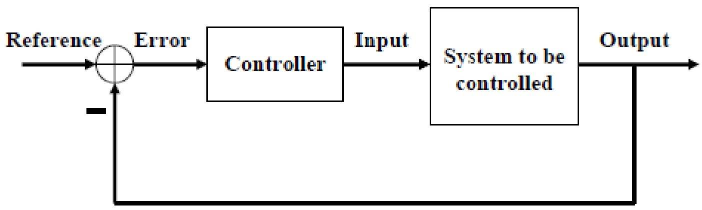


Figure 2.3: Conventional feedback control system (Moore, 2006)

Figure 2.3 shows a general conventional feedback control system. As seen in Figure 2.3, the input to the system to be controlled is adjusted based on the output deviation error from the reference trajectory. There is no memory based learning involved in this system. The reference point is fixed throughout the batch run and is changed manually as and when it is necessary. The reference point is expected to be tracked as accurately as possible. The reference trajectory is the exact expected outcome for that particular batch run. In the ILC, the desired trajectory is set higher than that could be achieved by the present batch run. The ILC control system is expected to achieve the desired trajectory iteratively while learning the system to be controlled. This way optimal performance can be obtained after several iterations in the ILC while in the conventional feedback control system the outcome is the same after every trial. There will be no improvement from trial to trial. Manual adjustment is needed at the reference trajectory if further improvement is seen feasible in the conventional feedback system. That involves offline calculations and heuristic knowledge of experienced operator. It is evident that the memory based learning characteristics of ILC overcomes the shortage of conventional feedback system. The ILC is able to self-learn the system and achieve desirable process control and optimization iteratively and automatically.

The other advantage of ILC is that it can work in two dimensions which are trial to trial in the batch direction (k) and/or from a step to step in a trial in the time (t) direction (Gao et al., 2001). The conventional feed-back system only works in time (t) dimension. This characteristic defines the calculation method for future input in each of the system.

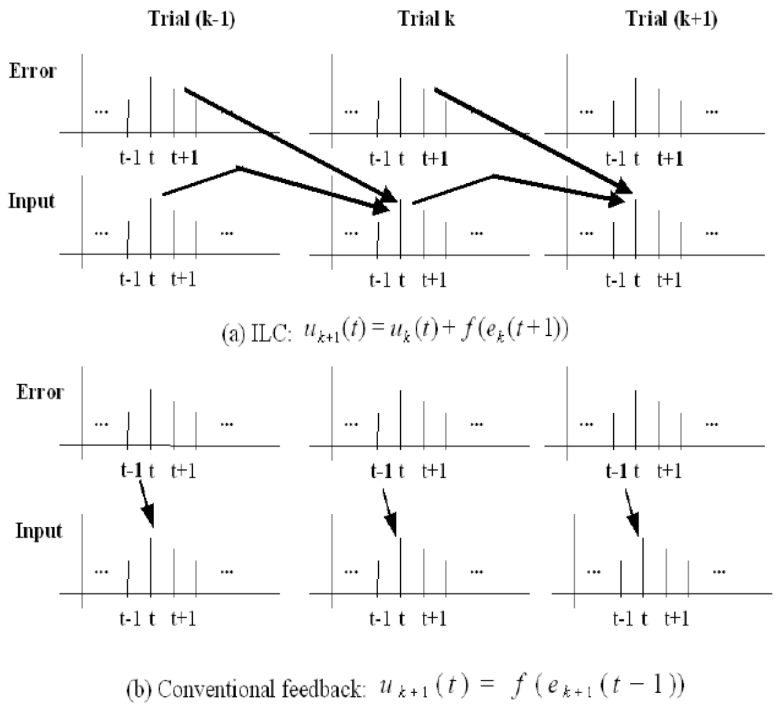


Figure 2.4: ILC versus conventional feedback control system (Moore, 2006)

A clear difference is shown in Figure 2.4 as to how future input is calculated between ILC and conventional feedback control system. For conventional feedback control system in time dimension, the error in previous time interval is used to correct the next time interval control action. The changes are meant to reduce deviation from the fixed reference trajectory for that particular batch. There is no progression from batch to batch. As for ILC, both the previous batch input and error is used to develop a control policy for the next batch run. In addition to that, ILC is also able to use information from the current cycle and past cycle to improve performance of the current trial. That way, a memory based learning that allows progressive improvement is made feasible.

2.4.2 Basic ILC algorithm

The history of ILC concept is traced back to the two popular publications by Uchiyama (1978) and Arimoto et al. (1984). Since the publication by Uchiyama (1978) is in Japanese, the publication by Arimoto et al. (1984) was initially recognised as the starting point of ILC when initial researches were conducted in US and Europe. The interpretations of ILC technique by Arimoto et al. (1984) was easily understood and widely accepted by the research community. Arimoto et al. (1984) pointed out that information from previous consecutive trials in a repetitive system can be used to develop a new input for the current repetition. Using simple ILC algorithm Arimoto et al. (1984) demonstrated that as the iteration reaches infinity, the ILC will learn the system and the tracking error will eventually become zero resulting in perfect tracking (Owens and Daley, 2008).

A typical ILC algorithm has the form :

$$u_{k+1}(t) = u_k(t) + Le_k(t)$$

$$e_k(t) = y_d(t) - y_k(t)$$

The ILC algorithm works on a standard assumptions which are:

- The system dynamics is stable
- The system returns to the same initial conditions at the start of each trial
- Each trial has the same length

From the algorithm above it is to be noted that in ILC, the new control policy, u_{k+1} is the sum of previous batch control policy, u_k and learning from previous batch, Le_k. The ILC algorithm is very simple yet highly potential. The convergence of ILC algorithm can be obtained and its proof can directly be derived from the convergence theorems in literature (Dinh Van et al., 2012; Xiong et al., 2005; Xu and Tan, 2003).

ILC is expected to learn from the repeated run of the control system and how the learning takes places is what matters in an ILC algorithm. The learning takes place via the error term, Le_k. The vast numbers of ILC algorithms are developed from different ways of using the information in the error term. The error, e_k is simply the deviation of the previous output, y_k from the desired trajectory, y_d. The learning gain, L plays a vital role in learning the system and determining the control action for the current batch. In orientating the basic ILC algorithm, the task is to identify an algorithm which generates a control input that progressively improves the product quality of the following batches (Gao et al., 2001).

The initial ILC controllers were the P-type ILC (Arimoto et al., 1985) and D-type ILC (Arimoto et al., 1984). Both the P-type and D-type ILC algorithms have the same outlook as the equation presented above. The difference is on how the error term is used in the algorithm. In the D-type ILC the error is the derivative while in the P-type ILC the error is simply used with the proportional gain. The P-type is preferred to the D-type due to potential small noise signal amplification though differentiation which may eventually affect the control system stability (Cai, 2009; Xu and Tan, 2003).

Following the initial P-type and D-type ILC algorithms more studies were done in this area to further improve the algorithm (Wang, 2000; Saab, 2003; Cai, 2009). The other combinations of the P, I and D terms were studied and its impact on the ILC algorithm has been evaluated. There are P-type ILC (Wang, 2000; Moore, 2001; Moore et al.,2002; Saab, 2003; Ratcliffe et al., 2005), D-type ILC (Wang, 2000; Saab, 2003), PI-type ILC (Astrom et al.,1998; Chen and Moore, 2002), PD-type ILC (Chen and Hwang, 2006; Baolin et al., 2006) and PID-type ILC (Park et al., 1999; Ji and Luo, 2005; Chien, 2006) algorithms studied on various aspects such as monotonic convergence, robustness, trajectory tracking and applications. Each of these combined techniques has its own advantages and disadvantages. Regardless of those, the interesting part of these combinations would be to understand how each of the proportional (P), integral (I) and derivative (D) affects the ILC algorithm. The P-component contributes to the stability hence ensuring monotonic convergence, the I-component rejects the effect of non-zero initial error thus increasing the convergence rate while the D-term reduces the effect of input disturbances (Madady, 2008). The merits of each of the PID terms makes it a popular choice for the ILC algorithm.

2.4.3 PID (Proportional plus Integral and Derivative) controller versus ILC

ILC can be used to directly update the input policy (Amann et al., 1996; Xiong et al., 2005) by replacing the current PID controller. The issue with PID controllers is that its constant coefficients do not allow implementation of adaptive behaviour to the controlled system. Therefore, the controller is only able to track the desired trajectory to a moderate degree of accuracy and the tracking results are the same for each repetition. The ILC works on the trial dependent update law which allows adaptive behaviour (Verwoerd, 2005). ILC mimics human learning process and therefore works iteratively to progressively improve control accuracy.

2.4.4 PID (Proportional plus Integral and Derivative) controller assisted ILC

ILC can also be used as a supervisory control scheme (Riad et al., 2009). At present, the PID controllers are widely used in industrial applications. The PID controllers are very popular simply because it has simple structure, robust and effective in tracking a fixed reference trajectory. Therefore, rather than uprooting the whole present control scheme, it would be economical to enhance it. ILC can complement the existing control scheme. ILC ideally functions as a feed-forward compensator for open loop control systems (Gao et al., 2001). In such systems the feed-forward action is magnified to enhance convergence without compensating closed loop stability. ILC can be incorporated either in serial arrangement or parallel arrangement. The serial arrangement could lead to increment in noise. The parallel arrangement is preferred because the ILC controller could act independently in the event of unexpected disturbances in the system dynamic which may affect the system stability (Cai, 2009).

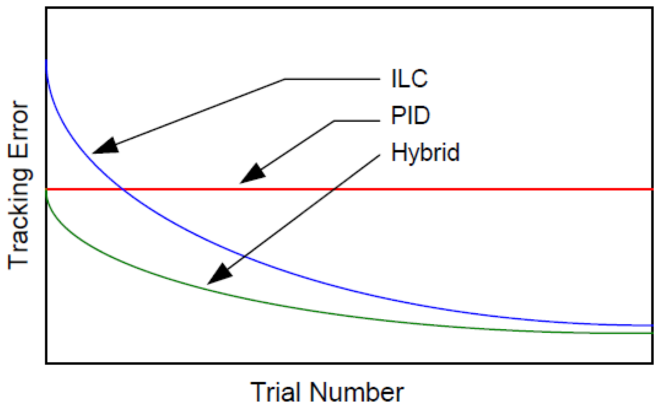


Figure 2.5: Performance comparison of ILC, PID and hybrid approaches (Cai, 2009)

Figure 2.5 clearly exhibits the performance of ILC, PID and PID assisted ILC (hybrid) control scheme. The red line representing PID controller in Figure 2.5 shows that the tracking error is the same for every trial. There is no improvement in trial to trial when using PID controller alone. The blue line representing ILC controller shows that the tracking error reduced as the trial number increases. The green line representing the PID assisted ILC (hybrid) controller, the trial to trial error reduction is even better for many cases. As to what extent the hybrid controller performs better than the ILC control depends on the system which is being controlled. It is important to note that towards a bigger trial number, the difference in the tracking error of the ILC and hybrid controller becomes smaller.

2.4.5 ILC versus other learning controllers

Learning controllers are intelligent controllers that are developed to emulate human learning behaviour to a certain extent. All the learning controllers vary on the basis of how the information from a control system is extracted, represented, stored and updated to enhance control performance (Verwoerd, 2005). ILC is one of the learning controllers. There are other popular learning controllers that are widely used in the industry and research arena other than ILC. They are adaptive controllers, neural networks and repetitive controllers (Cai, 2009). ILC portrays significant difference from these controllers which makes it unique for application in repetitive systems. For instance, the adaptive controller does not use the repetitive information while ILC works entirely based on stored repetitive information. The adaptive controller design modifies the control law of a controller to adapt to the changes in the system while the ILC only modify the control input signals. The neural network modifies controller parameters and calls for large network modifications. Therefore, extensive training data is required and relatively fast convergence is difficult making it less attractive for real-time process because it takes longer to calculate. The ILC is attractive due to its simplicity and potential effectiveness. The repetitive control (RC) has a very similar concept to ILC. The difference is that the initial condition is set back to the same value for every repeat in ILC whereas in the RC the initial condition continues from the final condition of the previous run. The RC is used for continuous system while ILC for discontinuous system. Therefore, ILC is applicable to fed-batch process while RC is used in continuous batch process (Bristow et al., 2006; Tan et al., 2009; Cai, 2009). In summary, the distinct difference between these controllers and ILC is that these controllers assume adjacent control tasks to be related while ILC assumes identical control task trial after trial (Cai, 2009).

2.4.6 Advantages of ILC

Iterative learning control is an attractive choice of control method for a system that repeats with the exact same initial conditions and runs for a finite period of time. ILC is designed to use previous information to improve the system performance iteratively. This method works in such a way that the tracking error (between the output and specified desired reference trajectory) or the state error (between real state and the reference model state) is sequentially reduced to zero from run-to-run (Bouakrif, 2010).

The most distinct advantageous characteristic of ILC is its capability to capture previous trials information and exploit every possibility to incorporate the information into present trial control input. ILC has a very simple structure that works on a batch domain, time domain or both. It is open-loop on the time domain and closed loop on the iteration domain (Cai, 2009). Its memory based learning records the necessary information on either or both the domains which will then be synthesized into current batch control policy.

There are two main ILC algorithm types. One of them is algorithms with no knowledge of the system and works based on tracking error vector. The other is the model based algorithms where a mathematical representation of the plant in incorporated into the algorithm (Cia, 2009). In other words, ILC can work on systems with minimal information. ILC is capable of desired output trajectory tracking without any prior knowledge of the system state. ILC design can be model independent with limited offline measurement requirement (Waissman et al., 2002). Since ILC uses information from previous executions in order to improve tracking performance from trial to trial, it does not require any on-line measurement or on-line sensor. However, it is important that the control task which is the desired output trajectory is the same for all the trials.

It is interesting to note that ILC can also steer a system with uncertainties to follow a reference model which is independent of the real system, which means neither the parameters nor the structure of both the systems have to be same (Bouakrif, 2010). Bouakrif (2010) has proven that the real state system is convergent to the reference model state in the presence of disturbances to achieve desired performance. The reference model state used in the study has different structure and parameter compared to the real system state.

2.4.7 Limitations of ILC

There are a few limitations to ILC. ILC assumes reset to the same initial conditions for every repeat. Perfect reset trial after trial is not always feasible. However, learning could still take place for perturbed initial conditions. Hence, control performance can still be improved (Verwoerd, 2005).

Any dynamic systems are prone to have uncertainties. Significant and uncontrolled uncertainties will lead to unsatisfactory performance. It is important that a controller is able to suppress or possibly eliminate the uncertainties in the system to render stable and optimal performance. The uncertainties can be classified as repeatable and non-repeatable. Repeatable uncertainties are invariant over iterations and can easily be detected. ILC works very well with continuous disturbance from trial to trial. Non-repeatable uncertainties are variants over iterations which are difficult to detect and nature of the disturbance maybe unknown (Bouakrif, 2010). The performance of ILC can be affected for non-repetitive uncertainties depending on the type and magnitude of the disturbance.

ILC applied on iteration axis, will improve the batch to batch performance but the within batch performance may have been neglected. There is often trade-off between trial to trial error reduction and within trial performance.

ILC uses pattern recognition technique to learn the system to be controlled. Therefore the learning of the controller is limited to the stored knowledge. The quantity and particularly quality of the stored information will affect the controller performance.

2.4.8 Desirable performance criterion for ILC

The objective of ILC is to achieve asymptotic convergence. An asymptotic convergence is when the solution is not achieved but a result parallel to desired trajectory is attained and sustained. An asymptotic convergence is a very desirable performance of an optimization algorithm in global convergence (Chachuat, 2007). Asymptotic stability of the closed-loop system is guaranteed over the whole finite time intervals when the iteration number tends to infinity (Bouakrif F., 2010).

Performance of an ILC algorithm is evaluated based on its convergence speed, minimum tracking error and long-term stability. Convergence speed is a measure of how fast a desired trajectory is reached without compensating the stability of the system. It is important that the model is stable for both the time and batch dimensions. The stability of time direction is usual and simple. In a batch-to-batch optimization ILC, stability of k^{th} batch is influenced by the convergence rate. For a little known plant/process model, fast convergence is less robust while slow convergence is more robust (Amann et al., 1996). Previous studies in this area proved that an ILC algorithm based on optimisation principles ensures stability, convergence and robustness (Amann et al., 1996; Gao et al., 2001).

A quadratic cost objective function has to be solved to minimize tracking error whilst optimizing control policy to attain desired output trajectory. The cost function simply represents identifying optimal control policy change for $(k+1)^{th}$ batch by reducing tracking error and without deviating too much from the previous control input to ensure stability and robustness (Amann et al., 1996). It is also a major concern in the real industry to minimize the number of batches needed to achieve desired target in order to optimize effective production.

2.5 Batch to batch model predictive iterative learning control

Initially, the ILC algorithm was used as an open-loop feed-forward controller but it was insufficient to handle disturbances on-line and slow to converge (Bien and Xu, 1998; Cueli and Bordon, 2008; Lee et al., 1999). Combination of the feedback control to the feed-forward based ILC is reported to exhibit exponential convergence by Amann et al. (1996). A combination of the batch to batch control into ILC can address both repetitive and non-repetitive disturbances (Cueli and Bordons, 2008). In the batch to batch iterative learning control approach, the input trajectory is updated directly by the feedback control after every batch (Xiong and Zhang, 2003).

It is essential that development of a batch to batch ILC should also take into consideration that it does not respond to the random noise within a batch and produce variances in the final output. In order to tackle this problem, application of model predictive control (MPC) is advantageous because of its ability to differentiate between process and state noise which is a lacking criteria in a batch to batch controller (Campbell et al., 2002). MPC is the most popular linear method and is widely accepted in the industry (Ashoori et al., 2009). Having known that the process variables in a batch process will possess highly non-linear changes throughout the operation, the conventional MPC is expected to suffer severe tracking errors issue (Morari and Lee, 1999; Ashoori et al., 2009). Xiong and Zhang (2003) expressed that tuning a system using iterative learning control (ILC) will enhance the performance of tracking control system.

In a few studies, a combination of ILC and MPC was suggested expecting combined benefit of these methods (Lee et al., 2000; Srinivasan et al., 2002; Chen and Fang, 2004; Cueli and Bordons, 2008). However, it was reported that combined application of ILC control and MPC causes conflicting effect. MPC will refute the control action of ILC due to discrepancy in the prediction of the two. This issue calls for integration of ILC and MPC (Srinivasan and Bonvin, 2007; Lee and Lee, 2003). Built in controllers into the system is expected to be more efficient and simpler in comparison to designing feedback controller separately (Lee et al., 1999). Lee and Lee (2003) studied on the integration and noted that MPC imparts on-line corrective action while ILC focus on refining disturbances to achieve desired end product. They also added that MPC deals with real-time feedback and ILC have large robustness margin.

Xiong and Zhang (2003) worked on the batch to batch linearized perturbation model iterative learning control method. In order to tackle process nonlinearities they have worked on the perturbation variables than the measured process variables themselves. In their work, the perturbation model is updated after every batch by using the immediate previous batch as the nominal batch. This way, the unexpected process and parametric disturbances is expected to be captured and removed to render a more precise model prediction. This method is expected to address process nonlinearities, repetitive disturbances, non-repetitive disturbances, random noise in the batch, model-plant mismatches and reference and set point trajectory tracking issue. It seems to be the potential solution for the issues in bioprocess control and optimization. This area of study is still in infancy and can be fine-tuned to be an efficient and widespread bioprocess control and optimization algorithm. In this work, the prospective of batch to batch perturbation model iterative learning control optimization method in fermentation process is studied.

2.6 Product quality control and optimization

Bioprocess monitoring, control and optimization are important in order to attain and sustain product quality (Neeleman, 2002; Johnson, 1987). The quality of the product is adversely affected by the presence of by-product. Bonvin (1998) mentioned that product quality does not mean highest possible purity but lowest possible variation amongst optimal purity as well as negligible or zero by-product production. Product variation has direct effect on the marketability, customer acceptance and production cost. This is indeed a critical requirement in the pharmaceutical industry because it determines the exact dosage needed to be delivered to a patient's health requirement. Simultaneously, production cost and waste generation has to be controlled to ensure economical and sustainable production (Zhang et al., 2008; Bonvin, 1998; Muske et al., 2004; Sendin et al., 2006).

It is understood that the most accurate long range predictions of product quality variables can be obtained with the presence of a comprehensive mechanistic model of the process (Muske et al., 2004) but it is expensive and time-consuming. On-line robust quality measurement sensors are almost nil for industrial use today. In the industry, the final product quality is controlled through measurable variables such as pH, temperature and feed rate. The optimal trajectory of the measured variable is set and every batch run for the same process follows the similar fixed pattern (Xiong and Zhang, 2003). This strategy fails when process disturbance in non-measured variables such as feedback condition, raw material properties, impurities and catalyst activities is present and affects the product quality (Lee et al., 1999). This means a consistent input trajectory will not ensure product quality especially when unknown disturbances are present. This setback brings about the idea to update the input trajectory after every run and re-optimize to achieve desired product quality iteratively (Xiong and Zhang, 2003).

It is important to take note that, measured variables alone cannot determine a product quality. It is of best interest to directly optimize product quality trajectory (Lee et al., 1999). In practice today, quality measure is done in offline analysis. Samples are collected at few intervals during the batch run and at the end of a batch and analysis is done offline (Srinivasan et al., 2002). Providing a reference trajectory for product quality for a whole batch is reasonably tough because the values may vary in a wide range throughout a batch process. Therefore, batch to batch control can be used to improve product quality by developing a reference trajectory based on the final product quality value of immediate previous batch (Xiong and Zhang, 2005a).

Gao et al. (2001) reviewed that by adding weighting matrices, Q and R to the objective function, process disturbances and possible differences in the process initial values can be apprehended. The Q and R ratio and not the real values influence the elimination of disturbances and ability to track back the desired trajectory. The improvised objective function and linearised perturbation model batch to batch iterative learning control have been used in the study. This method is expected to address process nonlinearities, repetitive disturbances, non-repetitive disturbances, random noise in the batch, model-plant mismatches and reference and set point trajectory tracking issue.

Models used in ILC can be obtained using multivariate analysis techniques or regression models such as PLS and PCA (Bonvin, 1998; Wang and Srinivasan, 2009). Zhang (2008a) added that PLS and PCA and its variants have been delivering robust empirical models for data with high colinearity and dimensionality. Multiple linear regression (MLR) is widely accepted as linear regression relating predictor variable and response variable for steady state processes. However, this method fails in the presence of high degree of correlation among variables. In such situation, latent variable methods need to be used.

2.7 Introduction of MLR, PCR and PLS

A brief introduction of MLR, PCR and PLS is presented in this section. Details about PLS and PCR can be found in (Geladi and Kowalski, 1986).

2.7.1 Multiple linear regressions

Considering the following linear model

$$y = x_1\theta_1 + x_2\theta_2 + \ldots + x_n\theta_n \tag{2.1}$$

where y is the model output, x_1 to x_n are model inputs, and θ_1 to θ_n is model parameters.

Given a set of input and output data, **X** and **Y**, the model parameters can be obtained from MLR as

$$\hat{\boldsymbol{\theta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \tag{2.2}$$

where $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1\ \hat{\theta}_2 \cdots \hat{\theta}_n)^T$ is a vector of the estimates of model parameters.

In many batch processes, the control policies are typically determined to optimise the product quality at the end of a batch. Therefore, the control actions during different stages of a batch are usually correlated. In such cases, appropriate linearised model may not be obtained from MLR. When the model input variables are correlated, Eq. (2.2) gives unreliable estimates since $(\mathbf{X}^T\mathbf{X})^{-1}$ is close to singular. To overcome the colinearity in the regression variables, PCR or PLS can be used to obtain the linearised models (Geladi and Kowalski, 1986).

2.7.2 Principal component regression

As for PCR method, the matrix **X** is decomposed into the sum of a series of rank one matrix through principal component decomposition.

$$\mathbf{X} = \mathbf{t}_1\mathbf{p}_1^T + \mathbf{t}_2\mathbf{p}_2^T + \ldots + \mathbf{t}_N\mathbf{p}_N^T \tag{2.3}$$

In the above equation, $\mathbf{t}_i$ and $\mathbf{p}_i$ are the i^{th} score vector and loading vector respectively. Both the score vectors and loading vectors are orthogonal and of unit length. The loading vector $\mathbf{p}_1$ defines the direction of the greatest variability. The score vector $\mathbf{t}_1$, also known as the first principal component, represents the projection of each column of **X** onto $\mathbf{p}_1$. The first principal component is thus the linear combination of the columns in **X** explaining the greatest amount of variability ($\mathbf{t}_1=\mathbf{X}\mathbf{p}_1$). The second principal component is the linear combination of the columns in **X** explaining the next greatest amount of variability ($\mathbf{t}_2=\mathbf{X}\mathbf{p}_2$) subject to the condition that it is orthogonal to the first principal component. Principal components are arranged in decreasing order of variability explained. Since the columns in **X** are highly correlated, the first few principal components can explain the majority of data variability in **X**.

$$\mathbf{X} = \mathbf{T}_k \mathbf{P}_k^T + \mathbf{E} = \sum_{i=1}^{k} \mathbf{t}_i \mathbf{p}_i^T + \mathbf{E} \tag{2.4}$$

where $\mathbf{T}_k = [\mathbf{t}_1\ \mathbf{t}_2\ ...\ \mathbf{t}_k]$, $\mathbf{P}_k = [\mathbf{p}_1\ \mathbf{p}_2\ ...\ \mathbf{p}_k]$, k represents the number of principal components to retain, and **E** is a matrix of residuals of unfitted variation.

If the first k principal components can adequately represent the original data set **X**, then regression can be performed on the first k principal components. The model output is obtained as a linear combination of the first k principal components of **X** as

$$\hat{\mathbf{Y}} = \mathbf{T}_k \mathbf{w} = \mathbf{X}\mathbf{P}_k \mathbf{w} \tag{2.5}$$

where **w** is a vector of model parameters in terms of principal components.

The least squares estimation of **w** is:

$$\mathbf{w} = (\mathbf{T}_k^T \mathbf{T}_k)^{-1} \mathbf{T}_k^T \mathbf{Y} \ = (\mathbf{P}_k^T \mathbf{X}^T \mathbf{X}\mathbf{P}_k)^{-1} \mathbf{P}_k^T \mathbf{X}^T \mathbf{Y} \tag{2.6}$$

The model parameters calculated through PCR is then

$$\boldsymbol{\theta} = \mathbf{P}_k \mathbf{w} \ = \mathbf{P}_k (\mathbf{P}_k^T \mathbf{X}^T \mathbf{X}\mathbf{P}_k)^{-1} \mathbf{P}_k^T \mathbf{X}^T \mathbf{Y} \tag{2.7}$$

Eq. 2.7 is equivalent to Eq. 2.2.

2.7.3 Partial least squares

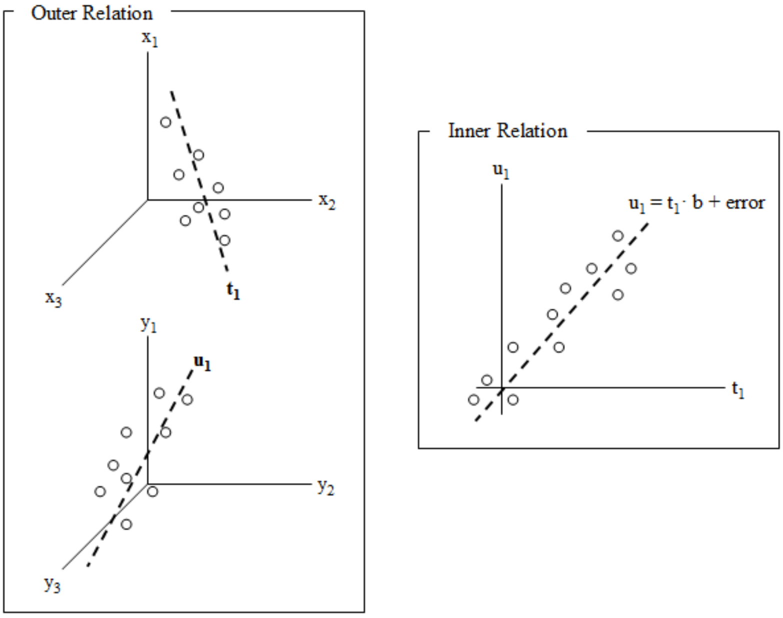


Figure 2.6: Description of PLS (Hong, 2011)

Referring to Figure 2.6, PLS projects the X and Y matrices to a subset of latent variables, t and u, respectively.

$$\mathbf{X} = \sum_{j=1}^{k} \mathbf{t}_j \mathbf{p}_j^T + \mathbf{E} \quad (2.8)$$

$$\mathbf{Y} = \sum_{j=1}^{k} \mathbf{u}_j \mathbf{q}_j^T + \mathbf{F} \quad (2.9)$$

E and F are residual matrices of unfitted variation. If sufficient eigenvectors (large k) are used, then both E and F can be made zero. The objective is to fit a linear relationship between X and Y by performing least square regression between each pair of corresponding t and u latent vectors while making$\|F\|$ as small as possible.

$$\hat{\mathbf{u}}_j = \mathbf{t}_j \mathbf{b}_j \qquad j=1,2,\ldots k \quad (2.10)$$

where b_j is the coefficient from the inner linear regression between the j^{th} latent variables u_j and t_j which is

$$\mathbf{b}_j = \left(\mathbf{t}_j^T \mathbf{t}_j\right)^{-1} \mathbf{t}_j^T \mathbf{u}_j \quad (2.11)$$

Eq. 2.11 is equivalent to Eq. 2.2 and Eq. 2.7. PLS provides a bilinear decomposition of the **X** and **Y** matrices into a number of rank-one matrices. The decomposition can be defined as the product between each pair of input scores vector, **t**, and predicted output scores vector, $\hat{\mathbf{u}}_j$, and a set of corresponding input and output loading vectors **p** and **q**.

The number of principal components, k, to be retained in the model for both PLS and PCR is usually determined through cross-validation (Wold, 1978). The data set for building a model is partitioned into a training data set and a testing data set. The PCR or PLS models with different number of principal components are developed on the training data and then tested on the testing data. The model with the smallest testing errors is then selected.

2.8 Theory of batch to batch iterative learning control

2.8.1 Nonlinear representation of batch processes

The model based ILC developed by Xiong and Zhang (2003) is reviewed here. Consider batch processes where the batch run length (t_f) is fixed and consists of N sampling intervals (i.e. $N=t_f/h$, with h being the sampling time). Product quality variables (outputs), $y \in R^n$ ($n \geq 1$), can be obtained off-line by analysing the samples taken during the batch run and the manipulated variable, $u \in R^m$, can be measured at each sampling time on-line. The product quality and control trajectories are defined, respectively, as

$$\mathbf{Y}_k = [y_k^T(1), y_k^T(2), \ldots, y_k^T(N)]^T \qquad (2.12)$$

$$\mathbf{U}_k = [u_k(0), u_k(1), \ldots, u_k(N\text{-}1)]^T \qquad (2.13)$$

where the subscript k denotes the batch index. The desired reference trajectories of product quality are defined as

$$\mathbf{Y}_d = [y_d^T(1), y_d^T(2), \ldots, y_d^T(N)]^T \qquad (2.14)$$

A batch operation is typically modelled with a dynamic model, but it would be convenient to consider a static function relating the control sequence to the product quality sequences over the whole batch duration (Lee et al., 1999). Due to the causality, the product quality variables at time t, $y_k(t)$, is a non-linear function of all control actions up to time t,

$\mathbf{U}_k(t) = [u_k(0), u_k(1), \ldots, u_k(t\text{-}1)]^T$, i.e.

$$y_k(t) = f_t(\mathbf{U}_k(t)) + v_k(t), \quad t=1, 2, ..., N, \quad y_k(0)=y_0 \qquad (2.15)$$

where $f_t(\cdot)$ represents the non-linear function between $\mathbf{U}_k(t)$ and $y_k(t)$ and $v_k(t)$ is the measurement noise at time t. Eq. (2.15) can be rewritten in matrix form as

$$\mathbf{Y}_k = \mathbf{F}(\mathbf{U}_k) + \mathbf{v}_k \qquad (2.16)$$

where $\mathbf{F}(\cdot)$ represents the non-linear static functions between $\mathbf{U}_k(t)$ and $y_k(t)$ at different sampling times and $\mathbf{v}_k = [v_k^T(0), v_k^T(1), \ldots, v_k^T(N\text{-}1)]^T$ is a vector of measurement noises

2.8.2 Linearisation of nonlinear batch process model

In batch processes, linearization is done around nominal process operation trajectory because unlike continuous process, there is no steady state. The nominal trajectory can be the mean of all the previous control trajectories and its corresponding outputs or the immediate previous batch control trajectory and its corresponding output. Both these nominal trajectories have been tested in this study to understand the effect of it in achieving desired output. The nominal trajectories can also be a fixed best performing batch data. This too has been tested but found to be not very promising for a batch to batch process control and optimization

Let the nominal trajectory and its corresponding yield to be $\mathbf{Y}_s$ and $\mathbf{U}_s$. The subscript *s* denotes nominal batch. Linearising the non-linear batch process model described by Eq. (2.16) with respect to $\mathbf{U}_s$ around the nominal trajectories ($\mathbf{U}_s$, $\mathbf{Y}_s$), the following can be obtained.

$$\mathbf{Y} = \mathbf{Y}_s + \left.\frac{\partial \mathbf{F}(\mathbf{U})}{\partial \mathbf{U}}\right|_{\mathbf{U}_s} (\mathbf{U} - \mathbf{U}_s) + \mathbf{w} + \mathbf{v} \tag{2.17}$$

where $\mathbf{w} = [w(1), w(2)\ldots w(N)]^T$ is a sequence of model errors due to the linearisation and $\mathbf{v}$ represents the effects of noise and unmeasured disturbances. The linearised model is defined as $\mathbf{G}_s$ and it represents the equation below.

$$\mathbf{G}_s = \left.\frac{\partial \mathbf{F}(\mathbf{U})}{\partial \mathbf{U}}\right|_{\mathbf{U}_s} \tag{2.18}$$

The structure of $\mathbf{G}_s$ is restricted to the following lower-block-triangular form due to the causality.

$$\mathbf{G}_s = \begin{bmatrix} g_{11} & 0 & \cdots & 0 \\ g_{21} & g_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N1} & g_{N2} & \cdots & g_{NN} \end{bmatrix} \tag{2.19}$$

2.8.3 Linear time varying pertubation variables and model

In batch process operation, linearisation of the changes in variables, also termed as pertubation variables, is preferred in comparison to the actual variable values. The reason is by using pertubation variables, significant batch to batch process nonlinearities are eliminated resulting in a more accurate linear models for control purpose. The pertubation variables can be obtained by either substracting a preset nominal trajectories (Xiong and Zhang, 2003) from the available batch data or substraction between two adjacent batches (Vilas et al., 2004). The substraction of two adjacent batches is not favoured because the difference could be small and the developed model may not be accurate. The sustraction from the preset nominal trajectory is preferred in this work. The nominal trajectories can be a mean of all the historical batch data or the best performing batch data amongst all the historical batches and fixed for all the batch runs. In a batch to batch process control, the historical data pool is enriched after every batch run. This gives a freedom to reidentify the nominal trajectories for the betterment of the process control. Therefore the nominal trajectory can be

a) mean of all historical batches; the mean value can be reidentified after every batch run using a new set of batch data
b) best batch data, reidentified after every batch run from the growing historical batch number
c) immediate previous batch data, updated after every batch run

The mean trajectory method may not represent the current batch environment. The mean can be affected by other lower or much better performing batches and so the control policy may not be suitable for the current process condition. The best data as reference trajectory, may not allow development of control policy for current batch performance/condition, because the control policy will be developed based on the best performance and not based on the possibly affected batch performance.Therefore, in this work, the pertubation variables are obtained by substracting the immediate previous batch data from the historical batches. This way the most recent batch environment e.g presence of unknown disturbances, one-off disturbances can be represented and necessary control action for that condition can be developed for improvement of the following batch.

The pertubation variables of the control and product quality variables are defined, respectively as

$$\bar{\mathbf{U}} = \mathbf{U} - \mathbf{U}_s \qquad (2.20)$$

$$\bar{\mathbf{Y}} = \mathbf{Y} - \mathbf{Y}_s \qquad (2.21)$$

Subsequently, the linearised perturbation variable obtained from Eq.2.17 is

$$\bar{\mathbf{Y}} = \mathbf{G}_s \bar{\mathbf{U}} + \mathbf{d} \qquad (2.22)$$

where $\mathbf{d}$ is defined as the model disturbance sequence

$$\mathbf{d} = \boldsymbol{w} + \mathbf{v} \qquad (2.23)$$

and is supposed to be bounded by a small positive constant B_d as shown below

$$|\mathbf{d}| < B_d \qquad (2.24)$$

The $\mathbf{G_s}$ here is considered a linear time varying (LTV) perturbation model because it varies with $\mathbf{U_s}$, which usually varies from batch to batch. The linearised perturbation model can be identified from historical process operation data using MLR, PLS and PCR (Xiong and Zhang, 2003). To address the process deviation, the linearised model can be re-identified after each batch with data from the most recent batch added to the historical process data. Furthermore, the control trajectory and quality variable trajectory from the most recent batch can be used as the nominal trajectories.

2.8.4 Batch to batch iterative learning control

As batch process dynamics are non-linear and the perturbation model is linearised around the nominal operation trajectories of a batch process, offsets always occur due to modelling errors and unmeasured disturbances. The perturbation model predictions of the current batch run can be corrected by adding model prediction residuals of previous batch runs (Xiond and Zhang, 2003).

The prediction of perturbation model is defined as

$$\hat{\bar{\mathbf{Y}}}_k = \hat{\mathbf{G}}_s \bar{\mathbf{U}}_k \qquad (2.25)$$

and the absolute model prediction is defined as

$$\hat{\mathbf{Y}}_k = \mathbf{Y}_s + \hat{\bar{\mathbf{Y}}}_k = \mathbf{Y}_s + \hat{\mathbf{G}}_s \bar{\mathbf{U}}_k \qquad (2.26)$$

After completion of the k^{th} batch run, prediction errors between off-line measured or analysed product qualities and their model predictions can be calculated as

$$\varepsilon_k = \mathbf{Y}_k - \hat{\mathbf{Y}}_k = \overline{\mathbf{Y}}_k - \hat{\overline{\mathbf{Y}}}_k \tag{2.27}$$

Based on the prediction errors of the k^{th} batch run, the modified prediction of perturbation model in the $(k+1)^{th}$ batch run is obtained as

$$\tilde{\overline{\mathbf{Y}}}_{k+1} = \hat{\overline{\mathbf{Y}}}_{k+1} + \varepsilon_k \tag{2.28}$$

The absolute modified model prediction is defined as

$$\tilde{\mathbf{Y}}_{k+1} = \hat{\mathbf{Y}}_{k+1} + \varepsilon_k = \mathbf{Y}_s + \hat{\overline{\mathbf{Y}}}_{k+1} + \varepsilon_k \tag{2.29}$$

The modified prediction error is defined as

$$\tilde{\varepsilon}_{k+1} = \mathbf{Y}_{k+1} - \tilde{\mathbf{Y}}_{k+1} = \overline{\mathbf{Y}}_{k+1} - \tilde{\overline{\mathbf{Y}}}_{k+1} \tag{2.30}$$

From the definitions in Eq(2.27) and Eq(2.28),

$$\tilde{\varepsilon}_{k+1} = \varepsilon_{k+1} - \varepsilon_k \tag{2.31}$$

Assuming that the prediction error of the perturbation model is bounded by a certain small positive constant B_m such that

$$|\varepsilon_k| < B_m \tag{2.32}$$

The prediction error bound B_m is a measure to represent the deviation of $\hat{\overline{\mathbf{Y}}}_k$ from $\overline{\mathbf{Y}}_k$ or $\hat{\mathbf{Y}}_k$ from $\mathbf{Y}_k$. The higher the value of B_m is, the poorer the identified model is. The modified prediction error is bounded by $2B_m$ as follows

$$|\tilde{\varepsilon}_k| < |\varepsilon_k| + |\varepsilon_{k-1}| < 2B_m \tag{2.33}$$

The tracking errors of process and perturbation model are respectively defined as

$$\mathbf{e}_k = \mathbf{Y}_d - \mathbf{Y}_k = \overline{\mathbf{Y}}_d - \overline{\mathbf{Y}}_k \tag{2.34}$$

$$\hat{\mathbf{e}}_k = \mathbf{Y}_d - \hat{\mathbf{Y}}_k = \overline{\mathbf{Y}}_d - \hat{\overline{\mathbf{Y}}}_k \tag{2.35}$$

where $\overline{\mathbf{Y}}_d$ is the deviated desired trajectory and defined as

$$\overline{\mathbf{Y}}_d = \mathbf{Y}_d - \mathbf{Y}_s \tag{2.36}$$

The tracking errors of modified prediction of perturbation model is defined as

$$\tilde{\mathbf{e}}_k = \mathbf{Y}_d - \tilde{\mathbf{Y}}_k = \overline{\mathbf{Y}}_d - \tilde{\tilde{\overline{\mathbf{Y}}}}_k \tag{2.37}$$

From the definitions in Eq(2.27), Eq(2.34) and Eq(2.37), the following relationship among these three tracking errors can be obtained

$$\varepsilon_k = \hat{\mathbf{e}}_k - \mathbf{e}_k \tag{2.38}$$

$$\tilde{\mathbf{e}}_k = \hat{\mathbf{e}}_k - \boldsymbol{\varepsilon}_{k-1} \tag{2.39}$$

From Eq(2.35) and Eq(2.25), an iterative relationship for $\hat{\mathbf{e}}_k$ along the batch index k can be obtained as

$$\hat{\mathbf{e}}_{k+1} = \hat{\mathbf{e}}_k - \hat{\mathbf{G}}_s \Delta\overline{\mathbf{U}}_{k+1} \tag{2.40}$$

where $\Delta\overline{\mathbf{U}}_{k+1}$ is defined as

$$\Delta\overline{\mathbf{U}}_{k+1} = \overline{\mathbf{U}}_{k+1} - \overline{\mathbf{U}}_k \tag{2.41}$$

From the definition of perturbation variables, we can have

$$\Delta\overline{\mathbf{U}}_{k+1} = \overline{\mathbf{U}}_{k+1} - \overline{\mathbf{U}}_k = \mathbf{U}_{k+1} - \mathbf{U}_k \tag{2.42}$$

Substitute Eq(2.38) and Eq(2.40) to Eq(2.39), we have

$$\tilde{\mathbf{e}}_{k+1} = \hat{\mathbf{e}}_{k+1} - (\hat{\mathbf{e}}_k - \mathbf{e}_k) = \mathbf{e}_k - \hat{\mathbf{G}}_s \Delta\overline{\mathbf{U}}_{k+1} \tag{2.43}$$

On the other hand, Eq(2.38) can be rewritten as

$$\mathbf{e}_k = \hat{\mathbf{e}}_k - \varepsilon_k \tag{2.44}$$

From Eq(2.44) and Eq(2.40), an iterative relationship for $\mathbf{e}_k$ along the batch index k can also be obtained as

$$\mathbf{e}_{k+1} = \mathbf{e}_k - \hat{\mathbf{G}}_s \Delta\overline{\mathbf{U}}_{k+1} - \tilde{\varepsilon}_{k+1} \tag{2.45}$$

Given the error transition model in the form of Eq(2.43) and Eq(2.45), the objective of ILC is to design a learning algorithm to manipulate the control policy so that the product qualities follow the specific desired reference trajectories. It is required that the learning algorithm has the following property (Lee et al., 2000)

$$\lim_{k\to\infty} \|\mathbf{e}_k\|_{\mathbf{Q}}^2 \to \min_{\mathbf{U}} \|\mathbf{e}\|_{\mathbf{Q}}^2 \tag{2.46}$$

By the certainty-equivalence principle (Lee and Lee, 1997), lets consider solving the following quadratic objective function based on the modified prediction errors upon the completion of the k^{th} batch run to update the input trajectory for the $(k+1)^{th}$ batch run

$$J_{k+1} = \min_{\Delta\overline{\mathbf{U}}_{k+1}} \frac{1}{2}[\tilde{\mathbf{e}}_{k+1}^T \mathbf{Q} \tilde{\mathbf{e}}_{k+1} + \Delta\overline{\mathbf{U}}_{k+1}^T \mathbf{R} \Delta\overline{\mathbf{U}}_{k+1}] \tag{2.47}$$

where **Q** and **R** are positive definitive matrices (Xiong and Zhang, 2003). Note that the objective function, Eq(2.47), has a penalty term on the input change $\Delta\overline{\mathbf{U}}_{k+1}$ between two adjacent batch runs, the algorithm has an integral action with respect to the batch index k (Lee et al., 2000). The weighting matrices **Q** and **R** should be selected carefully. The ratio of **Q** and **R** affect the optimal performance of the process and not the real values. The **Q** and **R** ratio influence the elimination of disturbances and ability to track back the desired trajectory (Gao et al., 2001). The weighting matrix **Q** is related to the final product quality variables while R is related to the control actions. A larger weight of **R** on the input change will lead to more conservative adjustments and slower convergence. Slow processes should use small **R** values while highly correlated output errors should be using larger **R** values (Campbell et al., 2002). There are also other variants of the objective function. For example, the weighting matrices **Q** and **R** may be set as **Q** = diag{Q(1), Q(2), …, Q(N)}, **R** = diag{R(0), R(1), …, R(N-1)}, where Q(i) and R(j) increase with respect to the time intervals t in proportion to its effect of the final product quality. For the sake of simplicity, **Q** and **R** are selected in this study as $\mathbf{Q}=\lambda_q\cdot\mathbf{I}_N$ and $\mathbf{R}=\lambda_r\cdot\mathbf{I}_N$.

By finding the partial derivative of the quadratic objective function Eq(2.47) with respect to the input change $\Delta\overline{\mathbf{U}}_{k+1}$ and through straightforward manipulation, the following ILC law can be obtained

$$\Delta\overline{\mathbf{U}}_{k+1} = \hat{\mathbf{K}}\mathbf{e}_k \tag{2.48}$$

where $\hat{\mathbf{K}}$ is defined as the learning rate

$$\hat{\mathbf{K}} = [\hat{\mathbf{G}}_s^T \mathbf{Q} \hat{\mathbf{G}}_s + \mathbf{R}]^{-1} \hat{\mathbf{G}}_s^T \mathbf{Q} \tag{2.49}$$

From Eq(2.42) and Eq(2.48), the ILC law for the control trajectory can be written as

$$\mathbf{U}_{k+1} = \mathbf{U}_k + \hat{\mathbf{K}}\mathbf{e}_k \tag{2.50}$$

2.9 Summary

Developing an efficient control and optimization method for industrial fermentation is deemed complex due to unavailability of accurate process models, non-linearity of the bioprocess and fed-batch operation, lack of accurate on-line sensors of important parameters due to the microorganism sensitivity, difficulty in predicting and controlling the internal environment of the living cell and the reactor and slow process response. It is arguable that data based empirical models are not as reliable as the mechanistic models due to the absence of insight details of the processes. However, in practice most batch processes are meant for small quantity manufacturing and the product recipe keeps changing. Therefore, time and cost consuming investment on detailed mechanistic model for every product process is not necessary. A reliable representation of a system based on the available data is sufficient. It is also essential that the control algorithm should be kept simple and widespread in application. Iterative learning control has been identified to be the most appropriate control method for fed-batch fermentation system. Batch to batch iterative learning control with updated process model is expected to improve product quality prediction for fed-batch fermentation process system.

CHAPTER 3: SIMULATION OF A FED-BATCH FERMENTATION PROCESS

3.1 Introduction

In the context of this study, the scope of fermentation is restricted to transformations of substances by microorganisms or cells in submersed cultures on an industrial scale to achieve one or more of the following goals: 1) degradation of complex substances into simple components, 2) synthesis of substances which may be accumulated in the microorganisms or excreted to the medium, 3) production of biomass from some nutrients. Usually the processes run in a kind of bioreactor to guarantee somewhat homogeneous conditions and to perform mass transfer of gaseous components creating the necessary turbulence for the reaction to take place. Fed-batch fermentation is a popular choice for amino acid, penicillin, cell mass and enzyme production. The input flow rate can be manipulated at predefined interval to maximise desired production (Hong, 1986).

In this study, a simulated industrial Baker's yeast fermentation process (Yuzgec et al., 2009) is used as a case study for the batch to batch ILC using updated linearised model. Since the mechanism of this fermentation process has been studied extensively, it can be securely used to simulate fermentation process to study the effectiveness of the proposed control methods. In most of the studies related fed-batch fermentation control, pilot/lab scale simulation work is used. In this work, simulation of the industrial scale fed-batch fermentation process is used to understand the impact of ILC for mass production. In developing the fermentation process model, both cell kinetic model and reactor dynamic model were considered and combined. In this chapter the reproduced simulation results were verified with the original work and then, appropriate simulation conditions were set.

The chapter is organised as follows. In Section 3.2, an overview of Baker's yeast fermentation system is briefly discussed. The biological pathways involved in the Baker's yeast cultivation that influenced the cell models are presented. A brief justification to use fed-batch reactor for this simulation and the expected outcome of the simulation is discussed. In Section 3.3, the cell kinetic and reactor dynamic mathematical models are presented and defined. Section 3.4 outlines the simulation conditions for Baker's yeast fed-batch fermentation that will be used for this study. The initial conditions are spelled out. Section 3.5 presents the simulation verification results. The simulation results were verified with the results in the reference paper to ensure accurate simulation coding. Finally, Section 3.6 summarises the chapter.

3.2 An overview of Baker's yeast fermentation

A concise background of the industrial baker's yeast fermentation is presented here. Baker's yeast is cultivated from the strain of *Saccharomyces cerevisiae*. The specialty of this strain is that it can ferment or respire depending on the culture condition. Assuming that the nitrogen and other necessary supplements and growth conditions are adequate, in the presence of oxygen and limited glucose, *Saccharomyces cerevisiaes* actively respire and produce baker's yeast biomass. This is referred as the oxidative pathway and the process is called glycolysis pathway. When the cells are actively growing, oxygen is fast used up by the cells. There is a possibility of oxygen deficiency to occur due to lagged oxygen transfer. At that point of anaerobic condition, glucose is fermented into ethanol and carbon dioxide. This is known as reductive pathway. When oxygen level increases again, ethanol will be converted into biomass. This is known as oxido-reductive pathway. This conversion takes place only when glucose flux is below maximum metabolic rate of yeast cells (George et al., 1998; Rieger et al., 1983; Sonnleitner and Kappeli, 1985).

The overall growth reaction is given below (Rieger et al., 1983; Sonnleitner and Kappeli, 1985):

$$C_6H_{12}O_6 + aN\mathit{X}\,[NH_3] + bO_2 \rightarrow yCH_{H\mathit{X}}O_{O\mathit{X}}N_{N\mathit{X}} + zC_2H_6O + cH_2O + dCO_2 \qquad (3.1)$$

Ethanol production can also happen in the presence of abundant of oxygen if there are high concentrations of glucose present (Daramola and Zamparaka, 2008). In surplus glucose feed, the biomass production capacity becomes saturated. Then, the remaining glucose is consumed by the ethanol producing mechanism in the microbes. In this case both biomass and ethanol will be produced simultaneously (Sonnleitner and Kappeli, 1985; George et al., 1998). The ethanol fermentation is not favoured in the industrial baker's yeast biomass production because it reduces the biomass yield on the glucose feedstock. Although the ethanol will eventually be converted to biomass, previous literature calculation revealed that the total biomass yield is reduced when carbon combustion takes place via ethanol formation (Verduyn et al., 1991; George et al., 1998; Rieger et al., 1983). The elemental composition of ethanol-grown biomass is also found to be different than the glucose-grown ones (Sonnleitner and Kappeli, 1985). However, the composition of glucose-grown biomass is not affected by the presence of ethanol. Therefore, the objective of baker's yeast cultivation is to maximize biomass production through oxidative pathway and eliminate or reduce ethanol formation (Berber et al., 1998; Yuzgec et al., 2009; Pertev et al., 1997; Karakuzu et al., 2006). Previous studies have revealed that biomass growth is closely dependant on glucose and oxygen feed rate (Bajpai and Reuss, 1980; Zhang, 2008a; Pertev et al., 1997). Provided that ample oxygen is supplied, the focus will be to manipulate feed profile to attain desired growth rate.

In industries, fed-batch reactors are used so that the glucose feeding rate can be controlled. Fed-batch is commonly used in a wide range of fermentation industry especially for manufacturing high value added products (Lee and Lee, 2003; Karakuzuku et al., 2006; Gosling, 2003). Fed-batch operation is the one with varying feed rate, fed at preset intervals according to the process phases and the yield is removed at the end of the cycle. Varying feeding rates at pre-specified intervals seem to enhance the productivity of microbial cells via metabolic control. The feed rate and feed intervals may be based on a pre-programmed trajectory or determined using a feedback controller. As to whether the desired results are achieved, it depends on the model used to generate the feeding profile (Daramola and Zamparaka, 2008; George et al., 1998).

In baker's yeast production, both the product quality and quantity are important measures for optimal production. The quantity is measured via biomass produced per litre of substrate. The quality of the yeast produced is measured in a number of ways such as colour, consistency, smell and shelf-life. The most important quality criteria is the gassing power, in other words, the ability of yeast to raise a dough by producing carbon dioxide (George et al.,1998). The on-line product quantity measurement can be measured via optical sensors (Salgado et al., 2001). There are almost no sensors or devices that can directly measure the online quality of fermentation yields. The intention of batch to batch iterative learning control is to predict and improve product quality, but since there is no lab data available, this work is carried out by improving product quantity via computer simulation.

3.3 Baker's yeast fed-batch fermentation process model

In developing the fermentation process model, both cell model and reactor model were considered and combined. The cell kinetic model was based on the well documented Monod Kinetics investigation by Sonnleitner and Kappeli (1985). The hypothesis assumes that excessive glucose and limited oxygen will result in ethanol fermentation. Karakuzu et al. (2006), added additional terms, $\left(1 - e^{-t/_{td}}\right)$, to the glucose uptake rate and $\frac{K_i}{K_i + C_e}$,to the oxidation capacity in the cell kinetic model. They took into consideration the possible co-consumption of oxygen for ethanol metabolism and the delay in the glucose uptake rate caused by change in metabolism during biomass production from ethanol. The improved version with additional terms mentioned above by Karakuzu et al. (2006) is used in this work and presented a below. The same process models were used by Yuzgec et al. (2009) as case study in his work. Therefore, model equations were adapted from Yuzgec et al. (2009).

3.3.1 Cell kinetic model

The cell kinetic model is given as follows.

Glucose uptake rate : $Q_s = Q_{s,max} \frac{C_s}{K_s + C_s} \left(1 - e^{-t/_{td}}\right)$ (3.2)

Oxidation capacity : $Q_{o,lim} = Q_{o,max} \frac{C_o}{K_o + C_o} \frac{K_i}{K_i + C_e}$ (3.3)

Specific growth rate limit : $Q_{s,lim} = \frac{\mu_{cr}}{Y_{x/s}^{ox}}$ (3.4)

Oxidative glucose metabolism : $Q_{s,ox} = \min \begin{pmatrix} Q_s \\ Q_{s,lim} \\ Q_{o,lim/Y_{o/s}} \end{pmatrix}$ (3.5)

Reductive glucose metabolism : $Q_{s,red} = Q_s - Q_{s,ox}$ (3.6)

Ethanol consumption rate : $Q_{e,up} = Q_{e,max} \left(\frac{C_e}{K_e + C_e}\right) \left(\frac{K_i}{K_i + C_e}\right)$ (3.7)

Oxidative ethanol metabolism : $Q_{e,ox} = \min\begin{pmatrix} Q_{e,up} \\ (Q_{o,lim} - Q_{s,ox}Y_{o/s})Y_{e/o} \end{pmatrix}$ (3.8)

Ethanol production rate : $Q_{e,pr} = Q_{s,red}Y_{e/s}$ (3.9)

Total specific growth rate : $\mu = Q_{s,ox}Y_{x/s}^{ox} + Q_{s,red}Y_{x/s}^{red} + Q_{e,ox}Y_{x/e}$ (3.10)

Carbon dioxide production rate : $Q_c = Q_{s,ox}Y_{c/s}^{ox} + Q_{s,red}Y_{c/s}^{red} + Q_{e,ox}Y_{c/e}$ (3.11)

Oxygen consumption rate : $Q_o = Q_{s,ox}Y_{o/s} + Q_{e,ox}Y_{o/e}$ (3.12)

Respiratory Quotient : $RQ = \frac{Q_c}{Q_o}$ (3.13)

3.3.2 Dynamic reactor model

The dynamic reactor model is based on material balances and represented by the following ordinary differential equations.

$$\frac{dC_s}{dt} = \frac{F}{V}(S_o - C_s) - \left(\frac{\mu}{Y_{x/s}^{ox}} + \frac{Q_{e,pr}}{Y_{e/s}} + Q_m\right)C_x \quad (3.14)$$

$$\frac{dC_o}{dt} = -Q_oC_x + k_La_o(C_o^* - C_o) - \frac{F}{V}C_o \quad (3.15)$$

$$\frac{dC_e}{dt} = (Q_{e,pr} - Q_{e,ox})C_x - \frac{F}{V}C_e \quad (3.16)$$

$$\frac{dC_x}{dt} = \mu C_x - \frac{F}{V}C_x \quad (3.17)$$

$$\frac{dV}{dt} = F \quad (3.18)$$

Assumptions made for the above mentioned process model are:

1) The process only involves liquid phase. The gas phase and microorganism were not considered.

2) The reactor content were homogenous in axial and radial direction

3) Energy balance is expected to be under control with effective temperature control. The optimal temperature for yeast to ferment sugar is 32°C. In warmer temperature (45°C) the yeast cells will die. Therefore, it is of primary concern to have an effective temperature control (Daramola and Zamparaka, 2008).

The model parameters are listed in Table 3.1.

Table 3.1: Parameters used in cell kinetics and reactor dynamic model (Yuzgec et al., 2009)

Parameter	Unit	Value	Parameter	Unit	Value
K_e	gL^{-1}	0.1	$Y_{c/e}$	gg^{-1}	0.645
K_o	gL^{-1}	9.6 x 10^{-5}	$Q_{e,max}$	$gg^{-1}\ h^{-1}$	0.238
K_i	gL^{-1}	3.5	$Q_{o,max}$	$gg^{-1}\ h^{-1}$	0.255
K_s	gL^{-1}	0.612	$Q_{s,max}$	$gg^{-1}\ h^{-1}$	2.943
$Y_{x/s}^{ox}$	gg^{-1}	0.585	Q_m	$g\ g^{-1}\ h^{-1}$	0.03
$Y_{x/s}^{red}$	gg^{-1}	0.05	μ_{cr}	h^{-1}	0.21
$Y_{o/s}$	gg^{-1}	0.3857	S_o	$g\ L^{-1}$	325
$Y_{e/o}$	gg^{-1}	1.1236	C_o^*	$g\ L^{-1}$	0.006
$Y_{e/s}$	gg^{-1}	0.4859	A_R	m^2	12.56
$Y_{x/e}$	gg^{-1}	0.7187	t_d	h	2
$Y_{c/s}^{ox}$	gg^{-1}	0.5744	klao	h^{-1}	700
$Y_{c/s}^{red}$	gg^{-1}	0.462			

3.4 Baker's yeast fed-batch fermentation process simulation

Both the cell kinetic model and dynamic reactor model was coded using Matlab version 7.1. The differential equations were solved using ODE45 function in MATLAB. As recommended in the MATLAB manual, ODE45 is the best function to apply as a first try for most problems. In the event that ODE45 encounters problems, other ODE functions in MATLAB can be attempted. In this study, ODE45 was able to solve the equations satisfactorily.

The simulated process operation conditions were set as follow:

Initial reactor volume, V_o: 0.05 m^3

Fermentor volume, V_{fer} : 100 m^3

Total fermentation time, t_f : 16.5 h

Data record interval, h: 0.001 h (3.6s)

Initial substrate concentration, Cs_o: 7.0 gL^{-1}

Initial biomass concentration, Cx_o: 15.0 gL^{-1}

Initial oxygen concentration, Co_o: 0.006 gL^{-1}

Initial ethanol concentration, Ce_o: 0.0 gL^{-1}

In this study, the batch duration is divided into 10 equal stages and the feed rate is kept constant within each stage. The initial feed was estimated from that reported by Yuzgec et al. (2009) as shown in Figure 3.1.

The feed rate in Figure 3.1 is based on the equation below:

$$F = F_o e^{\alpha t} \quad (3.19)$$

Figure 3.1 shows that when t=0, F_o=500 L/h. The constant, α, is fixed at 0.052.

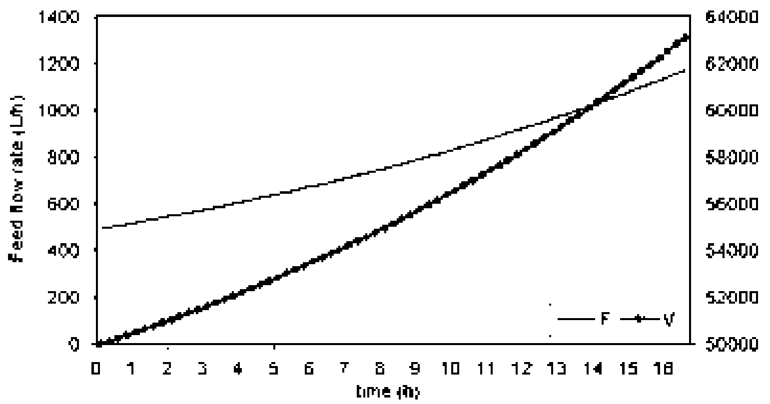


Figure 3.1: Profiles of feed flow rate and volume for all cases (Yuzgec et al., 2009)

The initial feed rate derived and used in this study is

Feed rate = [544.8; 593.6; 646.8; 704.; 767.9; 836.7; 911.6; 993.3; 1082.3; 1179.2].

The corresponding biomass concentration at the end of the batch was obtained from simulation and is 46.1827 g/L.

In this study, fed-batch reactor is used to feed molasses at an incremental rate at specified time interval to suppress metabolite repression of microbes due to excessive sugar concentration (Karakuzuku et al., 2006). The feed rate was also controlled in such a way that the average biomass growth rate is always close to the critical growth rate.

This is because oxygen deficiency may occur due to increased yeast growth which will then favour ethanol production. Therefore, growth rate should be limited to suppress the ethanol fermentative metabolism. In addition to that, the feed rate generated should also guarantee that the total biomass volume at the end of the batch run should be lower than the reactor volume. In a fed-batch, feedstock is fed at pre-specified intervals without any withdrawals. Therefore, care must be taken to avoid overflow (Daramola and Zamparaka, 2008; Yuzgec et al., 2009). In this study the reactor volumes is 100 $m^3$and the final biomass solution is kept well below it.

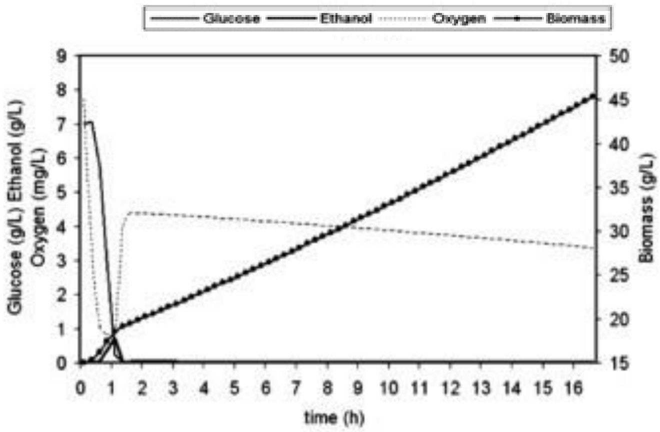
The initial conditions in case 1 of (Yuzgec et al., 2009), which is biomass concentration at 15g/L and glucose concentration at 7g/L, is selected as the initial condition in this work. The choice was made because the case using these concentrations produces highest biomass concentration amongst the four tested cases as can be found in (Yuzgec et al., 2009). It is perceived to be an optimal ratio of biomass and glucose for initial condition to maximise biomass production and minimise ethanol production (Yuzgec et al., 2009).

The initial control profile for this work is obtained via approximation of feed profile in Figure 3.1. The initial control profile is simulated to obtain the first batch output data. This initial control profile is then used as the base feedrate to generate more data to be used as historical batches for ILC application. The development of the ILC algorithm parameters using the data from this simulation results is discussed in chapter 4.

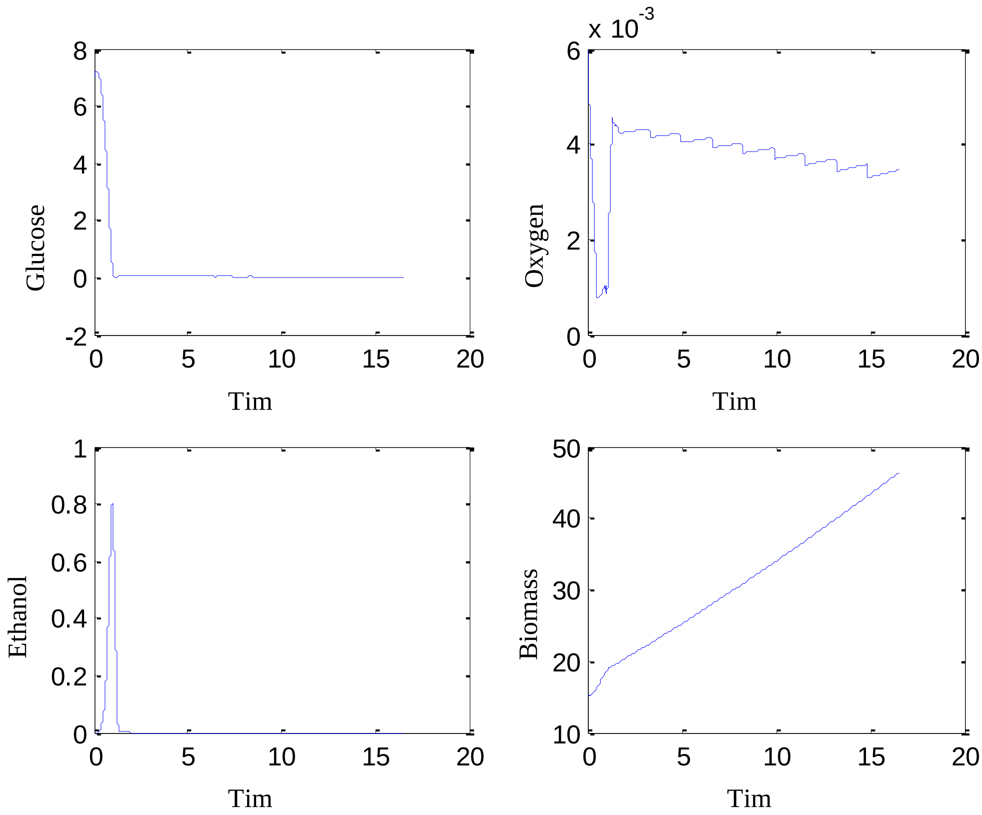
3.5 Fermentation simulation validation with literature

The simulation programme was run to reproduce the results obtained by Yuzgec et al. (2009). There were four cases with different initial values studied by Yuzgec et al. (2009). The results were replicated for all the four cases to verify the simulation. The yeast fermentation simulation has successfully reproduced the results in Yuzgec et al. (2009) for all the four cases. The result for case one with the initial concentrations as stated in Section 3.3 is presented here. Figure 3.2(a) is the result presented by Yuzgec et al. (2009) and Figure 3.2(b) is the results produced in this work. There are no numerical values presented in the reference paper. Hence, the results in Figure 3.2(b) are compared approximately at the starting point, end point and performance pattern to match the graphical results in Figure 3.2(a). The final concentrations and changes in the concentration during the run seem to match with

Figure 3.2(a) reliably. The slight difference in the oxygen plot is due to the approximation of the continuous feeding profile in Yuzgec et al. (2009) by a piecewise constant feeding profile. The process simulation also produced the same results as those given by Yuzgec et al. (2009) for all the other three cases.



(a) Extracted from (Yuzgec et al., 2009)



b) Results produced in this work

Figure 3.2: Glucose, Oxygen, Ethanol and Biomass concentration produced in the process simulation

3.6 Summary

In summary, the details of the Baker's yeast fed-batch fermentation dynamics and simulation conditions have been clearly defined. The parameter values for the system model have been identified. The cell kinetics and reactor dynamics were successfully coded in Matlab to reproduce the results as in the reference paper. The reproducibility of the results using the mathematical coding proves that the simulation is reliable and can be safely used as case study throughout this work.

CHAPTER 4: BATCH TO BATCH ITERATIVE LEARNING CONTROL USING UPDATED LINEARISED MODEL

4.1 Introduction

Fed-batch fermentation control and optimization simply involves controlling the feed rate to achieve desired product quality and/or quantity. There is high demand for optimal control policy design these days especially for production with stringent product quality requirement (Huang et al., 2010). Although the task is to control only one parameter, which is the feed rate and this sounds like a simple control problem, identifying optimal feed rate has been a reasonably difficult task due to its non-linear system dynamic (Xiong et al., 2008). The optimization can be made feasible by identifying a reliable dynamic process model that can assist long range model prediction (Huang et al., 2010).

In recent years, empirical models developed from process input and output data alone is gaining attention due to its simplicity in development. The empirical models can be used to develop a simple model that closely represents the system to be controlled by finding a relationship between the input data and output data. One of the ways to develop empirical models is by using regression principles. Simple linear regression such as multiple linear regressions (MLR), principle component regression (PCR) and partial least squares (PLS) can be used to identify an appropriate system dynamic model. The regression model based ILC is gradually becoming an area of interest due to its potential. The ILC with PLS model have been reported in a few papers but work on ILC with PCR and MLR is rare. The PLS is designed to arrest high colinearity issue. Though PLS is a more advanced regression method, it would be interesting and academically beneficial to investigate the performance of PCR and MLR model based ILC on fed-batch fermentation process. It would be interesting to compare the performance of three models on fed batch fermentation process.

In a basic ILC algorithm, there are only three elements used to construct the current batch control policy. They are the previous batch control policy, previous batch output deviation from the desired trajectory and iterative learning rate. The first two parameters can be directly obtained from the previous batch data but the learning rate has to be developed from the previous batches information. The information contained in the learning rate plays an important role on the performance of the current batch. The identified process models come into use in constructing the iterative learning law. The learning rate carries information on the system model and magnitude of control action to develop an optimized control policy. The optimal control policy is expected to render a stable and converging performance. In order to obtain an optimal control policy, optimal ILC technique is used in this work. Optimal ILC is one of the popular methods used in designing iterative learning law. In optimal ILC, a quadratic objective function is solved to construct the learning rate. The quadratic objective function optimizes the changes in control policy in accordance to the system error without affecting the system stability. The system stability is maintained by carefully selecting the weighting parameter, Q and R used in the quadratic objective function (Eq.2.47). The ratio of Q and R dominates the magnitude of changes in the control policy for the controlled batch. The optimal ratio ensures that there is no sudden hikes or drops in the control policy that will affect the stability of the batch performance.

The major twist here is that ILC is a linear controller and the regression models are linear models too. Fed-batch reactors are highly non-linear and operate in transient states. Due to absence of steady states, standard linear optimal control techniques that could be successfully used for continuous process cannot be used directly for fed-batch fermentation system. The repetitive nature of fed-batch fermentation allows information from previous batch to be used to improve current batch performance iteratively. The input and output data a system to be controlled can be linearised around a selected nominal trajectory. That way the ILC method using regression models can be applied. In addition to that, process non-linearities could be eliminated through linearisation.

In order to improve the batch performance iteratively, batch to batch ILC using updated linearised models is suggested in this chapter. Batch to batch optimization serves as a solution for systems like fed-batch fermentation where there is no steady state. The previous batch data can be used as the nominal trajectory to improve the current batch performance. By updating the batch-wise updated linearised model batch after batch, more recent system conditions can be captured hence optimal control policy for the current batch can be identified. The proposed method is expected to address model-plant mismatch and fed-batch fermentation nonlinearities issues. The ILC techniques should track the desired trajectory asymptotically within ten batches. A steady and converging performance is desired for this control method. The batch to batch updated linearised model ILC method proposed in this work has been applied to a computer simulated fed-batch fermentation process. The details of the simulation can be found in Chapter 3.

This chapter consists of results for two different sets of historical batch data. The first part of the chapter from Sections 4.2 to 4.6 is a preliminary run details and results with first selection of simulation testing conditions. The second part of the chapter, from Sections 4.7 to 4.12, is the results for modification of the first part details. The second part conditions will be used for the rest of the work in this study. The chapter is organised as follows. Section 4.2 describes the development of linearised models using the first set of historical batches. In section 4.3, the Q and R values for the first set of historical batches is identified for all the three linearised models. Section 4.4 presents the results of batch to batch ILC using updated linearised model using MLR, PLS and PCR regression methods. The convergence of biomass concentration for updated and non-updated models was investigated. In Section 4.5, the feed rate profile of the batch to batch updated models based ILC is presented. Section 4.6 summarises the performance of the proposed method for the initial set of simulation testing conditions. Section 4.7 outlines the necessary amendment needed on the simulation testing conditions to better suit the requirements of this study. Section 4.8 describes the development of new set of historical batches and the respective linearised model. In Section 4.9, the Q and R values were selected again. Section 4.10 presents the results for batch to batch ILC using updated and non-updated models and nominal trajectories. The performances of the regression models; MLR, PCR and PLS for the new set of data were investigated for three different ILC conditions. The conditions are outlined as Case 1, 2 and 3 in the section. Section 4.11 presents the feed rate profile for updated model and nominal trajectory case study. Finally, section 4.12 summarises the performances of the proposed control method for the new set of historical batches.

4.2 Development of linearised models

A set of historical batch data containing input and output information, is needed to generate an ILC model. Since there are 10 piecewise constant inputs for every batch, there has to be more than 10 historical batches in order to identify the model. More historical batch data will give more information about the system. Therefore, initially, 40 historical batches were generated from the initial feed rate presented in chapter 3, section 3.4. The initial feed rate is used as the nominal control profile to develop the required historical batches by adding random variations to it. The random variations follow normal distribution with zero mean and standard deviation of 50. The linearised model is developed based on the equations presented in Chapter 2, Section 2.8. There are slight changes in the equations in Section 2.8.1.

In this study, the idea is to generate a batch to batch control. Therefore, intra batch condition is not considered. The focus will be on the inter batch performance. The historical batch data is in batch dimension rather than time dimension. Therefore, Equations 2.12-2.14 are reproduced as below:

$$\mathbf{Y} = [y\ (i)]^T = [y(1), y(2), \ldots, y\ (M)]^T \qquad (4.1)$$

$$\mathbf{U} = [u(i)]^T = [u(1), u(2), \ldots, u(M)]^T \qquad (4.2)$$

with i=1,2…M

where

M is the number of historical batches used in the study.

Y is the end-batch biomass concentration produced after every batch run. There is only one yield data for each batch run. The end-batch biomass concentration is the controlled variable.

U is the glucose feed rate profile for all the historical batches. The feed rate is the manipulated variable.

$u(\mathrm{i}) = [u_{i(1)},u_{i(2)},u_{i(3)},u_{i(4)},u_{i(5)},u_{i(6)},u_{i(7)},u_{i(8)},u_{i(9)},u_{i(10)}]^T$ represents 10 feed rates for each batch feed profile. Each of the 10 feed rates is introduced at intervals of 1.65hours for 16.5 hours.

Since there is only one yield value for every batch, the desired product trajectory is also one value. In this study, it is set that

$$\mathbf{Y}_d = 74 \text{ g/L} \qquad (4.3)$$

For M number of batches, the non-linear function in Eq.2.16 will remain as

$$\mathbf{Y} = \mathbf{F}(\mathbf{U}) + \mathbf{v} \qquad (4.4)$$

where **F(·)** represents the non-linear static functions between **U** and **Y** and $\mathbf{v}=[v(1), v(2),\ldots,v(M)]^T$ is a vector of measurement noises for M batches.

Once the 40 historical batches were obtained, the nominal control profile, U_s and nominal yield, Y_s were selected, either by calculating the mean of the 40 batches data or by using the 40th batch data. Then the perturbation variables data were generated by subtracting U_s and Y_s from the original historical batches of input and output data respectively. The non-linear batch process model in Eq(4.4) is linearised as per Eq(2.17). Multiple linear regressions (MLR), partial least square (PLS) or principal component regression (PCR) has been used to estimate the linearised model parameters. The details on how to estimate each of the regression models can be found in Chapter 2, Section 2.7. PLS and PCR functions in Matlab 7.1 were used for this purpose. The developed model is of the following form:

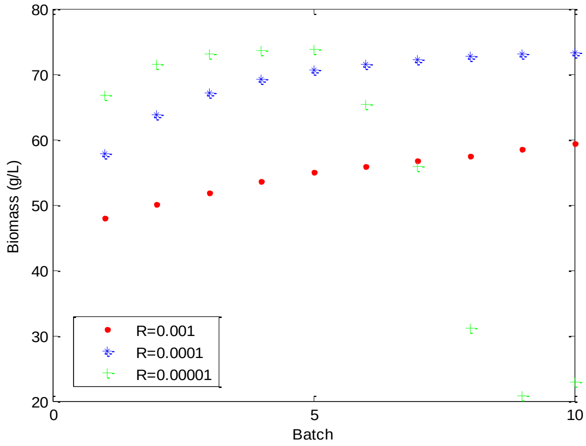
$$Y = \theta_1 u_1 + \theta_2 u_2 + \ldots + \theta_{10} u_{10} \tag{4.5}$$

where Y is the perturbation variable for end batch biomass concentration and u_1 to u_{10} are the perturbation variables for the substrate feed rates at 10 intervals during a batch. The developed linearised models were used in calculating updated control policies for batch to batch ILC method as per equations in Section 2.8.4.

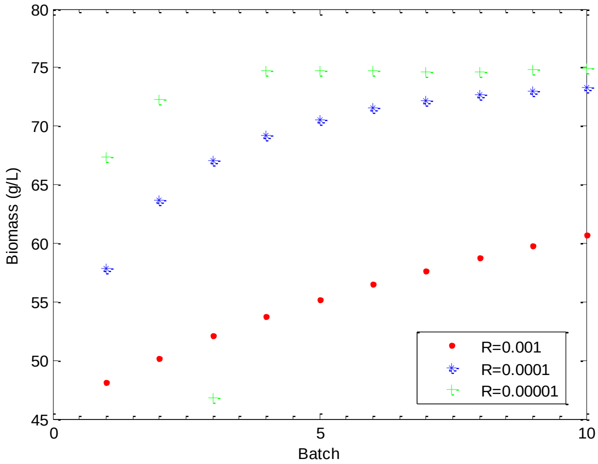
4.3 Selection of weighting matrices Q and R

Weighting matrices, **Q** and **R**, are used to balance between convergence speed and robustness respectively. The ratio of **Q** and **R** determines the relative weighting between the two terms in the objective function as in Eq(2.47). As observed in Eq(2.47), the **Q** value affects the tracking error reduction while **R** value limits the control changes. Bigger ratio tends to give faster tracking, but could also lead to oscillation. In this case study, $Q=\lambda_q$ is a scalar, **R** is considered as a diagonal matrix $\mathbf{R}=\lambda_r \cdot \mathbf{I}_{10}$, $\lambda_q \geq 0$ and $\lambda_r > 0$ are real scalars, and $\mathbf{I}_{10}$ is a 10×10 identity matrix. In this study, Q was fixed at 1 and the **R** values were varied to find a value that gives fast convergence without compromising the stability.

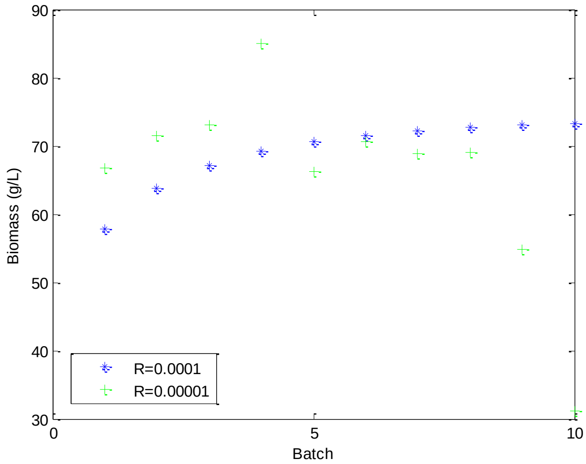
The MLR model was first used to determine the choices of **R** values which affect the ILC performance. The **R** values tested were 0.1, 0.01, 0.001, 0.0001 and 0.00001. The biomass yield is expected to increase from 45.66g/L (40^{th} historical batch run yield) to 74g/L which is the desired yield. The **R** value was determined by the trajectory tracking capability of ILC. The **R** values of 0.1 and 0.01 generated very small increments on the input predictions which resulted in very slow convergence of output. The values of 0.001, 0.0001 and 0.00001 exhibited better convergence within ten batch runs than 0.1 and 0.01. Therefore, these three values were tested for PLS and PCR method as well to determine the **R** value for each of the regression method. The best value will be the one that exhibits fastest and stable convergence towards desired trajectory.



a) ILC with MLR model



b) ILC with PLS model



c) ILC with PCR model

Figure 4.1: Comparison of different R values

Figure 4.1 exhibits the effect of three different **R** values, 0.001, 0.0001 and 0.00001, on ILC performance with the three regression models. The responses are shown for 10 tested batch runs. Figure 4.1(a) displays the outcome for batch to batch control based on the MLR model. The output with **R**=0.001**I** shows steady increments towards the desired output, Y_d=74g/L. However, the response is very slow. Therefore, it is not favoured in this work.

Besides stability, convergence speed is also important in selecting **R** value. The response with **R**=0.0001**I** shows significant leap at the first four batches and kept increasing steadily towards desired value till the 10th batch. This **R** value is preferred because it converges to the desired point steadily within only ten batches. As for the response with **R**=0.00001**I**, the output accelerated towards desired value till the 4th batch and then decelerated to 20g/L biomass yield in the following batches. This occurrence may be attributed to the high prediction of feed rate with **R**=0.00001**I** which resulted in excessive glucose in the reactor. The glucose surplus causes detrimental effect to the microbial growth and affects the biomass concentration.

In Figure 4.1 (b) the response with **R**=0.001**I** increases steadily but very slowly towards the desired output value. The response with **R**=0.00001**I** reached the desired value within 2 batches but the controlled variable is fluctuating. It is not suitable due to instability. The response with **R**=0.0001**I**, leaped significantly for the first 4 batches. Following that, the response increased steadily towards the desired value. The 10th batch achieved 74.3g/l biomass concentration. It is clear that the response with **R**=0.0001**I** displays stable and desirable results. Therefore, **R**=0.0001**I** is selected for ILC with the PLS model.

Since **R** value of 0.001 exhibits very slow response for ILC with both MLR and PLS models, it was omitted in the testing for ILC with PCR model. Only 0.0001 and 0.00001 were used and the result is presented in Figure 4.1 (c). The response using

R=0.0001 in Figure 4.1(c) exhibits fast and stable convergence to the desired value. The 10th batch produced 73.3g/L biomass and the desired value is 74g/L. The response with **R**=0.00001**I**, shows increasing output from the 1st to 3rd batch and then fluctuates till the 10th batch. In the PCR model too, **R**=0.0001**I** is preferred.

In summary, Figure 4.1 reveals that **R** value of 0.0001 exhibits steadfast converging responses towards desired trajectory within 10 batches for all the three regression methods. Therefore, the weighting matrices Q and **R** are fixed at 1 and 0.0001 respectively for the simulation study.

4.4 ILC performance on the simulated fed-batch fermentation process

The results of batch to batch ILC based on MLR, PLS and PCR models with and without disturbance are presented in this section. A total of 30 test batches were run for each linear regression model and each case study. The first 20 batches were simulated to test convergence and stability at a condition without any disturbances. At the 21st batch, a parametric disturbance was introduced. The initial substrate concentration, S_o, which was 325g/l under normal condition, was changed to 305g/l. It was then changed back to 325g/L for the 22nd to 30th batch. The disturbance was introduced to study the ability of the control method to handle the disturbance effect and track back to the desired trajectory. The disturbance was a non-repetitive one.

Figures 4.2, 4.3 and 4.4 show the simulation results for three different case studies used to investigate the performance of batch to batch ILC method. The 40th historical batch is referred as 0th batch in these plots. It is used as the reference batch. The three cases which were investigated are referred to as Case 1, Case 2 and Case 3. Each case condition is explained below:

<u>Case 1: Constant Gs, Ys and Us</u>

The model parameter, Gs was generated from 40 historical batches. Us in the mean feed rate and Ys is the mean biomass yield of the 40 historical batches. The three variables were kept constant for all future batches.

<u>Case 2: Updating Gs, Ys and Us</u>

In this case, for the first test batch run, the Gs value was calculated from the 40 historical batches. The Us and Ys are the 40th batch feed rate and yield respectively. After the first simulation, the feed rate and its corresponding yield were added into the 40 historical batches and a new Gs was generated from the total of 41 batches. The Us and Ys for the new batch are the 41st batch feed rate and yield respectively. The cycle repeats for the next batch. The Gs, Ys and Us was updated after every batch run.

<u>Case 3: Updating Gs (with constant Ys and Us)</u>

As for Case 3, the Gs value was updated after every batch as mentioned in Case 2 descriptions. The Ys and Us value was the mean of 40 historical batch and remains the same for all the future batch runs.

4.4.1 ILC using multiple linear regression models

Case 1 has been used as a reference in this study. It exhibits responses when ILC is used with no batch to batch model and nominal batch updating. As observed in Figure 4.2 below, the output of Case 1 increased steadily for the first 10 batches and almost attained the desired value. However, the responses were not maintained for the following batches. After the 10th batch, the biomass yield deteriorated. This is the effect of glucose surplus. Berber et al. (1998) noted that the improvement between every batch is very large in the beginning and then gradually slows down as it reaches desired result. For large improvements, more glucose feed is needed. The feed rate should be increased at a higher proportion from one batch to another. When the biomass increment rate reduces as it reaches and maintains asymptotical to the desired trajectory, the feed rates increment should also be adjusted accordingly to evade glucose surplus which has detrimental effect on biomass growth. The feed rates may have to be maintained at an optimal profile with no further increments when necessary. In Case 1, there is no batch to batch model updating after every batch run. Consequently, the feed rates calculated may not be optimal when the process operating condition differs from that in the nominal reference batch. In the first ten batches, the constant high dosage of glucose was needed to increase the yield rate. In the following ten batches, as the growth rates slowed down but the substrate increment rate was constant, the yeast cells becomes suffocated due to substrate surplus.

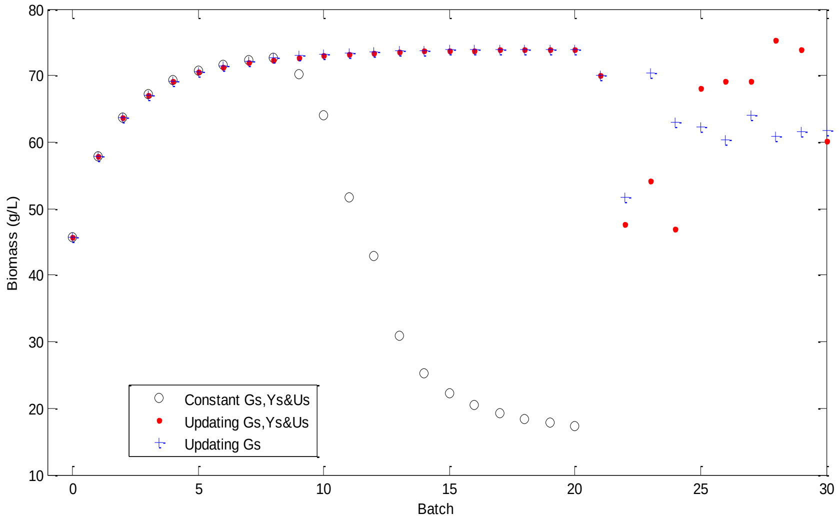


Figure 4.2: Batch to batch ILC using MLR model

Cases 2 and 3 were studied to understand the effect of updating model parameter, Gs, and nominal trajectories, Ys and Us. In Figure 4.2, the responses for Cases 2 and 3 have increased asymptotically to the desired value, 74g/L, at similar rate for the first 20 batches which was conducted to analyse convergence speed and stability. In the first ten runs, the yields have reached close to the desired value, with the 10th batch at 73g/l. In the following 10 batches, the simulation outputs steadfastly, increased asymptotically towards the desired product value with the 20th batch is about 73.95g/l. This result proves that iterative leaning control with batch wise updated models is efficient in achieving and maintaining optimal results under normal operating conditions with no disturbances. There is no significant difference in responses for updated and constant nominal variables. When disturbance was introduced at the 21st batch, the output in both Cases 2 and 3 became unstable and exhibited fluctuating pattern. Case 3 with constant nominal reference trajectories, showed decreasing response pattern and was not competent to attain the desired trajectory. As for Case 2, there is some increasing response towards desired trajectory during batches 25 to 29. It can be deduced that, in the presence of disturbance, updated nominal variables and model parameter exert a positive impact in tracking the reference trajectory in comparison to the constant nominal reference trajectories and updated model parameters. However, iterative learning control with batch wise updated models using MLR did not exhibit desired trajectory tracking characteristic. This can be attributed to the fact that MLR is not able to handle the colinearity among the control actions at different stages of a batch. The MLR model is inappropriate for correlated control actions.

4.4.2 ILC using partial least squares model

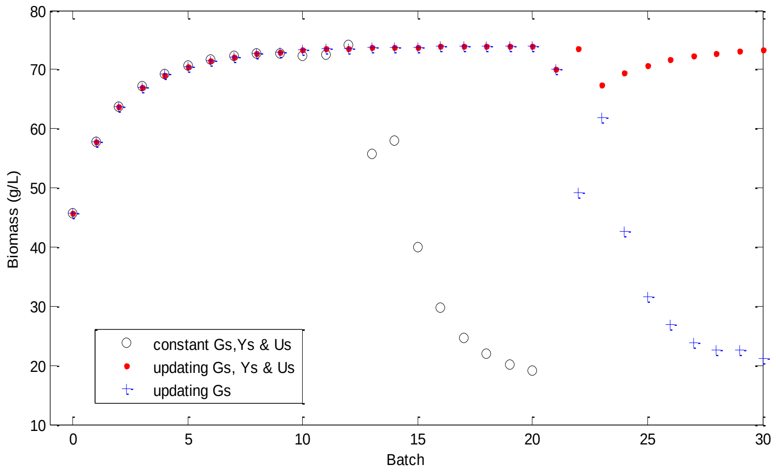


Figure 4.3: Batch to batch ILC using PLS model

Figure 4.3 exhibits the biomass produced at three tested cases for batch to batch control based on the PLS models. The responses in Case 1 exhibit positive increment till the 9th batch. The 10th batch onwards the response fluctuated and eventually decreased to below 20g/L. This is expected to happen since there is no batch to batch model updating to update the control policy changes necessary to cater to asymptotic performance. Constant increment rate on the control policy causes substrate surplus and adversely affects the end-batch biomass concentration. Both Cases 2 and 3 in Figure 4.3 demonstrated steady increase towards desired value at similar rate for the first 20 batches. When disturbance was introduced at batch 21, there was a slight decrease in the biomass value for both Cases 2 and 3 in comparison to batch 20. After the 21st batch, the responses of Case 2 and 3 changed. The response for Case 2 fluctuated for few batches before increasing towards the desired value gradually. The output value at the 30th batch is 73.3g/L. In contrast, the response for Case 3, declined appreciably to below 20g/L. These results indicate that the control with batch wise updated model and nominal reference trajectories is able to handle disturbances and track the reference trajectory steadily whereas the only updating model method is insufficient to do so. It is observed that PLS method can be used efficiently to develop model parameters for batch to batch model updated ILC method.

4.4.3 ILC using principal component regression model

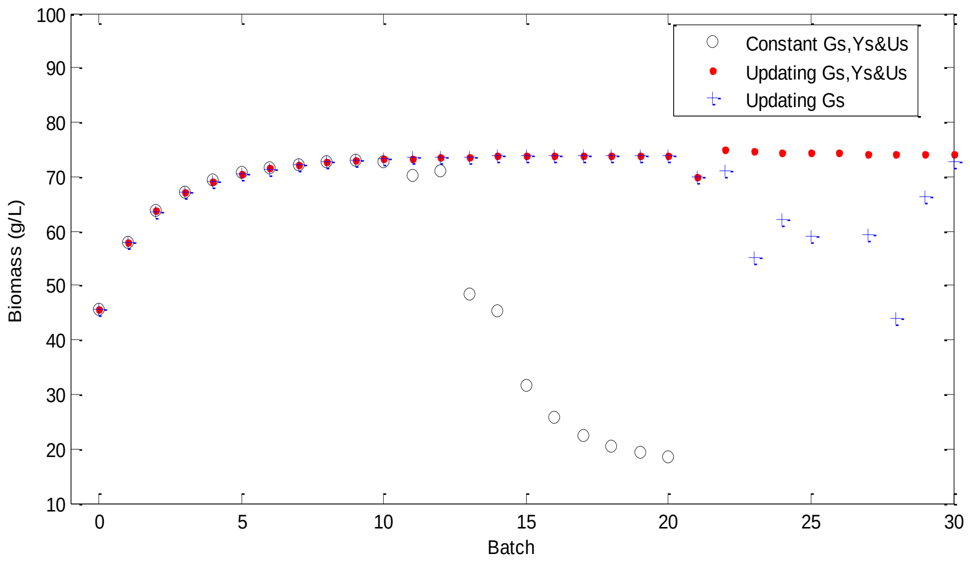


Figure 4.4: Batch to batch ILC using PCR model

In Figure 4.4, the response for all the three cases increased towards the desired value in the first ten batch runs. As expected, the biomass concentrations in Case 1 declined continuously for the following ten batches to below 20g/L. Meanwhile, the responses for Cases 2 and 3 increased asymptotically to the desired value in the next ten batches. When disturbance was introduced at batch 21, the outputs for Cases 2 and 3 reduced to about 70g/L. Following that, the response for Case 2 rose above the desired value to 74.85g/L and then gradually decreased asymptotical to the desired output trajectory. The control performance appears to be very satisfactory because it did not show large sway from the previous batches after the disturbance and it portrayed a stable trajectory. This is the outcome of control aimed to be achieved for the batch to batch process control and optimization. As for Case 3, after disturbance was introduced, the response generally fluctuates between 40g/L to 100g/L. The control with updating model parameter, Gs, alone becomes unstable when disturbance is introduced.

4.4.4 Comparison of Case 2 performance for ILC using MLR, PLS and PCR models

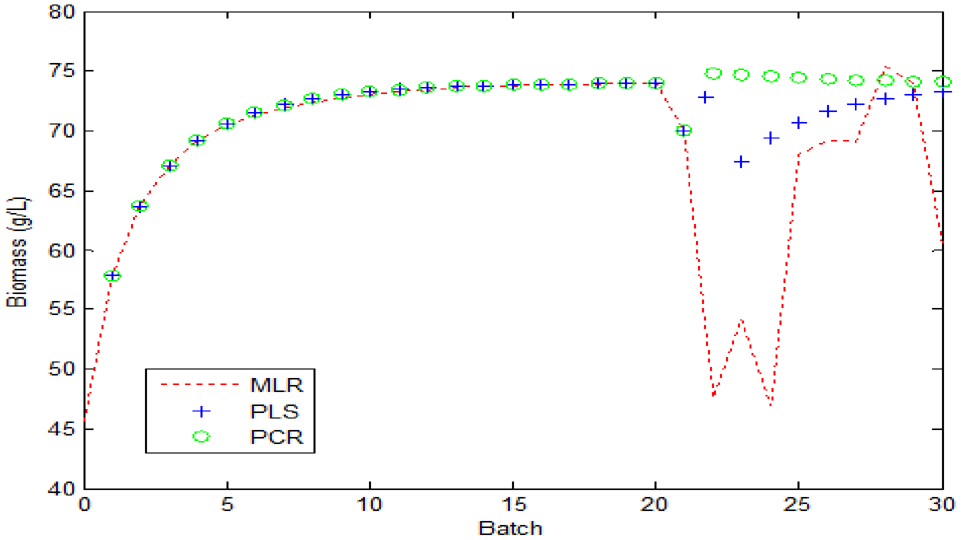


Figure 4.5: Comparison of the control strategies based on MLR, PLS and PCR models for Case 2 (updating Gs, Ys and Us)

With reference to Figures 4.2, 4.3 and 4.4, it is evident that batch to batch updating of Gs, Us and Ys (Case 2) leads to the best control performance in the presence of disturbance. The responses in Case 2 for ILC with the three models are displayed in Figure 4.5 for comparison purpose. As seen in Figure 4.5, the PCR model seem to exhibit very satisfying results. The PLS came second due to slight fluctuation after the disturbance was introduced. The MLR response is unstable and appeared not feasible to be used in batch to batch control with the presence of disturbances. Both the PCR and PLS method with updating model, nominal feed and nominal output can be used in the batch to batch process control and optimization with the presence of parametric disturbances. The difference in the results for PCR and PLS is due to the difference in the method used to develop process model as discussed in Chapter 2. The response pattern of Case 2 is further explained by the change in the feed rate profiles in the next section.

4.5 Feed rate profiles for batch to batch ILC with updated linearised models

The feed rate profile plays an important role in the process control and optimization to produce optimal output. In this section, the change in the feed rate trajectory for MLR, PLS and PCR method in the case of updating model, nominal output and nominal feed rate, Case 2, is displayed and analysed.

4.5.1 Control profile under ILC with multiple linear regression model

Figure 4.6 (a) shows that, in the first 40 historical batch run, the feed rate profile follow a pre-set incremental exponential profile. With the introduction of ILC with linearised model from MLR, the feed rate profile has been changed to a significant pattern to suit the process need. Furthermore, the feed rates have been increasing gradually to attain optimal feed rate profile to achieve desirable biomass concentration.

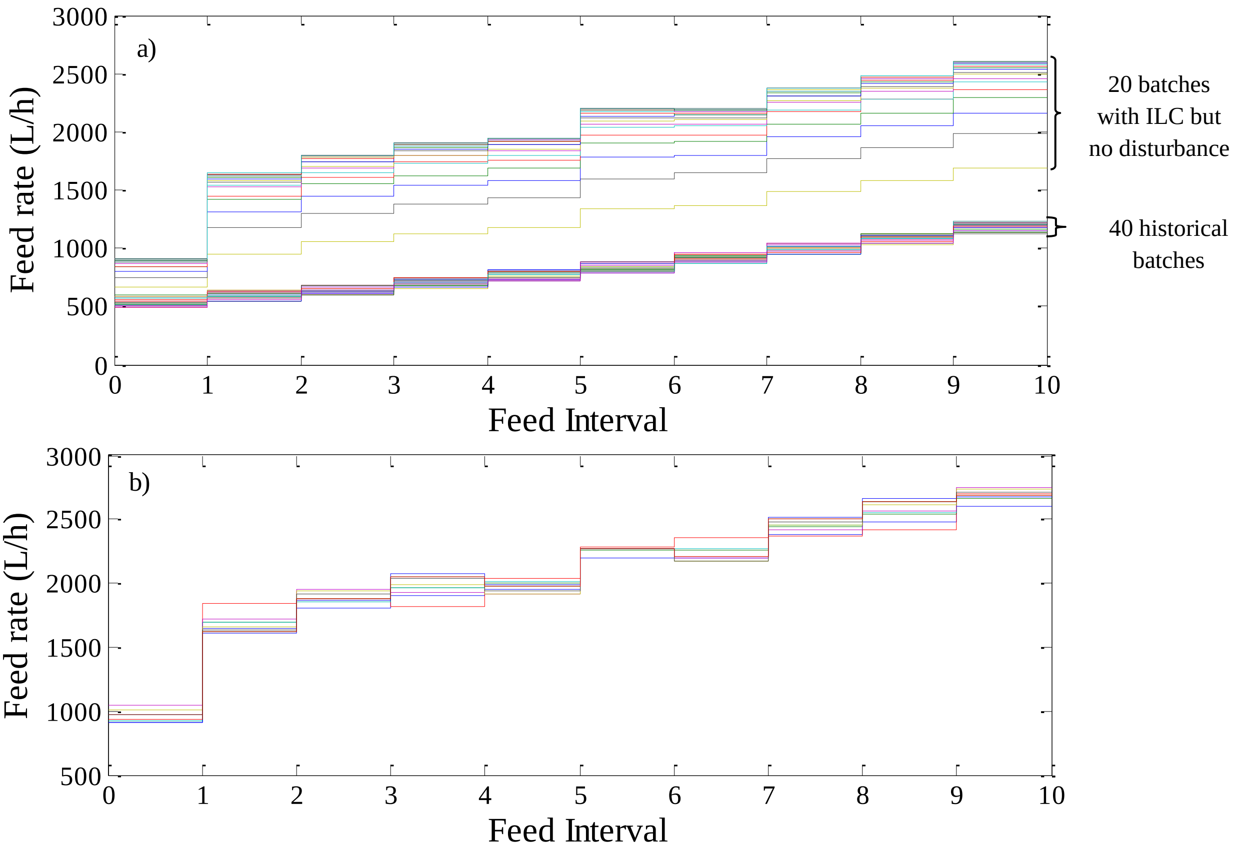


Figure 4.6: Feed rate profile for batch to batch ILC using MLR model in Case 2

Figure 4.6 (b) exhibits feed rate changes when disturbance was introduced. In general, the improved profile as in Figure 4.6 (a) remains. A closer look at it reveals that the feed rate profile have been disturbed and shows irregular pattern. The feed rate adjustment is not stable. This result further proves the fact that MLR method is not compatible to be used in the batch to batch bioprocess control with the presence of disturbances.

4.5.2 Control profile under ILC with partial least squares model

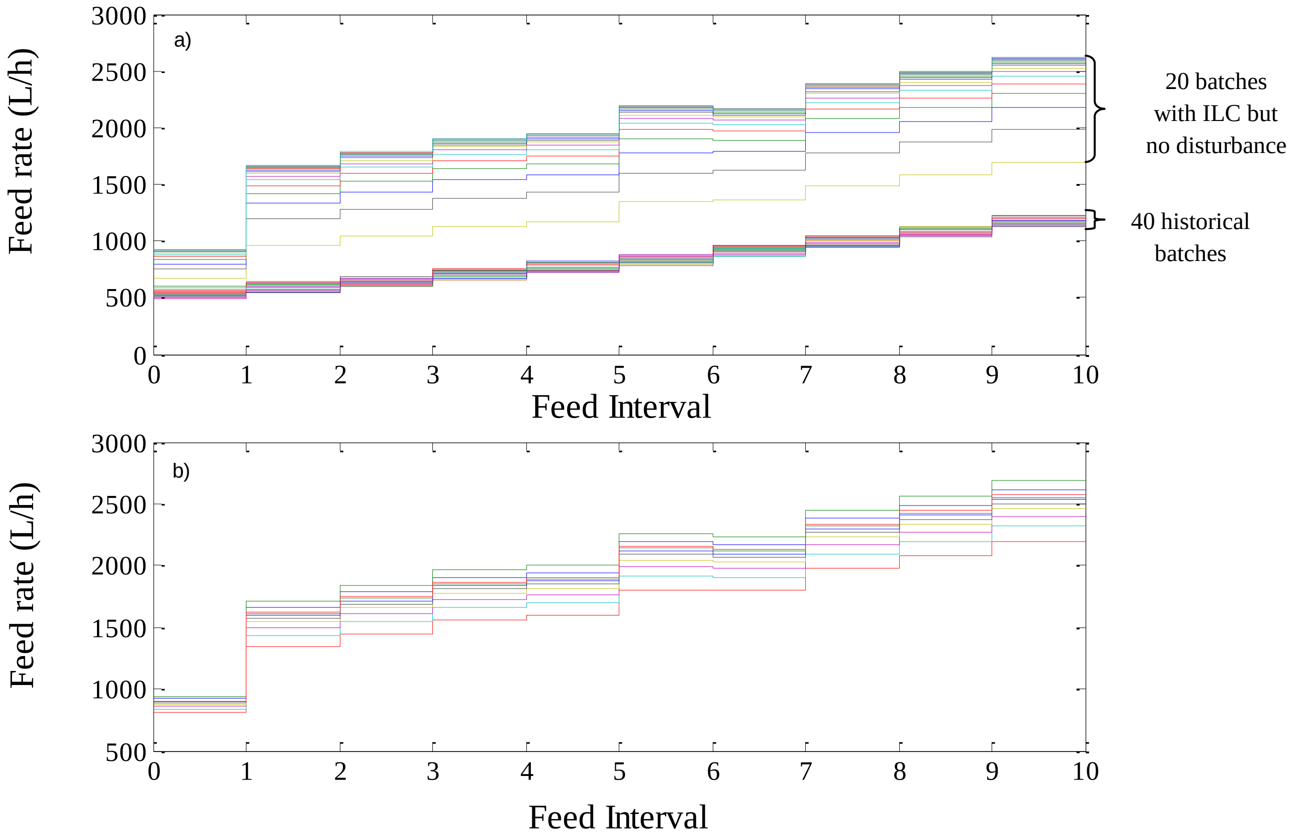


Figure 4.7: Feed rate profile for batch to batch ILC using PLS model in Case 2

With reference to Figure 4.7 (a), the first 40 historical batches exhibited an exponential feed rate profile. When the control model was introduced, the feed rate profiles exhibit a distinct pattern. The feed rate increases progressively for the first 10 batches as the batch to batch control was targeted to attain the desired trajectory. In the following 10 batches without disturbance, where the aim is to sustain an asymptotic growth, the feed rate profiles were maintained at an optimal setting. This pattern explains the increase in the output towards the desired value and then it grows asymptotically to the desired trajectory as shown in Figure 4.5 for PLS response. In Figure 4.7 (b), the feed rate pattern remains and all except one batch shows increasing profile. The one odd pattern is the feed rate for the batch immediately after disturbance was introduced. This explains the fluctuating output in Figure 4.5 for few batches after batch 21. Apart from that, the other profiles show a steady improvement in the feed rate in order to attain desired biomass output.

4.5.3 Control profile under ILC with principal component regression model

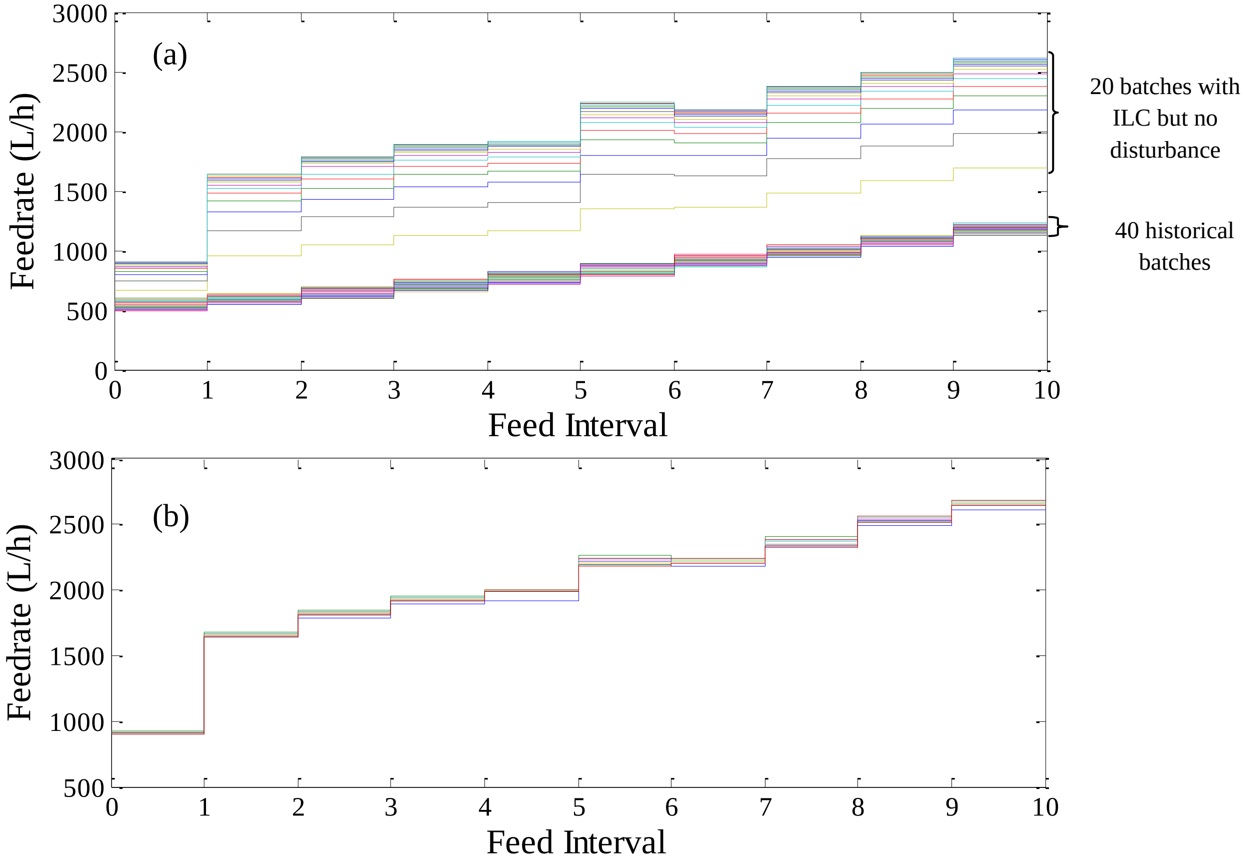


Figure 4.8: Feed rate profile for batch to batch ILC using PCR model in Case 2

As seen in Figure 4.8 (a), the historical batches are plain positive exponential feed rate profile. When the PCR linearised model was used, the feed rate profile pattern changed. The feed rate also increased for several batches and then maintained at certain range when the output increases asymptotical to the desired value. After disturbance was introduced, the feed rate profile is as shown in Figure 4.8 (b). The feed rate profile of the ten last batches seemed like one thick line, which means the differences between the feed rates for each batch is very small. This outcome goes to prove that the disturbance did not cause a significant alteration in the feed rate generation. The control model has been able to react to the disturbance to make necessary corrections in the feed rate to suppress possible huge changes in the output. The feed rate patterns seem steady and smooth and most preferred in the study.

4.6 Summary 1

In summary, using the above mentioned set of historical batches and simulation conditions, a preliminary idea on the performance of batch to batch ILC in Baker's yeast fed-batch fermentation is obtained. The non-updated model and nominal trajectories as in Case 1 for MLR, PCR and PLS is able to track the desired trajectory within the first ten batches. However, in the following ten batches, all the three regression models failed to asymptotically track the desired trajectory. In ILC, long range asymptotic tracking of desired trajectory is a desired performance pattern. For Case 3, with updated model and constant nominal trajectories, all the three models achieved desired trajectory within ten batches and then maintained asymptotic tracking for the following ten batches. The performance for batches without disturbance for Case 3 was very satisfactory. However, in the presence of non-repetitive disturbance, updated model alone was not sufficient to manage the effect of disturbance and sustain the controller performance. The MLR and PCR model sufferred high instability in the control performance though the biomass concentration was always above 40g/L. For PLS model, there were slight instability in the first few batches, and then the biomass concentration steadily reduced to almost 20g/L in the follwing batches.

The batch to batch updated model and nominal trajectories, which is Case 2, delivered favorable results for PLS and PCR model in the presence of non-repetitive disturbance. It is interesting to note that a simple batch to batch ILC with updated PCR or PLS models and nominal trajectories is capable of handling a non-repetitive disturbance without causing high instability in the batch to batch performance. After a few batches with fluctuating performance, both the batch to batch system were tracking back towards desired trajectory. However, it takes about 10 batches for the performance trajectory to recover. If in case there were more non-repetitive disturbances present within the 10 batches, the performance of the proposed batch to batch updated ILC model is doubtable. The MLR model was clearly not fit for this application in the presence of disturbance. There were high instability in the performance pattern.

The preliminary simulation run results presented above have brought about the neccessity to make several amendments to the simulation testing conditions to better suit the requirements of this study. The necessary amendments made and the results of the new conditions are presented in the sections below.

4.7 Alterations in batch to batch updated model ILC simulation testing conditions

The results from application of batch to batch iterative learning control on baker's yeast fermentation process discussed in the previous sections brought about the idea for experimentation and alteration of quantity of historical batches, number of trial batches, historical batch range and number of batches with disturbances.

It is suggested in the literature that the historical batches data should be more than the number of input variable. However, how many batches are sufficient is not known. Therefore, apart from 40 batches, 20, 30 and 50 historical batches were used in the batch to batch ILC control to investigate the effect of historical batches. It was identified that 20 historical batches were sufficient to deliver desired performance for this case study.

The next issue with historical batches is the range of the data. The output range in the case above is between 45 to 50 g/l. In the real fermentation process, the historical data may have wider range. The small range of data may not represent the real fermentation process data variation. In addition to that, the small range may not ensure that the same performance can be obtained in the presence of bigger data range. Therefore, the historical batch data range was widened for the biomass concentration to fall between 40 to 60 g/l.

Then, the number of trials needed to test the control method has to be reduced. There is no need for 30 batches of test runs. In the results above, it is notice that for both, with and without disturbance, the effectiveness of the control method can be measured within 10 batches. Therefore, 10 batch runs with disturbances and 10 batch runs without disturbance would be enough. For every techniques introduced a total of 20 batch runs will be tested, with and without disturbances.

The next query that rose from the previous results is regarding disturbances. The objective was to preserve the final biomass concentration, as nearly as possible to its ideal final value, even if the process input was subjected to changes. There are two types of disturbance that can be introduced in this case. It can be the initial conditions (Yuzgec e t al, 2009) or the initial biomass concentration. The main process input is the initial biomass concentration which has an ideal initial value that may change according to chemical laboratory quality assurance procedures (Riad et al., 2009). Considering the two, the disturbance in the initial biomass concentration was used since it could be the most likely parameter that could affect the product quality. In addition to that, this is a non-measured parameter and so could be one of the possible unknown disturbances. Therefore, the initial biomass concentration disturbance is retained as source of disturbance for this work.

Settled with the type of disturbance, the next issue is the number of batches with disturbance has to be determined. In the previous results, the disturbance was only introduced for one batch which was batch 21. The proposed batch to batch updated ILC model technique did work on the non-repetitive disturbance moderately well. However, to be more realistic, the disturbance may be present more often than once in 30 batches. In the real plants, more often than not, there will be continuous disturbances present in a batch to batch operation (Gao et al., 2001). Furthermore, a simple ILC algorithm is known to work very well with continuous disturbance rather than a non-repetitive disturbance. Since the objective of this study is to enhance the basic ILC algorithm to improve batch to batch performance, it would be better to demonstrate the efficiency of the proposed techniques with continuous disturbances. Therefore, to evaluate the robustness of the proposed control techniques in this study, the same disturbance, which initial biomass concentration will be introduced continuously for 10 batches for the rest of the simulation works.

These relevant changes were applied to the simulation and its performance was evaluated. All the other simulation conditions were retained.

4.8 New set of historical batches

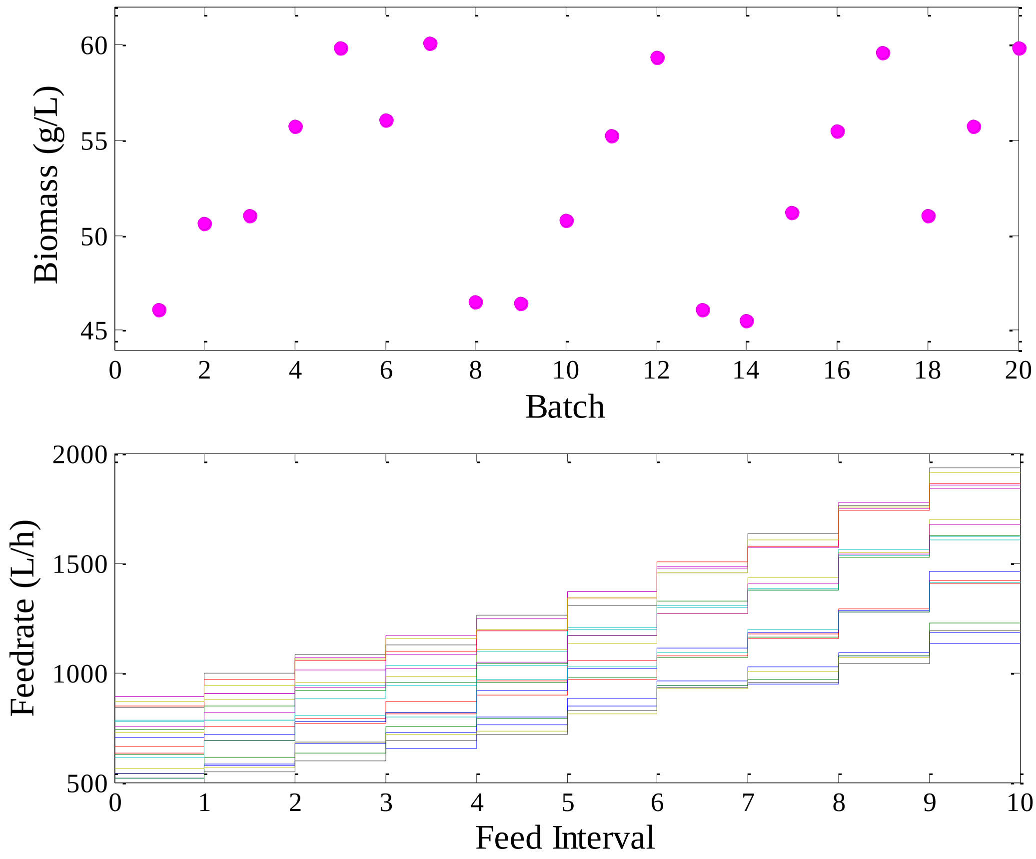
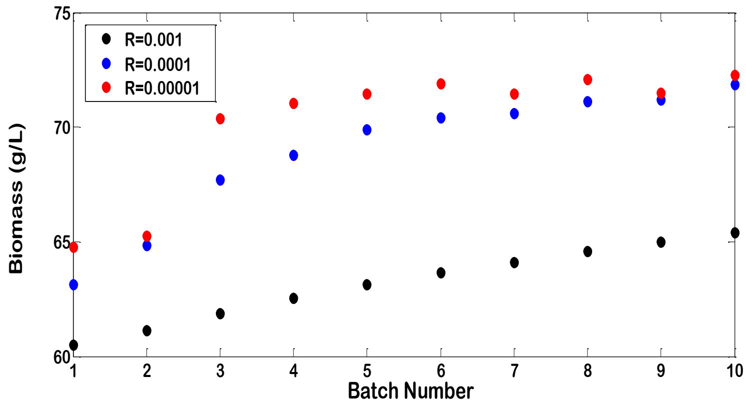


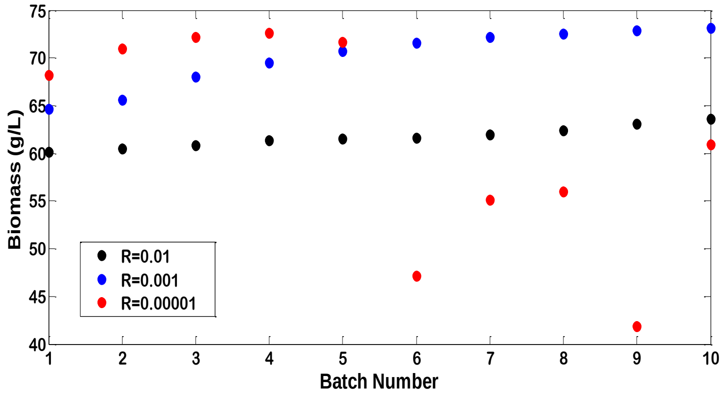
Figure 4.9: Historical batches for the new simulation condition

A new set of historical batches were generated as seen in Figure 4.9. Twenty historical batches were generated by adding random variations to the initial control profile presented above. The added random variations follow normal distribution with zero mean and standard deviation of 50 to 100. The end-batch biomass concentrations ranged between 40 and 60 g/l.

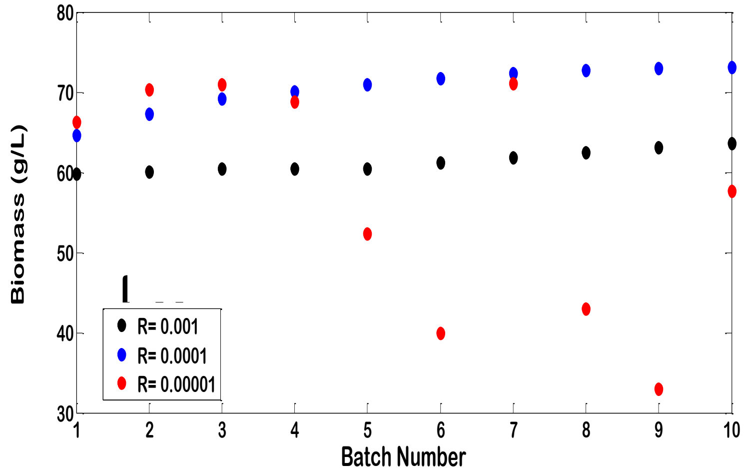
4.9 Selection of Q and R value for new set of historical batches



(a) ILC with MLR model



(b) ILC with PLS model



(c) ILC with PCR model

Figure 4.10: Comparison of different R values for new historical batches

The Q value has been set as 1, just the same as for the previous historical data. The **R** values were decided again for the new batch of historical data. The same set of **R** values, which are 0.001, 0.0001 and 0.00001 were compared to identify the most favourable value. Referring to Figure 4.10 (a), (b) and (c), it is notable that the performance patterns for **R**=0.00001**I** are unstable for all the three regression models. Meanwhile, the performance when **R**=0.001**I** is steady but very slow for the three models. The performance with R=0.0001 is the best for all the three models. There is steady and faster convergence rate. Therefore, the **R** value was fixed at **R**=0.0001**I**.

4.10 Results for Case studies 1, 2 and 3 with new set of historical batches

Batch-to-batch iterative learning control based on the three types of models (MLR, PCR, and PLS) were tested and compared for the following three different cases:

Case 1: Constant Gs, Ys and Us

Case 2: Updated Gs, Ys and Us

Case 3: Updated Gs, fixed Ys and Us

Each of the cases was run for 20 batches to investigate the control performance. The desired final bio-mass concentration value was set at 74g/L. The first 10 batches were run without any disturbance to test convergence and stability. Then, a continuous parametric disturbance was introduced from batch 11 to batch 20. The disturbance is the initial substrate concentration, S_0, which was changed to 305g/l from its nominal value of 325g/l. The disturbance was introduced to study on the ability of the control method to handle the disturbance effect and track desired trajectory within 10 batches.

For updated model runs, the data from the 20th batch were selected as nominal feed rate, Us, and nominal yield, Ys. Then, perturbation variables, U and Y were generated. These variables were used to develop model parameters using MLR, PLS or PCR. After the first simulation, the feed rate and its corresponding yield were added into the 20 historical batches and a new model parameter was generated from the total of 21 batches. The Us and Ys for the new batch are the 21st batch feed rate and yield, respectively. The cycle repeats for the next batch. The model parameter, nominal feed rate and nominal yield were updated after every batch.

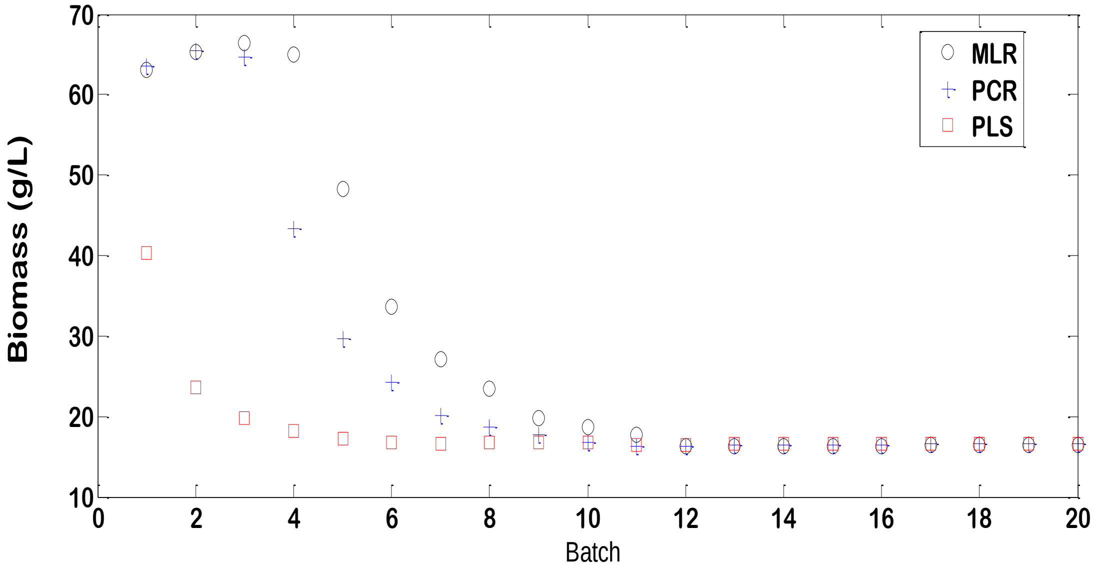


Figure 4.11: End-batch biomass concentration for Case 1: fixed Gs, Ys & Us

Figure 4.11 shows the control performance for MLR, PCR and PLS model for Case 1 with the new historical batch. It can be seen from Figure 4.11 that batch to batch control with fixed model and fixed reference trajectories does not perform for both, with and without disturbance. The end-batch biomass output is far away from the desired output which is 74 g/L. This is due to the fact that the fixed nominal model becomes invalid when the operation trajectories shift away from the nominal trajectories.

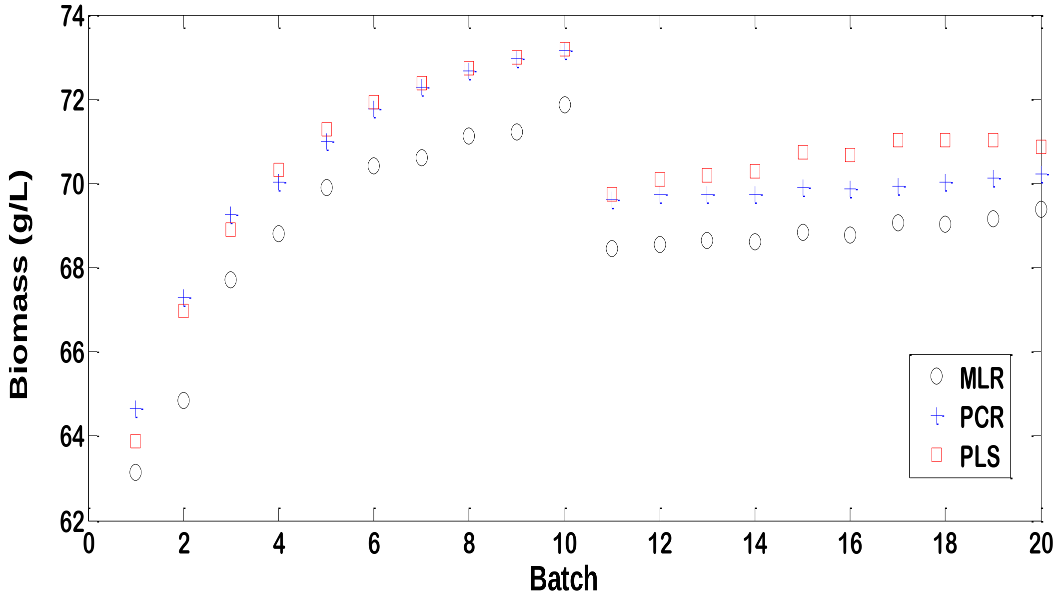


Figure 4.12: End-batch biomass concentration for Case 2: updated Gs, Ys & Us

In contrary to Case 1, the control performance for Case 2 with all the three regression models gives positive results for both, with and without disturbances as seen in Figure 4.12. The MLR model in Case 2 exhibits slightly unstable performance for all the 20 batches. This could be due to that the MLR model is not appropriate due to the correlations among the control actions during different batch stages. As for PLS model, the performance is improving steadily without disturbance but a little unsteady in the presence of unknown disturbances. However, the end-batch biomass value is the highest for PLS model. The PCR model showcases steadily improving results for all the 20 batches. It is notable that with updated linearised models and reference trajectories, batch-to-batch control using PCR model or PLS model gives satisfactory performance when unknown disturbances are present. This significant improvement over batch to batch control using MLR model is due to that the PCR and PLS models are more appropriate for correlated control actions.

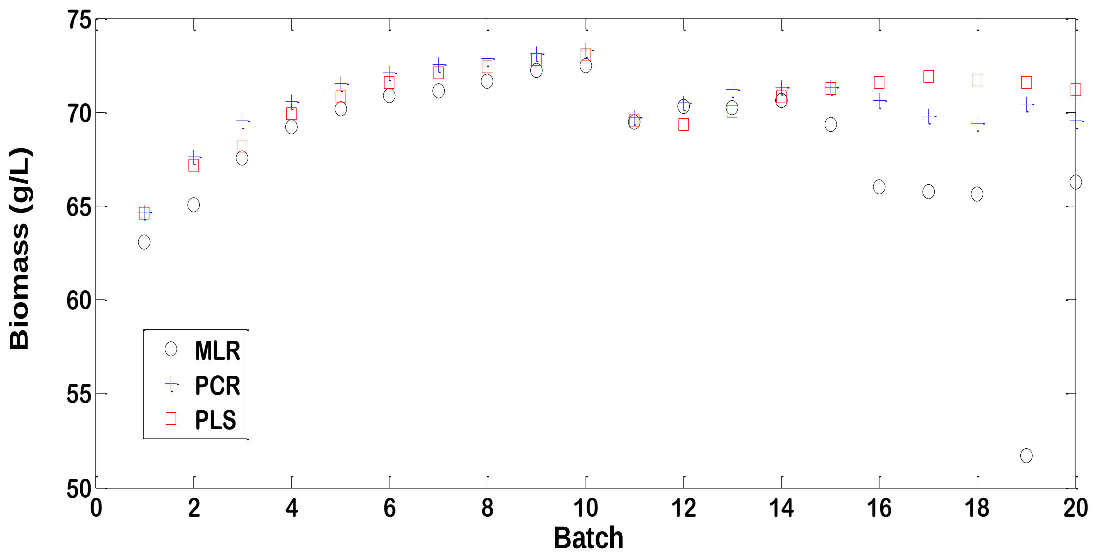


Figure 4.13: End-batch biomass concentration for Case 3: updated Gs, Fixed Ys & Us

The control performance for Case 3 with all the three regression models gives mixed results for both, with and without disturbances as seen in Figure 4.13. The MLR, PCR and PLS models in Case 3 exhibit steadily converging biomass concentrations. Amongst the three models, PCR reveals highest biomass concentrations for all the batches, followed by PLS and the lowest is using MLR model. All three models gave favourable performance pattern for batches without disturbance. In the presence of disturbance, the biomass concentration convergence rate varied with the model used. The

MLR model was unstable for all the 10 batches. This could be due to that the MLR model is not appropriate due to the correlations among the control actions during different batch stages. The PCR model showcases steadily improving results for the first 5 batches with disturbance but then fluctuates for the balance 5 batches and so is not a favourable performance pattern. As for PLS model, the performance is a little unsteady in the presence of unknown disturbances. However, the end-batch biomass value is the highest for PLS model. The results show that updating the nominal trajectories does not impact the performance when there is no disturbance, but it certainly does have a positive impact in the presence of disturbances.

4.11 Feed rates for Case 2 with new set of historical batches

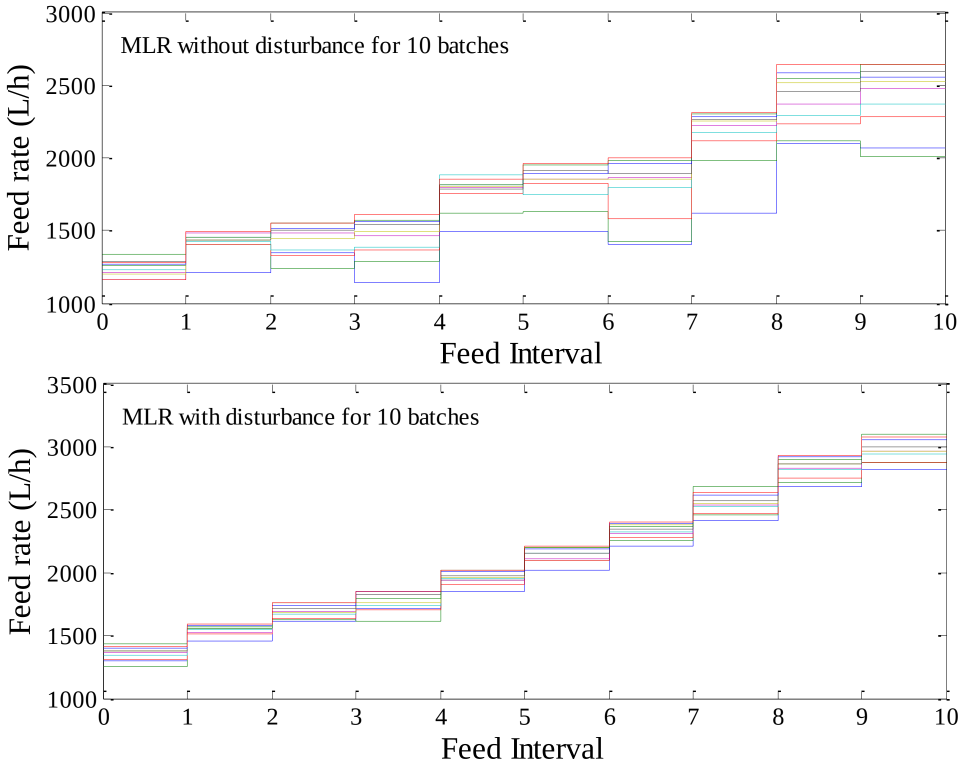


Figure 4.14: Feed rate profile for Case 2 with MLR model

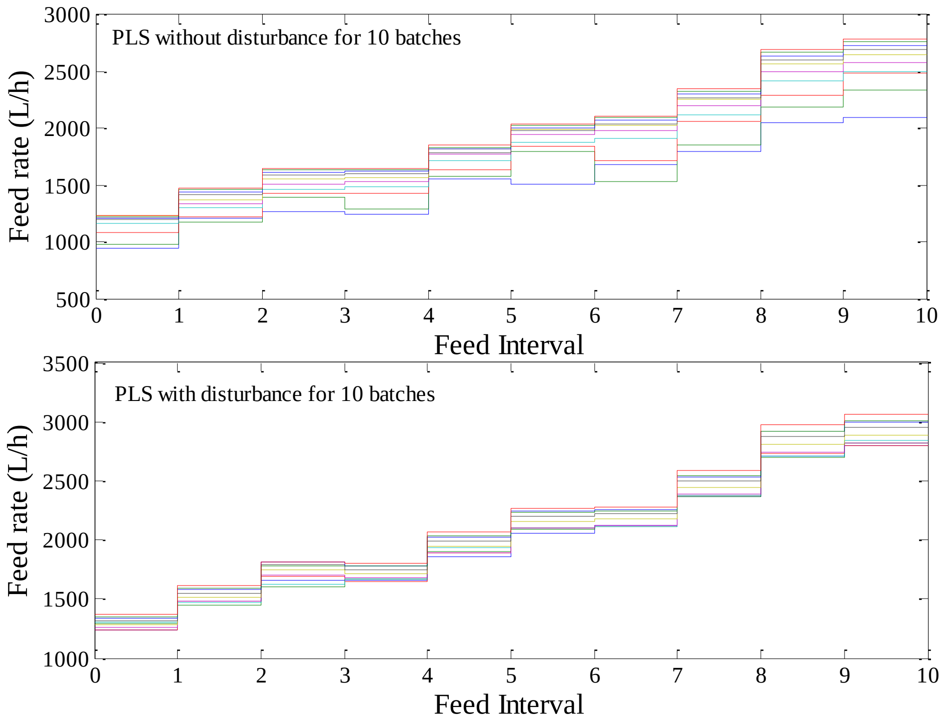


Figure 4.15: Feed rate profile for Case 2 with PLS model

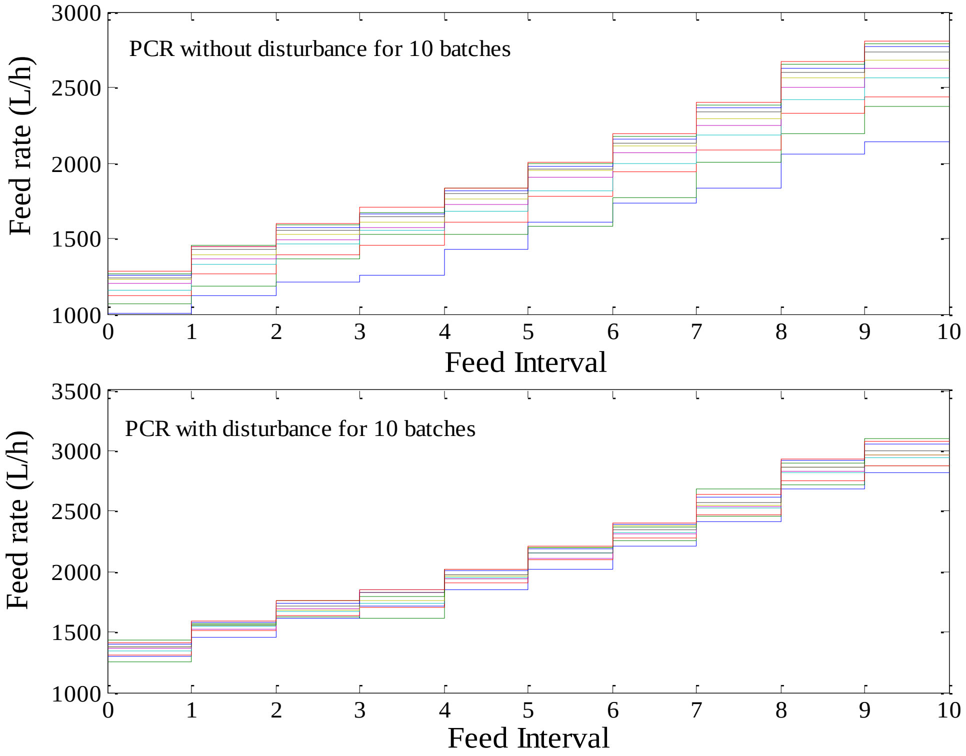


Figure 4.16: Feed rate profile for Case 2 with PCR model

Referring to Figures 4.14, 4.15 and 4.16, the feed rate profiles explain the biomass concentration plots in Case 2 for the three different models as shown in Figure 4.12. In the absence of disturbance, both the PLS and PCR models as seen in Figures 4.15 and 4.16, were gradually and stably increasing towards an optimal feed rate. That is seen as the steady increment in biomass concentration as in Figure 4.12. The MLR model in Figure 4.12 was a little unsteady and that is reflected in the feed rate profile. The feed rate profile in Figure 4.14 varied significantly for a few batches till the most appropriate ones are identified towards the end of the first ten batches. In the presence of disturbance, the PCR model exhibit steadily converging biomass concentration as shown in Figure 4.12. It can be seen in Figure 4.16, in the plot with disturbance, the feed rate profile has reached an optimal pattern and is not perturbed much by the disturbance, hence the steady performance. Both MLR and PLS models feed rates in the presence of disturbance are slightly perturbed as seen in Figures 4.14 and 4.15 respectively, which results in slightly unstable results in Figure 4.12 for batches with disturbance. The feed rate profiles are certainly modified by the batch to batch ILC method to cater to the process condition. The end batch biomass concentration performances are clearly reflected in the feed rate profile pattern.

4.12 Summary 2

In this work, **R** value has been selected as 0.0001 for all the three regression methods because it produces steady responses that converge to the desired trajectory within ten test batch runs. The Q was set to 1. Then, performance of MLR, PLS and PCR at three different cases were investigated. In contrast to the initial set of historical batches performance for Case 1, the batch to batch ILC using constant model and mean nominal trajectory of the new set of historical batches was not able to track the desired trajectory within the first ten batches. The biomass concentration dropped to 20g/L within the ten batches. Batch to batch ILC was not even able to track the desired trajectory when the historical batches were of wider range than the initial set of data. In Case 3, all the three models potrayed steadily converging biomass concentration for the first ten batches with PCR leading in performance following by PLS and then MLR. In the presence of continuous disturbance, all the three models exhibited fluctuation patterns.

Case 2 with batch to batch ILC using updated model and nominal reference trajectories produced steadfast asymptotical convergence for the first 10 batches without disturbance for PLS and PCR models. The MLR model plot portrayed slower convergence compared to the other two models with some instability. In the presence of disturbance, all the three models plot exhibit batch to batch improvement with slight instability. The PLS model exhibited highest biomass concentration for all the 10 batches with disturbances followed by PCR and then MLR model. These results shows that further enhancement can be done to the proposed method to further improve the batch to batch performance.

CHAPTER 5: RELIABLE BATCH TO BATCH ITERATIVE LEARNING CONTROL OF A FED-BATCH FERMENTATION PROCESS

5.1 Introduction

Utilising the repetitive nature of batch or fed-batch processes, fermentation operation recipes can be modified from batch to batch through iterative learning control (ILC) in order to overcome the detrimental effect of model-plant mismatches and unknown disturbances. As many batch processes are highly nonlinear, the batch-wise linearised model may only be valid over a small operating range. Thus control actions from the ILC strategy may not be reliable if the mismatch between the linearised model and the plant becomes large. In order to overcome this problem, a reliable model based ILC strategy is proposed in this chapter.

Model prediction confidence bounds give an indication of the reliability of the associated model prediction. The calculation of model prediction confidence bounds is dependent on the data distribution of the process data used for modelling. An accurate confidence bound can be obtained if there is sufficient knowledge of the data distribution. In a complex system like fed batch fermentation, exact data distribution is difficult to be ascertained. Confidence bounds can be identified by assuming a distribution pattern or population density based approach. Assumption based confidence bounds calculation methods may not accurately define the confidence region of a complex system (Martin and Morris, 1996). Therefore, confidence bounds developed from the natural process data distribution is expected to give a more realistic and reliable prediction value.

Model prediction confidence bounds for expected response can be obtained from historical process operation data used for model identification. The model prediction confidence bound is incorporated into the model based ILC optimisation objective function and control policies leading to wide model prediction confidence bounds are penalised. The question is how often the bounds should be re-identified. In a complex, dynamic system, the physical and physiological condition may be changing over time. In order to cope with process nonlinearities, the batch-wise linearised model is re-identified after each batch run with the immediate previous batch as the reference batch. Similarly, the confidence bound is re-identified after batch run by including the immediate previous batch to update changes. Introduction of weighted parameter to the model prediction confidence bounds renders certain control on how the confidence bounds affects the objective function.

Zhang (2004) presents a study on reliable neural network based optimisation control strategy on batch polymerization process. Bootstrap aggregated network is used to calculate the model prediction confidence bound which is then incorporated into objective function. The term $\lambda^T\sigma_e$ was incorporated into objective function where σ_e is a vector of standard prediction errors while λ is a vector of weightings for σ_e. This term is meant to penalise wide model prediction confidence bounds to render a more reliable control policy. As the weighting of the standard prediction error increases, the confidence bounds become narrower and the actual final product quality was closer to the defined constraints. The individual network predictions were less diverse indicating improved reliability. When the weighting increases the reduction in standard prediction error becomes less significant. This work adopts the same philosophy to ILC where model prediction confidence bounds are incorporated into the ILC optimisation objective function. The proposed method is applied to a simulated fed-batch fermentation process and the results demonstrate that the proposed reliable ILC strategy is very effective.

The chapter is organised as follows: Section 5.2 presents a bit of theory on confidence interval and its relation to model prediction confidence bounds for predicted response. Section 5.3 presents the development of reliable ILC with MLR, PCR and PLS models. Section 5.4 presents the results and discussion for the proposed method. Finally, Section 5.5 concludes the chapter.

5.2 Confidence intervals for multiple linear regression model predictions

Considering the following multiple input single output (MISO) linear model

$$y = u_1\theta_1 + u_2\theta_2 + \ldots + u_n\theta_n \tag{5.1}$$

where y is the model output, u_1 to u_n are model inputs, and θ_1 to θ_n are model parameters. For a set of pre-processed input and output data, **X** and **Y**, the model parameters can be obtained from least square estimator as

$$\hat{\boldsymbol{\theta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \tag{5.2}$$

Since $\hat{\boldsymbol{\theta}}$ are unbiased estimators of $\boldsymbol{\theta}$,

$$\hat{y} = u_1\hat{\theta}_1 + u_2\hat{\theta}_2 + \ldots + u_n\hat{\theta}_n \tag{5.3}$$

The expected model response is

$$E[y] = u_1\theta_1 + u_2\theta_2 + \ldots + u_n\theta_n \tag{5.4}$$

Let input, $\boldsymbol{u} = (u_1 \;\; u_2 \;\; u_3 \ldots u_n)^T$, then

$\hat{y} = \boldsymbol{u}^{T}\hat{\boldsymbol{\theta}}$ (5.5)

$E[\hat{y}] = \boldsymbol{u}^{T}E[\hat{\boldsymbol{\theta}}] = \boldsymbol{u}^{T}\boldsymbol{\theta} = E[y]$ (5.6)

$Var[\hat{y}] = \boldsymbol{u}^{T}Var[\hat{\boldsymbol{\theta}}]\boldsymbol{u} = \sigma^{2}\,\boldsymbol{u}^{T}(\mathbf{X}^{T}\mathbf{X})^{-1}\boldsymbol{u}$ (5.7)

The variances of the $\hat{\boldsymbol{\theta}}$ are obtained using the $(\mathbf{X}^{T}\mathbf{X})^{-1}$ matrix.

The normal distribution of the predicted responses are defined as

$\hat{y} \sim N(E[y], Var[\hat{y}])$ (5.8)

$\hat{y} \sim N(E[y], \sigma^{2}\boldsymbol{u}^{T}(\boldsymbol{X}^{T}\boldsymbol{X})^{-1}\boldsymbol{u})$ (5.9)

From Eq.5.9, the model prediction confidence bound for predicted response is proportional to normal prediction variance. The variance-covariance matrix of the estimated regression coefficients is obtained as follows:

$\mathbf{C} = \hat{\sigma}^{2}(\mathbf{X}'\mathbf{X})^{-1}$ (5.10)

where **C** is a symmetric matrix whose diagonal elements, $\mathbf{C}_{jj}$, represent the variance of the estimated j^th regression coefficient, $\hat{\boldsymbol{\theta}}_{jj}$. The off-diagonal elements, $\mathbf{C}_{ij}$, represent the covariance between the i^th and j^th estimated regression coefficients, $\hat{\boldsymbol{\theta}}_{i}$ and $\hat{\boldsymbol{\theta}}_{j}$. The value of $\hat{\sigma}^{2}$ is obtained using the error mean square, MSE. The positive square root of C_{jj} represents the estimated standard deviation of the j^th regression coefficient, $\hat{\boldsymbol{\theta}}_{j}$, and is called the *estimated standard error* of $\hat{\boldsymbol{\theta}}_{j}$.

5.3 Incorporation of model prediction confidence bounds to enforce reliability

5.3.1 Reliable ILC with multiple linear regression models

The linearised model can be identified from historical process operation data using MLR (Xiong and Zhang, 2003). Let **X** and **Y** be the deviations from the reference trajectories of historical data in the manipulated variables and product quality variables respectively, then

$\mathbf{Y} = \mathbf{G}_{s}\mathbf{X}$ (5.11)

and the linearised model $\mathbf{G}_{s}$ can be obtained through MLR as

$\mathbf{G}_{s} = (\mathbf{X}^{T}\mathbf{X})^{-1}\mathbf{X}^{T}\mathbf{Y}$ (5.12)

Model predictions for the k^{th} batch can be calculated as

$\hat{\mathbf{Y}}_{k} = \mathbf{G}_{s}\Delta\mathbf{U}_{k}$ (5.13)

The model prediction confidence interval is proportional to variance of model predictions (Gunst and Mason, 1980), $\sigma^2 \Delta\mathbf{U}_k^T(\mathbf{X}^T\mathbf{X})^{-1}\Delta\mathbf{U}_k$, where σ^2 is the model prediction error variance. A narrower confidence bounds indicate a more reliable model prediction.

In the batch to batch ILC law as presented in (Xiong and Zhang, 2003), the following quadratic objective function was solved upon the completion of the kth batch run to identify the necessary changes needed to update the input trajectory for the $(k+1)^{th}$ batch run.

$$J_{k+1} = \min_{\Delta\overline{\mathbf{U}}_{k+1}} \frac{1}{2}[\tilde{\mathbf{e}}_{k+1}^T \mathbf{Q} \tilde{\mathbf{e}}_{k+1} + \Delta\overline{\mathbf{U}}_{k+1}^T \mathbf{R} \Delta\overline{\mathbf{U}}_{k+1}] \tag{5.14}$$

In order to enhance the reliability of batch-to-batch ILC, model prediction confidence bounds penalty term is incorporated in the ILC optimisation objective function as shown below

$$J_{k+1} = \min_{\Delta\mathbf{U}_{k+1}} \frac{1}{2}[\mathbf{e}_{k+1}^T \mathbf{Q} \mathbf{e}_{k+1} + \Delta\mathbf{U}_{k+1}^T \mathbf{R} \Delta\mathbf{U}_{k+1} + \lambda \Delta\mathbf{U}_{k+1}^T (\mathbf{X}^T\mathbf{X})^{-1} \Delta\mathbf{U}_{k+1}] \tag{5.15}$$

where $\mathbf{e}_{k+1} = \mathbf{Y}_d - \mathbf{Y}_k$ is the tracking error for the (k+1)th batch, **Q** and **R** are positive definitive matrices, and λ is a weighting parameter to penalise wide model prediction confidence bounds.

Note that the objective function, Eq.(5.15), has two penalty terms, one on the input change $\Delta\mathbf{U}_{k+1}$=($\mathbf{U}_{k+1}$-$\mathbf{U}_k$) between two adjacent batch runs and the other on model prediction reliability. The two penalty terms can be combined and the objective function can be re-written as

$$J_{k+1} = \min_{\Delta\mathbf{U}_{k+1}} \frac{1}{2}[\mathbf{e}_{k+1}^T \mathbf{Q} \mathbf{e}_{k+1} + \Delta\mathbf{U}_{k+1}^T (\mathbf{R} + \lambda(\mathbf{X}^T\mathbf{X})^{-1}) \Delta\mathbf{U}_{k+1}] \tag{5.16}$$

As the calculated control action are the increments between the (k+1)th and the kth batches, the ILC algorithm has an integral action with respect to the batch index k (Xiong and Zhang, 2003). Large values of λ will prevent the control actions going to regions where the model predictions are not confident making the ILC strategy reliable.

By finding the partial derivative of the quadratic objective function Eq(5.16) with respect to the input change $\Delta\overline{\mathbf{U}}_{k+1}$ and through straightforward manipulation, the following ILC law is obtained

$$\Delta\overline{\mathbf{U}}_{k+1} = \hat{\mathbf{K}} \mathbf{e}_k \tag{5.17}$$

where $\hat{\mathbf{K}}$ is defined as the learning rate

$$\hat{\mathbf{K}} = [\hat{\mathbf{G}}_s^T \mathbf{Q} \hat{\mathbf{G}}_s + \mathbf{R} + \lambda(\mathbf{X}^T \mathbf{X})]^{-1} \hat{\mathbf{G}}_s^T \mathbf{Q} \quad (5.18)$$

$\mathbf{e_k}$ = tracking error of the process and pertubation model

The ILC law for the control trajectory of batch *k+1* can also be written as

$$\mathbf{U}_{k+1} = \mathbf{U}_k + \hat{\mathbf{K}} \mathbf{e}_k \quad (5.19)$$

Part of the penalty term, which is ($\mathbf{X}^T\mathbf{X}$), is re-identified after every batch run. The immediate previous batch data is added into the historical batch data pool, linearised and ($\mathbf{X}^T\mathbf{X}$) is calculated using all of the historical batches. The pool of historical keeps growing with the batch trials. The weighting parameter, λ, is a scalar. It. was determined by trial and error. The performance of batch to batch control using the modified objective function for varying λ value is discussed in this chapter.

5.3.2 Reliable ILC with principal component regression models

As for PCR method, the matrix **X** is decomposed into the sum of a series of rank one matrix through principal component decomposition.

$$\mathbf{X} = \mathbf{t}_1 \mathbf{p}_1^T + \mathbf{t}_2 \mathbf{p}_2^T + \ldots + \mathbf{t}_N \mathbf{p}_N^T \quad (5.20)$$

where $\mathbf{t}_i$ and $\mathbf{p}_i$ are the i^{th} score vector and loading vector respectively. The model output is obtained as a linear combination of the first *k* principal components of **X** as

$$\hat{\mathbf{Y}} = \mathbf{T}_k \mathbf{w} = \mathbf{X} \mathbf{P}_k \mathbf{w} \quad (5.21)$$

where **w** is a vector of model parameters in terms of principal components. The least squares estimation of **w** is:

$$\mathbf{w} = (\mathbf{T}_k^T \mathbf{T}_k)^{-1} \mathbf{T}_k^T \mathbf{Y} = (\mathbf{P}_k^T \mathbf{X}^T \mathbf{X} \mathbf{P}_k)^{-1} \mathbf{P}_k^T \mathbf{X}^T \mathbf{Y} \quad (5.22)$$

The model parameters calculated through PCR is then

$$\boldsymbol{\theta} = \mathbf{P}_k \mathbf{w} = \mathbf{P}_k (\mathbf{P}_k^T \mathbf{X}^T \mathbf{X} \mathbf{P}_k)^{-1} \mathbf{P}_k^T \mathbf{X}^T \mathbf{Y} \quad (5.23)$$

Therefore the modified objective function is

$$J_{k+1} = \min_{\Delta \mathbf{U}_{k+1}} \frac{1}{2} [\mathbf{e}_{k+1}^T \mathbf{Q} \mathbf{e}_{k+1} + \Delta \mathbf{U}_{k+1}^T (\mathbf{R} + \lambda (\mathbf{T}^T \mathbf{T})^{-1}) \Delta \mathbf{U}_{k+1}] \quad (5.24)$$

5.3.3 Reliable ILC with partial least squares models

PLS projects the X and Y matrices to a subset of latent variables, t and u, respectively.

$$\mathbf{X} = \sum_{j=1}^{k} \mathbf{t}_j \mathbf{p}_j^T + \mathbf{E} \qquad (5.25)$$

$$\mathbf{Y} = \sum_{j=1}^{k} \mathbf{u}_j \mathbf{q}_j^T + \mathbf{F} \qquad (5.26)$$

The linear relationship between X and Y is obtained by performing least square regression between each pair of corresponding t and u latent vectors while making $\|F\|$ as small as possible.

$$\hat{\mathbf{u}}_j = \mathbf{t}_j \mathbf{b}_j \qquad j=1,2,\ldots k \qquad (5.27)$$

where b_j is the coefficient from the inner linear regression between the j^{th} latent variables u_j and t_j which is

$$\mathbf{b}_j = \left(\mathbf{t}_j^T \mathbf{t}_j\right)^{-1} \mathbf{t}_j^T \mathbf{u}_j \qquad (5.28)$$

Eq. 5.28 is equivalent to Eq. 5.12 and Eq. 5.22. The modified objective function for PLS is

$$J_{k+1} = \min_{\Delta \mathbf{U}_{k+1}} \frac{1}{2}[\mathbf{e}_{k+1}^T \mathbf{Q} \mathbf{e}_{k+1} + \Delta \mathbf{U}_{k+1}^T (\mathbf{R} + \lambda (\mathbf{t}^T \mathbf{t})^{-1}) \Delta \mathbf{U}_{k+1}] \qquad (5.29)$$

5.4 Results and Discussion

There were 20 historical batches used in this work. Batch 0 in all the graphs is the 20th historical batch used in the study. Batches 1 to 10 are batch trials without disturbance for which the objective is to attain desired biomass production, Y_d= 74g/L. At batch 11, a disturbance was introduced in that the initial substrate concentration was changed to 305g/l from its nominal value of 325g/l. The following 10 batches, which are batches 11 to 20 were introduced to the disturbance and the end-batch biomass production performance were analysed. The graph with λ=0 refers to performance of batch to batch ILC with the non-modified objective function that was presented in Chapter 4, Figure 4.12.

5.4.1 Results for ILC with multiple linear regressions model

As shown in Chapter 4, Figure 4.12, batch to batch ILC using updated MLR models with λ=0, which corresponds to the ILC strategy in (Xiong and Zhang, 2003), can improve process operation from batch to batch. However, the performance patterns can be further improved with λ>0. Several penalty parameters, λ, values ranging from 0 to

100 were tried to evaluate its impact on ILC performance. The final biomass output pattern when λ is 0.001 and 0.01 with and without disturbance had very small difference compared to when λ=0 but there were slight increment in biomass concentrations in every batch when λ was increased from 0.001 to 0.01. This initial trial gives an indication that bigger λvalue will give better performance. Therefore, more trials were done with λvalue bigger than 0.01 and the results are presented below.

Table 5.1: Selected end-batch biomass concentrations (g/l) for MLR for λ between 0.1 and 0.5.

Batch number	λ					
	0	0.1	0.2	0.3	0.4	0.5
0	59.8	59.8	59.8	59.8	59.8	59.8
1	63.1	62.0	63.6	63.7	63.8	63.8
10	71.8	72.4	72.0	71.8	71.8	71.8
11	68.5	68.7	68.5	68.3	68.3	68.3
20	69.4	69.8	69.6	69.4	69.5	69.4

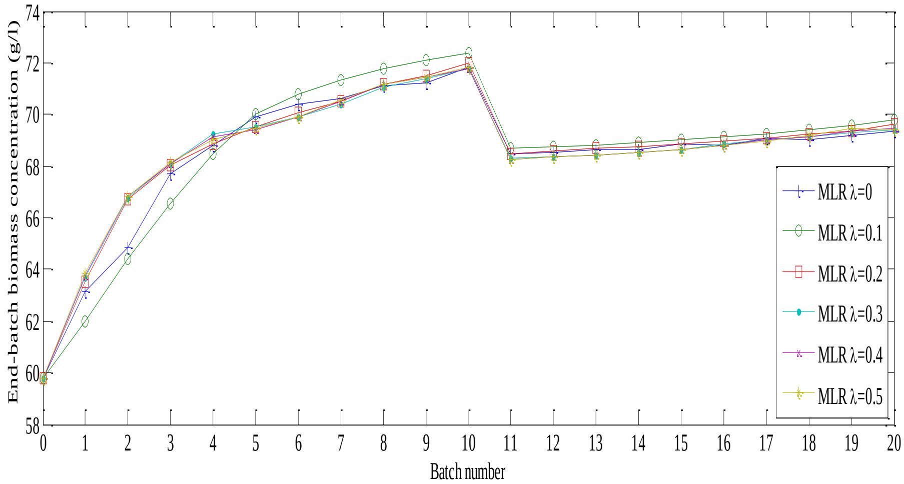


Figure 5.1: End-batch biomass concentration for ILC with MLR when λ is 0.1, 0.2, 0.3, 0.4 and 0.5.

Referring to Figure 5.1, it is noticeable that there is mixed performance pattern with different λ values. Clearly, λ=0.1 exhibit contrasting performance pattern compared to the rest of the λ values. The end-batch biomass concentration for every test batch when λ=0.1, increased steadily with and without disturbance. For the first 4 batches, the convergence were slowest compared to the other λ values but from the 5th batch to the 20th batch, steady and fastest converging biomass concentration were observed. The biomass concentrations for λ=0.2, 0.3, 0.4 and 0.5 have similar performance pattern with and without disturbance. Though there is slight improvement in the stability of the performance pattern for these values compared to λ=0, the increment in biomass concentration is less significant especially for batches with disturbances. The batches without disturbance increased significantly for the first 4 batches, and then the following 6 batches did not differ much from λ=0. Referring to Table 5.1, the final biomass concentrations of 10th batch for λ=0.3, 0.4 and 0.5 are similar to λ=0, which is 71.8g/l. There were slight increments for λ=0.2 and λ=0.1 which are 72g/l and 72.4 g/l respectively. For batch 20, there were small increment for λ=0.1, 0.2 and 0.4 which is 69.8g/l, 69.6g/l and 69.5 respectively compared to 69.4g/l for λ=0, 0.3 and 0.5. Overall, λ=0.1 has more steady and faster converging performance comparing to the performance of λ=0. Since the increments were small, bigger λ values were attempted.

Table 5.2: End-batch biomass concentrations (g/l) for ILC with MLR for λ between 1 and 5.

Batch number	λ							
	0	1	2	3	3.5	4	4.5	5
0	59.8	59.8	59.8	59.8	59.8	59.8	59.8	59.8
1	63.1	64.1	64.3	64.3	64.4	64.4	64.4	64.4
10	71.8	71.8	72.2	72.7	72.8	72.9	73.0	73.0
11	68.5	68.3	68.6	69.2	69.3	69.4	69.5	69.6
20	69.4	69.3	69.1	69.9	71.3	71.2	71.1	70.7

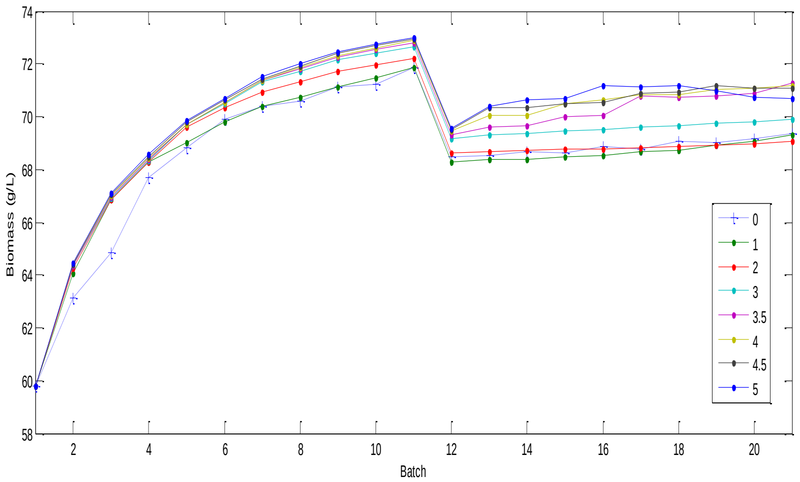


Figure 5.2 (a): End-batch biomass concentration for ILC with MLR for λ between 1 and 5.

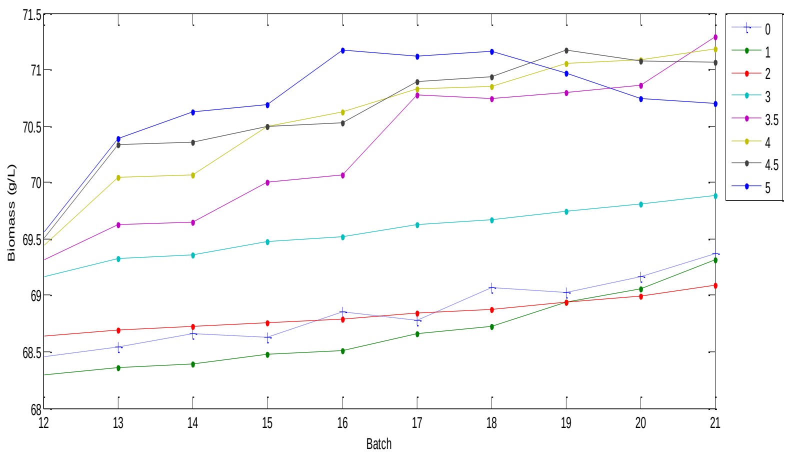


Figure 5.2 (b): End-batch biomass concentration for ILC with MLR for λ between 1 and 5 in the presence of disturbance.

Referring to Figure 5.2(a), for the first 10 batches with no disturbance, the increment of λ from 1 to 5 show noticeable batch to batch improvement in end-batch biomass concentration compared to the case when λ=0. As seen in Table 5.2, the biomass concentration at the 10th batch increased from 71.8g/l to 73 g/l with increasing λ values. For λ=1, the biomass concentration improved steadily for the first 3 batches and then matched the results for λ=0 for the rest of the 7 batches without disturbance. Therefore, the end of batch biomass concentration at the 10th batch is similar to that when λ=0. For the cases when λ values are from 3 to 5, the increments in the end of batch biomass concentration with increasing λ values were small. As seen in Table 5.2, the increments were only 0.1g/l for the 10th batch. The plot for λ=2 showed an intermediate performance.

In the presence of disturbances, the end batch biomass concentration increased for most of the batches with increasing λ values but the stability of the performance were affected for λ values between 3.5 and 5. A closer look at the batches with disturbance in Figure 5.2 (b) shows a clearer picture of the batch to batch performance for the studied λ values. For λ=1 and 2, although the end of batch biomass concentration revealed increasing pattern, most of the end-batch concentrations were slightly below that of λ=0. The biomass concentration increased noticeably for λ values between 3 and 5. It is clear that λ=3 shows steady but slow improvement. Meanwhile the cases with λ=3.5, 4, and 4.5 exhibit fluctuating but improving performance. For λ=5, the end-batch biomass concentration were the highest for batches 11 to 17 despite slight fluctuation, but the yield dipped a little from batch 18 to batch 20. Referring to Table 5.2, the end of batch biomass concentration at the 20th batch for λ= 5 is 70.7 g/l and those for λ= 3.5, 4 and 4.5 were about 71g/l. Overall, in this set of λ values 3.5 and 4 are favourable. The value λ= 5 would be a better choice if the performance can be improved for a steadier performance. Further increments in λ value were introduced to study more performance pattern and identify a suitable λ value for this case

Table 5.3: Selected end-batch biomass concentrations (g/l) for ILC with MLR for λ between 5.5 and 10.

Batch number	λ						
	0	5.5	6	7	8	9	10
0	59.8	59.8	59.8	59.8	59.8	59.8	59.8
1	63.1	64.4	64.5	64.5	64.5	64.5	64.5
10	71.8	73.0	73.1	73.1	73.2	73.2	73.2
11	68.5	69.6	69.6	69.7	69.7	69.7	69.7
20	69.4	70.6	70.5	70.3	70.4	70.7	70.8

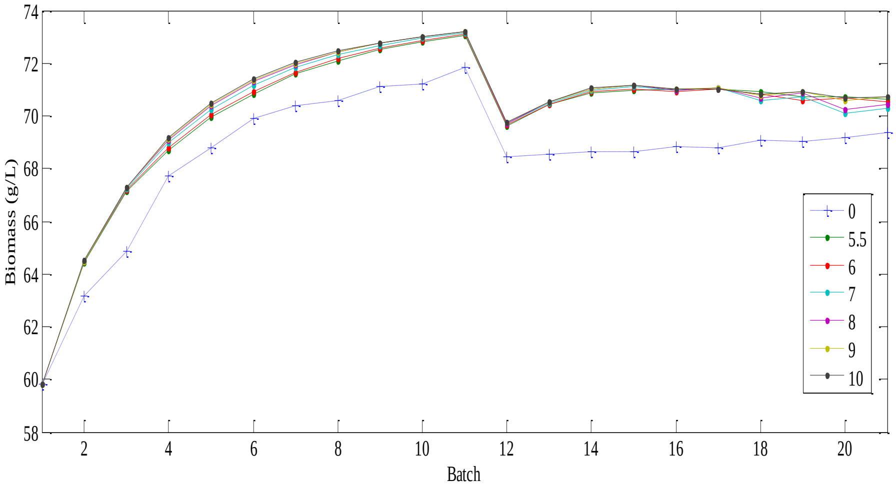


Figure 5.3 (a): End-batch biomass concentrations for ILC with MLR for λ between5.5 and 10

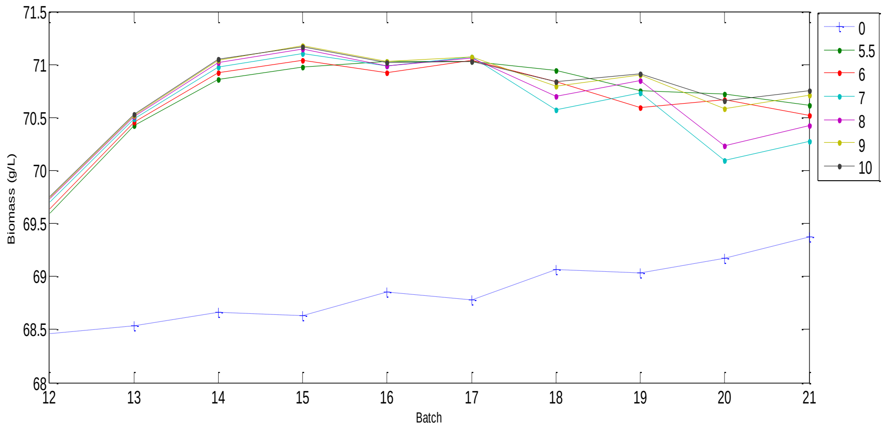


Figure 5.3 (b): End-batch biomass concentration for ILC with MLR for λ between 5.5 and 10 in the presence of disturbance

Increments in λ values from 5.5 to 10 with and without disturbances show similar performance pattern as seen in Figure 5.3(a). It can be seen in Table 5.3 that for the selected batches, the end-batch biomass concentrations did not differ much. The yield before disturbance did not have much variation amongst λ values from 5.5 to 10 for the first 2 batches but it is noticeable that from the 3rd batch onwards there is increment in end of batch biomass concentration with increasing λ values. There were steady and converging improvements in the biomass concentration without disturbance. There were significant improvements in biomass concentration with and without disturbances compared to the case with λ=0.

Referring to Figure 5.3(b), when disturbances were introduced, the improvement in final biomass concentration decreased when λ is from 5.5 to 7. At λ=9 and 10, the final biomass concentrations were similar to that when λ=5 but with more fluctuating pattern. In general, from λ=4.5 onwards, increment in the final biomass concentration show similar performance pattern for batches with disturbance. The pattern is like a dumb bell shape, where there is increment initially, reaches a peak and then decrease towards the end. In this set of λ values, λ=10 is the most acceptable performance pattern because the corresponding final biomass concentrations are the highest for almost all the trial batches. The fluctuations were small and could be improved further.

Table 5.4: Selected batches end-batch biomass concentrations (g/l) for λ is 11, 15, 25, 50 and 100

Batch number	λ					
	0	11	15	25	50	100
0	59.8	59.8	59.8	59.8	59.8	59.8
1	63.1	64.5	64.5	64.5	64.5	64.5
10	71.8	73.2	73.2	73.2	73.2	73.2
11	68.5	69.7	69.8	69.7	69.7	69.7
20	69.4	70.6	70.2	69.8	65.1	67.9

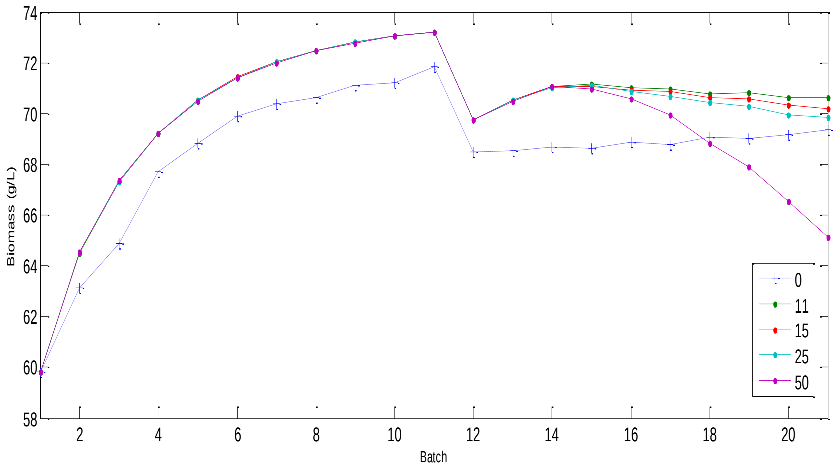


Figure 5.4: End-batch biomass concentrations for ILC with MLR when λ is 11, 15, 25 and 50.

Since λ=10 exhibits a high final biomass concentration, bigger λ values were attempted to observe if there will be improvement in both final biomass concentration and performance pattern. From Figure 5.4 it is evident that the performance pattern remains similar but the biomass production decreases as the λ value increases. The dumb bell pattern continued to grow wider as the λ value increases. Table 5.4 shows that final biomass concentration for the 1st, 10th and 11th batches are similar to the ones when λ is 10 (Table 5.3) and the yield at 20th batch is between 65.1g/l to 70.6g/l, which are lower than 70.8g/l, which is the 20th batch yield for λ=10. Therefore, λ value above 10 does not seem to be desirable.

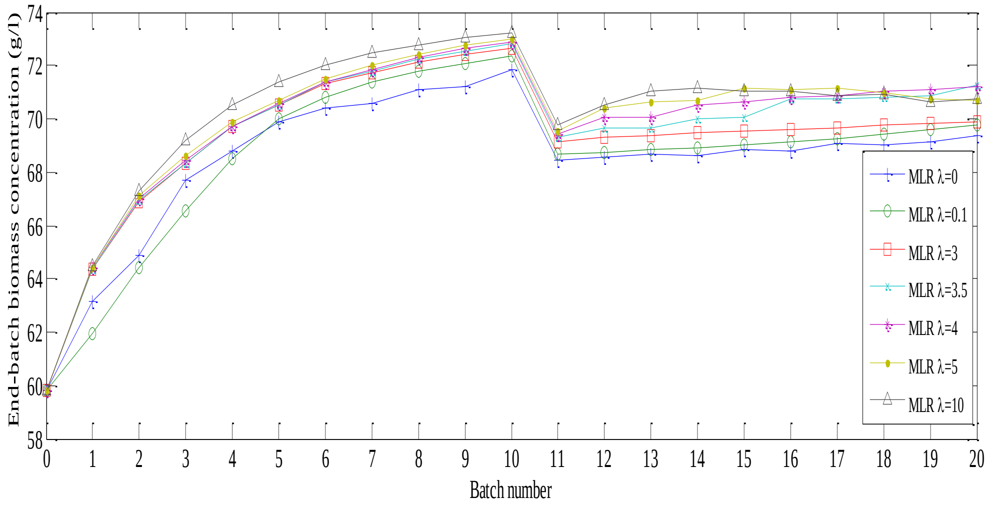


Figure 5.5: End-batch biomass concentration for ILC with MLR when λ is 0.1, 3, 3.5, 4, 5 and 10.

From all the λ values tested for ILC with MLR model, the selected values which gave favourable results are gathered and presented in Figure 5.5. Each plot exhibits steady increasing biomass value with and without disturbance as λ value increases. There is a mixture of performance pattern. Figure 5.5 shows that much improved control performance is obtained when λ takes the values of 3.5, 4, 5 or 10. The performances for the cases with λ=0.1 and 3 are very stable but slow improvement and low final biomass concentration. As for the one with λ=10, the results seem to fluctuate and drop but the final biomass concentration from batch 12 to batch 17 are the highest amongst all others and as for batches 18 to 21, biomass production is still higher than the cases when λ =0.1 and 3. Figure 5.5 show that the proposed reliable ILC strategy can increase the final biomass concentration from the initial value of 60 g/l to around 73 g/l after 10 batches. Due to the presence of unknown disturbance from batch 11, the final biomass concentration dropped at the 11th batch. However, it was increased from batch to batch under ILC using the modified objective function in the following ten batches.

In summary, incorporation of model prediction confidence bounds in the ILC objective function definitely improves the ILC performance with MLR model when compared with the base ILC in Chapter 4. In fact, the results are as good as the ones by the ILC with PCR model without the addition of the parameter as shown. The incorporation of the model prediction confidence bounds improves the model prediction reliability by further narrowing the confidence bounds. Therefore, reliable control results can be obtained. The results show that reliable ILC can effectively overcome the detrimental effect of model plant mismatches and unknown disturbances.

5.4.2 Results for ILC with principal component regression model

Batch to batch control with λ=0 exhibits steady and converging final biomass concentration. The final biomass concentration at the 10th batch is 73.1g/l. However, there was slight instability in the performance pattern, in the presence of disturbance. By incorporating the model prediction confidence bounds, a more stable and reliable improvement in the final biomass concentration especially in the presence of disturbance is expected.

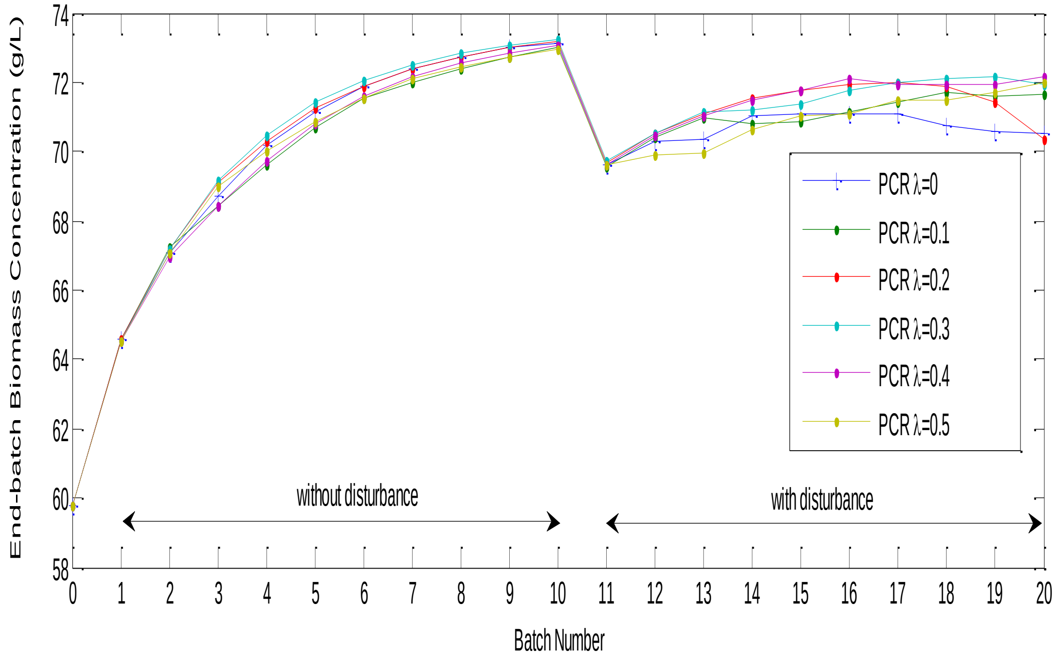


Figure 5.6: End-batch biomass concentration for ILC with PCR when λ is 0.1, 0.2, 0.3, 0.4 and 0.5

In contrast to the ILC with MLR performance pattern for these λ values, the PCR method shows significant improvement as seen in Figure 5.6. For the first 10 batches without disturbance, all the λ values exhibit steadily increasing performance towards desired value of 74g/l. The plot with λ=0.3 show slightly higher end-batch biomass concentration for every batch without disturbance in compariosn with the case when λ=0. Incorporation of model prediction confidence bounds has better impact on the performance in the presence of disturbance. The plots for all the λ values shown in Figure 5.6 except λ=0.2 show converging performance patterns with slight unsteadiness.

The performance patterns are certainly better than the case when λ=0 in the presence of disturbances. The plot with λ=0.2 has similar pattern as λ=0 but the biomass concentrations for all batches except batch 20 are much higher. Of all the discussed values in this plot, λ=0.3 exhibits steady and fast converging performance pattern with and without disturbance. More λ values were tested and the performance pattern were analysed.

Table 5.5: End-batch biomass concentrations (g/l) for ILC with PCR when λ is between 0.1 and 0.9

Batch	λ=0	λ=0.1	λ=0.2	λ=0.3	λ=0.4	λ=0.5	λ=0.6	λ=0.7	λ=0.8	λ=0.9
1	64.6	64.6	64.6	64.5	64.5	64.5	64.1	64.6	64.6	64.4
10	73.1	73.0	73.2	73.2	73.1	73.0	73.0	73.0	73.1	73.3
11	69.6	69.6	69.7	69.7	69.6	69.6	69.6	69.5	69.6	69.6
20	70.5	71.7	71.3	71.9	72.2	72.0	71.4	72.0	71.6	72.9

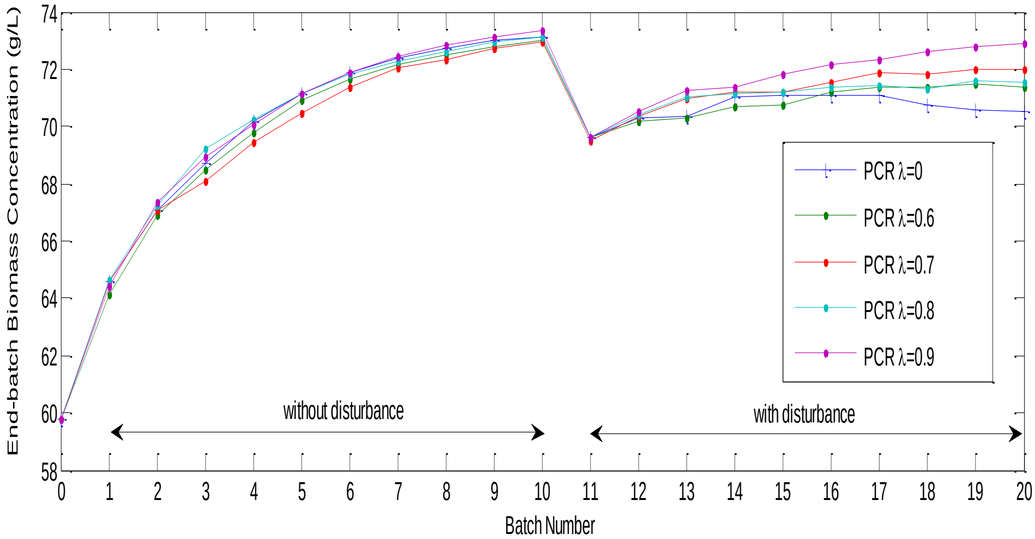


Figure 5.7: End-batch biomass concentration for ILC with PCR when λ is 0.6, 0.7, 0.8 and 0.9.

Referring to Figure 5.7, in this set of λ values, the end-batch biomass concentrations did not differ much in the first 10 batches where there is no disturbance. As seen in Table 5.5, the end batch biomass concentrations at the 10^{th} batch for the tested λ values are about 73g/l. The biomass concentration for λ=0.9 is the highest, which is 73.3g/l. In the presence of disturbance, the end-batch concentrations improved from batch to batch for all the λ values and had much steadier improving trend compared to the case when λ=0. Clearly λ=0.9 exhibits impressively steady and converging performance with and without disturbance compared to all the other λ values and λ=0. Refering to Table 5.5, it is to be noted that the final biomass concentration at the 20th batch for λ=0.9 is the highest, which is 72.9 g/l amongst all the tested λ values. It is almost 73 g/l, which means even in the presence of continuous disturbance, the system is able to track to the desired trajectory within 10 batches. The final biomass concentration at the 20th batch is almost as good as the 10th batch, which was without disturbance. A few other λ values were tested to study the performance pattern.

Table 5.6: End-batch biomass concentrations (g/l) for ILC with PCR for λ between1& 10

Batch	λ=0	λ=1	λ=2	λ=3	λ=3.5	λ=4	λ=5	λ=6	λ=7	λ=8	λ=9	λ=10
1	64.6	64.6	64.6	64.6	64.6	64.6	64.6	64.6	64.6	64.6	64.6	64.6
10	73.1	73.1	73.1	73.2	73.2	73.2	73.2	73.3	73.2	73.2	73.2	73.2
11	69.6	69.6	69.7	69.7	69.7	69.7	69.7	69.7	69.7	69.7	69.7	69.7
20	70.5	72.1	71.9	71.5	72.1	72	72	72.4	71.6	72	72	72.1

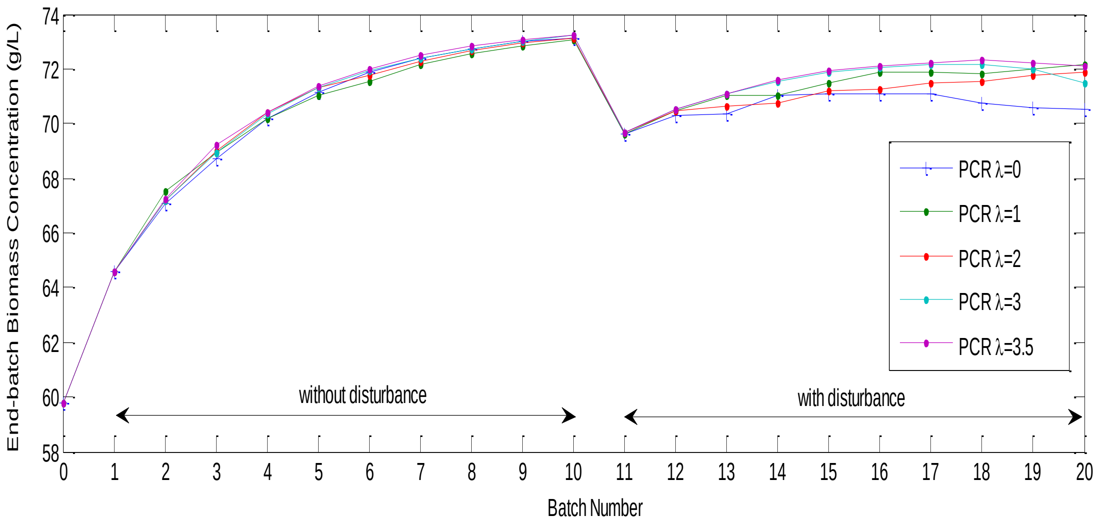


Figure 5.8: End-batch biomass concentrations for ILC with PCR when λ is 1, 2, 3 and 3.5.

With reference to Figure 5.8, increasing λ values to 1, 2, 3 and 3.5 did not differ much in the first ten batches without disturbances in comparison to λ=0. All the plots increased steadily and was converging towards the desired trajectory. Table 5.6 shows that, at the 10th batch, the end-batch biomass concentrations are the same or differs by 0.1 g/l for λ=1,2,3 or 3.5 when compared to λ=0. It is noticeable that the variation in plots with bigger λ values are smaller compared to the smaller λ values discussed in previous figures. This is because as the system reaches close to the desired trajectory, the Q and R ratio has less influence in the increment of the process yield. There is notable improvement in the batches with presence of disturbance for these λ values. Though there are slight instability either in the first few batches or towards the end of the 10 batches with disturbance, the final biomass concentrations are converging toards the desired value. Amongst these weighting parameter values, λ=3.5 exhibits the most favourable control performance. Under this weighting parameter value, the end batch concentrations were converging fast and steady with and without disturbance.

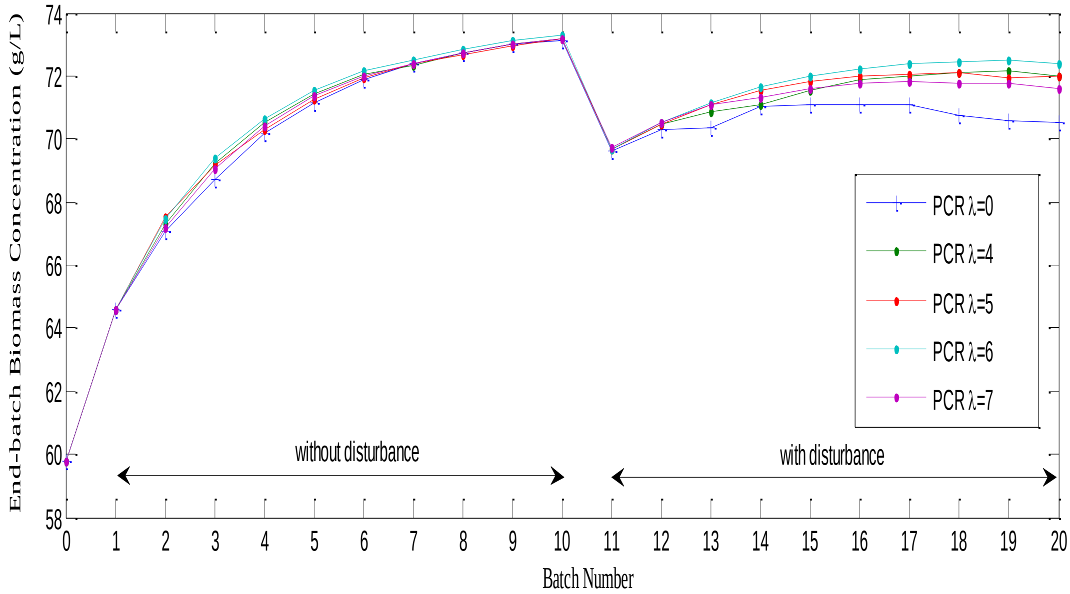


Figure 5.9: End-batch biomass concentrations for ILC with PCR when λ is 4, 5, 6 and 7.

With reference to Figure 5.9, increasing λ values to 4, 5, 6 and 7 produced improved biomass concentrations in the first ten batches without disturbances in comparison to λ=0. All the final biomass concentrations increase steadily and are converging towards the desired value. The end-batch biomass concentrations for all the first 10 batches of these λ values are higher than that of the λ=0 case. However, Table 5.6 shows at the 10^{th} batch, the end-batch biomass concentrations only differs by 0.1 g/l for λ=4, 5 and 7 and by 0.2g/l for λ=6 when compared to λ=0. The variation in final biomass concentrations with these λ values are even smaller compared to those discussed in Figure 5.8. There is significant improvement in the batches with presence of disturbance for these λ values. The instability has been reduced compared to that shown in Figure 5.8 and all the end batch concentrations are higher than that of the λ=0 case. In general, the final biomass concentrations are converging steadily towards the desired value for the 10 batches with disturbance. Amongst these values, λ=6 exhibits the most favourable performance. The end batch concentrations converge fast and steady with and without disturbance.

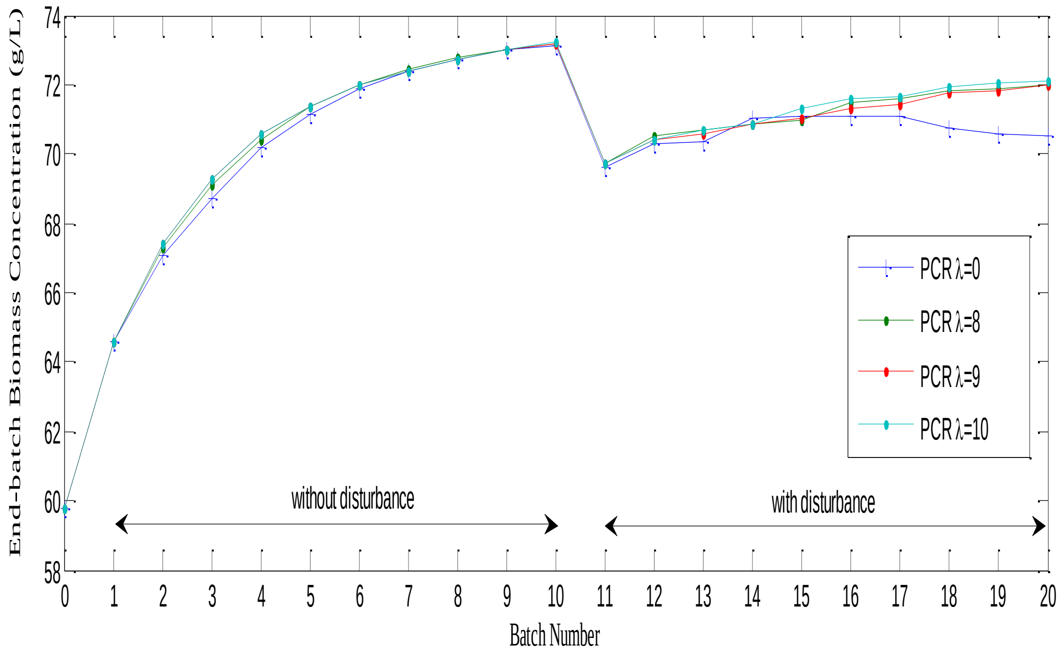


Figure 5.10: End-batch biomass concentration for ILC with PCR when λ is 8, 9 and 10.

Referring to Figure 5.10, further increment on λ values to 8, 9 and 10 exhibits improved biomass concentration for most of the first ten batches without disturbances compared to λ=0 plot. All the final biomass concentrations increase steadily and converge towards the desired value. The end-batch biomass concentrations for all the first 10 batches of these λ values are higher or similar to that of the λ=0 case. In the presence of disturbance, the plots are unstable but converging towards the desired trajectory for the 10 batches with disturbance. Amongst these values, λ=10 exhibits acceptable performance pattern. The end batch concentrations converges fast but with slight oscilations with disturbance. Since the performance pattern deteriorates with increasing λ values in PCR, bigger values were not attempted. Furthermore, since in the ILC with MLR cases, λ > 10 did not reveal imroving results, those values were not attempted for ILC with PCR model.

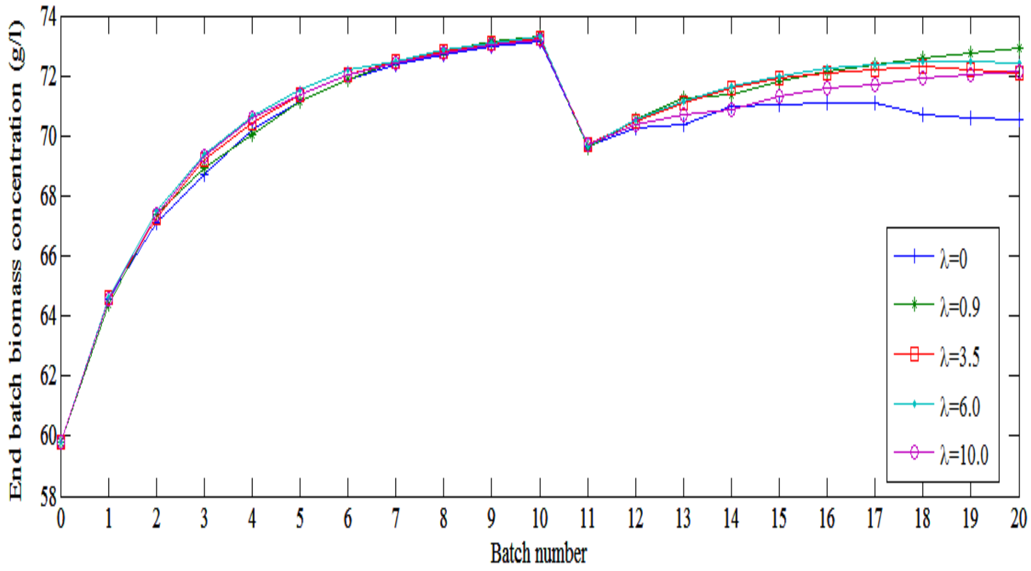


Figure 5.11: End-batch biomass concentrations for ILC with PCR model when λ is 0.9, 3.5, 6.0 and 10.

About 4 selected λ values from the figures above were collected and plotted for comparison in Figure 5.11. All the selected λ values revealed steadily converging results. There is no significant variation in the batches without disturbance. For the 10 batches with disturbances there are slight variations. The plot with λ=0.9 has the most steady and converging performance followed by λ=3.5 and 6 with more or less similar performance pattern. The plot with λ=10 has the slowest convergence rate amongst the compared λ values. It is noticed that smaller λ value is needed for PCR method to improve the model reliability.

In summary, incorporation of model prediction confidence bounds in the ILC objective function significantly improves the ILC performance, especially in the presence of disturbance. It is notable that with increasing λ values, the variations in the control performances are narrowed. Increment of λ>0.9 does not increase the final biomass concentrations. This is possibly because as the final biomass concentration gets closer to the desired value, smaller Q and R ratio is needed to maintain the performance. It comes to a point where changes in the Q and R ratio become insignificant in improving the performance pattern. The incorporation of the parameter in PCR model improves the model prediction reliability by further narrowing the confidence bounds. Therefore, higher biomass concentrations can be obtained. The results show that reliable ILC can effectively overcome the detrimental effect of model plant mismatches and unknown disturbances.

5.4.3 Results for ILC with partial least squares model

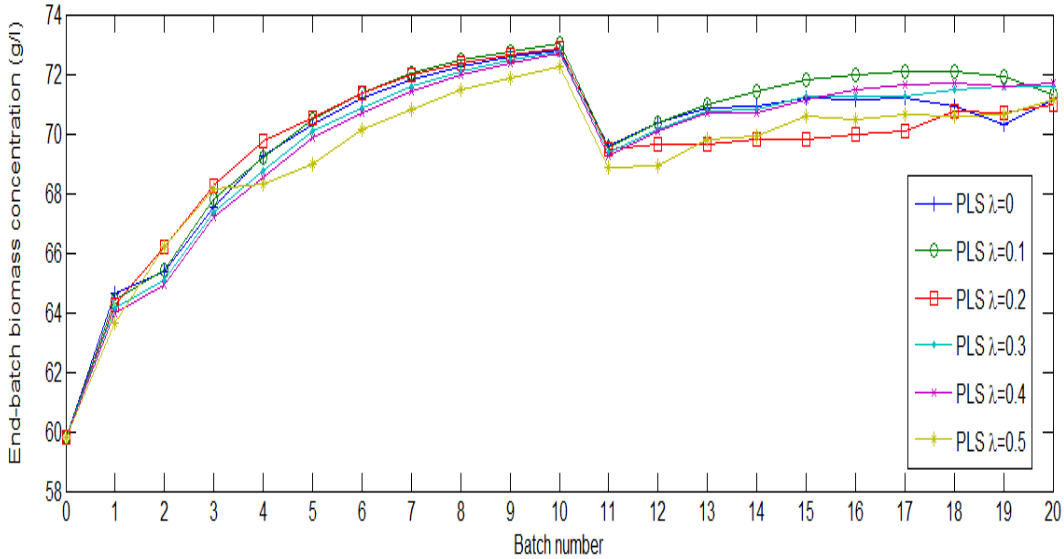


Figure 5.12: End-batch biomass concentrations for ILC with PLS model when λ is 0.1, 0.2, 0.3, 0.4 and 0.5.

With reference to Figure 5.12, incorporation of model prediction confidence bounds with λ values between 0.1 and 0.5 exhibit varying performance pattern. The plot for λ=0.5 shows the least biomass concentrations in most of the batches with and without disturbances. The plot for λ=0.2, shows very good convergence and stability for batches without disturbance but in the presence of disturbance the convergence rate mellowed down with slight instability towards the end of 20 batch trials. The plots of λ= 0.3 and 0.4 has similar performance pattern to λ=0. For batches without disturbance, the end-batch biomass concentrations were lower compared to the case when λ=0. For batches with disturbance, the biomass concentrations from batch to batch converge towards desired value with acceptable stability. The plot for λ=0.1 exhibit the most favourable performance for batches with disturbance. There is steady convergence compared to λ=0 except for slight drop at 20th batch which is still higher than that from λ=0. For batches without disturbances, the performance pattern and biomass concentrations for all the 10 batches are similar to that of the case when λ=0.

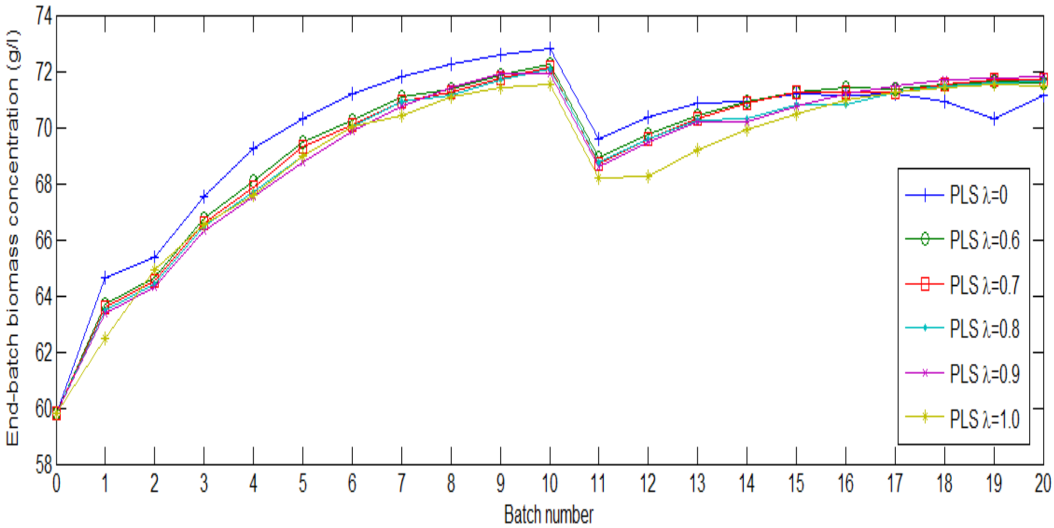


Figure 5.13: End-batch biomass concentrations for ILC with PLS model when λ is 0.6, 0.7, 0.8, 0.9 and 1.0.

Further increments in λ values, decreased the biomass concentrations for batches without disturbance as seen in Figure 5.13. In the presence of disturbance the performance stability and concentrations improved for most of the batches. This performance pattern certainly demonstrates that incorporation of model prediction confidence bounds improves reliability and stability in the presence of disturbance.

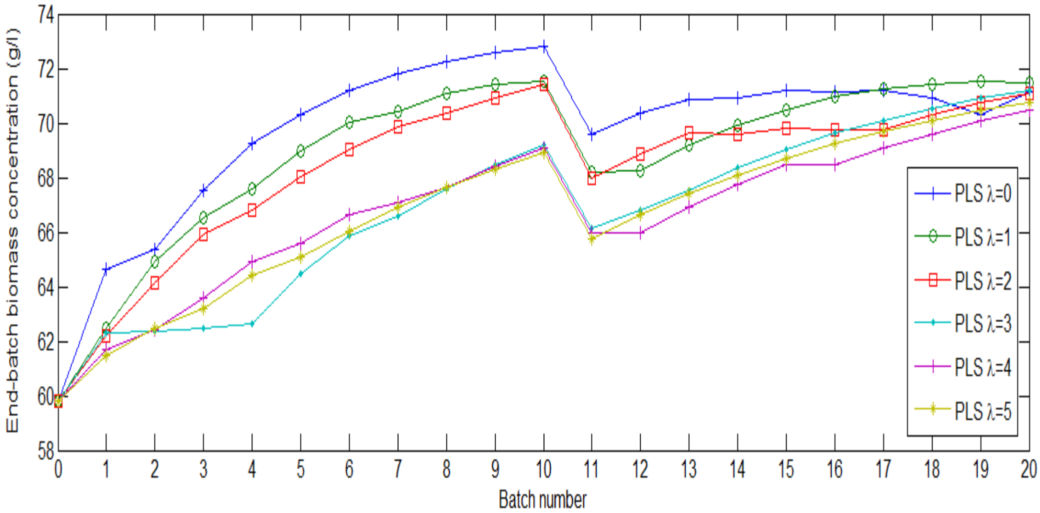


Figure 5.14: End-batch biomass concentrations for ILC with PLS model when λ is 1, 2, 3, 4 and 5.

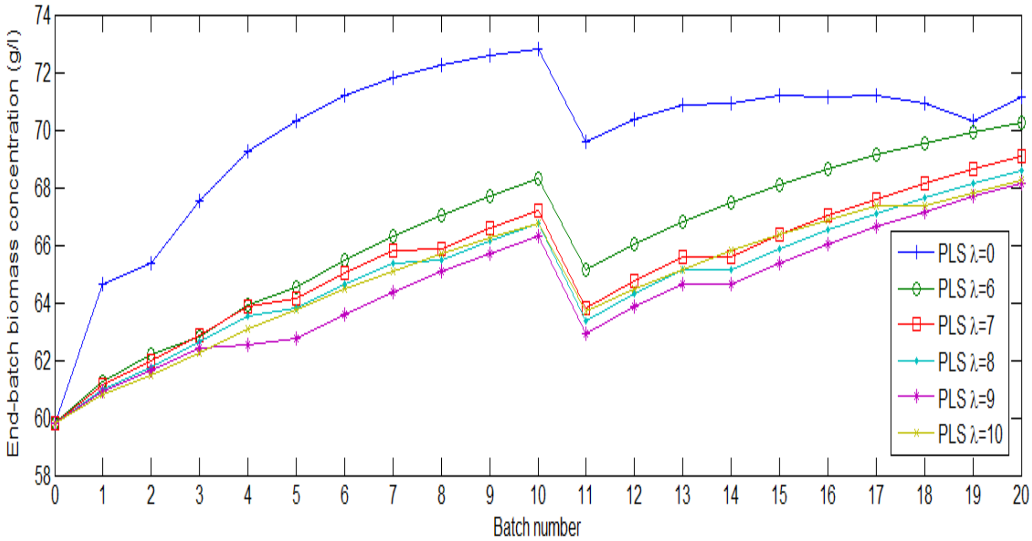


Figure 5.15: End-batch biomass concentrations for ILC with PLS model when λ is 6, 7, 8, 9 and 10.

Referring to Figures 5.14 and 5.15, it is clear that the convergence rate decreases with increasing λ value between 1 and 10. Though some of the λ values improved the stability of the control performance, due to slower convergence these values are not preferred.

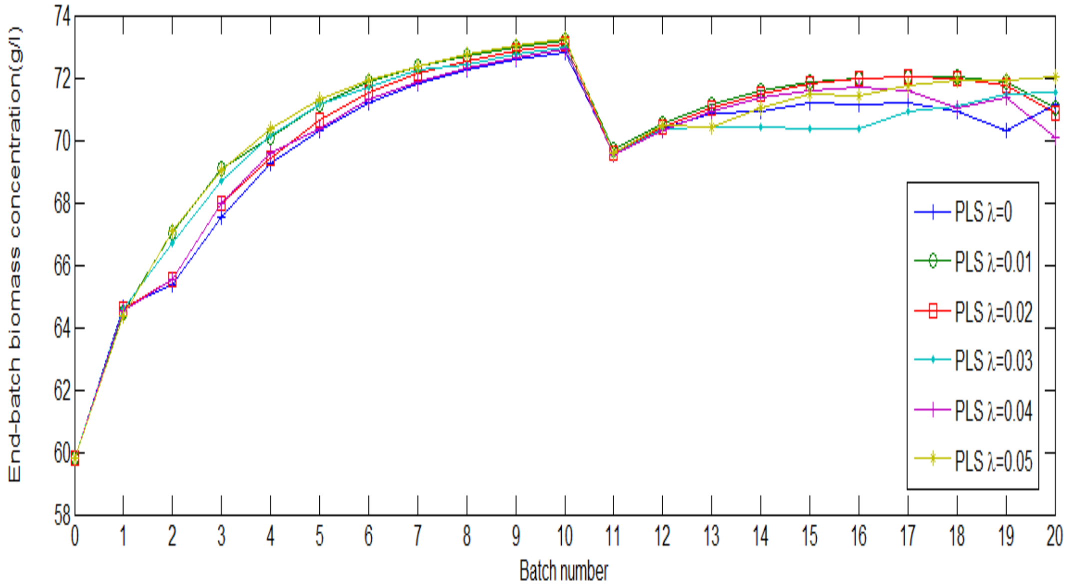


Figure 5.16: End-batch biomass concentrations for ILC with PLS model when λ is 0.01, 0.02, 0.03, 0.04, and 0.05.

Since λ values between 0.1 and 10 did not give satisfactory results, smaller values were tried to study the performance. It is possible that as the regression method becomes more complex, smaller modification is required to assert positive impact on the performance pattern. As seen in Figure 5.16, λ=0.05 reveals steady and reliable convergence for these tested λ values. The rest of the values performed reasonably well with and without disturbance but with slight instability and some with smaller convergence rate.

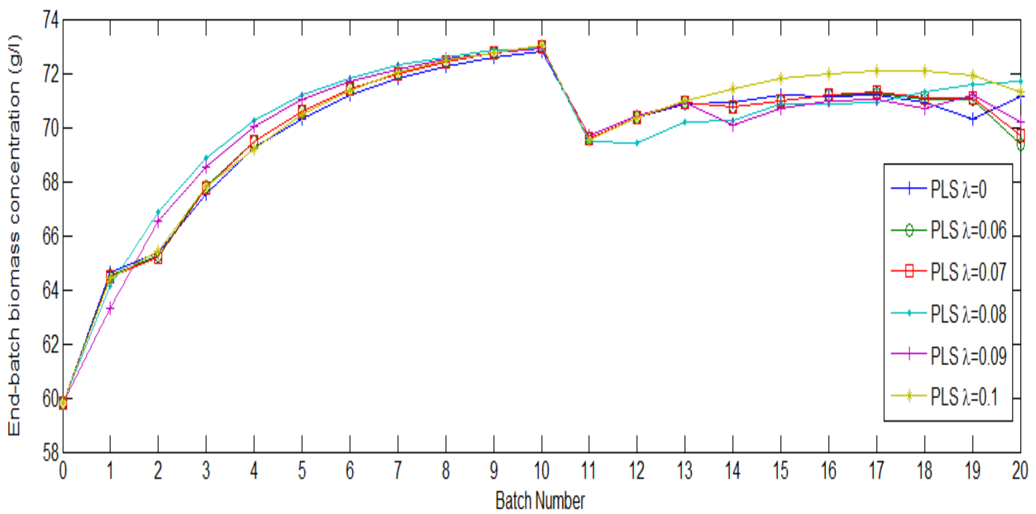


Figure 5.17: End-batch biomass concentrations for ILC with PLS model when λ is 0.06, 0.07, 0.08, 0.09 and 0.1

Referring to Figure 5.17, further increment in λ values between 0.06 and 0.1 exhibited favourable performance for batches without disturbance. In the presence of disturbance, most of the λ values had similar performance pattern to the case of λ=0. The plot for λ=0.08 revealed increasing biomass concentrations for almost all the batches with disturbance. There were slight fluctuations in the beginning but towards the end steadier performance was obtained. The performance when λ=0.1 is more favourable because the biomass concentrations are the highest for all the 10 batches with disturbance compared to the cases with other λ values except batch 20 which is slightly lower than λ=0.08.

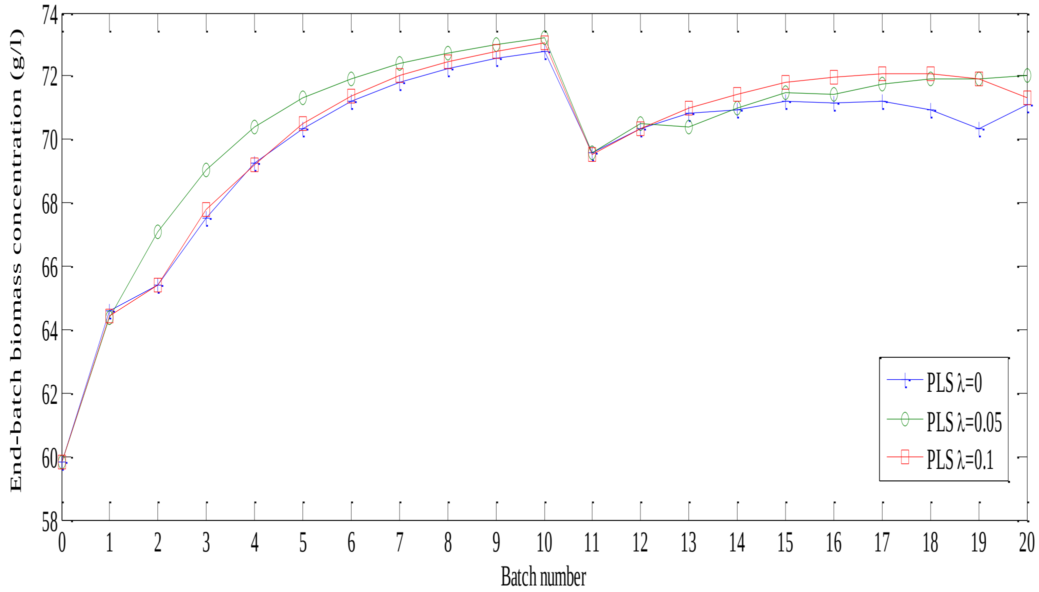


Figure 5.18: End-batch biomass concentration for ILC with PLS model comparing λ values 0.05 and 0.1

In Figure 5.18, 2 of the most favourable λ values were selected to be compared. The plot with λ=0.05 exhibits a more steady and reliable performance without disturbance. In the presence of disturbance the convergence improved from batch to batch with slight instability in the beginning. The plot with λ=0.1 exhibits very similar pattern to that with λ=0 but in the presence of disturbance, there are more steady and converging biomass concentrations from batch to batch for 8 batches with disturbance. The last 2 batches were dropping in performance. For batch to batch improvement, λ=0.05 will be more appropriate to apprehend model plant mismatches and unknown disturbances. The slight instability has to be addressed. In summary, incorporation of model prediction confidence bounds to PLS model improve the reliability but with slight stability. Smaller λ value is required for PLS method.

5.5 Conclusions

The effect of incorporation of model prediction confidence bounds into the objective function were studied in this chapter. Various λ values were introduced to evaluate the improvement in ILC with MLR, PCR and PLS models. The additional parameter shows significant improvement in ILC with MLR models. The ILC with MLR models did not exhibit good results with no modification in the objective function. With the addition of model prediction confidence bounds parameter, the method tends to reveal desirable results. The results are as good as the ones with PCR when λ=0. As for PCR and PLS, the modified objective function tends to produce significant improvement for batches with disturbance. The batch to batch performances are more stable and reliable when there were disturbances. Amongst the tested λ values, some performed well without disturbance and the performance deteriorates in the presence of disturbance and vice versa. The most effective λ values for ILC with MLR, PCR and PLS for batches with and without disturbance were identified.

CHAPTER 6: ITERATIVE LEARNING CONTROL WITH UPDATED LINEARISED MODELS FROM A SLIDING WINDOW OF HISORICAL BATCHES AND MEAN NOMINAL TRAJECTORIES

6.1 Introduction

In Chapters 4 and 5, batch-to-batch control study using linearised models from updated historical batches was discussed. The number of historical batches used to develop a current batch process model keeps building up after every batch run. In other words, after every batch run, the obtained data is added into the pile of historical batches. Then all previous batches are used to identify a new process model which is used to generate a new control policy for the current batch. The cycle repeats and the process model is developed using both old and new batch data.

The linearised process model development plays an important role in an ILC control method. It defines the learning rate which then influences the generation of the control policy to enhance convergence of a particular process. Since the linearised model development is solely dependent on the available historical batches, a question arises as to how many batches are really needed for a model development in an ILC control method. The perception is smaller number of control intervals uses smaller number of historical data to develop corresponding process model (Vilas et al., 2004). It is a norm to use historical batches more than the batch intervals to develop a reliable process model (Lee et al., 1999). Vilas et al. (2004) has studied on manipulating number of control intervals in a batch with growing number of historical data. In the presence of limited number of historical batches, the number of intervals in the current batch run is reduced in accordance to the historical batches available. For instance, the study started with 2 control intervals because there were only 2 historical batches available to develop the process model. The process model in Vilas et al., 2004 was developed from perturbation variables. The difference of input and output variables in two adjacent batches were used as the input and output data. After every batch run, the latest data is added into the historical batches pool. As the sample size increases after every batch run, the control intervals size was increased by one. Increased number of control intervals improves control performance due to increased degree of freedom in the control actions. After 9 batch runs, the desired control intervals of 10 is achieved. The following batch runs were run with 10 control intervals. At this juncture onwards, all the existing batches were used to identify the process model. It is a common approach to have 10 batch intervals and more than 10 historical batches (normally about 30) to develop a process model (Lee et al., 1999).

As for batch to batch updated linearised process model, a hypothesis was formed that it is possible that fewer numbers of the most recent historical batches is sufficient to improve batch to batch control performance. Not necessarily all the historical batches need to be used to develop the linearised model. A selected number of historical batches may be sufficient. In order for the updated model to capture the process behavior in the face of process variations, a new technique using a moving window of the historical batches to update batch-wise linearised models is developed in this chapter. The historical batches were updated after every batch run but only the *M* recent number of batches was kept. In other words, after every run the "oldest" batch is forgotten and the new batch is included into the sliding "window" of historical batches. In addition to that, identifying nominal process variable trajectories is essential for real-time process control and monitoring application (Dadebo and McCauley, 1995). Therefore, selecting an appropriate reference trajectory is important in non-linear process optimizing control. In this chapter, a method of ILC with updated linearised model and updated nominal trajectories is proposed. The updated nominal trajectories are obtained by averaging trajectories from the immediate previous *A* batches and the updated linearised model is identified using a sliding window of the M immediate previous historical batches.
The proposed strategies were applied to a simulated fed-batch fermentation process. The updated process models were developed using MLR, PLS and PCR. Different window sizes and averaged reference trajectories were studied and the performances were evaluated. The results show that the proposed strategy can enhance the control performance.

The chapter is organized as follows. Section 6.2 describes the linearised model updating strategy using sliding window approach. There were two moving windows in the method: one is the window of historical batches and the other is the window of nominal trajectory. Section 6.3 presents the results for batch to batch ILC with updated linearised models using historical batch moving window approach. In this section the window for nominal trajectories were fixed as the immediate previous batch data. The method was applied to MLR, PLS and PCR models. Section 6.4 presents the results for averaged nominal trajectories without sliding window approach for historical batches. The sliding window was only used for the averaged trajectory identification. A number of selected latest historical batches were averaged to identify the nominal trajectory. Section 6.5 illustrates the outcome of reliable batch to batch ILC using linearised MLR model with sliding window approach. Section 6.6 concludes the chapter.

6.2 Model updating using sliding window approach

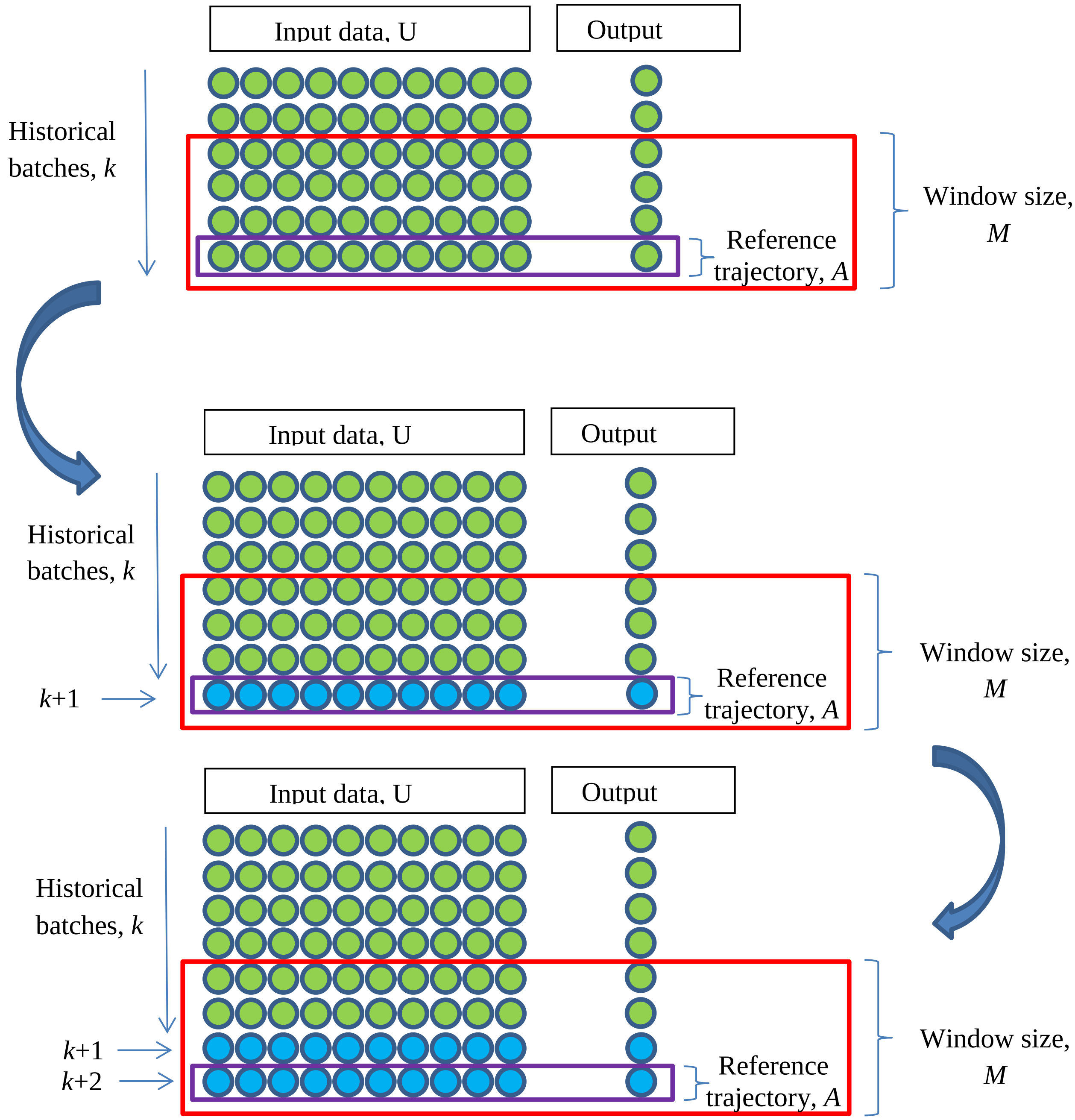


Figure 6.1: Model updating using a sliding window approach

In the sliding 'window' historical batches method, a fixed window of historical batches (*M*) for model generation is assigned as shown in Figure 6.1. In the figure, *k* represents the batch index (and the number of batches) in the original historical batches while *k + 1* and *k+2* represents the first and second batches, respectively, appended to the original historical batches. From the *k* historical batches, *M* number of batches is selected to develop the linearised model. After the first run with ILC method using sliding window approach, data from batch *k+1* is added into the historical batch database. To develop the control policy for the next batch, the sliding window, *M*, moves down including the latest previous batch, the (*k+1*)th batch, and the process repeats for the next batch run. After every batch run, the 'oldest' batch is forgotten and *M* number of batches including the immediate previous batch is used as historical batches. As a result, the 'window' keeps sliding forward with fixed number of batches adopting new batch runs data and dropping the older ones. The advantage is the updating historical batch data components provide a more up-to-date condition of the process. Therefore, sustaining steady improvements and achieving highest possible yield is made feasible.

There are two windows in the diagram. One is the historical batch window, *M* and the other is the updated nominal trajectory window, *A*. As seen in Figure 6.1, *A*, represents the number of batches used in calculating the average nominal trajectory. The window for *A* also slides down as the batch run progresses. The reference trajectory is either the latest batch (*A*=1) or the average of the latest *A* number of batches in the moving window of historical batches. Using average trajectories could remove the influence of measurement noise and random disturbance on the nominal trajectories.

Let *M* be the size of a sliding window of the past batches and use the immediate previous batch, the $(k\text{-}1)^{\text{th}}$ batch, as the nominal batch (i.e. *A*=1), then the deviations of the process input and output trajectories from their nominal trajectories in the sliding window can be represented as:

$$\Delta\mathbf{X}_{k=1} = \begin{bmatrix} \mathbf{U}_{k-M} - \mathbf{U}_{k=1} \\ \mathbf{U}_{k-M+1} - \mathbf{U}_{k=1} \\ \cdots \\ \mathbf{U}_{k-1} - \mathbf{U}_{k=1} \end{bmatrix} \tag{6.1}$$

$$\Delta\mathbf{Y}_{k=1} = \begin{bmatrix} \mathbf{Y}_{k-M} - \mathbf{Y}_{k=1} \\ \mathbf{Y}_{k-M+1} - \mathbf{Y}_{k=1} \\ \cdots \\ \mathbf{Y}_{k-1} - \mathbf{Y}_{k=1} \end{bmatrix} \tag{6.2}$$

The updated model parameters can be obtained using MLR, PCR or PLS. If correlations exist among the control actions at different stages of a batch, then PLS and PCR will give robust and reliable estimation of the model parameters.

6.3 Batch to batch ILC with updated models and moving window of historical batches

Further improvement from the results in Chapter 4, Figure 4.12, was done for PCR and PLS model using a sliding window of *M* historical batches to develop process models. As this section only investigates the effect of sliding window sizes, the number of batches for calculating the average nominal trajectories is fixed to 1 (*A*=1). In other words, the nominal trajectory for every batch run is the data from immediate previous batch. The window for *A* moves one batch down after every trial.

As for the *M*, after each batch run, the new batch data is added into the window of historical batches. The oldest batch in the window is removed. The idea is to use latest information to update the model and calculate the control policy for the current batch. Three sliding window sizes of 10, 15 and 20 historical batches were studied and the results are presented. The historical batches window size, *M* must be at least 10 because there are 10 piecewise-constant input control policy used in this fed-batch fermentation process. Other window sizes up to 50 were tried before these three were selected to be discussed. Window sizes 25, 30, 35, 40, 45 and 50 did not exhibit any better results compared to the results using window size of 20.

The batch to batch updated model using sliding window approach was mainly devised for PCR and PLS regression models. This is due to the fact that these regression methods are more advanced and possibly able to capture the necessary information using fewer but recent batch data. The MLR model always requires bigger number of batch data. The best method would be to use as many historical batch data as possible to obtain a reliable prediction. However, in curiosity to investigate the respond of MLR method for the suggested method, the results using MLR model for the chosen window sizes is also presented in this section. Some interesting results were obtained.

6.3.1 ILC using models identified from sliding window and MLR

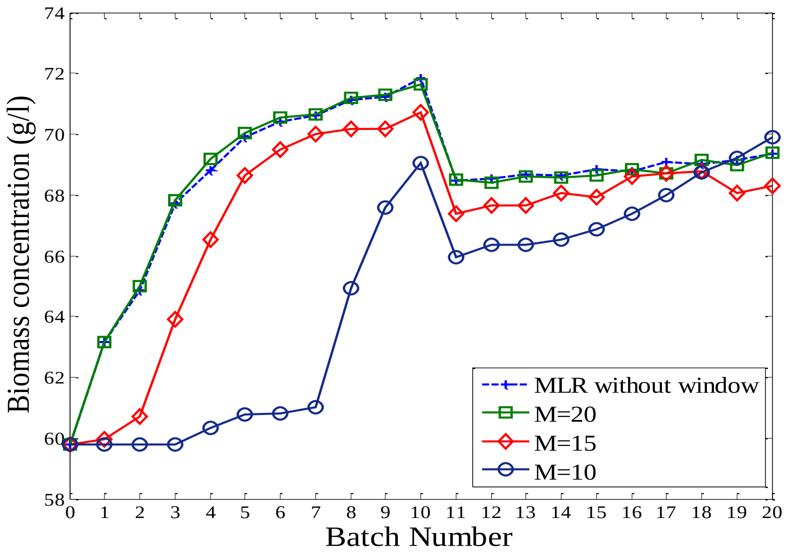


Figure 6.2: End of batch biomass concentration under ILC with batch-wise updated MLR models using a sliding window approach

Figure 6.2 exhibits the end batch biomass concentration for fed batch fermentation using batch to batch updated linearised MLR model using sliding window approach. Performances of different window sizes (*M*) were compared with the one without using sliding window which was presented in Chapter 4 (Figure 4.12). It is clearly noticeable that for the three different window sizes, varying performances are portrayed. None are better than the one without sliding window plot but certainly there is an improving pattern from *M*=10 to *M*=15 and then to *M*=20. The plot of *M*=20 gives very similar readings for almost all the tested batches for both with and without disturbances when compared to the plot without sliding window. The 20th batch biomass concentration is 69.37 g/l for the plot without sliding window application and 69.38 g/l for plot with *M*=20. The 10th batch biomass concentration of the former plot is 71.85 g/l and the latter is 71.64 g/l. It is interesting to note that 20, recently updated historical batches are sufficient to produce a result equivalent to increasing historical batch numbers, which was between 20 to 40 batches altogether. For window size of 15, the performance pattern without disturbance is similar to plot of *M*=20 but with much lower convergence rate. In the presence of disturbance, there were fluctuating increments from batch to batch and the convergence rate was still less. The performance pattern of window size 10 was a little interesting. In the first 7 batches without disturbances, the batch to batch biomass increment was very small. The sliding window approach did not have much influence on the biomass concentration improvement. However, there were significant improvements in biomass concentrations from 8th to 10th batch though the performance were lesser in comparison to *M*=15 and *M*=20. In the presence of disturbance, the biomass concentrations were steadily converging from batch to batch and the 20th batch finished better than all the other plots used for comparison. It can be deduced from the performance plot that, for *M*=10, when 70% and above of the batches are recent batch data with improving pattern, the sliding window approach is able to improve the batch to batch performance steadily for both with and without disturbances.

6.3.2 ILC using models identified from sliding window and PCR

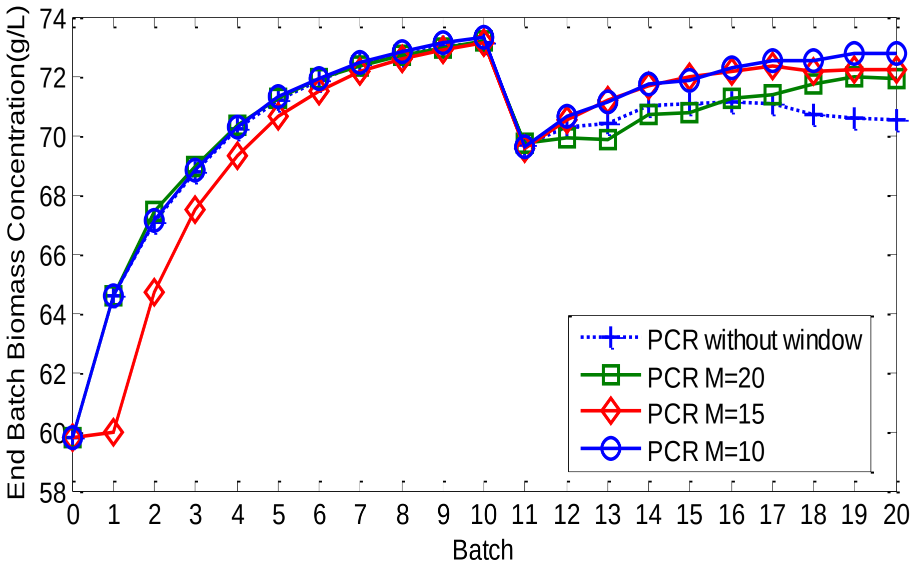


Figure 6.3: End of batch biomass concentration under ILC with batch-wise updated PCR models using a sliding window approach

Figure 6.3 shows that all three windows sizes exhibit improving results with varying stability before and after the disturbance was introduced. Performances of different window sizes (*M*) were compared with the one without using sliding window which was presented in Chapter 4 (Figure 4.12). From batch 1 to batch 10, when there is no disturbance, all the three window sizes showed satisfactory convergence rate and stability. Within the ten batches, the performances for M=20 and M=10 are very similar to that of ILC with PCR model without sliding window which is the desired performance pattern. The performance of M=15 is slightly lesser in the first 5 batches but then matched the ILC without window performance curve in the following 5 batches. The biomass concentrations for the 10th batch for window sizes 20, 15 and 10 are 73.18g/L, 73.12 g/L and 73.32 g/L respectively. Since, even without sliding window, the ILC with PCR model revealed very satisfactory performance pattern for batches with no disturbance, there is not much room for further improvements with the introduction of sliding window technique.

As for batches with disturbance, the unstable performance of batch-to-batch control without sliding window provided the possibility for further improvement. The effectiveness of the sliding window technique is noticed from batches 11 to 20. ILC with the entire three window sizes exhibit improving convergence rate and stability when disturbance is introduced in comparison to the performance of ILC with PCR models identified from the entire historical batches. For window size 20, the biomass concentrations were fluctuating though the trend was improving from batch to batch. The biomass concentrations in the last 4 batches were still higher than that without using sliding window. The batch to batch control performance for M=15 shows satisfactory convergence and stability from batches 11 to 17 but failed to sustain the good performance in the following three batches. However, the end batch biomass concentrations for all the ten batches were still higher than that without using sliding window. Performance trend of window size 10 in the presence of disturbance is as good as the ones without disturbance. The convergence rate and stability is very satisfactory. There is distinct improvement in the batch to batch control by using window size 10 when compared to that without using sliding window. The performance of batch-to-batch ILC improves with reducing window size. The final biomass concentrations for the 20th batch for window sizes 20, 15 and 10 are 71.88g/L, 72.20 g/L and 72.76 g/L respectively. As for ILC without sliding window, the final biomass concentration at the 20th batch is 70.53g/L. The PCR model was able to attain final output (20th batch) almost as good as without disturbance (10th batch, 73.13 g/L) within 10 batches. Amongst the three window sizes, window size of 10 gave the most stable and fastest converging performance. It is shown in the results that PCR method does not need a growing number of historical batches to develop a reliable model. An updated historical batch data with window size equal to the number of control policies used in the fed-batch fermentation process is able to generate optimal process model by using the PCR method.

6.3.3 ILC using models identified from sliding window and PLS

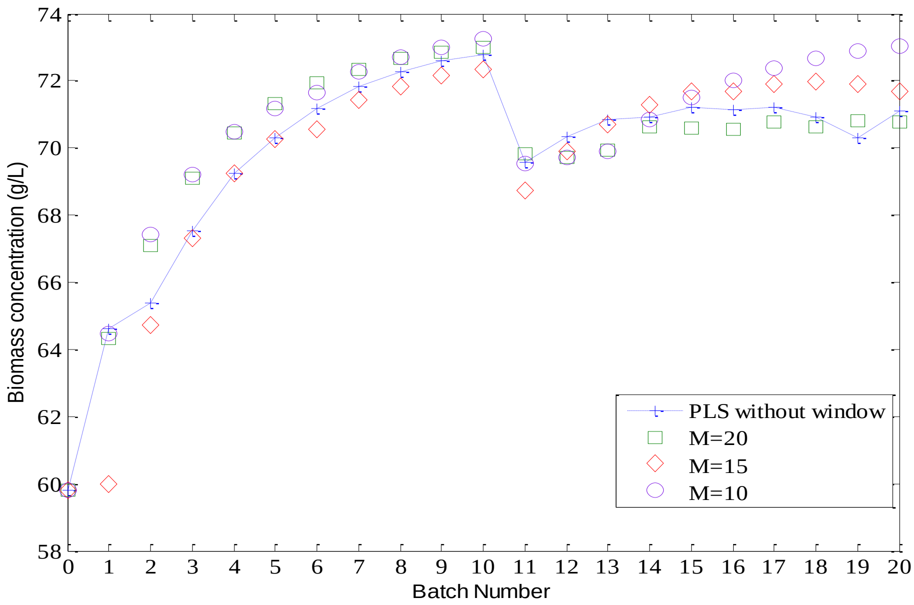


Figure 6.4: End of batch biomass concentration under ILC with batch-wise updated PLS models using a sliding window approach

Figure 6.4 shows that all three windows sizes exhibit improving results with varying stability before and after the disturbance was introduced. Performances of different window sizes (*M*) were compared with the one without using sliding window which was presented in Chapter 4 (Figure 4.12). The end batch biomass concentration without sliding window is 72.78g/L at the 10th batch and 71.11 at the 20th batch. For window size of 20, the convergence rate and stability were better and satisfactory when there is no disturbance. The 10th batch final biomass concentration is 73.0 g/l. In the presence of disturbance, the results for window size 20 fluctuated. The biomass concentrations for most of the batches were lower than the concentrations for ILC without using a sliding window. The 20th batch end-point biomass concentration was 70.76 g/l. As for window size of 15, the biomass concentrations were lesser for most of the batches with no disturbances but in the presence of disturbance, the convergence rate were improving steadily from batch 11 to batch 15. From batch 16 to batch 20 slight fluctuations were noticed though the biomass concentrations were higher than the plot with no window. The biomass concentration for the 10th batch is 72.33 g/l and 20th batch is 71.67 g/l. For window size of 10, the performance with no disturbance is as good as window size of 20, in fact the best amongst the three windows. The 10th batch biomass concentration is 73.25 g/l. For batch 12 and 13 the biomass yield was lesser than that under ILC without using sliding window, but in the following 7 batches the performance improved steadily. The convergence rate was very satisfactory. The PLS model was able to attain final output almost as good as without disturbance within 10 batches which is 73.05 g/l. Amongst the three window sizes, window size of 10 gave the most stable and fastest converging performance. It is shown in the results that PLS method does not need a growing number of historical batches to develop a reliable model. An updated historical batch data with window size equal to the number of control policies used in the fed-batch fermentation process is able to generate optimal process model by using the PLS method.

6.3.4 Summary

In summary, application results show that ILC based on batch-wise updated model using a sliding window of recent historical batches improves the control performance with and without disturbance for ILC using PLS or PCR models. It is shown that model updating using PCR or PLS does not need large window size in providing enhanced control performance. As for ILC with MLR model, a similar performance to batch to batch updated linearised ILC without window size can be achieved using a reasonable window size. It is not necessary to use all the available or growing number of historical batches to achieve the similar performance. This approach reduces the data handling load. The PCR and PLS being able to address the colinearity issue need fewer historical batches compared to the MLR method. In addition to that, further improved steady and stable performance is achieved for PCR and PLS models in comparison to the results discussed in Chapter 4, Figure 4.12 (Case 2 results) for batch to batch updated linearised model without sliding window approach as seen in Figure 6.5 below.

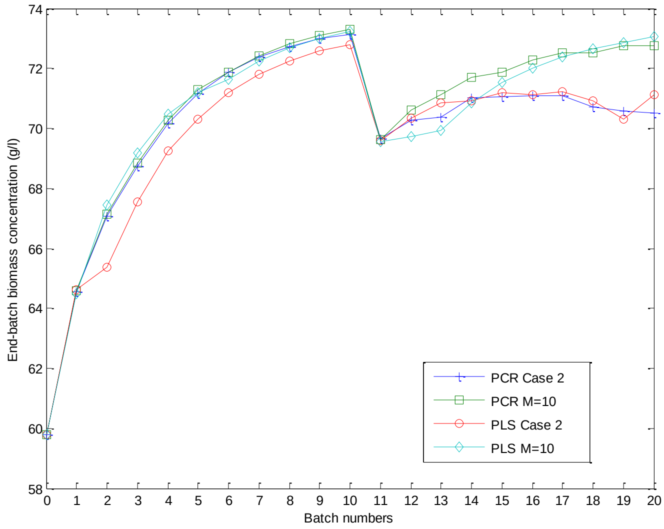


Figure 6.5: Performance of ILC using PLS and PCR models with and without sliding window approach

The impact of sliding window approach is more evident for batch to batch ILC with disturbance as can be seen from Figure 6.5. In the presence of disturbance there is significant batch to batch improvement in biomass concentrations for plots using sliding window approach than the ones without it. The batch to batch ILC with PCR model exhibits better performance compared to that with PLS model in the presence of disturbance in the sliding window approach. When, there is no diturbance, both control methods give similar stability and convergence towards desired trajectory.

6.4 Batch to batch updated model using averaged nominal trajectory with growing historical batches.

In this section, the effect of using average nominal trajectories (obtained from *A* number of batches) is studied. The number of historical batches kept increasing and is updated after every batch run. In other word, no sliding window was used for historical batches. The nominal trajectories were set to be the average of the latest 5, 4, 3 or 2 historical batches data. The number of batches to be averaged, *A*, is kept at 5 or below so that the variations among these batches will be kept small, as the purpose of using average nominal trajectories is mainly to eliminated the influence of measurement noise and random process variations.

The improvement on the product concentration of the simulated fed batch fermentation process is evaluated. The final product concentration for batch to batch updated model using ILC control method presented in Chapter 4, Figure 4.12 (case 2 plots) is also plotted to compare the performance of the averaged nominal trajectories approach. The plot is labelled as PCR Case 2, PLS Case 2 or MLR Case 2. For these plots the *A*=1, which means the immediate previous batch data is used as the nominal trajectories and the historical batch data was updated after every batch run and kept increasing.

6.4.1 Performance of ILC using MLR model

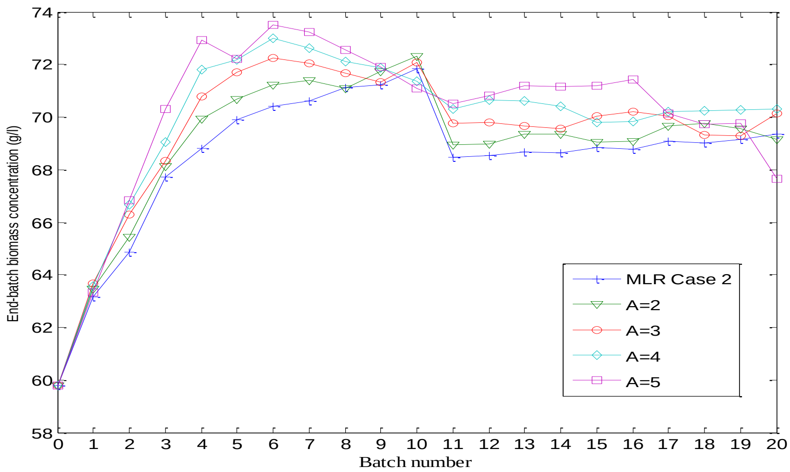


Figure 6.6: Batch to batch control performance using MLR models with averaged nominal trajectories (*A*=2 to 5)

With reference to Figure 6.6, for plots with nominal trajectories averaged from *A*= 2, 3 and 4 latest batches, the final product concentrations are certainly improved for most of the tested batches compared to the MLR Case 2 performance plot. However, the results are significantly unstable with and without disturbances as the averaged batch numbers, *A*, increases. This is possibly because the MLR model is not able to handle the noise accumulation from the averaged batches. The MLR model did not correspond well to the averaged nominal trajectory approach.

6.4.2 Performance of ILC using PCR model

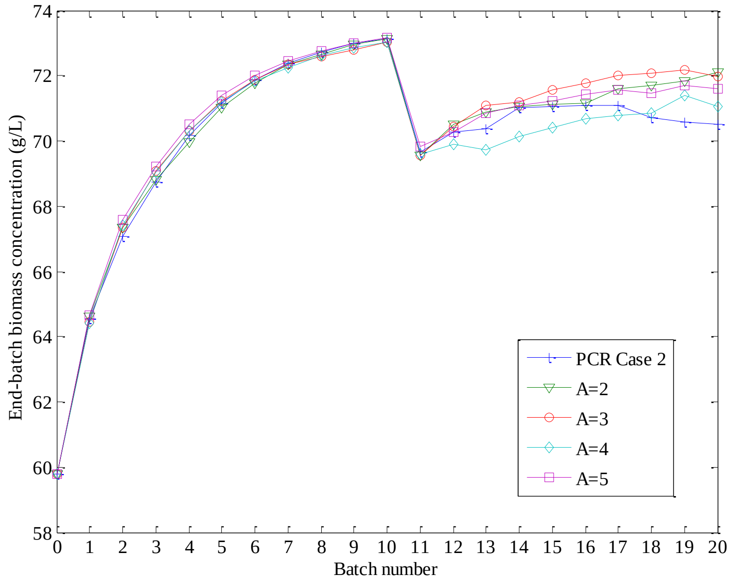


Figure 6.7: Batch to batch control performance using PCR models with averaged nominal trajectories (*A*=2 to 5)

With reference to Figure 6.7, the averaged nominal trajectories improve batch to batch performance for most of the tested batches. The PCR Case 2 plot is used as comparison to evaluate improvement in the batch to batch ILC with updated PCR model when using averaged nominal trajectories. When there is no disturbance, the batch to batch performance pattern for *A*=2 to 5 is has either slightly improved or very close to PCR Case 2. The batch to batch improvement was stable for all the tested A values for trials without disturbance. In the presence of disturbance, though there are slight fluctuations, overall there have been improvement from batch to batch for *A*=2, 3, 4, and 5. This could be because the average of *A*=2 to 5 historical batches is close to the true nominal trajectories as the influence of measurement noise is reduced. Except for *A*=4, the rest of the plots (*A*= 2, 3 and 5) exhibits higher biomass concentration for all the batches with disturbance in comparison to PCR Case 2 plot. It is notable that averaged nominal trajectory can be used to improve batch to batch updated model ILC performance but the instability has to be addressed. The moving window method with latest batch as nominal trajectory as discussed in Section 6.3.2 has a better performance than the averaged trajectory approach for PCR model.

6.4.3 Performance of ILC using PLS model

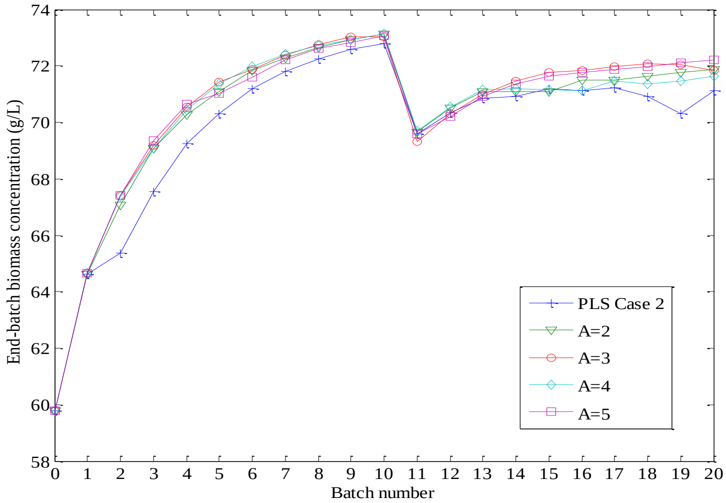


Figure 6.8: Batch to batch control performance using PLS models with averaged nominal trajectories (*A*=2 to 5)

With reference to Figure 6.8, it is noted that batch to batch updated models using averaged historical batches to calculate nominal trajectories reveals interesting results with PLS model. The PLS Case 2 result is used for comparison purpose. Overall, the performance patterns for averaged nominal trajectories are more stable and faster converging towards desired trajectory compared to the PLS Case 2 plot. There were significant batch to batch biomass concentration improvement for *A* = 2 to 5 for trials without disturbance compared to the PLS Case 2 plot. In the presence of disturbance, the plot for *A*=3 and 5 deliver a steadily improving batch to batch performance. The plot for *A*=3 shows slight drop in the 20th batch but still higher than 20th batch in PCR Case 2 plot. The plots for *A*=2 and *A*=4 were little unstable in the presence of disturbance but still produced batch to batch performance better the PCR Case 2 plot. The performance for *A*=3 and *A*=5 was better than the *A*=2 and *A*=4 ones.

Although the batch to batch increment is not as much as the moving windows approach as discussed in Section 6.3.3, the stability of the trend is worthy a compliment. The graph suggests that averaging latest 3 or 5 batches enhances stability of the control method.

6.4.4 Summary

In summary, averaging nominal trajectories for PCR and PLS have an impact on improving the stability of the performance compared to the batch to batch updated linearised model as discussed in Chapter 4, Figure 4.12 (Case 2 plots). However, the results were not better than the sliding window historical batch approach as discussed in Section 6.3. Therefore, the sliding window approach with latest batch as the nominal trajectory is sufficient to enact positive impact for PCR and PLS model based batch to batch ILC. As for MLR, the averaged trajectory approach did improve the biomass concentration for almost all the batches using ILC compared to the non-averaged batch to batch updated trajectory method but portrayed unsatisfactory steadiness in batch to batch performance. The results are certainly better than the sliding window approach in terms of end batch biomass concentration. The unsteadiness suggests that further improvements on the MLR model for averaged trajectory approach may present favourable results.

6.5 Reliable ILC using MLR models with averaged nominal trajectories and sliding window approach

In order to further exploit the potential of averaged trajectory approach on the MLR model as discussed in Section 6.4.1, an interesting technique combination approach was attempted. A reliable ILC method through incorporation of model prediction confidence bounds (MPCB) into iterative law development using MLR model rendered significantly improved stability and convergence rate as discussed in Chapter 5, Figure 5.5. Average nominal trajectories are incorporated into the reliable ILC method. These techniques were tested using sliding window size, M=20 instead of using growing number of historical batches. This is because as seen in Figure 6.4, the sliding window size, M=20 gives very similar performance to the batch to batch updated model ILC without sliding window approach. The combination approach was tested for λ value 3.

6.5.1 The case with *M*=20 and λ= 3

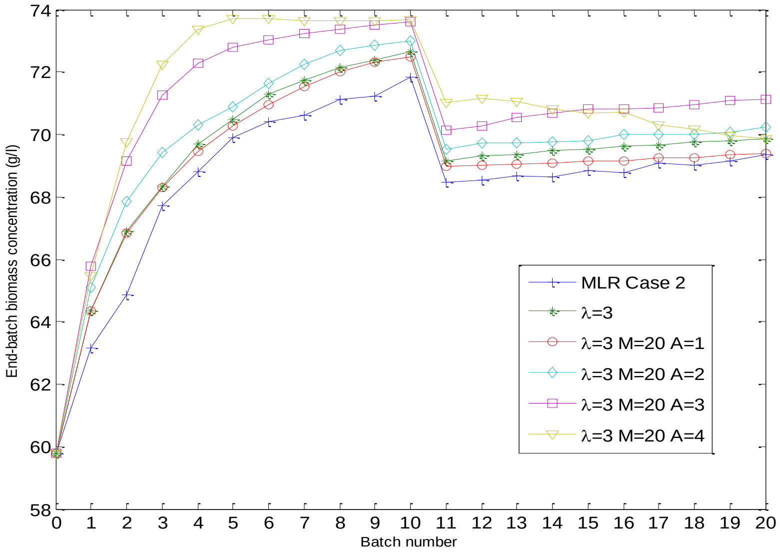


Figure 6.9: ILC control performance with *M*=20, λ= 3, and various values of *A*

Figure 6.9 exhibits the results for averaged reference trajectory, *A*=1, 2, 3 and 4 for sliding window *M*=20 with incorporation of model prediction confidence bounds and the penalty parameter is λ=3. Performance of the combination approach is compared with results from Chapter 4, Figure 4.12 and Chapter 5, Figure 5.5. The plot for MLR case 2 is without sliding window and MPCB approach as that was presented in Chapter 4 while the plot with λ=3 is from Chapter 5 with only MPCB technique applied and without sliding window approach.

It is evident that all the batches with combined approach exhibits better stability and convergence rate, with and without disturbance, compared to MLR Case 2. The plots stability of the batch to batch ILC with sliding window historical batches and averaged reference trajectory has improved significantly with the incorporation of model prediction confidence bounds. When compared to Figure 6.8, the improvement on both stability and steady convergence towards desired trajectory in Figure 6.9 is very evident.

Figure 6.9 shows that with increasing number of batches to be averaged, *A*, from 1 to 3, the batch to batch stability and convergence rates were increasing with and without disturbance. The plot for *A*=4 achieved the asymptotic trajectory to the desired trajectory within 5 batches without disturbance and was the best performing curve amongst the rest of the plots. In the presence of disturbance, the batch to batch biomass concentration for

A=4 decreased for all the 10 batch runs. Although the performance pattern deteriorated, the end-batch biomass concentrations for all the 10 batches were higher than MLR Case 2 plot.

The plot for *A*=1, *M*=20 and λ=3 exhibits better performance in terms of stability and batch to batch improvement compared to the MLR Case 2 plot. However, the plot for λ=3 has even better performance pattern than the two plots mentioned above. The plot for *A*=2, *M*=20 and λ=3 exhibits much higher batch to batch biomass concentrations than the plot for λ=3 though there were slight instability in the performance pattern.

The most favourable performance pattern is represented by plot *A*=3, *M*=20 and λ=3. It is the most optimal performing plot amongst all the compared simulation plots. For both with and without disturbance, the convergence rate was high and steadily increasing. When there was no disturbance, the simulation results attained asymptotic performance trajectory to the desired trajectory within 10 batches. In the presence of disturbance, there was slow but steady improvement in the results.

It can be concluded from Figure 6.9 that, in contrary to result in Figure 6.8, incorporation of model prediction confidence bounds into sliding window and averaged trajectory approach for MLR model have significantly improved the stability and convergence rate from batch to batch for all the 20 batches. Model prediction confidence bounds narrows the prediction bounds and improves the performance stability. An average of the latest 3 batches and historical batch sliding window size of 20 can be used as the reference trajectory to obtain better performance pattern.

6.6 Conclusions

In this chapter, model updating using a sliding window of historical batch is suggested. The method was applied to batch to batch updated model ILC fed batch fermentation simulation. The latest historical batch was used as reference trajectories. The proposed method proved efficient in reducing data load and capturing up-to-date information to improve process performance for all the three process model developing methods, which are MLR, PCR and PLS. A window size of only 10 latest batches is sufficient for PLS and PCR models based batch to batch updated linearised model ILC to produce steadily converging performance pattern with higher biomass concentrations than the Case 2 plots in Figure 4.12. The 10 batches are necessary possibly because there are 10 control policy intervals for the tested fed-batch simulation. The results suggest that possibly historical batch numbers equivalent to control policy intervals is sufficient to improve batch to batch control performance provided the historical is updated after batch run. As for MLR model, 20 updated historical batch data is sufficient to produce results resembling the MLR Case 2 results.

Following that, averaged reference trajectory approach was attempted with firstly, growing historical batches and then with the selected sliding window sizes for each of the regression models. For PCR and PLS models, both the approach did not improve the results any better than the sliding window size 10 with latest batch as the reference batch. For MLR model, the plots were highly unstable but the end batch biomass concentrations were higher than the MLR Case 2 plot. That brought about the idea to incorporate MPCB into the sliding window and averaged trajectory approach to exploit the possibility in improving the stability and end-batch concentration. The combined method approach certainly improved the results for MLR models. There was significant advancement in the stability and convergence towards desired trajectory for MLR model plots.

The sliding windows approach with window size 10 batches with latest batch as reference trajectory works very well with PCR and PLS models. The combined control method with A=3, M=20 and incorporation of model prediction confidence bounds significantly enhances the results for MLR model based batch to batch ILC.

CHAPTER 7: ILC WITH ADAPTIVE WEIGHTING PARAMETERS IN THE OBJECTIVE FUNCTION

7.1 Introduction

In this chapter a study on iterative learning control with adaptive weighting parameters in the objective function is presented. This method requires clever solution of the linear quadratic optimisation problem to improve convergence speed and ensuring stability. Attempts to achieve very fast convergence typically cause instability and vice-versa. Both this control criteria is determined by minimization/maximization a quadratic objective function. In a discrete control system, stability is typically related to convergence rate.

A typical quadratic objection function is as given below:

$$J_{k+1} = \min_{\Delta \overline{\mathbf{U}}_{k+1}} \frac{1}{2} [\tilde{\mathbf{e}}_{k+1}^T \mathbf{Q} \tilde{\mathbf{e}}_{k+1} + \Delta \overline{\mathbf{U}}_{k+1}^T \mathbf{R} \Delta \overline{\mathbf{U}}_{k+1}] \quad (7.1)$$

The first term in the cost criterion is the variance and the second one is the input change need to minimize the system variance while improving the system convergence. In ILC based on linear quadratic optimal control, the weighting matrices **Q** and **R** are important design parameters (Amann et al., 1996). The weighting matrices **Q** and **R** are used to maintain an optimal ratio between control policy change and error minimization. Convergence rate depends on the ratio of the weighting parameters in the quadratic objective function. It is essential that the control policy does not deviate too much from that of the previous batch (the k^{th} batch) whilst reducing tracking error in order to maintain stability (Amann et al., 1996). In this study, optimisation based on linearised models is used to calculate the control policy for the new batch. As for the batch-to-batch control policy, the control policy for every successive batch should be reducing the tracking error in product quality.

In Chapters 4, 5 and 6, the Q and **R** ratio in quadratic objective function were kept constant for all the batch runs with ILC. The weighting parameters used are Q is 1 and **R** is 0.0001. There have been significant improvements in batch to batch product concentrations when different techniques were introduced. However, there were some performance patterns that revealed high biomass concentration for the first few batches and then deteriorated for the rest of the batches forming a dumbbell curve as seen in

Chapter 5, Figure 5.5 for λ=10. In Chapter 4, Figure 4.12(b), the batch to batch product improvement for Q=1 and **R**=0.00001**I** exhibited very good convergence rate for the first four batches and then deteriorated for the following batches. If only these performances can be sustained or further improved, the desired trajectory can be achieved asymptotically much faster within smaller number of batches. Bearing this in mind, an idea to adjust the Q and **R** ratio from batch to batch was developed. Instead of using fixed Q and **R** ratio, an adaptive ratio may improve the convergence rate without compensating the system stability. The question here is as to how to systematically adjust the Q and **R** ratio from batch to batch.

It is known that ratio of **Q** and **R** in the objective function determines the convergence rate of biomass concentration towards desired trajectory. For batch-to-batch ILC, stability is closely related to convergence. A converging performances means the system is steadily improving from batch to batch. Having a control over the convergence speed of the system to be controlled will be beneficial. Amann et al. (1996) developed a strategy to control the convergence speed by manipulating the weighting parameters, **Q** and **R** in the objective function to achieve desired convergence rate. An optimization principle using Ricatti feedback in combination with conventional feed-forward ILC was introduced. A performance criterion which evaluates both current run feed-back mechanism and feed-forward of previous trial data was developed to be used to tune the quadratic objective function.

Amann et al. (1996) suggested that $\mathbf{R_0}$ is fixed and then it is defined by the objective function through $\mathbf{R}=\rho\mathbf{R_0}$. The weighting parameter **R** is determined automatically. The scalar, ρ is a variable. The ρ is derived from equation $\sigma^2= \sigma_o^2/\rho$ where σ is the smallest singular value of gain (G) and σ_o is the smallest singular value corresponding to $\mathbf{R_0}$. The smaller the ρ is, the faster the convergence will be. Hence, the parameter ρ can be used to control the convergence rate of a controlled system. The study was tested on linear, continuous steady state. The proposed method improved convergence rate for tracking error and input sequence trajectory. It was used for time-variant dimension within a trial. The detailed study revealed that determination of control policy in the same manner for batch-to-batch control was not delivering similar performance.

The advantage of the automated **Q** and **R** ratio used in constructing control policy is that we have control over convergence speed. Drawback is the magnitude of feed-forward is highly dependent on the state variance. Therefore, in worst case scenario, a sudden hike in input control policy may exist. The author did not consider the non-linear case, presence of any kind of disturbances and robustness of the study. Complete automated ratio tuning may not deliver stable performance for non-linear systems, in the presence of disturbance and/or model-plant mismatches.

Gao et al. (2001) diversified from Amann et al. (1996) by applying **Q** and **R** ratio modulating idea to general batch process. This study was done on a non-linear actuator process. Finite time interval is present and not automatically decided. In Amann et al., 1996 the batch run time is decided upon attainment of the desired performance which is decided by the automated **Q** and **R** ratio tuning. The effect of initialization error and unknown disturbance were taken into consideration in the work by Gao et al., 2001. The work presented by Gao et al. (2001) was on exponentially reducing the **Q** and **R** ratio, trial after trial. The ratio of **Q** and **R** was first fixed and then it was reduced trial after trial in the order of $0.6^{k\text{-}1}$. In this work ρ was the fixed **Q** and **R** ratio, which is 0.6. The ρ is to approach zero with increasing cycle number, *k*. The experimental results without the proposed method demonstrated that accumulation of initialization error and unknown disturbances causes instability in constant **Q** and **R** ratio. The exponentially reducing ρ after every batch run exhibits desirable convergence whilst minimizing tracking error and suppressing unknown disturbances and initialization errors.

This work is an adaptation of the works presented by Amann et al. (1996) and Gao et al. (2001). Both the introduction of ρ and systematically reducing Q and **R** ratio has been incorporated into this study. Fermentation process involves life mechanism and so it may not be that easy to automate the magnitude of feed rate. It is important that feed does not sway too high compared to previous feed rate to prevent cell damage. A more controlled adaptive Q and **R** ratio scheme is introduced in this chapter. The simulation results are discussed for PCR, PLS and MLR updated models for batch to batch iterative learning control.

This chapter is organised as follows. Section 7.2 presents the details of the proposed method for this chapter. Two methods are proposed in this section. One is the continuously reducing Q and **R** ratio from trial to trial and the other is error adaptive Q and **R** ratio. Section 7.3 presents the results for batch to batch continuously decreasing Q and **R** ratio for batch to batch ILC with updated linearised MLR, PCR and PLS models. Two different $\mathbf{R_0}$ values were tested for this method. Section 7.4 presents the results for error adaptive Q and **R** ratio applied in batch to batch ILC with updated linearised MLR, PCR and PLS models. This method was also tested with 2 different $\mathbf{R_0}$ values. Section 7.5 presents the results for combination of model prediction confidence bounds (the method proposed in Chapter 5) and continuously decreasing Q and **R** ratio technique. The combined method was applied to MLR, PCR and PLS models. Section 7.6 concludes the chapter.

7.2 The proposed method

There is no proper guidance on the selection of **Q** and **R**. It is the ratio of **Q** and **R** rather than the absolute value of these weighting matrices that matters in determining the convergence speed. Large ratio leads to unstable system due to insensitivity of the systems variance. A strong feed-forward action tends to accumulate process uncertainties causing strong fluctuation in control policy. Very small ratio causes slow convergence and is undesirable (Gao et al., 2001).

Multi-objective optimisation problem does not have a straight forward solution (Tousain et al, 2001). Identification of **Q** and **R** can be made simpler by fixing the **Q** to Q=1 and then identifying the appropriate **R** to achieve desired convergence rate as well as minimize tracking error, ensuring stability and robustness by eliminating the effect of disturbances (Rogers, 2008; Tousain et al, 2001). **R** can be used to tune the performance of the learning controller (Phan, 1998; Tousain et al, 2001). Therefore, Q is usually fixed as 1 and **R** is identified by trial and error as discussed in Chapter 4. In Chapter 4, the **R** value was selected firstly based on the stability of the control system and then the convergence rate was considered. The **R** value selected for study in the previous chapters is the one with convergence rate compensated for stability. Actually, much smaller **R** value produced higher convergence rate and desired trajectory is attainable within 3 to 4 batch runs without disturbances. However, the constant high feed-forward signal did not provide a stable outcome to sustain the good performance.

When there is no disturbance, a constant and big ratio gives very good convergence rate. When there are disturbances, the ratio should be smaller to ensure stability. In addition to that, when the desired trajectory is almost achieved, there is no need to improve the feed so much and so reduced ratio would be good enough. Therefore, the ratio adapting to the error is suggested. It does not have to be reducing all the way through for all the batches as suggested by Gao et al. (2001). When there is disturbance, the ratio just has to be smaller and reducing till a second stable platform is found. When it is stable, the smaller ratio can be made constant. It is necessary that varying weighting schemes are applied to cater to practical application possibilities (Gao et al., 2001).

In this study, the initial Q and **R** values were fixed as $Q_0=1$ and $\mathbf{R_0}=0.0001\mathbf{I}$ or $\mathbf{R_0}=0.00001\mathbf{I}$. The value $\mathbf{R}=0.0001\mathbf{I}$ has been used throughout the study and so it will be used as R_0 for this technique. The **R** value of 0.00001**I** was dismissed in Chapter 4 due to instability in performance. It will be interesting to evaluate the performance of this value with the adaptive technique. The other $\mathbf{R_0}$ value that was tested for the proposed method is 0.00001**I**. The Q and **R** ratio is then reduced in accordance to a pre-specified weighted parameter, ρ, exponentially. The ρ is decided on trial and error basis. The adaptive method was tried in two ways to evaluate the simulation performances. The ratio was reduced either based on error or with increasing batch numbers. The methods used are further explained below.

Firstly, the ratio was reduced with batch index. The ratio is defined as,

$$\text{Ratio} = (Q_0/\mathbf{R_0}) \times \rho^{(k-1)}$$

The second method is reducing the ratio with error, whereby when $y_d - y_k < 3$g/l, the ratio will reduce as per definition below. A minimum ratio is set to prevent a reduction to zero ratios.

$$\text{Ratio} = \max\left((Q_0/\mathbf{R_0})\times\rho^{(k-1)}, 3000\right)$$

When y_d-$y_k > 3$g/l, the ratio becomes constant as; Ratio = $(Q_0/\mathbf{R_0})$, to produce bigger feed-forward control policy.

The proposed method was applied to the batch to batch ILC with updated linearised models and nominal trajectories technique. It is the Case 2 results discussed in Chapter 4, Figure 4.12. The simulation results for the proposed method are presented in the sections below.

7.3 Continuously decreasing Q/R ratio

In this section, batch wise continuously reducing Q and **R** ratio method is applied to batch to batch ILC with updated linearised MLR, PCR and PLS models. Two $\mathbf{R_0}$ values were tested for each model. The values are $\mathbf{R_0}$=0.0001**I** and $\mathbf{R_0}$=0.00001**I**. The scalar ρ used to modulate the Q/**R** ratio was varied between 0 and 1 for each of the models. The ρ values that give presentable results are presented in this section.

7.3.1 ILC using MLR models with $\mathbf{R_0}$=0.0001I

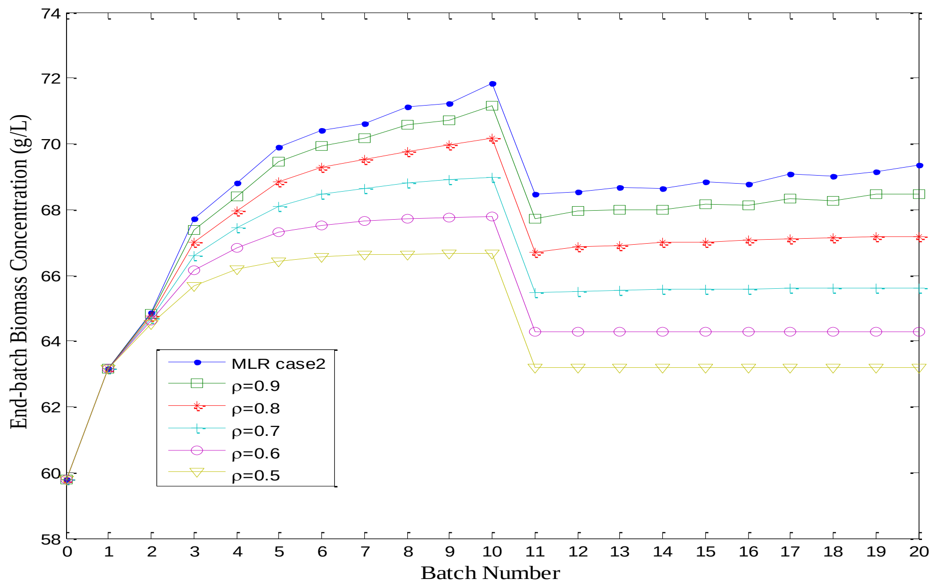


Figure 7.1: ILC performance with continuously decreasing Q/**R** ratio for ρ between 0.5 and 0.9 using MLR models with $\mathbf{R_0}$=0.0001**I**

Figure 7.1 shows results for continuously decreasing Q/**R** ratio for ρ between 0.9 and 0.5 using MLR model with $\mathbf{R_0}$=0.0001**I**. It is clear that this technique is not improving the convergence rate any better than the reference plot from Chapter 4, labelled as MLR Case 2 in Figure 7.1. Though the stability improved, the convergence rate decreased with decreasing ρ values.

7.3.2 ILC using MLR models with $\mathbf{R_0}$=0.00001I

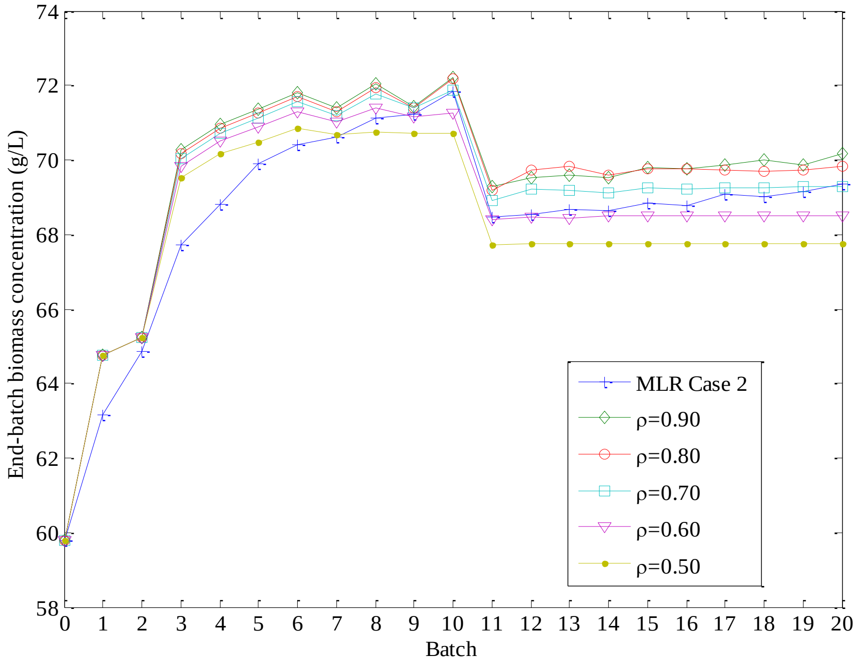


Figure 7.2: ILC performance with continuously decreasing Q/**R** ratio for ρ between 0.5 and 0.9 using MLR models with $\mathbf{R_0}$=0.00001**I**

In Figure 7.2, the continuously decreasing Q and **R** ratio was applied to MLR model with $\mathbf{R_0}$=0.00001**I**. The convergence rate certainly increased for most of the batches without disturbance for all the ρ values but batch to batch improvement is not consistent. For trials with disturbance, the batch to batch improvement is slow. The control performances for ρ values of 0.9, 0.8 and 0.7 were better than the reference performance. Overall, the continuously decreasing Q and **R** ratio technique did not have very favourable impact for MLR model. In the event if the best performing ρ value is to be picked, then ρ = 0.9 gives better convergence rate for batches with and without disturbances compared to the reference performance although the performance is a little inconsistent.

7.3.3 ILC using PCR models with $\mathbf{R_0}$=0.0001I

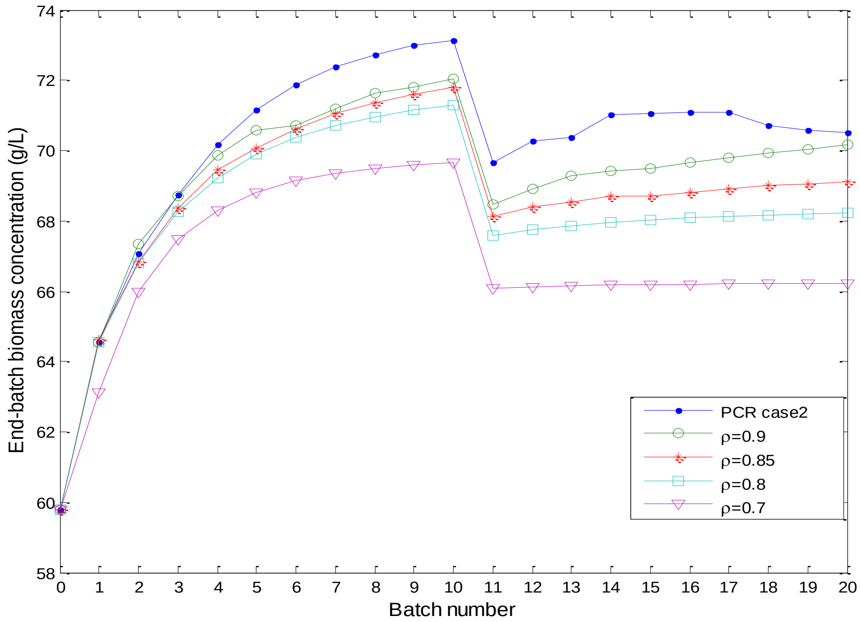


Figure 7.3: ILC performance with continuously decreasing Q/**R** ratio for ρ between 0.7 and 0.9 using PCR models with $\mathbf{R_0}$=0.0001**I**

Figure 7.3 shows results for continuously decreasing Q/**R** ratio for $\mathbf{R_0}$=0.0001**I** applied to a batch to batch ILC updated linearised model and nominal trajectories. The results were compared to the PCR results in Chapter 4, Figure 4.12 which was without the Q_0/$\mathbf{R_0}$ modulation. A few ρ values were tried on the simulation. It is evident in the plot that the suggested method for ρ values 0.9, 0.85, 0.8 and 0.7 does not improve the convergence rate compared to the plot without the technique application. The convergence rate decreases with decreasing ρ values. However, it is noticeable that the stability of the control system in the presence of disturbance is well improved. It is noted that the performance with **R**=0.0001**I** without decreasing ratio gives much better convergence rate but with slight instability in the presence of disturbance. To improve the instability, ρ values in between 0.9 and 1 were attempted. The values between 0.99 and 0.95 have portrayed some improving results. The control performances are presented in Figure 7.4.

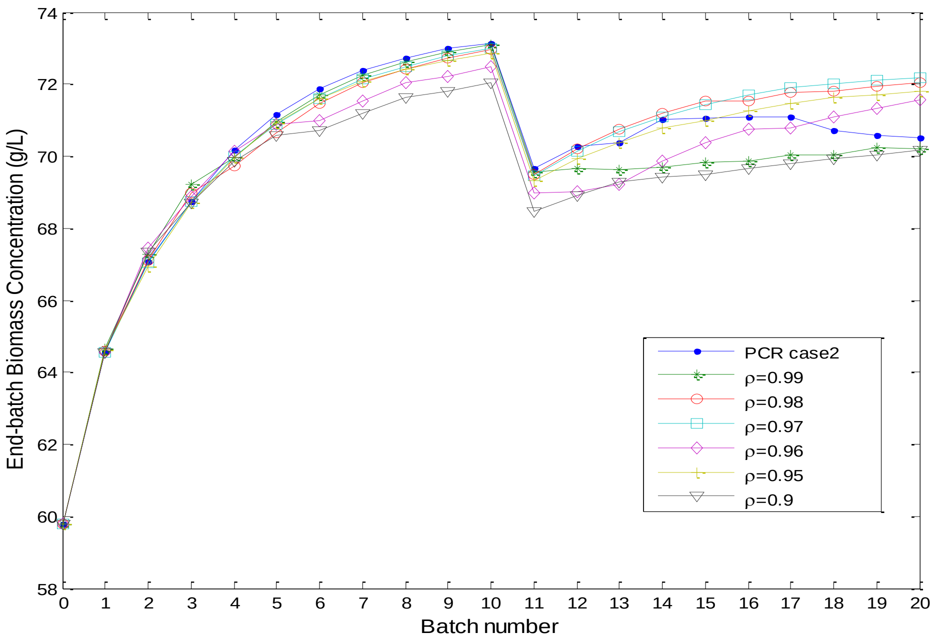


Figure 7.4: ILC performance with continuously decreasing Q/**R** ratio for ρ between 0.99 and 0.95 using PCR models with $\mathbf{R_0}$=0.0001**I**

In Figure 7.4, despite results for ρ values between 0.95 and 0.99, the performance plots for ρ = 0.9 from Figure 7.3 and PCR Case 2 from Figure 4.12 are shown for comparison purpose. All the ρ values presented in Figure 7.4 show higher final biomass concentration compared to the case of ρ = 0.9. In the following discussion, the performance pattern of the ρ values between 0.95 and 0.99 will be compared to the PCR Case 2 performance, referred to as the reference performance in the following discussion. From Figure 7.4, it can be noted that for ρ = 0.95, 0.97 and 0.98 there were steady increment in convergence rate from trials to trials for both with and without disturbance conditions. For trials without disturbance, the biomass concentrations for all the 10 batches for ρ = 0.95, 0.97 and 0.98 were very close to one another and also to the reference performance. The improvement is very significant in the presence of disturbance. Both the stability and convergence rate is improved in the presence of disturbance with the continuously decreasing ratio technique. The ρ values of 0.98 and 0.97 show improvement in biomass concentration for all the batches with disturbance in comparison to the reference performance. The control performance for ρ = 0.95 exhibits slightly lesser biomass concentrations for the first 5 batches with disturbance compared to the reference performance. In the following 5 batches, the trials to trial production increased steady, way better than the reference performance. Decreasing the Q and **R** ratio continuously, trial after trial in the presence of disturbance, does exhibit favourable results. The performance for the case of ρ=0.97 portrays best performance pattern for with and without disturbance in comparison to the other ρ values in Figure 7.4. As for ρ = 0.96, though the biomass concentrations were improving from trial to trial for all the 20 batches, there were slight instability and lesser convergence rate in comparison to ρ = 0.95, 0.97 and 0.98. The performance pattern for ρ = 0.99 was promising for batches without disturbances but did not sustain the performance for batches with disturbances, the biomass concentrations were achieving similar value as that of the ρ=0.9 case.

7.3.4 ILC using PCR models with $\mathbf{R_0}$=0.00001I

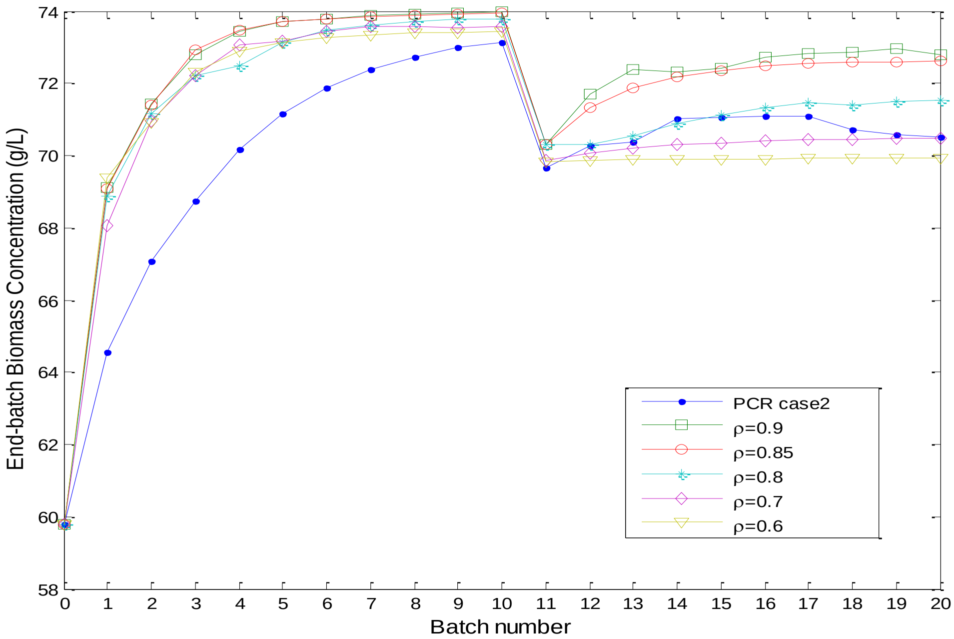
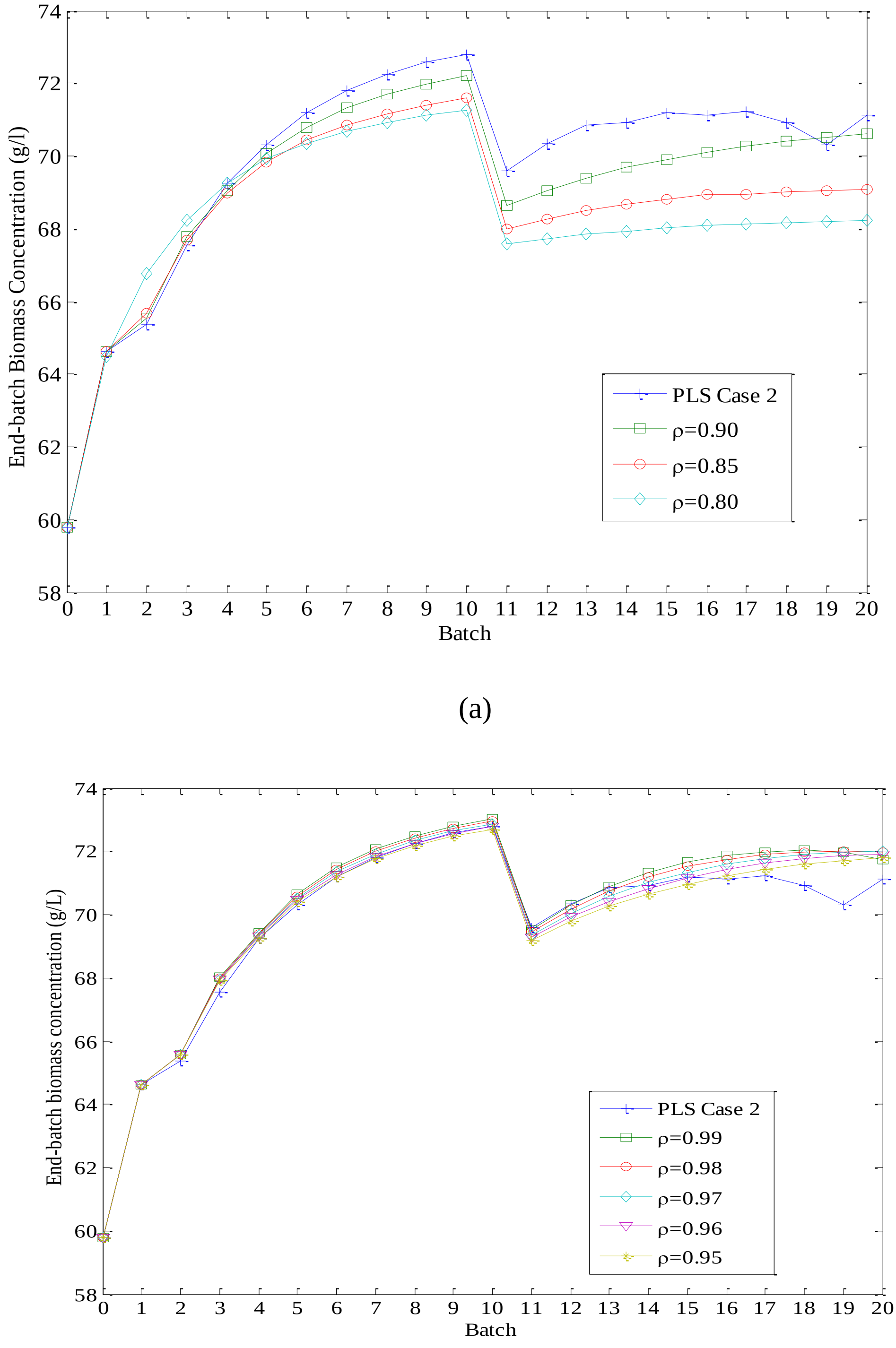


Figure 7.5: ILC performance with continuously decreasing Q/**R** ratio for ρ between 0.6 and 0.9 using PCR models with $\mathbf{R_0}$=0.00001**I**

In Figure 7.5, the continuously decreasing Q and **R** ratio was applied with $\mathbf{R_0}$=0.00001**I**. In Chapter 4, Figure 4.10 (c) shows the performance for **R**=0.00001**I** portrayed highly unstable performance. It is possibly due to high feed-forward input which is not necessary for the system. The technique was applied with varying ρ values with $\mathbf{R_0}$=0.00001**I** and Q_0=1. The performances were compared to the case of **R**=0.0001**I** without decreasing ratio (reference performance). It is evident that without disturbance; all the ρ values exhibited very fast converging results compared to the reference performance. The cases with ρ=0.85 and 0.9 exhibit most desirable results. The desired trajectory was asymptotically attained within 5 batches and the performance were gradually improved and maintained close to the desired trajectory for the following 5 batches. In the presence of disturbance, there were varying performance patterns among the ρ values. The convergence rates were improving with increasing ρ values. All the ρ values portrayed desirable stability except for $\rho = 0.9$. Both $\rho = 0.85$ and 0.9 exhibit highest biomass concentrations for trials with disturbances amongst the tested ρ values. The performance pattern for ρ=0.85 is the most favourable based on the stability and the convergence rate. The method to decrease Q and **R** ratio greatly improves the simulation results even for smaller **R** value.

7.3.5 ILC using PLS models with R_0=0.0001I



(a)

(b)

Figure 7.6: ILC performance with continuously decreasing Q/**R** ratio for ρ between 0.8 and 0.9 (a) and between 0.95 and 0.99 (b) using PLS models with $\mathbf{R_0}$=0.0001**I**

Figure 7.6(a) shows results for continuously decreasing Q and **R** ratio for ρ between 0.8 and 0.9 using PLS model. The performance for PLS Case 2 without decreasing Q and **R** ratio, as adapted from Chapter 4, Figure 4.12, is used as the reference performance to evaluate the performance of different ρ values. It is noted that the convergence rate decreases from ρ=0.9 to ρ=0.8. Though the control performance is improving steadily from batch to batch, higher convergence rate is desired. Following that in Figure 7.6(b), ρ values ranging between 0.95 and 0.99 were attempted. It is interesting to note that there are favourable improvements in the performance pattern. The first ten batches without disturbance for all the tested ρ values improved from trial to trial but the convergence rate did not differ much as compared to the reference performance. There were slight increments from the reference performance in every batch trial without disturbance. The convergence rate increased steadily with increasing ρ values. For trials with disturbance, the batch to batch biomass concentrations were improving steadily and converging towards the desired trajectory. The stability of the tested cases is certainly improved compared to the reference performance. The convergence rate increases with increasing ρ values. Among the considered ρ values, the best performing ρ value is 0.99.

7.3.6 ILC using PLS models with R_0=0.00001I

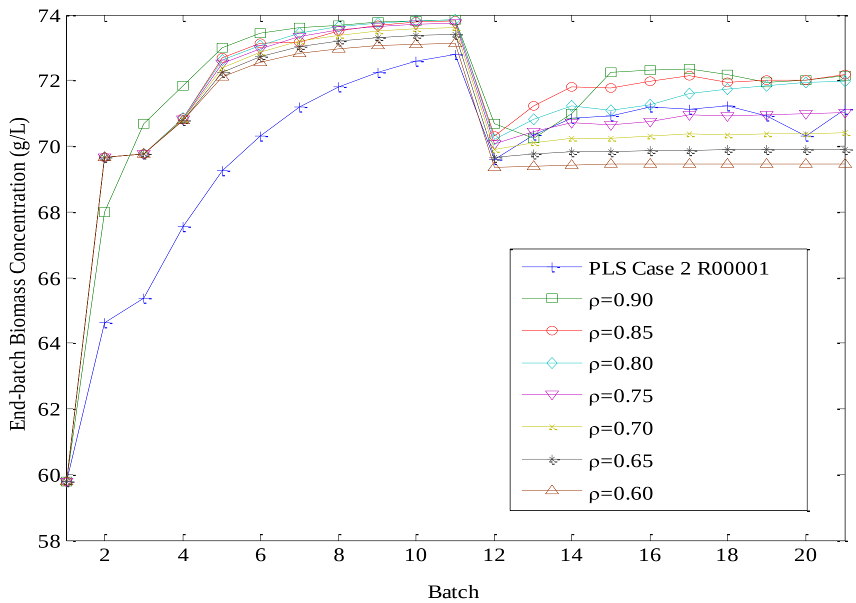


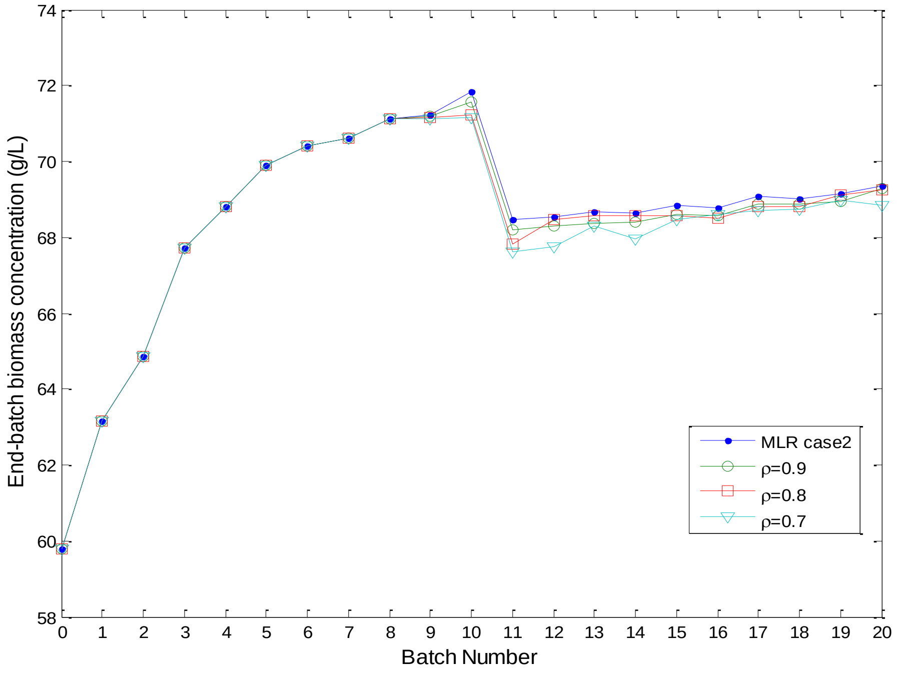
Figure 7.7: ILC performance with continuously decreasing Q/**R** ratio for ρ between 0.60 and 0.90 using PLS models with $\mathbf{R_0}$=0.00001**I**

With reference to Figure 7.7, it is noticeable that $\mathbf{R_0}$=0.00001**I** can be used for the batch to batch ILC control with the continuously decreasing Q and **R** ratio technique. This was not possible with just batch to batch ILC as discussed in Chapter 4. The convergence rates for batches with and without disturbances increases with increasing ρ values. The first ten trials meant for tracking desired trajectory, achieved asymptotic performance within 6 batches. The stability of all the ρ values was favourable. The best performing was for ρ=0.90. As for batches with disturbances, there were varying convergence rate and system stability. The convergence rate decreased with decreasing ρ values. The stability of the system deteriorated with increasing ρ values. The plots for ρ=0.90, 0.85 and 0.80 portray higher convergence rate than the reference plot but the performance stability is a little inconsistent.

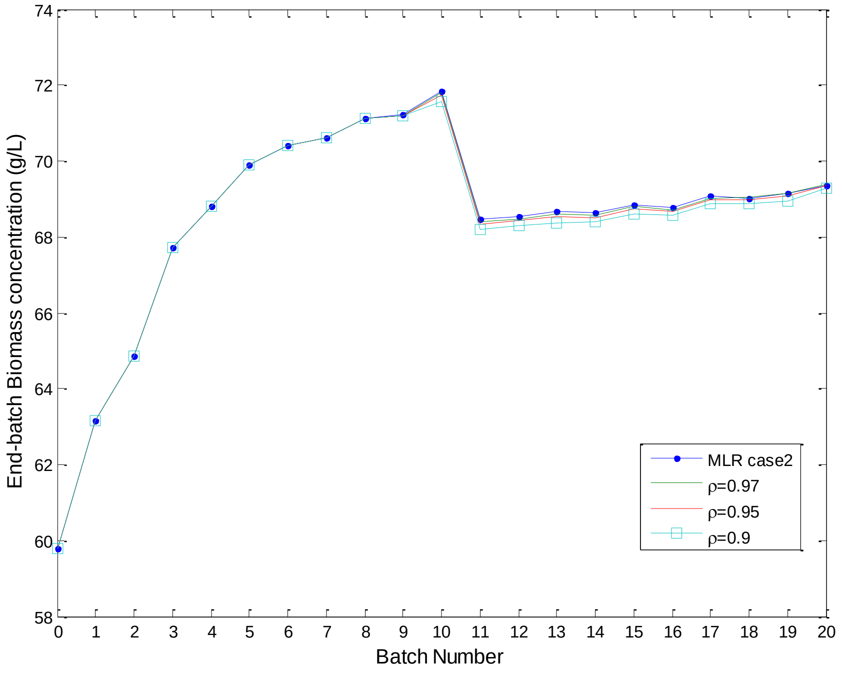
7.4 Adaptive Q/R ratio for pre-defined error limit

In this slightly modified method than the continuously decreasing ratio discussed in Section 7.3, the Q_0 and $\mathbf{R_0}$ ratio remains constant when the error is bigger than 3g/l. For any batches when the output error is 3g/l or less, the ratio will start decreasing from batch to batch till a minimum ratio of 3000 is reached. In the presence of disturbance, when the error becomes bigger than 3g/L, the ratio goes back to the constant $Q_0/\mathbf{R_0}$ whereby Q_0=1 and $\mathbf{R_0}$=0.0001**I**. The plots with different ρ values are compared to the batch to batch ILC updated model plots for MLR, PCR and PLS from Chapter 4, Figure 4.12. These plots are referred to as reference plot in the respective graphs.

7.4.1 ILC using MLR models with R_0=0.0001I



(a)



(b)

Figure 7.8: ILC performance with error dependant adaptive Q/**R** ratio for ρ between 0.7 and 0.9 using MLR models with $\mathbf{R_0}$=0.0001**I**

In Figure 7.8, the adaptive Q and **R** ratio method is applied to MLR model with $\mathbf{R_0}$=0.0001**I**. As seen in both Figure 7.8(a) and (b), this method does not improve the batch to batch performance. This is because y_d - y_k is more than 3 for most of the batches and so the adaptive method is not used for most of the batches.

7.4.2 ILC using MLR models with $\mathbf{R_0}$=0.00001I

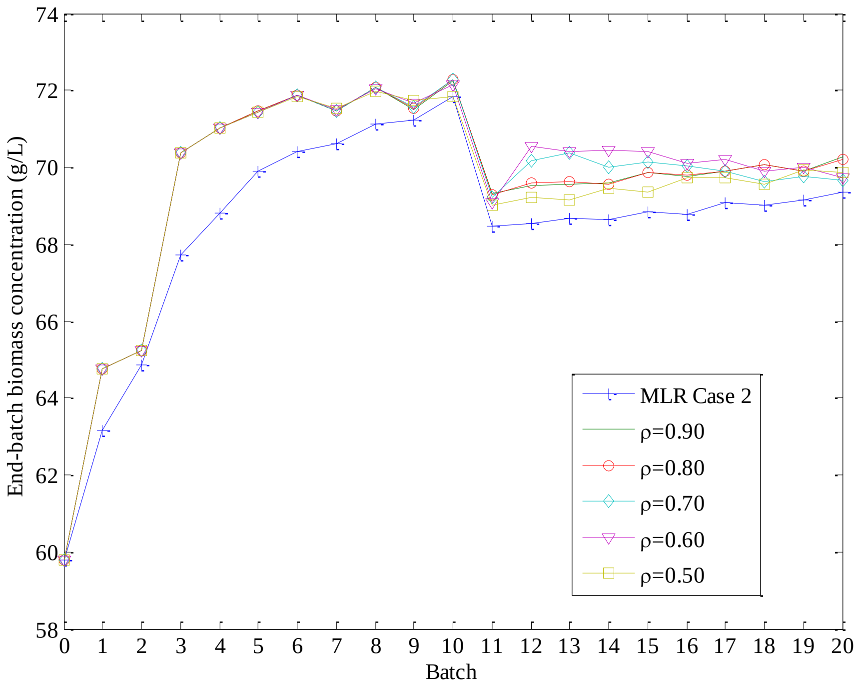
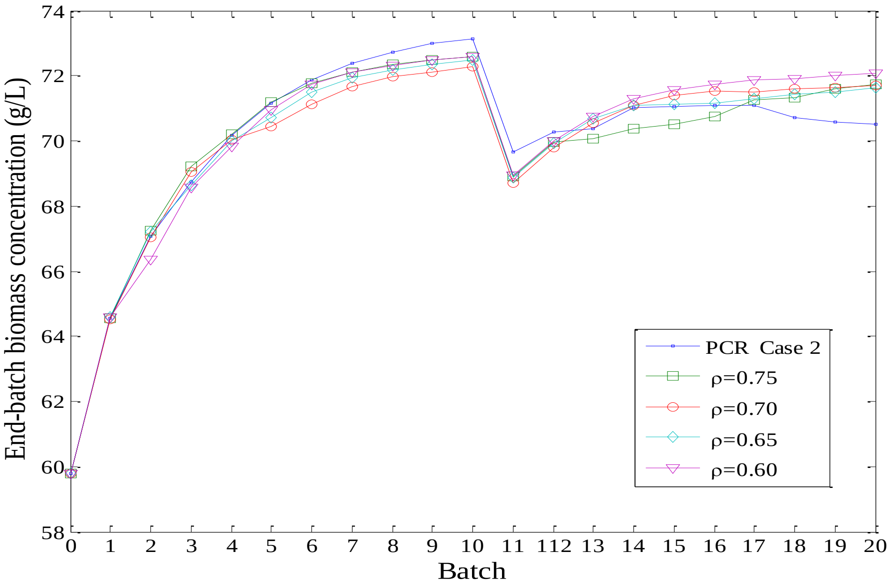


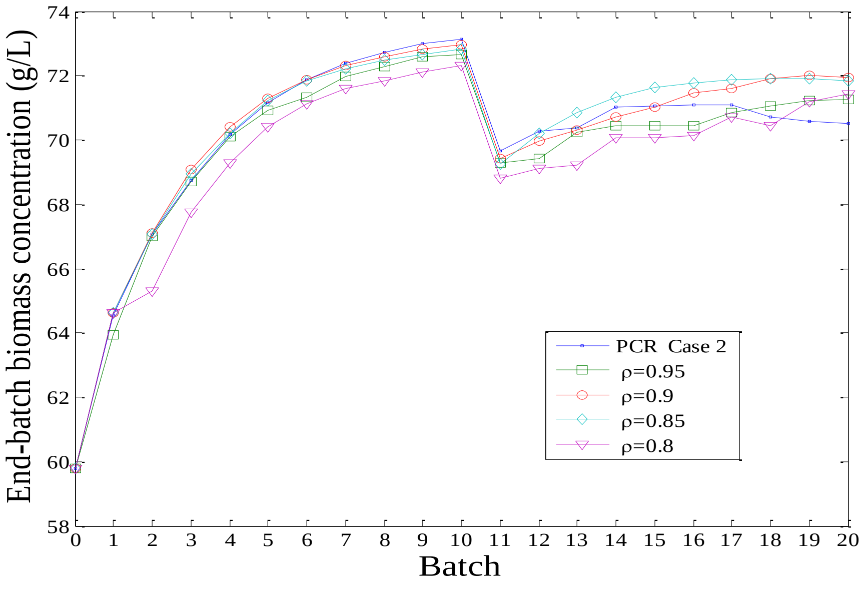
Figure 7.9: ILC performance with error dependant adaptive Q/**R** ratio for ρ between 0.5 and 0.9 using MLR models with $\mathbf{R_0}$=0.00001**I**

As for $\mathbf{R_0}$=0.00001**I**, the tested ρ values revealed higher biomass concentrations for all the batches compared to the reference plot. However, they were inconsistent, batch to batch product increments for both with and without disturbances. Therefore, there results were not favourable.

7.4.3 ILC using PCR models with R_0=0.0001I



(a)

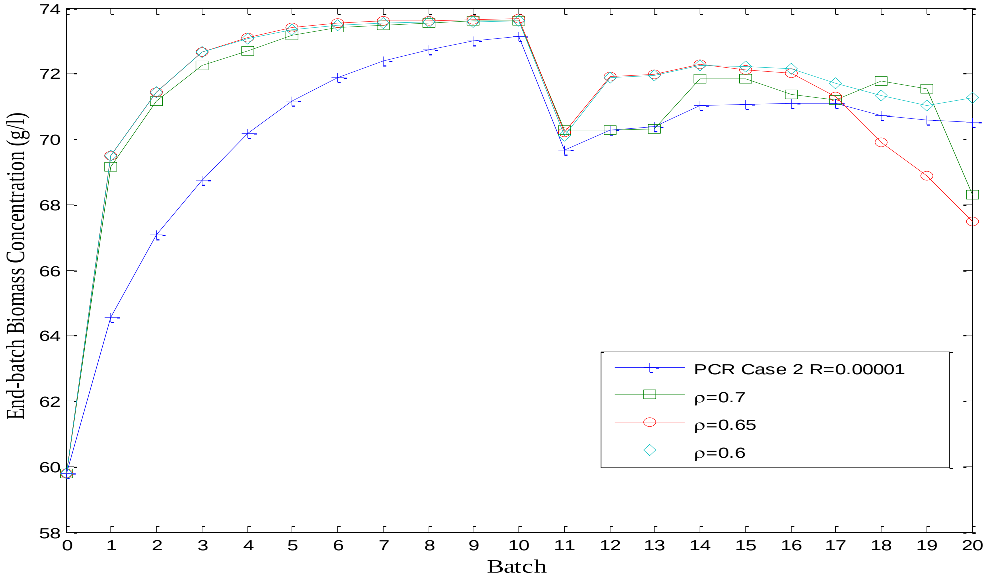


(b)

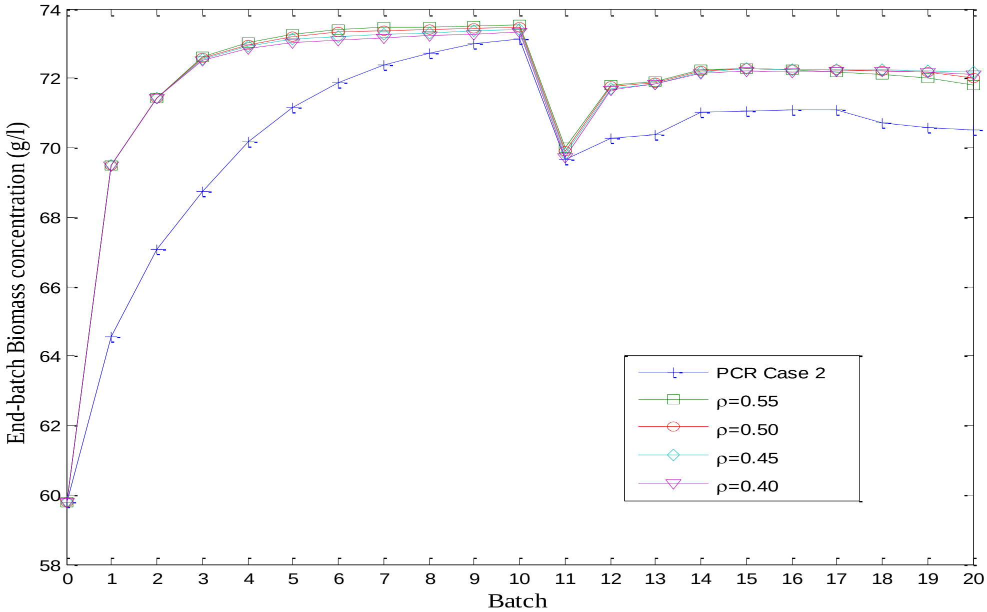
Figure 7.10: ILC performance with error dependant adaptive Q/**R** ratio for ρ between 0.6 and 0.95 using PCR models with $\mathbf{R_0}$=0.0001**I**

Figure 7.10 presents the results for error dependant adaptive Q and **R** ratio for $\mathbf{R_0}$=0.0001**I** using PCR models. Referring to Figure 7.10, all the tested ρ values exhibit batch to batch increment for all the 20 batches. The impact of the adaptive ratio can be seen for trials with disturbance. In Figure 7.10 (a), the performance for ρ=0.60 exhibits better convergence and stability compared to the other ρ values in the presence of disturbance. For batches without disturbances, most of the batches revealed lesser biomass concentration compared to the reference plot. Therefore, bigger ρ values were attempted and presented in Figure 7.10 (b). In Figure 7.10(b), for the first ten trials without disturbance, the performances for ρ=0.85 and 0.90 reveal similar results to the reference performance except for the last three batches, there were slight decrease in the concentration. In the presence of disturbance, both these control performances were improving steadily from batch to batch and produced higher biomass concentration for most of the batches compared to the reference performance. In this case, both ρ=0.85 and 0.90 exhibit favourable performance pattern. The plot for ρ=0.85 exhibits higher biomass concentrations for most of the batches compared to the case of ρ=0.90 in the presence of disturbance. In contrast to the continuously decreasing Q and **R** ratio discussed above, smaller ratio is needed to achieve similar performance pattern with the error dependent adaptive Q and **R** ratio.

7.4.4 ILC using PCR models with R_0=0.00001I



(a)



(b)

Figure 7.11: ILC performance with error dependant adaptive Q/**R** ratio for ρ between 0.4 and 0.7 using PCR models with $\mathbf{R_0}$=0.00001**I**

In Figure 7.11, the adaptive Q and **R** ratio is applied for $\mathbf{R_0}$=0.00001**I** using PCR model. As seen in Figure 7.11(a), the adaptive method exhibits favourable results for trials without disturbance. In the presence of disturbance, high instability is portrayed although most of the batches have higher biomass concentrations compared to the reference plot. Figure 7.11 (b) shows that for ρ = 0.40, 0.45, 0.50 and 0.55, there were stable batch to batch increment for trials with and without disturbance. The biomass concentrations did not differ much between the plots. Though the biomass concentrations for all the tested batches were higher than the reference plot, the convergence rate could have been better.

7.4.5 ILC using PLS models with R_0=0.0001I

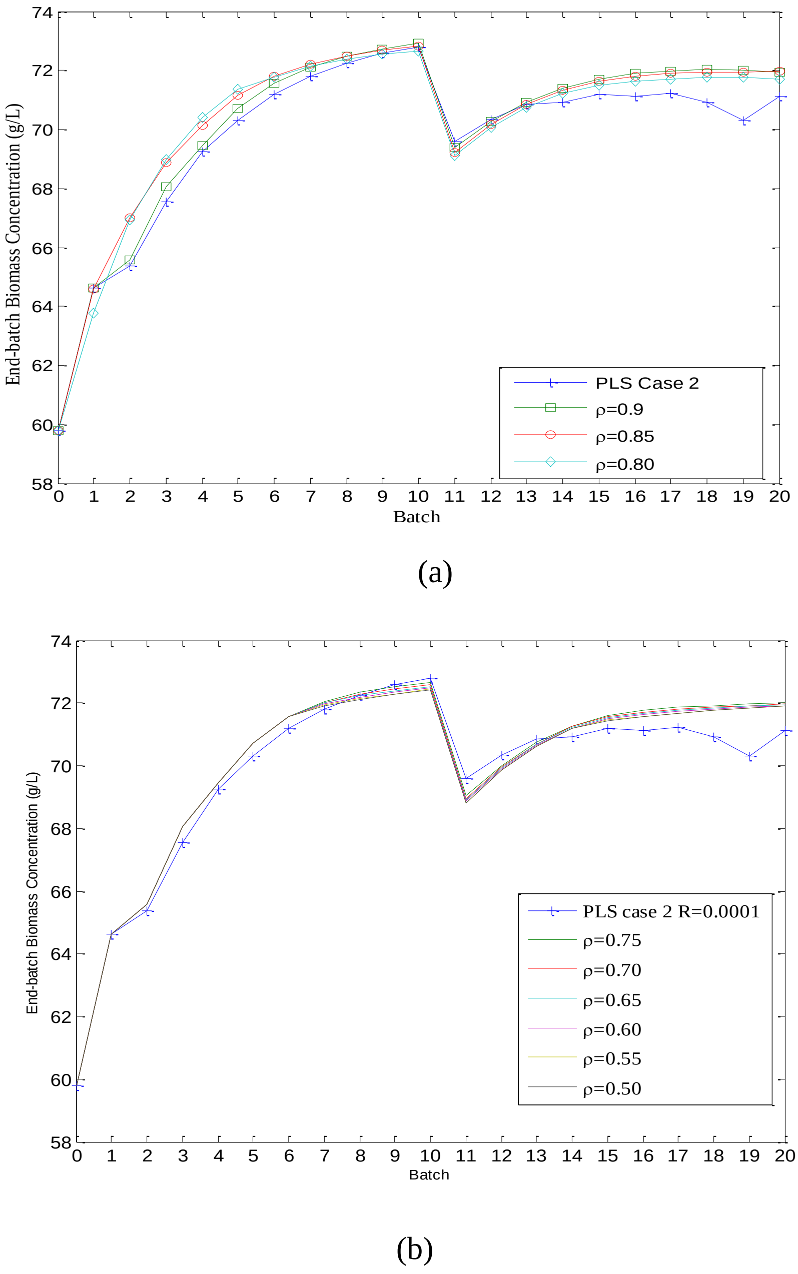


Figure 7.12: ILC performance with error dependant adaptive Q/**R** ratio for ρ between 0.50 and 0.90 using PLS models with $\mathbf{R_0}$=0.0001**I**

In Figure 7.12 (a) and (b), the performance of PLS model using adaptive Q and **R** ratio with varying ρ value is presented. It is evident that for all the presented ρ values, there is steady batch to batch improvement for trials with and without disturbance. With reference to Figure 7.12 (a), the control performance for ρ=0.85 and 0.80 show steady and improved convergence rate for batch to batch trials for runs with and without disturbances compared to the reference plot. The performance for ρ=0.90 revealed slight improvement for most of the batches without disturbance and significant improvement from 4^{th} trial on wards for trials with disturbance compared to the reference performance. All the three tested cases in Figure 7.12 (a) give favourable performance pattern. There is notable improvement in the stability and convergence rate in these plots. In comparison to Figure 7.6, the adaptive method seems to reveal better results with lower Q and **R** ratio with the PLS model. In Figure 7.12 (b), smaller ρ values were attempted. The plots for ρ between 0.50 and 0.75 revealed similar performance to one and another. The biomass concentrations were improved for most of the tested batches. The stability of the control system is also improved for trials with disturbance. However, the plots for ρ values in Figure 7.12 (a) are more favourable simply because the batch to batch improvements are better. The adaptive method did not work well with the $\mathbf{R_0}$=0.00001**I** especially in the presence of disturbance, most probably due to very high feed-forward control policy.

7.5 Reliable ILC with adaptive Q and R ratio

In this section, the continuosly decreasing Q and **R** ratio technique is combined with the reliable ILC method through incoporating model prediction confidence bounds for MLR, PLS and PCR models. For all the three models, the performance of reliable ILC is compared to the combined technique.

7.5.1 Reliable ILC using MLR models with adaptive Q and R ratio

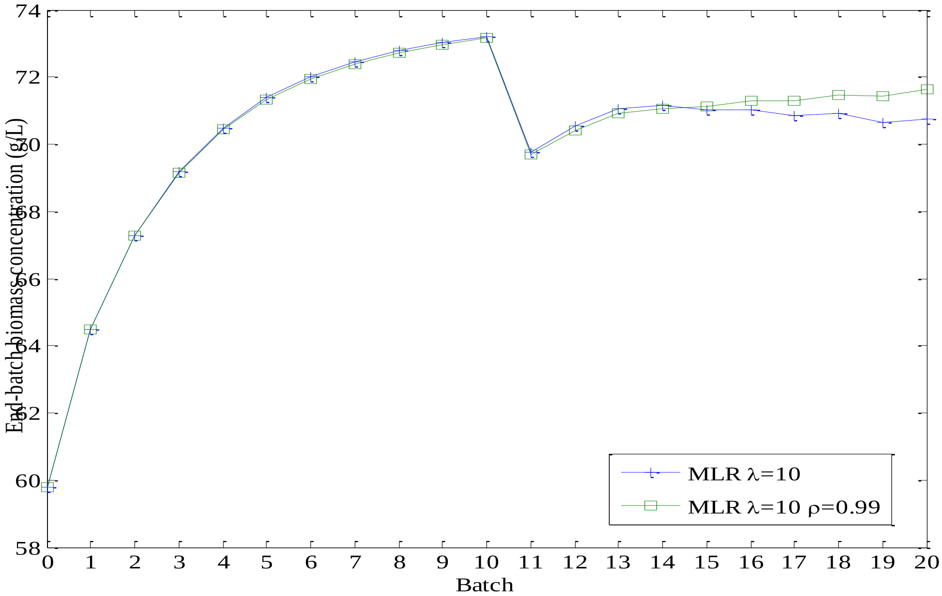
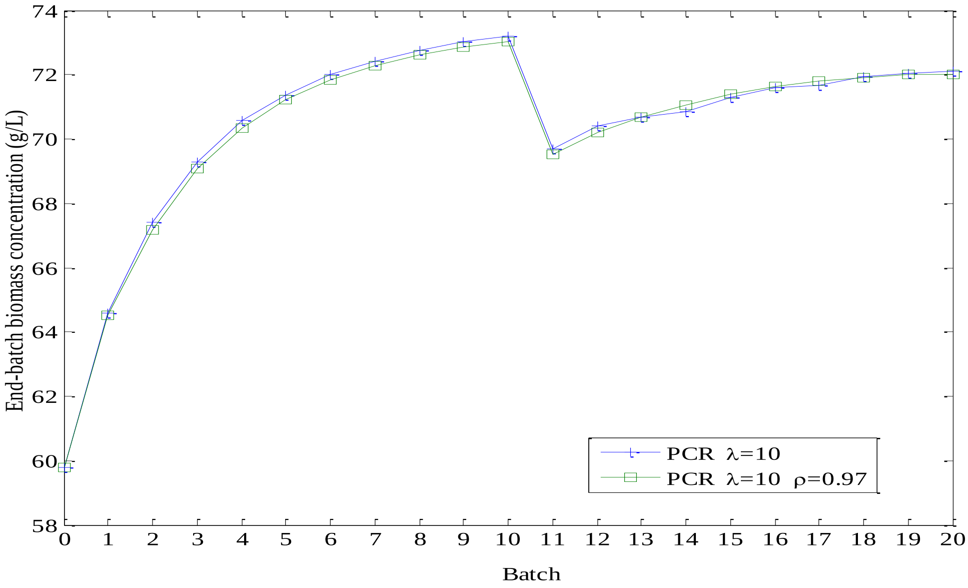


Figure 7.13: Performance of reliable ILC with adaptive Q and **R** ratio using MLR models

As discussed in the previous section, the continuously decreasing Q and **R** ratio improves the system stability rather than the convergence rate. In Figure 7.13, the performance or reliable ILC is compared with that of the combined technique. For batches without disturbance, the stability and convergences rates did not differ much for the two methods. For batches with disturbance, the combined technique has improved the stability and batch to batch concentration for the last 5 trials. Smaller Q and **R** ratio contributed to the noticed improvement.

7.5.2 Reliable ILC using PCR models with adaptive Q and R ratio



.

Figure 7.14: Performance of reliable ILC with adaptive Q and **R** ratio using PCR models

As for reliable ILC using PCR models in Figure 7.14, the performance of reliable ILC is in favourable performance pattern except for very slight instability for few batches with disturbance. The combined technique improves that small bit of instability. The convergence rate for both the plots for the entire 20 batch does not differ much.

7.5.3 Reliable ILC using PLS models with adaptive Q and R ratio

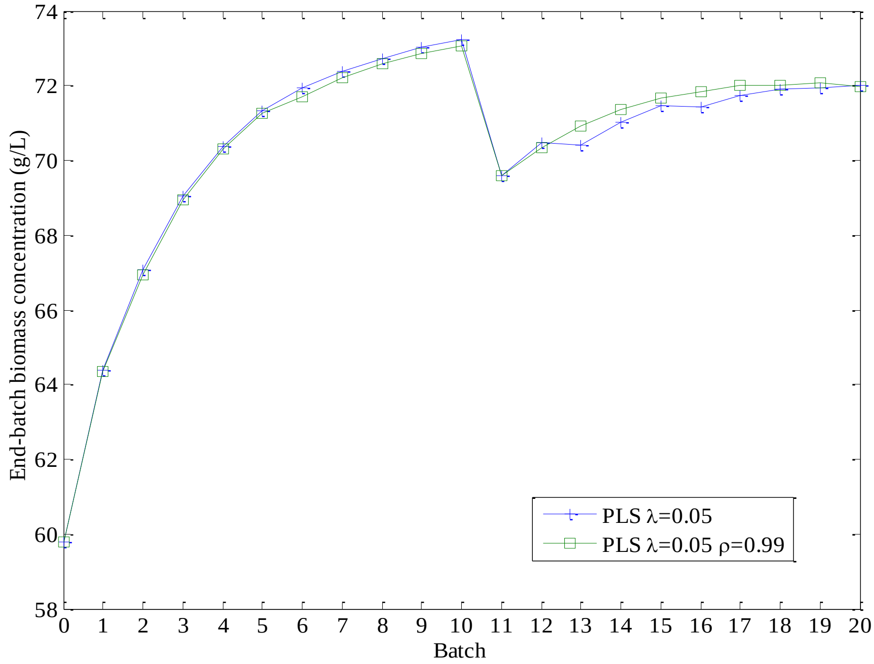


Figure 7.15: Performance of reliable ILC with adaptive Q and **R** ratio using PLS models

As for reliable ILC using PLS models, as seen in Figure 7.15, the stability and convergence rate for batches without disturbance did not differ much for the first 5 batches but there were slight decrease in final biomass concentration is the following 5 batches for the combined technique. In the presence of disturbance, the slight instability is reliable ILC is improved with the combined technique. Decreasing the Q and **R** ratio improved the stability and batch to batch final product concentration in the presence of disturbance for PLS model.

7.6 Conclusions

In this chapter, adaptive Q and **R** ratio technique was attempted and the results were discussed. It is evident that in most of the cases this technique improves the stability of the batch to batch control but does not always improve the biomass concentrations compared to the reference plot. There were two different techniques attempted and applied to ILC with PCR, PLS and MLR models. The MLR model was not suitable for this technique. There were no improvements in results noticed. Both the PCR and PLS models exhibited improving results in terms of convergence rate and stability, for both the techniques. For $\mathbf{R_0}$=0.0001**I**, there were significant improvement in the stability and convergence rate for trials with disturbance. The adaptive Q and **R** ratio enables smaller **R** value to start off with, which is $\mathbf{R_0}$=0.00001**I**. The adaptive Q and **R** ratio gives more control over the batch to batch performance resulting in certain limitation to the performance pattern. The results for continuously reducing Q and **R** ratio are more varied in performance. The adaptive ratio method used smaller ρ values compared to the continuously reducing method.

The continuously reducing ratio technique was combined to the reliable ILC method incorporating model prediction confidence bounds with the intention to improve the system stability and possibly the convergence rate. As expected, the stability is improved for all the three tested models, MLR, PLS and PCR.

CHAPTER 8: CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORKS

8.1 Conclusions

Fed-batch fermentation has been vital and present for many years in the industry. Even then, the development of advanced control and optimization in this area is still lagging compared to the continuous systems. This is mainly due to its non-linear and complex nature. At present, control policy to obtain desired product quality in a fed-batch process is calculated offline. Due to model-plant mismatches and the presence of unknown disturbances, off-line calculated control policy may not be optimal when implemented to the real process. The repetitive nature of batch process allows information from the previous batches being used in modifying the control policy of the next batch in the framework of iterative learning control. Iterative learning control (ILC) exploits every possibility to incorporate past control information, in particular the past tracking error and control input signals, into the construction of the present control action through memory based learning. It forms the feed-forward part of the current control action to complement the existing control methods. In this work, implementation and improvement of batch to batch iterative learning control into fed-batch fermentation are studied.

A batch to batch iterative learning control strategy based on incrementally updated linearised model has been implemented on a simulated fed-batch fermentation process in this study. The linearised models were developed using multiple linear regressions (MLR), principle component regression (PCR) and partial least squares (PLS). The proposed strategy could overcome the detrimental effects of model-plant mismatches and unknown disturbances by incrementally updating the control policy using information from the previous batches. Control policy updating is calculated using a linearised model identified from process operation data. In order to tackle the nonlinear behaviour of fermentation processes, the linearised model is updated from batch to batch and the immediate previous batch is used as the reference batch. In the effort to address the colinearity among the control actions during different batch stages, the linearised models are identified using PCR and PLS.

The initial work carried out in this study suggests that both PCR and PLS regression methods can be used to develop batch-wise linearised models for batch to batch iterative learning control. It also reveals that updating of the model parameter, Gs and nominal reference variables, Ys and Us, is necessary to steadily track desired trajectory in the presence of disturbance. Model updating using PCR and PLS leads to better control performance than model updating using MLR. Comparing batch to batch control strategies based on the PLS and PCR models, the one based on the PCR model is preferred due to its feed rate stability and enhanced productivity rate under disturbance.

Following that, a reliable batch to batch iterative learning control strategy based on incrementally updated linearised model is discussed. Model prediction confidence is incorporated in the ILC optimisation objective function to penalise wide model prediction confidence bounds. Application results on a simulated fed-batch fermentation process demonstrate that the proposed technique is very effective. The performance of reliable ILC with MLR model is significantly improved. The reliable ILC with MLR model revealed desirable tracking ability and produced favourable result in the presence of continuous disturbance. Reliable ILC with PCR and PLS models also show improved performance.

A sliding window historical batches approach was integrated with the batch to batch ILC with updated linearised models. This method suggests that for PLS and PCR, the number of historical batches equal to the number of feed intervals is sufficient to develop a process model. In order to capture the recent process environment, the historical batches should be updated after every batch run. The optimal window size for MLR is 20 and for PCR and PLS is 10. The MLR model with sliding window size 20 gave very resembling results to ILC with batch to batch updated models and without sliding window approach. There was no further increment in the end product concentration but it suggest that 20 historical batches updated after every batch run is enough to produce same results as the non-moving window approach for MLR model. Incorporation of model prediction confidence bounds into MLR model with sliding window and averaged reference trajectory revealed significantly improved performance pattern.

Then, a strategy to modulate the Q and R ratio in the objective function was proposed. An adaptive weighted parameter was added to the Q and R ratio. The ratio was either adapted to the increasing batch numbers or error magnitude. For the continuously decreasing ratio for all the 20 batches, impressive results were attained for PLS and PCR models. There were improvement in the stability and batch to batch end product concentration for both these models. ILC with MLR model did not respond well to this strategy. For the error adaptive method, there were also improvements in the batch to batch biomass concentration and system stability for PCR and PLS models.

The reliable ILC using batch-wise adaptive ratio gives improving results for using MLR models. Incorporation of model prediction confidence bounds (MPCB) increased the batch to batch biomass concentrations with and without disturbance and so improves the stability of the system. For ILC with PLS and PCR models, there were small improvement in the performance stability of the system. The incorporation of MPCB improves the reliability of ILC with all the three models.

8.2 Suggestions for future works

The presence of non-repetitive disturbances is unavoidable in a fermentation system. Batch to batch control is efficient in controlling future batch runs but has no control on the current batch run. In the event the disturbances randomly changes from one batch to another, batch to batch ILC would become incompetent and may amplify the repercussion of the disturbances. It would be interesting to study the integration of the proposed methods to within batch ILC with updated model. ILC is unique because it can be used for both within batch and batch to batch trials. The integration strategy could be a more robust and reliable control method.

On the other hand, the batch to batch ILC is mainly aimed at complementing the present PID control system in the industry. A simulation study on applying the proposed method as supervisory control system to a PID control system may be the next step ahead. This way the proposed method can be tuned to be used for present industrial application.

Convergence is closely related to economical production. In the industry, producing desired product specification within minimal trial to maximize product over cost yield and reaching the market soonest possible is highly desirable. In addition to that, with the rapidly changing consumer requirement on product specifications, it is vital that a control method is capable of rapid convergence while minimizing the effect of possible errors such as model-plant mismatches, repetitive and non-repetitive disturbances and/or initialization error. The ideal solution would be to attain the desired results on the very first trial. At the moment this ideal solution is not yet possible. Attaining the desired results in lowest possible batch trials is highly desirable in the industry especially for scale up plants.

For batch to batch ILC between 10 and 20 batch data is needed to develop a process model. In agile manufacturing environment, the process changes in short period of time. Product development has to be done in a very short period of time. In addition to that, the trial runs may be costly and so very few data is available. There could be just 2 or less than 5 batches data available. It is essential to develop a strategy on how batch to batch ILC proposed in this work can be used for limited batch data.

Performance assessment of the models used in batch to batch ILC with updated linearised model method should be done. The correlation between the model updates and corresponding input prediction has to be accessed to further investigate the efficiency of the proposed techniques. The developed control policy should be tested for reliability and stability. Then, the model-plant mismatch elimination competence could be analysed. Once a robust and reliable ILC method is identified, it could be applied in the real-time industrial plant and its competency should be analysed.

REFERENCES

Alford J.S. (2006). "Bioprocess control: Advances and challenges." Computers and Chemical Enginering, **30**: 1464-1475.

Amann N., Owens D.H. and Rogers E. (1996). "Iterative learning control for discrete-time systems with exponential rate of convergence." IEE Proceeding Part-D, Control Theory and Application, **143**: 217-224.

Arimoto S., Kawamura S. and Miyazaki F. (1984). "Bettering operation of robots by learning." Journal of Robotic Systems, **1**:123-140.

Arimoto S., Miyazaki F., Kawamura S., and Tamaki S. (1985). "Learning control theory for dynamical systems." In Proceedings of the 24th Conference on Decision and Control, pp. 1375-1380, Fort Lauderdale, Florida.

Ashoori A., Moshiri B., Khaki-Sedigh A. and Bakhtiari M.R. (2009). "Optimal control of a nonlinear fed-batch fermentation process using model predictive approach." Journal of Process Control, **19**: 1162-1173.

Astrom K.J., Panagopoulos H., and Hagglund T. (1998)."Design of PI controllers based on non-convex optimization," Automatica, **34**(5):585-601.

Bajpai R. and Reuss M. (1980). "A mechanistic model for penicillin production." Journal of Chemical Technology and Biotechnology, **30**: 330-344.

Baolin Z., Gongyou T., and Shi Z. (2006). " PD-type iterative learning control for nonlinear time delay system with external disturbance." Journal of Systems Engineering and Electronics, **17**(3): 600-605.

Berber R., Pertev C. and Turker M. (1998). "Optimization of feeding profile for baker's yeast production by dynamic programming." Bioprocess Engineering, 20: 263-269.

Beyeler W., DaPra E. and Schneider K. (2000). "Automation of Industrial Bioprocesses," History of Modern Biotechnology II, Advances in Biochemical Engineering/Biotechnology, **70**:139-164.

Bien Z. and Xu J. (1998). “Iterative learning control.” Boston:Kluwer Academic Publisher.

Bonvin D. (1998).”Optimal operation of batch reactors-a personal view.” Journal of Process Control, **8** (5 & 6): 355-368.

Bonvin D. and Srinivasan B. (2003). “Optimal Operation of Batch Processes Via the Tracking of Active Constraints.” ISA Transactions, **42**: 123-134.

Bouakrif F. (2010). “Reference model iterative learning control for nonlinear systems with repeatable and non-repeatable uncertainties.” Int. J. Adv. Manuf. Technology, **51**:1159-1169.

Bristow D.A., Tharayil M. and Alleyne A.G. (2006). “A Survey of Iterative Learning Control:A Learning-Based Method For High-Performance Tracking Control.” IEEE Control Systems Magazine, 96-114.

Cai Z., 2009. Iterative Learning Control: Algorithm Development and Experimental Benchmarking. PhD thesis. School of Electronics and Computer Science, University of Southampton.

Camacho J., Pico J. and Ferrer A. (2007) “Self-tuning run-to-run optimization of fed-batch processes using unfold-PLS.” AIChE Journal, **53**(7):1789-1804.

Campbell W.J., Firth S.K., Toprac A.J. and Edgar T.F. (2002). “A comparison of run-to-run control algorithms.” Proceedings of the American Control Conference, May 8-10.

Caramihai M. and Severin I. (2009). “Quality fed-batch bioprocess controls a case study.” World Academy of Science, Engineering and Technology, **56**:674-677.

Cazzador, L. (1988). “On the optimal control of fedbatch reactors with substrate-inhibited kinetics.” Biotechnnol. Bioengineering, **31**: 670-674.

Chachuat B. (2007). “Nonlinear and dynamic optimization: From theory to practice.” Automatic Control Laboratory, EPFL, Switzerland.

Chen H.F. and Fang H.T. (2004). "Output tracking for nonlinear stochastic systems by iterative learning control." IEEE Transactions on Automatic Control, **49** (4): 583-587.

Chen C.K. and Hwang J. (2006). "PD-type iterative learning control for trajectory tracking of a pneumatic X-Y table with disturbances." JSME International Journal (Series C), **49**(2): 520-526.

Chen Y.Q. and Moore K.L. (2002). "PI-type iterative learning control revisited." Proc. of the American Control Conference, 1262-1267, Anchorage, Alaska.

Chien C.J. (2006). "An adaptive PID-type iterative learning controller for nonlinear systems with non-repeatable control tasks," Transactions of the Chinese Institute of Engineers, Series A, **29**(2): 279-287.

Cueli J.R. and Bordons C. (2008). "Iterative nonlinear model predictive control. Stability, robustness and applications." Control Engineering Practice, **16**: 1023-1034.

Dadebo S.A. and McAuley K.B. (1995). "Dynamic optimization of constrained chemical engineering problems using dynamic programming." Computer Chemical Engineering, **19**:513-525

Daramola M.O. and Zampaka L., (2008). "Experimental study of the production of biomass by *Sacharomyces cerevisiae* in a fed batch fermentor." African Journal of Biotechnology, **8**:1107-1114.

Dinh Van T., Freeman C. T., Lewin P.L. and Tan Y. (2012). "Convergence and robustness of a point-to-point iterative learning control algorithm." In, 51st IEEE Conference on Decision and Control, Maui, Hawaii, USA, 10 - 13 Dec 2012. 6pp. (In Press).

Feng M. and Glassey J. (2000)."Physiological state-specific models in estimation of recombinant *Escherichia coli* fermentation performance." John Wiley & Sons, Inc.:495-503.

Flores-Cerrillo J. and MacGregor J.F. (2003). "Within batch and batch-to-batch inferential adaptive control of semi-batch reactors: a partial least squares approach." Ind. Eng. Chem. Res., **42**: 3334-3345.

Fu P. and Barford J.P. (1992). "Simulation of an iterative learning control system for fed-batch cell culture processes." Cytotechnology, **10**:53–62.

Gao F., Yang Y. and Shao C. (2001). "Robust iterative learning control with application to injection molding process." Chemical Engineering Science, **56**:7025-7034.

Geladi, P. and Kowalski B.R. (1986). "Partial least squares regression: a tutorial." Analytica Chimica Acta, **185**, 1-17.

George S., Larsson G., Olsson K. and Enfors S-O. (1998). "Comparison of the Baker's yeast process performance in laboratory and production scale." Bioprocess Engineering, **18**:135-142.

Gosling I. (2003). "Optimizing productivity in multiple batch facilities." Genetic Engineering News, September, www.chemsim.com.

Gunst R.F. and Mason R.L. (1980). "Regression Analysis and Its Application - A Data-Oriented Approach." Marcel Dekker, Inc.

Hong J. (1986). "Optimal substrate feeding policy for a fed-batch fermentation with substrate and product inhibition kinetics." Biotechnology and Bioengineering, **28**:1421-1431.

Huang W-H., Shieh G.S. and Wang F-S. (2010). "Run-to-run optimization of fed-batch fermentation of ethanol production." Chemical Engineering Technology, 33(9):1488-1494.

Jiang Z., Droms K., Geng Z., Casnocha S., Xiao Z., Gorfien S., Jacobia S.J. (2012). "Fed-batch cell culture process optimization." BioProcess International, **10**(3): 40–45.

Ji G. and Luo Q. (2005). "An open-closed-loop PID type iterative learning control algorithm for uncertain time-delay systems," Proc. of the International Conference on Machine Learning and Cybernetics,1154-1159.

Hong J. J. (2011). "Multivariate statistical modelling for fault analysis and quality prediction in batch processes." PhD thesis. School of Chemical Engineering and Advanced Materials, Newcastle University.

Hong J.J. and Zhang J. (2009). "Quality prediction for a fed-batch fermentation process using multi-block PLS." The EU-KOREA Conference on Science and Technology, Reading, UK: 155-162.

Johnson A. (1987). "The control of fed-batch fermentation Processes – A survey." International Federation of Automatic Control, **23**(6):691-705.

Karakuzu C., Turker M. and Ozturk S. (2006). "Modelling, on-line state estimation and fuzzy control of production scale fed-batch baker's yeast fermentation." Control Engineering Practice, **14**:959-974.

Lee J.H., Lee K.S. and Kim W.C. (2000). "Model-based iterative learning control with a quadratic criterion for time-varying linear systems." Automatica, **36**: 641-657.

Lee K., Lee J.H., Yang D.R. and Mahoney A.W. (2002). "Integrated run-to-run and on-line model-based control of particle size distribution for a semi-batch precipitation reactor." Computers Chemical Engineering, **26**:1117-1131.

Lee K.S. and Lee J.H. (1997). "Model predictive control for nonlinear batch processes with asymptotically perfect tracking." Computers and Chemical Engineering, **21**: s873-s879.

Lee K.S. and Lee J.H. (2003). "Iterative learning control-based batch process control technique for integrated control of end product properties and transient profiles of process variables." Journal of Process Control, **13**: 607-621.

Lee K.S., Chin I. S., Lee H.J and Lee J.H. (1999). "Model predictive control technique combined with iterative learning for batch processes."AIChE Journal, October, **45**(10).

Lei F., Rotboll M., Jorgensen S.B. (2001). “A biochemically structured model for *Saccharomyces cerevisiae*.” Journal of Biotechnology, **88**:205–221.

Madady A. (2008). “PID type iterative learning control with optimal gains.” International Journal of Control, Automation, and Systems, **6** (2): 194-203.

Martin E.B. and Morris A.J. (1996). “Non-parametric confidence bounds for process performance monitoring charts,” Journal Process Control, **6** (6):349 358.

Mezghani M., Roux G., Cabassud M., Dahhou B., Le Lann M.V. and Casamatta G. (2001). “Robust iterative learning control of an exothermic semi-batch chemical reactor.” Maths and Comps in Sim.,**57**:367–385.

Moore K.L. (2001). “An observation about monotonic convergence in discrete-time, P-type iterative learning control.” Proc. of the IEEE International Symposium on Intelligent Control, pp. 45-49, MX, USA.

Moore K.L., Chen Y.Q., and Bahl V. (2001) “Feedback controller design to ensure monotonic convergence in discrete-time, P-type iterative learning control.” Proc. of the 4th Asian Control Conference, pp. 440-445, Singapore.

Moore K.L. (2006). “An introduction to iterative learning control.” EGES 504/604A Seminar, Colorado School of Mines, January 24.

Morari M. and Lee J.H. (1999). “Model predictive control: past, present and future.” Computers and Chemical Engineering, **23**:667-682.

Morris A.J. and Zhang J. (2009). “Performance monitoring and batch to batch control of biotechnological processes.” *In:* Nicoletti, M.C., Jain, L.C, ed. Computational Intelligence Techniques for Bioprocess Modelling, Supervision and Control. Berlin; New York: Springer-Verlag: 281-310.

Muske K.R., Badlani M., Dell’Orco P.C. and Brum J. (2004). “Open-loop optimization for batch reaction systems: a case study.” Chemical Engineering Science, **59**: 1167-1180.

Neeleman R. (2002). "Biomass performance monitoring and control in biopharmaceutical production." Ph.D. Thesis, Van Wageningen University.

Ng C.W. and Hussain M.A. (2004). "Hybrid neural network-prior knowledge model in temperature control of a semi-batch polymerization process." Chemical Engineering and Processing, **43**:559–570.

Owens D. H. and Daley S. (2008). "Iterative learning control-monotonicity and optimization." Int. J. Appl. Math. Computing. Sci., **18**(3): 279-293.

Park K.H., Bien Z., and Hwang D.H. (1999). "A study on the robustness of a PID-type iterative learning controller against initial state error." International Journal of Systems Science, **30** (1):49-59.

Pertev C., Turker M. and Berber R. (1997). "Dynamic modelling, sensitivity analysis and parameter estimation of industrial yeast fermenters." Computers Chemical Engineering, **21**:S739-S744.

Phan M.A. and Longman R.W. (1988). "A mathematical theory of learning control for linear discrete multivariable systems." Proceedings of the AIAA/AAS Astrodynamics Specialist Conference, Minneapolis, MN: 740-746.

Ratcliffe J.D., Hatonen J.J., Lewin P.L.,Rogers E.,Harte T.J., and Owens D.H. (2005). "P-type iterative learning control for systems that contain resonance," International Journal of Adaptive Control and Signal Processing, **19** (10):769-796.

Reiss R., Wojsznis W. and Wojewodka R. (2010). "Partial least squares confidence interval calculation for industrial end-of-batch quality prediction." Chemometrics and Intelligent Laboratory Systems, **100**: 75-82.

Renard F. and Wouver A.V. (2008). "Robust adaptive control of yeast fed-batch cultures." Journal of Computers and Chemical Engineering, **32**:1238-1248.

Riad et al, 2009….Riad G.H., Yousef A.H. and Sheirah M.A. (2009). "Supervisory control system for a fermentation plant," IEEE: 205-210. (DOI: 978-1-4244-4113-6/09)

Rieger M., Kappeli O. and Fiechter A. (1983). "The role of limited respiration in the incomplete oxidation of glucose by *Saccharomyces cerevisiae*." Journal of General Microbiology, **129**:653-661.

Rogers E. (2008). "Robustness of iterative learning control-algorithms with experimental benchmarking." Bulletin of the Polish Academy of Sciences, Technical Sciences, **56**(3): 205-215.

Rotem Y., Wachs A. and Lewin D.R.(2000). " Ethylene compressor monitoring using model-based PCA." AlChE Journal, **46**:1825–1836.

Saab S.S. (2003). "Stochastic P-type/D-type iterative learning control algorithms." International Journal of Control, **76**(2):139-148.

Salgado A.M., Folly R.O.M. and Valdman B. (2001). "Biomass monitoring by use of a continuous on-line optical sensor." Sensors and Actuators B: Chemical, **75** (1–2): 24–28.

San, K. Y. and Stephanopoulos, G. (1986). "The effect of growth rate delays in substrate-inhibited kinetics on the optimal profile of fed-batch reactors." Biotechnology Bioengineering, **28**: 356-361.

Schugerl K. (2001). "Progress in monitoring, modelling and control of bioprocesses during the last 20 years." Journal of Biotechnology, **85**:149-173.

Sendin J.O.H., Muras I.O., Alonso A.A. and Banga J.R. (2006). "Improved optimization methods for the multiobjective design of bioprocesses." Industrial & Engineering Chemistry Research, **45**(25):8594-8603.

Shukla P.K. and Pushpavanam S. (1998). "Optimisation of biochemical reactors analysis of different approximations of fed-batch operation." Chemical Engineering Science, **53**(2):341-352.

Sonnleitnert B. and Kappeli O. (1986). "Growth of *Saccharomyces cerevisiae* is controlled by its limited respiratory of capacity: Formulation and verification of a hypothesis*." Biotechnology and Bioengineering, **28**: 927-937.

Srinivasan B. and Bonvin D. (2007). "Controllability and stability of repetitive batch processes." Journal of Process Control, **17**: 285-295.

Srinivasan B., Bonvin D., Visser E. And Palanki S. (2002). "Dynamic optimization of batch processes II. Role of measurements in handling uncertainty." Computers and Chemical Engineering , **27**:27-44.

Srinivasan B., Primus C., Bonvin D. and Ricker N. (2001). "Run-to-run optimization via control of generalized constraints." Control Engineering Practice, **9**:911-919.

Tan K.K., Zhao S., Huang S., Lee T.H. and Tay A. (2009). "A new repetitive control for LTI systems with input delay." Journal of Process Control, **19**: 711-716.

Tian Y., Zhang J. and Morris J. (2002). "Optimal control of a batch emulsion copolymerisation reactor based on recurrent neural network models." Chemical Engineering and Processing, **41**:531-538.

Tousain R., Meche E.v-d. and Bosgra O. (2001). "Design strategies for iterative learning control based on optimal control." Selected Topics in Signals, Systems and Control, **12**:1-8

Uchiyama M. (1978). "Formation of high-speed motion pattern of a mechanical arm by trial." Transactions of Society for Implementation and Control Engineers, **14**:706-712.

Verduyn C., Stouthammer A.H., Scheffers W.A. and van Dijken J.P. (1991). "A theoretical evaluation of growth yields of yeasts." International Journal of Genetics and Molecular Microbiology, **59**: 49-63.

Verwoerd M.H.A. (2005). "Iterative learning control: a critical review."PhD thesis. Dutch Institute of Systems and Control (DISC).

Vilas, C., García, M.R., Fernández, M.R., Balsa-Canto, E., Banga, J.R., Alonso, A.A. (2004). "On systematic model reduction techniques for dynamic optimization and robust control of distributed process systems." Computer Aided Chemical Engineering, **18**(C): 841-846.

Vlassides S., Ferrier J.G. and Block D.E. (2001). “Using historical data for bioprocess optimization: Modelling wine characteristics using artificial neural networks and archived process information.” John Wiley & Sons Inc., Biotechnology Bioengineering, **73**: 55-68.

Waissman J.V., Youssef C.B. and Vaquez G.R. (2002). “Iterative learning control for a fed batch lactic acid reactor.” IEEE International Conference on Systems, Man and Cybernetics, **6**:

Wang D. (2000). “On D-type and P-type ILC designs and anticipatory approach,” International Journal of Control, **73** (10): 890-901.

Wang Y., Chu J., Zhuang Y., Wang Y, Xia J and Zhang S. (2009). “Industrial bioprocess control and optimization in the context of systems biotechnology,” Biotechnology Advances, **27** (6): 989–995.

Wang D. and Srinivasan R. (2009). “Multi-model based real-time final product quality control strategy for batch processes.” Computers and Chemical Engineering, **33**:992-1003.

Weigand, W. A. (1981) “Maximum cell productivity by repeated fed-batch culture constant yield case.” Biotechnology Bioengineering, **23**: 249 266.

Wold S., (1978). “Cross validatory estimation of the number of components in factor and principal components models.” Technometrics, **20**: 397-404.

Xiong Z. and Zhang J. (2003). “Product quality trajectory tracking in batch processes using iterative learning control based on time-varying perturbation models.” Industrial Engineering Chemical Research, **42**: 6802-6814.

Xiong Z. and Zhang J. (2005) a. “A batch-to-batch iterative optimal control strategy based on recurrent neural networks models.” Journal of Process Control, **15**: 11-21.

Xiong Z. and Zhang J. (2005)b. “Neural network model-based on-line re-optimisation control of fed-batch processes using a modified iterative dynamic programming algorithm.” Chemical Engineering and Processing, **44**:477-484.

Xiong Z., Zhang J., Wang X. and Xu Y. (2005). "An integrated batch-to-batch iterative learning control and within batch control strategy for batch processes." American Control Conference, Portland, OR, USA.

Xiong Z., Xu Y., Zhang J. and Dong J. (2008). "Batch-to-batch control of fed-batch processes using control-affine feed-forward neural network." Neural Comput. & Applic. **17**:425–432

Xu J. X., Lee T. H. and Zhang H. W., 2004. "Analysis and comparison of iterative learning control schemes." Engineering Applications of Artificial Intelligence, **17**: 675–686.

Xu J.X. and Tan Y., (2003). "Linear and nonlinear iterative learning control."Springer Berlin Heidelberg NewYork.

Yuzgec U., Turker M. and Hocalar A. (2009). "On-line evolutionary optimization of an industrial fed-batch yeast fermentation process." ISA Transactions, **48**: 79-92.

Zhang H. (2008a). "Optimal control of a fed-batch yeast fermentation process based on least square support vector machine." Int. J. Engineering Systems Modelling and Simulation, **1**(1): 63-68.

Zhang J. (2004). "A Reliable Neural Network Model Based Optimal Control Strategy for a Batch Polymerization Reactor." Industrial & Engineering Chemistry Research, **43** (4):1030-1038.

Zhang J. (2005). "Batch process modeling and optimal control based on neural network models," Acta Automatica Sinica, **31**(1):19-31

Zhang J. (2008b). "Batch-to-batch optimal control of a batch polymerisation process based on stacked neural network models." Chemical Engineering Science, **63**: 1273-1281.

Zhang J., Nguyan J. and Morris J. (2008). "Batch to batch iterative learning control of a fed-batch fermentation process using linearised models." 10th International Conference on Control, Automation, Robotics and Vision, Hanoi, Vietnam, 17-20 December. 745-750.