Neural Model-Based Advanced Control of Chylla-Haase Reactor

Magdi Mohammed Nabi

In partial fulfilment of the requirement of Liverpool John Moores University for the degree of Doctor in Philosophy

March 2015

Abstract

The objective of this thesis is to develop advanced control method and to design advanced control system for the polymerization reactor (Chylla-Haase) to maintain the high accurate reactor temperature. The first stage of this research start with the development of mathematical model of the process. The sub-models for monomer concentration, polymerization rate, reactor temperature and jacket outlet/inlet temperature are developed and implemented in Matlab/Simulink.

Four conventional control methods were applied to the reactor: a Proportional – Integral-Derivative (PID), Cascade control (CCs), Linear-Quadratic-Regulator (LQR), and Linear model predictive control (LMPC). The simulation results show that the PID controller is unable to perform satisfactorily due to the change of physical properties unless constant re-tuning takes place. Also, Cascade Control the most common control method used in such processes cannot guarantee a robust performance under varying disturbance and system uncertainty. In addition, LQR and linear MPC methods lead to better results compared with the previous two methods. But it is still under an assumption of the linearized plant.

Three advanced neural network based control schemes are also proposed in this thesis: radial basis function RBF neural network inverse model based feedforward-feedback control scheme, RBF based model predictive control and multi-layer perception (MLP) based model predictive control. The major objective of these control schemes is to maintain the reactor temperature within its tolerance range under disturbances and system uncertainty. Satisfactory control performance in terms of effective regulation and robustness to disturbance have been achieved.

In the feedforward-feedback control scheme, a neural network model is used to predict reactor temperature. Then, a neural network inverse model is used to estimate the valve position of the reactor, the manipulated variable. This method can identify the

controlled system with the RBF neural network identifier. A PID controller is used in the feedback control to regulate the actual temperature by compensating the neural network inverse model output. Simulation results show that the proposed control has strong adaptability, robustness and satisfactory control performance. These advanced methods achieved the much improved control performance compared with conventional control schemes.

The main contribution of this research lies in the following aspects. The MPC theory is realised to control Chylla-Haase polymerization reactor. Two adaptive reactor models including the RBF network model and MLP model are developed to predict the multiple-step-ahead values of the reactor output. Their modelling ability is compared with that of the models with fixed parameters and proven to be better. The RBF neural network and the MLP is trained by the recursive Least Squares (RLS) algorithm and is used to model parameter uncertainty in nonlinear dynamics of the Chylla-Haase reactor. The predictive control strategy based on the RBF neural network is applied to achieve set-point tracking of the reactor output against disturbances. The result shows that the RBF based model predictive control gives reliable result in the presence of some disturbances and keeps the reactor temperature within a tight tolerance range around the specified reaction temperature.

Moreover, RBF neural network based model predictive control strategy has also been used to reduce the batch time in order to shorten the reaction period. RBF neural network is considered as a prediction model for control purpose which is based to minimize a cost function in order to determine an optimal sequence of control moves. The result shows that the RBF based model predictive control gives reliable result in the presence of variation of monomer and presence of some disturbances for keeping the reactor temperature within a tight tolerance range around the specified reaction temperature without harming the quality of the temperature control.

Acknowledgement

I would like to express my special appreciation and thanks to my supervisor Prof. Ding-Li. Yu, you have been a tremendous mentor for me, also I would like to thank him for being patient with me during my thesis.

A special thanks to my family. Words cannot express how grateful I am to my father, my mum, and my lovely wife, for all of the sacrifices that you have made on my behalf. Your prayer for me was what sustained me thus far.

Again, my warmest thanks go to my wife, my son and my daughter for their love, understanding, and patience, during my study.

Brothers, sisters, and my friend Bashir thank you so much for your unconditional support and encouragement.

In memory of my uncle Hussain, my cousin Hamed, my best friend Bassem.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Semi-batch polymerization reactor

Discontinuous reactors (batch and semi-batch) are widely used in the production of fine chemicals, pharmaceuticals, specialties and other high value products because of their flexibility in operation mode. This type of reactor is industrially important and particularly well-matched for the production of polymers of varying grades, whose quality is measured in terms of strength, process ability, etc.

The chemical reactors can be roughly classified into three categories, based on three ideal models, continuously stirred tank reactor, tubular reactor, and batch and semi-batch reactor. Semi-batch and continuous reactors are probably the most frequently used types of reactors in the chemical industry (Bequette, 2003), although continuous reactors are preferred in the chemical industry because of their potential for good quality control and large production capacities. The semi-batch mode of operation has proven to be very attractive for polymerization. Typically, this mode of operation does not require as much expertise for operation and maintenance as the continuous mode. It also offers more flexible control capabilities for the quality of the polymer produced. Moreover, use of a semi-batch reactor intrinsically permits more stable and safer operation than a batch or continuous operation because of the characteristics it has; in addition to better yields and selectivity, gradual addition or removal assists in controlling temperature, particularly when the net reaction is highly exothermic (Seborg et al., 2004, Nanda and Pharm, 2008).

From the practical point of view, safety and product quality are the mostly interesting aspects. Process control engineers have developed considerable expertise in continuous processes, characterized by steady-state operation condition, but the application of this expertise to discontinuous processes rarely achieves comparable

success. Control and optimization of batch processes are real challenges for process control engineers because of some technical and operational considerations.

Polymerization is the process of reacting monomer molecules together in chemical reactions to form three dimensional networks of polymer chains. The main goal of this process is to achieve quality products, which are complicated due to the process exhibiting dynamic behaviour.

In the last 20 years, polymer manufacturers have been working to improve the quality of their products and the efficiency of their operations in order to improve their ability to characterize the physical properties of various polymer products and to quantitatively understand the influence of reaction conditions on the polymer produced. These efforts have required a much better understanding of polymerization kinetics and optimum recipes. The scope of this work is not to investigate new facts about the polymerization reaction or its physical properties, or aspects of reactor design, but to effectively utilize certain known facts to guide the course of polymerization processes. Therefore, finding out the best process recipes, typical feed rate (i.e., monomer) and temperature profile, and effectively maintain them during the batch time is very important. Consequently, it requires a process control and optimization technique.

Strong interactions between process variables, strong nonlinear behaviour, long and frequent time delays, unanticipated changes in process characteristics, and inequality constraints on process variables are all process control problems that can be handled by applying advanced process control and optimization techniques (Seborg et al., 2004; Edgar et al., 2001).

This work deals with modelling and control of a Chylla-Haase polymerization reactor. The corrected model given by (Graichen et al., 2006) has been used for simulation. As the process is exothermic in nature, strict control action needs to be taken so that the temperature can be constantly maintained. A very precise temperature

control is required as a monomer feed, disturbances and multi-product poses a significant demand on the control, so it is necessary to improve the control strategy of such a process in order to ensure that the end product will be of an acceptable quality.

Several researchers have addressed control problems with a Chylla-Haase reactor. (Seborg et al., 2004, Graichen et al., 2005, Richards and Congalidis, 2006) used cascade control which is commonly used in batch semi-batch reactors. It provides robust control but often lacks control performance related to the required temperature tolerance. Feed-forward and feedback, it has also been used in a Chylla-Haase reactor resulting in the best approach since the feedback control provides the necessary compensation for the effect of model measurement in accuracies and the feed-forward control reduces the effect of measured disturbances (Richards and Congalidis, 2006). Moreover, model predictive control is one of the best solutions that have been used with chemical processes in general. One of its best advantages is allowing the controller to deal with the exact model of the real process dynamics, implying a much better control quality and considering plant behaviour over future horizon in time (Jalili-Kharaajoo and Araabi, 2003).

To model such a highly nonlinear process, one of the most important tools is an Artificial Neural Network (ANN). Many research activities have been carried out to solve the problem of modelling batch and semi-batch processes and various modelling schemes have been reported in literature (Fernandes and Lona, 2005). In this work, for modelling a Chylla-Haase polymerisation reactor a Neural Network (NN) approach has been implemented. To ensure temperature control within the tolerance range in order to guarantee the end product will be of an acceptable quality, the neural network inverse model has been used. This is done by combining the conventional control techniques with the evolving RBFNN methodologies. Also, MPC controller has been designed and adapted to different batch conditions. The MPC control concept based

on a neural network model presented in this contribution has been used to estimate the future output of the plant and to minimize the cost function based on the error between the predicted output of the process and the reference trajectory.

As illustrated in this work, with the proposed MPC controller the magnitude of error is reduced far below the requirement ±0.6K and it shows similar performance for different products operating under different disturbance effects.

1.2 Research Motivation

The Chylla-Haase benchmark problem is highly nonlinear and time varying in nature. The polymerization reactor considered is a multi-product and multi-batch process. Within a batch, this semi-batch process operates in different phases, such as feed phase and hold phase for each polymer. These features make the process complex and interesting.

- The temperature control of each following batch of a particular product becomes more difficult because of heat transfer surface fouling, therefore tight temperature control needs to be implemented.
- The heat transfer coefficient decreases significantly during a batch due to an increasing batch viscosity, and from batch to batch due to surface fouling.
- Production of multiple products A and B characterized by different reaction kinetics.
- The reaction kinetics are nonlinear and subject to the gel effect (Graichen et al., 2006).
- The monomer feed starts and stops abruptly, hence the tight control of temperature using conventional methods is not satisfactory. Thus, it requires the design of an advanced control scheme.

Because of such problems mentioned above need to be considered, we planned to include advanced control (i.e. adaptive model predictive control MPC).

1.3 Aims and Objectives

The aims of the research is develop model-based methods for temperature regulation and batch time reduction for the Chylla-Haase reactor to against its non-linearity, strong disturbances and time varying features. The specific objectives of the project are:

- Develop a Neural Network model that captures the entire dynamics of the process.
- Design an adaptive inverse model control of reactor temperature based on an RBF neural network.
- Develop temperature control based on an adaptive MPC to cope with model uncertainty and evaluate the performance.
- Reduce batch time using an MPC based on an RBF neural network.
- Evaluate the developed methods using computer simulation.

1.4 Thesis Outline

The thesis is organized into ten chapters. Chapter 1 is an introduction chapter which gives an overview of the work carried out. It states the motivation behind the research work and the importance of semi-batch reactors. It also gives the aims and objectives of the research work. Chapter 2 gives a comprehensive review of the latest advances in control technology for semi-batch reactor modelling and control.

Chapter 3 describes the process operation and model simulation. It also presents the dynamics of the model. The process model consists of dynamic material balance and dynamic energy balance, along with the data for polymer A and polymer B being presented. Set up a Simulink model of the reactor, which can be used to evaluate the developed control methods. The data for two products, polymer A and B are both used to generate two models. Chapter 4 describes development of primary

control methods and presents the design of the conventional PID control, cascade control, LQR control and linear model predictive control (LMPC).

Chapter 5 explains in detail the modelling of the Chylla-Haase polymerization process using neural networks. Two popular networks, the radial basis function (RBF) network and the multi-layer perceptron (MLP) network, are used in this section. Analysis of the simulation results is also given to show the capability of the neural models. Chapter 6 proposes an inverse model-based feed-forward and feedback control scheme. The inverse model is based on an RBF neural network.

Chapter 7 presents an adaptive model that is used to predict reactor future output to determine the optimal control variables to control the temperature to the set-point. This model predictive control is based on the developed neural network model and uses non-linear optimization. Chapter 8 studies the ability of MPC to reduce the batch time as well as to control the temperature within the performance bounds. In the original setup of the Chylla−Haase benchmark problem it is not possible to reduce the batch time because the monomer inlet flow is constant. A modified version of the benchmark problem where variable monomer feed was allowed is considered.

Chapter 9 summarizes the work carried out based on the achievements of the study and its contribution. It also discusses directions for future work.

1.5 Research Novelty and Originality

Existing investigations into the control of the Chylla-Haase reactor have considered cascade control, internal model control, neural inverse model control, etc. However, as the methods in these work cannot cope with model uncertainty and external disturbance, the existing methods are not robust against disturbance and can only cope with nonlinearities. When the reactor model uncertainty is significant, the control performance will not be satisfactory. The Chylla-Haase reactor has severe nonlinear

dynamics and multi-variables with significant coupling effects. Therefore, the model's uncertainties will be inevitable.

The novelty of the presented research is in its aim to develop adaptive model predictive control for the reactor. Firstly, the nonlinear model of the reactor will be developed using an RBF and MLP networks. These networks have been proven to be able to sufficiently model nonlinear dynamic systems and will therefore provide accurate multi-step ahead predictions of the reactor's behaviour. This will enable the model predictive control to cope with the nonlinearity and interactions between variables. Secondly, the RBF network model has been made adaptive in online mode, i.e. the model adapts its parameters using online input/output data to model uncertainties and time-varying dynamics. In this way, the control performance has not been degraded by time-varying dynamics and its robust against model uncertainty. Moreover, a possible time reduction using MPC based RBF technique also implemented.

The proposed control strategy has not been reported in terms of its application for the control of a Chylla-Haase reactor. Thus, the originality of the proposed project stands.

Chapter 2 Literature Survey

2.1 Introduction

There is a large quantity of literature in the area of modelling, control and parameter estimation for polymerization reactors. Many researchers have placed academic and industrial focus upon polymerization processes due to difficulties in operating such processes (i.e. high non-linearity). Different types of neural network modelling and control schemes have been reported for polymerization processes. But the Chylla-Haase reactor is specifically reviewed in this chapter.

Development and improvement of the model and control methods for the Chylla-Haase reactor are considered to be the aims of this research. With the objective of improving the performance of polymerization reactor controllers, an essential literature survey has been carried out and it is described in this chapter.

Sections 2.2 to 2.5 of this chapter give a background to the different types of polymerization processes and also provide general knowledge about neural network modelling and control of the polymerization processes. Section 2.6 provides a further review of existing work regarding the control of the Chylla-Haase polymerization reactor, which is the main focus of this research.

2.2 Polymerization reactors

The demand for polymer products has much increased over the last 20 years and so the method of production has become more and more important. Therefore, polymer manufacturers have been working to improve the quality of their products and the efficiency of their operations so as to improve their ability to characterize the chemical products and develop more effective reactors. The aim of this work is not to investigate new facts about the polymerization reaction or its physical properties, but to find out the best controller design that leads to the best processes for producing quality products.

Different types of polymerization processes are discussed here to highlight their importance and the differences between them.

- Batch process

This is a process in which all reactants are added together at the beginning of the process and products are removed at the termination of the reaction. In this process, all mixture components are added at the beginning and no addition or withdrawal is made while the reaction is progressing. Batch processes are preferred for small-scale production of high priced products (Artin Hatzikioseyian and Remoundaki, 2005).

- Continuous process

This is a process in which the reactants and products are continuously added and withdrawn. Continuous production will normally result in lower production costs compared to batch production and it is simpler and easier to design. However, it faces the limitation of a lack of flexibility. Continuous reactors are usually preferred for large scale production (Artin Hatzikioseyian and Remoundaki, 2005).

- Semi-batch process

A semi-batch reactor is similar to a batch reactor but has the additional feature of continuous addition or removal of one or more components. In addition to better production and selectivity, gradual addition or removal assists particularly in controlling temperature. Thus, use of a semi-batch reactor permits more stable and safer operation than a batch reactor (Artin Hatzikioseyian and Remoundaki, 2005).

2.3 Process modelling

Modelling, optimisation and advanced control of chemical processes have more applications due to their capabilities to analyse the behaviour of real processes, impose certain operating conditions which would be impossible to carry out onsite due to safety or financial reasons, study existing processes in more detail and faster, compare

different alternatives without modifying the real plant, and make decisions based on results obtained (Seborg et al., 2004).

In order to design more efficient polymerization technology and to develop improved or new products, a better understanding is needed of polymerization reactions and process behaviour. This can be derived from the fundamental chemistry and physics of polymerization processes that can be used to calculate reaction rates and polymer architectural parameters. This is called first principles modelling. For certain polymerization systems, complex molecular structures are not appropriate for first principles modelling and hence empirical or semi-empirical models such as neural network models are the practical alternatives, as they do not require much information about internal structure (Won Jung Yoon et al., 2004).

Artificial neural networks have become a usual application in many areas of engineering and they are well suited for chemical processes due to their ability to describe multi-variable nonlinear models. Reactor modelling methods presented are based on the techniques of system identification for control purposes.

The use of neural networks in chemical engineering offers a potentially effective means of handling three problems: complexity, nonlinearity and uncertainties. Neural networks are well known for their ability and have been used to model and control nonlinear dynamic systems (Yu and Gomm, 2003). A neural network can be modelled to represent the behaviour of dynamical processes. For example, (Jalili-Kharaajoo and Araabi, 2003, Ng and Hussain, 2004) used a neural network to approximate unknown system dynamics by measuring the state variables of one batch for training. Kurt and Marcel (1998, cited in (Ng, 2004) and (Fazlur Rahman et al., 2000) have shown that the use of prior knowledge in combination with a neural network structure results in an optimal model. Models obtained via neural networks represent only the input-output

behaviour of the plant and carry little information about internal structure, therefore high accuracy and good flexibility can be achieved with these modelling techniques.

From a literature point of view, there are two fundamental classes of neural networks: feed-forward networks and recurrent networks that has at least one feedback loop. Both of these can be observed as system input-output mapping and consist of different layers that can be separated into the input layer, one or more hidden layers and the output layer. Each layer is made up of many neurons that transform the mapping information through their activation functions and weights. These parameters of neural networks are adapted via a training process which can make neural networks learn about the embedded environment (Howlett and Jain, 2001).

There are two types of training algorithm: off-line training algorithms and on-line training algorithms. On-line training algorithms are more accurate because the neural network will be adapted, especially with uncertainties and disturbances. The training neural network has to be tested using a set of pre-sampled testing data. The neural network can be used as a nonlinear model for the corresponding system.

Neural network based modelling and control has been a very active research area in recent years. Much research into polymerization reactor modelling and control using neural networks has been carried out over the past few decades and most of this research is based on feed-forward modelling including the back-propagation (BP) network and the radial basis function (RBF) network.

An RBF network is a kind of multi-layer feed-forward neural network with good performance. It has the capability of strong nonlinear mapping. The advantages of the simplicity of its structure are faster learning algorithms and better approximation capabilities (Zhang et al., 2011). Therefore, the RBF can be used to identify a model in the online mode with high accuracy.

Early work was based largely on the MLP network, except the similarities of topologies between RBF and MLP, RBF networks and MLP networks behaves very differently. Firstly, RBF networks are simpler than MLP networks, which may have more than three layers architectures, so the training process is generally faster than that of MLP networks. Secondly, RBF networks act as local approximation networks, because the network outputs are determined by specified hidden units in certain local receptive fields while MLP networks work globally, since the network outputs are decides by all the neurons. Finally, it is essential to set correct initial states for RBF networks while MLP networks use randomly generated parameters initially (Xie et al., 2011).

Efforts have been made to use advanced non-conventional control methods to develop and test alternative control schemes for improving operational performance. Some of these contributions are presented in the following paragraphs.

(Tian et al., 2001) modelled a batch polymerization reactor using a hybrid stacked recurrent neural network model. A novel stacked recurrent neural network architecture was presented that effectively integrated the knowledge acquired by different networks to obtain a better predictive model to express the difficult gel effect.

(Yuan et al., 2001) proposed a stacked recurrent neural network that effectively integrated the knowledge acquired by different networks to obtain a better predictive model for a batch polymerization reactor. They found through comparison with the best-single-network-based hybrid model that stacked recurrent neural network modelling demonstrated a strong capability for the modelling of a wide variety of nonlinear complex polymerization processes.

Yu, et al. developed a pseudo-linear radial basis function (PLRBF) network to model the real process of a laboratory-scaled three input three-output chemical reactor. Simulation results showed that the adaptive model significantly reduced the instant

modelling error for a time variant process or process-model mismatch (Yu et al., 2002, Yu and Yu, 2003).

(Shahrokhi and Pishvaie, 1998) used a feed-forward neural network to estimate the heat generation rate of polymerization reaction. This estimate was used in a feed-forward/feedback structure for controlling the reactor temperature and it showed a robust performance in that the neural network could predict the reaction heat generation rate very well.

Nonlinear modelling of the multilayer perceptron networks was applied to the prediction of polymer viscosity from real plant data by (Lightbody et al., 1994). In addition, some contributions that have aimed to minimize batch time by maximizing monomer conversion in order to save time can be found in literature as well. For example, a model based control scheme for batch time minimization was proposed in (Finkler et al., 2013).

2.4 Control of a polymerization process

Strong interactions between process variables, strong nonlinear behaviour, long and frequent time delays, unanticipated changes in process characteristics, and inequality constraints on process variables are all process control problems that can be handled by applying advanced control techniques (Seborg et al., 2004).

Control of polymerization reactors has become one of the most challenging issues in control engineering, as such a process is characterized by highly exothermal reactions with complex nonlinear dynamics and exact temperature control is often required. Also, there is a lack of online measurement of important variables such as heat transfer coefficient and reaction heat. Many articles have been published in the area of polymerization reactor control.

The PID controller was an early approach used to perform the desired control target for chemical processes because of its remarkable efficiency and the fact that it requires

minimal process knowledge for its design (Richards and Congalidis, 2006). However, chemical processes usually show a large time constant and time varying dynamics, and sometimes it is necessary to track complex set-point trajectories. Under these situations, the standard PID controller might not achieve the required performance (Flores-Cerrillo and MacGregor, 2005, Shamekh and Lennox, 2010).

Cascade control is commonly used in industrial polymerization reactors, especially when there is nonlinear system behaviour. It consists of a master controller for the reactor temperature and an underlying slave controller for the cooling circuit. Cascade control provides robust operation but often lacks in control performance concerning the required strict temperature tolerances (Seborg et al., 2004, Graichen et al., 2005, Richards and Congalidis, 2006).

2.4.1 *Feed-forward feedback control*

Feed-forward control has also been used widely in chemical processes, especially in polymerization reactors, because normal cascade control can be significantly improved by additional feed-forward control. (Graichen et al., 2006, Richards and Congalidis, 2006) simulated a cascade controlled reactor together with nominal feed-forward control, showing that temperature error can be reduced significantly compared to when conventional cascade control is used without a feed-forward part. A combination of feed-forward and feedback control results in a better approach, since feedback control provides the necessary compensation for the effect of model mismatch and feed-forward control reduces the effect of measured disturbances (Richards and Congalidis, 2006).

A fuzzy logic model supervised artificial neural network feed-forward PI feedback control strategy was developed by (Huafang et al., 1995). This strategy was applied to a batch polymerization reactor. The ANN part consisted of inverse back propagation and a recurrent network to calculate the feed-forward signal and to estimate heat

reaction respectively, and the fuzzy part supervised the final feed-forward output to enhance the reliability of the ANN.

2.4.2 *Model predictive control*

One of the most successful control techniques for industrial applications is model predictive control (MPC). Most studies to date have considered model predictive control for the control of polymerization reactors (Bequette, 2003). The main reasons for the popularity of predictive control strategies is that they allow the controller to deal with the exact model of real process dynamics, implying much better control quality. The presence of constraints is the feature that most clearly distinguishes MPC from other process control techniques, leading to tighter control and a more reliable controller. Furthermore, MPC considers plant behaviour in the future. Thus, the effect of both feed-forward and feedback disturbances can be anticipated and eliminated, which permit the controller to drive the process output more closely to the reference trajectory (Jalili-Kharaajoo and Araabi, 2003, Jalili-Kharaajoo and Araabi, 2011).

Predictive controllers are used in many areas where high quality control is required (Qin and Badgwell, 2003). Model based predictive control (MPC) refers to a class of control algorithms which are based on a process model. MPC can be applied to such systems with multivariable, non-minimum phase, open loop unstable, and nonlinear, as well as systems with a long time delay. Constrained model predictive control becomes the standard algorithm for advanced control in process industries.

There are several versions of MPC techniques, including model algorithmic control (MAC) (Richalet et al., 1978), dynamic matrix control (DMC) (Cutler and Ramaker, 1980), and internal model control (IMC) (Garcia and Morari, 1985). Although the above techniques differ from each other in some details, they are fundamentally the same; all of them are based on linear process modelling. If the nonlinear model is

available, the computational requirements are expected to be very high (Garcia et al., 1989), especially for nonlinear MIMO processes.

The theory behind MPC can be summarized as follows:

The future outputs for a determined horizon N, called the prediction horizon, are predicted at each instant time t using the process model. These predicted outputs $y(t+k)$, $k = 1 \ldots \ldots N$ depend on the known values up to instant t (past input and outputs) and on the future control signals $u(t+k)$, $k = 0 \ldots . . N-1$, which are those to be sent to the system and to be calculated (Wang, 2009, Robert Haber, 2011).

The set of future control signals are calculated by optimising a determined criterion in order to keep the process as close as possible to a reference set point. This criterion usually takes the form of a quadratic function of the errors between the predicted output signal and predicted reference trajectory. The control effort is included in objective function in most cases.

The control signal $u(t)$ is sent to the process while the next control signals calculated are rejected, because at the next sampling instant $y(t+1)$ is already known and the calculations are repeated, yielding a new control and new predicted state path. The prediction horizon keeps being shifted forward and for this reason MPC is also called receding horizon control (Camacho and Alba, 2013). Figure 2.1 presents the strategy of MPC.

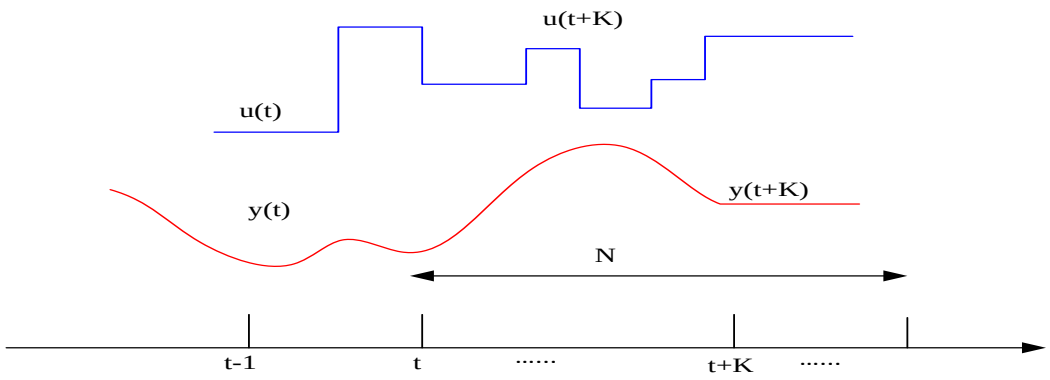


Figure 2.1. MPC strategy

The structure of a typical MPC system is shown in Figure 2.2. In brief, the theory behind it is that the process model is used to generate a prediction of future subsystem behaviour. In each time step, past measurements and inputs are used to estimate the current state of the system. An optimal control problem is solved over a finite future horizon of time steps. Only the first input move is injected into the model. In the subsequent time step, the system state is re-estimated using new measurements, and then the optimization is repeated, as demonstrated in Figure 2.3 (Venkat, 2006).

As mentioned before, predictive control considers future reference and/or measurable or observable disturbance and predicted the output signal sequences, unlike non-predictive control (like PI(D) control) which works with current and through the internal memory, as well as with past values (Robert Haber, 2011).

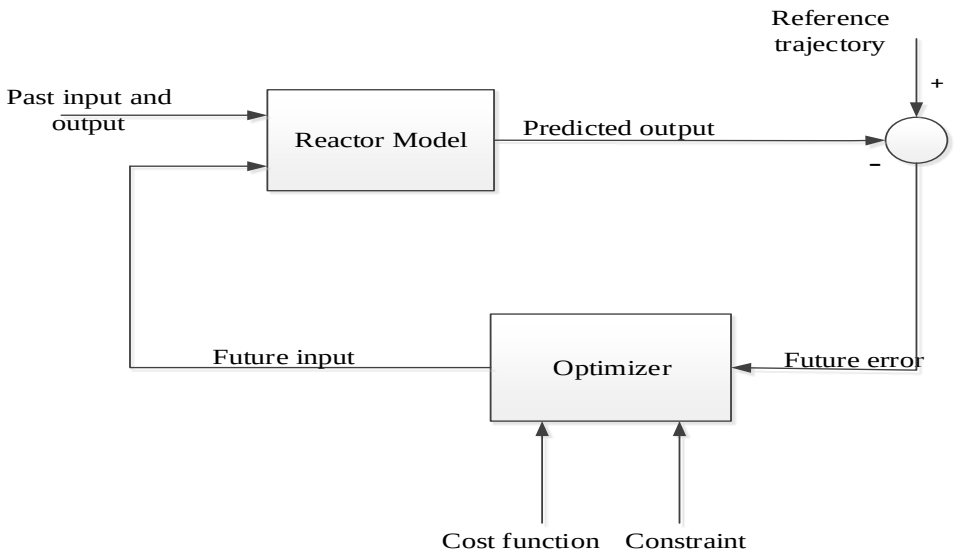


Figure 2.2. Typical structure of MPC

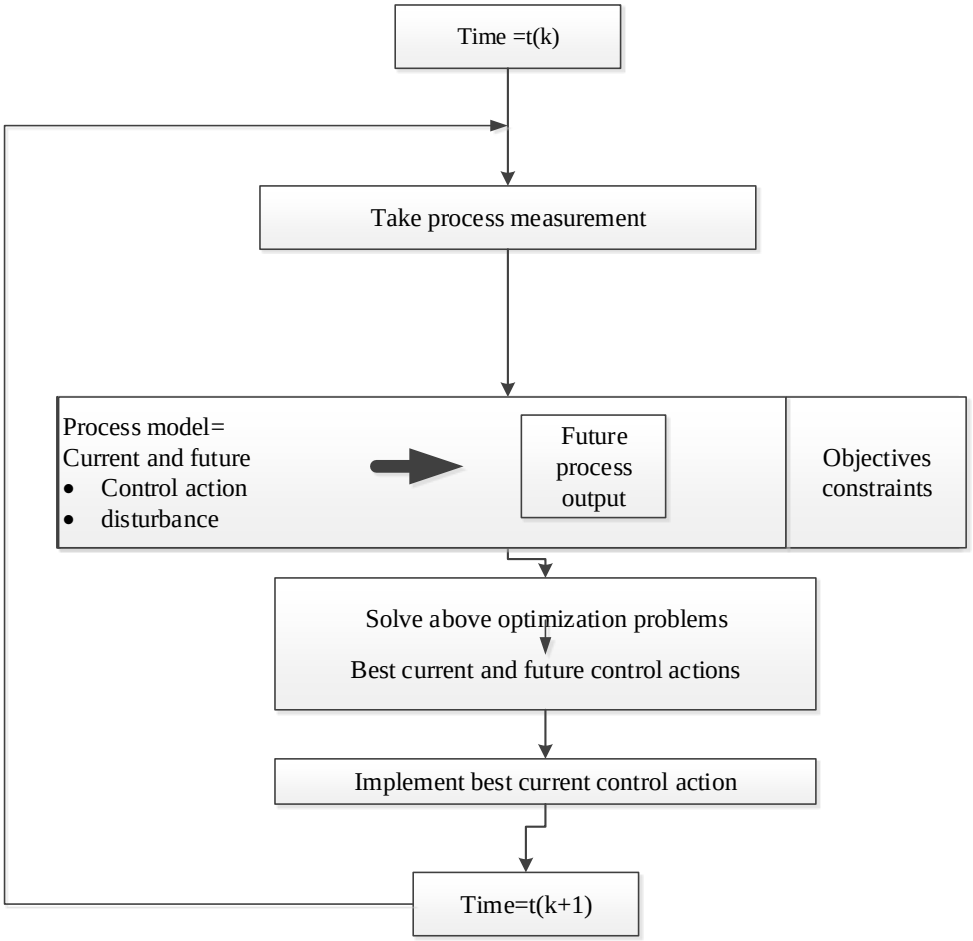


Figure 2.3. Model predictive control scheme

Model predictive control based on identification is widely used as an alternative to nonlinear predictive control, since the dynamic behaviour of a polymerization reactor is inherently nonlinear. NMPC shows good performance despite the fact that it is quite complicated to design the NMPC (Cervantes et al., 2003).

Some examples from literature of MPC based identification will now be discussed. Takagi-Sugeno (TS) fuzzy models are particularly suitable for application with an MPC algorithm. These models can be utilized in different MPC algorithms. Fuzzy models are directly used to formulate the optimization problem but lead to nonlinear formulation. Another kind of model for predictive control is fuzzy DMC (FDMC). In a simulation and/or experiment, DMC can be modified in an easy way using the fuzzy approach, offering better control performance (Marusak, 2009).

Neural Network Model Predictive Control (NNMPC) is another typical and straightforward application of neural networks to nonlinear control. When a neural network is combined with the MPC approach, it is used as a forward process model for the prediction of process output (Vasičkaninová and Bakošová, 2009). NNMPC is one of the most successful methods used to control nonlinear process systems. (Bahman ZareNezhad and Aminian, 2011) have proven that the developed NNMPC structure has the ability to track set-point changes in process output and the disturbance rejection. The main objectives of a predictive control strategy using neural predictors is to estimate the future output of the plant and to minimize the cost function based on the error between the predicted output of the processes and the reference trajectory (Jalili-Kharaajoo and Araabi, 2011).

The NN-MPC structure consists of three components in addition to the plant: a neural network model, an optimizer used as controller, and a filter for a selected time horizon. The controller optimizes the plant input by using the neural network model to calculate controller moves and predicting plant output. The neural network model is trained to predict the plant output for use in optimisation (Sakata et al., 2009).

A model is used to predict future plant outputs, based on past and current values of the plant input and output and the proposed optimal future control actions. These actions are calculated by the optimizer, taking into account the cost function, as well as constraints. The process model plays, in consequence, a decisive role in the controller. The chosen model must be capable of capturing the process dynamic so as to precisely predict future outputs, as well as being simple enough to implement and understand.

Model predictive control based on an adaptive PLRBF model was applied by (Yu and Yu, 2003) to a laboratory-scaled three input three-output chemical reactor. The PLRBF used much less hidden layer nodes while predicting much more accurately than a standard RBF network when used to model multivariable real processes.

(Pottmann and Seborg, 1997, Haichen and Zhijun, 2006) applied model predictive control based on an RBF neural network, first using RBF to model a nonlinear system using the Least Squares and stepwise estimation methods and using the model to train the RBF network.

An on-line optimization of semi-batch reaction systems through MPC in combination with state estimation can be found in (Helbig et al., 1998). Also, in (Helbig et al., 1996), a nonlinear model predictive control in combination with an extended Kalman filter was applied. The MPC was used to track the set-point and the extended Kalman filter was used to estimate certain parameters, such as heat transfer coefficient and reaction heat.

In (Bhat and Banavar, 1998), the controller design methodology was based on process inverse dynamics modelling. The learning database for the controller training was generated in an open-loop fashion and training of the network was carried out off-line by considering the future plant outputs as the reference set points.

A nonlinear adaptive controller consisting of a nonlinear controller based on differential geometric concepts, combined with an extended Kalman filter for on-line estimation and control was applied by (Clarke-Pringle and MacGregor, 1997). This was compared with a conventional PID controller and feed-forward feedback controller.

An inverse neural network in a hybrid scheme used to model and control the semi-batch polymerization process can be found in (Ng and Hussain, 2004). A more robust model predictive control was proposed in (Lucia et al., 2012).

Predictive control based on neural networks of the chemical process was applied by (Haichen and Zhijun, 2006). The predictive model was constructed by using a feed-forward neural network in the algorithm. The Levenberg-Marquardt was incorporated into the back-propagation algorithm for training the neural network off-line and

modifying the predictive model online. This scheme was applied to control a CSTR reactor and the results showed a good match between the process model and neural network model. In addition, the controller was able to force the process output to follow the target values.

2.4.3 *MPC based Volterra model*

In addition to the above mentioned MPC based on RBF neural networks, several studies have proposed predictive control based on the second-order Volterra series model. The Volterra series is an important model used in the design of nonlinear MPC. The model parameters can be obtained from either Carleman linearization of a nonlinear fundamental mode or input-output data when the fundamental is not available (Yingnong et al., 2000).

A model predictive control algorithm based on the second order Volterra model was investigated by (Maner and Doyle, 1997), who presented simulation results for the control of two isothermal continuous stirred tank reactors. The responses of both reactors controlled with a linear MPC scheme and the second order Volterra MPC scheme were compared to a desired linear reference trajectory. In the majority of cases examined, the response obtained by the Volterra controller followed the reference trajectory more closely.

(Medina-Ramos and Nieto-Chaupis, 2010) proposed a strategy of control based on a nonlinear model predictive control (NMPC) for nonlinear systems, together with a respective technique of system identification using the Volterra model where its respective kernels were projected onto orthogonal basis functions. The performance of the proposed scheme showed efficient results.

A nonlinear model-predictive control scheme based on the autoregressive-plus Volterra model was employed by (Maner and Doyle, 1997). This scheme showed how

the Volterra model was straight forward for identification. One of the main advantages of using this scheme is that global stability conditions are available for the model.

(Dorado and Bordons, 2005) applied model predictive control based on the Volterra model to a CSTR reactor. Efficient results were obtained for the optimization problem when the Volterra model was compared with the NLP technique.

An NMPC strategy based on a second-order Volterra series model was applied to greenhouse temperature control by (Gruber et al., 2011). The strategy was used to approximate the nonlinear process and then the model was applied to the controller as a reference trajectory to eliminate the disturbance effect.

2.5 Estimation of parameters

When parameters of a dynamic model are efficiently estimated using input-output data, the quality of the model is improved. It is, therefore, very important for modellers of chemical engineering processes to have access to efficient parameter estimation algorithms, since this will eventually lead to better product quality and safer processes (Varziri, 2008).

The Kalman filter has been extensively used for state estimation of dynamic systems. The development of a Kalman filter for state and parameter estimation of a biotechnical process was discussed by (Bellgardt et al., 1986). However, several studies have shown the inadequacy of these methods for highly non-linear processes, such as (Soroush, 1997), because these methods use linear approximation of the nonlinear process model.

An extended Kalman filter and nonlinear dynamic parameter estimation were surveyed to estimate key kinetic parameters of a continuous polymerization process by (Sirohi and Choi, 1996). Parameter estimation using an extended Kalman filter has been shown to perform robustly, even in the presence of substantial measurement noise because of greater flexibility in tuning parameters, while parameter estimator based on

nonlinear programming techniques (nonlinear dynamic parameter estimator) has stronger sensitivity to measurement noise due to a lack of sufficient tuning parameters. (Wang et al., 1993) proposed an adaptive control of input/output linearizable systems, together with an extended Kalman filter (EKF). An EKF of continuous-discrete form was employed on-line to offer estimates of state variables and it was shown that an observer is often necessary for reconstruction of missing states in the implementation of a nonlinear controller.

2.6 Existing research on the Chylla-Haase reactor

The Chylla–Haase polymerization reactor is widely accepted as a benchmark process for the evaluation of control strategies for batch reactors. Many researchers have paid attention to the Chylla-Haase reactor's problems and have applied a wide range of control methods to the reactor. Some of the control methods that have been implemented are as follows:

- PID control: standard PID has already been employed in this kind of reactor to maintain the desired reaction temperature throughout the batch, although the nonlinear kinetics provide a challenge for replacement of the PID with a more advanced method (Chylla and Haase, 1993).
- PI cascade control: provides robust operation but often lacks in the area of control performance because of strict temperature requirement (Graichen et al., 2005).
- The extension of a conventional cascade control using a feed-forward has been proposed by (Graichen et al., 2005, Graichen et al., 2006). Improvement of performance control has been proven compared with standard cascade feedback control. It has also been proven that the reaction heat is estimated without a time delay.

- Adaptive exact linearization control (AELC) has been applied to the Chylla-Haase reactor using a Sigma-Point Kalman Filter (SPKF) by (Beyer et al., 2008), which has shown that the SPKF provides accurate estimates of internal reactor variables, which are essential for the performance of the applied control concept with the reactor.
- Analysis and nonlinear model predictive control of the Chylla-Haase reactor has been carried out by (Helbig et al., 1996), combining MPC and EKF to replace the cascade controller. It has been shown that the control strategy is robust against uncertainties and batch-to-batch variations.
- (Clarke-Pringle and MacGregor, 1997) have proposed nonlinear adaptive control for the Chylla-Haase reactor. A nonlinear controller has been designed based on differential geometric control theory and the control formulation has involved identifying the time varying factor, such as the heat transfer coefficient using EKF, where the uncertainty has been assumed to be a random signal.
- Usually the Chylla-Haase reactor is controlled by cascade control, but a new scheme proposed by (Finkler et al., 2012) aims to control and reduce the batch time of the process. The idea behind this scheme is to add a PI controller to the original cascade controller. This PI controller manipulates the monomer inlet flow in order to keep the cooling usage at a pre-established optimal level and using a NMPC scheme as the reference solution. Simulation results show that the proposed scheme is able to robustly operate the process.
- (Bhat and Banavar, 1998) have presented a controller design methodology based on process inverse dynamics modelling using a neural network. The neural network generates plant input by using past information of the system states. In other words, the neural network is trained to model the inverse

dynamics of the Chylla-Haase reactor to generate the control variable C which is the manipulated variable in the next sample time.

- Active disturbance rejection control (ADRC) has been propose by (Li et al., 2014). The central idea of ADRC is to treat the collective effect of internal and external uncertainties as the total disturbance in the input channel. An extended state observer (ESO) is then used to estimate and cancel this total disturbance, thus reducing the plant to a cascade integral form that can be easily controlled by a proportional−derivative (PD) controller.
- A multi stage nonlinear model predictive control has been presented by (Lucia et al., 2012). The idea behind this work is that the uncertainty is modelled by a tree of discrete scenarios and feedback information is taken into account by assuming that new information about the true state of the plant will become available at each sampling instant, and that the future control inputs can be adapted to the evolution of the uncertainties and hence, the error is largely reduced.
- Design of a self-tuning regulator for the Chylla-Haase reactor has been proposed by (Vasanthi et al., 2012). An unscented Kalman filter (UKF) has been used to estimate the reaction heat and heat transfer coefficient, which is used to calculate a trajectory for the cooling jacket temperature in order to follow a reactor temperature set-point. The robustness of this work has been proven by a comparison with a conventional cascade control. It has been found that the performance of the self-tuning cascade control is very good.
- (Vasanthi et al., 2011) have estimated the reaction heat as well as the heat transfer coefficient in the energy balance by using an artificial neural network. A self-tuning cascade control concept design calculates a trajectory for the

cooling jacket temperature in order to follow a predefined trajectory of the reactor temperature.

2.7 Summary

This chapter has reviewed previous work in the area of polymerization reactor modelling and control, neural network control theory, MPC control theory and estimation of nonlinear process parameters. Many different methods of dynamic polymerization reactor modelling have been reviewed, providing the fundamental knowledge for the new development for the semi-batch reactors.

The semi-batch polymerization process reported by (Chylla and Haase, 1993) and the corrected model given by (Graichen et al., 2006) have been simulated to show the efficiency of the modelling and control method proposed in this work. To control the polymerization process tight temperature control is required and due to the presence of time dependent parameters the adaptive control technique is necessary. Hence, online adaptive control is attempted in this work. The subsequent chapter describes the dynamic behaviour of the polymerization process and develops ordinary differential equation models using Matlab/Simulink.

Chapter 3 Chylla-Haase Dynamics and Simulation

In the last 20 years, polymer manufacturers have been working to improve the quality of their products and the efficiency of their operation to improve their ability to characterize the chemical products, to quantitatively understand the influence of the reaction conditions on the polymer produced and to develop more effective reactors. These efforts have required a much better understanding of polymerization kinetics and optimum recipes.

The process for this research work is a polymerization process, which is a benchmark problem, described by (Chylla and Haase, 1993) . This process is a multiproduct, semi batch polymerization reactor. Many complications, nonlinearities and constraints that exist transform the control problem into a challenging one. Industrial chemical reactor is a complex device in which heat transfer, mass transfer, diffusion and friction may occur along with chemical reaction and it must be safe and controllable.

In order to control such a process and get optimum recipes, a mathematical description of the process is needed. A chemical process can mathematically be described by heat and mass balances, resulting in a differential-algebraic equation (DAE) system. For this, a description of the process and mathematical model is followed in next section. (Nyström, 2007).

Nowadays simulations have big advantages with the increasing computation power and speed of the computers followed by the decreasing costs. The simulation process usually starts with the modelling of the system and the result of the modelling is the mathematical model of the system, which describes the most important variables and relations between them (Vojtesek and Dostal, 2009). The mathematical model of the examined Chylla-Haase reactor is described by the set of five nonlinear differential

equations and set of algebraic equation and it has been derived by (Chylla and Haase, 1993) and corrected by (Graichen et al., 2006).

3.1 Process Description

The industrial polymerization reactor proposed by Chylla-Haase consists of a stirred tank reactor with a working volume of 30 gallon, and is used to make specialty emulsion polymer in the process. The cooling jacket and cooling recirculation are with volume of 5.7 gallons. A common strategy is shown in Figure 3.1.

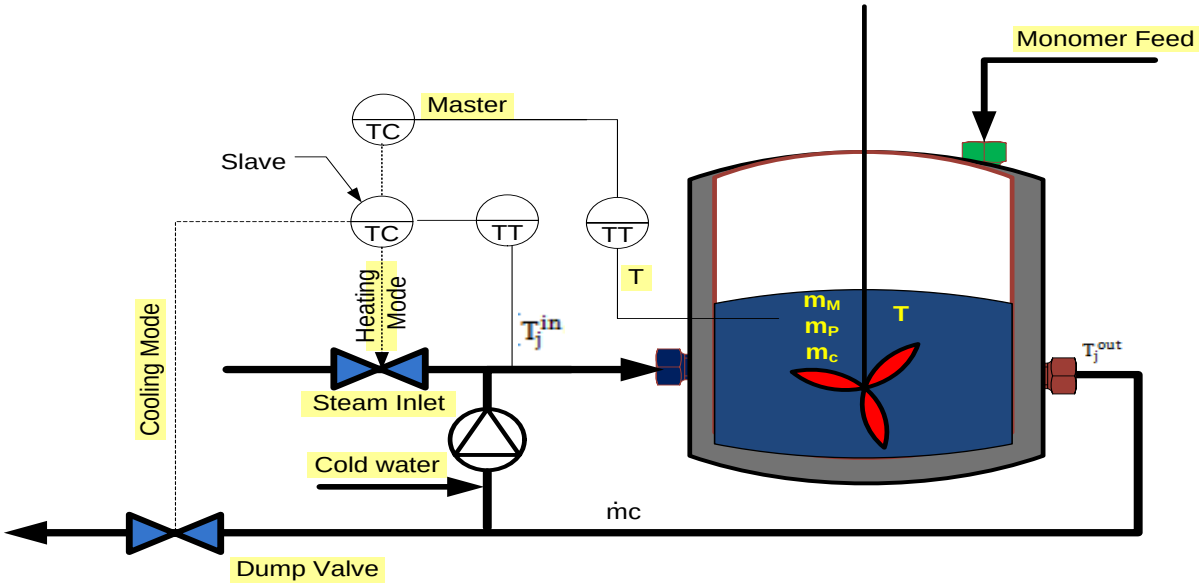


Figure 3.1 Chylla-Haase schematic

The polymerization process is simulated for a product which comprises a specific recipe which is given below.

The recipe for each batch of a specific polymer consists of a heating phase from 0 to 1800 s, a feed phase from 1800 to 6000 s, for product A

- Initial charges of polymer, monomer and water are placed into the reactor at ambient temperature T_{amb}.
- The temperature of the initial charge is raised to the reaction temperature set point T_{set}. at 1800 s.
- After 1800 s, monomer is fed into the reactor at 0.0075 kg/s until 6000 s.

- After the feed addition has stopped, the temperature of the reaction is held at its set point value T_{set}.

For the recipe B:

The recipe for each batch of a specific polymer consists of a heating phase from 0 to 1800 s, a feed phase from 1800 to 9600 s,

- Initial charges of polymer, monomer and water are placed into the reactor at ambient temperature T_{amb}.
- The temperature of the initial charge is raised to the reaction temperature set point T set at 1800 s.
- During the feed phase from 1800s to 5400s and from 7200s to 9600s feed pure monomer into the reactor at 6.048 x 10-3 Kg/s.
- After the first and second feed addition period is complete, hold at reaction temperature at T set = 353.160 K from 5400s to 7200s and 9600s to 12000s.

The temperature of the reactor is often ramped up from the ambient reactor charge conditions to a temperature where the reaction begins to take off. The heat released through the reaction must be removed by circulating cold water through the jacket, where both hot and cold jacket steam are available. When the jacket temperature controller output is between 0 and 50%, the valve is opened and cold water is inserted, and when the jacket controller output is between 50% and 100%, the valve is opened and steam is inserted (Helbig et al., 1996, Vasanthi et al., 2012).

This process includes five batches, between batches the product is removed but the reactor is not cleaned. Therefore, the heat transfer ability of the system is much lower in the fifth batch, compared to the first batch. After five batches, the polymer builds up on the reactor walls, which impede heat transfer.

3.2 Mathematical Model

Mathematical models of chemical and polymer processes can be very complex due to their typical characteristics including non-linearity, stochastic behaviour, time variation and also chemical reaction. Using the mathematical model to represent the real process allows the model user to study and understand the relationships between the elements of the system without having to manipulate the actual system. Thus, use of a model to investigate the working of a process, certainly give many advantages rather than using the real process.

The reactor simulation model used here has been developed using MATLAB/SIMULINK. The dynamic model of the reactor is defined by material balances (3.1) and (3.2) for the monomer mass $mM(t)$ and the polymer mass $mP(t)$.

$$\frac{dmM}{dt} = \dot{m}_M^{in} - R_p \tag{3.1}$$

$$\frac{dmP}{dt} = R_p \tag{3.2}$$

Where $\dot{m}_M^{in}$ is monomer feed rate and R_p is the rate of polymerization.

Usually the reaction rate is a function of temperature and represented by equation (3.3) and (3.4).

$$R_p = i * k * m_M \tag{3.3}$$

$$K = K_0 e^{-E/RT} (K_1 \mu)^{K_2} \tag{3.4}$$

Where K is the reaction rate, K_0 exponential factor, E the activation energy, R the ideal gas constant (8.314 $KJ/KmolK$), T the absolute temperature scale.

The energy balance (3.5) with the reactor temperature $T(t)$.

$$\frac{dT}{dt} = \frac{1}{\sum_i m_i C_{p,i}} \left[\dot{m}_M^{in} C_{p,M} (T_{amb} - T) - UA\left(T - T_j\right) - (UA)_{loss} (T - T_{amb}) + Q_{rea}\right] \tag{3.5}$$

$(i = M, P, W)$.

The terms on the right hand side account for the heat of the feed mixture, heat transfer through the jacket, heat loss to the environment and heat generation due to reaction, respectively.

For the definition of all variables and physical parameters, see the notation, Table 3.2 and the nomenclature in the appendix A.

The energy balances of the cooling jacket and the recirculation loop with the outlet and inlet temperatures T_j^{in} &T_j^{out} of the coolant C are shown in 3.6 and 3.7.

$$\frac{dT_j^{out}}{dt} = \frac{1}{m_c C_{p,c}}\left[\dot{m}_c C_{p,c}(T_j^{in}(t-\theta_1) - T_j^{out}) + UA(T - T_j)\right] \tag{3.6}$$

$$\frac{dT_j^{in}}{dt} = \dot{T}_j^{out}(t-\theta_2) + \frac{T_j^{out}(t-\theta_2) - T_j^{in}}{\tau_p} + \frac{K_p(c)}{\tau_p} \tag{3.7}$$

The viscosity of the reactor contents is given in (3.8) as a function of both mass function in the reacting mixture and its temperature.

$$\mu = c_0 e^{c_1 f} * 10^{c_2\left(\frac{a_0}{T} - c_3\right)} \tag{3.8}$$

The solid mass fraction f is readily calculated by (3.9).

$$f = \frac{m_p}{m_M + m_p + m_w} \tag{3.9}$$

The jacket heat transfer area can be calculated by the following equation:

$$A = \left(\frac{m_M}{\rho_M} + \frac{m_p}{\rho_P} + \frac{m_w}{\rho_W}\right)\frac{P}{B_1} + B_2 \tag{3.10}$$

The reactor-side film heat transfer coefficient is given in (3.11) as a function of wall viscosity.

$$\mu_{wall} = c_0 e^{c_1 f} * 10^{c_2\left(\frac{a_0}{T_{wall}} - c_3\right)} \tag{3.11}$$

The wall viscosity, μ_{wall}, is determined by (3.8) evaluated at the wall temperature T_{wall} which is simply the mathematics mean of reactor and jacket temperature

$T_{wall} = \frac{T + T_j}{2}$. The overall heat transfer coefficient, U, is calculated by using the simplified equation (3.12).

$$U = \frac{1}{h^{-1} + h_f^{-1}} \qquad \text{with } h = d_0 e^{d_1 \mu_w} \tag{3.12}$$

During successive batches of the same product, $h_f{}^{-1}$ should change as shown in Table 3.1.

TABLE 3.1. Reactor fouling factor

Batch	1	2	3	4	5
h_f^{-1}	0.0	0.176	0.352	0.528	$0.704 \mathrm{m^2 K\ kw^{-1}}$

Table 3.2 lists the variables and parameters of the polymerization reactor model. The parameter values of the model and the polymer are listed in Table 3.3.

TABLE 3.2. Notation of the Chylla-Haase reactor

$\dot{m}_M^{in}$	Monomer feed rate [kg/s]
$Q_{\mathrm{Re}a} = -\Delta H \bullet R_p$	Reaction heat [kw]
R_p	Rate of polymerization [kg/s]
$-\Delta H$	Reaction enthalpy [kj/kg]
U	Overall heat transfer coefficient [kw/m^2k]
A	Jacket heat transfer area [m^2]
$(UA)_{loss}$	Heat loss coefficient [kw/k]
$C_{p,M}, C_{p,c}, C_{p,p}$	Specific heat at constant pressure [KJ/kgk]
θ_1, θ_2	Transport delay in jacket and recirculation loop [s]
$T_j = {}^{T_j^{in} + T_j^{out}}\!/_{2}$	Average cooling jacket temperature [k]
$K_p(c)$	Heating/cooling function [k]
τ_p	Heating/cooling time constant [s]

The heating/cooling function $K_p(c)$ is defined by (3.13) and is a function of the valve position $C(t)$, the input function $K_p(c)$is a function depending on whether the reactor is cooled or heated. (Chylla and Haase, 1993, Beyer et al., 2008).

$$K_p(c) = \begin{cases} 0.8 \cdot 30^{-c/50}(T_{inlet} - T_j^{in}(t)), & C < 50\% \\ 0 & C = 50\% \\ 0.15 \cdot 30^{-c/50-2}(T_{steam} - T_j^{in}(t)), & C > 50\% \end{cases} \quad (3.13)$$

Various disturbances and uncertainties are specified in order to model the following practical problems with the control of polymerization reactors. The main uncertainties and disturbances that effect the process are summarized in four variables, the impurity factor i, the fouling factor $1/h_f$, the ambient temperature T_{amb} and time delay. The impurity factor i varies from 0.8 to 1.2 and describes the fluctuations in the reaction rate caused by impurities in the row materials. It is constant during one batch, but it changes randomly from batch to batch. The fouling factor $1/h_f$ is as described in table 3.1. (Chylla and Haase, 1993, Graichen et al., 2006, Vasanthi et al., 2012).

The ambient temperature that affects the monomer inlet feed and the initial conditions is different during summer and winter, for which more details are given in Table 3.3.

The delay times θ_1 and θ_2 of the cooling jacket and the recirculation loop may vary by ±25% compared to the nominal values in Table 3.3.

3.3 Matlab Simulink Model Development

Simulink is a part of Matlab that can be used to simulate dynamic systems and to facilitate model definition. In this section, model are created and developed according to the mathematical model equation (3.1)-(3.13).

The Simulink model of the process consists of four sub-models, material balances, recirculation loop, jacket and the actual temperature as shown in Figure 3.2 below.

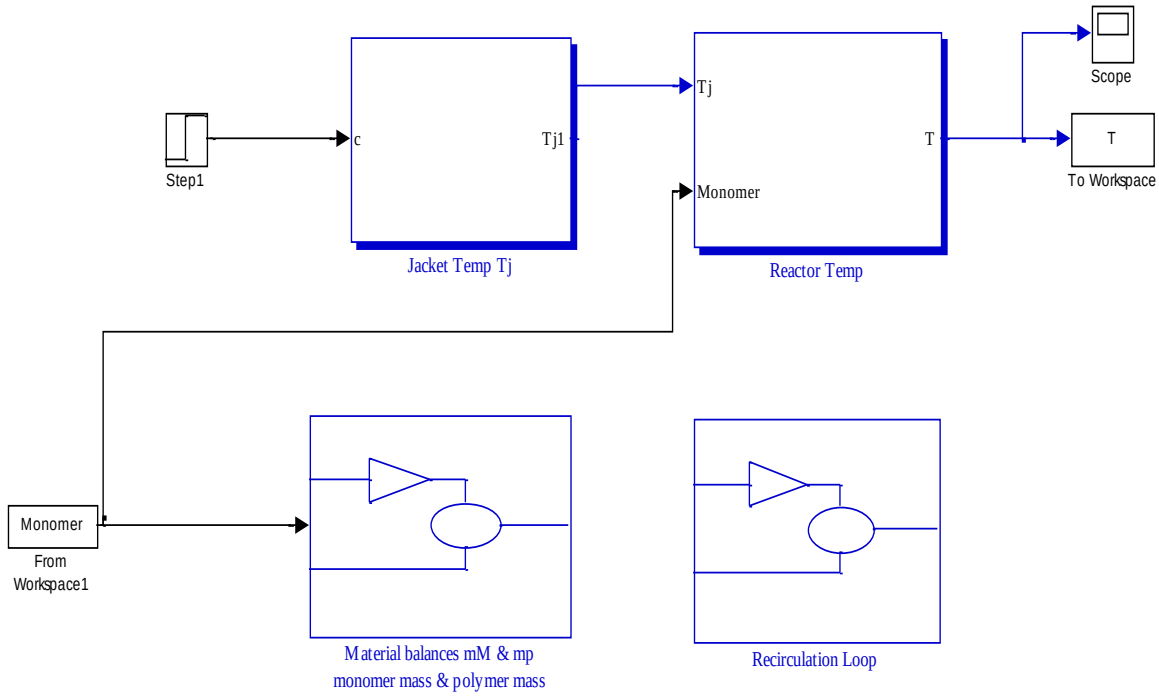


Figure 3.2 Chylla-Haase Simulink System

Based on the mathematical model described in previous section Figure 3.3 represents equation (3.1)-(3.4), (Material balance). The solution can be performed in SIMULINK, and the corresponding simulation diagram is:

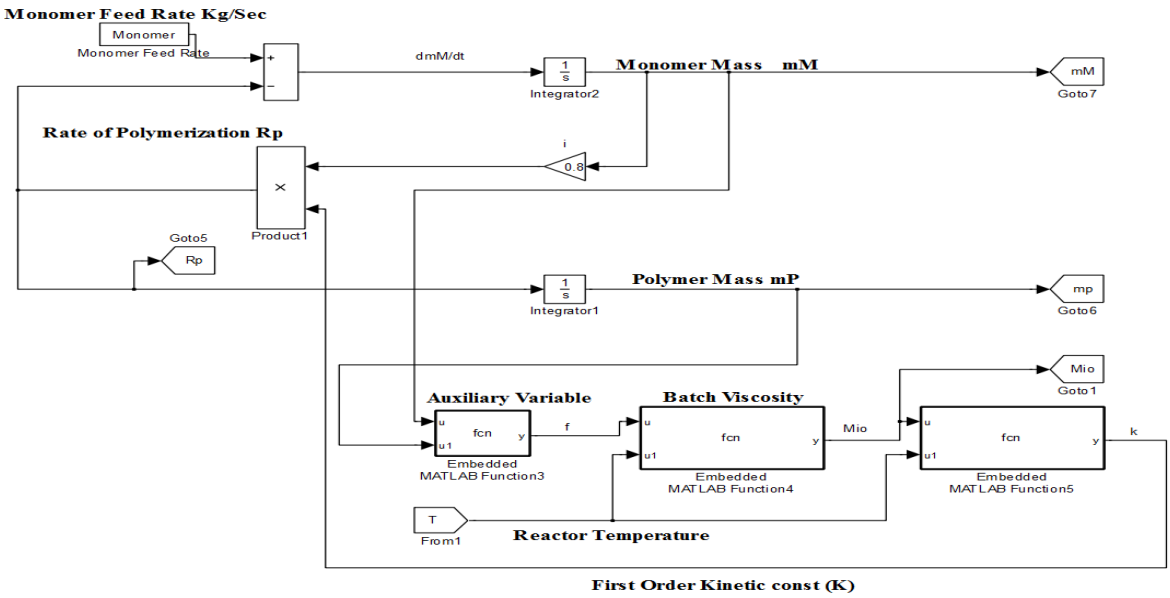


Figure 3.3 Simulink block diagram of material balance

Starting with initial conditions shown in table 3.3 (set up through the 'integrator' block), Embedded Matlab Function has been set to solve the empirical relation for the polymerization rate R_P and other algebraic equations.

Figure 3.4 shows the simulation model for equations (3.6),(3.7), and (3.13). Which describes the energy balance for Jacket temperature and heating/cooling function $K_p(c)$ for reactor valve position. All the variables are named and sent to the MATLAB workspace.

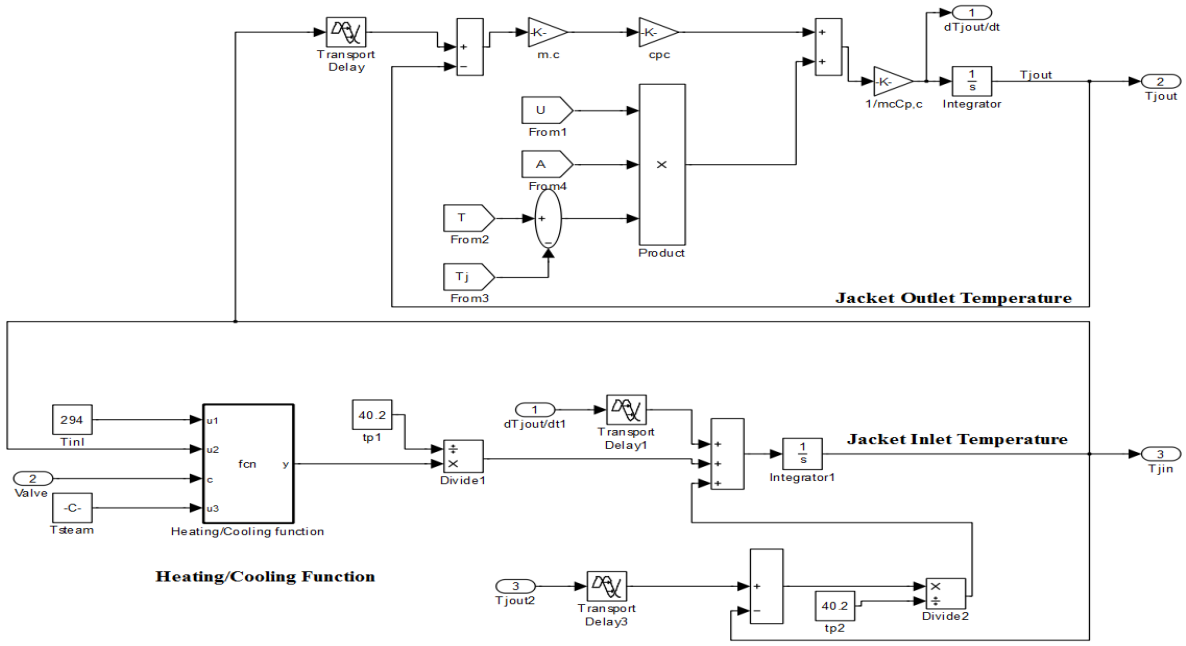


Figure 3.4 Simulink block diagram of jacket and recirculation loop

Figure 3.5 presents algebraic equations (3.8), (3.11), (3.12), it has been set to embedded Matlab function and when the system runs these equations are simultaneously solved.

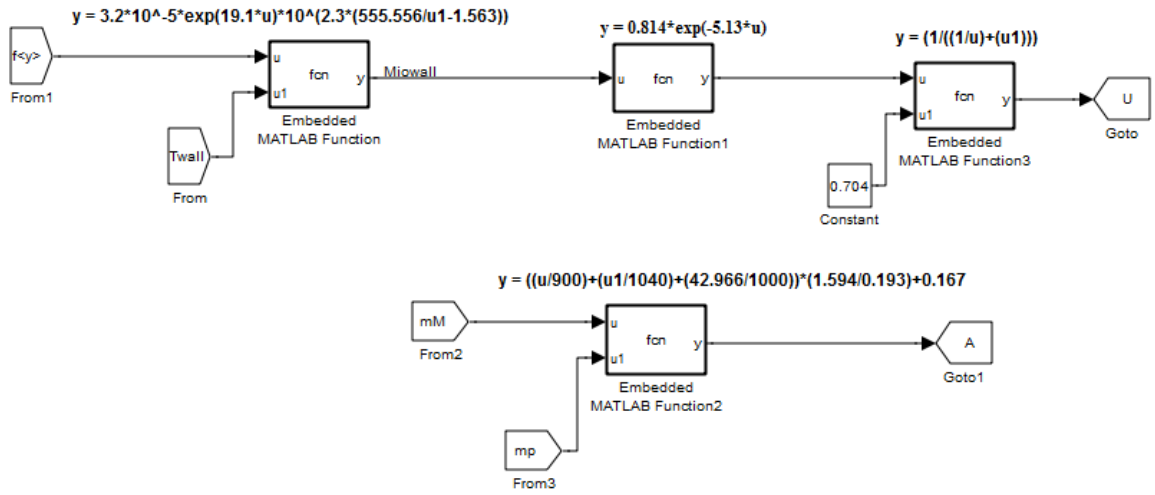


Figure 3.5 Overall heat transfer coefficient and Jacket heat transfer area

All previous blocks set in subsystem as shown in Figure 3.6.

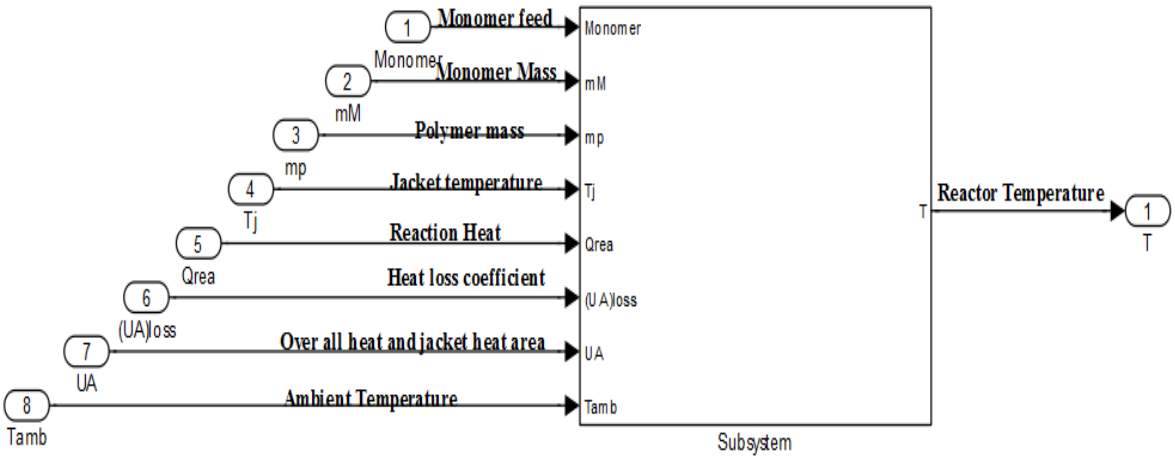


Figure 3.6 Subsystem of Reactor temperature

The reaction temperature is the main factor which affects the rate of the polymerization process and it has been chosen as the control variable. The considered state variables are the monomer, polymer and jacket temperature, and the manipulated variable is the valve position. The subsystem of the Chylla-Haase reactor model with two inputs and one output as going to be considered in this study multi-input/single output (MI/SO) presented in Figure 3.7. All optimum values of these variables have been calculated from model equations using a simulation program (MATLAB). When simulations are preformed, the equations in the embedded function are solved simultaneously with other blocks in the block diagram.

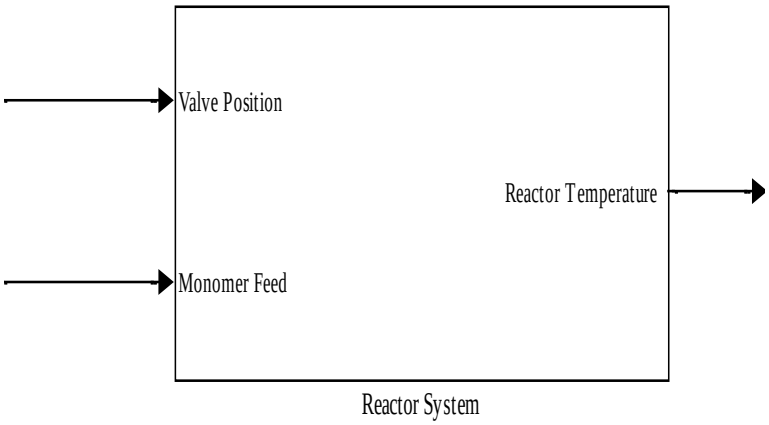


Figure 3.7 Reactor System

3.3.1 Open loop response of polymerization process for polymer A.

Open-loop responses show the process characteristics visually. The data shown in this section is based on product A (Table 3.3). To generate the open-loop (uncontrolled) response, a fully open valve has been set up in order to heat the reactor before the feed of monomer. Reactor temperature goes beyond the set-point as its expected because full steam is inserted. At the initial stage, the reactor temperature reaches its $450K$ when the valve is fully open. The reactor temperature keeps increasing until monomer feed stops even without an external heat supply, at the moment when monomer stop the temperature drops down (Figure 3.8, 3.9).

Figure 3.9 shows monomer feed versus time, it can be seen the effect of the monomer towards the temperature. The fast monomer conversion also causes the fast viscosity increase shown in Figure 3.10. At the end of the monomer feed period (time point at 6000 Sec), the viscosity has been increased which effect the heat transfer between the reactor and its contents.

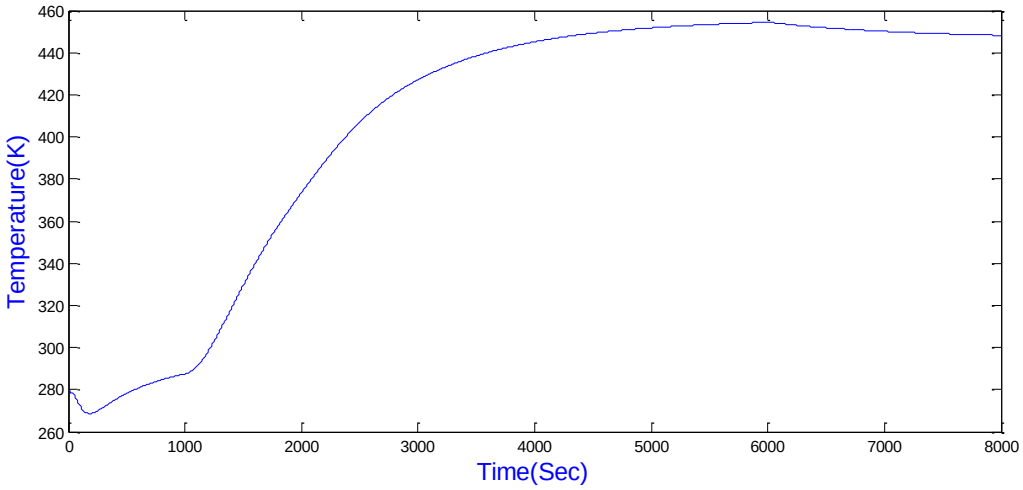


Figure 3.8 Open loop response of reactor temperature

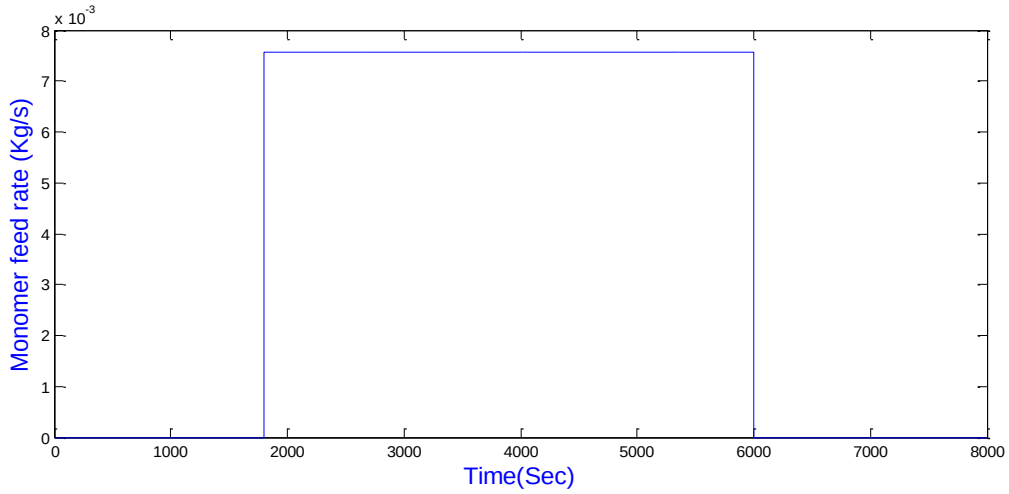


Figure 3.9 Monomer feed rate

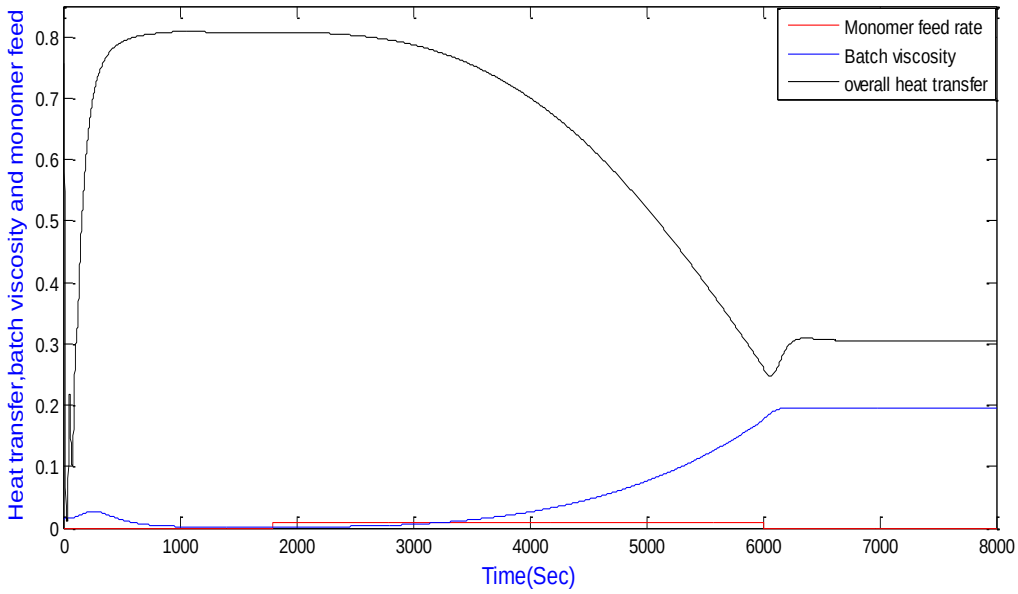


Figure 3.10 Monomer feed effects on viscosity and overall heat transfer coefficients

The opposite behaviour of the overall heat transfer coefficient presented in Figure 3.10. Since the overall heat transfer coefficient directly affects the amount of heat transferred from reactor to jacket, its behaviour simply indicates the nonlinearity of the process gain. Due to nonlinearity of reaction kinetics, the heat transfer coefficient sharply decreases during a batch as the viscosity builds at the reactor wall.

Polymer mass also increases nearly linearly during the feed phases of monomer and remains nearly constant during feed stop. This is called standstill reaction as shown in Figure 3.11 and monomer mass shown in Figure 3.12. Here the reactor must be cooled

down since the reaction release energy. This loss of energy to the environment must be replaced through a heating phase to keep the reactor temperature on the prescribed reaction temperature set-point.

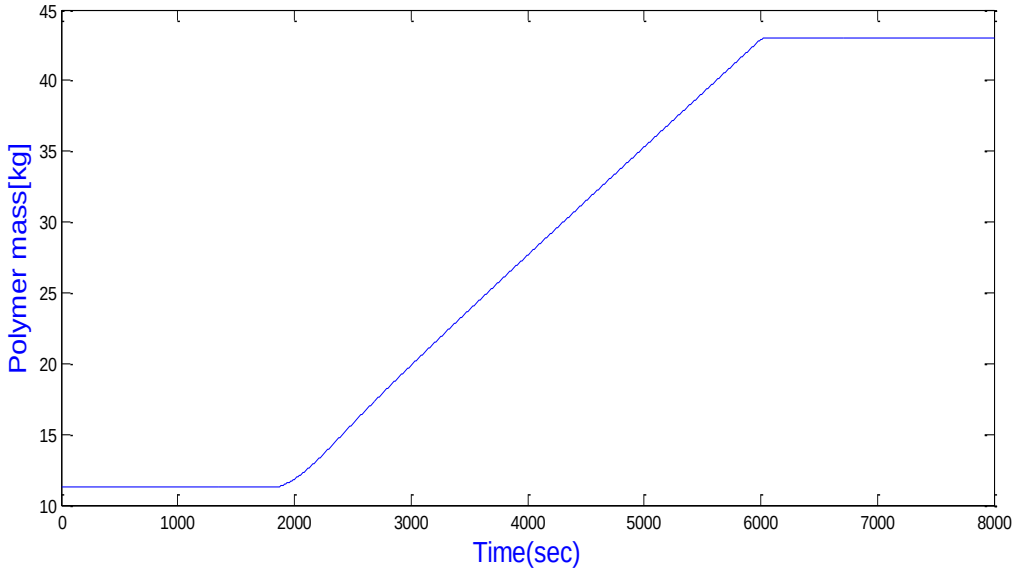


Figure 3.11 The change of Polymer mass

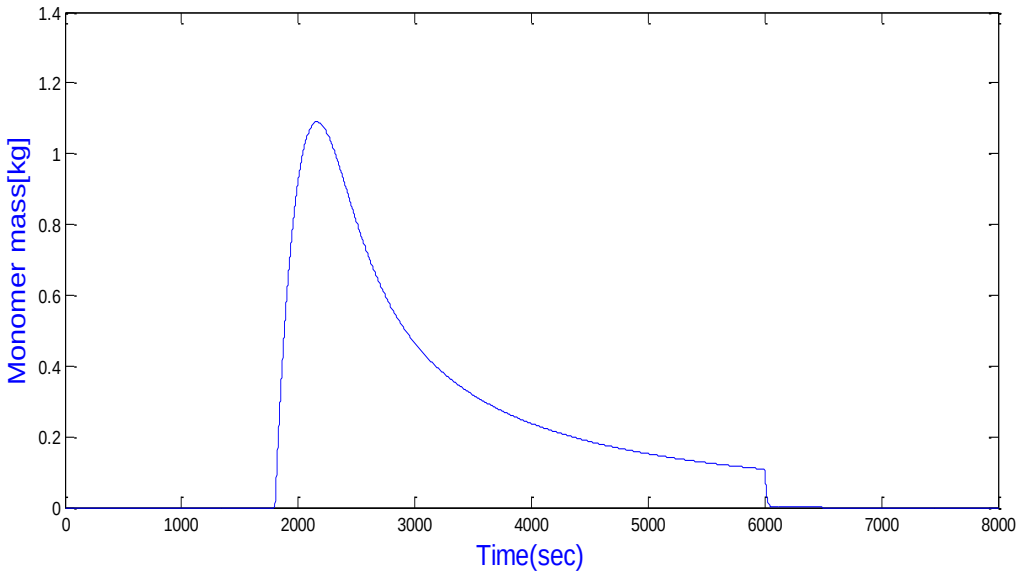


Figure 3.12 The change of Monomer mass

Furthermore, the fouling factor $1/h_f$ increases from batch to batch, causing a decrease in overall heat transfer coefficient, as shown below in Figure 3.13. The blue line shows

that the fouling factor is equal to zero and the red line shows that the fouling factor is 0.704 $m^2K\ kW^{-1}$ (fifth batch).

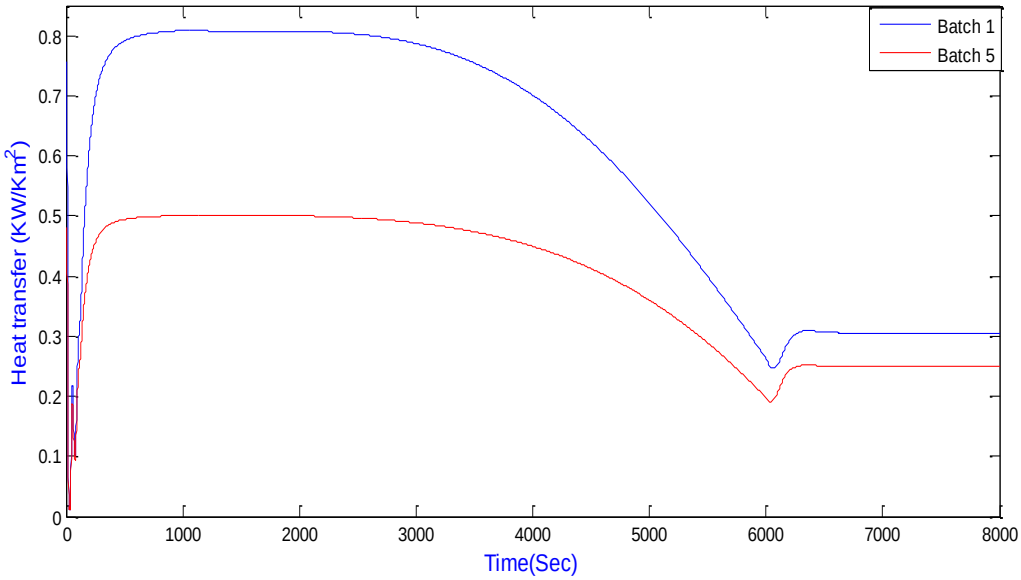


Figure 3.13 The effect of the fouling factor on overall heat.

Reactor temperature affected by fouling factor (batch 1 and batch 5) presented in Figure 3.14 below.

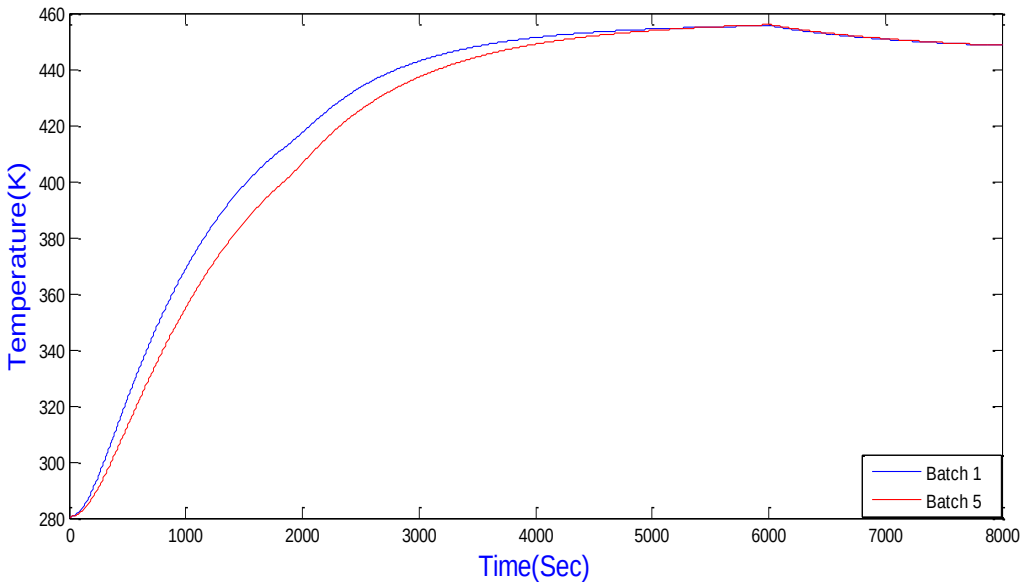


Figure 3.14 The effect of the fouling factor on reactor temperature

The ambient temperature (T_{amb}) is different during summer (305.382K) and winter (280.382K); this affects the initial condition of $T(0)$, T_j^{in} and T_j^{out}, as shown in Figure 3.15.

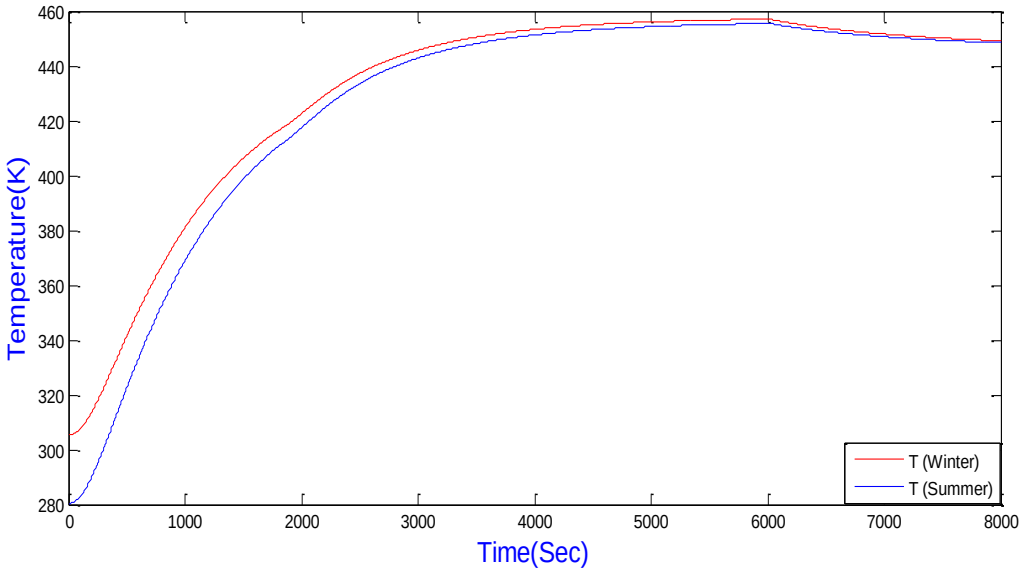


Figure 3.15 Different reactor temperatures during summer and winter.

3.3.2 Open loop response of polymerization process for polymer B

For the recipe of product B, different parameters value, different rates and feed time as seen in Figure 3.16, 3.17.

Feed monomer into the reactor at 0.00684 Kg/Sec for 3600 Sec. After the feed addition period is complete, hold at reaction temperature for 1800 Sec. Another feed of monomer is operated into the reactor at 0.00684 Kg/Sec for 2400 Sec. After the second feed addition period, hold at reaction temperature for 2700 Sec, as shown in figure 3.17.

The uncontrolled signal for the recipe B is shown in Figure 3.16 in different batches, and monomer feed rate is shown in Figure 3.17. These clearly show the effect of monomer on reactor temperature as when the monomer stops the temperature drop down and the other way around.

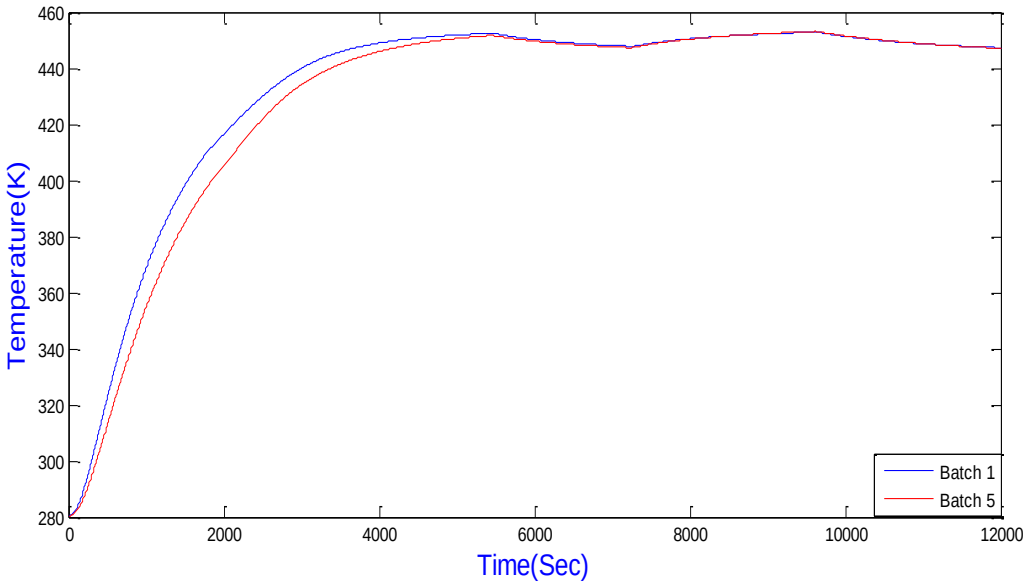


Figure 3.16 Open loop response of reactor temperature for different batches

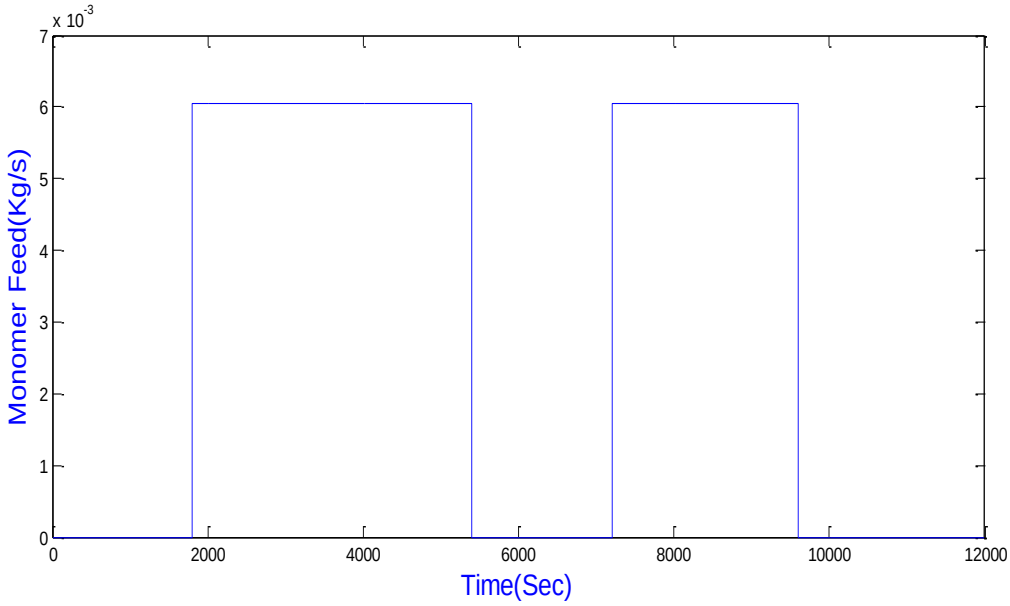


Figure 3.17 Monomer feed rate

Figure 3.18, 3.19 shown polymer mass and monomer mass and it's been discussed in details in previous section 3.3.1.

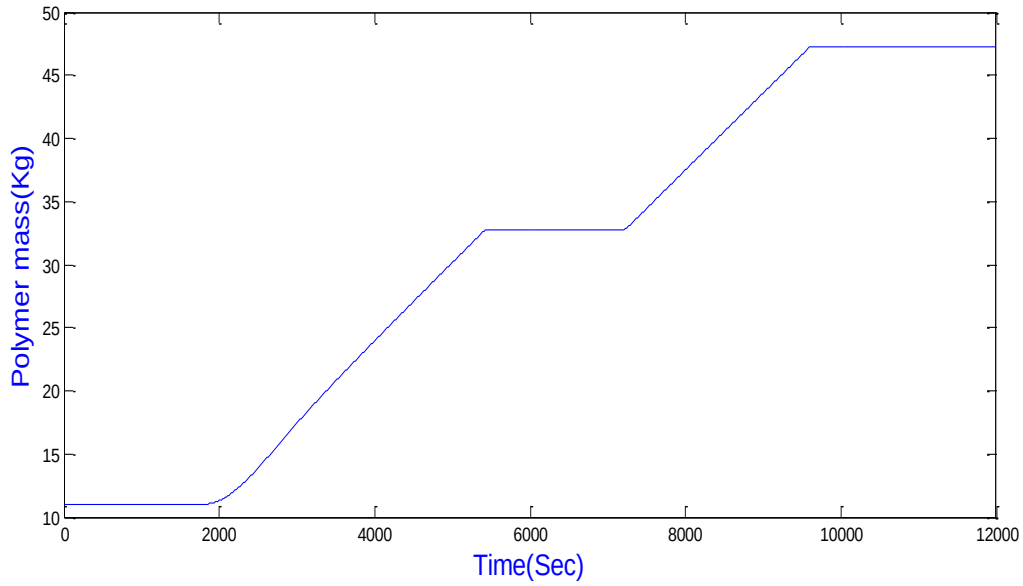


Figure 3.18 The change Polymer mass

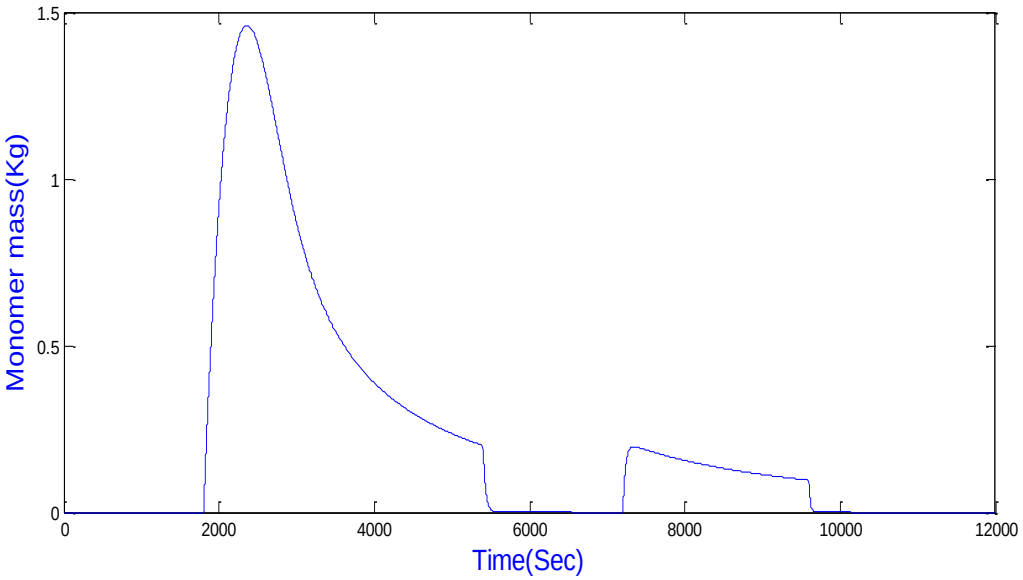


Figure 3.19 The change of Monomer mass

It has been found, its easier to keep the temperature within the tolerance range of the set-point for the product A, but in product B the reactor temperature shows a deviation from the set-point during the second feed of monomer because of exponentially increasing viscosity of the reactor contents and the resulting over exponentially increasing film heat transfer coefficient. Technically, a highly viscous film forms at the reactor wall and prevents so the heat transfer between jacket and reactor contents. For this reason, the heat transfer coefficient is limited to a minimum value of 0.2

$KW/m^{-2}K^{-1}$, as shown in figure 3.20, 3.21. This is necessary because the heat transfer coefficient between the wall and reactor falls to zero during product B, making it impossible to further control the reactor temperature.

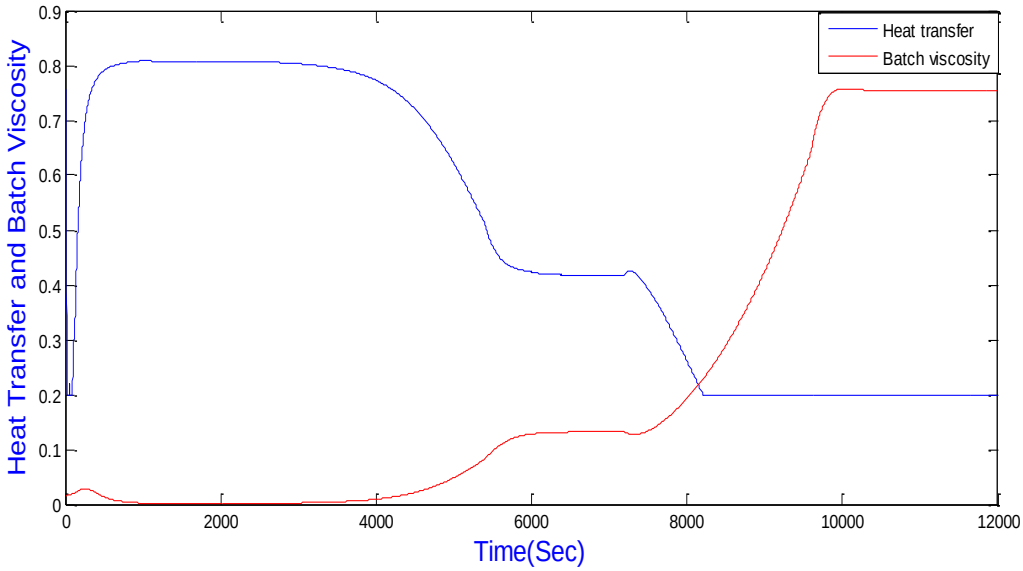


Figure 3.20 Overall heat transfer coefficient VS batch viscosity

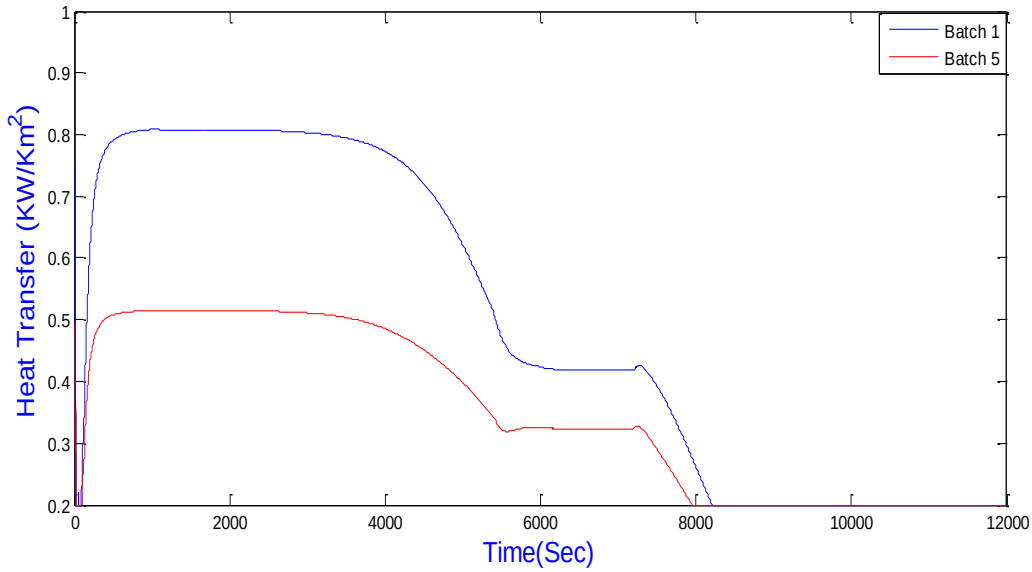


Figure 3.21 The effect of the fouling factor on overall heat.

Figure 3.22 presented the effect of the ambient temperature (T_{amb}) during summer and winter. Its small difference but in fact has a big impact on the quality of product.

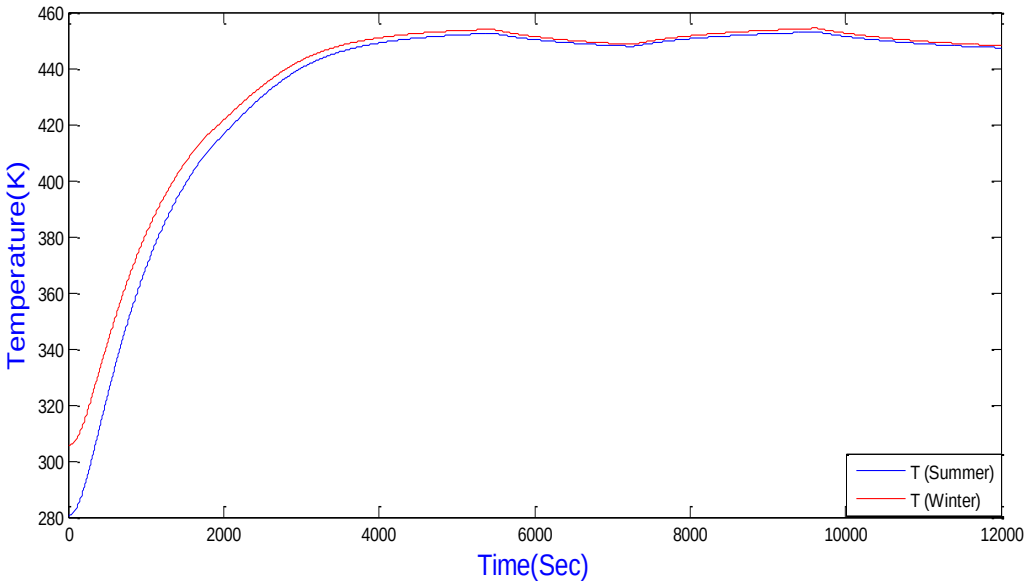


Figure 3.22 Different reactor temperatures during summer and winter.

3.4 Summary

The main objective of this chapter is to understand or improve chemical process operation which is a major objective for developing dynamic process model for the reactor.

This part of work is focused primarily on the development of ordinary differential equation models that describe the dynamic behaviour of polymerization process by using Matlab/Simulink.

The detailed description of the Chylla and Haase polymerization process, the recipe of polymer A and polymer B, the mass balance and energy balance equations are represented. Also the disturbance and uncertainties in the process is discussed in details. The use of Matlab and Simulink for modelling and analysis is demonstrated.

Model is developed here to illustrate the behaviour of polymerization process and Matlab used to investigate input/output control pairing in order to identify fundamental nature of the solution polymerization control problem and to determine the best control system structure. In the following chapter, the control design of conventional control and linear model predictive control LMPC of polymerization reactor will be discussed.

TABLE 3.3
Parameter values of Chylla-Haase reactor

Symbol	Unit	Value of polymer A	Value of polymer B
$m_M, 0$	Kg	0	0
$m_p, 0$	Kg	11.227	11.010
m_w	Kg	42.750	42.010
ρ_m	Kg m^{-3}	900.0	900.0
ρ_p	Kg m^{-3}	1040.0	1040.0
ρ_w	Kg m^{-3}	1000.0	1000.0
$c_{p,}M$	KJ Kg^{-1} K^{-1}	1.675	1.675
$c_{p,}P$	KJ Kg^{-1} K^{-1}	3.140	3.140
$c_{p,}W$	KJ Kg^{-1} K^{-1}	4.187	4.187
m_c	Kg	21.455	21.455
$\dot{m}_c$	Kg s^{-1}	0.9412	0.9412
$c_{p,}c$	KJ Kg^{-1} K^{-1}	4.187	4.187
k_0	s^{-1}	55	20
k_1	m s Kg^{-1}	1000	1000
k_2	-	0.4	0.4
E	KJ Kmol^{-1}	29560.89	29560.89
c_0	Kg $m^{-1}s^{-1}$	5.2× 10^{-5}	3.2×
10^{-5}			
c_1	-	16.4	19.1
c_2	-	2.3	2.3
c_3	-	1.563	1.563
a_0	K	555.556	555.556
ΔH_p			
d_0	KW $m^{-2}K^{-1}$	0.814	0.814
d_1	m s Kg^{-1}	-5.13	-5.13
$m_M^{in,max}$	Kg s^{-1}	7.560× 10^{-3}	6.048× 10^{-3}
$[t_M^{in}, 0, t_M^{in}, 1]$	min	[30,100]	[30,90]
$[t_M^{in}, 2, t_M^{in}, 3]$	min	-	[120,160]
T_{set}	K	355.382	353.160
P	m	1.594	
B_1	m^2	0.193	
B_2	m^2	0.167	

R	KJ Kmol^{-1} K^{-1}	8.314
UA_{Loss}	KW K^{-1}	0.00567567
τ_p	s	40.2
θ_1	s	22.8
θ_2	s	15
i	-	[0.8:1.2]
${h_f}^{-1}$	m^2 k kW^{-1}	[0.00,0.176,0.352,0.528,0.704]
T_{Amb}	K	280.38 (winter), 305.38 (summer)
T_{inlet}	K	278.71 (winter), 294.26 (summer)
T_{steam}	K	449.82

Chapter 4 Development of Primary Control Methods

Its been mentioned before a very precise temperature control is required in order to guarantee that the final product has the desired quality. In the original paper in which this benchmark problem was presented (Chylla and Haase, 1993) it was stated that the temperature must stay within a very tight tolerance range of ±0.6 K around the specified reaction temperature, 355.382 K for product A and 353.382 K for product B. From a practical point of view, such a tight tolerance may be questioned, though, e.g. because of the limited accuracy of the temperature sensors. The main challenge of this benchmark problem is to guarantee precise and robust temperature control under a variety of uncertainties and disturbances mentioned in chapter 3.

4.1 PID Control

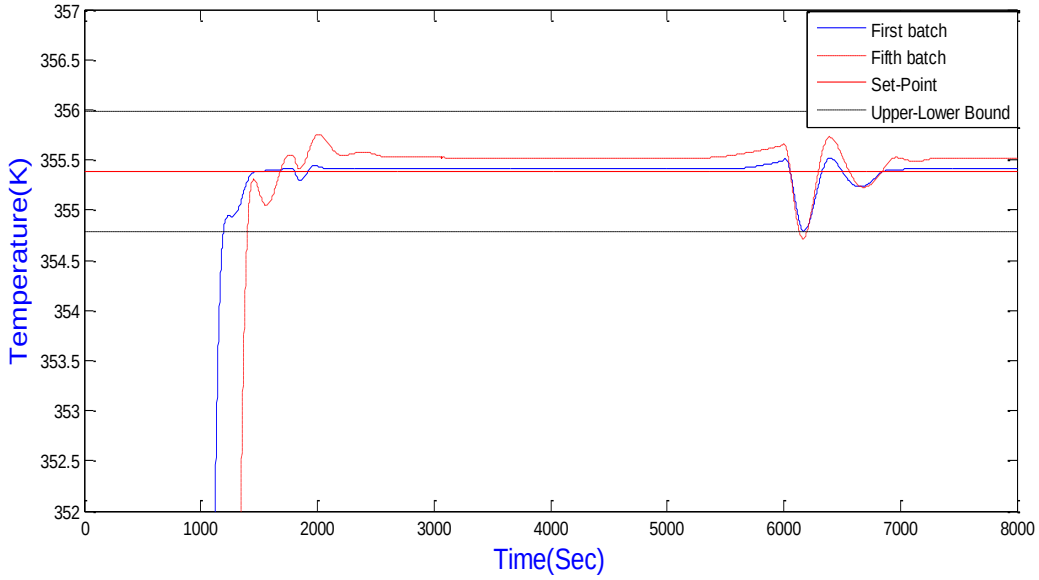
The PID controller equation given in (Ogata, 1997) has the following form:

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \tag{4.1}$$

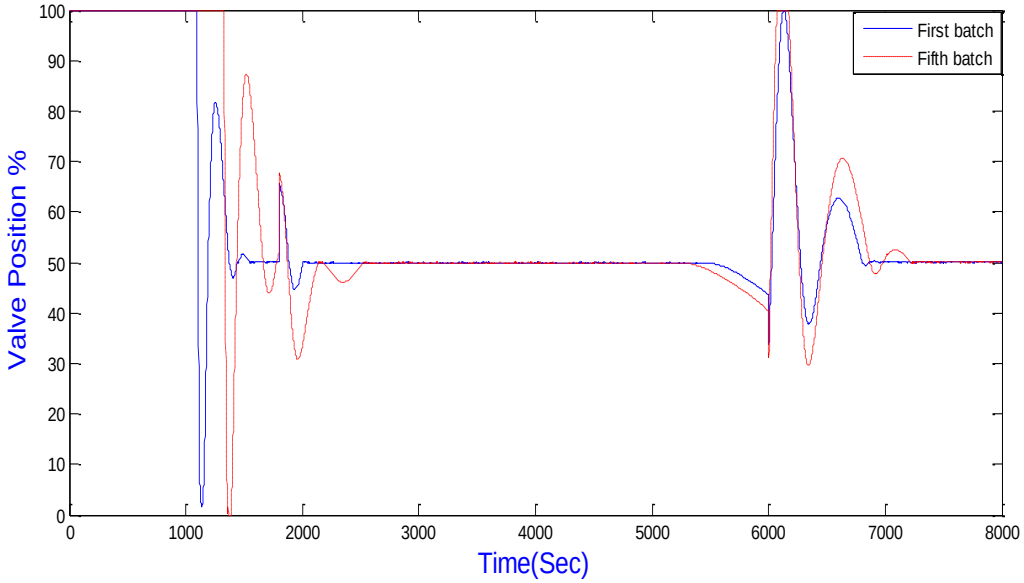
Where, K_p , K_i and K_d are proportional, integral and differential gains respectively. The common methods to tune the PID parameters are the Zigler and Nichols open–loop and closed-loop methods (Ogata, 1997). Two methods were used to tune the PID parameters, but we haven't got s-shape response as the same as closed loop we haven't got sustain oscillation response, also, the trial and error process unnecessary as recent MATLAB provides tools for automatically choosing optimal PID gains. The PID tuner provides a fast and widely applicable single-loop PID tuning method for the Simulink PID controller blocks. This method can tune PID parameters to achieve a robust design within the desired response time. PID self-tuning automatically computes a linear plant model from the Simulink model and designs an initial controller, computing PID parameters that robustly stabilize the system. After fine tuning, the PID controller that is used here is given in equation (4.2).

$$G_c(s) = 65(1 + 1/0.0008s + 2500s) \tag{4.2}$$

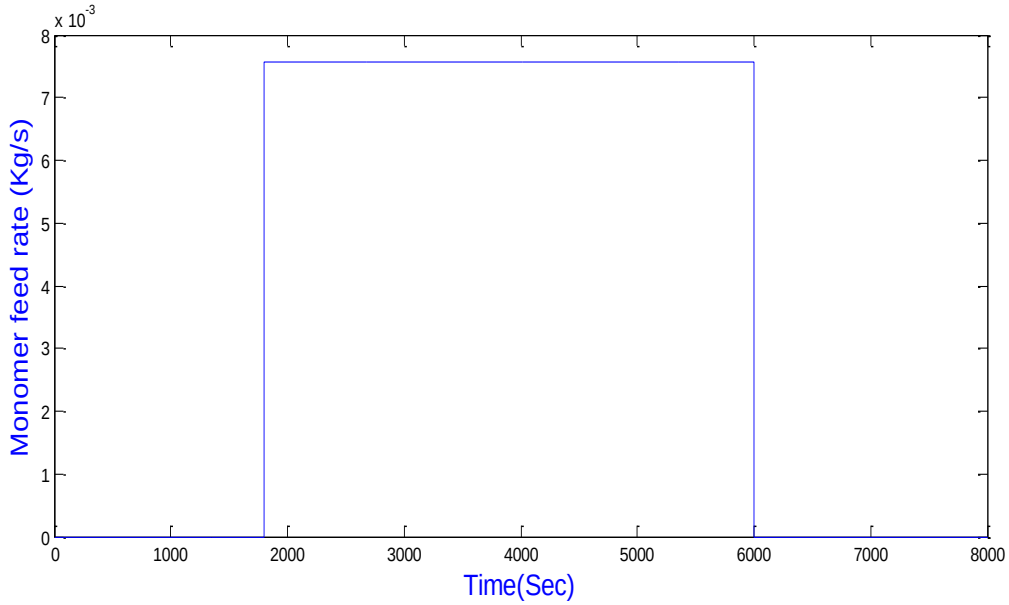
4.1.1 Response of the reactor with PID control for polymer A.



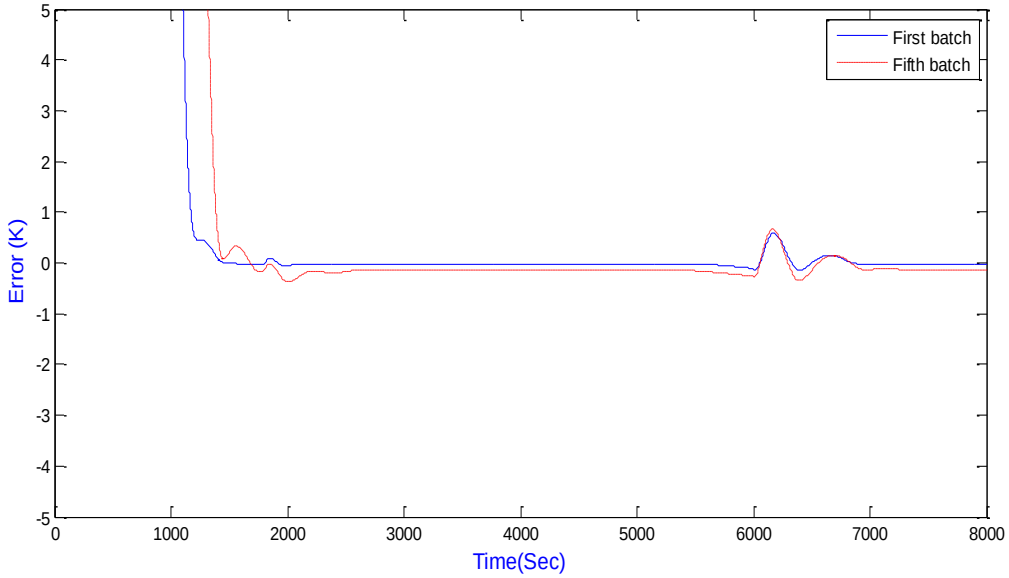
(a) Temperature for different batches (Winter)



(b) Valve position for different batches



(c) Monomer feed rate



(d) Error for different batches

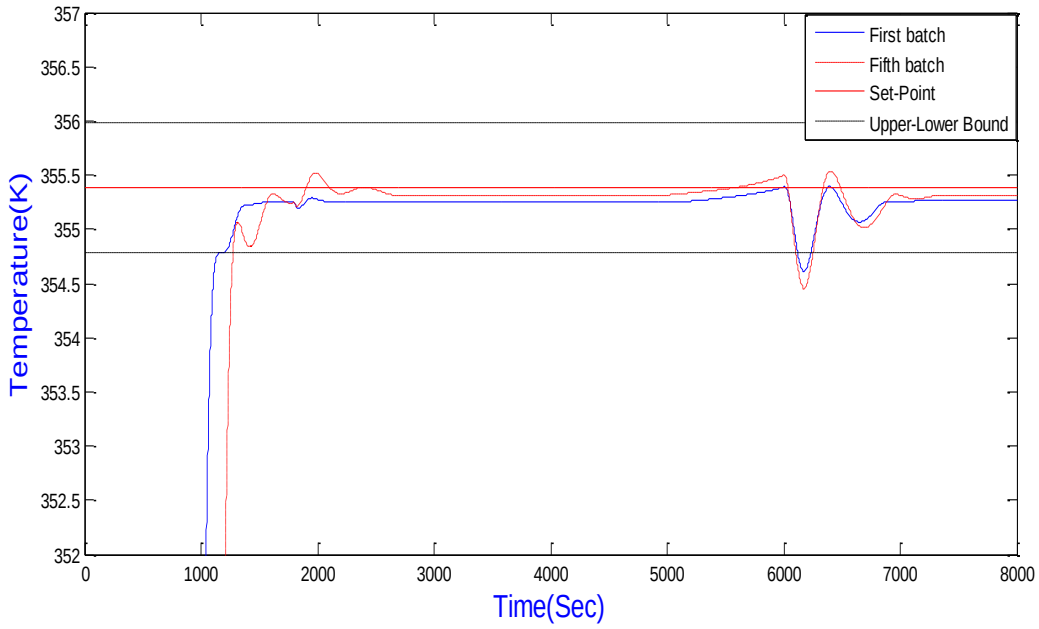
Figure 4.1 Simulation result for polymer A with PID control during winter

Figures 4.1 (a) – (d) show the simulation results for the simulated polymerization reactor with Automatic PID Tuning control for batch 1 and 5 for winter season. Figure 4.1 (a) shows the reactor temperature of both first and fifth batch during winter season settling at the desired set point of 355.382 K starting from the ambient temperature of 280.382 K (winter). It can be seen in the first batch the temperature still within the tolerance range but in the fifth batch at 6000s when the monomer feeding stops, the

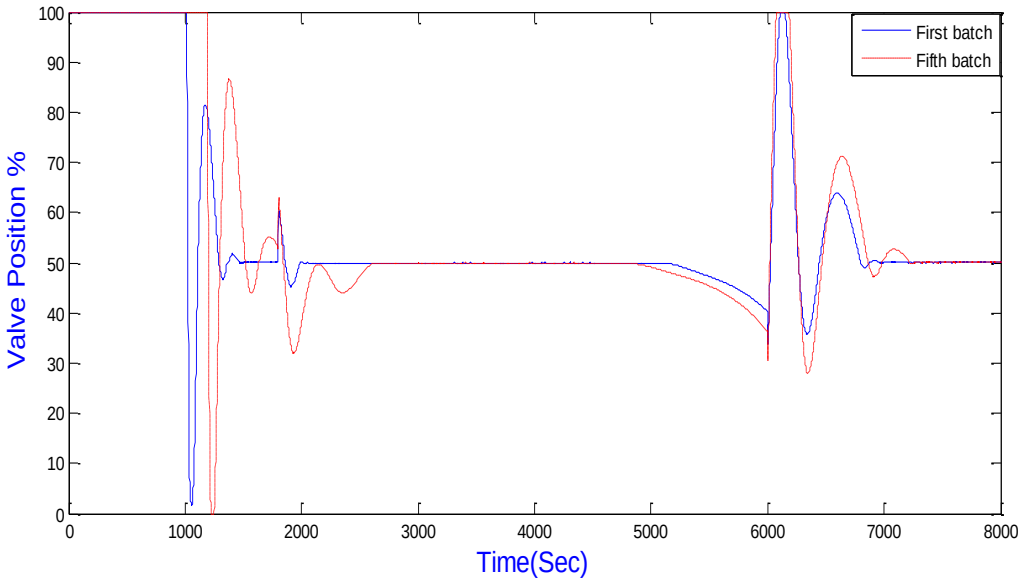
temperature goes slightly below the tolerance range which is not acceptable for such a high quality product.

Figure 4.1 (b) shows the valve position of both first and fifth batch. The response shows that the valve position is not smooth especially during the initial time. Monomer is added at 1800s and when the monomer is stopped at 6000s as shown in Figure 4.1 (c). Figure 4.1 (d) shows the error, which is the difference between the desired set temperature and the response from the simulated polymerization process.

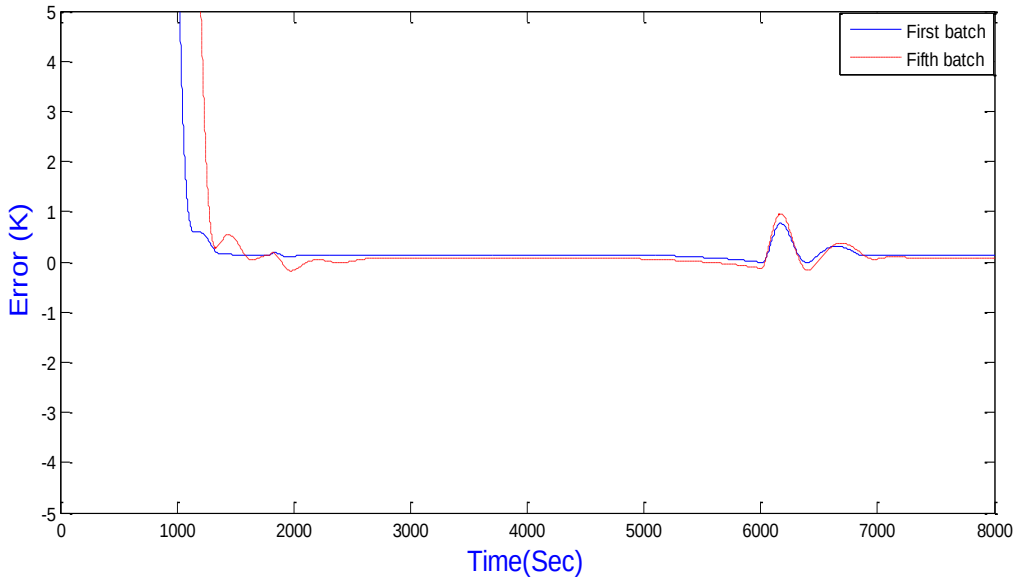
For summer season, Figure 4.2 (a)-(c) demonstrates the simulation result. It can be seen that the PID control still cannot guarantee keeping the reactor temperature within the tolerance range.



(a) Temperature for different batches (Summer)



(b) Valve position for different batches



(c) Error for different batches

Figure 4.2 Simulation result for polymer A with PID control during summer

4.1.2 Response of the reactor with PID control for polymer B.

The PID tuning parameter used for product B is demonstrated in equation (4.3).

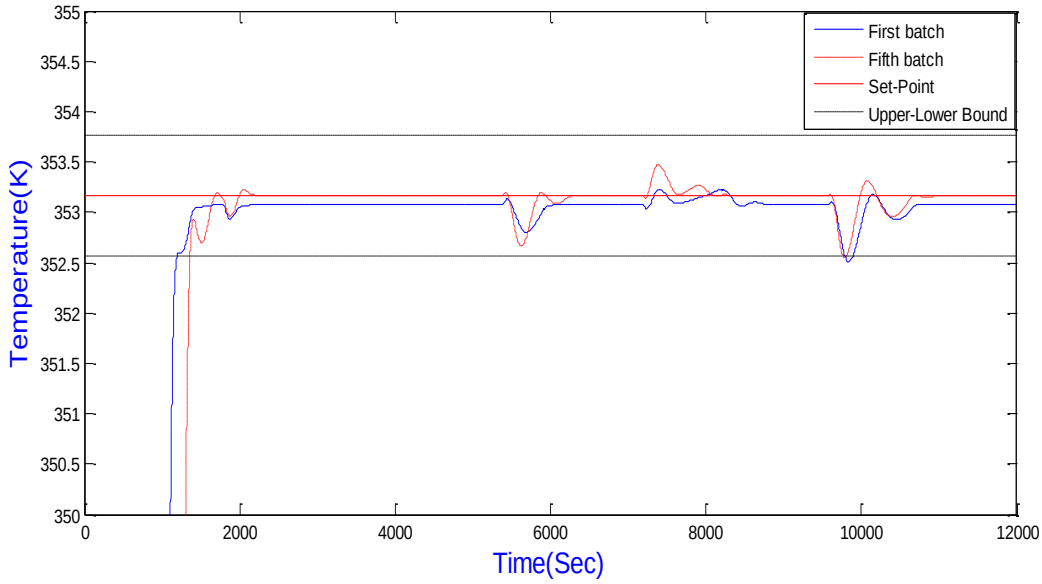
$$G_c(s) = 60(1 + 1/0.0008s + 3500s) \quad (4.3)$$

The response of reactor temperature, valve position and feedback error for polymer B based on PID control for first and fifth batch considering ambient

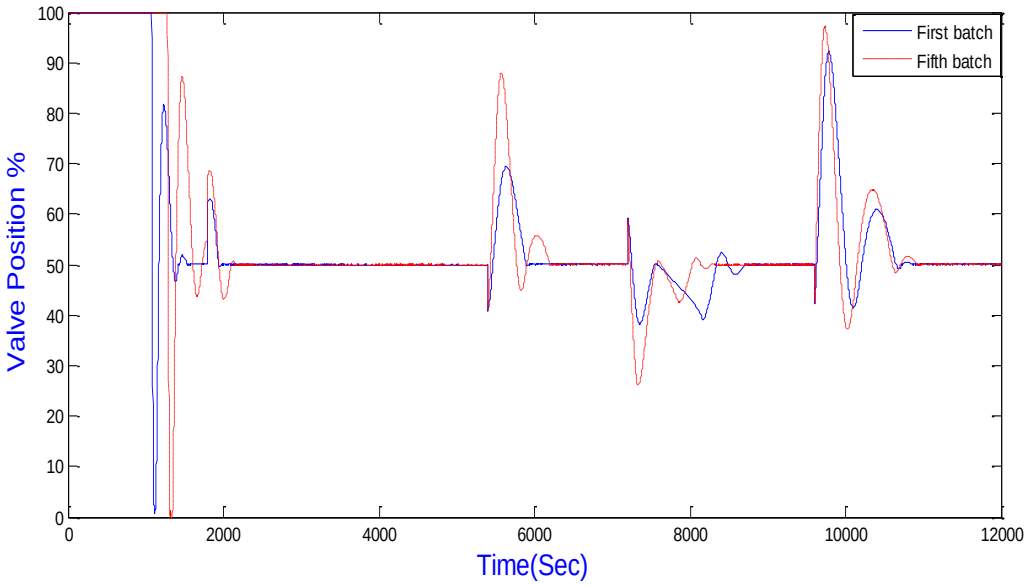
temperature for winter and summer demonstrated in Figure 4.3 (a)-(d) and 4.4 (a)-(c).

For product B the heat transfer coefficient is limited to a minimum value of 0.2 $KW/m^{-2}K^{-1}$, because it is impossible to control the reactor temperature when the heat transfer between wall and reactor falls to zero.

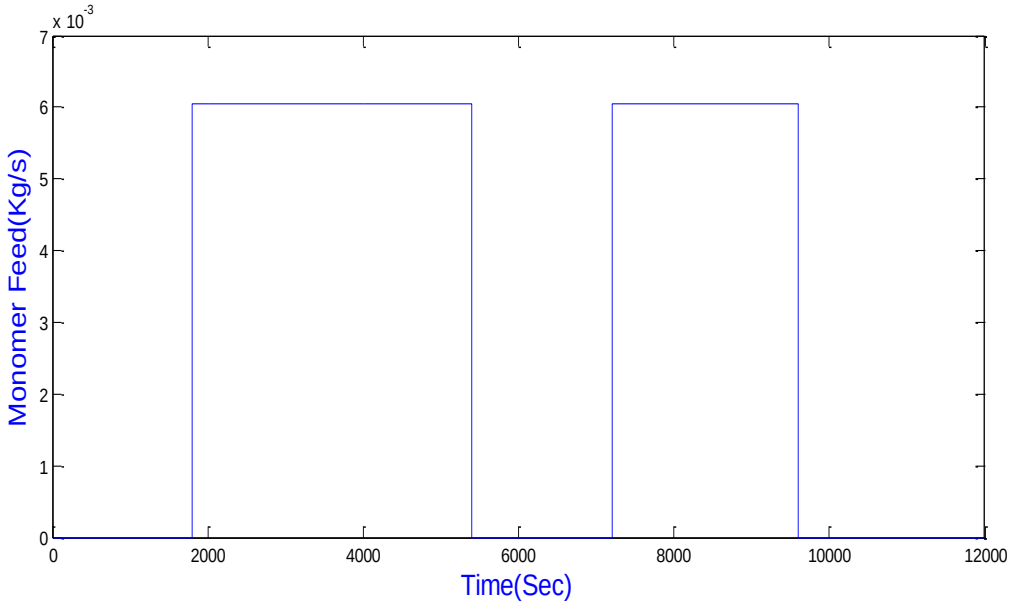
Figure 4.3 (a) shows the temperature during the fifth batch reach the lower bound, Figure 4.3 (b) presents the valve position, monomer add in and stop as seen in Figure 4.3 (c) and finally Figure 4.3(d) present the error between the actual and desired temperature.



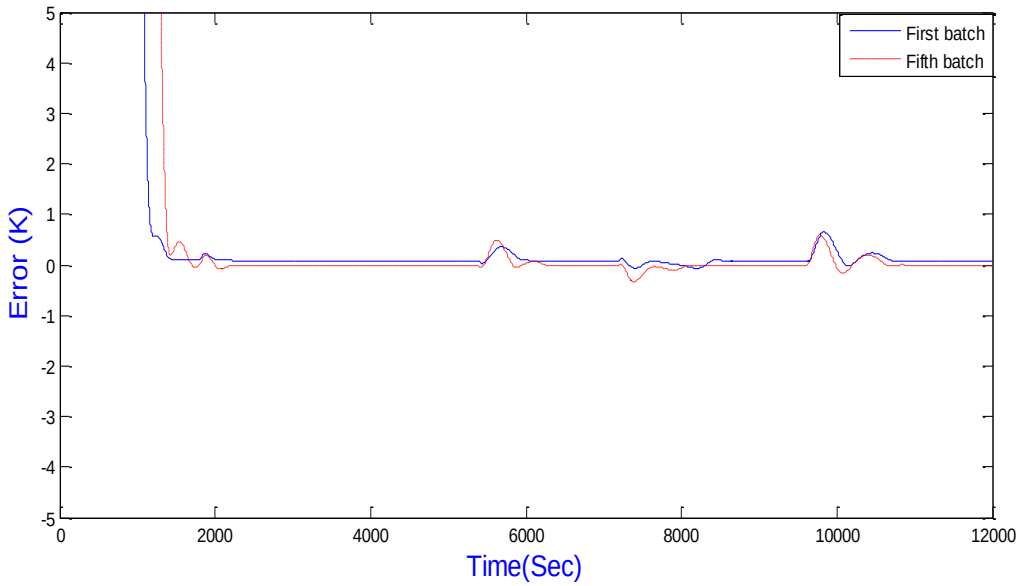
(a) Temperature for different batches, product B (Winter)



(b) Valve position for different batches, product B



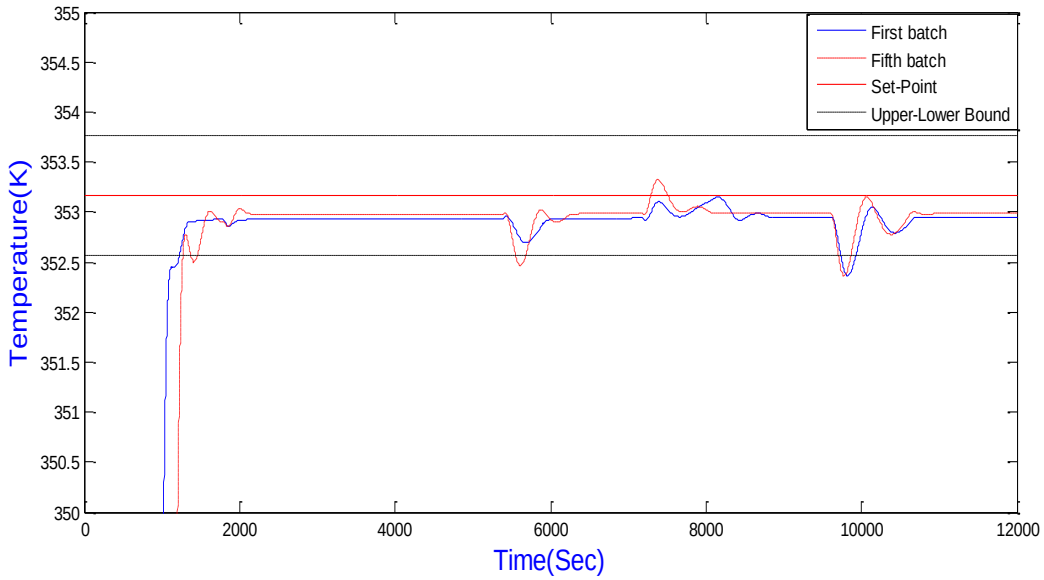
(c) Monomer feed rate, product B



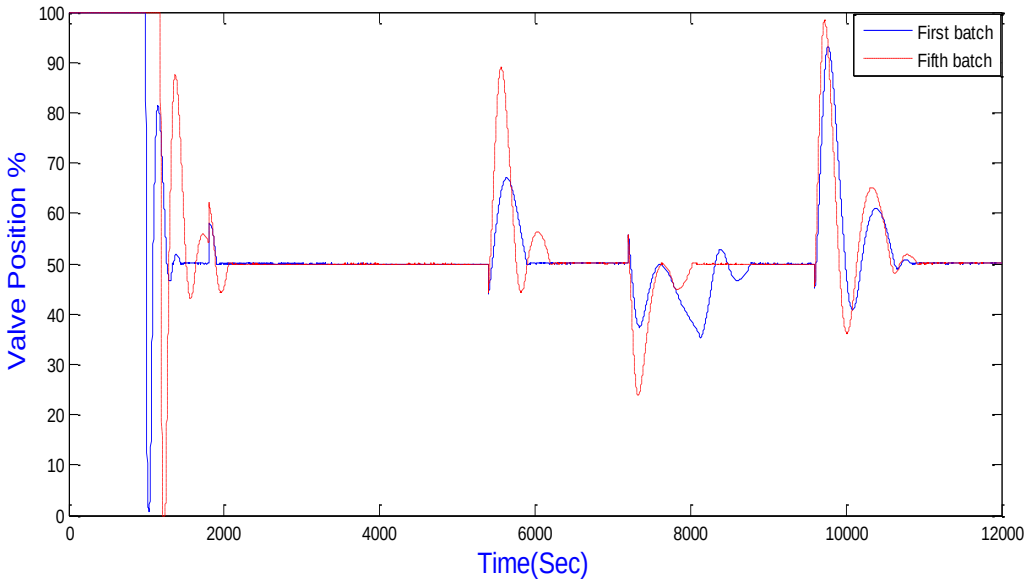
(d) Error for different batches, product B

Figure 4.3 Simulation result for polymer B with PID control during winter

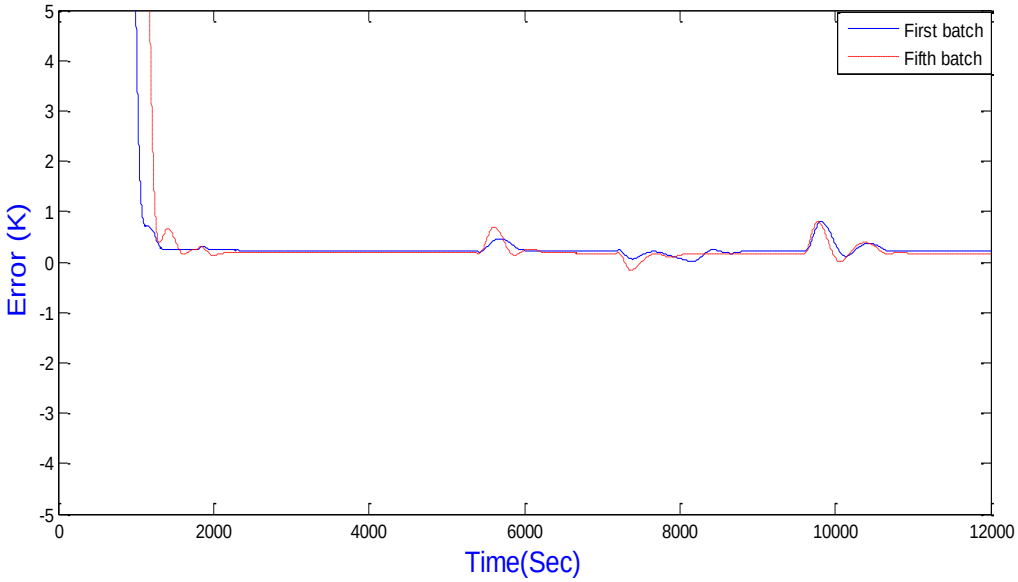
For summer season, Figure 4.4 (a)-(c) demonstrates the simulation result, the reactor temperature for first batch and fifth batch goes beyond from tolerance interval of (±0.6k)



(a) Temperature for different batches, product B (Summer)



(b) Valve position for different batches, product B



(c) Error for different batches, product B

Figure 4.4 Simulation result for polymer B with PID control during summer

From simulation, PID controller able to control such process in the first batch (no disturbance effect), but the effect of fouling factor, impurity factor and ambient temperature drive the temperature out of the tolerance which directly effect on the quality of products as it can be seen in Figure 4.4. Therefore it will be necessary to retune the controller and adaptive control should be considered. Also when the temperature goes beyond the target (tolerance) the valve position goes above 100% in

order to heat the reactor which is in practice not reasonable and this is one of the main points to use advanced control.

4.2 Cascade Control

Cascade control is still by far one of the most popular solutions that are applied in the process industry nowadays, implemented to improve the disturbance rejection properties of the controlled system. The introduction and use of an additional sensor that allows for a separation of the fast and slow dynamics of the process, results in a nested loop configuration, as shown in Figure 4.5. The controller of the inner loop is called the secondary controller whereas the controllers of the outer loop as the primary controller, being the output of the primary loop the variable of interest. The basis behind this configuration is that the fast dynamics of the inner loop will provide faster disturbance reduction and minimize the possible effect disturbance before they affect the primary output. This set up involves two controllers. It is therefore needed to tune both PIDs.

Cascade control is desired when single loop control does not provide satisfactory control performance and when a measured secondary variable is available. A potential secondary variable must satisfy three criteria. First, it must indicate the occurrence of an important disturbance; that is the secondary variable must respond in a predictable manner every time the disturbance occurs. Naturally, the disturbance must be important or there would be no reason to reduce its effect. Second, the secondary variable must be influenced by the manipulated variable this causal relationship is required so that a secondary feedback control loop functions properly. Finally, the dynamics between the final element and the secondary must be much faster than the dynamics between the secondary variable and the primary controlled variable so it is important to select the outer loop that are faster than the slave temperature controller on the jacket. Similarly, the jacket control loop should contain less dead time than its

master, which would usually difficult, one possible method of reducing the dead time of the jacket outlet to the jacket inlet. This usually is not recommended, because when this is done, the slave will do much less work because the nonlinear dynamics of the cascade slave loop is to move the measurement from the jacket have been transferred into the master loop. There are several advantages of cascade control, effectively accounts for external disturbances, reduces dead time in variable response, and compatible with other Control Systems, such as Feed-Back and Feed-Forward Control. Commonly used for chemical reactors is a PI cascade control structure, which provides a robust operation but often lacks in control performance. The cascaded control structure (CCs), which is illustrated by the block diagram in Figure 4.5, is configured such that the master controller in the outer loop is the primary controller that regulates the primary controlled variable (reactor temperature T) by setting the set-point of the inner loop. The slave controller in the inner loop is the secondary controller adjusts the valve position C in order to control the mean jacket temperature T_j by the master controller.

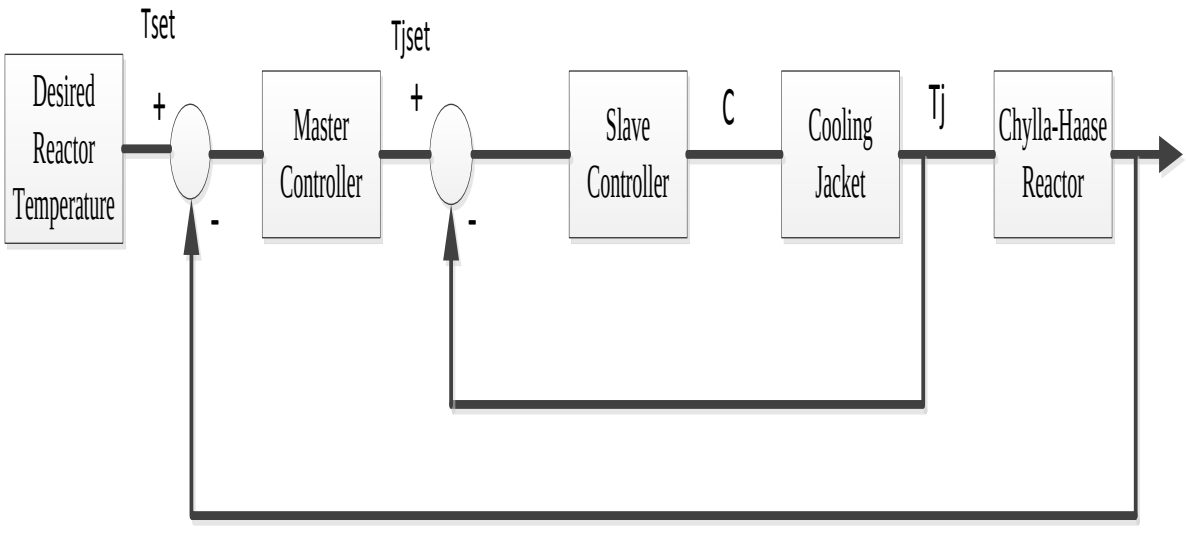


Figure 4.5 Cascade control scheme for the polymerization reactor

The process was simulated with the conventional PID cascade control structure and the performance of the control scheme for the reactor with Polymer A was obtained.

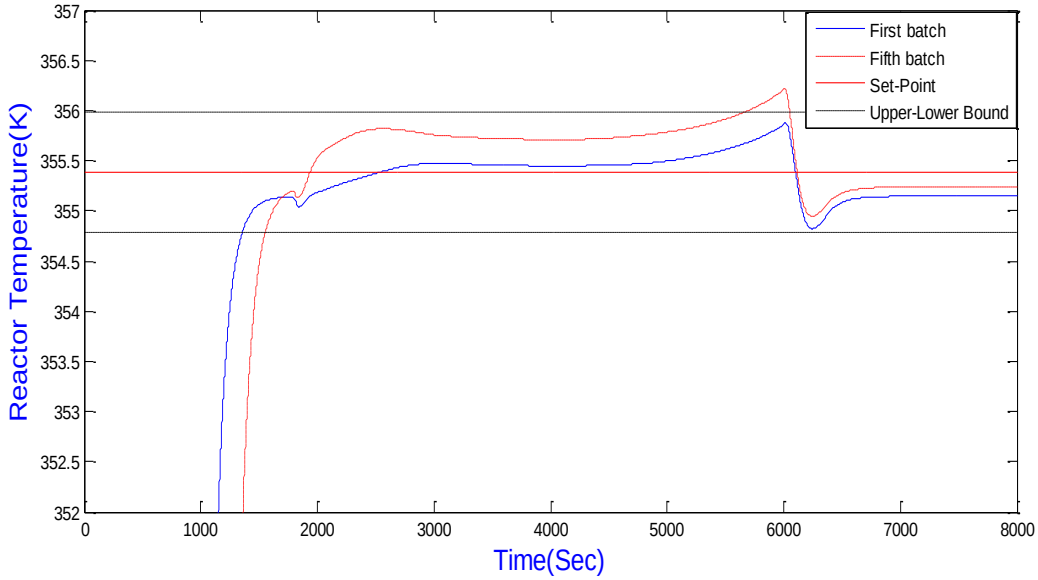
The sampling time for cascade controller is set to 1s. The deviations of the reactor temperature between T and T_{set} (desired reactor temperature) in this conventional cascade control scheme was due to the time delayed response of the slave controller to the respective set point adjusted by the master controller.

4.2.1 Reactor performance with cascade control for polymer A

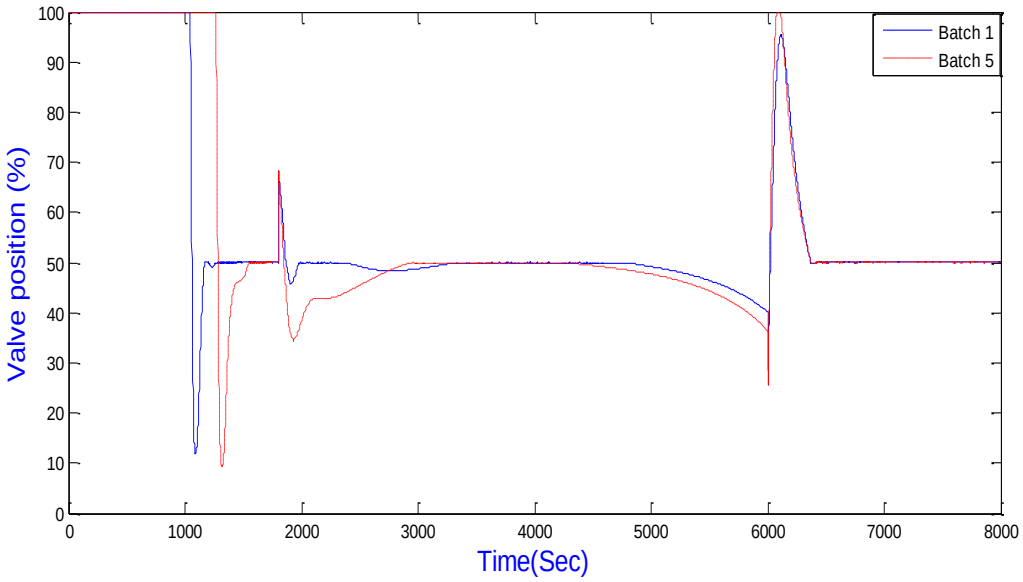
The Figures 4.6 (a) – (c) show the simulation results for the simulated polymerization reactor with conventional cascade control for batch 1 and 5 for winter season. Figure 4.6 (a) shows the reactor temperature of both first and fifth batch during winter season cannot keep settling at the desired set point of 355.382 K in the fifth batch due to the effect of fouling and impurity factors.

Figure 4.6 (b) shows the valve position of both first and fifth batch. The response shows that the valve position is not smooth especially during the initial time, however its smoother than the PID controller.

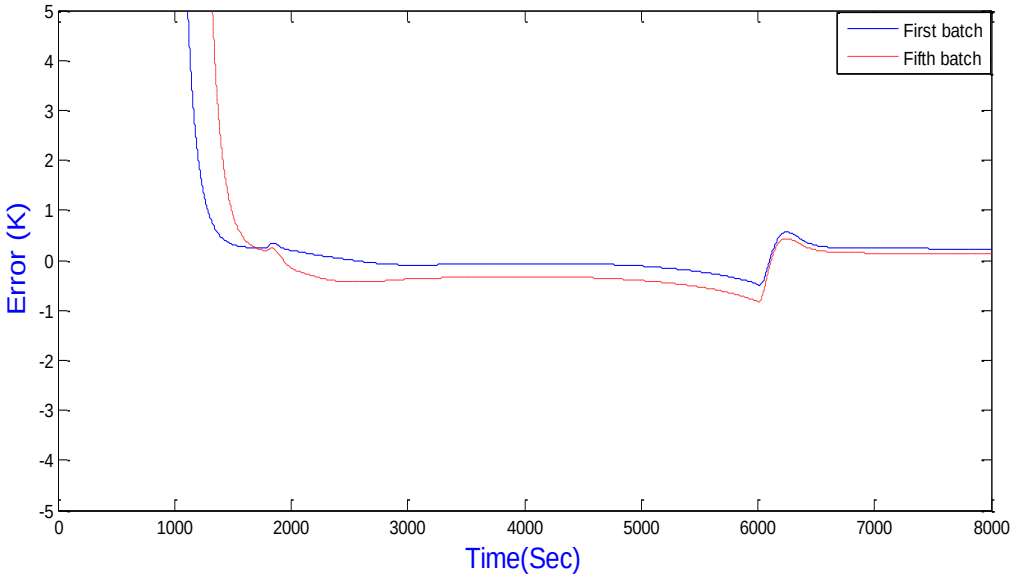
Figure 4.6 (c) shows the error, which is the difference between the desired set temperature and the response from the simulated polymerization process.



(a) Temperature for different batches, product A (Winter)



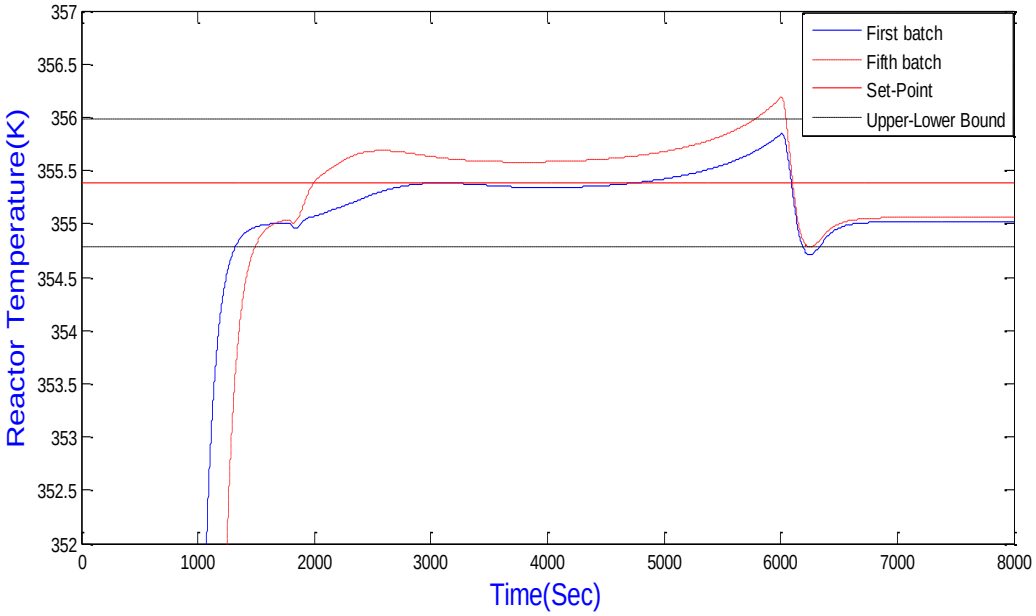
(b) Valve position for different batches, product A (Winter)



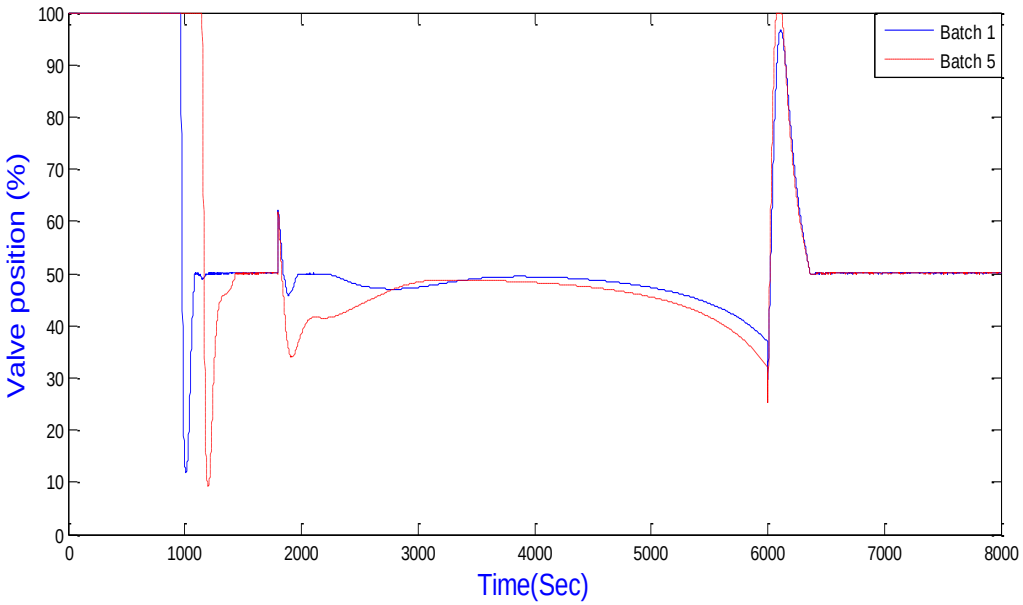
(c) Error for different batches, product A

Figure 4.6 Simulation result for polymer A with cascade controller during winter

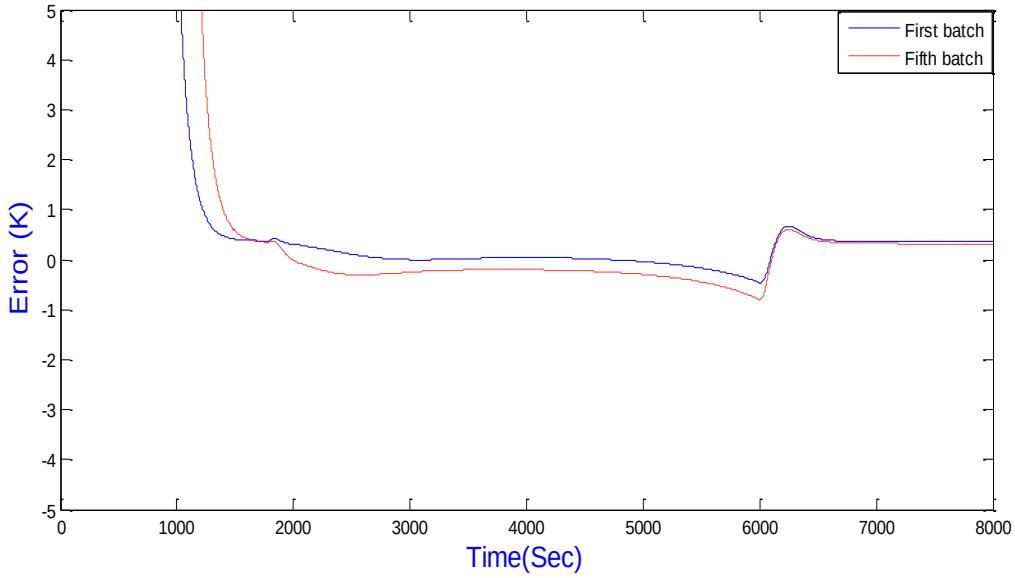
For the summer season T_{amb} and initial conditions of reactor changing Figure 4.7 (a)-(c) demonstrate the impact on the reactor temperature during summer. Its clear that the cascade controller still has significant problems with temperature regulation at the end of the feed period (6000 sec).



(a) Temperature for different batches, product A (Summer)



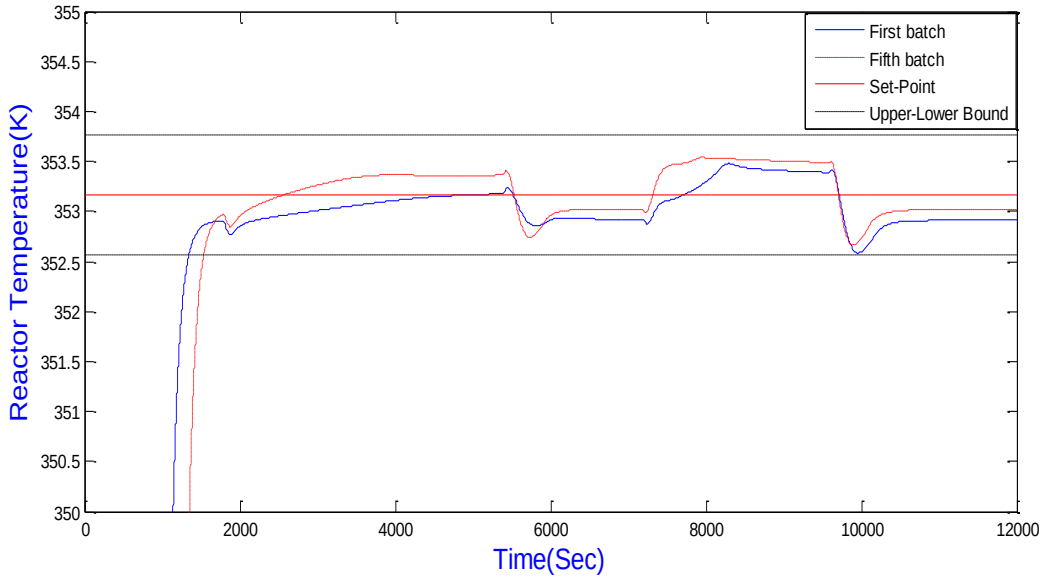
(b) Valve position for different batches, product A (summer)



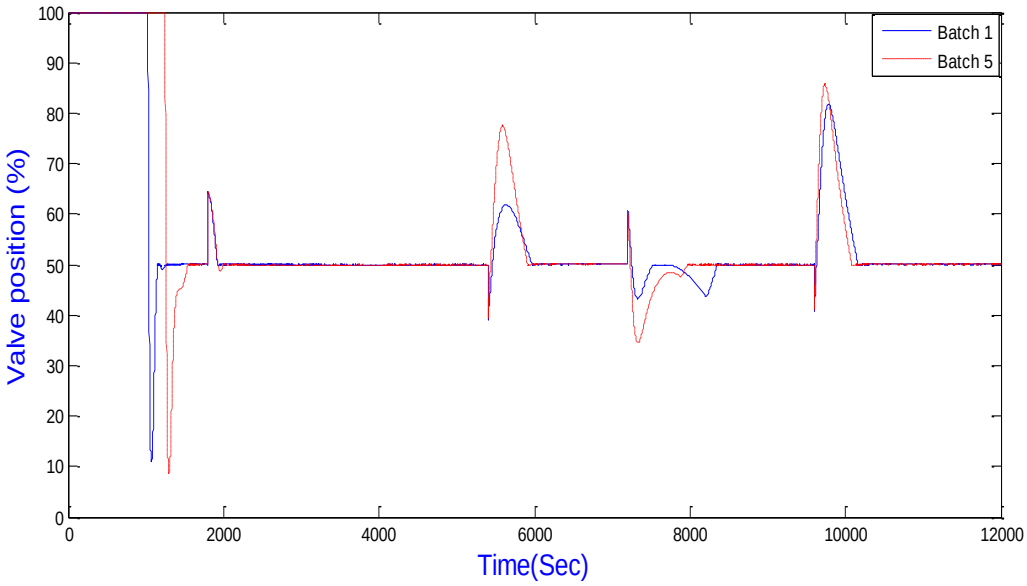
(c) Error for different batches, product A (summer)

Figure 4.7 Simulation result for polymer A with cascade control during summer

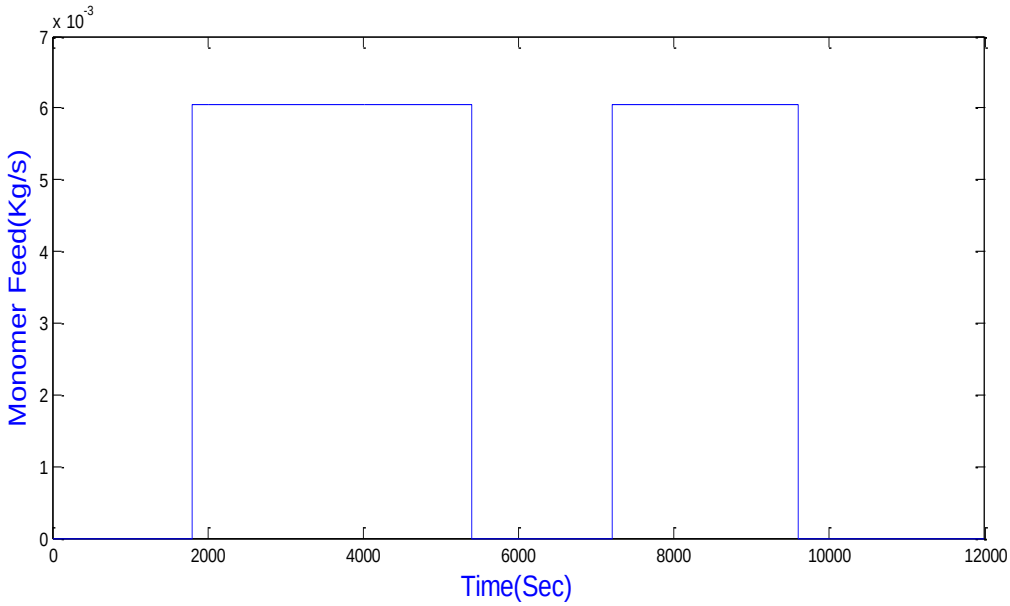
4.2.2 Reactor performance with cascade control for polymer B



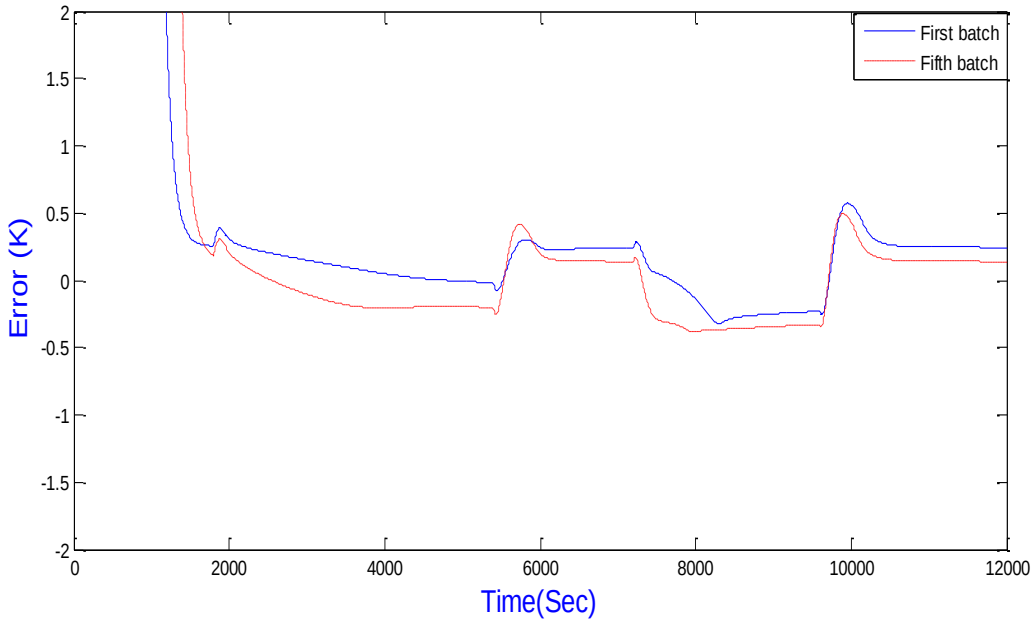
(a) Temperature for different batches, product B (winter)



(b) Valve position for different batches, product B (winter)



(c) Monomer feed rate, product B (winter)

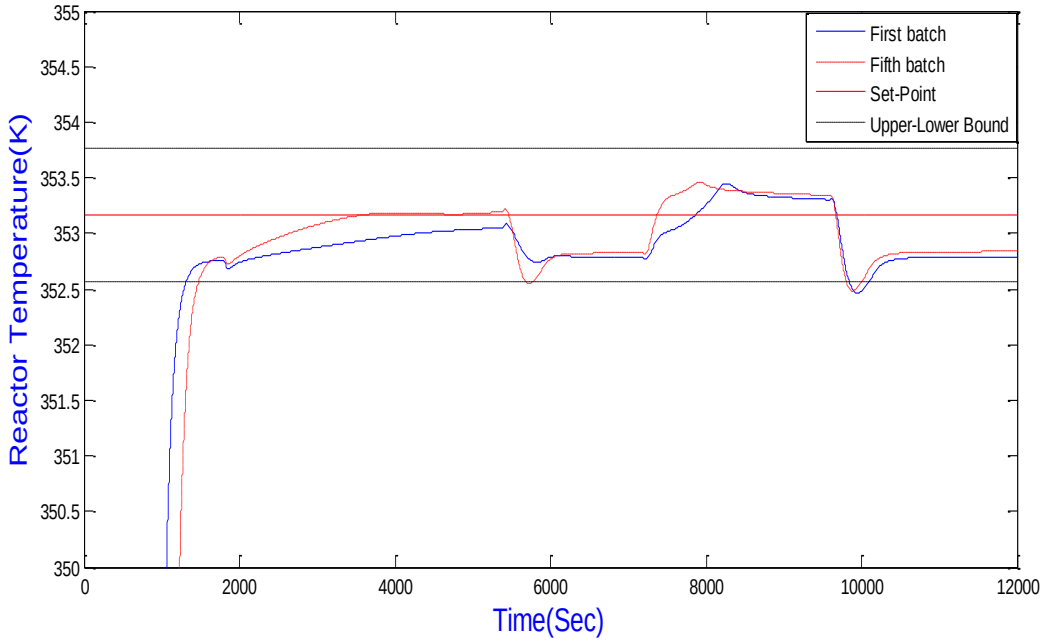


(d) Error for different batches, product B (winter)

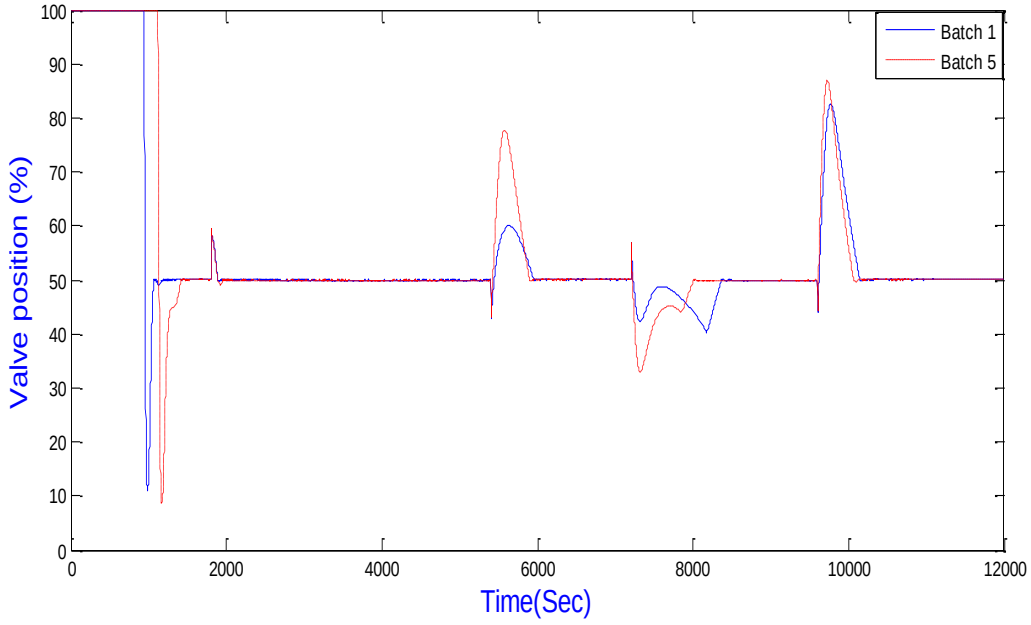
Figure 4.8 Simulation result for polymer B with cascade control during winter

The Figures 4.8 (a)-(d) show the simulation results for the simulated polymerization reactor for polymer B with conventional cascade control for batch 1 and 5, winter season. Figure 4.8 (a) shows the reactor temperature of both first and fifth batch during winter season settling at the desired set point of 353.160 *K* starting from the ambient temperature of 280.382 *K*.

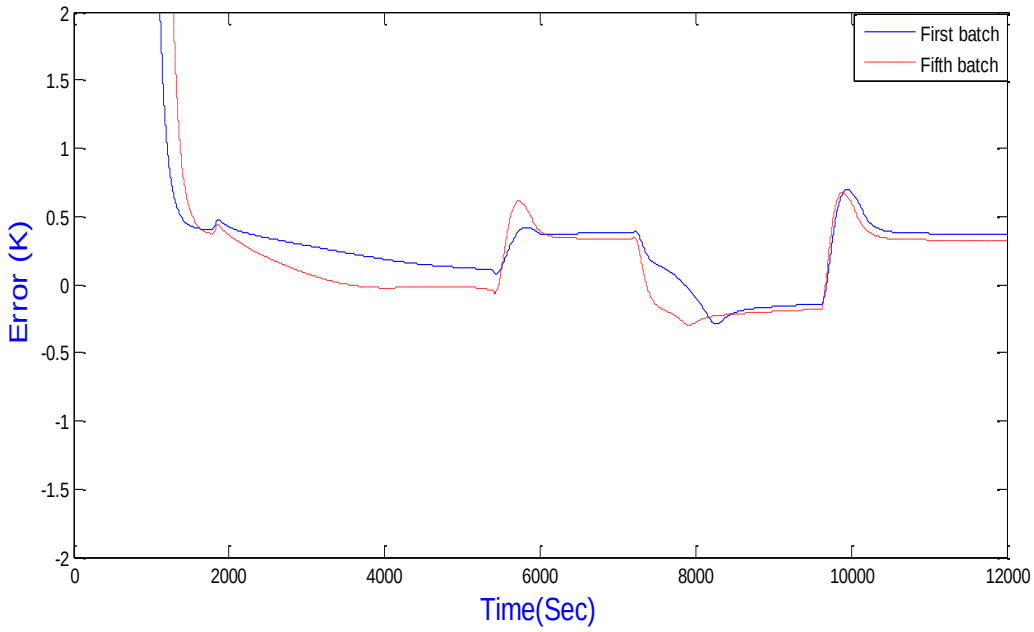
Figure 4.8 (b) shows the valve position of both first and fifth batch. The response shows that the valve position is not smooth especially during the initial time, the time when monomer is added at 1800 s, when the monomer is stopped at 5400s, during the second feed period when the monomer is added at 7200 s and when the monomer is stopped at 9600 s. Figure 4.8 (d) shows the error, which is the difference between the desired set temperature and the response from the simulated polymerization process.



(a) Temperature for different batches, product B (summer)



(b) Valve position for different batches, product B (summer)



(c) Error for different batches, product B (summer)

Figure 4.9 Simulation result for polymer B with cascade control during summer

The Figures 4.9(a)-(c) presented the outcome of the simulation for polymer B summer condition. As it can be seen from the plots in Figure 4.9, especially during monomer stops period, the temperature control is crucial, the proposed scheme follows the reference trajectory very closely until the point at which the monomer feed stop as shown in Figure 4.9(a). Figure 4.9(b) presents the valve position, first and fifth batch during summer and finally Figure 4.9(c) presents the error.

4.3 LQR Control

Linear quadratic regulation LQR is an optimal control method. The theory of optimal control is concerned with operating a dynamic system at minimum cost. The case where the system dynamics are described by a set of linear differential equations and the cost is described by a quadratic function is called the LQ problem (Bequette, 2003).

LQR control is calculated based on a linear model of the plant under control. If the linear model represents plant exactly, then the controller is optimal. However, if there is a mismatch due to model inaccuracy (i.e. nonlinearity) then the resulting controller will degrade and the system may even become unstable.

The mathematical model that describe the behaviour of chylla-haase are generally nonlinear, while commonly used control strategies are based on linear system theory, so its important to be able to linearize nonlinear model. The method that used here from linear model is based on Matlab/control tool box uses LTI (linear, time invariant) objects to represent dynamic models, these objects can be state space or transfer function models.

The proposed linear model for the Chylla-Haase reactor is based on the MATLAB default operating point; two points have been chosen, the input (valve position) and the output (reactor temperature), as an open loop system.

Figure 4.10 demonstrates the linearization method of the Chylla-Haase reactor using MATLAB control design toolbox.

Figure 4.10 shows how to use the Linear Analysis Tool to linearize a model at the operating point specified in the model. The model operating point consists of the model initial state values and input signals.

The Linear Analysis Tool linearizes at the model operating point by default.

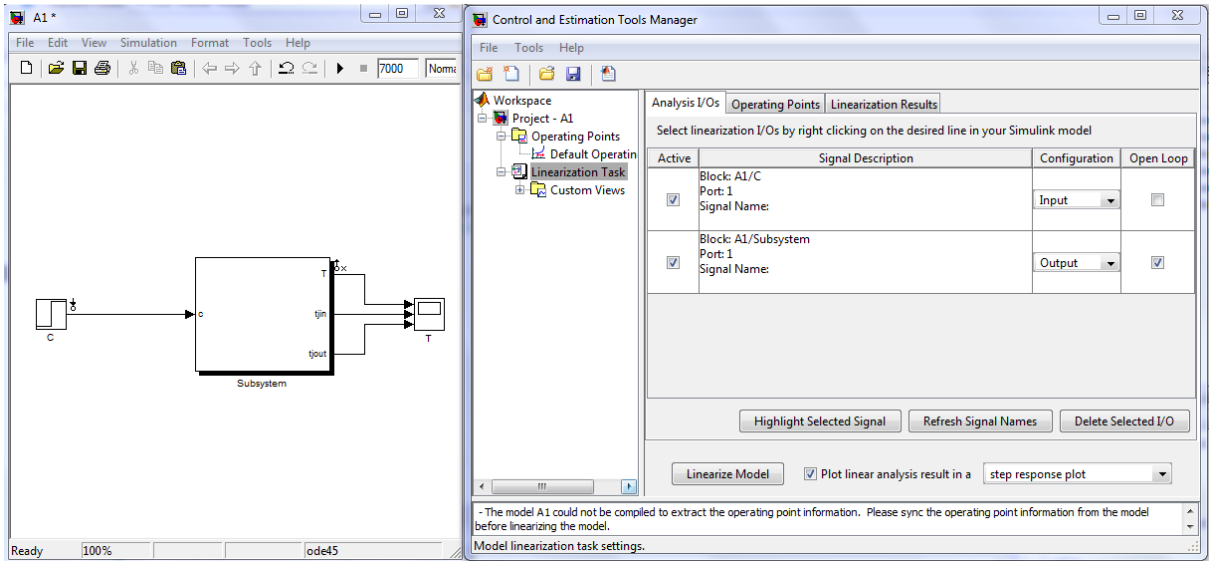


Figure 4.10 Linearization using Simulink control design tools

The transfer function for the linear model after using linearization based on the MATLAB control design toolbox is displayed below:

$$G(s) = \frac{0.00231s+0.000202}{s^3+0.0359s^2+0.000363s+3.598e-007} \tag{4.4}$$

4.3.1 LQR Controller

LQR is a control scheme that provides the best possible performance with respect to some given measure of performance. The LQR design problem is trying to design a state feedback controller K such that the objective function J (equation 4.7) is minimized. In this method a feedback gain matrix is designed which minimizes the objective function in order to achieve some compromise between the use of control effort, the magnitude, and the speed of response that will guarantee a stable system (Bequette, 2003, Ogata).

For a continuous-time linear system described by

$$\dot{X} = Ax + Bu \tag{4.5}$$

Where (A, B) is stabilizable, we want to design state feedback control, equation 4.8, to stabilize the system.

A linear state-space reactor model is developed to ease the control system design and tuning. The state-space model is:

$$A = \begin{bmatrix} 0.01269 & -0.02156 & 0.005126 \\ 0.04131 & -0.04643 & 0.005126 \\ 0.001075 & 0.001075 & -0.002176 \end{bmatrix} \quad (4.6)$$

$$B = \begin{bmatrix} 0.0215 \\ 0 \\ 0 \end{bmatrix}, C = [0 \quad 0 \quad 1], D = 0$$

With a cost functional defined as:

$$j = \int_0^\infty (X^T QX + u^T Ru)dt \quad (4.7)$$

Where Q and R, are positive definite matrices.

The feedback control law that minimizes the value of the cost is:

$$U = -Kx \quad (4.8)$$

$$K = R^{-1}B^T P \quad (4.9)$$

and *P* can be found by solving:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (4.10)$$

4.3.2 Simulation results of LQR control for polymer A

The considered linear model is based on the system described previously in section 4.3.1. The selection of *Q* and *R* is weakly connected to the performance specification, and certain amount of trial and error is required before a satisfactory design result. Generally speaking, selecting *Q* large means that, to keep *J* small, the state *x(t)* must be smaller. On the other hand, selecting *R* large means that the control input must be smaller to keep *J* small. Both the state *x(t)* and the control input are weighted in J, so if *J* is small then *x(t)* and *u(t)* can be too large.

The selected *Q* and *R* weighting matrices are:

$$Q1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, R = 1 \quad (4.11)$$

As the objective in optimal control is to find the state variable feedback matrix *K*, MATLAB can solve this with the codes below.

Figure 4.11 presents the Simulink model of the LQR closed loop system.

sys= ss(A,B,C,D)

K= lqr (sys,*Q*,*R*)

$K1 = [1.0131 \quad 0.3671 \quad 0.3470]$

$Kr = (-C * (A - (B * K))^{-1} * B)^{-1}$,

Kr =1.7460.

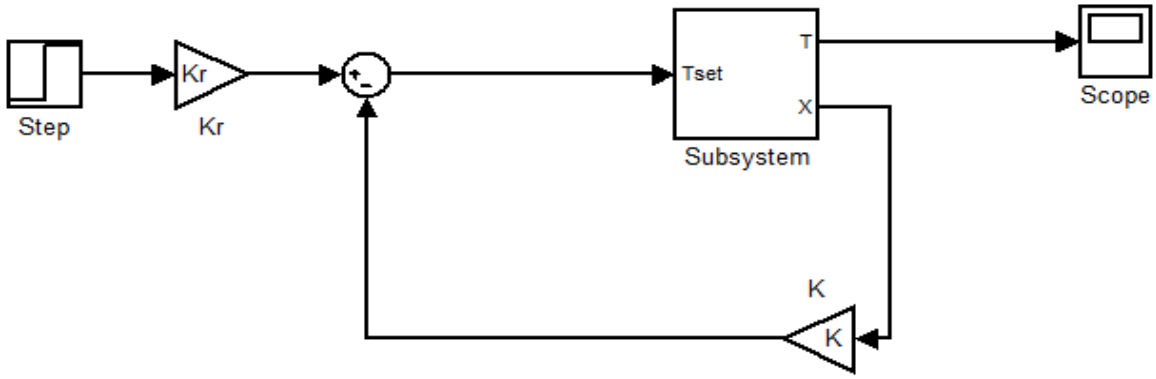


Figure 4.11 LQR subsystem

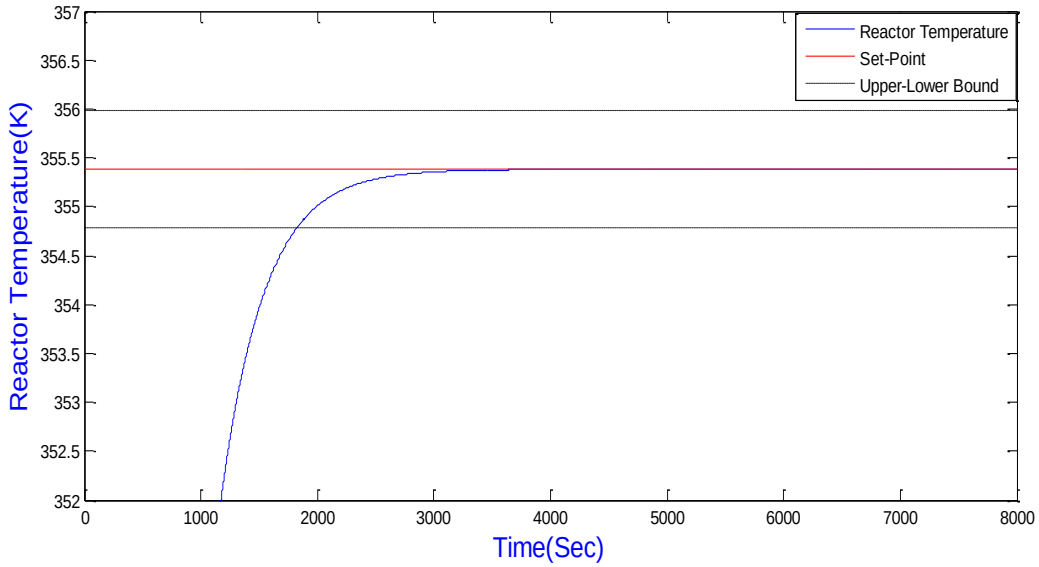


Figure 4.12 Reactor temperature using LQR

It is clear from Figure 4.12 that the control variable shows good performance following the reference input and based on the linear system the result shows no damping or overshoot.

The LQR design was redone with a different control weighting matrix, as the states were known and the most important factor is the reactor temperature which is highly considered in this design.

$$Q2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 50 \end{bmatrix}$$

The new state variable feedback matrix K recalculated by MATLAB:

$$K2 = [1.0168 \quad 0.4190 \quad 5.7592]$$

Kr2= 7.2145

The simulation was run again and the result is shown in Figure 4.13.

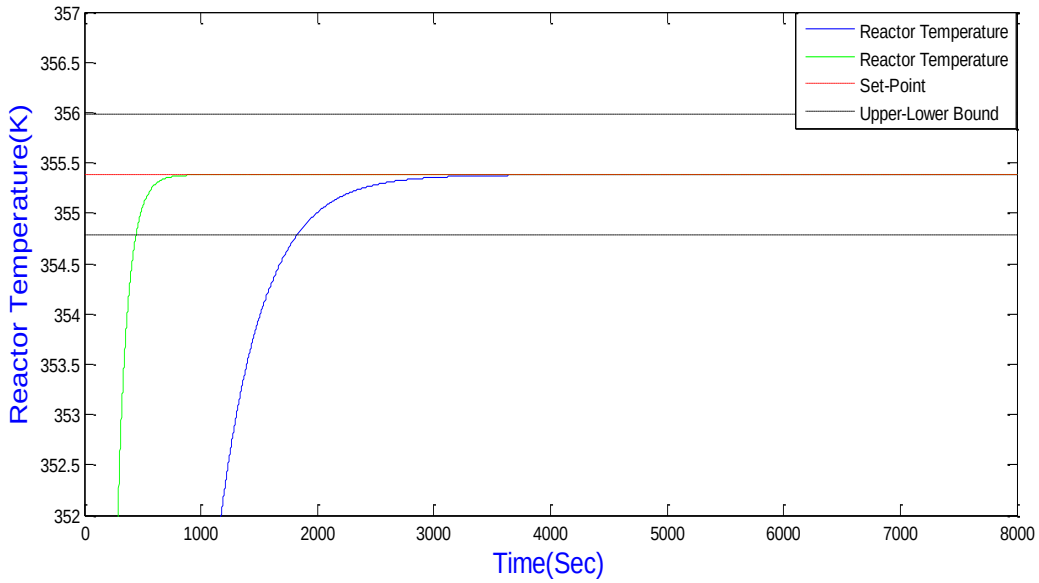


Figure 4.13 Comparison between two LQR designs.

It is clear that the reactor temperature in the second design (green line) is much better, as shown in Figure 4.13 and has a fast response.

4.3.3 Implementation of linear LQR to the nonlinear model for product A

The previous LQR design shows a good result to the linearized Chylla-Haase model, but in fact the system is nonlinear so implementing this design to the nonlinear model is essential.

From the previous design,

Q=eye (3), R=1,

$K1 = [1.0131 \quad 0.3671 \quad 0.3470]$

Kr1= 1.746

Figure 4.14 shows the control of the nonlinear reactor model based on the designed LQR controller. Figure 4.15 (a)-(c) shows the reactor temperature, valve position and error respectively. The performance of the reactor temperature using the LQR controller based on the linear system shows a good result due to the accuracy of the linearized model. However, the behaviour of the valve is not satisfactory for the system based on the linearization assumption, which may lead to instability if there are any mistakes. Moreover, the lack of explicit constraint handling capabilities, from figure 4.15b at the start and the end of monomer feeding, the valve exceed its limitation (constraint), it goes above 100% and below 0%, which is not acceptable in practise.

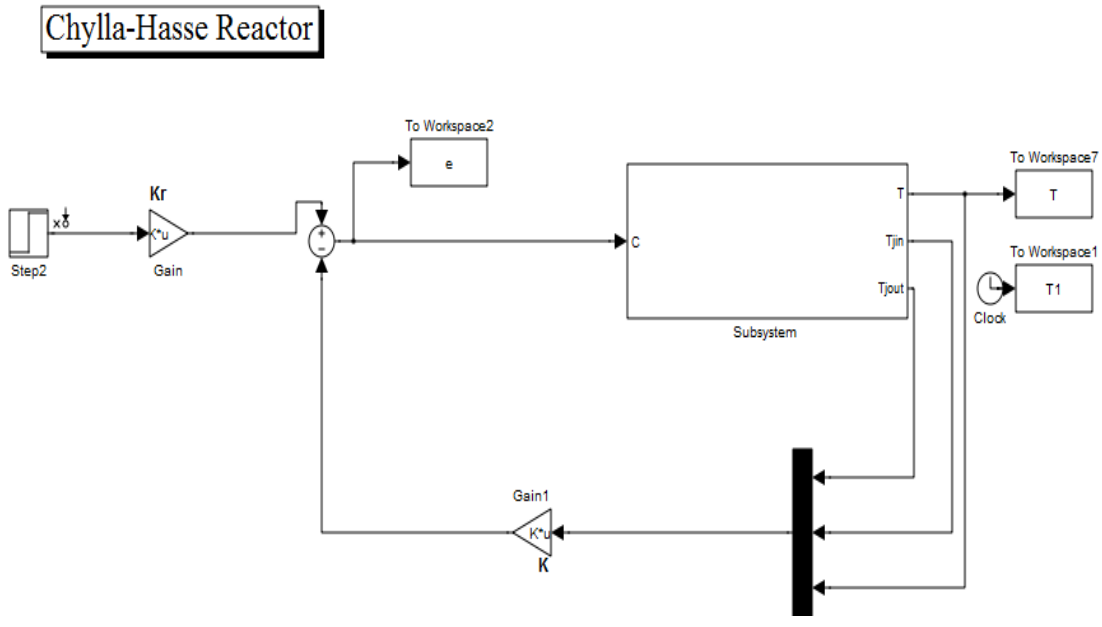
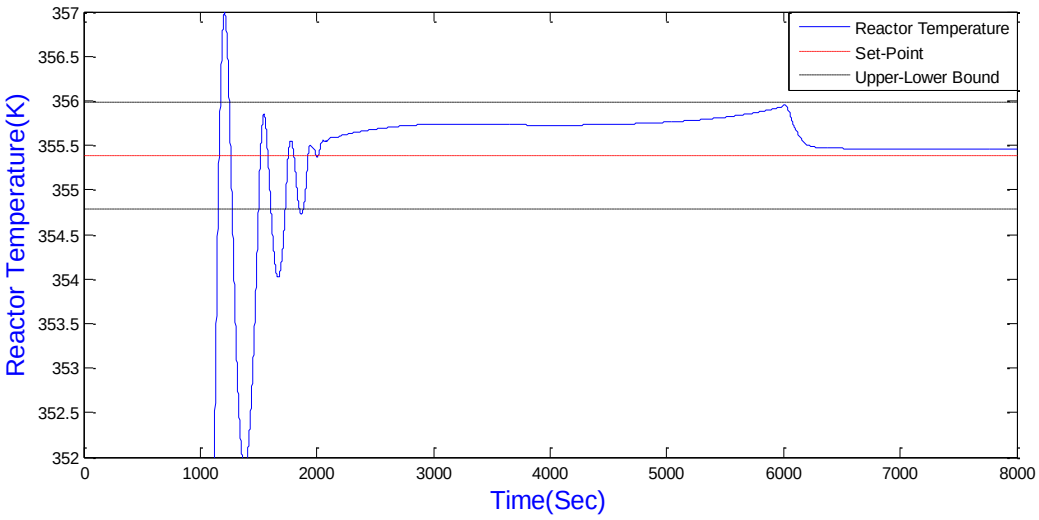
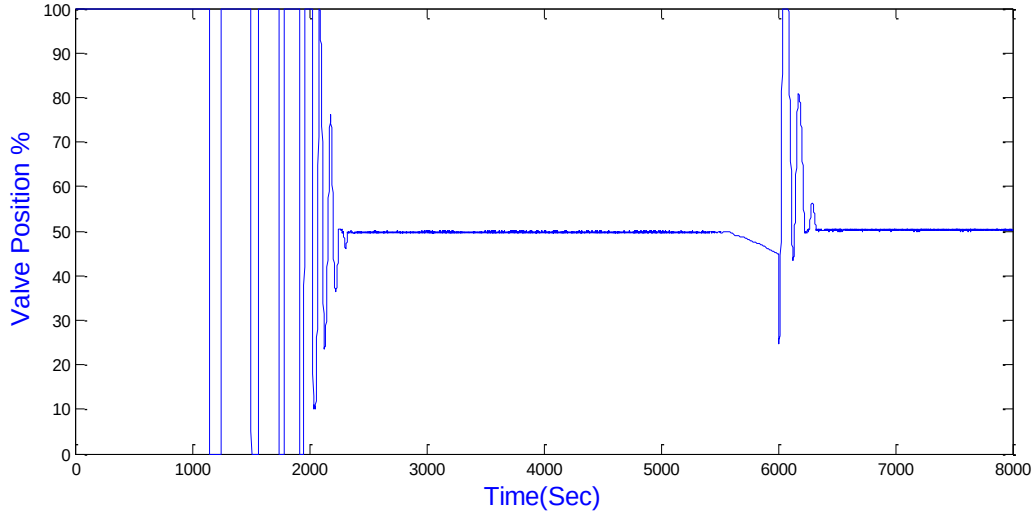


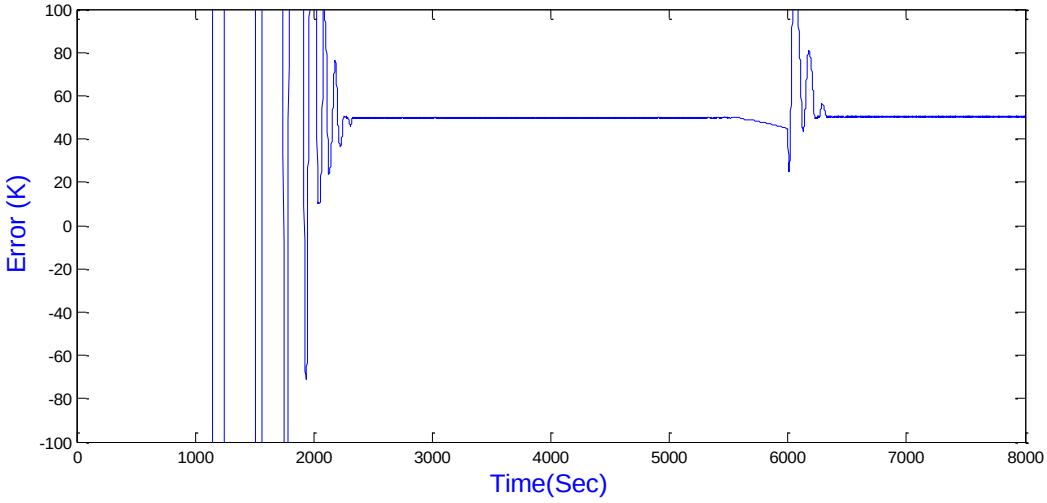
Figure 4.14 Nonlinear model with LQR feedback control.



(a) Reactor temperature with designed LQR controller



(b) Valve position with designed LQR controller



(c) Error.

Figure 4.15 Response of the simulated polymerization reactor for polymer A with designed LQR controller

4.3.4 Simulation results of LQR control for polymer B

Rerun of the simulation took place with the second design as follows:

$$Q2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 50 \end{bmatrix}$$

$$K2 = [1.0186 \quad 0.4190 \quad 5.7592]$$

Kr2= 7.2145

Figures 4.16, 4.17 below shows the simulation result of LQR control design for polymer B with same strategy and construction as product A.

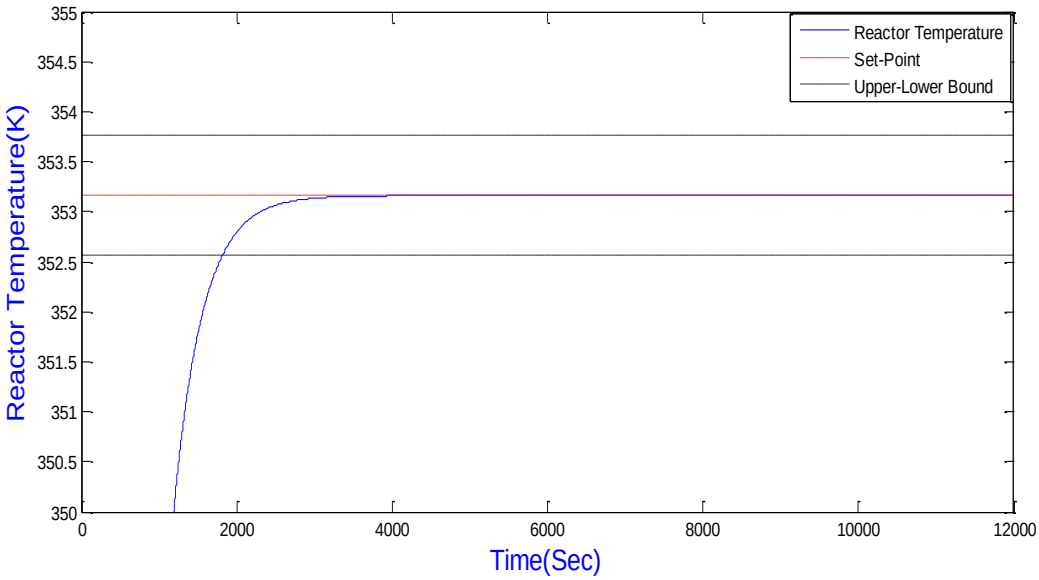


Figure 4.16 Reactor temperature with designed LQR

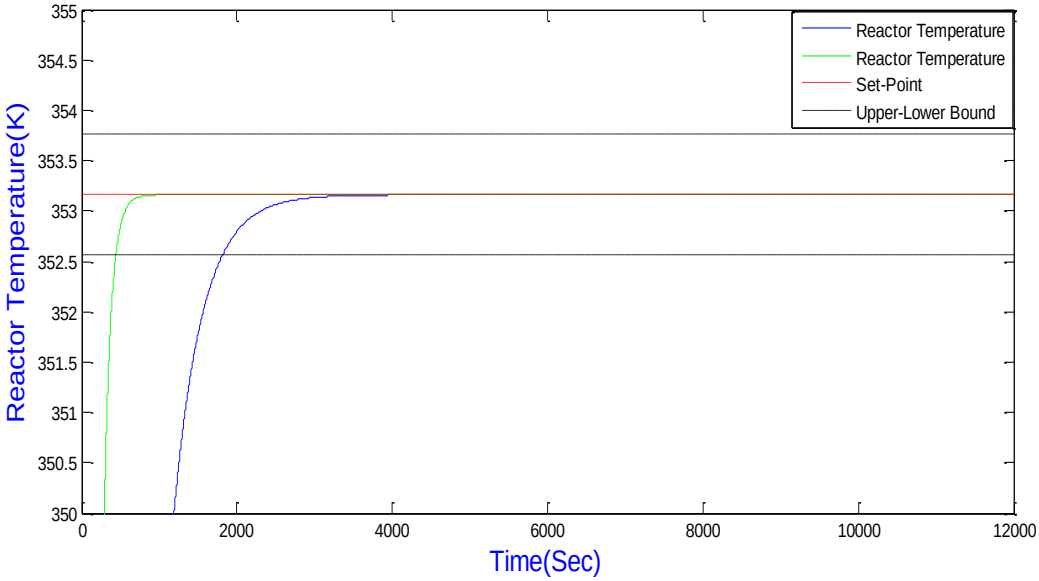
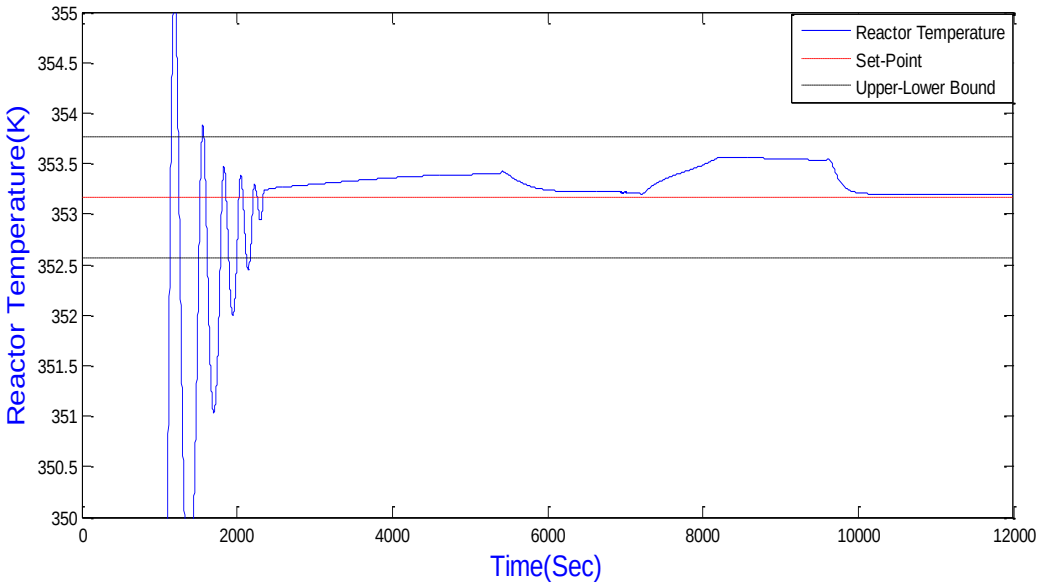


Figure 4.17 Comparison between two LQR designs

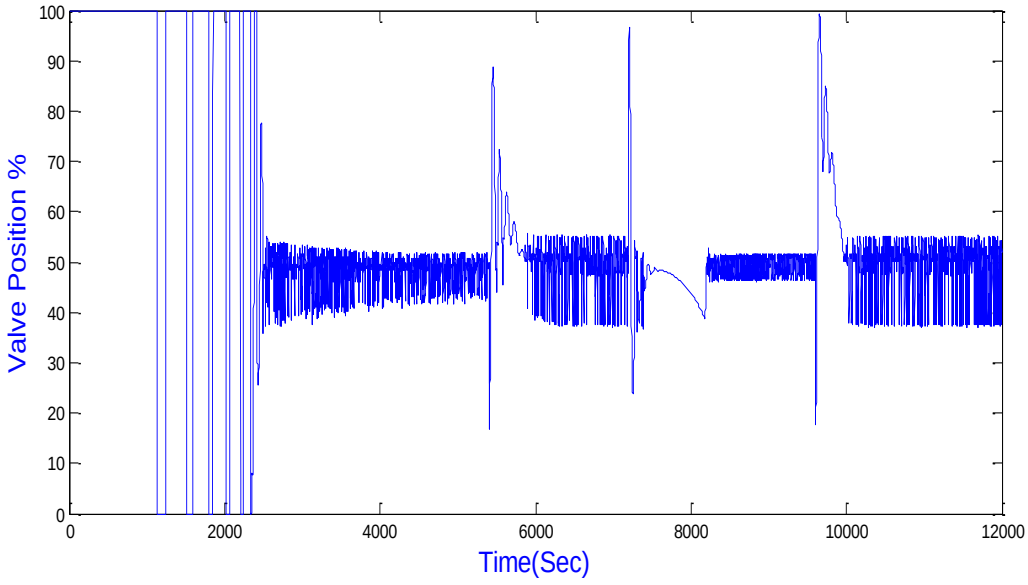
4.3.5 Implementation of linear LQR to the nonlinear model for product B

With the same strategy used in product A, the LQR for product B is capable to maintain the reactor temperature within the tolerance range as shown in Figure 4.18.

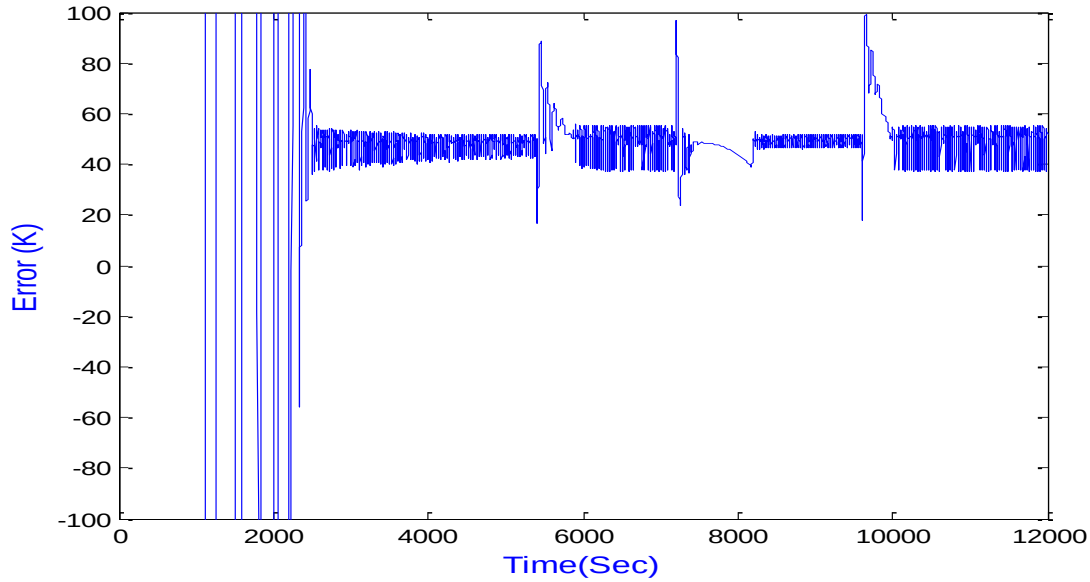
Figure 4.18 (a)-(c) presents reactor temperature, valve position and error. The valve position has a deviation and tackles which still in practise not acceptable.



(a) Reactor temperature with designed LQR controller



(b) valve position with designed LQR controller



(c) Error.

Figure 4.18 Response of the simulated polymerization reactor for polymer B with nonlinear LQR

4.4 Linear MPC

MPC is by far the most commonly applied advanced control technique in chemical process industry. That is because its success and ability to handle constraints. The basic idea behind MPC is shown in Figure 2.1. in chapter 2. At each time step *t*, an optimization problem is solved. An objective function based on prediction over prediction horizon of *P* time step is minimized by a selection of manipulated variable moves over a control horizon of *M* control moves. Although *M* moves are optimized, only the first move is implemented. After *u(t)* is implemented, the measurement of the next time step *y(t+1)* is obtained. Since the measured output *y(t+1)* may not be equal to the model predicted value, a new optimization problem is then solved again over a prediction horizon of *P* steps by adjusting *M* control moves (Robert Haber, 2011).

The Linear MPC used in this research is based on Dynamic matrix control method DMC which basically based on step response model shown in equation 4.12

Key features of the DMC control algorithm include:

• Linear step response model for the plant

• Quadratic performance objective over a finite prediction horizon

• Future plant output behaviour specified by trying to follow the set point as closely as possible

• Optimal inputs computed as the solution to a least-squares problem

The linear step response model used by the DMC algorithm relates changes in a process output to a weighted sum of past input changes, referred to as input moves.

For the SISO case the step response model looks like:

$$\hat{y}_k = \sum_{i=1}^{N-1} s_i \Delta u_{k-i} + s_N u_{k-N} \tag{4.12}$$

Where $\hat{y}_k$ is the model prediction at time step *K*, and u_{k-N} is the manipulated input N steps in the past.

So for the *j-th* step into the future:

$$\hat{y}_k = \sum_{i=1}^{j} s_i \Delta u_{k-i+j} + s_N u_{k-N} + \sum_{i=j+1}^{N-1} s_i \Delta u_{k-i+j} + s_N u_{k-N+j} \tag{4.13}$$

In matrix vector form:

$$\begin{bmatrix} \hat{y}_{k+1} \\ \hat{y}_{k+2} \\ . \\ . \\ . \\ \hat{y}_{k+j} \\ . \\ . \\ \hat{y}_{k+p} \end{bmatrix} = \begin{bmatrix} s_1 & 0 & 0 & \dots & 0 & 0 \\ s_2 & s_1 & 0 & \dots & 0 & 0 \\ . & . & . & \dots & \dots & . \\ . & . & . & \dots & \dots & . \\ s_j & s_{j-1} & s_{j-2} & \dots & \dots & s_{j-M+1} \\ . & . & . & \dots & \dots & . \\ . & . & . & \dots & \dots & . \\ s_p & s_{p-1} & s_{p-2} & \dots & \dots & s_{p-M+1} \end{bmatrix} + \begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ . \\ . \\ \Delta u_{k+M-2} \\ \Delta u_{k+M-1} \end{bmatrix}$$

$$+ \begin{bmatrix} s_2 & s_3 & s_4 & \dots & s_{N-2} & s_{N-1} \\ s_3 & s_4 & s_5 & \dots & s_{N-1} & 0 \\ . & . & . & \dots & 0 & 0 \\ . & . & . & \dots & 0 & 0 \\ s_{j+1} & s_{j+2} & \dots & s_{N-1} & 0 & 0 \\ . & . & . & \dots & \dots & . \\ . & . & . & \dots & \dots & . \\ s_{p+1} & s_{p+2} & \dots & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} \Delta u_{k-1} \\ \Delta u_{k-2} \\ . \\ . \\ \Delta u_{k-N+3} \\ \Delta u_{k-N+2} \end{bmatrix} + s_N \begin{bmatrix} u_{k-N+1} \\ u_{k-N+2} \\ . \\ . \\ . \\ u_{k-N+P} \end{bmatrix}$$

Using matrix notation:

$$\hat{y} = S_f \Delta u_f + S_{past} \Delta u_{past} + S_N u_p \tag{4.14}$$

Where $\hat{y}$ is the predicted output, S_f future dynamic matrix, Δu_finput future move and $S_{past}\Delta u_{past} + S_N u_p$ is the effect of past move.

The difference between the set-point and future prediction is

$$\mathrm{r} - \hat{\mathrm{y}} = \mathrm{r} - \mathrm{S_f}\Delta\mathrm{u_f} - [\mathrm{S_{past}}\Delta\mathrm{u_{past}} + \mathrm{S_N u_p}] \tag{4.15}$$

If we call $r - \hat{y}$ as E (error) then the objective function can be written as

$$\phi = (\mathrm{E} - \mathrm{S_f}\Delta\mathrm{u_f})^{\mathrm{T}}(\mathrm{E} - \mathrm{S_f}\Delta\mathrm{u_f} + (\Delta\mathrm{u_f})^{\mathrm{T}}\mathrm{W}\Delta\mathrm{u_f} \tag{4.16}$$

The solution for the minimization of the objective function is:

$$\Delta\mathrm{u_f} = (\mathrm{S_f^T S_f} + \mathrm{W})^{-1}\mathrm{S_f^T E} \tag{4.17}$$

Summarizing the main steps involved in implementing DMC on a process are as follows:

1. Develop a discrete step response model with length N based on sample time Δt.

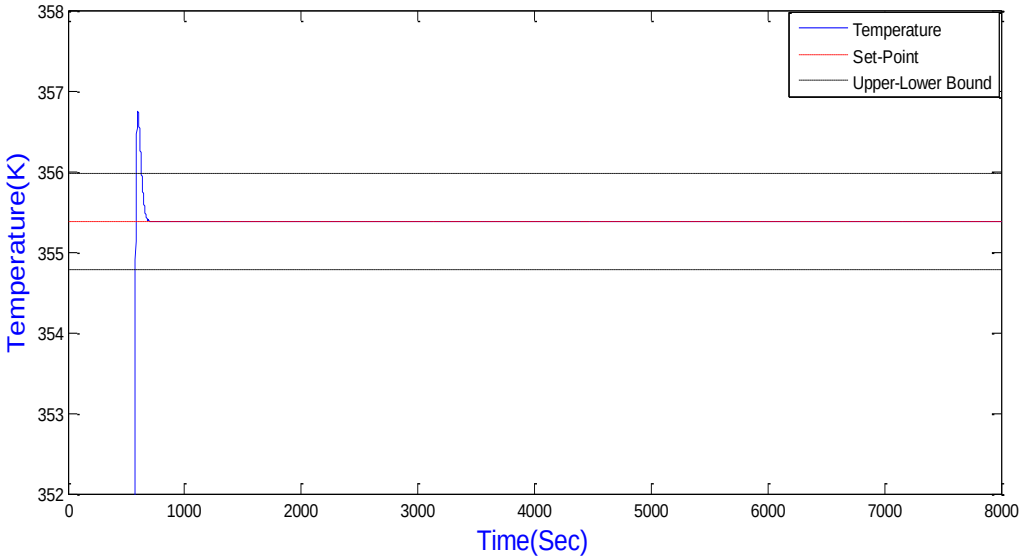
2. Specify the prediction (P) and control (M) horizons. N≥P≥M

3. Specify the weighting on the control action (w=0 if no weighting on the control action (w=0 if no weighting).

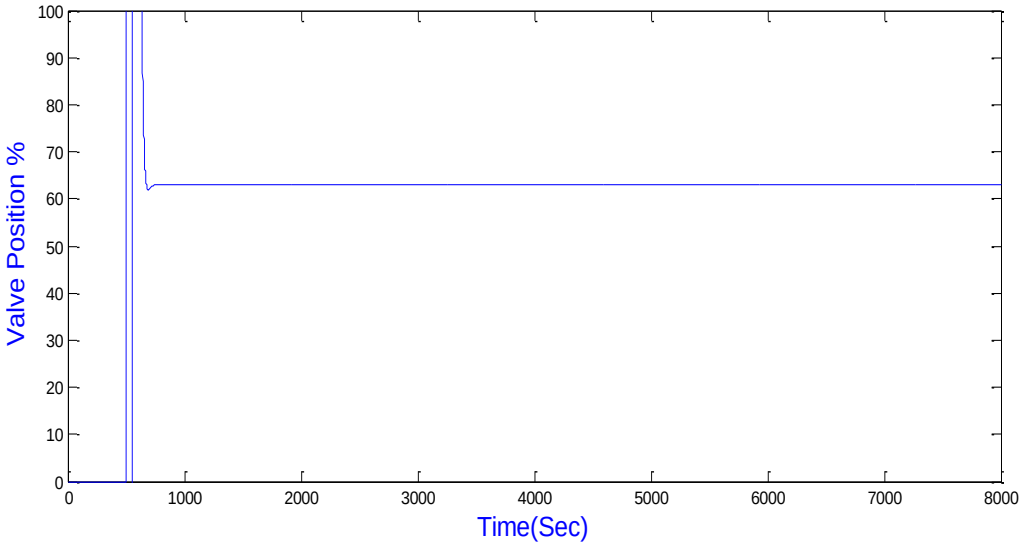
4. All calculations assume deviation variable form, so remember to convert to/from physical units.

Consider Chylla-Haase reactor problem. The continuous state space model is given by equations (4.5, 4.6).

Here the sample time is 4 sec, prediction horizon P=70 and control horizon M=3.



(a) Simulated output with Linear MPC for linearized model



(b) Valve position with linear MPC.

Figure 4.19 Simulation for polymer A with Linear MPC for linearized model

Figure 4.19 (a)-(b) demonstrate the reactor temperature and valve position, as it can be seen the temperature followed the set point perfectly and that's due to the perfection of the linearized system by Matlab tool box as mentioned in previous section.

“Note: no disturbance applied here”.

The importance of the prediction horizon has been tested and shown in Figure 4.20,4.21 below. It is clear that as the prediction P increases as slower response we get, however less prediction horizon requires much more control action.

Figure 4.21 with P=20 the response of the reactor temperature is fast but the control action is reaches 4000 which is not reliable in practice. On the other way with P=400 the response is slower and the input action is more less comparing with P=20.

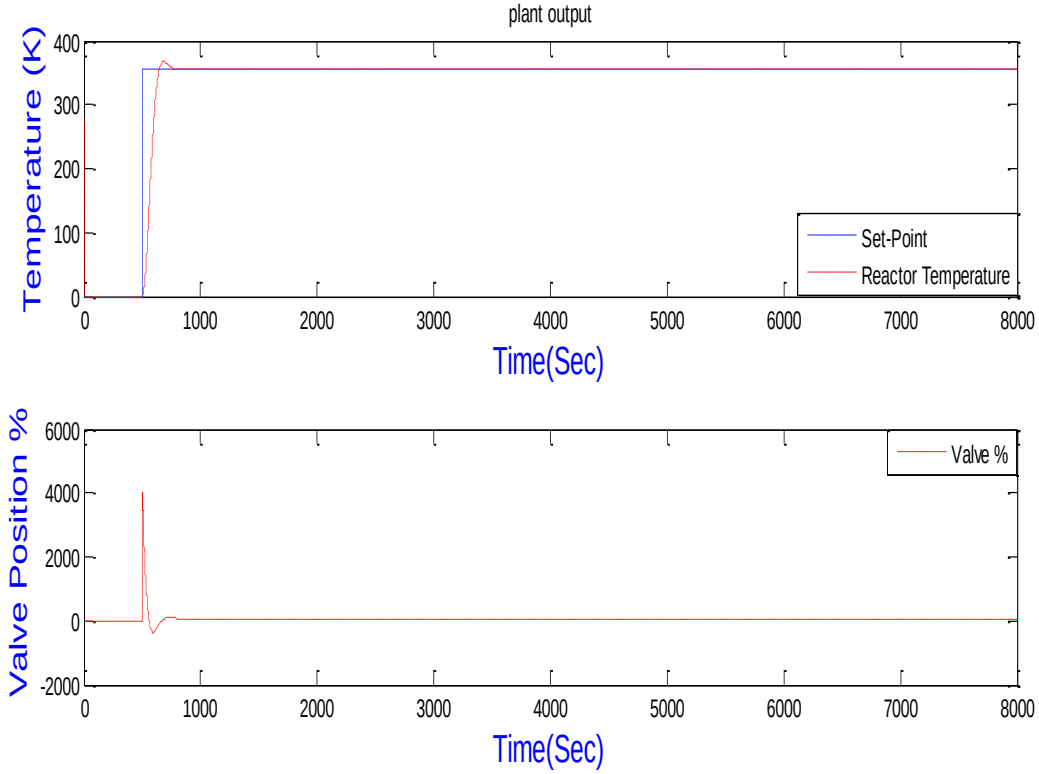


Figure 4.20 Simulation for polymer A with Linear MPC for linearized model. Prediction horizon, P=20

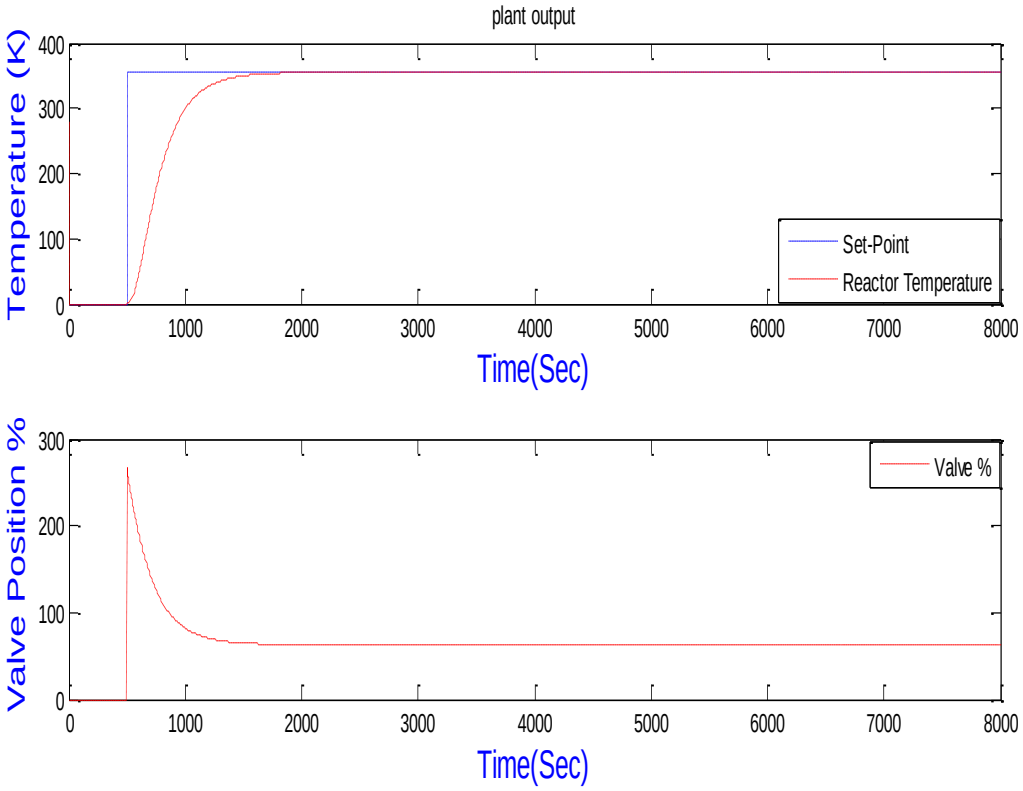


Figure 4.21 Simulation for polymer A with Linear MPC for linearized model. Prediction horizon, P=400

4.4.1 Implementation of the linear MPC to the nonlinear model for product A

For the design of model predictive control based on linearized model, MPC Toolbox in the MATLAB software used in this work. Model predictive control toolbox provides Matlab functions, graphical user interface and Simulink blocks, for designing and simulating MPC in matlab and Simulink. This controller optimize the performance of the system subjected to input/output constraints. Figure 4.22 demonstrate MPC Simulink block with Chylla-Haase reactor.

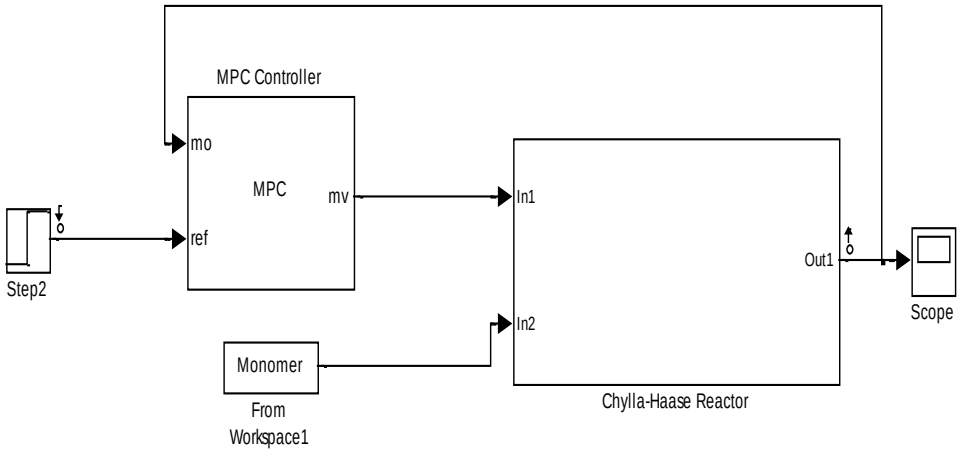


Figure 4.22 Simulink model for closed-loop simulations of a model predictive controller and a nonlinear plant model.

Definitions of MPC block shown in Figure 4.22 above are given as, mo, mv and ref, measured output, manipulated variable and reference signal respectively.

Few steps demonstrate the procedures to design MPC using Matlab toolbox:

First step is to extract the linearized model using the previous method presents in section 4.3 (LTI method) or by MPC blocks itself which is able to import the linearized model created from measured input-output data using System Identification.

Secondly, from the MPC blocks showed above designing controllers by specifying the controller parameters includes, prediction horizon, control horizon, sampling period, weights on manipulated variables, manipulated variable rates, and output variables as shown in Figure 4.23, 4.24 below.

Model Predictive Control Toolbox provides several tools to optimize controller performance by adjusting controller constraints and weights.

.

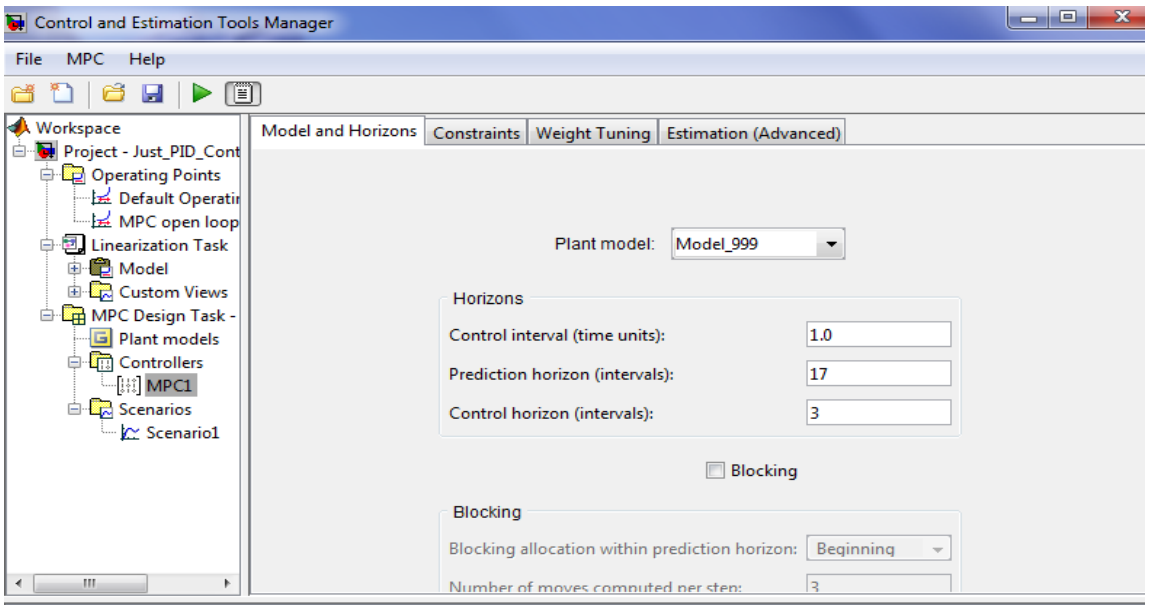


Figure 4.23 Dialog box for specifying the parameters of MPC.

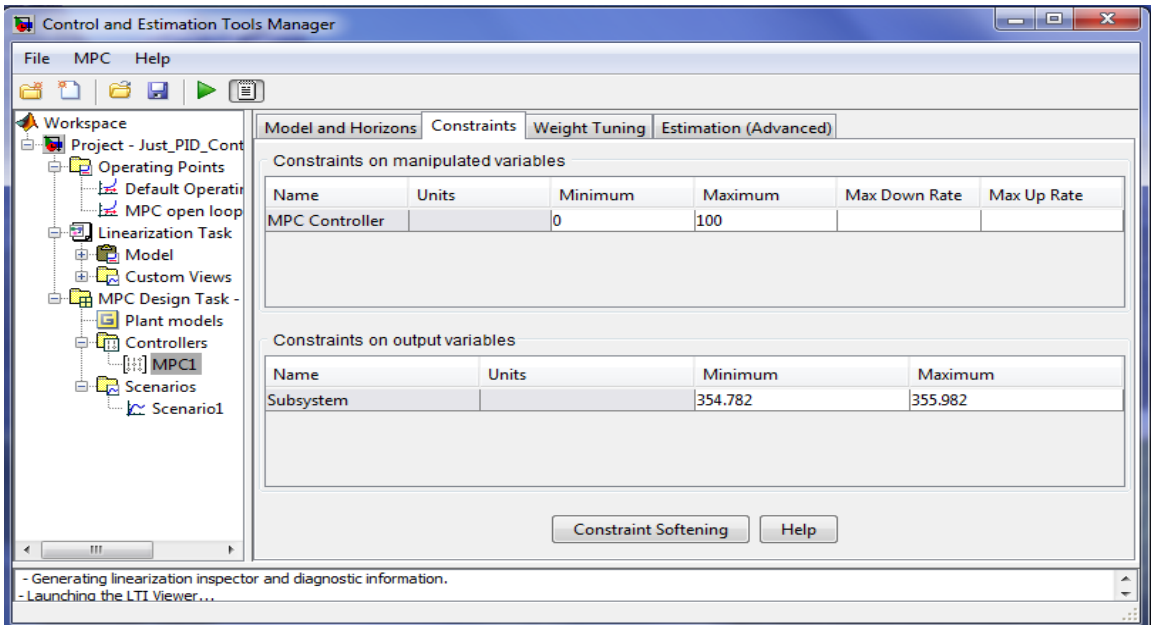


Figure 4.24 Dialog box for setting constraints.

Finally, Running Closed-Loop Simulations against linear plant models as shown in Figure 4.25.

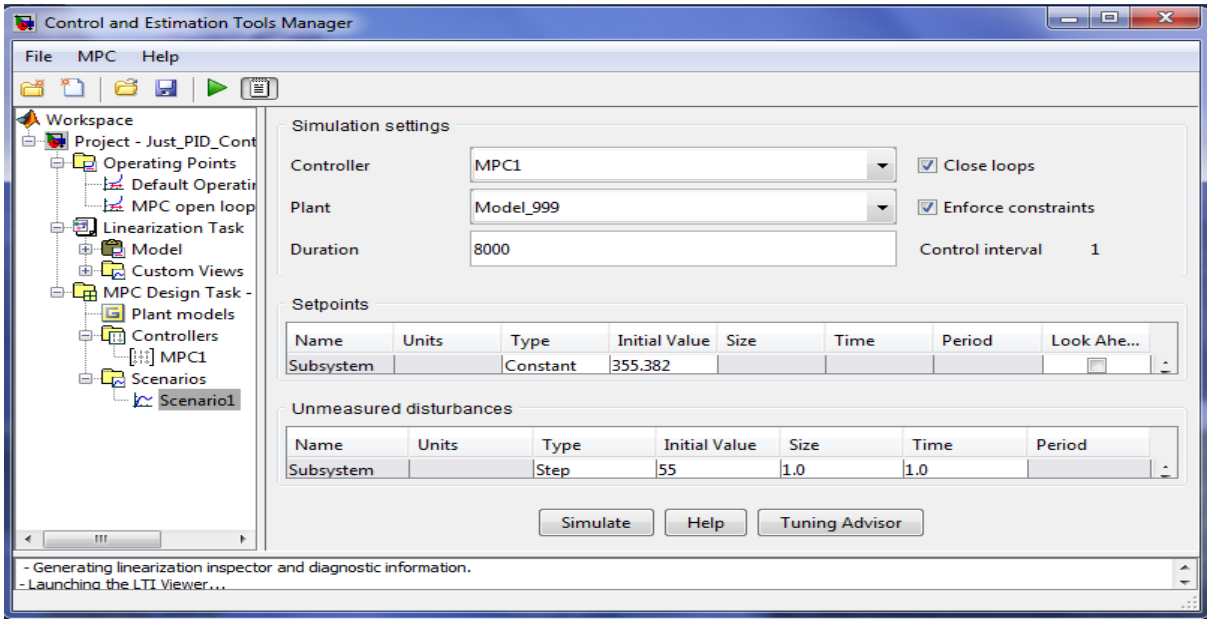


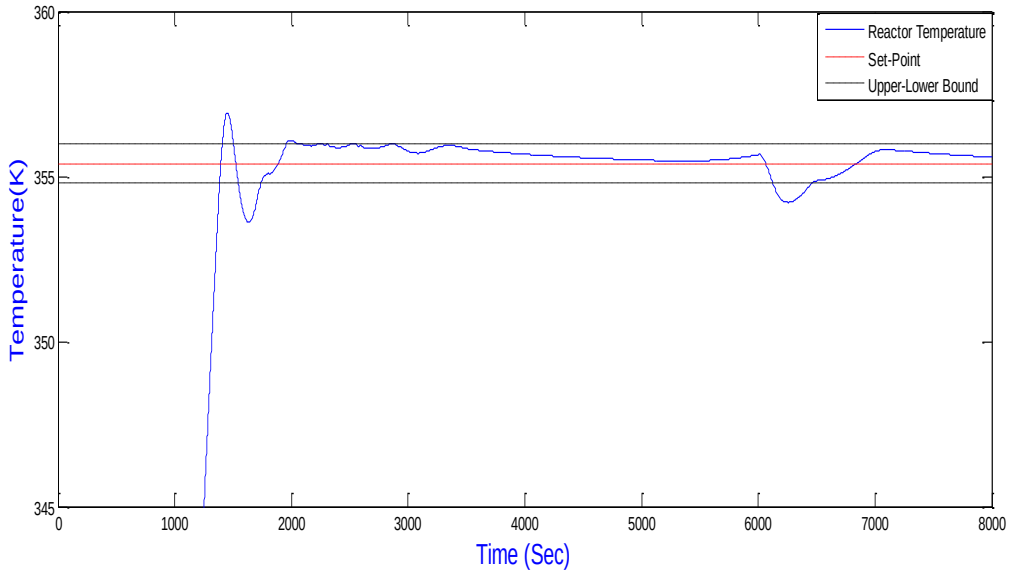
Figure 4.25 Dialog box for running simulation.

Obtained the results shown in Figure 4.26 (a)-(b). After fine tuning the best result found based on:

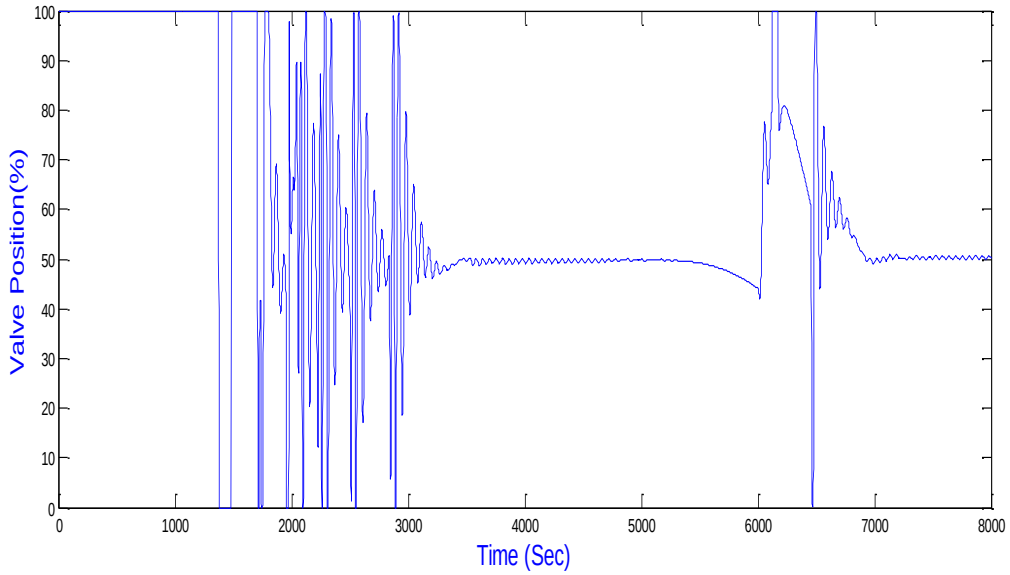
Prediction horizon=17, control horizon =3.

As it can be seen from Figure 4.26, the MPC controllers based on linearized plant model is not able to robustly operate the system when the monomer inlet flow is stop at 6000 sec. is clearly observed at the end of the feeding period temperature fall beyond the lower bound of tolerance limit.

Although an MPC controller can regulate a nonlinear plant, the model used within the controller is linear. In other words, the controller employs a linear approximation of the nonlinear plant. The accuracy of this approximation significantly affects controller performance.



(a) Reactor temperature with designed MPC.



(b) Valve position with designed MPC.

Figure 4.26 Simulation for polymer A with nonlinear MPC

4.5 Summary

This chapter presents conventional control methods designed using linearized model. In reality, a process model is only approximate representation of the dynamic behaviour of the system.

Methods include:

- The PID Tuner using Matlab tool box is applied and results show that this PID cannot provide a robust control due to operating condition change (summer and winter), and disturbance effect.
- Cascade control provides a robust operation but often lacks in control performance concerning the required strict temperature tolerance.
- LQR control is designed based on linear model, in practical linear model cannot effectively describes the dynamic behaviour for a highly nonlinear exothermic reactions.
- LMPC here based on step response model which is linear. For Chylla-Haase reactor where the operating conditions are changed frequently, a single linear model cannot describe the dynamic behaviour of the process over the wide range of conditions. So better control performance need to be achieved for the nonlinear process which comes in next section.

The simulation results for the temperature control of Chylla-Haase polymerization reactor using PID control, cascade control, LQR control and LMPC control scheme was presented. The results shown that traditional controllers could not lead to acceptable responses because of nonlinear characteristics of Chylla-Haase reactor. It was proved that, they are incapable of keeping reactor temperature within the tolerance range in the presence of different conditions and disturbances. When the reactor is with different disturbances effect, these controllers need to be adapted online to reject disturbance and keep the temperature within the tolerance range.

Chapter 5 Neural Network Modelling for the Process

5.1. Introduction

Polymerization processes present serious difficulties for modelling, requiring considerable effort due to high non-linearity and physical properties. The requirement of an accurate process model for the quality control is very important, concerning the required strict temperature tolerances for quality product and safety. Development of models to predict the polymer quality under the process operating conditions in a polymerization reactor has been proven to be a very important methodology to efficient control of product quality and was reported by (Dubé et al., 1996, Noor et al., 2010).

Neural networks are well known for their ability to model and control nonlinear dynamic systems, having been used since the last decade. A neural network is a straightforward choice for representing the behaviour of dynamical processes (Yu and Gomm, 2003). A large number of architectures and algorithms for identification and control using NNs were proposed. For example, (Kandroodi and Moshiri, 2011, Nagy et al., 2007, Ng and Hussain, 2004) used a neural network to approximate unknown system dynamics by measuring the state variables of one batch for training. (Mills et al., 1993, Gomm et al., 1996) have shown that the use of prior knowledge in combination with a neural network structure results in an optimum model. Neural networks appear to be able to model any nonlinear functions essential to control systems with higher degree of dependence because of their ability to learn, to approximate functions, to classify patterns and because of their potential for particularly parallel hardware implementation.

Radial basis function (RBFNN) is a type of feed forward neural network that has universal approximation abilities. Similar to the backpropagation networks (BPNN), RBFNN has also good approximation property. It has been acknowledged that

approximation accuracy properties of RBFNN are advantageous as compared to the other methods. Even more important for many applications, the RBFNNs provide linear approximation in the network weights. This feature makes powerful tools of the linear system theory applicable to the RBFNN identification of nonlinear systems. The "linear in parameters" of the radial basis functions guarantees the convergence of the parameters to the global minimum. Furthermore, RBFNNs are not as sensitive to the architecture as BPNNs (McLoone et al., 1998, Jung-Wook et al., 2002).

Many types of NN that can be used to model dynamic system, such as Radial basis function (RBF), diagonal recurrent neural network (DRNN), recurrent neural network (RNN) and multi-layer perceptron network (MLPN). RBFNN and MLPNN have been chosen in this study due to their simple structure and that they are easy to train.

5.2. Radial basis function neural networks

5.2.1 RBFNN Structure

RBF network structure is divided into three layers showed in Figure 5.1. The first layer is input layer, the second layer is hidden layer. And the third layer is output layer. The first layer transfer input signal to the hidden layer, hidden layer nodes configuration by the Gaussian function and output layer nodes is linear function.

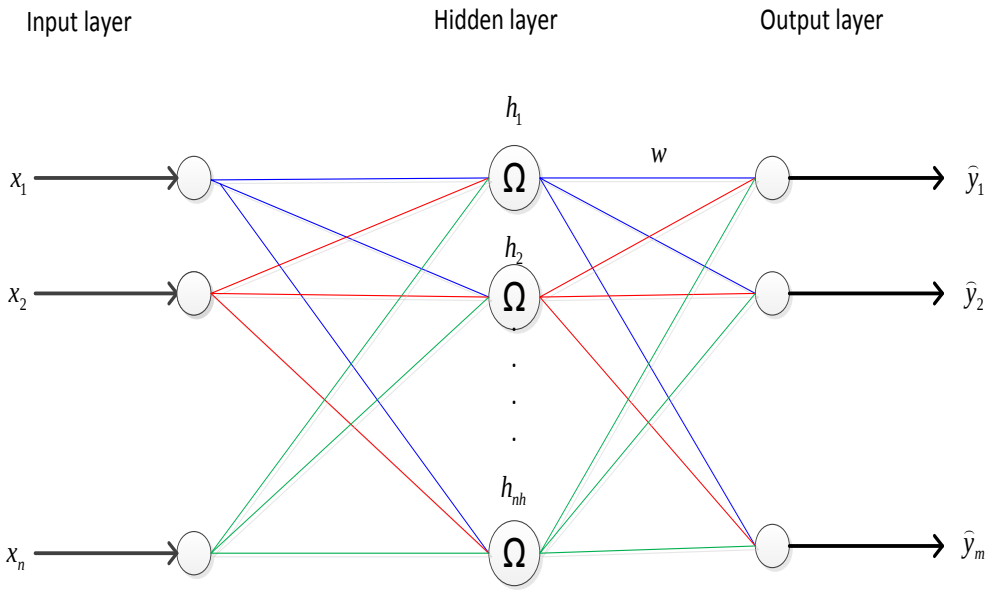


Figure 5.1 The structure of RBFNN

From Figure 5.1 the network output can be obtained by:

$$\hat{y}_i = \sum_{j=1}^{nh} W_{i,j}\ h_j \qquad i = 1, \ldots \ldots m \tag{5.1}$$

Where $\hat{y}$ is the network output (predicted process output), h_j denotes the radial basis function of the *j-th* hidden node, $W_{i,j}$ denotes the hidden-to-output weight corresponding to the *i-th* hidden node, and *j-th* output, and n_h is the total number of hidden nodes.

Hidden nodes number depends on the complexity of the problem to determine, more nodes can improve the network accuracy, but network training will be long. There are various well-known radial basis function are recovered to calculate the Euclidean distance between the centre and the network input vector X_1 (Howlett and Jain, 2001), for example the plate splin function in (5.2), the reciprocal multi-quadric function (RMQ) in (5.3) and Gaussian basis function in (5.4).

$$\emptyset_i(t) = \|X(t) - C_i(t)\|^2 \log(X(t) - C_i(t)) \tag{5.2}$$

$$\emptyset_i(t) = (\|X(t) - C_i(t)\|^2 + \sigma)^{-1/2} \tag{5.3}$$

A normalized Gaussian function usually used as the radial basis function and commonly used here, is:

$$\emptyset_i(t) = \exp\left(\frac{\|X(t) - C_i(t)\|^2}{\sigma_i^2}\right), \qquad i = 1, \ldots \ldots, nh \tag{5.4}$$

Where, X is the input vector, σ_i is a positive scalar called width, which is a distance scaling parameter to determine over what distance in the input space the unit will have a significant output and C_i is the number of centres.

5.2.2 Training algorithm

Once the structure of the RBF networks has been selected the training of the RBF aims to determine appropriate values of the RBF centres, weights and width of the hidden layer outputs, K-means clustering algorithm is used to choose the centres of the RBF to minimize the sum squared distance from each input data to its closest centre so that

the data is adequately covered by the activation function, *p*-nearest algorithm decides the widths, and the recursive training algorithm calculates weights. All this algorithms are discussed in brief below.

- **Calculation of Centres**

The neural networks centres describes by K-means algorithm, the aim of the K-means clustering method is to minimize the sum squared distance from each input data to its closest centre so that the data adequately covered by the activation function. Steps below briefly describes clustering data procedure (Orr, 1996, Yu and Zhai, 2008):

1. The data set is separated into K clusters and the data points are randomly assigned to the clusters resulting in clusters that have roughly the same number of data points.
2. For each data point:
   2.1. Calculate the distance from the data point to each cluster.
   2.2. If the data point is closest to its own cluster, leave it where it is. If the data point is not closest to its own cluster, move it into the closest cluster.
3. Repeat the above step until a complete pass through all the data points results in no data point moving from one cluster to another. At this point the clusters are stable and the clustering process ends.
4. The choice of initial partition can greatly affect the final clusters that result, in terms of inter-cluster and intracluster distances and cohesion.

- **Width Calculation**

The neural network widths computed by the *p*-nearest neighbour's method. The excitation of each node should overlap with other nodes (usually closest) so that a smooth interpolation surface between nodes is obtained. In this method, the widths for each hidden node are set as the average distance from the centre to the *p* nearest centres as given by:

$$\delta_i = \sqrt{\sum_{j=1}^{p} \|c_i - c_j\|^2} \qquad i = 1, \ldots, nh \tag{5.5}$$

The value of *p* is chosen according to different problems.

- **Recursive Least Squares Algorithm (Weight Estimation)**

After finishing with the calculation of the centres and widths of the RBF**,** the objective is to minimize the error between the observed output and desired one. It is commonly trained using the RLS algorithm. The RLS method is used for on-line training. It helps to construct on-line system models which should be more accurate and flexible than the off-line fixed models.

The equations of this algorithm are summarized as follows (Yu and Zhai, 2008).

$$y_p(k) = y_c(k) - W(k-1)\emptyset(k) \tag{5.6}$$

$$g_z(k) = \frac{P_z(k-1)\emptyset(k)}{\mu + \emptyset^T(k)P_z(k-1)\emptyset(k)} \tag{5.7}$$

$$P_z(k) = \mu^{-1}\,[\,P_z(k-1) - g_z(k)\emptyset^T(k)P_z(k-1)] \tag{5.8}$$

$$W(k) = W(k-1) + g_z(k)y_p(k) \tag{5.9}$$

Where, $W(k)$ represent the RBF network weights which is trained initially and then on-line updated using the RLS algorithm, $\emptyset(k)$ is the activation function output, $y_c(k)$ is the process output vector, $P_z(k)$ and $g_z(k)$ are middle terms, μ is called the forgetting factor ranging from 0 to 1 and chosen to be 1 for offline training. The parametersP_z, g_z, and W are updated orderly for each sample with the change in the activation function output $\phi(k)$.

5.2.3 Chylla-Haase reactor modelling using RBF

RBF neural networks have been applied very successfully in the identification and control of dynamic systems, and arise in problems of approximation and in problems of learning input/output mappings from given set of data.

When attempting to identify a model of a dynamic system, it is common practice to follow the procedure below.

5.2.3.1 Data collection and scaling

The purpose of this task is to collect a set of data that describes how the system behaves over its entire operating range. The input signal should be able to cover the entire operating range of the system. For nonlinear model, it is important that the input signal should represent all amplitudes and frequencies.

From a modelling point of view, a proposed model of the Chylla-Haase reactor links the output variable (temperature T) to the control variable (valve position C) and to the disturbances (monomer feed rate $\dot{m}_M^{in}$, fouling factor $1/h_f$, impurity factor i and ambient temperature T_{amb}). Firstly, A set of random amplitude signals (RAS) were applied to only the first reactor input C (valve position) and the second input here is considered to be the monomer feed rate, which is a fixed value at a specific time as shown in Figure 5.3. The valve position range used in simulation is from (0% ~100%) and the monomer feed is (0~0.0075) kg/s with fixed time for product A and (0~0.00641) kg/s for product B (longer batch). See table 3.3, chapter 3.

Data were collected for reactor temperature at each sample time. The random amplitude signals (RAS) were designed (0~2000 samples) for polymer A and (0~3000 samples) for polymer B to cover the operating range as shown in figure 5.2, 5.3, 5.4 and 5.5.

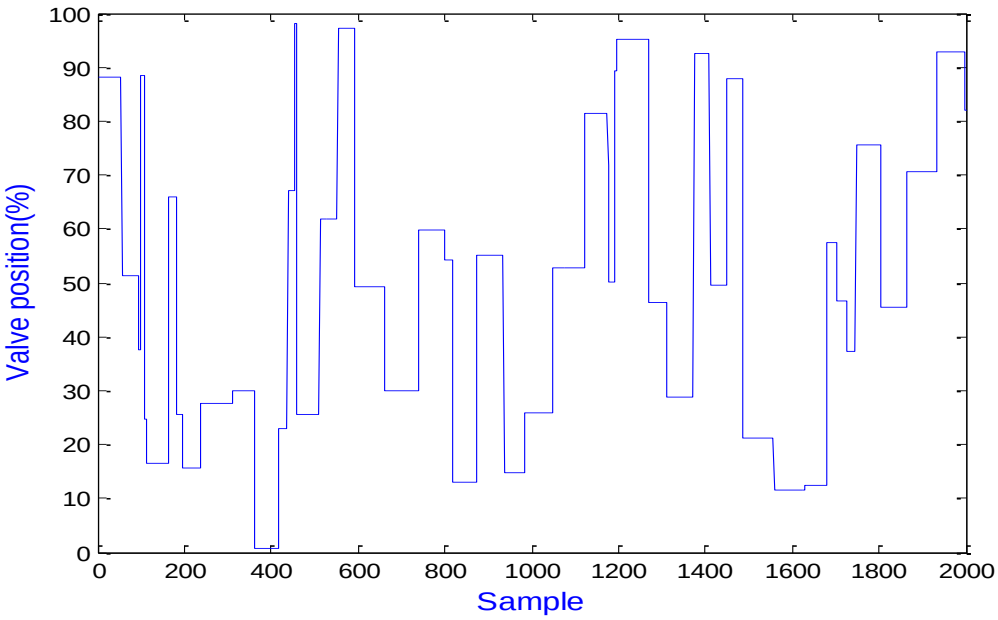


Figure 5.2 RAS of Valve position for polymer A

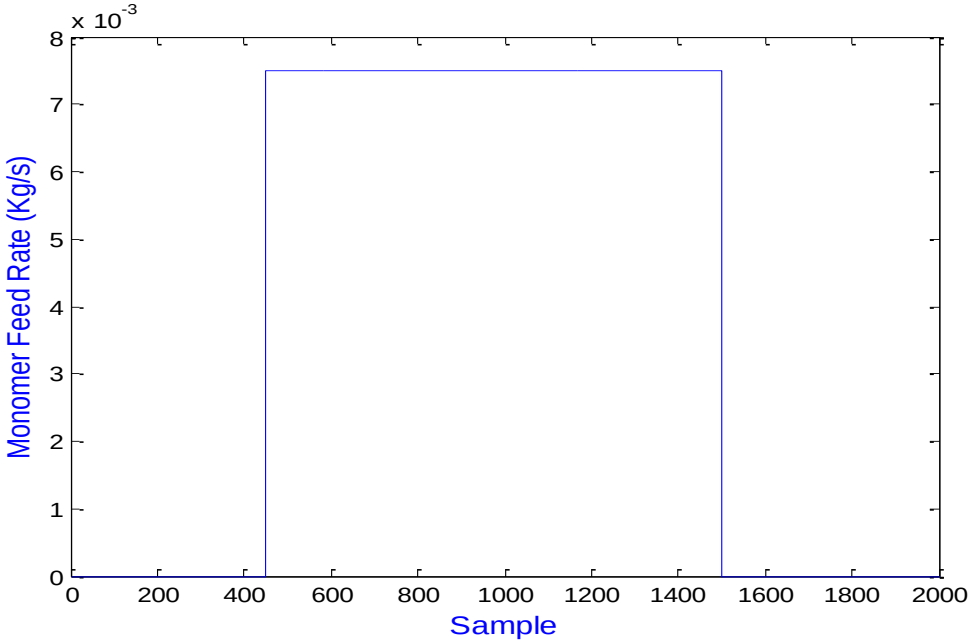


Figure 5.3 Monomer feed Rate for Polymer A

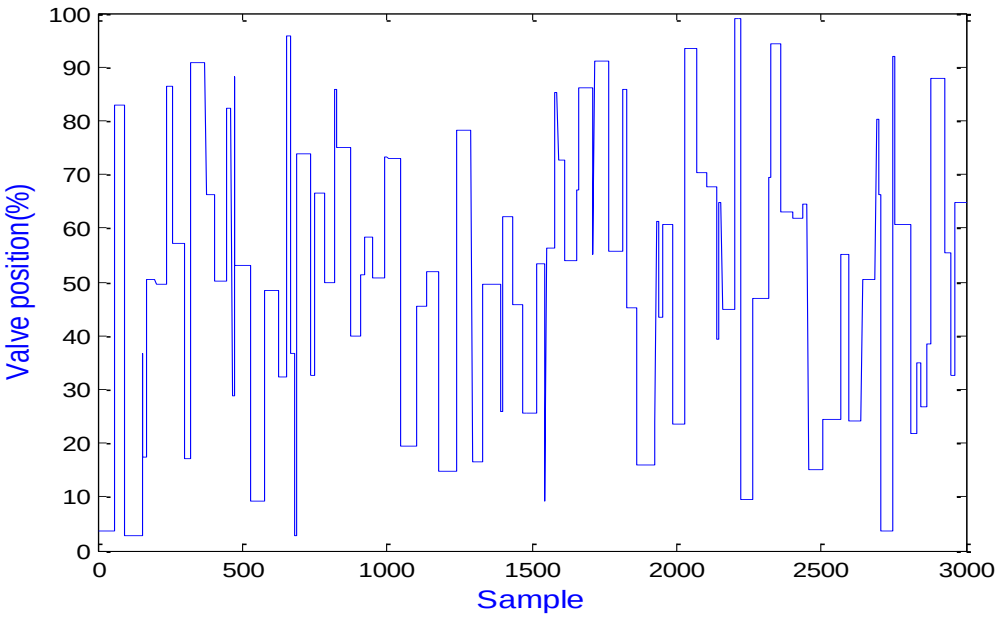


Figure 5.4 RAS of Valve position for polymer B

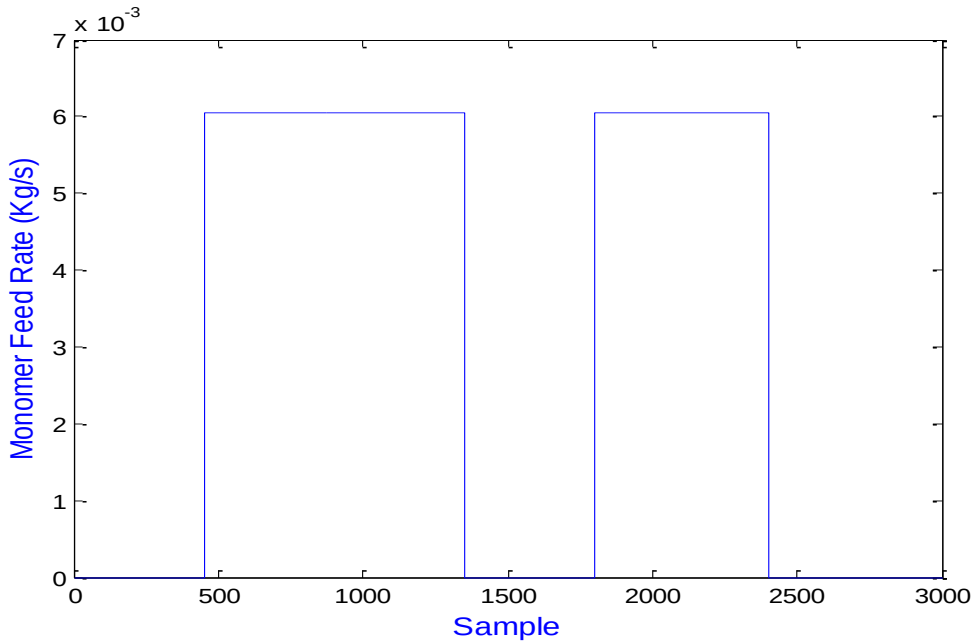


Figure 5.5 Monomer feed Rate for Polymer B

Selection of the sampling time plays a major role. In this case sampling time is chosen to be 4 seconds after trial and error, as it has been found a high sampling time provides a fast reference tracking and smoother signal in identification task but in control design the higher sampling time the less controllable signal as more dynamic information missing.

It is important and recommended to scale all the signals to avoid attributes in greater numeric ranges dominating those in smaller numeric ranges and more important, proper data makes the training algorithm numerically robust and lead to fast convergence. In this work the data set scaled between 0 and 1 by using the following equations:

$$u_s(k) = \frac{u(k)-u_{min}}{u_{max}-u_{min}} \tag{5.10}$$

$$y_s(k) = \frac{y(k)-y_{min}}{y_{max}-y_{min}} \tag{5.11}$$

where, u_{min} and u_{max} are the minimum and maximum inputs among the data set, while $u_s(k)$ and $y_s(k)$ are the scaled input and output respectively. To testify validity of the NN model, data divided into two parts, the first data set is used for training neural network and the second different data is used for NN model validation. Generally, the modelling error of the training data set is often smaller than the test data set. The mean absolute error (MAE) is used to evaluate the modelling and control performance in this research, which is given by the following equation:

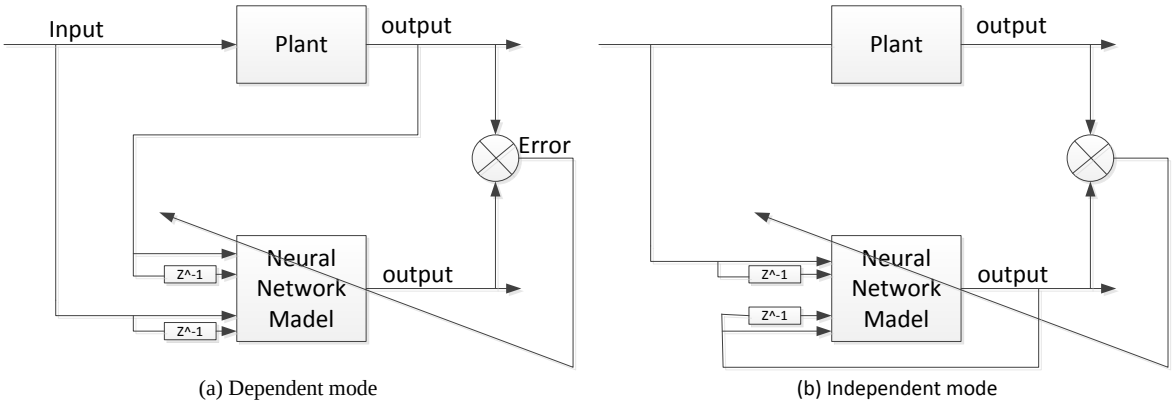
$$\text{MAE} = \frac{1}{N}\sum_{k=1}^{N}|\hat{y}(k) - y(k)| = \frac{1}{N}\sum_{k=1}^{N}|e(k)| \tag{5.12}$$

Where $\hat{y}(k)$ is the predicted output by the neural network model and *y(k)* is the output of the Chylla-Haase model.

5.2.3.2 Model structure selection

Using neural network for modelling a non-linear dynamic system can be classified into two models, dependent model and an independent model as shown in Figure 5.6. The dependent model and an independent model are implemented in parallel with the system. The first model referred to dependent model, since the past system output is used as network inputs. Thus, the model is dependent on the system output and cannot operate independently from the system. In the independent model, the past model

output is used as network inputs. Therefore, the model is not dependent on the system output, and can operate independently from the system. The independent model has an advantage, in which the model can be used to simulate the system to obtain long-range prediction. In opposite, the dependent model is performing as one-step-ahead predication (Orr, 1996).



(a) Dependent mode (b) Independent mode

Figure 5.6 Modelling mode using NN modelling

However, dependent mode more accurate to mimic the actual output but in fact the independent mode can be used as a multi-head prediction which make the independent mode has superb to be used in model predictive control and many applications.

This work based on independent model and from experiment and analysis of reactor temperature it's been found that, the reactor temperature is a function of valve position and monomer feed rate. According to this analysis three variables were chosen to be the network inputs from the reactor simulation, valve C, Monomer feed $\dot{m}_M^{in}$, and reactor temperature T. Considering that the two inputs and signal output nonlinear process can be given as a discrete time model:

$$\hat{y}(k+1) = f[y(k-1), \dots\dots, y(k-n), u_1(k-1), \dots\dots, u_1(k-m), u_2(k-1), \dots\dots, u_2(k-l)] \quad (5.13)$$

Where $y(k)$is the process output, $u1(k)$ is the first input, $u2(k)$ is the second input (disturbance) and $f(.)$ is the unknown non-linear function, n, m, l respectively are the orders of the output and input.

Third order structure with 23 hidden nodes has been chosen because they give a minimum prediction error, The structure was selected based on testing different networks that vary in terms of structure and simulation parameters, Figure 5.7 illustrate the structure of the RBF neural networks for modelling and the neural network model of Chylla-Haase reactor used in training is defined by 5.14.

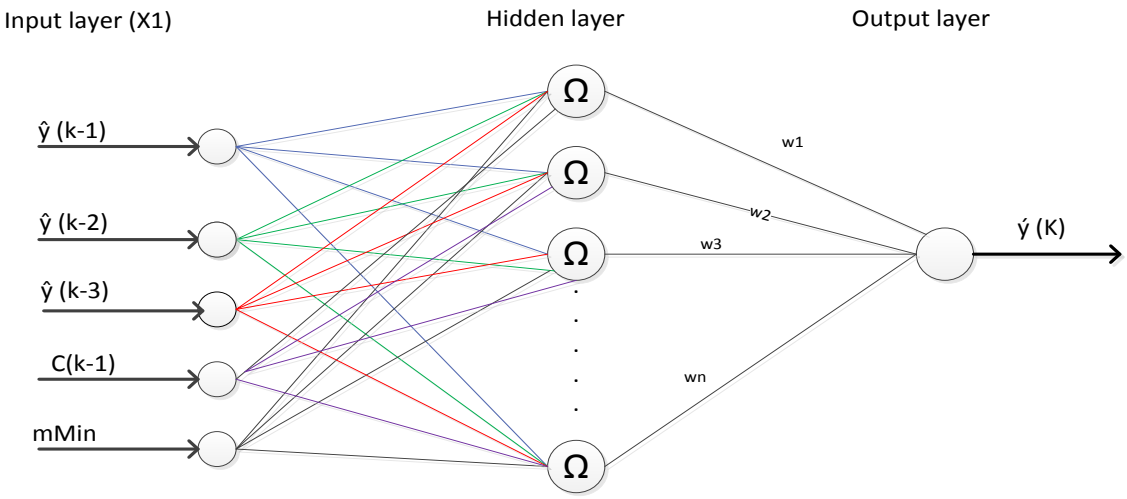


Figure 5.7 RBFNN structure with input variables

$$X_1 = [\hat{y}(k-1)\ \hat{y}(k-2)\ \hat{y}(k-3)\ C(k-1)\ \dot{m}_M^{in}(k-1)] \qquad (5.14)$$

Where, X_1 is the vector input for the RBF model, ŷ is the predicted reactor temperature, C is the valve position and $\dot{m}_M^{in}$ is the monomer feed rate.

After the data collection and model selection, the network is made to learn mapping of the collected input/output data (training). The training can be done by following the procedure in previous section 5.2.2.

Once the training is over, the network is tested by using some other signals which not learned by the network during training. If the prediction error of the model does not deviate much from the process response, then the developed model is accepted. Otherwise, the steps for system identification are repeated until the desired performance is reached by chosen different hidden nodes numbers.

RAS data obtained in section 5.2.3.1 used for training and a different RAS data used for testing (validating) as shown in Figure 5.8 & 5.9 for product A, and Figure 5.10 & 5.11 for product B.

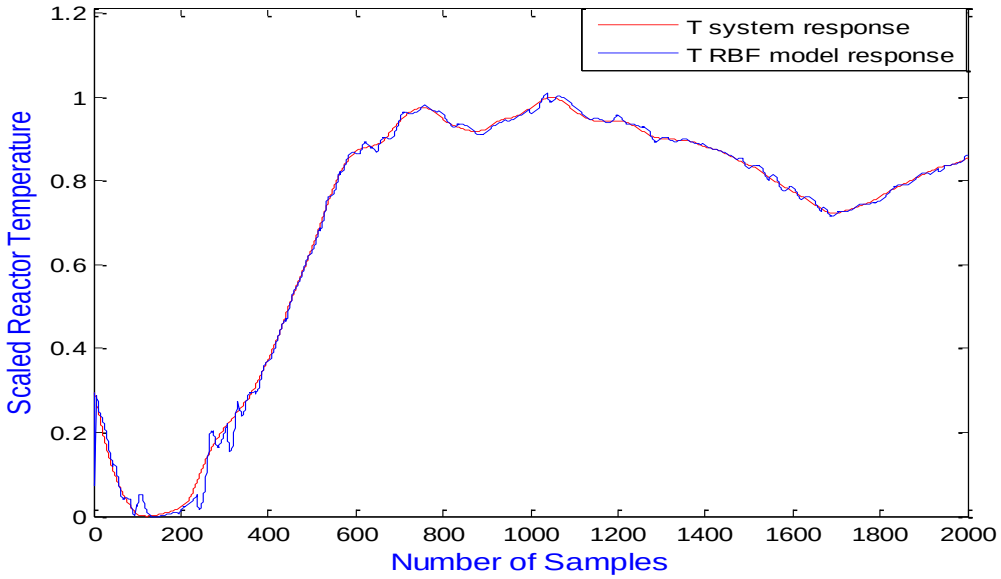


Figure 5.8 Modelling result (Train) of the RBF model for Polymer A MAE = 0.0080

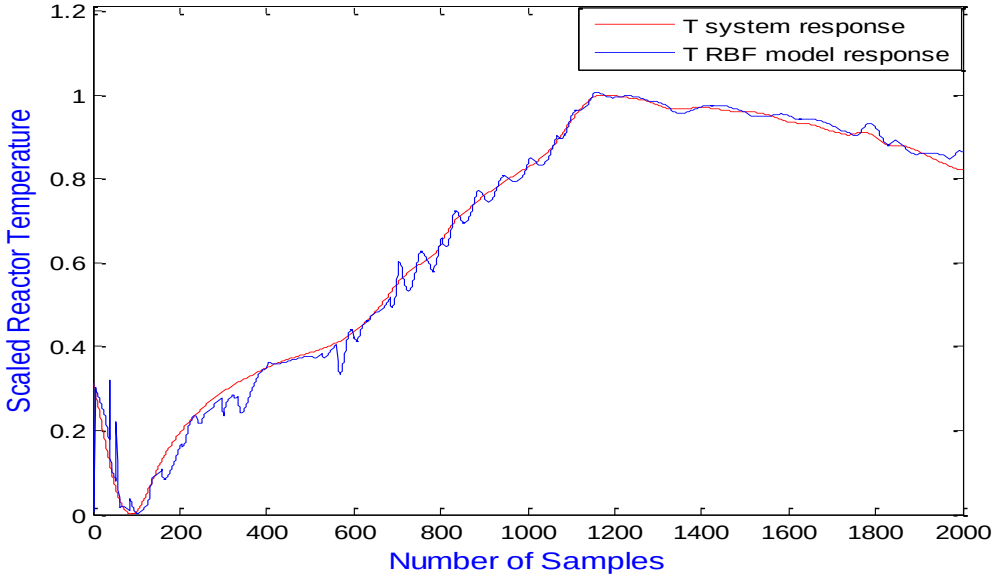


Figure 5.9 Modelling result (Test) of the RBF model for polymer A MAE= 0.0136

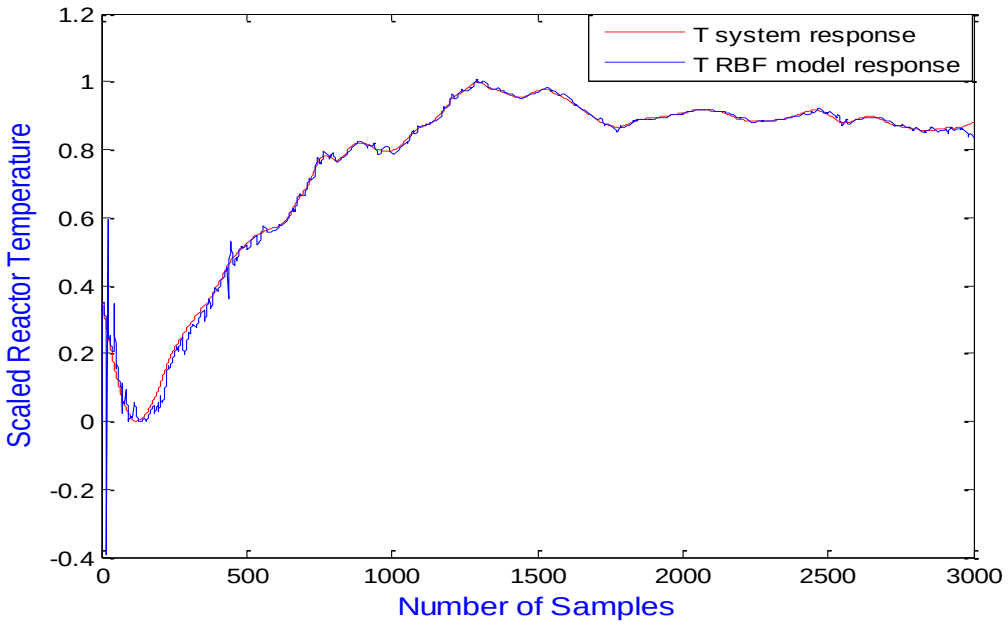


Figure 5.10 Modelling result (Train) of the RBF model for Polymer B MAE = 0.0084

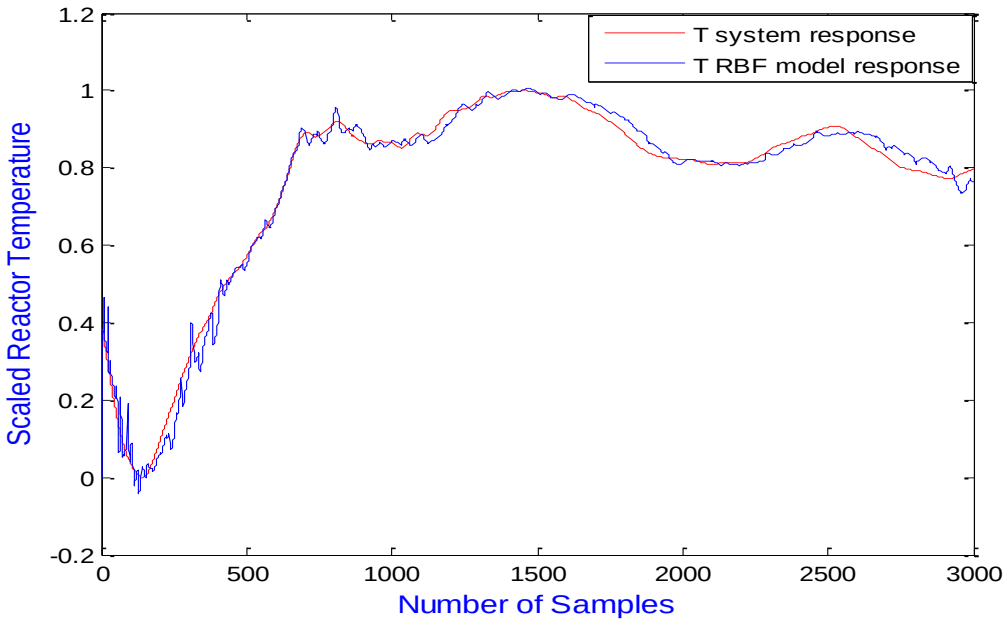


Figure 5.11 Modelling result (Test) of the RBF model for Polymer B MAE = 0.0191

It can be observed in Figures 5.8, 5.9, 5.10 & 5.11 that the RBF neural model predicts the process outputs quite precisely for 2000 samples for product A and 3000 samples for product B. It presents a good match between reactor output and RBFNN output, this result can be obtained with the following parameters.

$$\sigma = 11,\ \rho(0) = 1.0 \times 10^{-2} \times I_{nh \times nh},\ \mathrm{w}(0) = 1.0 \times 10^{-2} \times U_{nh \times 1}$$

Where σ is width of hidden layer, p and w is RLS parameters which used for weight updating (Yu et al., 2000, Yu and Yu, 2006). I is an identity matrix and U stands for a matrix whose components are ones.

It is noted that the modelling errors measured by the MAE decrease with the increase of the orders. Further investigations have shown that when the orders used were higher than those in 5.14, the modelling error stopped decreasing. This is because the improvement of the model accuracy due to higher model orders is compensated for by the degradation caused by the complexity of the model. The modelling results are consistent with those given in 5.14. Therefore, the RBFN (5:23:1) is the most successful one.

5.3 Multi-layer perceptron neural networks

Multilayer perceptron neural network (MLPNN) with error back propagation (BP) learning method is often used in ANNs. This type of neural network is known as a supervised network because it requires a desired output in order to learn. As the same as RBFNN the purpose of this type of network is to create a model that correctly maps the input to the output using historical data so that the model can then be used to produce the output when the desired output is unknown (Doherty, 1999).

The MLP and many other neural networks learn using an algorithm called back propagation. With back propagation, the input data is repeatedly presented to the neural network. With each presentation the output of the neural network is compared to the desired output and an error is computed. This error is then fed back (back-propagated) to the neural network and used to adjust the weights such that the error decreases with each iteration and the neural model gets closer and closer to producing the desired output. This process is known as "training".

5.3.1 MLPNN Structure

Feed forward neural networks are the basic type of neural networks and the multi-layer perceptron neural network (MLPNN) is used most. A MLP consists of an input layer, several hidden layers, and an output layer as shown in Figure 5.12.

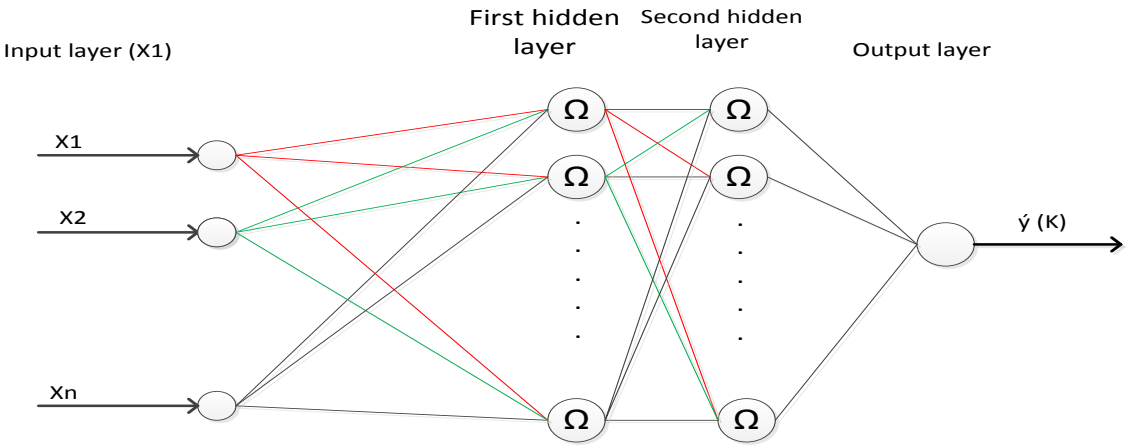


Figure 5.12 MLPNN structure with input variables

MLPNN structure can be adjusted according to complex degree of simulated system. The adjustment includes number of hidden layers and number of neurons of every hidden layer. For MLPNN, the number of hidden layers depends on complexity of nonlinear mapping. Although it has been proved that one layer network is enough to accurately approach any nonlinear function (TIAN et al., 2007, Baykan and Yilmaz, 2011).

5.3.2 Training algorithm

The training efficiency of MLPNN is high if the network layers are increased suitably. When number of hidden layer is assured, number of neurons of every hidden layer is important to the training of MLPNN. In the same way, it need less neurons if the mapping is sample, and large neurons versus complex mapping. However, too more hidden layer and neurons are not suitable because it may cause network over-fitting and affect network generalization and considering time consuming (TIAN et al., 2007).

In MLPNN, each neuron i in the hidden layer sums its input signals x_i after multiplying them by the strengths of the respective connection weights w_{ij} and output.

$$h_i = \sum_{i=0}^{N} w_{ij} x_i \tag{5.15}$$

And the output of the hidden layer neuron h_o is a nonlinear function of h_i,

$$h_o = f(h_i) \frac{1}{1+e^{-h_i}} \tag{5.16}$$

The activation function f (.) can be a simple threshold function, or a sigmoid, hyperbolic tangent, or radial basis function. But derivative function is preferred because in back propagated process, a derivation of the function is used.

The output of network is given by:

$$\hat{y}_{mlp} = w_{jk} h_o \tag{5.17}$$

Where w_{jk} is the weight connecting the output layer and the output of hidden neuron. The BP algorithm is designed to reduce the error between the actual output and the output of the network in gradient decent manner given by:

$$\mathrm{J} = \sum_{\mathrm{j}=1}^{\mathrm{g}} \mathrm{e}_{\mathrm{j}}^2 \tag{5.18}$$

The weights between the output layer and output of hidden neuron are updated by:

$$\mathrm{w}_{\mathrm{jk\_new}} = \mathrm{w}_{\mathrm{jk\_old}} + (\alpha \mathrm{h}_{\mathrm{o}} \mathrm{e}) \tag{5.19}$$

Where α is the learning rate.

The weights between the input of hidden neuron and the input are updated by:

$$\mathrm{w}_{\mathrm{ij\_new}} = \mathrm{w}_{\mathrm{ij\_old}} + \left(\beta \left(\frac{(1-\mathrm{h}_{\mathrm{o}})(1+\mathrm{h}_{\mathrm{o}})}{2}\right) \mathrm{x}_{\mathrm{i}}\right) \tag{5.20}$$

Where β is the learning rate.

5.3.3 Chylla-Haase reactor modelling using MLP

5.3.3.1 Data collection and scaling

To conduct a simulation using MLP algorithm, the same procedure and data mentioned in section 5.2.3 was used. As well as the same equation as in 5.10, 5.11 & 5.12 is

applied for scaling and evaluated performance. Based on the error obtain, the training process evaluated using equation:

$$e = y - \hat{y} \tag{5.21}$$

Where y and $\hat{y}$ is the Chylla-Haase output and the MLP model prediction respectively.

5.3.3.2 Model structure selection

The MLP model structure will be as the same as mentioned in section 5.2.3.2 equation (5.14), where, three variables were chosen to be the MLPNN inputs: predicted reactor temperature $\hat{y}$, valve position C and monomer feed rate $\dot{m}_M^{in}$. The structure of the MLP neural networks model of the Chylla-Haase reactor is 5:6:1 (inputs: hidden nodes: outputs).

In this work, three layers of MLP network, input layer, hidden layer and output layer are used for training, the same structure used in the RBFNN in previous section used in MLPNN.

The training process, 2000 samples for product A and 3000 samples for product B to train the MLPNN. Simulation results below can be obtained with the following parameters.

$\beta = 0.5, \alpha = 0.1$

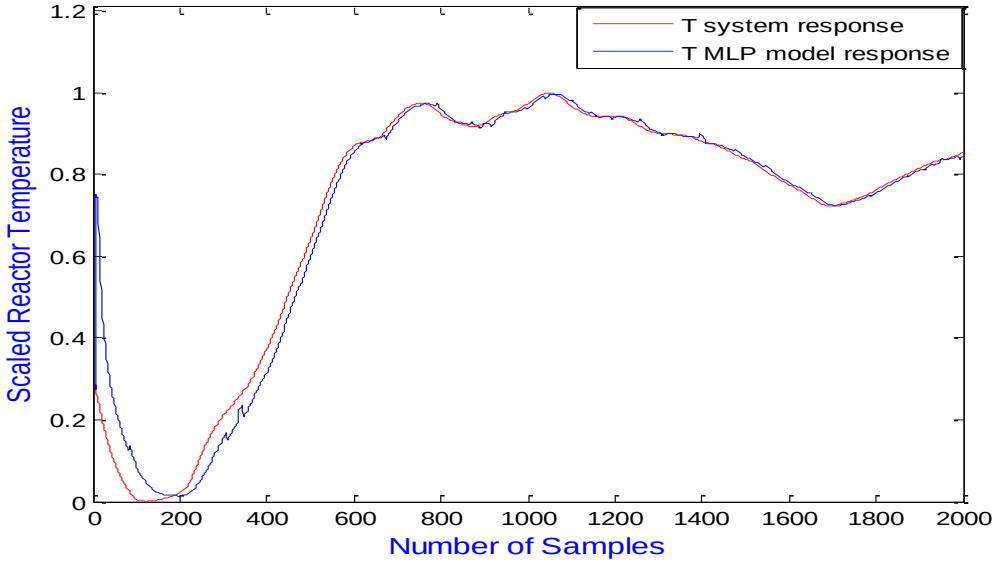


Figure 5.13 Modelling result (Train) of the MLP model for Polymer A MAE = 0.0123

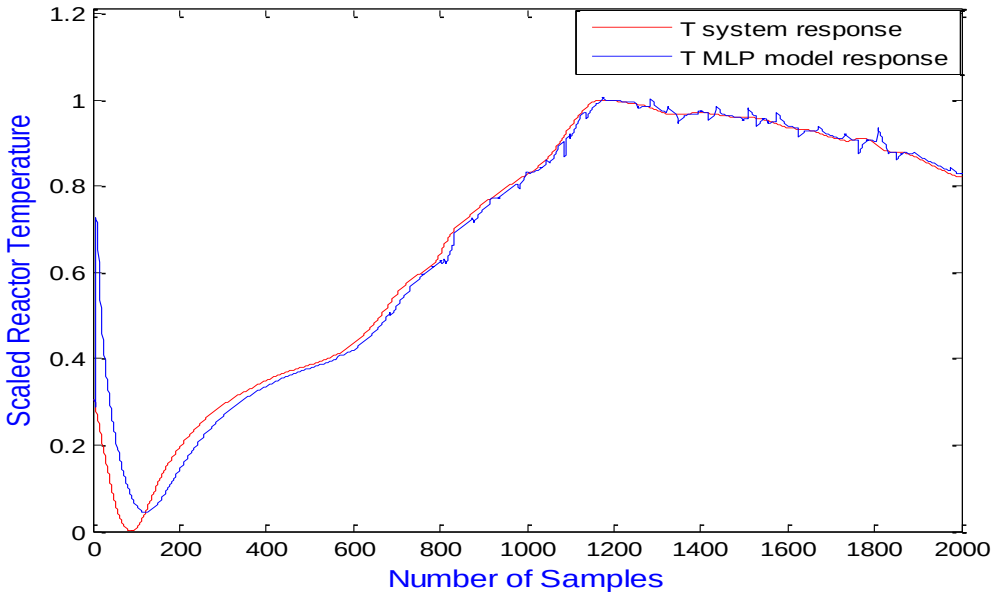


Figure 5.14 Modelling result (Test) of the MLP model for polymer A MAE= 0.0491

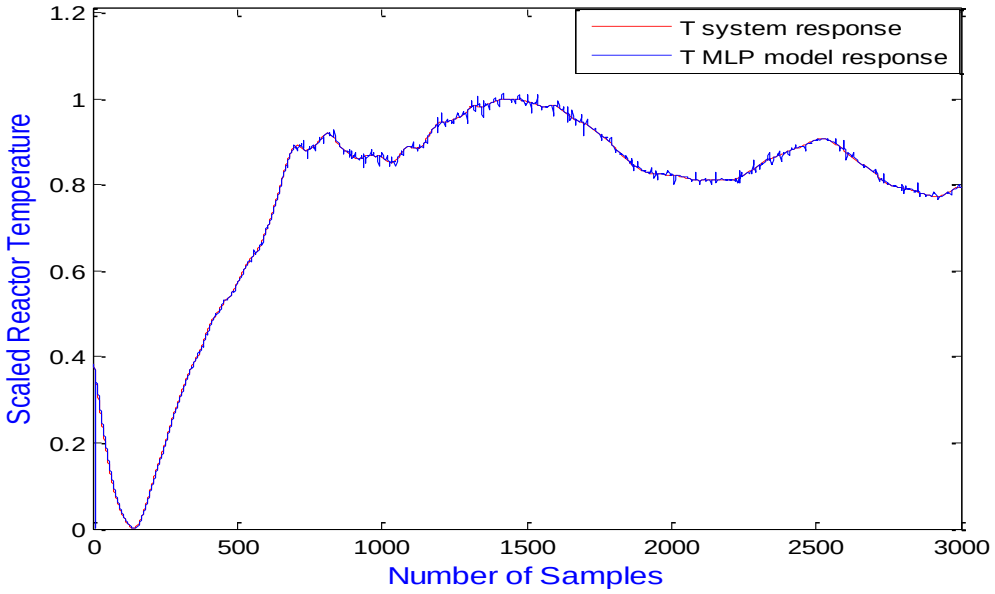


Figure 5.15 Modelling result (Train) of the MLP model for Polymer B MAE = 0.0033

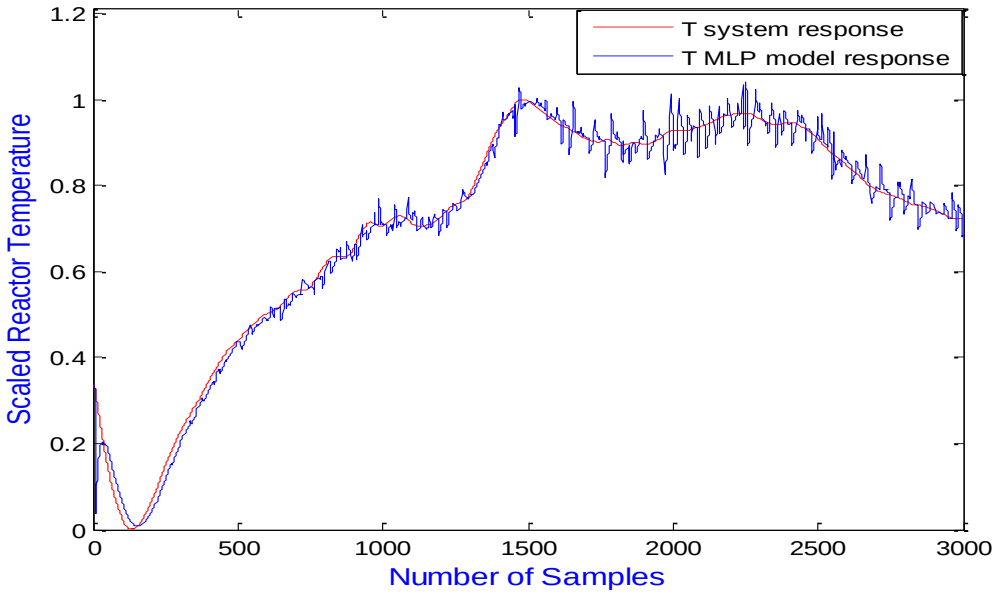


Figure 5.16 Modelling result (Test) of the MLP model for Polymer B MAE = 0.0190

Figure 5.13 & 5.14 present the reactor temperature response and MLPNN model for product A. also Figures 5.15 & 5.16 represent the MLPNN for product B. The modelling errors of the MLPNN models were measured by mean absolute error (MAE) as showed in the figures above. The error is accepted and it can be used as a simulation model.

5.4 Summary

RBF and MLP neural network described in this chapter to be used to model the nonlinear polymerization process. The simulation studies done on the neural network based model developed for the process, proves it to be a promising data based modelling tool The simulation shows that the trained neural network was capable of capturing dynamics with a very small performance tolerance indicating its high prediction accuracy.

The minimum MAE obtained during the training process of the RBF network was less than that in the MLP network. Thus, the RBF network produced a more fitted output to the cross validation data set than the MLP network. Both ANNs showed a strong robustness in prediction of high non-linear process.

Comparing the two adaptive neural network models, there is not distinct difference between them. The modelling error of the RBF network is a little smaller than that of the MLP network. In order to further test their suitability for the proposed MPC scheme for Chylla-Haase reactor control, two MPC schemes are realised by using the RBF model and the MLP model in chapter 7.

Using past system input and output signals as an input to NN in order to predict the future outputs at each sample time and by feeding back the model outputs to the input nodes of the network, dynamical models are generated as it will be explained in details in the next section with the design of the controller based on these models.

Chapter 6 RBF NN-Based Inverse Model Control

Control of polymerization reactors is a difficult and challenging process in control engineering, due to its high non-linearity and complicated reaction mechanisms associated with the large numbers of interactive reactions, also lack of online measurement of important variables and different reaction dynamics being involved. Thus the traditional feedback control performance is not satisfactory. Therefore, by adding feed-forward control scheme, the significant improvements are expected to achieve. Feed-forward and feedback (FB) control is developed in this work for Chylla-Haase polymerization reactor.

Firstly, we have tried to combine NN modelling techniques, in particular of RBF, with classical PID control strategies so that we can augment the capabilities of this technique and improve the overall control for the Chylla-Haase reactor. Secondly, we have attempted to preserve, as much as possible from a practical perspective, the PID structure, for easy design and implementation. Thirdly, we have developed a strategy for controller tuning to maintain the temperature with specific tolerance against monomer feed inlet effect and disturbance.

In this chapter, the effectiveness and robustness of RBFNNs in identification and control of a reactor temperature is investigated. The RBFNN is used to estimate the valve position for the reactor by considering the temperature and one of the disturbances as an input and the valve position will be the output of RBF model (Inverse model). PID controller used as a feedback to compensate the neural RBFNN inverse model output.

6.1 Chylla-Haase reactor modelling using RBFNN inverse model

6.1.1 Inverse Model

The radial basis function neural network has an ability to model any nonlinear function. However, this kind of neural network need many nodes to achieve the

required approximating properties (Pottmann and Seborg, 1997). The first step in the reactor modelling is the generation of a suitable training data set. To conduct a simulation using RBFNN algorithm, the same procedure and data mentioned in section 5.2 was used. The accuracy of the neural network modelling performance will be influenced by the training data. For RBF neural network training, the K-means algorithm is used to choose the centers, P-nearest neighbor algorithm decides the widths and the recursive training algorithm calculates the weights for the output layer (Yu and Zhai, 2008) as referred in previous chapter .

In this section, an RBFNN is trained to model the inverse dynamics of Chylla-Haase reactor to generate the control variable C which is the manipulated variable in the next sample time. the relevant inputs affecting C and $\dot{m}_M^{in}$ can be determined from measurable states.

The dynamics between the actual reactor output (Temperature *T*) and the control variable (valve position C), can be represented by (6.1) with different order and time delay.

$$T(k) = f[T(k-1), T(k-2), C(k-1), C(k-2), \dot{m}_M^{in}(k-1)] \qquad (6.1)$$

Where, *f (.),* is the nonlinear function of the network and *T(k)* is reactor temperature, *C* is the manipulated variable and $\dot{m}_M^{in}$ is the monomer feed. Equation (6.1) describes the relevant inputs and their time dependencies on T.

$C\ (k)$ can be predicted using equation (6.1) as follows:

$$C(k-1) = g[C(k-2), T(k), T(k-1), T(k-2), \dot{m}_M^{in}(k-1)] \qquad (6.2)$$

Where g (.) is the inverse nonlinear function of (6.1).

If C is predicted at current sample instant (k) the equation is:

$$C(k) = g[C(k-1), T(k+1), T(k), T(k-1), \dot{m}_M^{in}(k)] \qquad (6.3)$$

$T(k+1)\ and\ T(k)$ represent the future and present temperature, which are not available yet at present sample period k. So, they are replaced by reference

signal $r(k+1)$ and $r(k)$, this is reasonable as the temperature $T(k)$ is to be controlled to follow the reference signal $r(k)$.

$$C(k) = g[C(k-1), r(k+1), r(k), T(k-1), \dot{m}_M^{in}(k)] \tag{6.4}$$

So the RBF network input vector presented in equation (6.5).

$$X_1 = g[C(k-1), r(k+1), r(k), T(k-1), \dot{m}_M^{in}(k)] \tag{6.5}$$

Where, X_1 is the vector input for the inverse RBF model, $C(k)$ is the valve position for the reactor, $T(k)$ is the reactor temperature, $\dot{m}_M^{in}(k)$ monomer feed rate and $r(k)$ is the reference signal. Generally, $\hat{C}$, the RBF neural output (estimated valve position), is a weighted sum of the hidden node outputs and this has been explained in details in previous chapter, which is presented by equation 6.6.

$$\hat{C} = \sum_{j=1}^{nh} \emptyset_j(t) * W_{j,i} \qquad i = 1, \dots\dots, q \tag{6.6}$$

Where, $\emptyset_j$ are defined by (5.4) in previous chapter, W are the output layer weights and q is the number of outputs. The method used in this work to calculate the Euclidean distance between the centre and the network input vector X_1 is the same as the method used in previous section, which is Gaussian basis function.

The RBFNN block diagram is illustrated in Figure 6.1. The RBFNN model has five inputs as given in equation 6.5. Different orders of network inputs and different numbers of hidden layer nodes were used in training experiments and a network structure with 15 hidden nodes was chosen, which gives the minimum MAE prediction (0.453) for training and 0.501 for test. The centres c and the widths σ in the hidden layer nodes of the RBF network were determined using K-means algorithm and p-nearest neighbours respectively and σ here set to be 13. The Recursive Least Squares algorithm (see section 5.2.2) is used for training the network weights with its parameters set μ=0.998, $W(0) = 1.0 \times 10^{-8} U_{nh\times 1}$ and $P(0)=1.0\times 10^6 I_{nh\times nh}$, where, μ is the

forgetting factor, *I* is an identity matrix and *U* stands for a matrix whose components are ones. All the data are scaled using equation 5.10 & 5.11.

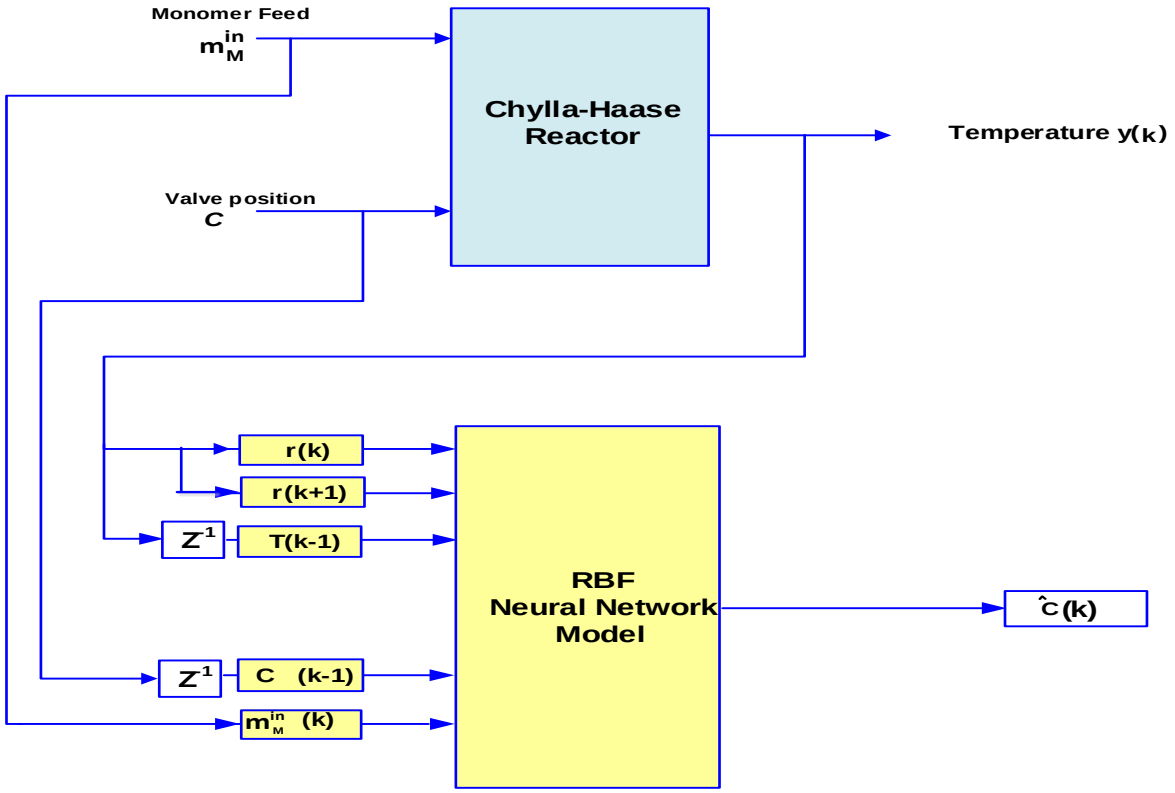


Figure 6.1 The RBFNN block diagram with input variables

Training data set with 3000 samples was used to train the RBFNN model for polymer B. Then, the test set with different 3000 samples was applied to the trained model and the model output prediction results are displayed in Figure 6.2 & 6.3. It can be seen from the simulation results, the good match between the actual valve position and the RBFNN output during the model validation phase.

In order to get the non-scaled valve position the following equation should be apply:

$$C = C_{min} + C(k) * (C_{max} - C_{min}) \tag{6.7}$$

Where, C_{min} is the minimum valve position, C_{max} is the maximum value of the valve. At the sample time *k*, the neural model for *C* makes one-step ahead prediction on the valve position *C* using the measured states including reactor temperature *T(k-1)*, monomer feed and valve position C itself at *t = k-1*.

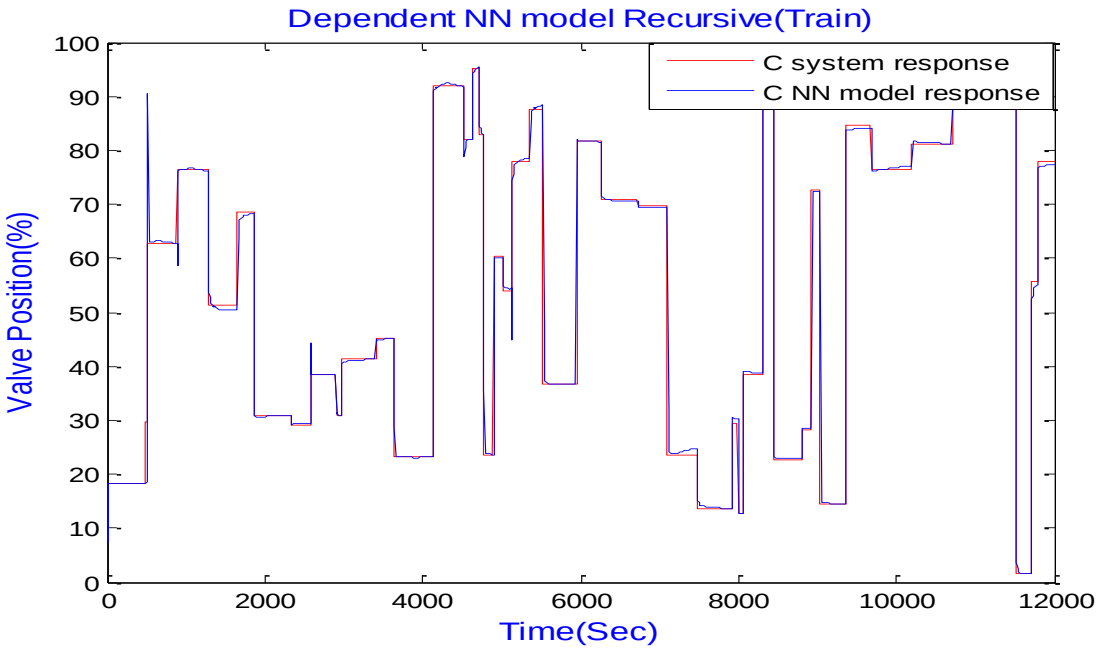


Figure 6.2 Training Data for RBFNN Model MAE (Mean Absolute Errors=0.453)

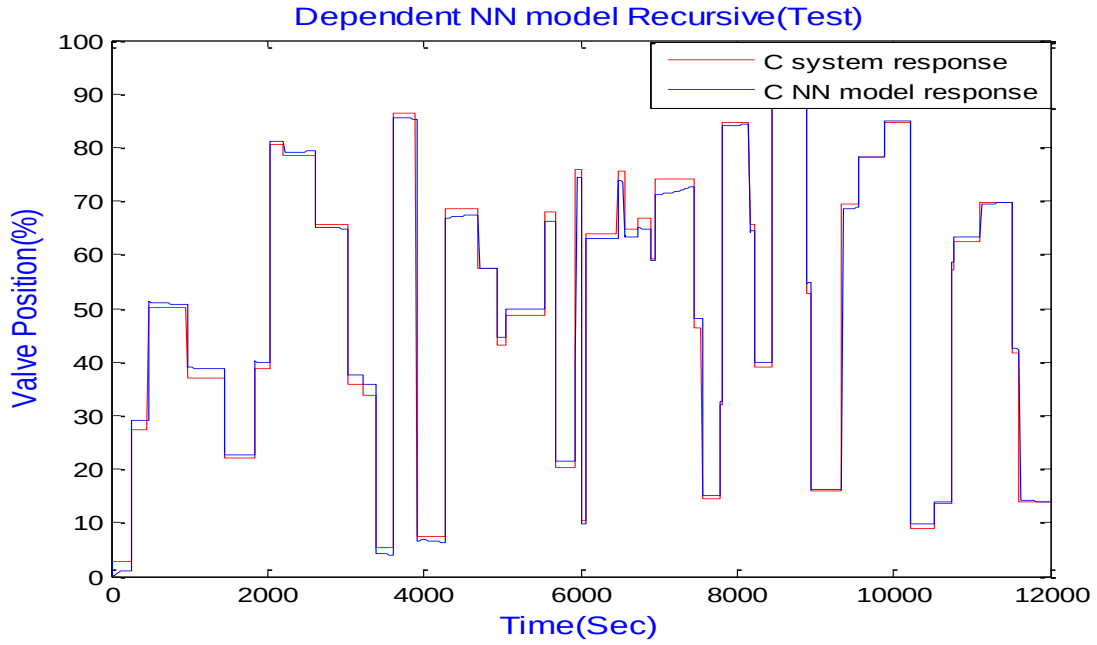


Figure 6.3 Validation Data for RBFNN Model MAE (Mean Absolute Errors=0.501)

The *C* in Figure 6.2 & 6.3 is the scaled back value. The output of the neural network is nearly equal to the actual data. This implies that the trained inverse model is quite accurate.

6.2 Feed-forward and feedback control constitution

In the previous sections RBFNN inverse model to estimate valve position was presented and a satisfactory result was obtained, which approximately equal to the

generated data for the valve position. This inverse model is then used in the Chylla-Haase reactor feed-forward path in order to keep the reactor temperature output at the request value. The RBFNN based adaptive FF with FB control system structure in our implementation is shown in Figure 6.4. With the trained RBFNN inverse model, a set of satisfactory control results were achieved. In the implementation, all the recursive Least Square parameters *w(0)*,μ and *P(0)* have been saved, then this model is used in the feed-forward path to predict the $\hat{C}(k)$,

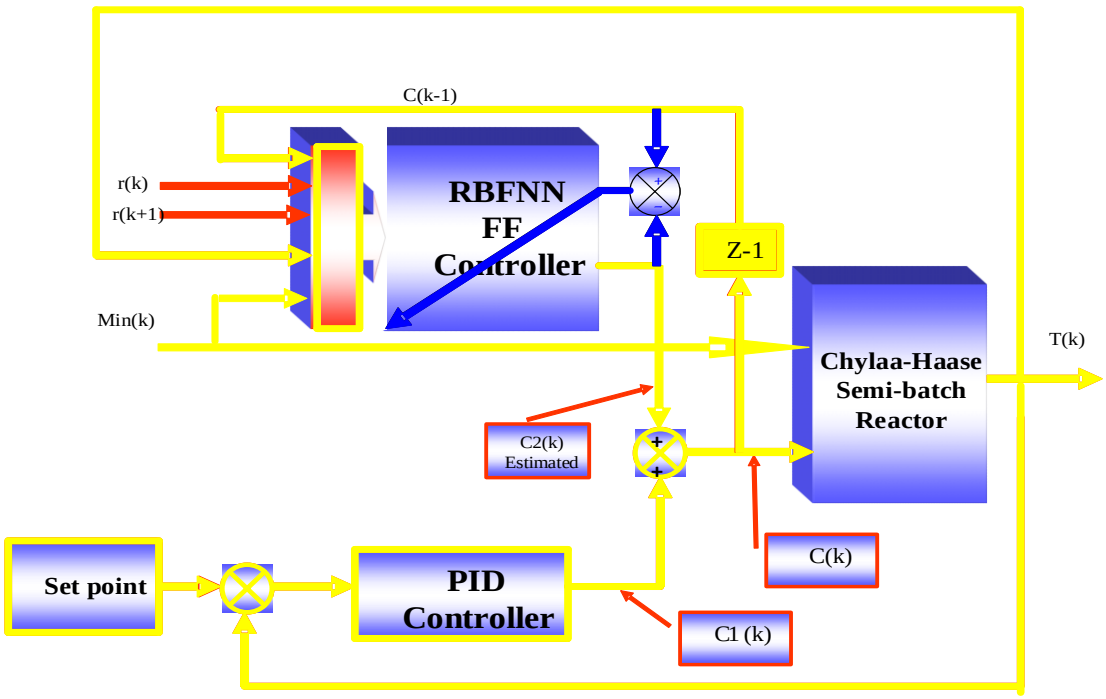


Figure 6.4 Adaptive FF and FB control system

In the control block diagram, measured *T(k-1),r(k+1),r(k), C(k-1),* and $\dot{m}_M^{in}(k)$, form an input vector for the inverse model of the reactor temperature. The PID controller is used as the feedback controller. In this case the activating valve position is the sum of the two controller output variables, one is from the RBF based feed-forward neural network controller, the other from the feedback PID controller as shown in Figure 6.4. The algorithm of the PID Controller is as follows (Yu and Zhai, 2008, Ming-guang et al., 2005).

$$C(k) = C(k-1) + k_p\left[\left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right)e(k) - \left(1 + \frac{2T_d}{T}\right)e(k-1) + \frac{T_d}{T}e(k-2)\right] \quad (6.9)$$

After fine tuning, the PID controller that is use here with RBF based neural network controller for Chylla-Haase reactor is:

$$C(k) = C(k-1) + [k_1 * e(k) + k_2 * e(k-1) + k_3 * e(k-2)] \quad (6.10)$$

The on-line adaptation is implemented as follows. Firstly, the measurements are fed into the RBFNN inverse model. Then the output of the model is compared to Chylla-Haase reactor output to measure the error. After this, the target will be modified according to whether the error is produced. If the error is detected the on-line training target value will be changed to the target value corresponding to the occurred error. Then the measurements and the modified target are used to update the RBFNN inverse model parameters.

In the adaptation, the centres and the widths remain fixed, as they have been chosen distributed in the whole operating space, while the weights are updated to minimize the error caused by any time-varying dynamics and model uncertainty. The weights are adapted using the recursive Least Squares (RLS) algorithm in (5.9).

6.3 Simulation results and Control Performance

The simulation with the new control strategy is done for the semi-batch polymerization reactor. The performance of the RBFNN in tracking control was demonstrated with different disturbance effects, impurity factor and fouling factor with two scenarios, first and fifth batch of the process.

The investigations presented in this chapter are restricted to product B only and the following two scenario variables are considered presented in table 6.1, the purity factor *i* varies from 0.8 to 1.2 so that the fluctuations in the reaction rate caused by impurities in the raw-materials can be described. It changes randomly from batch to batch, but it is constant during one batch. The fouling factor *1/hf* varies from zero to 0.704 m^2K/KW to simulate the decrease in *U* resulting from the formation of a polymer

film on the reactor wall during successive batches.

TABLE 6.1
SCENARIO CONSIDERED FOR CONTROL ANALYSIS

Scenario	$i[-]$	$1/hf[m^2K/KW]$
1	0.8	0.0
2	1.2	0.704

In the RBF tuning of PID control, the three parameters of PID control were given as follows: K_p is 35, K_i is 0.008, K_d is 150. The Sampling time as 4s, the network structure is 5-15-1. Significant improvement can be seen for the reactor temperature in Figure 6.5, using RBFNN inverse model plus PID control comparing with conventional cascade controller in chapter 4. Figure 6.6 & 6.7 demonstrate valve position and monomer feed rate respectively.

The adaptive feed-forward control achieves excellent results as shown in Figure 6.5, especially at the critical time points when the monomer feed stops and rises arbitrarily, this is mainly due to on-line adaptation of the RBFNN, which enables the fast detection of the sharp flank of the reaction heat (Q_{rea}) at the end of the monomer feed. Also in the fifth batch, which has big disturbance effect on the temperature, the RBFNN plus PID still can maintain the temperature within the tolerance range which is ($\pm$0.6K) from the set point. However, from Figure 6.6, the valve position (control variable) is quite oscillatory and sometimes exceeds its upper (100%) and lower (0%) bounds. After considering these effects, the overall performance is not very good and is unacceptable in practice, so more advance control needs to be applied.

The tracking performance of the control system evaluated by the mean absolute error that has been defined in equation 5.12. The tracking MAE equals to 0.0045 in this simulation.

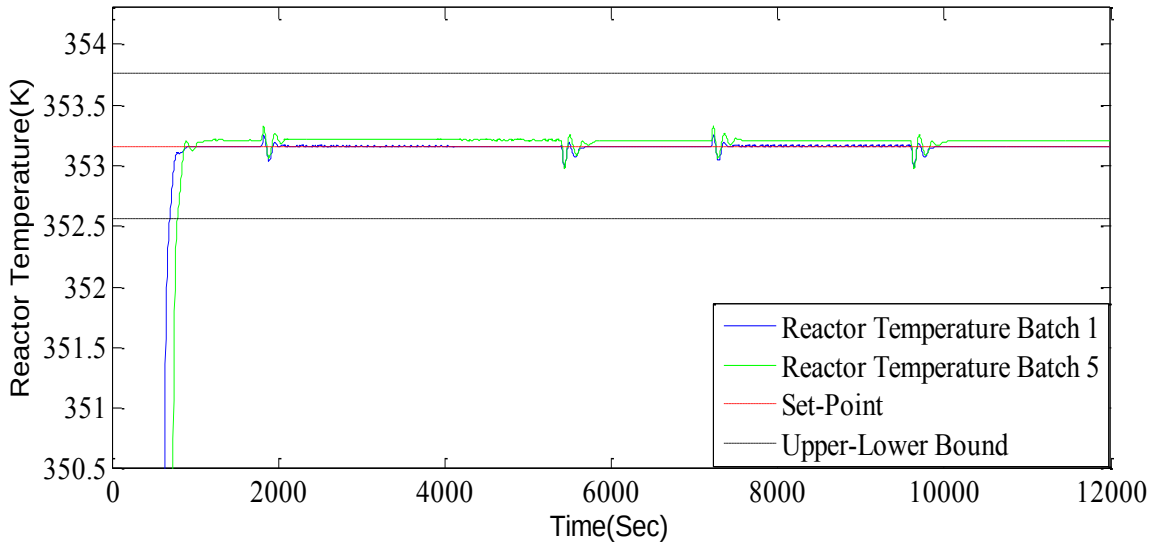


Figure 6.5 Reactor temperature with adaptive RBFNN Inverse Model for first and fifth batch

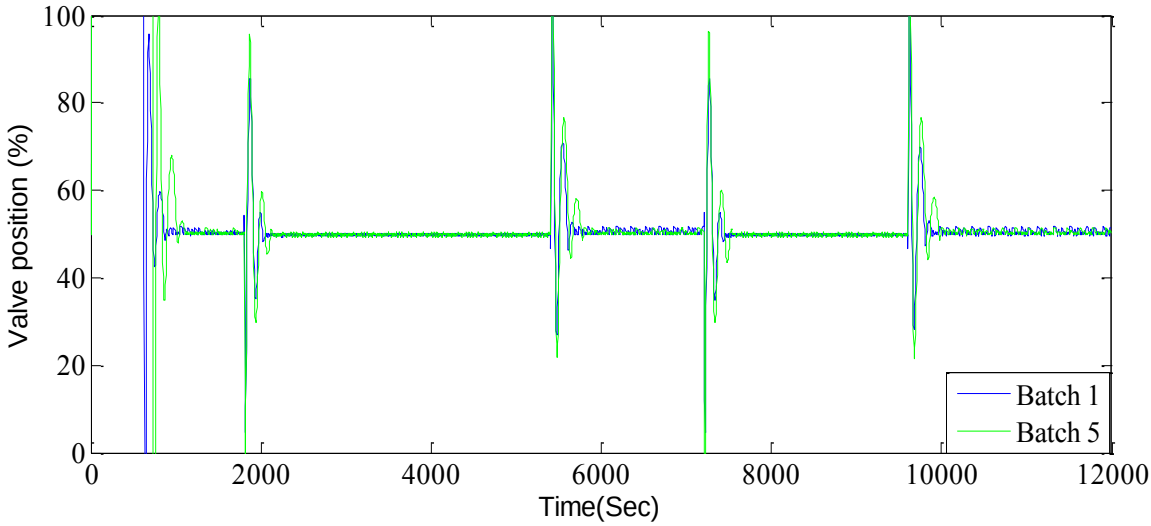


Figure 6.6 Valve position with adaptive RBFNN Inverse Model for first and fifth batch

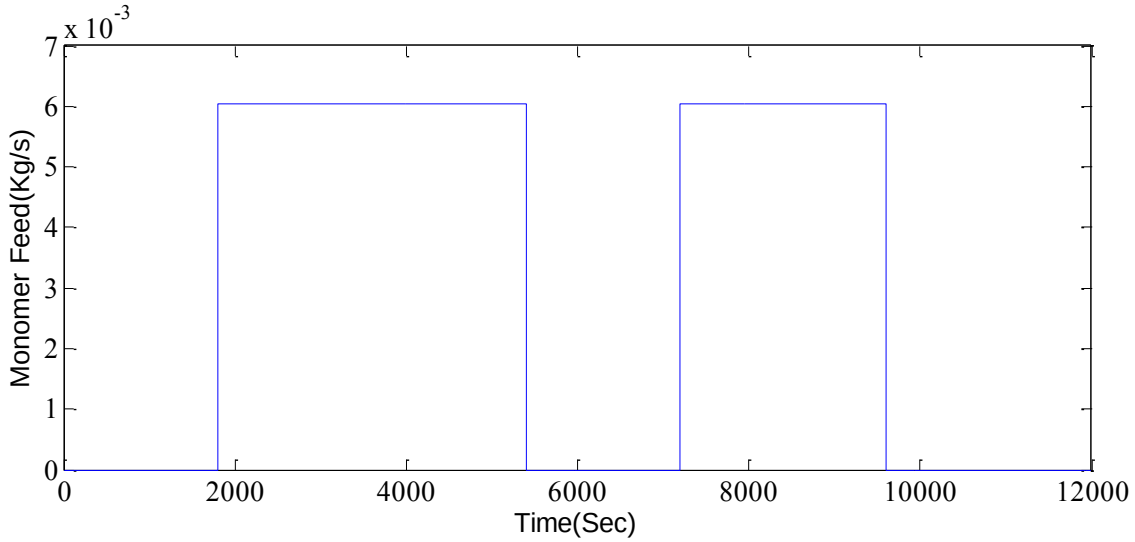


Figure 6.7 Monomer feed rate

6.4 Summary

It is anticipated that PID controllers will continue to play a key role in process control for a long time to come. Therefore, we attempted, in this work to develop control strategies to maintain the PID structure while enhancing its performance capabilities. This is done by combining the conventional control techniques with the evolving RBFNN methodologies.

The proposed control strategy has advantages of both self-learning capability of neural network and simplicity of PID controller. During the practice, adaptive controller has the superiority such as strong robustness and simple structure. The effectiveness of the proposed method is obvious from simulation result and by evaluating the control performance using the mean absolute error. It can been seen the proposed control method is more capable of compensating the disturbance effects than the previous presented control methods.

Chapter 7 Neural network-based model predictive control

7.1 Introduction

MPC is by far the most commonly applied advanced control technique in the chemical engineering processes. The main reason for such a popularity is the ability to handle constraints and formulation of multivariable systems. The main idea behind the MPC is an online solution of optimization problem at every time instant required to compute optimal control input over a fixed number of future time instant (Seborg et al., 2004, Wang, 2009).

Previous research have shown that the most of the nonlinear MPC strategies are based on physical process model, which are often very inaccurate. However if imperial models are identified from input/output system data, MPC techniques can be applied to a wide range of nonlinear processes.

This chapter presents a novel MPC control scheme based on neural networks (RBFNN and MLPNN) for semi-batch polymerization reactor. RBFNN and MLPNN is trained using Recursive Least squares method online. A nonlinear optimisation algorithm sequential quadratic programming is used to generate the optimal control signals of Chylla-Haase reactor.

The RBF and MLP networks are usually employed in the same kind of applications (nonlinear mapping approximation and pattern recognition), however their internal calculation structures are different (Jung-Wook et al., 2002).

The main objective of the predictive control strategy using neural network is to predict the future output of the plant and to minimize the cost function based on the error between the predicted output of the process and the reference trajectory.

Neural network modelling results in chapter 5 showed that the trained neural network was capable of capturing dynamics with a very small performance tolerance indicating

its high prediction accuracy. Moreover, from literature, its known that combining RBFNN and MLPNN with MPC controller shows that this scheme were able to force process output to follow the target within its tolerance against disturbances. This work will allow comparing the performances of both approaches, the MLP based predictive control and the RBF-based predictive control.

7.2 MPC system Configuration

The NN-MPC structure for the reactor is shown in Figure 7.1. The obtained RBF and MLP neural networks model in chapter 5 is used to predict the reactor output for N_2 steps ahead. The nonlinear optimizer minimizes the errors between the set-point and the predicted output, as well as the increment of the optimized control variable by using the cost function (7.1).

$$J = \sum_{i=t+N_1}^{t+N_2}[y_{set}(i) - \hat{y}(i)]^2 + \lambda \sum_{i=t}^{t+N_u}[u(i) - u(i-1)]^2 \quad (7.1)$$

$$u_{min} \leq u \leq u_{max} \quad (7.2)$$

This minimization is subject to constraint (7.2) valve position. Here, N_1, N_2 defines the prediction horizon, λ is a control weighting factor, N_u control horizon, y_{set} is the set-point to introduce a feedback in order to compensate the system steady-state error and against disturbance effects. The remaining main problem of MPC is to solve the nonlinear optimization problem in each sample period, calculate a series of optimal $u(t), u(t+1), \ldots, u(t+N_u\text{-}1)$ from which the neural network model generates output to minimize *J* (Yu et al., 1999, Robert Haber, 2011) . The optimization of control variable *u(t)* is subjected to the constraint that is giving in (7.2) in this case.

The filtered model prediction error in Figure 7.1 is also used to compensate the model outputs in the MPC scheme to correct the future predictions. This is to modify the model outputs in (5.1) by adding the corresponding filtered error to the right-hand side of the equations.

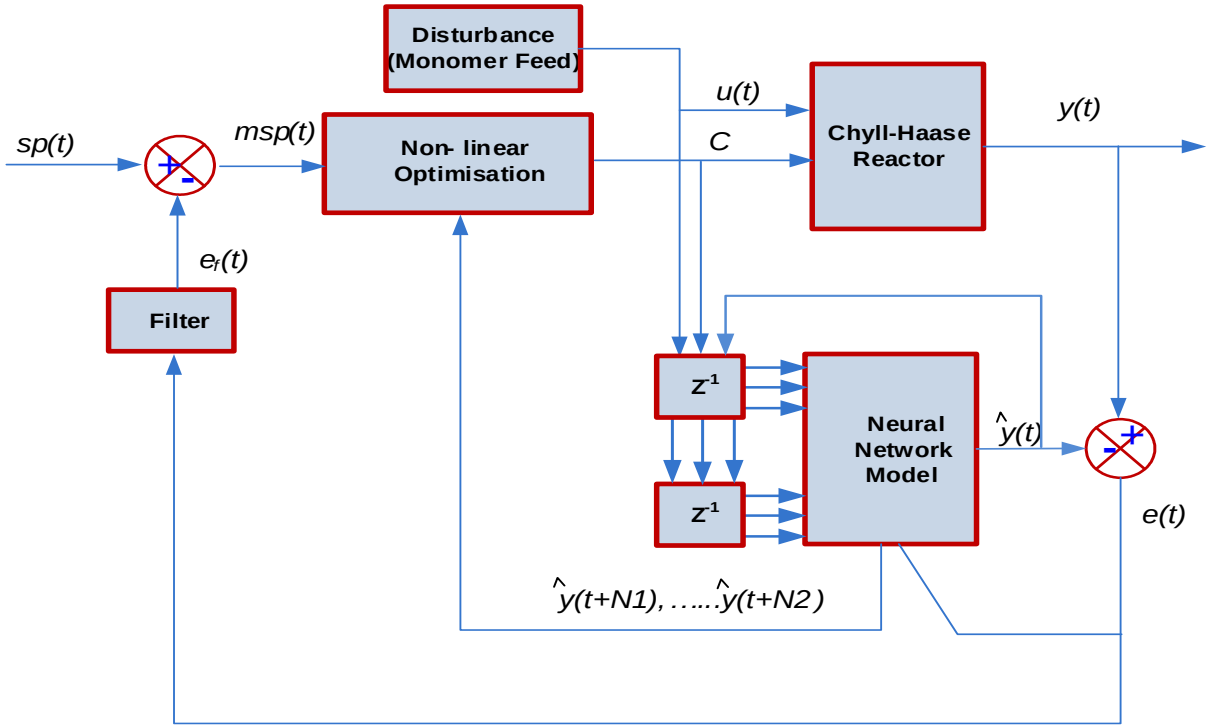


Figure 7.1 The scheme of neural network based model predictive control

The minimization of (7.1) yields a set of future $u(k)$ and only the first control move is implemented at every sampling instant. Then the whole procedure will be repeated in the next sampling period.

7.3 Chylla-Haase reactor modelling using RBFNN

Neural Network model for Chylla-Haase polymerization reactor (RBF and MLP) is given in Figure 7.2. RBFNN and MLPNN is developed with the data collected from the simulated polymerization reactor in chapter 5. Third order structure with valve position and monomer feed used here, more details in section 5.2.3.2. Since NMPC can be applied to nonlinear systems to fined optimal control, it will be used here to achieve robust control against disturbance.

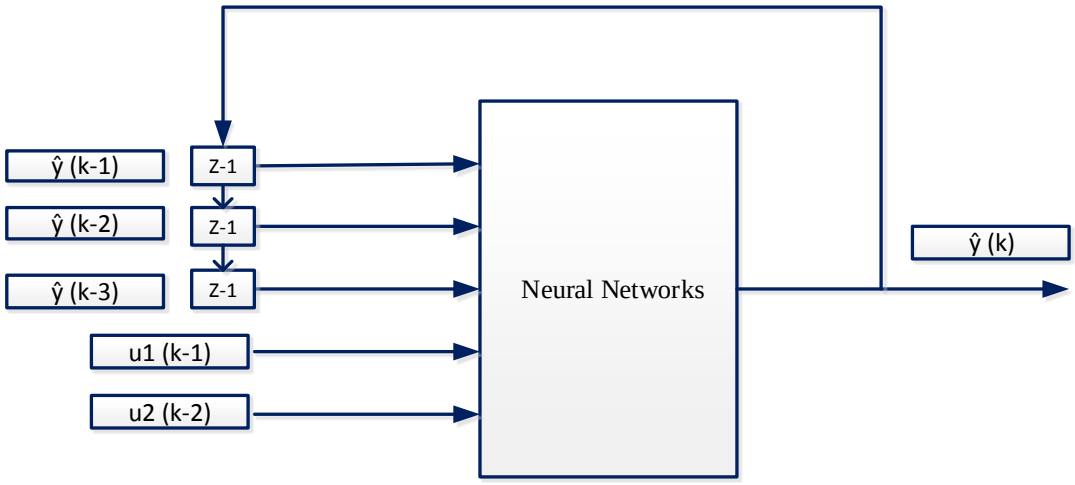


Figure 7.2 structure of RBF and MLP neural with input variables

7.3.1 Data Collection and Scaling

Firstly, to model the reactor using neural network, a set of suitable input/output data sufficiently covering the region in which the system is going to be controlled must be generated. Because the accuracy of the neural network modelling performance depends on this data and this data randomly selected so in practice it will cover the region of the control interests. This stage as the same as explained steps in chapter 5 section 5.2.3.1

Random amplitude signal (RAS) is used to generated the data set (0~2000 samples) of RAS generated for valve position (manipulated variable) and monomer feed rate for product A as shown in Figures 5.2 & 5.3 respectively in chapter 5. Moreover, (0~3000 samples) for product B as shown in Figures 5.4 & 5.5.

The valve position range Figure 5.2 in chapter 5 in the process is from (0% ~100%) and the monomer feed is (0~0.0075) kg/s with fixed time for product A and (0~0.00641) kg/s for product B (longer batch). The sample time set to be 4 seconds.

7.3.2 RBF training algorithm

As the same in chapter 5, three variables were chosen to be the network inputs from the reactor simulation, valve C, Monomer feed $\dot{m}_M^{in}$, and reactor temperature T.

Third order structure with 23 hidden nodes has been chosen because they give a minimum prediction error, The structure was selected based on testing different networks that vary in terms of structure and simulation parameters, the RBFNN input will be as a vector X_1, where X_1 is given by the following equation:

$$X_1 = [\hat{y}(k-1)\ \hat{y}(k-2)\ \hat{y}(k-3)\ C(k-1)\ \dot{m}_M^{in}(k-1)] \tag{7.3}$$

Where, X_1 is the vector input for the RBF model, ŷ is the predicted reactor temperature, C is the valve position and m_M^{in}, is the monomer feed rate (disturbance). Generally, $\hat{y}$ is a weighted sum of the hidden node outputs.

Once the training is over the network is validated by passing some other signals which not learned by the network during training. If the prediction error of the model does not deviate much from the process response, then the developed model is accepted else, the steps for system identification are repeated until the desired performance is reached. The RAS data obtained in chapter 5 used for training and a different RAS data used for testing (validating). Figures 5.8 & 5.9 for product A, and Figures 5.10 & 5.11 for product B presents a good match between reactor output and RBFNN output, mentioned in chapter 5.

7.3.3. MLP training algorithm

As the same in chapter 5, three variables were chosen to be the network inputs from the reactor simulation, valve C, Monomer feed $\dot{m}_M^{in}$, and reactor temperature T.

The multi-step ahead prediction based on the identification structure is given by equation 7.3.

2 hidden layers are considered here, learning algorithm is based on Back-propagation method. Training of MLPNN is done with an objective of mean squared modelling

error to determine a set of optimal weights, so that the network will produce predicted output near to true outputs

7.4 RBF-based MPC for Polymer A

The neural network obtained in chapter 5 is used in this section to generate the Chylla-Haase model for the MPC scheme. Figure 7.3 illustrate the modelling diagram. The network model has 3 input variables, as mentioned before, the valve position C, monomer feed rate m_M^{in} and the reactor temperature output y. The denote y represents the target value of the reactor temperature and $\hat{y}$ is the output of the neural network model. The input of the RBFNN was determined experimentally and a suitable was equation 7.4. in case of using non-adaptive RBFNN, all the parameters (w (0), μ and P(0)) of the RBFNN is saved in order to use them in non-adaptive MPC but due to system parameters variation and uncertainty observed in process operation an adaptive MPC is needed that adapts itself to such changing conditions.

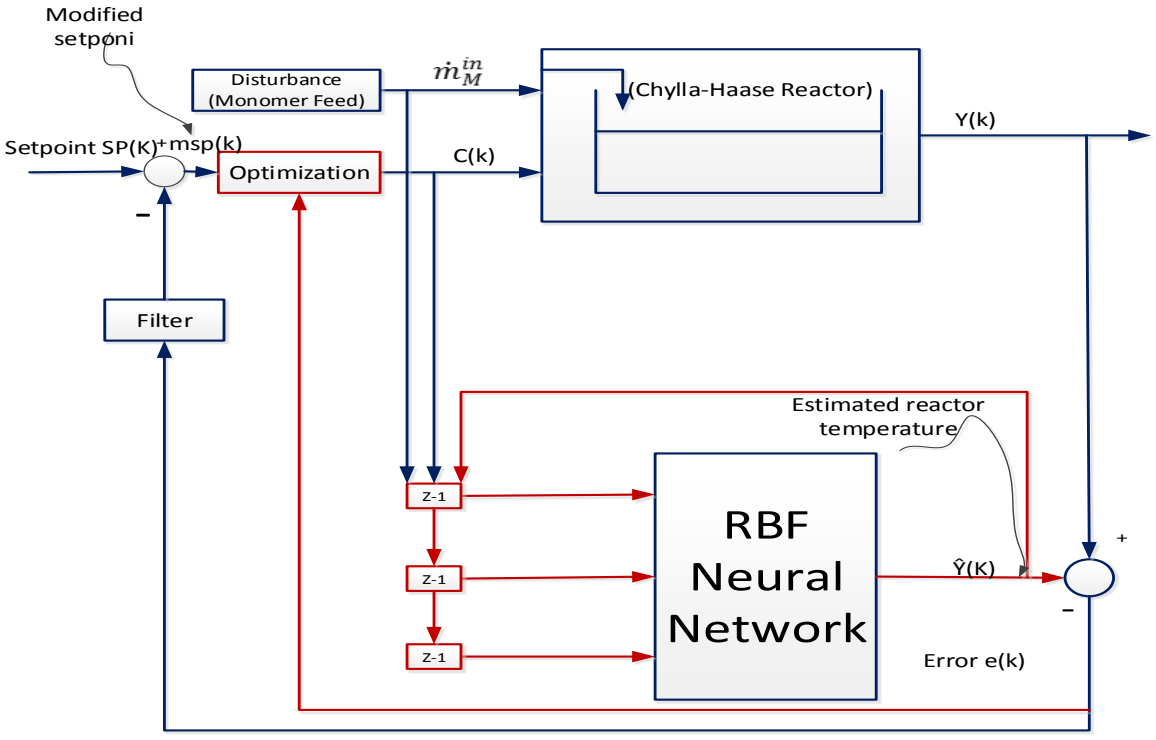


Figure 7.3. The strategy of non-adaptive RBFNN based MPC

7.5 Adaptive RBF-based MPC

Adaptive control is the specific type of control where the process is controlled in closed loop and where knowledge about the system characteristics is obtained online while the system is operating. Based upon refreshed information obtained during process operation, specific interventions in the control loop are made in order to fulfil the control goal. Adaptive control covers a set of techniques which provide a systematic approach for automatic adjustment of controller in real time, in order to achieve or to maintain a desired level of control system performance when the parameters of the plant dynamic model are unknown and/or change in time. First, the case when the parameters of the dynamic model of the plant to be controlled are unknown but constant (at least in a certain region of operation) is considered. In such cases, although the structure of the controller will not depend in general upon the particular values of the plant model parameters, the correct tuning of the controller parameters cannot be done without knowledge of their values. Adaptive control techniques can provide an automatic tuning procedure in closed-loop for the controller parameters. In such cases, the effect of the adaptation vanishes as time increases. Changes in the operation conditions may require a restart of the adaptation procedure.

7.5.1 Model adaptation

The strategy of the adaptive model is illustrated in Figure 7.4. Adaptation is applied to the RBF model of the nonlinear polymerization reactor to follow any parameter change and achieve robust control against disturbances. The adaptive RBFNN is used to model reactor temperature, and nonlinear optimizer is used to predict the optimal control valve position. The main difference between the adaptive and non-adaptive RBF based MPC is that the weighting in the adaptive RBF will be updated according to the error between RBF output and the reactor output. Moreover, the initial parameters of the

off-line of the non-adaptive RBF training will be saved in order to use them in the adaptive RBF at first sample time.

The main advantage of adaptive MPC over a non-adaptive is the on-line training of the adaptive model, which means it has the capability to model any unexpected changes in the process. Three different stages are applied to design the adaptive method, starting with collecting data from the polymerization reactor and finished by adaption of the RBFNN based MPC. The initial off-line training parameters of the non-adaptive RBFNN, *w* (0), sigma (0), were saved in order to use them in the adaptive MPC in the first sample. The centres and width remain fixed as they have been chosen to be distributed in the whole operating space, while the weights are adapted to minimize the error caused by disturbances and uncertainty.

The input of the RBFNN is illustrated in equation (7.3). The mean absolute error is used to evaluate the modelling and control performance.

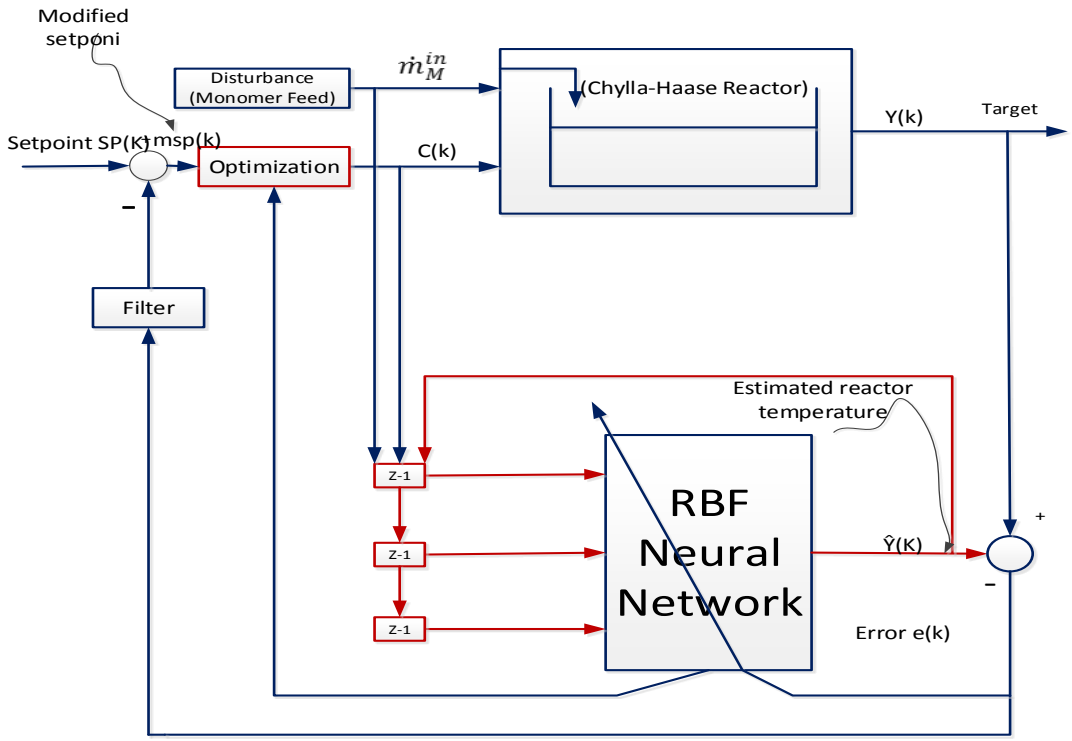


Figure 7.4 The strategy of adaptive RBFNN based MPC

7.5.2 Tuning of control parameters

The prediction horizons were tuned in the simulation to the values of N_1=1, N_2=22, the control horizon N_u is set to be zero. For the minimization of the cost function, and due to constraints and nonlinear nature of predictors the numerical optimization of the MPC was necessary. The controller used constrained (Fmincon) method from Matlab Optimization Toolbox as a nonlinear optimization algorithm. Fmincon allows imposing constraints with respect to the value of the control input such as upper or lower bounds. After attempting different values, the control weight is chosen as λ = 0.025, data sampling interval is 4 seconds.

7.5.3 Simulation result for polymer A

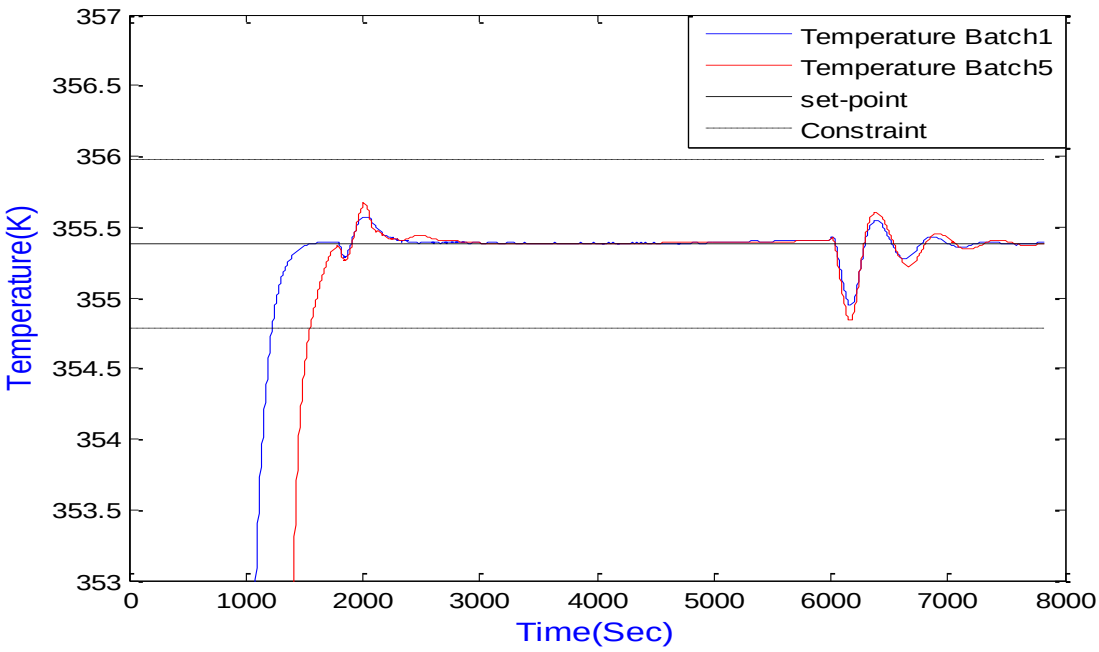
RBF neural network based model predictive control have already been introduced in many industries in particular to the chemical processes (Helbig et al., 1996, Pottmann and Seborg, 1997). Here, application to a complex process simulation of semi-batched reactor (Chylla-Haase) is considered. The proposed method was conducted to verify two objectives, set-point following with high quality product and disturbance rejection. The outcome of the simulation that the proposed optimizing scheme performs is shown in Figure 7.5. The set-point of the system is 355.382K within tolerance interval of $(\pm 0.6K)$ and the different values of monomer feed as disturbance are shown in Figure 7.5(c). The manipulated variable (Valve Position) is displayed in Figure 7.5(b). The MAE of the temperature tracking the set-point is 0.0217 for the first batch and 0.025 for the fifth batch. It is observed in Figure 7.5 that the reactor temperature follows the set-point within the tolerance range $(\pm 0.6K)$ with disturbances applied as shown in table 7.1. This result is due to the multi-step ahead prediction considered in the objective function.

Scenarios that has been applied in this research:

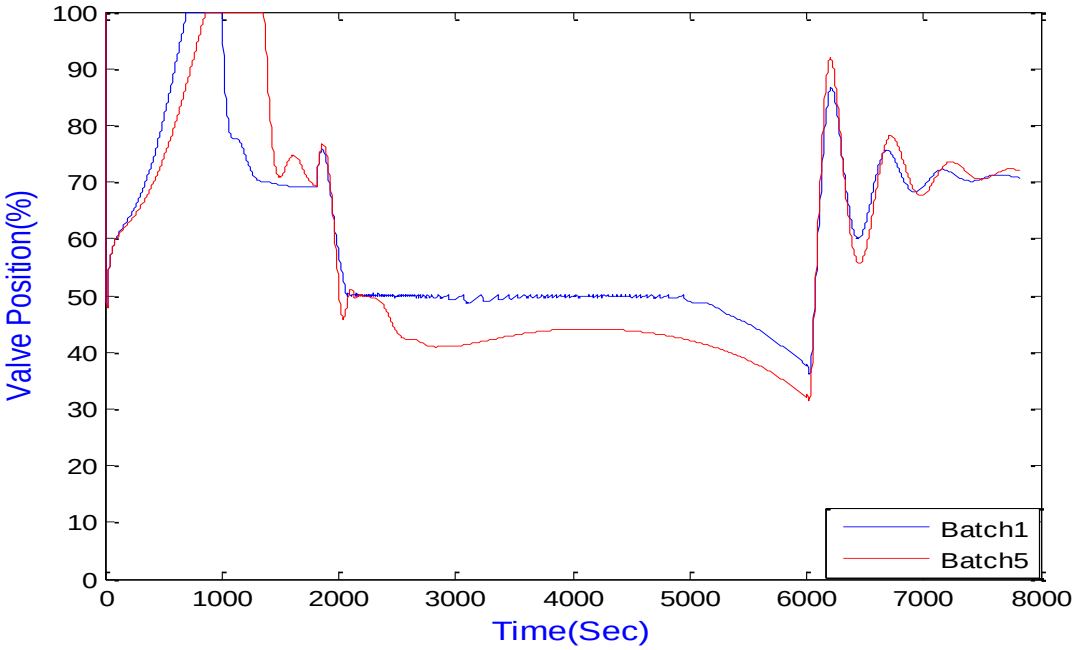
TABLE 7.1
CONSIDERED DISTURBANCES SCENARIO

scenario	$i[-]$	$1/_{hf}\,[\frac{m^2k}{kw}]$	T_{amb}
1	0.8	0	280.382
2	1.2	0.704	280.382
3	0.8	0	305.382
4	1.2	0.704	305.382

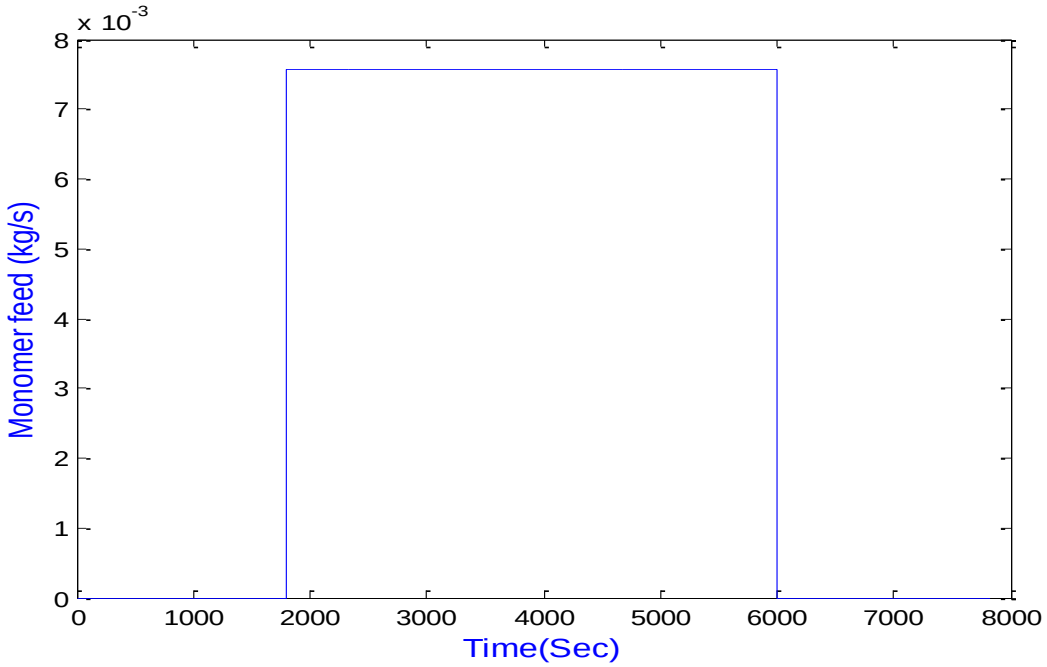
The response of reactor temperature for polymer A with impurity factor of 0.8 and 1.2 during winter condition for first and fifth batch respectively were obtained.



(a)

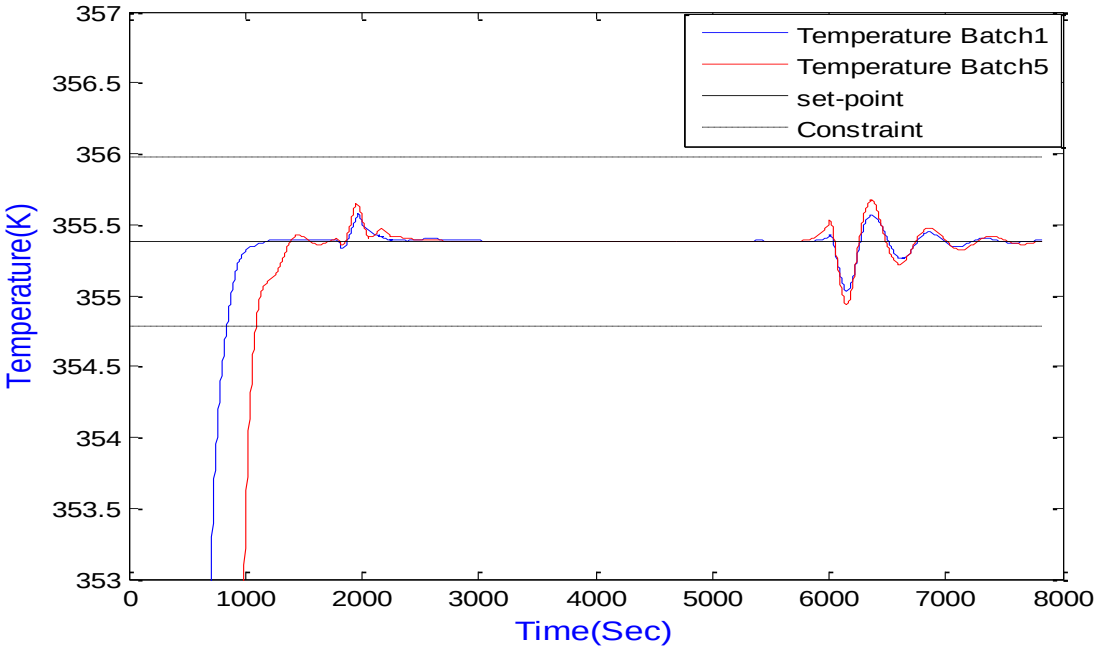


(b)

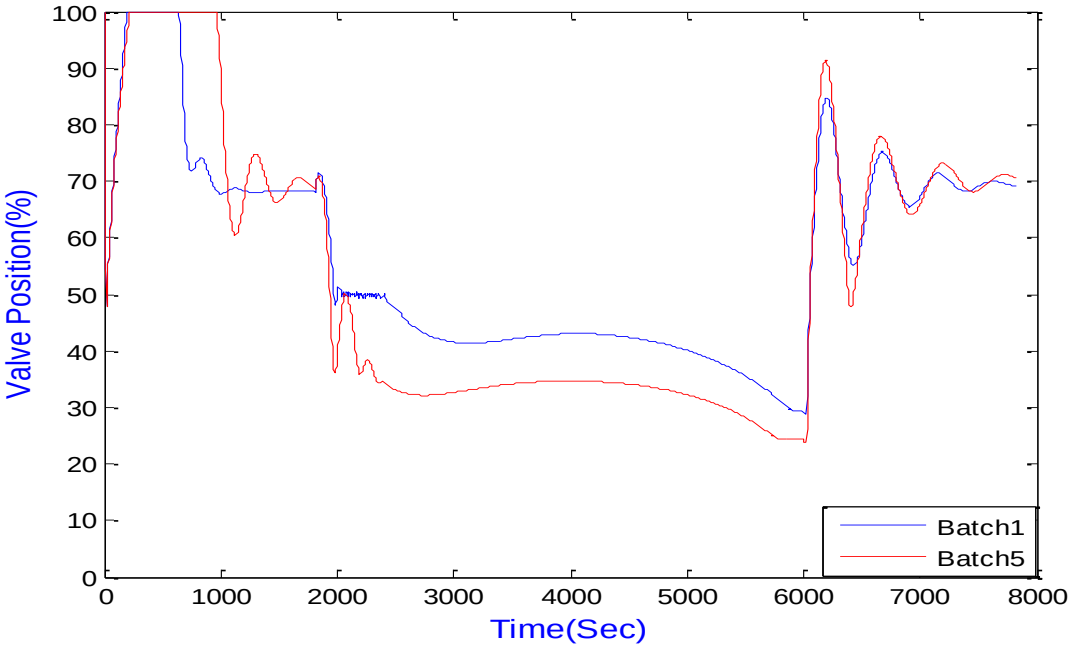


(c)

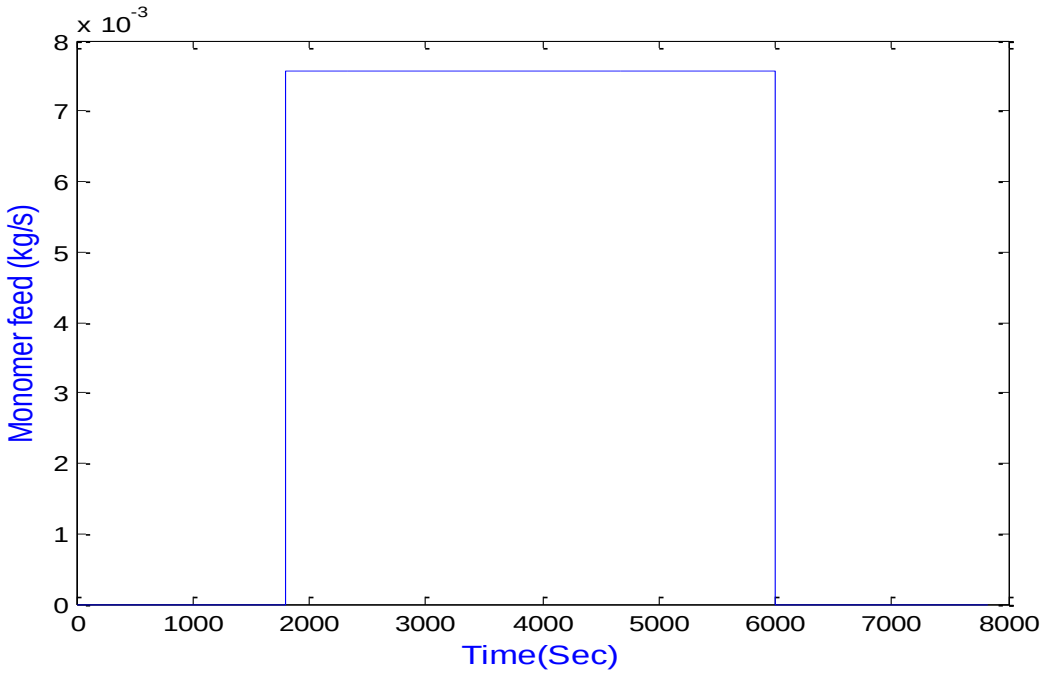
Figure 7.5 MPC result for polymer A for first and fifth batch with *i* =0.8 and 1.2 during winter condition: (a) reactor temperature (b) Valve Position (c) Monomer feed rate.


Temperature Batch1
Temperature Batch5
set-point
Constraint
Temperature(K)
Time(Sec)
357
356.5
356
355.5
355
354.5
354
353.5
353
0
1000
2000
3000
4000
5000
6000
7000
8000

(a)


Valve Position(%)
Time(Sec)
Batch1
Batch5
100
90
80
70
60
50
40
30
20
10
0
0
1000
2000
3000
4000
5000
6000
7000
8000

(b)



(c)

Figure 7.6 MPC result for polymer A for first and fifth batch with i =0.8 and 1.2 during summer condition: (a) reactor temperature (b) Valve position (c) Monomer feed fate.

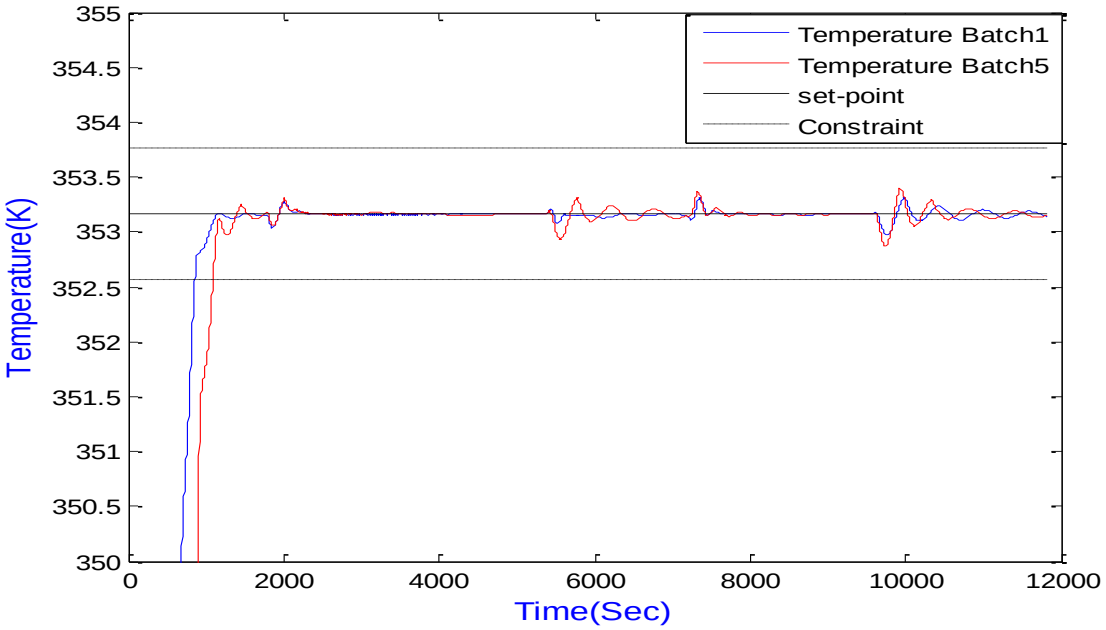
7.6 RBF-based MPC for polymer B

7.6.1 Tuning of control parameters

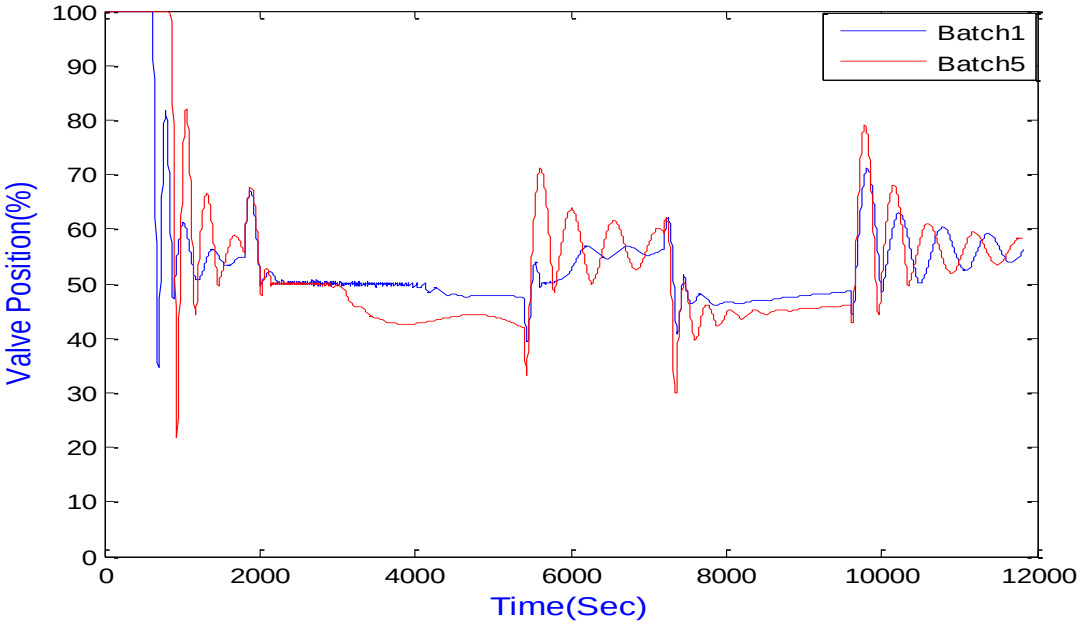
The set-point, the process batch time and some parameters are different here comparing with product A, see table 3.3. The set-point is 353.160K. So the tuning parameters such as the prediction horizons were tuned in the simulation to the values of $N_1 = 1$, $N_2 = 30$. The control horizon N_u is set to be zero. For the minimization of the cost function, and due to constraints and nonlinear nature of predictors the numerical optimization of the MPC was necessary. The constrained (optimization) method from Matlab Optimization Toolbox, "Fmincon", is used as the nonlinear optimization algorithm. Fmincon allows imposing constraints with respect to the value of the control input such as upper or lower bounds. After attempting different values, the control weight $\lambda = 0.045$ is chosen. The data sampling interval is 4 seconds.

7.6.2 Simulation results for polymer B

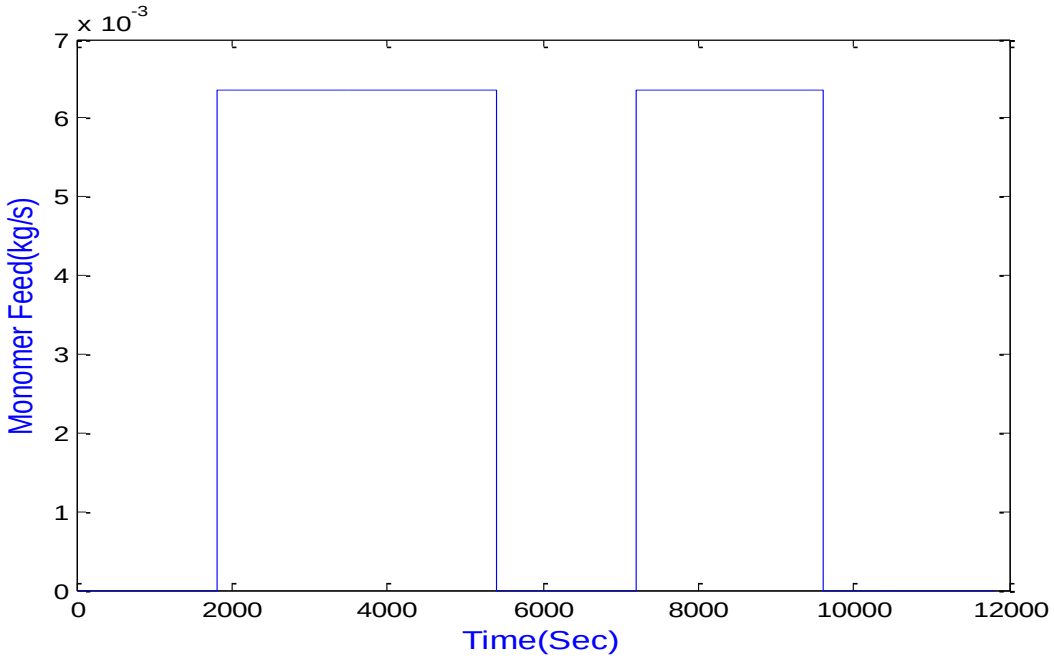
The response of reactor temperature for polymer B with impurity factor of 0.8 and 1.2 during winter condition for first and fifth batch together with disturbance and optimized control variable were obtained, and displayed in Figure. 7.7



(a)



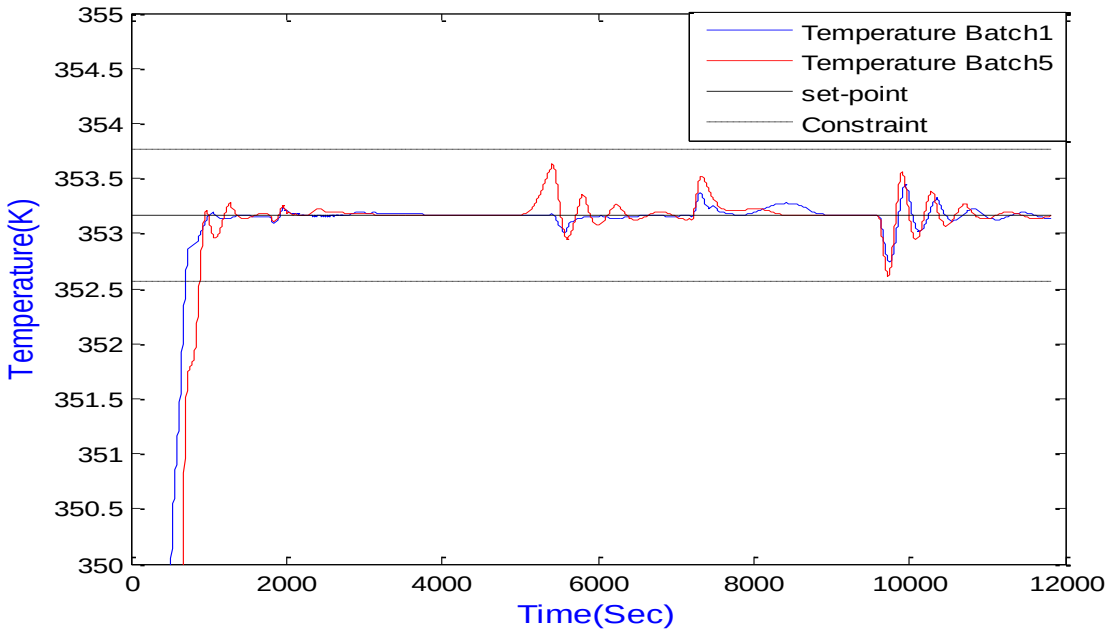
(b)



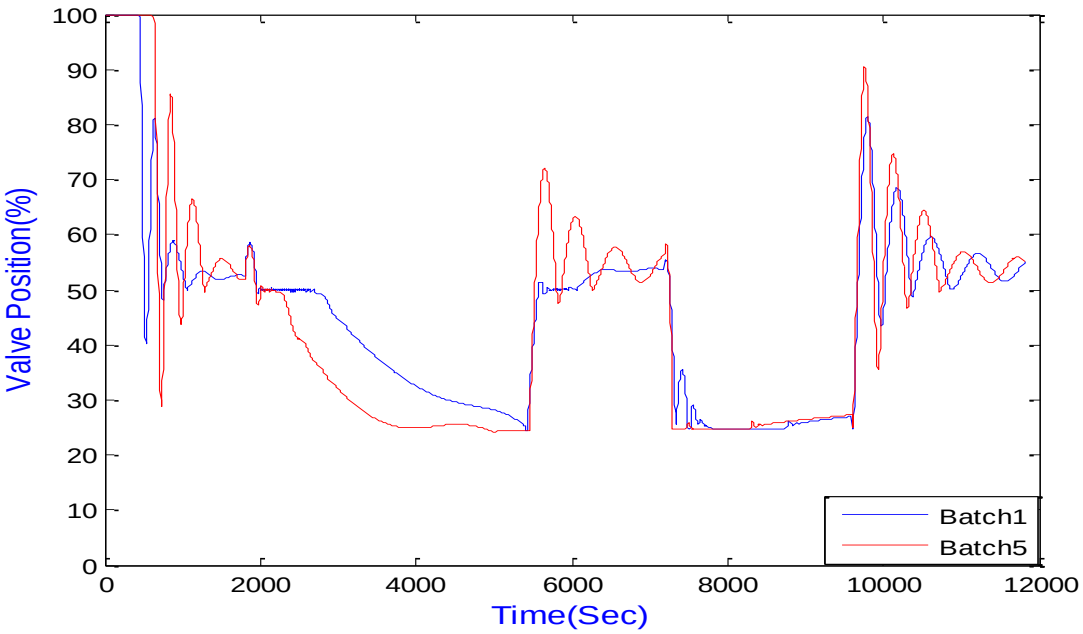
(c)

Figure 7.7 MPC result for polymer B for first and fifth batch with *i* =0.8 and 1.2 during winter condition: (a) reactor temperature (b) Valve position (c) Monomer feed rate

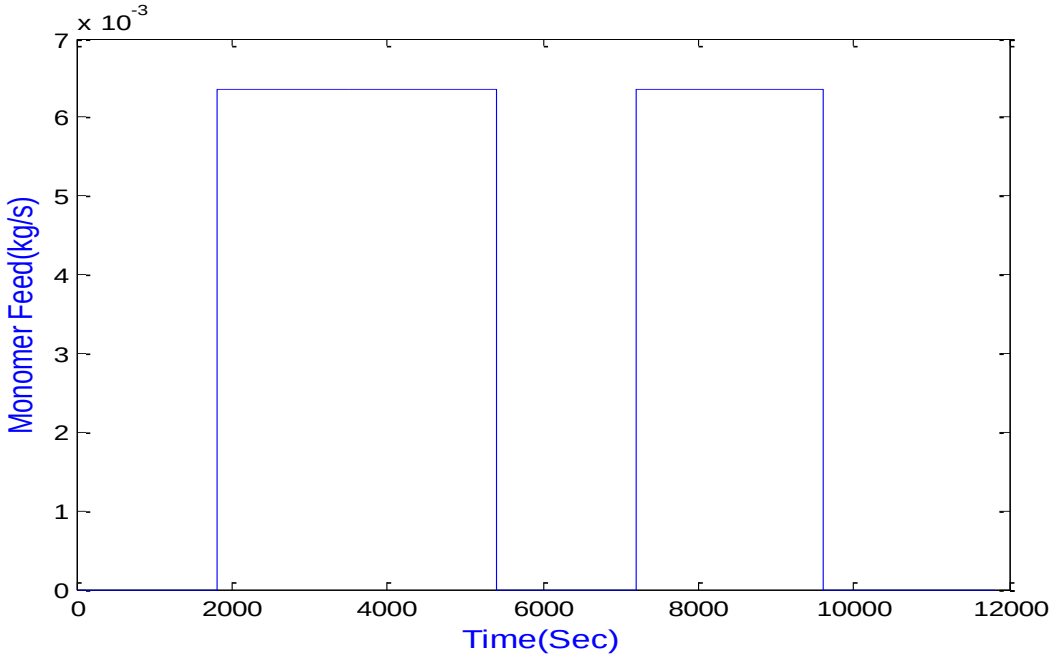
The response of reactor temperature for polymer B with impurity factor of 0.8 and 1.2 during summer condition for first and fifth batch are shown in Figure 7.8.



(a)



(b)



(c)

Figure 7.8 MPC result for polymer B for first and fifth batch with i =0.8 and 1.2 during summer condition: (a) reactor temperature (b) Valve position (c) Monomer feed rate

From Figure 7.7 and 7.8 it can be seen that adaptive RBFNN based MPC with prediction horizon values N_2= 30 has the ability to regulate the reactor temperature within an interval of +0.6K around the desired setpoint under all operation conditions and disturbances. The MAE of the temperature tracking are 0.032 and 0.039 for batch

1 and 5 for product B respectively. This performance is much better than the cascade controller that usually used to control such a process, this is due to the perfect modelling using RBF networks. The non-linear MPC controller based on RBF network captures the non-linear dynamics of the Chylla-Haase reactor and responds well to the changes. It can also see that when the control variable has an quick change due to monomer feed rate for both products A and B the manipulated variable (valve position) respond quickly to keep those changes within the limit and that's due to robust optimization consideration.

The proposed methods were conducted to verify two objectives, set-point following and disturbances to be anticipated and eliminated to permits the controller to drive the process output more closely to the reference trajectory. The outcome of the simulation that the proposed optimizing scheme performs is showed in previous section. It is observed that the reactor temperature follows the set-point within the tolerance range within presence of some disturbances and multi-product production, this is due to the multistep ahead prediction considered in the objective function.

7.7 MLP-based MPC for polymer B

Using the predictive control strategy with identified model (NNMPC) it is possible to calculate the optimal control sequence for nonlinear plant.

A feedforward, multilayer perceptron (MLP), error back propagation NN was used for the system identification as illustrated in chapter 5. The training and validation data sets are presented in Figures 5.15 and 5.16. The objective of the identification process is to minimize the error signal $e(t) = y(t) - \hat{y}(t)$ when the plant and NN model are subjected to the same input, see Figure 8.1, where $\hat{y}(k)$ is the NN model output, $y(k)$ is the plant output, $msp(t)$ is the modified set-point, $e_f(t)$ is the filtered error.

The MLPNN-MPC structure for the reactor is shown in Figure 7.9. The obtained MLP neural network model in chapter 5 is used to predict the reactor output for N_2 steps

ahead. The nonlinear optimizer minimizes the errors between the set-point and the predicted output, as well as the increment of the optimized control variable by using the cost function (7.1).

The strategy of the adaptive model is illustrated in Figure 7.9. Adaptation is applied to the MLP model of the nonlinear polymerization reactor to follow any parameter change and achieve robust control against disturbances.

This strategy is the same as the strategy used in RBF neural network.

The input of the MLPNN is illustrated in equation (7.3). The mean squared error is used to evaluate the modelling and control performance.

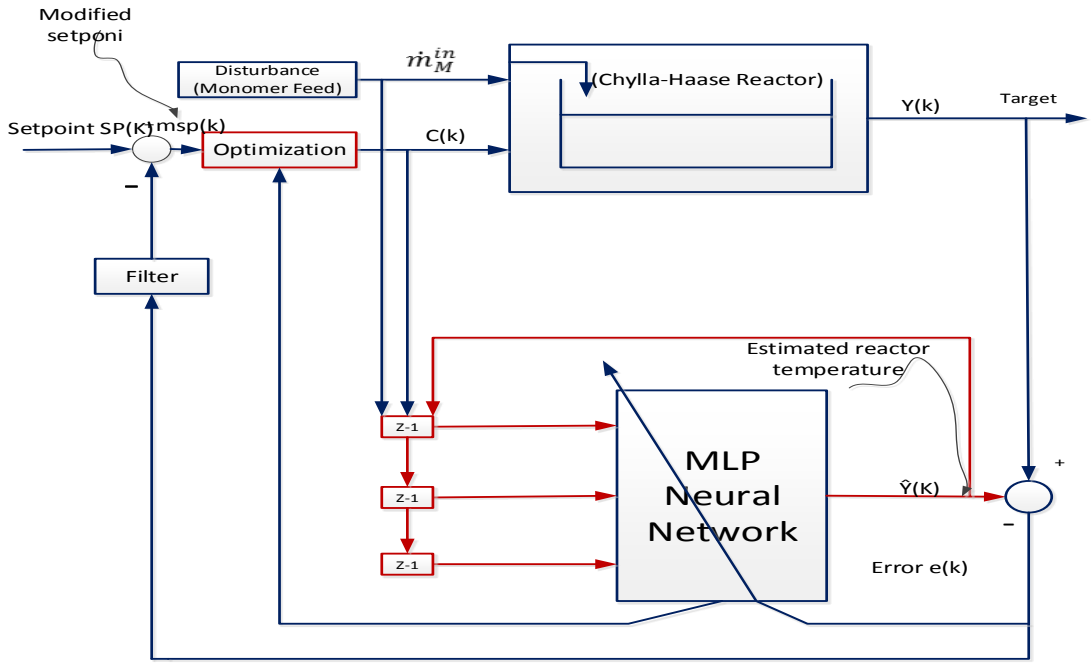


Figure 7.9 The strategy of adaptive MLPNN based on the MPC neural network

7.7.1 Tuning of control parameters

The control parameters in the objective function in adaptive MPC are N_1=1, N_2=20 and λ=0.03 and N_u=0. The optimization algorithm used to solve objective function (7.1) is Fmincon, which is available in the Matlab library, data sampling interval is 4 seconds.

7.7.2 Simulation results and discussion for polymer B

The MLPNN-MPC controller were applied to the Chylla-Haase reactor nonlinear model. The MLPNN-based identification process was implemented in MATLAB. The Chylla-Haase responses showed in Figure 7.10 show a good command tracking.

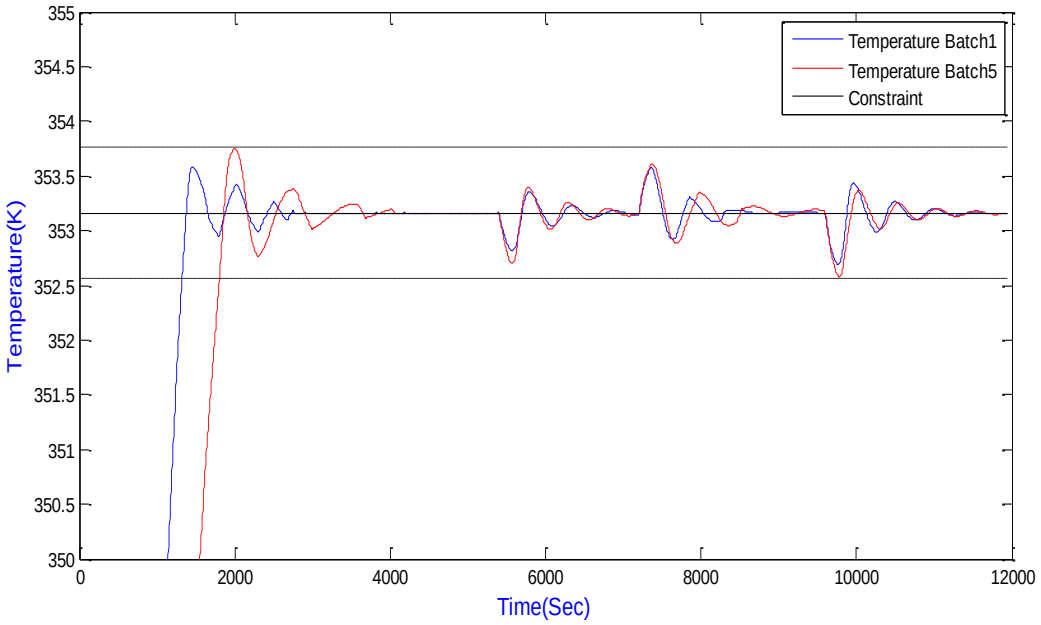
Figure 7.10 a,b,c. Displays reactor temperature, manipulated variable (Valve Position) and monomer feed rate respectively. The MAE of the temperature tracking the set-point is 0.875. It is observed that the reactor temperature follows the set-point within the tolerance range ($\pm 0.6K$) with disturbances applied as shown in table 7.2. This result is due to the multi-step ahead prediction considered in the objective function.

Scenarios that has been applied as followed:

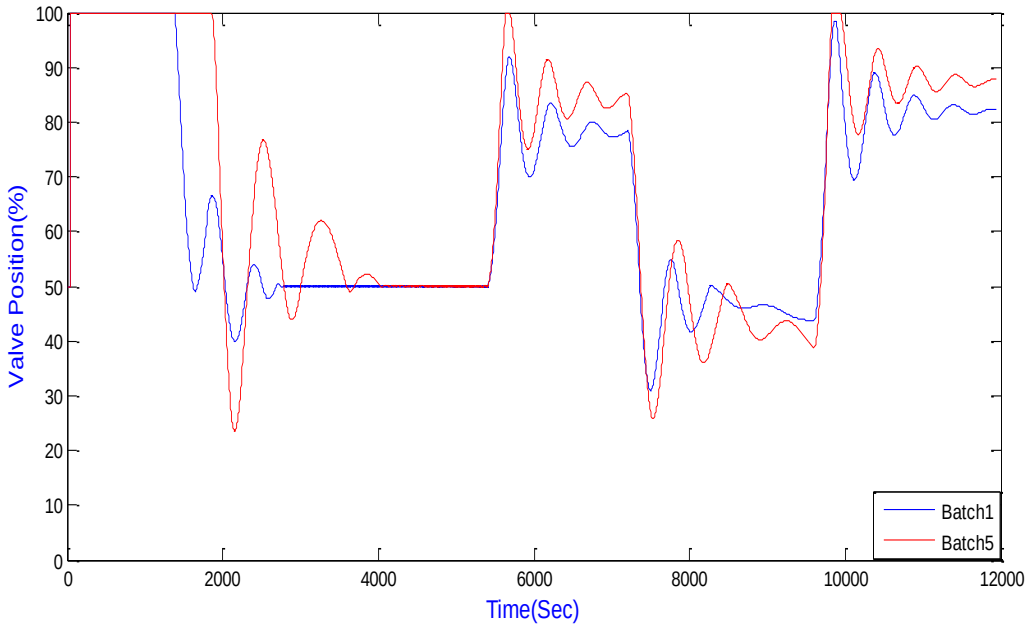
TABLE 7.2
CONSIDERED DISTURBANCES SCENARIO

scenario	$i[-]$	$1/_{hf}\,[\frac{m^2k}{kw}]$	T_{amb}
1	0.8	0	305.382
2	1.2	0.704	305.382

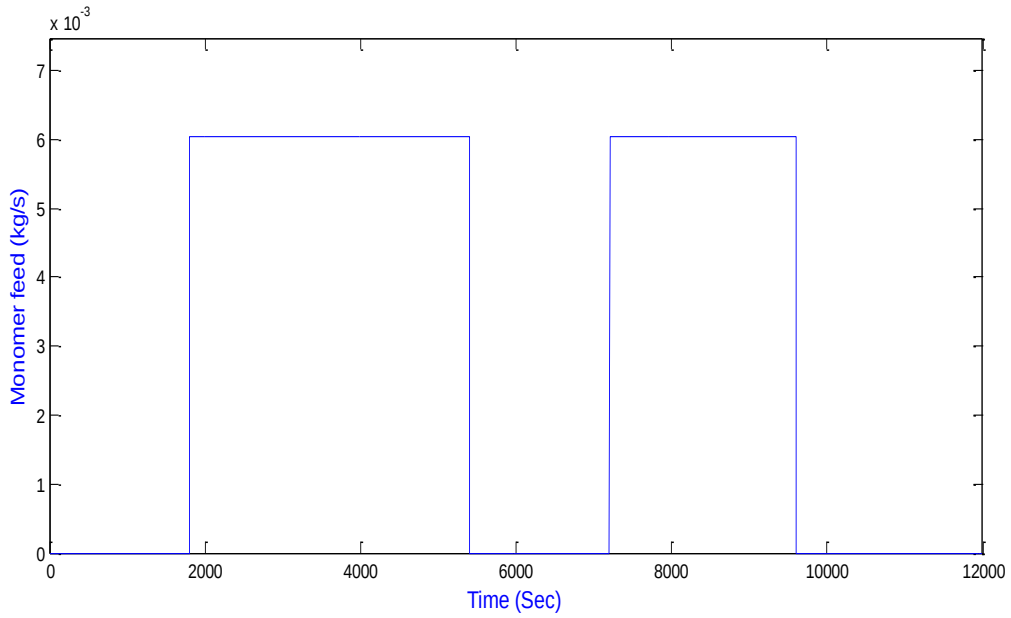
Only polymer B with impurity factor of 0.8 and 1.2 during winter condition for first and fifth batch respectively were considered here as it expected to be the most difficult scenario.


Temperature Batch1
Temperature Batch5
Constraint
Temperature(K)
Time(Sec)
355
354.5
354
353.5
353
352.5
352
351.5
351
350.5
350
0
2000
4000
6000
8000
10000
12000

(a)


Batch1
Batch5
Valve Position(%)
Time(Sec)
100
90
80
70
60
50
40
30
20
10
0
0
2000
4000
6000
8000
10000
12000

(b)



(c)

Figure 7.10 MPC result based on MLP for polymer B for first and fifth batch with *i* =0.8 and 1.2 during summer condition: (a) reactor temperature (b) Valve Position (c) Monomer Feed Rate

7.8 Summary

This chapter has discussed the RBF and MLP neural networks based MPC for polymerization reactor given by Chylla-Haase. The RBF and MLP used to model a nonlinear model and to predict the future output. The simulation shows that the trained neural network was capable of capturing dynamics with a very small performance tolerance indicating its high prediction accuracy. Implementation of the NNMPC controller was able to force process output to follow the target within its tolerance against influence of monomer feed and with different disturbances, impurity factor and fouling factor with four scenarios, first and fifth batch in summer, first and fifth batch in winter, also for different products, A and B.

On analysis of the response graphs, it can be seen that the NNMPC strategy successfully tracks the reference signal.

The result obtained for the random reference signal illustrates and proves the tracking ability of controller.

Chapter 8 Batch time minimization using MPC

8.1 Introduction

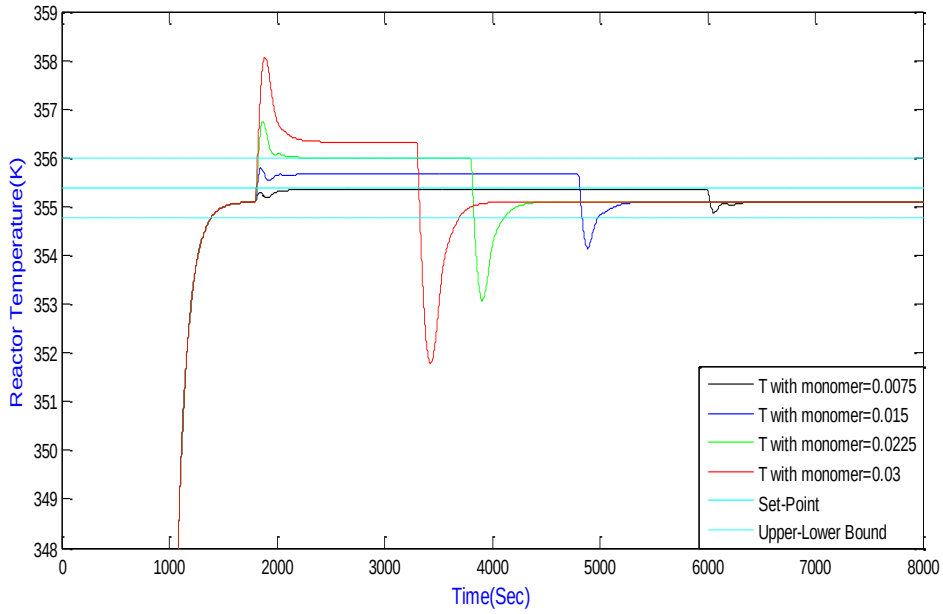
Polymerization reactors are usually operated with a constant monomer feed, monomers are usually slowly dosed into the reactor, leading to long batch times and limiting the process productivity (Finkler et al., 2013). In this chapter, model predictive control based on radial basis function is used for the time optimal operation of Chylla-Haase reactor that can be implemented for more flexibility and time saving for operators. The results show that the MPC control scheme can greatly shorten the reaction period without risking the quality of the temperature control. As the investigations here focus on the reduction of the batch time, the original cascade control structure is tested for cases where the value of monomer feed is increased so that the feeding period can be shortened and operation time can be reduced.

8.2 PID cascade controller

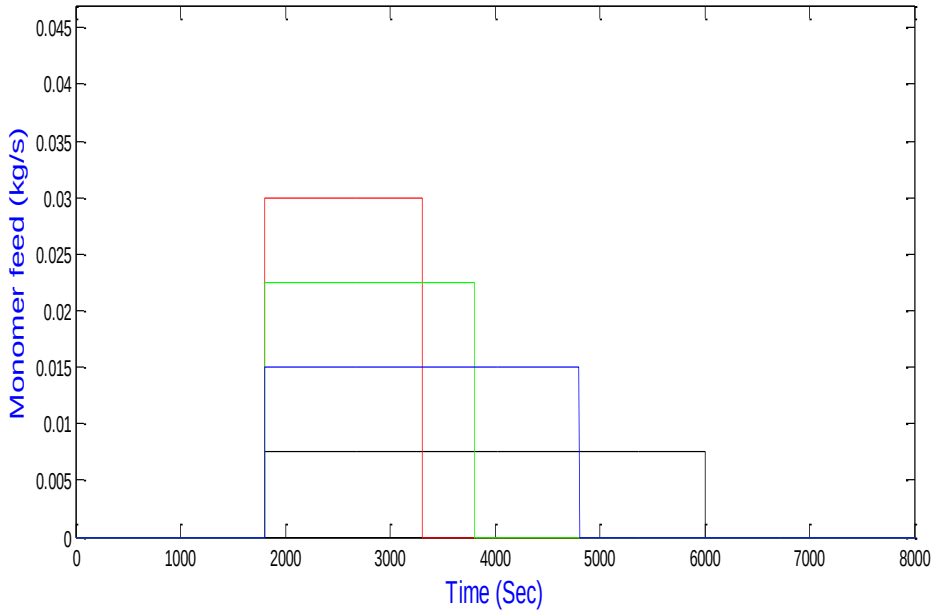
In the original control solution for the Chylla-Haase reactor, the system is operated using a constant monomer flow rate during the whole feeding period while a cascade of PID controllers is used to track a constant trajectory for the reactor temperature, as described in previous chapters. The Cascade control which is illustrated in chapter 4 is configured such that the master controller regulates the reactor temperature by manipulating the set point of the slave controller which controls the recirculation water temperature by acting on the heating/cooling mode.

From previous results obtained in chapter 4, the cascade of PID controllers is quite able to operate the system when the monomer inlet flows within the original range $7.5 * 10^{-3} kg/s$. However, when monomer feed rate goes beyond the original value in order to minimize the batch time and make more flexibility for operators, a temperature rise cannot be controlled due to a consequence of the continuous decrease in heat transfer coffecient (*UA*) due to the polymer formation along the batch, which

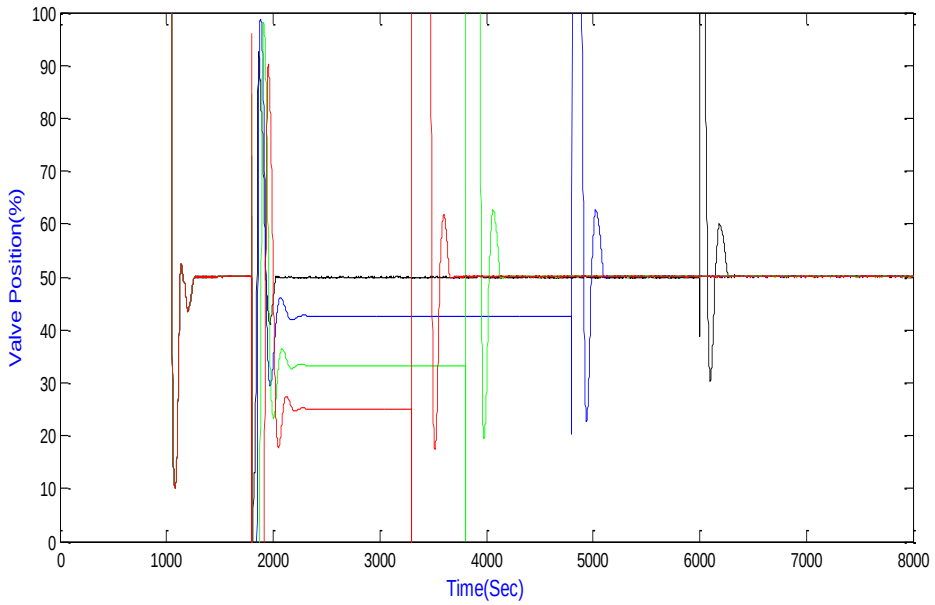
affects the close-loop system as a ramp-like disturbance also the amount of heat produced by the reaction is larger than the capacity of the cooling system and the reactor is driven out of control as it can be seen in Figure 9.1.



(a)



(b)



(c)

Figure 8.1 Performance of the CCs for polymer A for different monomer inlet during winter condition: (a) reactor temperature (b) Monomer feed rate (c) Valve position.

The chapter proposed is based on solving this issue to operate the system at large monomer inlet flow rates and reject disturbances so that a fast (less batch time) and robust operation during the feeding period can be guaranteed.

8.3 Multi-objective optimization and control

The same structure used in chapter 7 is used here, the NN-MPC structure for the reactor is shown in previous chapter Figure 7.3. The obtained RBF neural network model is used to predict the future process response over the specified horizon N_2. The predictions are passed to a numerical optimization routine, which attempts to minimize a specified cost function (9.1) in the calculation of a suitable control signal at each sample instant. Minimizing the inverse of monomer feed rate is to maximize the monomer feed rate, so that the batch time is minimized (shortened). The objective function used in the optimization is:

$$J = \sum_{i=t+N_1}^{t+N_2}[y_{set}(i) - \hat{y}(i)]^2 + \lambda_1 \sum_{i=t}^{t+N_u}[u(i) - u(i-1)]^2 + \lambda_2 \sum_{i=t+N_1}^{t+N_2} \frac{1}{\dot{m}_M^{in}} \quad (8.1)$$

$$u_{min} \leq u \leq u_{max} \tag{8.2}$$

$$\dot{m}_M^{in} > 0 \tag{8.3}$$

This minimization is subject to constraint (9.2 & 9.3) valve position. Here, N_1 , N_2 defines the prediction horizon, λ is a control weighting factor, N_u control horizon, u is manipulated variable, y_{set} is the modified set-point to introduce a feedback in order to compensate the system steady-state error and against disturbance effects.

The remaining main problem of MPC is to solve the nonlinear optimization problem in each sample period, calculate a series of optimal $\mathrm{u(t)}, \mathrm{u(t+1)}, \ldots, \mathrm{u(t+N_u\text{-}1)}$ from which the neural network model generates output to minimize J (Pottmann and Seborg, 1997, Ding-Li et al., 2006).

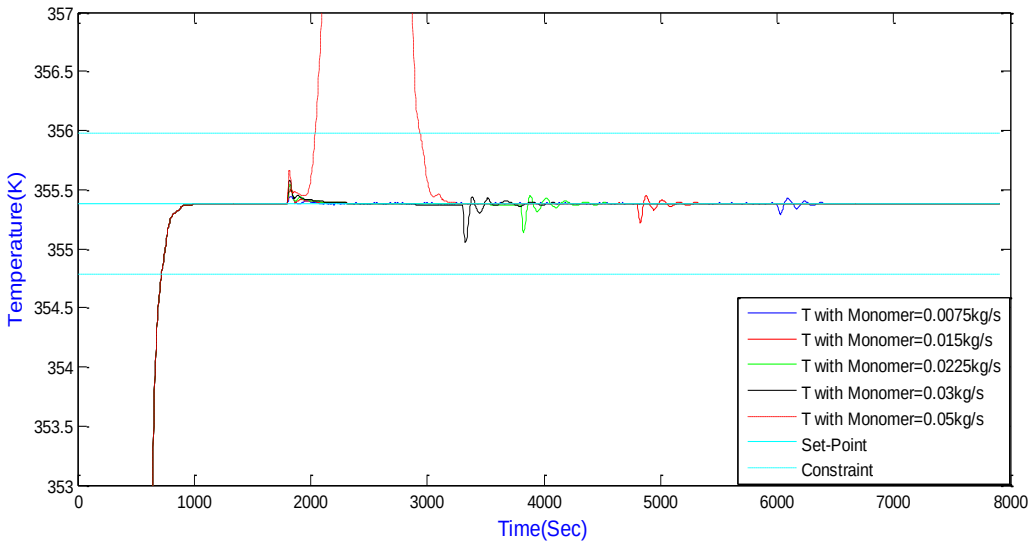
8.4 Control parameter tuning

The prediction horizons were set at their best values of N_1=1, N_2=30, the control horizon N_u is set to be zero. As mentioned before, for the minimization of the cost function, and due to constraints and nonlinear nature of predictors the numerical optimization of the MPC was necessary, the controller used a multi-objective optimization algorithm from Matlab Optimization Toolbox, (gamultiobj) allows imposing constraints with respect to the value of the control input such as upper or lower bounds. After attempting different values, the control weight is $\lambda_1 = 0.025$, $\lambda_2 = 0.1$ data sampling interval is 4 seconds.

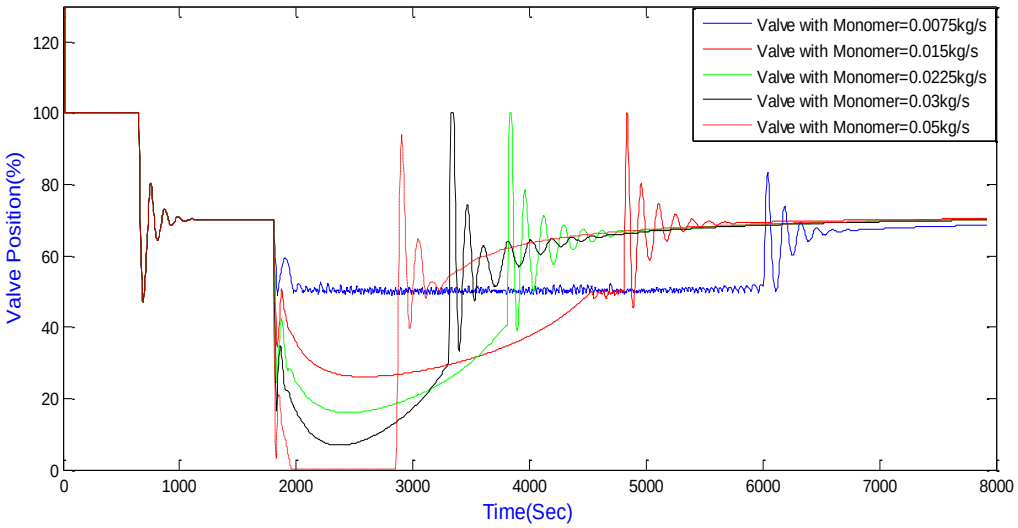
8.5 Simulation results

The proposed methods were conducted to verify two objectives, set-point following and reducing batch time (less batch duration) with high quality product. The outcome of the simulation that the proposed optimizing scheme performs is shown in Figure 9.2. The set-point of the system is 355.382K within tolerance interval of ($\pm 0.6\ K$) the different value of monomer feed (disturbance) shown in Figure 9.2 (c). The

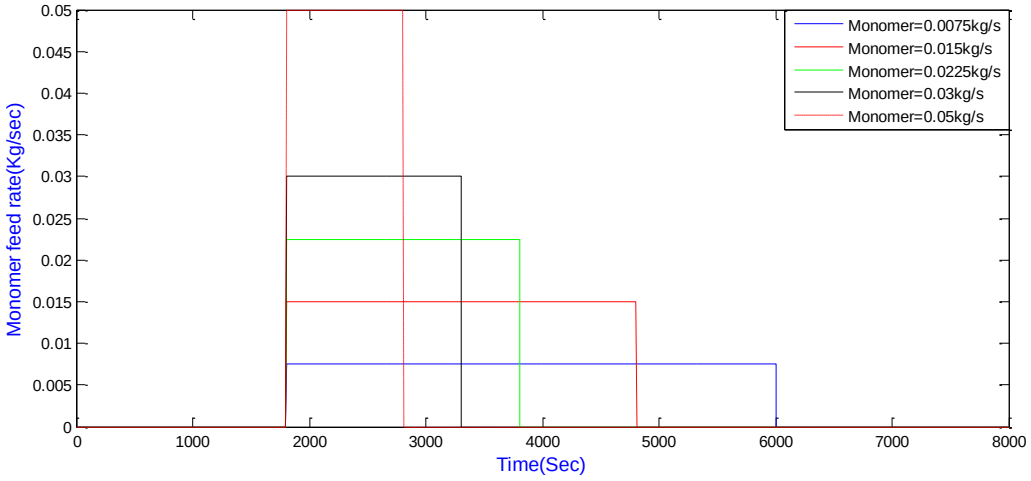
manipulated variable (Valve Position) displayed in Figure 9.2 (b). It is observed that, the reactor temperature follows the set-point within the tolerance range ($\pm 0.6\ K$) with maximum of 0.03kg/s of monomer beyond this value its impossible to keep the temperature with in the tolerance range as demonstrated in Figure 9.2 (a) with dotted red line. The results of an investigation of higher monomer flows with the goal to reduce the feeding period using the CCs for temperature control are compared with MPC-RBF. In these simulations, the performance of the temperature control is compared for the cases where monomer feed rate in is set to 0.00756 kg/s, 0.015 kg/s, 0.0225 kg/s, 0.03 kg/s and 0.045kg/s. As 45 kg of monomer have to be fed into the reactor during the feeding period, the corresponding durations of the feeding period are 70, 50, 33.3, 25 and 16.6 min, respectively.



(a)



(b)



(c)

Figure 8.2 MPC result for polymer A for different monomer inlet during winter condition: (a) reactor temperature (b) Valve Position (c) Monomer Feed Rate.

8.6 Summary

Model predictive control based on an adaptive RBF model is applied to Chylla-Haase polymerization reactor. RBFNN has strong ability to model highly nonlinear process and is a powerful tool specially for an on-line adaptation of the predictor during the control. The main goal was to reduce the batch time as much as possible without driving the temperature control out of the performance bounds. As in the original setup of the Chylla−Haase benchmark problem it is not possible to reduce the batch time

because the monomer inlet flow is constant. A modified version of the benchmark problem where variable monomer feed are allowed was considered. A new objective function involving batch time function is formulated and the optimization problem is solved by employing a multi-objective function optimization algorithm in Matlab optimization toolbox. The control scheme fulfils the temperature requirements for chosen values of the uncertain parameter and robustly minimizes the batch time.

Efficient optimizing control structure that maximizes the monomer feed by using a NMPC based on RBFNN, this scheme has the ability to robustly optimize the control variable by minimizing an objective function. The objective function is defined in terms of both present and predicted system variables and is evaluated using an explicit model to predict future process outputs.

The simulation results show that the proposed scheme is able to robustly operate the process even with the large impact results from maximize the monomer feed rate. The reaction period was significantly shorted compared to the original recipe with constant monomer flow rate.

Chapter 9 Conclusions and Further Work

9.1 General Discussion

On comparison of the responses obtained from the simulated polymerization reactor with the conventional control and feed-forward feedback control (chapter 4) the performance of the proposed model predictive control (chapter 7) is found to be superior and meets the control objective of maintaining the reactor temperature within the tolerance interval of ±0.6K from the set point. Although, feed-forward feedback control based on RBF neural networks gives an excellent result for keeping the reactor temperature within the tolerance range but still cannot give a good result because the control variable sometimes is higher than the upper bound 100% and lower than the lower bound 0%.

The results of the neural network based MPC were obtained for both seasons, different batches and different products, and for the reduction of batch time only product A was used here (chapter 8). It may be considered in the future to also use product B for the possibility of batch time reduction.

The present work leads to further development of advanced model predictive control schemes considering stability and optimization speed, which is explained in more details in the section 9.3.

9.2 Conclusions

This work undertaken a theoretical and experimental study treating the subjects of modelling, control and optimization of Chylla-Haase polymerization reactor. The research studies have achieved the following main results:

- The first part of this research is focused primarily on the development of ordinary differential equation models based on the kinetic and differential equations published by (Chylla and Haase, 1993). Also develop a simulation

model that describes the dynamic behaviour of polymerization process using Matlab/Simulink. It allows to understand the behaviour of such a high non-linear process.

- Next, conventional control methods are designed and implemented for Chylla-Haase reactor. PID control is investigated and the results are discussed. Moreover, LQR control is implemented based on a linear model of the plant using Matlab linearization tools. Furthermore, design of model predictive control based on linearized model using MPC Toolbox is investigated .
- Further, some advanced control scheme are proposed in this research using different methods. The RBFNN based inverse model control is implemented using PID controller. A PID controller is used in the feedback control to adjust the deference between the requested and the actual reactor temperature by compensating the neural network inverse model output. The RBFNN inverse model is made adaptive to cope with the significant parameters uncertainty, disturbances and environment changes. Moreover, RBFNN based MPC scheme is proposed for Chylla-Haase reactor. The adaptive RBFNN is trained by the RLS method for modelling, so that the change on reactor parameters is compensated. The RBFNN is used to model a nonlinear model and to predict the future outputs. The nonlinear optimizer minimizes the errors between the set-point and the predicted output, as well as the increment of the optimized control variable by using the cost function. The developed MPC scheme shows an excellent result in terms of control of the reactor temperature. The optimization scheme using Matlab's Optimal Toolbox functions (Fmincon) shows a good robustness, which allows dealing with either unconstrained or constrained optimization problems. Matlab function "Fmincon" allows

imposing constraints with respect to the value of the control input such as upper or lower bounds.

In addition, replacing the RBFNN with MLPNN and investigate the comparison. From simulation result the RBF networks generally give better results than MLPs. In general, the RBF network is more strong against a bad training set than an MLP and, hence, provides better results. For a good training set, a significant improvement would be expected for an RBF network relative to an MLP.

Research on MPC scheme on Chylla-Haase reactor has also been done by using the same MPC scheme based on RBFNN used to reduce the batch time as much as possible without driving the temperature control out of the performance bounds. As in the original setup of the Chylla−Haase benchmark problem, it is not possible to reduce the batch time because the monomer inlet flow is constant. A modified version of the benchmark problem where variable monomer feed are allowed was considered. The simulation shows that implementation of the NNMPC controller was able to force process output to follow the target within its tolerance against disturbances and monomer feed inlet variation in limit.

In conclusion, the four advanced schemes for Chylla-Haase polymerization reactor have been proposed and developed in this research. Their effectiveness and robustness have been assessed by computer simulation on a widely used Matlab. Flexibility and feasibility of these methods are also analysed and discussed by comparison. It provides important research base for further development of control methods for Chylla-Haase control problems, especially

for the scheme of MPC based on RBFNN for control and reduction of batch time.

9.3 Further work

Although the methods presented in this thesis have demonstrated the effectiveness of the proposed approach, they could be further developed in a number of ways:

While model predictive controllers can be effective in the proper situation, they have limitations. Optimisation speed, exact linearization MPC and to guarantee the whole system stability are the main problems that need to be solved. For the speed of the optimization, more rapid non-linear optimisation algorithms need to be developed and investigated. Exact linearization will allow to cancel the nonlinearities in a nonlinear system in such a way that the closed-loop dynamics appears in a linear form. Another unsolved and active research problem of MPC is its stability proof for the system involving neural networks, which is also one of the most important and difficult research topics in the neural network community.

Different neural networks models combined with MPC have been widely employed in process control applications. It is necessary to evolve a scheme for development of a black box model in which the model structure can be selected relatively easily and the resulting model is valid over a wide operating range.

Finally, exact linearization MPC is an interesting subject that maybe implemented on the proposed scheme. Roughly speaking, exact linearization tries to cancel the nonlinearities in a nonlinear system in such a way that the closed-loop dynamics appears in a linear form. It is a transformation for mapping a nonlinear system into a linear one without any approximation which

occurs, for instance, in the Taylor-expansion-based linearization approach. In this research RBF neural network is used to approximate the non-linear dynamic system function in high-precision. Therefore, it is suggested that if the parameter linearization technique of Taylor series is employed to do partial or impartial linearization of RBF neural network output, then the results will be more accurate. On other words, instead of approximating the entire nonlinear system with neural network, we get the exact linearization model which allow to get a better results.

REFERNCES

ARTIN HATZIKIOSEYIAN & REMOUNDAKI, E. 2005. *Bioreactors for metal bearing wastewater treatment* [Online]. Available: http://www.metal.ntua.gr/~pkousi/e-learning/bioreactors/page_04.htm [Accessed 2014].

BAHMAN ZARENEZHAD & AMINIAN, A. 2011. APPLICATION OF THE NEURAL NETWORK-BASED MODEL PREDICTIVE CONTROLLERS IN NONLINEAR INDUSTRIAL SYSTEMS. CASE STUDY. *Journal of the University of Chemical Technology and Metallurgy,* 46**,** 67-74.

BAYKAN, N. A. & YILMAZ, N. 2011. A mineral classification system with multiple artificial neural network using k-fold cross validation. *Mathematical and Computational Applications,* 16**,** 22.

BELLGARDT, K. H., KUHLMANN, W., MEYER, H. D., SCHUGERL, K. & THOMA, M. 1986. Application of an extended Kalman filter for state estimation of a yeast fermentation. *Control Theory and Applications, IEE Proceedings D,* 133**,** 226-234.

BEQUETTE, B. W. 2003. *Process control: modeling, design, and simulation*, Prentice Hall Professional.

BEYER, M.-A., GROTE, W. & REINIG, G. 2008. Adaptive exact linearization control of batch polymerization reactors using a Sigma-Point Kalman Filter. *Journal of Process Control,* 18**,** 663-675.

BHAT, A. & BANAVAR, R. N. The Chylla-Haase problem: a neural network controller. Control Applications, 1998. Proceedings of the 1998 IEEE International Conference on, 1-4 Sep 1998 1998. 192-196 vol.1.

CAMACHO, E. F. & ALBA, C. B. 2013. *Model predictive control*, Springer Science & Business Media.

CERVANTES, A. L., AGAMENNONI, O. E. & FIGUEROA, J. L. 2003. A nonlinear model predictive control system based on Wiener piecewise linear models. *Journal of Process Control,* 13**,** 655-666.

CHYLLA, R. W. & HAASE, D. R. 1993. Temperature control of semibatch polymerization reactors. *Computers & Chemical Engineering,* 17**,** 257-264.

CLARKE-PRINGLE, T. & MACGREGOR, J. F. 1997. Nonlinear adaptive temperature control of multi-product, semi-batch polymerization reactors. *Computers & Chemical Engineering,* 21**,** 1395-1409.

DING-LI, Y., DING-WEN, Y. & GOMM, J. B. 2006. Neural model adaptation and predictive control of a chemical process rig. *Control Systems Technology, IEEE Transactions on,* 14**,** 828-840.

DOHERTY, S. K. 1999. *Control of pH in chemical processes using artificial neural networks.* Liverpool John Moores University.

DORADO, F. & BORDONS, C. Constrained nonlinear predictive control using Volterra models. Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference on, 19-22 Sept. 2005 2005. 7 pp.-1013.

DUBÉ, M. A., PENLIDIS, A., MUTHA, R. K. & CLUETT, W. R. 1996. Mathematical Modeling of Emulsion Copolymerization of Acrylonitrile/Butadiene. *Industrial & Engineering Chemistry Research,* 35**,** 4434-4448.

FAZLUR RAHMAN, M. H. R., DEVANATHAN, R. & ZHU, K. 2000. Neural network approach for linearizing control of nonlinear process plants. *Industrial Electronics, IEEE Transactions on,* 47**,** 470-477.

FERNANDES, F. & LONA, L. 2005. Neural network applications in polymerization processes. *Brazilian Journal of Chemical Engineering,* 22**,** 401-418.

FINKLER, T. F., DOGRU, M. B. & ENGELL, S. 2012. Optimizing Control of Semi-batch Polymerization Reactors. *Proceedings of Advances in Control and Optimization of Dynamic Systems ACODS-2012***,** 1-5.

FINKLER, T. F., LUCIA, S., DOGRU, M. B. & ENGELL, S. 2013. Simple Control Scheme for Batch Time Minimization of Exothermic Semibatch Polymerizations. *Industrial & Engineering Chemistry Research,* 52**,** 5906-5920.

FLORES-CERRILLO, J. & MACGREGOR, J. F. 2005. Latent variable MPC for trajectory tracking in batch processes. *Journal of Process Control,* 15**,** 651-663.

GARCIA, C. E. & MORARI, M. 1985. Internal model control. 2. Design procedure for multivariable systems. *Industrial & Engineering Chemistry Process Design and Development,* 24**,** 472-484.

GOMM, J., YU, D. L. & WILLIAMS, D. A new model structure selection method for non-linear systems in neural modelling. Control '96, UKACC International Conference on (Conf. Publ. No. 427), 2-5 Sept. 1996 1996. 752-757 vol.2.

GRAICHEN, K., HAGENMEYER, V. & ZEITZ, M. Adaptive Feedforward Control with Parameter Estimation for the Chylla-Haase Polymerization Reactor. Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on, 12-15 Dec. 2005 2005. 3049-3054.

GRAICHEN, K., HAGENMEYER, V. & ZEITZ, M. 2006. Feedforward control with online parameter estimation applied to the Chylla–Haase reactor benchmark. *Journal of Process Control,* 16**,** 733-745.

GRUBER, J. K., GUZMÁN, J. L., RODRÍGUEZ, F., BORDONS, C., BERENGUEL, M. & SÁNCHEZ, J. A. 2011. Nonlinear MPC based on a Volterra series model for greenhouse temperature control using natural ventilation. *Control Engineering Practice,* 19**,** 354-366.

HAICHEN, Y. & ZHIJUN, Z. Predictive Control Based on Neural Networks of The Chemical Process. Control Conference, 2006. CCC 2006. Chinese, 7-11 Aug. 2006 2006. 1143-1147.

HELBIG, A., ABEL, O., M'HAMDI, A. & MARQUARDT, W. Analysis and nonlinear model predictive control of the Chylla-Haase benchmark problem. Control '96, UKACC International Conference on (Conf. Publ. No. 427), 2-5 Sept. 1996 1996. 1172-1177 vol.2.

HELBIG, A., ABEL, O. & MARQUARDT, W. Model predictive control for online optimization of semi-batch reactors. American Control Conference, 1998. Proceedings of the 1998, 21-26 Jun 1998 1998. 1695-1699 vol.3.

HOWLETT, R. J. & JAIN, L. C. 2001. *Radial basis function networks 1: recent developments in theory and applications*, Springer Science & Business Media.

HUAFANG, N., DEBELAK, K. & HUNKELER, D. Fuzzy logic supervised ANN feedforward control for a batch polymerization reactor. American Control Conference, Proceedings of the 1995, 21-23 Jun 1995 1995. 1777-1781 vol.3.

JALILI-KHARAAJOO, M. & ARAABI, B. N. Neuro predictive control of a heat exchanger: comparison with generalized predictive control. Electronics, Circuits and Systems, 2003. ICECS 2003. Proceedings of the 2003 10th IEEE International Conference on, 14-17 Dec. 2003 2003. 675-678 Vol.2.

JALILI-KHARAAJOO, M. & ARAABI, B. N. 2011. Neural network based predictive control of a heat exchanger nonlinear process. *IU-Journal of Electrical & Electronics Engineering*, 4**,** 1219-1226.

JUNG-WOOK, P., HARLEY, R. G. & VENAYAGAMOORTHY, G. K. Comparison of MLP and RBF neural networks using deviation signals for on-line identification of a synchronous generator. Power Engineering Society Winter Meeting, 2002. IEEE, 2002 2002. 274-279 vol.1.

KANDROODI, M. R. & MOSHIRI, B. Identification and model predictive control of continuous stirred tank reactor based on artificial neural networks. Control, Instrumentation and Automation (ICCIA), 2011 2nd International Conference on, 27-29 Dec. 2011 2011. 338-343.

LI, D., LI, Z., GAO, Z. & JIN, Q. 2014. Active Disturbance Rejection-Based High-Precision Temperature Control of a Semibatch Emulsion Polymerization Reactor. *Industrial & Engineering Chemistry Research,* 53**,** 3210-3221.

LIGHTBODY, G., IRWIN, G. W., TAYLOR, A., KELLY, K. & MCCORMICK, J. Neural network modelling of a polymerisation reactor. Control, 1994. Control '94. International Conference on, 21-24 March 1994 1994. 237-242 vol.1.

LUCIA, S., FINKLER, T., BASAK, D. & ENGELL, S. 2012. A new Robust NMPC Scheme and its Application to a Semi-batch Reactor Example. *8th IFAC Symposium on Advanced Control of Chemical Processes,The International Federation of Automatic Control*. Furama Riverfront, Singapore,.

MANER, B. R. & DOYLE, F. J. 1997. Polymerization reactor control using autoregressive-plus Volterra-based MPC. *AIChE Journal,* 43**,** 1763-1784.

MARUSAK, P. M. 2009. Advantages of an easy to design fuzzy predictive algorithm in control systems of nonlinear chemical reactors. *Applied Soft Computing,* 9**,** 1111-1125.

MCLOONE, S., BROWN, M. D., IRWIN, G. & LIGHTBODY, G. 1998. A hybrid linear/nonlinear training algorithm for feedforward neural networks. *Neural Networks, IEEE Transactions on,* 9**,** 669-684.

MEDINA-RAMOS, C. & NIETO-CHAUPIS, H. Controlling the temperature of a batch reactor through a NMPC scheme based on the Laguerre-Volterra model. Industry Applications (INDUSCON), 2010 9th IEEE/IAS International Conference on, 8-10 Nov. 2010 2010. 1-6.

MILLS, P. M., ZOMAYA, A. Y. & TADE, M. O. Application of adaptive neural model-based control. Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on, 15-17 Dec 1993 1993. 2804-2809 vol.3.

MING-GUANG, Z., WEN-HUI, L. & MAN-QIANG, L. Adaptive PID Control Strategy Based on RBF Neural Network Identification. Neural Networks and Brain, 2005. ICNN&B '05. International Conference on, 13-15 Oct. 2005 2005. 1854-1857.

NAGY, Z. K., MAHN, B., FRANKE, R. & ALLGÖWER, F. 2007. Evaluation study of an efficient output feedback nonlinear model predictive control for temperature tracking in an industrial batch reactor. *Control Engineering Practice,* 15**,** 839-850.

NANDA, S. & PHARM, M. 2008. Reactors and fundamentals of reactors design for chemical reaction. *Doctoral Report. Dept of pharmaceutical sciences. MD University Rohtak, Haryana*.

NG, C. W. & HUSSAIN, M. A. 2004. Hybrid neural network—prior knowledge model in temperature control of a semi-batch polymerization process. *Chemical Engineering and Processing: Process Intensification,* 43**,** 559-570.

NOOR, R., AHMAD, Z., DON, M. & UZIR, M. 2010. Modelling and control of different types of polymerization processes using neural networks technique: a review. *The Canadian Journal of Chemical Engineering,* 88**,** 1065-1084.

NYSTRÖM, A. 2007. *Modeling and Simulation of a Multi Phase Semi-batch Reactor*. LICENTIATE OF ENGINEERING, Chalmers University of Technology and G ¨oteborg University.

OGATA, K. 1997. Modern Control Engineering, terza edizione, 1997. Prentice Hall.

ORR, M. J. 1996. Introduction to radial basis function networks. Technical Report, Center for Cognitive Science, University of Edinburgh.

POTTMANN, M. & SEBORG, D. E. 1997. A nonlinear predictive control strategy based on radial basis function models. *Computers & Chemical Engineering,* 21**,** 965-980.

QIN, S. J. & BADGWELL, T. A. 2003. A survey of industrial model predictive control technology. *Control Engineering Practice,* 11**,** 733-764.

RICHARDS, J. R. & CONGALIDIS, J. P. 2006. Measurement and control of polymerization reactors. *Computers & Chemical Engineering*, 30**,** 1447-1463.

ROBERT HABER, R. B., AND ULRICH SCHMITZ. 2011. *Predictive Control in Process Engineering,* WILEY-VCHVerlag GmbH & Co. KGaA.

SAKATA, R., MASUDA, S., SUGIYAMA, T. & SATO, T. A design method for nonlinear model predictive control with a step-type input constraint. ICCAS-SICE, 2009, 2009. IEEE, 2169-2172.

SEBORG, D. E., F.EDGAR, T. & MELLICHAMP, D. A. 2004. *Process Dynamics and Control*, John Wily & sons, USA.

SHAHROKHI, N. & PISHVAIE, M. R. Artificial neural network feedforward/feedback control of a batch polymerization reactor. American Control Conference, 1998. Proceedings of the 1998, 21-26 Jun 1998 1998. 3391-3395 vol.6.

SHAMEKH, A. & LENNOX, B. 2010. Latent Variable MPC and its Consistency in Temperature Control of Batch Processes. *Proceedings of the 9th International Symposium on Dynamics and Control of Process Systems.* Leuven, Belgium.

SIROHI, A. & CHOI, K. Y. 1996. On-Line Parameter Estimation in a Continuous Polymerization Process. *Industrial & Engineering Chemistry Research,* 35**,** 1332-1343.

SOROUSH, M. 1997. Nonlinear state-observer design with application to reactors. *Chemical Engineering Science,* 52**,** 387-404.

TIAN, Y., ZHANG, I. & MORRIS, A. J. 2001. Modeling and Optimal Control of a Batch Polymerization Reactor Using a Hybrid Stacked Recurrent Neural Network Model. 40**,** 4525-4535.

TIAN, Y., ZHANG, X. & ZHU, R. 2007. Design of Waveguide Matched Load Based on Multilayer Perceptron Neural Network. *Proceedings of ISAP, Niigata, Japan*.

VARZIRI, M. S. 2008. *PARAMETER ESTIMATION IN NONLINEAR CONTINUOUS-TIME DYNAMIC MODELS WITH MODELLING ERRORS AND PROCESS DISTURBANCES.* Doctor of Philosophy, Queen's University.

VASANTHI, D., PRANAVAMOORTHY, B. & PAPPA, N. Artificial Neural Network tuned cascade control for temperature control of polymerization reactor. Intelligent Control (ISIC), 2011 IEEE International Symposium on, 28-30 Sept. 2011 2011. 1367-1372.

VASANTHI, D., PRANAVAMOORTHY, B. & PAPPA, N. 2012. Design of a self-tuning regulator for temperature control of a polymerization reactor. *ISA Transactions,* 51**,** 22-29.

VASIČKANINOVÁ, A. & BAKOŠOVÁ, M. 2009. Neural network predictive control of a chemical reactor. *Acta Chimica Slovaca,* 2**,** 21-36.

VENKAT, A. N. 2006. *Distributed Model Predictive Control: Theory and Applications.* DOCTOR OF PHILOSOPHY, UNIVERSITY OF WISCONSIN–MADISON.

VOJTESEK, J. & DOSTAL, P. 2009. Simulation of adaptive control of continuous stirred tank reactor. *International Journal of Simulation Modelling,* 8**,** 133-144.

WANG, L. 2009. *Model Predictive Control System Design and Implementation Using MATLAB 2009,* Natick-USA, Springer.

WANG, Z. L., CORRIOU, J. P. & PLA, F. Nonlinear control of a batch polymerization reactor with on-line parameter and state estimations. Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on, 15-17 Dec 1993 1993. 3858-3863 vol.4.

WON JUNG YOON, YANG SOO KIM, IN SUN KIM & CHOI, A. K. Y. 2004. Recent advances in polymer reaction engineering: Modeling and control of polymer properties. *Korean Journal of Chemical Engineering.*, 21**,** 147-167.

XIE, T., YU, H. & WILAMOWSKI, B. Comparison between traditional neural networks and radial basis function networks. IEEE International Symposium on Industrial Electronics, 2011.

YINGNONG, D., CHONGXHAO, H. & FANG, Y. Robust nonlinear MPC based on simplified second-order Volterra series model. Intelligent Control and Automation, 2000. Proceedings of the 3rd World Congress on, 2000 2000. 3269-3273 vol.5.

YU, D. L. & GOMM, J. B. 2003. Implementation of neural network predictive control to a multivariable chemical reactor. *Control Engineering Practice,* 11**,** 1315-1323.

YU, D. L., GOMM, J. B. & WILLIAMS, D. 1999. Online predictive control of a chemical process using neural network models

Proc.14th IFAC Congress,. Beijing.

YU, D. L., GOMM, J. B. & WILLIAMS, D. 2000. Neural model input selection for a MIMO chemical process. *Engineering Applications of Artificial Intelligence,* 13**,** 15-23.

YU, D. L. & YU, D. W. 2006. A new structure adaptation algorithm for RBF networks and its application. *Neural Comput. Appl.*, 16**,** 91-100.

YU, D. L., YU, D. W., GOMM, J. B. & AND WILLIAMS, D. 2002. Model predictive control of a chemical process based on an adaptive neural network. *Proc. of 15th IFAC World Congress.* Barcelona, Spain.

YU, D. L. & ZHAI, Y. J. 2008. Radial-basis-function-based feedforward–feedback control for air–fuel ratio of spark ignition engines. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering,* 222**,** 415-428.

YU, D. W. & YU, D. L. 2003. A comparison study on a chemical reactor modelling with a physical model and PLRBF networks. *Engineering Applications of Artificial Intelligence,* 16**,** 629-645.

YUAN, T., JIE, Z. & MORRIS, J. On-line re-optimisation control of a batch polymerisation reactor based on a hybrid recurrent neural network model. American Control Conference, 2001. Proceedings of the 2001, 2001 2001. 350-355 vol.1.

ZHANG, Y., ZHAO, D. & ZHANG, J. Research on PID controller based on RBF neural network. Electronics and Optoelectronics (ICEOE), 2011 International Conference on, 29-31 July 2011 2011. V1-443-V1-445.

Appendix A

Symbols

$T(t)$	Reactor temperature
T_{amb}	Ambient temperature
T_{steam}	Steam temperature
T_{wall}	Wall temperature
T_j^{in}	Coolant inlet temperature
T_j^{out}	Jacket outlet temperature
$m_M(t)$	Monomer mass
$m_p(t)$	Polymer mass
m_M^{in}	Monomer feed rate
m_c	Mass of coolant
$\dot{m}_c$	Coolant rate
m_W	Mass of water
MW_M	Molecular weight of monomer
$C_{p,M}$	Specific heat of monomer at constant pressure
$C_{p,C}$	Specific heat of coolant at constant pressure
$C_{p,P}$	Specific heat of polymer at constant pressure
$C_{p,W}$	Specific heat of water at constant pressure
Q_{rea}	Reaction heat
ΔH	Reaction enthalpy
A	Jacket heat transfer area
U	Overall heat transfer coefficient
$(UA)_{loss}$	Heat loss coefficient
$K_p(c)$	Heating/cooling function

$C(t)$	Control valve position
R_p	Polymerization rate
R	Natural gas constant
B_1	Reactor bottom area
B_2	Jacket bottom area
P	Jacket perimeter
θ_1	Jacket transport delay
θ_2	Recirculation transport delay
ρ_M	Density of monomer
ρ_P	Density of polymer
ρ_C	Density of coolant
μ	Batch viscosity
μ_{wall}	Wall viscosity
τ_p	Heating/cooling time constant
$1/h_f$	Fouling factor
h	Film heat transfer coefficient
d_0, d_1	Constants
i	Impurity factor
K	First order kinetic constant
$k_0, k_1, c_0, c_1, c_2,$ c_3, a_0, E	Constants
$\dot{m}_M^{in,max}$	Maximum monomer feed rate
$[t_{M,0}^{in}, t_{M,1}^{in}]$	Feed period time interval for polymer A and B
$[t_{M,2}^{in}, t_{M,3}^{in}]$	Feed period time interval for polymer B

Appendix B

Abbreviations

NN	Neural Network
CCs	Cascade Controller
MSE	Mean Square Error
MAE	Mean Absolute Error
MLP	Multi-Layer Perceptron
RBF	Radial Basis Function
FBFNN	Radial basis function neural network
PID	Proportional-Integral-Derivative
LQR	Linear Quartic Regulator
MPC	Model Predictive Control
LMPC	Linear Model Predictive Control
NNMPC	Neural network model predictive control
NMPC	Non-linear model predictive control
RLS	Recursive Least Square
FB	Feedback
FF	Feed-forward
MIMO	Multi-inputs multi-outputs
MISO	Multi-inputs single-outputs
PLRBF	Pseudo-linear radial basis function
DMC	Dynamic matrix control
EKF	Extended Kalman filter
SPKF	Sigma-Point Kalman filter
UKF	Unscented Kalman filter
RAS	Random amplitude signals

Appendix C

Author publication list

Conference paper:

1. M. Nabi, Yu. Dingli, and Feng Yu3, (2013). "RBF Model Based Predictive Control for Exothermic Semi-batch Polymerizations," Proceedings of the 19th International Conference on Automation & Computing, Brunel University, London.
2. M. Nabi and Yu. Dingli, (2014). “RBF Modelling and Optimization Control for Semi-Batch Reactors,” International Journal of Chemical, Nuclear, Metallurgical and Materials Engineering, London, Vol:8 No:7, pages 646-650.
3. M. Nabi and Yu. Dingli, (2014). “Nonlinear Adaptive PID Control for a Semi-Batch Reactor Based On an RBF Network.” International Journal of Computer, Information, Systems and Control Engineering, Paris, Vol:8 No:7, pages 1157-1162.