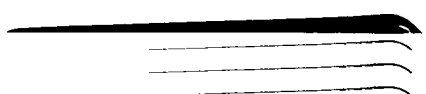


NONLINEAR PROCESS-MODEL-BASED CONTROL AND OPTIMIZATION OF BATCH POLYMERIZATION REACTOR

by

JIN YONG CHOI, B.S., M.S.Ch.E.

A DISSERTATION

IN

CHEMICAL ENGINEERING

Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
the Requirements for
the Degree of

DOCTOR OF PHILOSOPHY

Approved

May, 1994

ACKNOWLEDGMENTS

I am deeply indebted to Dr. R. Russell Rhinehart, chairman of my advisory committee, for his valuable advice, direction, encouragement and continuous financial support that he has given to me throughout the course of my PhD study.

A sincere appreciation is expressed to other committee members for their suggestions and criticism throughout this work. They are Dr. James Riggs, Dr. Richard Tock, Dr. Richard Bartsch and Dr. William Kolarik.

I also very grateful to The Dow Chemical Company for their financial support for this project. A special thanks goes to Dr. Kip Mercure of The Dow Chemical Company who coordinates the project, for his guidance.

This dissertation is dedicated to my parents, my wife Yoon, and daughter Seojin for their love, support and encouragement.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

ALPHABETICAL

A : heat transfer area, m^2

$\overline{A}$: predicted heat transfer area, m^2

A_b : reactor bottom area, m^2

a_1 : first-order autoregressive coefficient

b_i : smoothing constant for i-th time period

$C_{p,i}$: heat capacity of initiator, KJ $kg^{-1}K^{-1}$

$C_{p,m}$: heat capacity of monomer, KJ $kg^{-1}K^{-1}$

$C_{p,s}$: heat capacity of solvent, KJ $kg^{-1}K^{-1}$

$C_{p,p}$: heat capacity of polymer, KJ $kg^{-1}K^{-1}$

c : coefficient that determines drift amplitude

cv_{mtd} : coefficient of variance for mean temperature difference between reactor and jacket temperature

D : diffusivity of polymer molecule

D_i : moles of dead polymer with degree of polymerization i

$_nD$: moles of n^{th} class dead polymer

d : vector of measured or unmeasured disturbance

d_i : density of initiator, kg/m^3

d_m : density of monomer, kg/m^3

d_s : density of solvent, kg/m^3

d_p : density of polymer, kg/m^3

$E_d, E_{p0}, E_{\theta p}, E_{tm}, E_{ts}, E_{to}, E_{\theta t}, E_{tr}$: kinetic parameter, KJ $kgmole^{-1}$

EH_{loss} : heat loss to environment, KJ/sec

$F(s,t)$: generating function for dead polymer

$F_{cw,max}$: maximum flow obtainable from the cooling water control valve, m^3/sec

$F_{s,max}$: maximum flow obtainable from the steam flow control valve, kg/sec

$FS_{I,II,III}$: solvent feed rate at i^{th} period, kgmole/sec

$FI_{I,II,III}$: initiator feed rate at i^{th} period, kgmole/sec

f : efficiency factor for initiator

f_d : reactor side fouling factor, $secm^2{}^oK/KJ$

f_n : frequency function for a discrete variable n

$f(n)$: density function for a continuous variable n

f : normalized frequency function

$G(s,t)$: generating function
generating function for growing polymer

HF_{target} : higher tail fraction target of MWD

ΔH_p : heat of polymerization, KJ/kgmole

h_i : reactor side heat transfer coefficient, $KJ/m^2{}^oKsec$

h_o : jacket side heat transfer coefficient, $KJ/m^2{}^oKsec$

I : concentration of initiator, kgmole/m^3

ID_r : reactor inside diameter, m

I_o : initial initiator loading, kgmole/m^3

K : number of samples since last change in x_{spc}

k_D : mass transfer portion of the termination rate constant (k_t), m^3/kgmole sec

k_t : termination rate constant, m^3/kgmole sec

k_{to} : kinetic portion of the termination rate constant (k_t), m^3/kgmole sec

$k_{\theta t}$: rate expression which accounts for initiator-loading-dependence of k_D

k_1 : $1/\tau$, proportional gain of GMC law, sec^{-1}

k_2 : integral gain parameter of GMC law

k_d^o, $k_{\theta p}^o$, $k_{\theta p}^o$: kinetic parameter, s^{-1}

k_{po}^o, k_{tm}^o, k_{ts}^o, k_{to}^o, k_{tr}^o : kinetic parameter, $m^3kgmole^{-1}s^{-1}$

k_{tc} : termination rate by combination, $m^3sec/kgmole$

k_d : termination rate by disproportion, $m^3sec/kgmole$

k_p : propagation rate, $m^3sec/kgmole$

k_t : termination rate, $m^3sec/kgmole$

k_{tm} : chain transfer rate to monomer, $m^3sec/kgmole$

k_{ts} : chain transfer rate to solvent, $m^3sec/kgmole$

LF_{target} : lower tail fraction target of MWD

M : concentration of monomer, $kgmole/m^3$

M_o : initial monomer loading, kgmole

MW_i : molecular weight of initiator, kg/kgmole

MW_m : molecular weight of monomer, kg/kgmole

MW_s : molecular weight of solvent, kg/kgmole

m_s : steam flow rate, kg/sec

N : number of samples

n : chain length or class number of polymer

n_I : moles of initiator, kgmole

$\bar{n}_I$: predicted moles of initiator, kgmole

n_i : noise at i-th time period

n_M : number of moles of monomer

P : sum of moles of growing polymer, kgmole

P_i : moles of growing polymer with degree of polymerization i, kgmole

P(n) : probability of event n

$_nP$: moles of n^{th} class of growing polymer, kgmole

$P_n^{\cdot}$: growing polymer with chain length n

p : vector of parameters of the process

P(n) : probability of event n.

q_i : moving average/autoregressive moving average at ith period

R : universal gas constant, KJ/kgmole_°K

$R^{\cdot}$: initiator radical

r_1, r_2 : random number, 0.0<r1,r2<1.0

S : concentration of solvent, kgmole/m^3

$S^{\cdot}$: solvent radical

S_o : initial solvent loading, kgmole

T : sampling interval

$T_{I,II,III}$: reactor temperature setpoint at i^{th} period, °C

T_{amb} : ambient temperature, °C

T_{cw} : chilled water temperature, °C

T_j : jacket temperature, °C

$T_{j,i}$: initial jacket temperature, °C

T_r : reactor temperature, °C

$T_{r,i}$: initial reactor temperature, °C

T_w : reactor wall temperature, °C

T_{gp} : glass transition temperature, °K

T^{σ} : saturation temperature of steam, °C

U : overall heat transfer coefficient, kw/m^2sec°C

u : input variable

$\bar{u}$: filtered measured value of u

V_j : jacket volume, m^3

V_{cl} : circulation line volume, m^3

V_{sys} : system volume, m^3

$\dot{V}_w$: chilled water flow rate, m^3/sec

$w_{1,2,3,4}$: weighting factor

X_n : number average chain length

X_w : weight-average chain length of polymer MWD

x_{fi} : filtered value at i-th period

x_G : Gaussian (Normal) distributed variable with mean of zero and variance of 1

x_i : process measurement at i-th period

x_{spc} : CUSUM filtered value of x_i

y : output variable

$\bar{y}$: filtered measured value of y

GREEKS

ϵ : mixture volume expansivity

ϵ_m : error or monomer mass between measured and predicted, kgmole

ϵ_r : error on reactor temperature between measured and predicted, °C

ϵ_j : error on jacket temperature between measured and predicted, °C

λ_o : sum of moles of growing polymer species, kgmole

λ_s : latent heat of steam, KJ/kg

λ_i : i^{th} moment of growing polymer

μ_i : i^{th} moment of dead polymer

μ_w : wall viscosity, cp

ρ_w : density of water, kg/m^3

λ_k : k^{th} moment about the origin
k^{th} moment about the origin for growing polymer

μ_k : k^{th} moment about the origin
k^{th} moment about the origin for dead polymer

β_k : k^{th} moment of the normalized distribution about the origin

σ_k : k^{th} moment about the mean

σ_N^2 : number variance of MWD

σ_w^2 : weight variance of MWD

σ_i : standard deviation at i-th time period

μ : population mean

τ : time interval required to bring current process variable to setpoint, sec

τ_f : T*N

ϕ : adjustable model parameter

ϕ_p : volume fraction of polymer

ACRONYM

AMW : average molecular weight

AR : auto regressive

CUSUM : cummulative summation

CWR : chilled water return

CWS : chilled water supply

EWMA : exponentially weighted moving average

KU : kurtosis of MWD

KU_N : number kurtosis

KU_w : weight kurtosis

MPS : middle pressure steam

MV : manipulated variable

MWD : molecular weight distribution

MW	: molecular weight, kg/kgmole
MWD	: molecular weight distribution
NAMW	: number average molecular weight
NCLD	: number chain length distribution
PDI_n	: number-polydispersity index
PDI_w	: weight-polydispersity index
PDI	: polydispersity index
PID	: proportional-integral-derivative
PV	: process variable
SD	: standard deviation
SD_n	: number standard deviation
SD_w	: weight standard deviation
SPC	: statistical process control
SK	: skewedness of MWD
SK_n	: number skewedness
SK_w	: weight skewedness
TT97%	: time to 97% conversion
WAMW	: weight average molecular weight
WCLD	: weight chain length distribution

CHAPTER I

INTRODUCTION

Polymers (Ray and Laurence, 1977) have been synthesized in the laboratory for more than 150 years. When the commercial importance of polymers was realized in the early twentieth century, production began by scale-up from successful laboratory experiments. Because so little was understood about the polymerization step and the product was so difficult to characterize, a successful polymerization process was one which could reproduce the same recipe for every batch, thus producing reasonable product uniformity from batch to batch.

As might be expected, even minor modifications of the recipe produced a different product, so that a great plethora of product lines, each of small market volume, began to evolve. As the total demand for polymer products has vastly increased over the last 40 to 50 years, the method of production became more and more expensive. Thus in the last 20 years, polymer manufacturers have been working to improve the quality of their products and the efficiency of their operation through efforts: (1) to improve their ability to characterize the chemical makeup and corresponding physical properties of the various polymer products; (2) to quantitatively understand the influence of the reaction conditions on the polymer produced; and (3)

where necessary, to develop more effective reactors. These efforts have required a much better understanding of polymerization kinetics and optimum recipes. The scope of this work is not to investigate a new fact about the polymerization reaction or its physical properties or aspects of reactor design, but to effectively utilize some known facts to guide the course of a polymerization processes. Therefore, our objective is to find out the best process recipes, typically feed rate (i.e., monomer, solvent, initiator) and temperature profile, and effectively maintain them during the batch time. Consequently, it requires a process control and optimization technique.

As stated by Ray and Laurence (1977), a key polymer property that directly influences the end-use characteristic of the polymer is the molecular weight distribution (MWD). The reactor engineer, given a particular monomer, has little control over those effects caused by the nature of the monomer. However, the morphology can often be controlled by the choice of polymerization method, and, in particular, one can often manipulate the MWD through the judicious choice of polymerization conditions. Therefore, there is interest in being able to model the MWD and its moments. Polymer molecules don't all have the same chain length. The MWD can be thought of as a histogram of the amount of polymer within a chain length category versus the chain length. "Moments" are the standard descriptive statistics of average,

variance, skewness, etc. In the past, MWD control has been limited to controlling certain average properties of the distribution. Control objectives such as obtaining a particular number-average molecular weight or minimizing the spread in the distribution have typically been used. However, control algorithms based on average MWD characteristics which are obtained by the leading moments of MWD are likely to be inaccurate for bimodal and highly skewed distributions. This is unfortunate since bimodal distributions can commonly occur due to phenomena such as the gel effect in free-radical polymerization (explained in Section 3.1). Balke and Hamielec (1973) and Bogunjoko and Brooks (1983) have reported experimental observations of bimodal distributions in methyl methacrylate polymerization. There may be cases where it is desirable to control or optimize a polymerization process to achieve a multimodal or skewed distribution. A reliable control or optimization algorithm is not possible unless we can model the formation of these types of distributions.

As a basis of this study, a standard rigorous free-radical solution polymerization reaction in an industrial scale batch reactor is chosen, and given the kinetic information (Chiu et al., 1983, Baillagou and Soong, 1985), a realistic reactor simulator is developed. More details on the reactor and its peripheral are covered in Chapter III. In this work, the mathematical model of the molecular weight

distribution plays an important role in the simulator because it provides information about the dynamics of state variables, average chain length and polydispersity or the chain length distribution. It includes a standard modeling technique, the "method of moments" (Cantow, 1967; Biesenberger, 1983), and a new molecular weight distribution modeling procedure called the "class method."

The method of moments is a standard approach which offers computational simplicity, and it has been useful to describe the characteristics of the molecular weight distribution by calculating the average chain length and polydispersity index. It could not, however, describe how the actual molecular weight distribution (MWD) looks. Actually, there are an infinite number of possible molecular weight distributions which can be described by the leading parameters calculated by the method of moments. In addition to this drawback, this method can not be applied for the case which is accounting for the chain-length dependent termination rate constant because $k_t(n,m)$ introduces severe nonlinearities into the equations. As a solution to these problems, Zeman and Amundson (1963, 1965), Coyle et al. (1985) and Taylor et al. (1986) used the continuous variable approximation method. If the average chain length of the polymer is long, then it is possible to approximate the differential equations by a single partial differential equation in the continuous variable n, the chain length. Ray

(1972) states that this approximate continuous variable representation becomes increasingly better as n increases and the accuracy is very strongly dependent on the number of terms retained in the Taylor expansion for the difference term ($P_{n-1}-P_n$) and on the average chain length of the polymer.

The formulation of the kinetic equation using the continuous variable approximation is relatively simple. However, the resulting kinetic equation, the partial differential equation, no longer retains the original forms of the mechanistical species balance equations, and the solution method is quite complicated especially when the chain-length dependent termination rate constant is applied. As an alternative to the continuous variable approximation method, the class method is developed. Unlike the method of moments or the continuous variable approximation method, it retains the original forms of the kinetic differential equations and possesses a capability of handling the chain-length dependent termination rate constant without complicating the resulting kinetic equations. For this work, however, a chain-length independent termination rate constant is used. Here the entire growing polymer species are distributed into corresponding classes, and all species in each class are approximated as having the mid-chain-length of the corresponding class. This approach reduces the number of differential equations to be solved by several

orders of magnitude, yet it gives a reasonably accurate MWD as long as one discretizes the classes sufficiently fine. The simulator is programmed to have a capability of calculating the dynamics of the state variables, not only the moments but also the density function of MWD. Chapter IV covers these two MWD modeling methods in depth.

Batch or semi-batch reactors are still widely used to produce fine chemicals or pharmaceuticals due to production flexibility and similarity in principle with the laboratory scale reactor. To increase product quality and ensure reproducibility, it is necessary to improve automation of such processes. Having a proper temperature control technique for the batch reactor is of primary importance to this issue so that the process can track an optimum operation recipe (temperature trajectory). Here, to provide the optimum operation recipe, an off-line optimization technique also needs to be developed.

The temperature control of an exothermic batch reactor still remains as a challenge to researchers because of highly nonlinear and nonstationary process behavior. A dual-mode type control (Shinskey and Weinstein, 1965) which employs a standard PID algorithm is only locally valid. When the operating range is changed, the dual-mode should be reset. Most model-based control strategies, especially adaptive predictive control (Kipparisides and Shah, 1983; Cluett et al., 1983; Rafalimanana, 1992; etc.) developed so

far are computationally expensive and show difficulties in parameter estimation when the transition from the initial heat-up to the temperature maintaining stage is reached, and their strategies are not tested for real scale reactors. Their studies should have been extended for a commercial scale reactor control which is represented by a sluggish process response which could act against the controller performance.

The Nonlinear Process Model-Based Control (NPMBC) is chosen as a controller because it has been proven to be the most suitable for highly nonlinear and nonstationary process as experimentally demonstrated and published in the literature (Rhinehart et al., 1988; Rhinehart and Choi, 1988; Rhinehart, 1990; Rhinehart and Riggs, 1991; Rhinehart et al., 1988; Riggs et al., 1991; Mahuli et al., 1992,1993; Pandit and Rhinehart, 1992; Paruchuri and Rhinehart, 1993, 1994; Zhao et al., 1994; Friedman, 1994; Subawalla and Rhinehart, 1994; Natarajan and Rhinehart, 1994). This research started out by developing an NPMBC control strategy and applying it to a pilot scale emulsion batch polymerization reactor which is presented in the AIChE Industrial Challenge Problem (Chylla and Haase, 1991). An idealized polymerization process was used as a process simulator, and Nonlinear Process-Model-Based Control (NPMBC) was employed to control the reactor temperature.

While PID and linear model-based controllers essentially ignore material and energy balance information, NPMBC strategies use this information directly. In NPMBC, the process is controlled by using models based on, but not exactly the same as the process. To match to the process, the models are being updated by Incremental Model Parameterization On-Line (IMPOL), a technique which is proposed by Rhinehart and Riggs (1991). By comparison to linear and PID-type approaches, the NPMBC action is "intelligent" and has shown benefits in uniformity, disturbance rejection, and setpoint tracking. All of which translate into better process economics. It has been shown that NPMBC is an effective way to control the temperature even when the process is subject to noise and disturbances.

While the challenge problem expresses the severe control difficulties of process nonlinearity, it is relatively simple. Solving it does not create an industrially useful controller. It does, however, point the way. The same control strategy is applied for the temperature control of an industrial scale batch reactor. The single control strategy worked from the initial heat-up to the setpoint temperature maintaining stage. The temperature controllers for both the industrial scale reactor and the S.C. Johnson & Sons reactor are developed.

Operating factors, like temperature, recipe loading and its schedule, influence the MWD and operating cost. These

operating factors are decision variables in optimization, and we want to find their values which meet the product specification as closely as possible and give minimum operating cost. The conventional approache tries to obtain the optimum temperature trajectory directly from the kinetic model (Sacks et al., 1972; Chen and Lin, 1980). It essentially ignores the constraints arising from the system operation, and is mostly applicable for simple reaction systems.

The off-line optimization method proposed here searches for the decision variables which optimize batch quality/economic factors and which do not violate the controllability constraint when implemented. For implementation, the entire batch time is divided into several stages according to the monomer conversion. The optimizer searches for the best recipe (the reaction temperature trajectory and feeding schedule) of the corresponding stages in order to meet product and operational objectives as closely as possible. The technique uses a direct searching method, called the complex method (Reklaitis et al, 1983; Gill et al., 1981) which applies the same algorithm as the Nelder-Mead Simplex (unconstrained optimization) technique (Nelder and Mead, 1965; Reklaitis et al., 1983). It searches, however, the optimum point within the bounds of decision variables. It requires only the function value.

The functional evaluation is done by running the reactor simulator on the optimizer's recipe. Since the process simulator can generate the dynamics of the state variables, both the moments and the density function of MWD, one can have the optimizer search the optimum decision variables for either a specified average MW or a specified MWD itself. Since the temperature is being controlled while the simulator is running, one can monitor the temperature controllability which is defined by the current ratio of the heat generated to the maximum heat transferable. It is normally considered uncontrollable if the ratio is greater than 0.5. If the current set of decision variables lead to the temperature uncontrollability, the simulator stops running and returns a reasonably big penalty value to the optimizer so that it keeps the optimizer from searching in that direction again. The optimization results show how the optimum values on the decision variables vary with choices of product and operation target.

CHAPTER II

LITERATURE REVIEW

2.1 Fundamentals of Polymerization Kinetics and Polymers

Polymers are chemical aggregates assembled by the combination of a large number of small molecules called monomers, while the action of the chemical reactions through which the monomers are transformed to polymers is termed polymerization. In a given polymer molecule there may be more than several thousand monomer elements. Synthetic polymers usually have average molecular weights of $2*10^3$ to $1*10^5$. Most useful synthetic polymers have average molecular weight in this range. Polymerization mechanisms are divided into two fundamental categories as discussed in Lenz(1968) and Rodriguez(1989):

1. step growth polymerization, and
2. chain growth polymerization.

Although this classification may not be universal, it has found a rather large acceptance.

The primary characteristics of step growth polymerization is that any two species in a reaction mixture can react with each other. To illustrate this, let us consider the synthesis of a polyester from a diol and diacid. The first step in the reaction is the reaction of the diol and diacid monomers to form a dimer,

$$A + B \text{ -----> } AB. \qquad (2.1)$$

A second step is the reaction of the dimer with another dimer to form a tetramer,

$$AB + AB \text{ -----> } ABAB. \qquad (2.2)$$

or of the dimer with a monomer to form

$$AB + A \text{ -----> } ABA. \qquad (2.3)$$

$$AB + B \text{ -----> } BAB. \qquad (2.4)$$

This can be readily generalized to

$$P_n + P_j \text{ -----> } P_{n+j}, \qquad (2.5)$$

where P_n represents a growing chain with n monomer units. Thus, the chain length is built-up in steps by reactions between any two growing chains. With this mechanism the molecular weight of the reacting mass does not increase significantly until nearly all the monomer has reacted. This requires extremely high conversions of monomer in order to produce a polymer with a high average molecular weight.

In contrast to step growth polymerization, chain growth polymerization proceeds by adding one monomer molecule at a time to a growing chain until the monomer is exhausted or the chain growth is terminated. One common type of chain growth polymerization occurs through a free-radical mechanism of the form:

1. Initiation.

$$I \text{ -----> } 2R_o^{\cdot} . \qquad (2.6)$$

$$R_o^{\cdot} + M \text{ -----> } P_1^{\cdot} . \qquad (2.7)$$

2. Propagation.

$$P_j^{\cdot} + M \text{ -----> } P_{j+1}^{\cdot} . \qquad (2.8)$$

3. Termination.

$$P_j^{\cdot} + P_k^{\cdot} \begin{cases} \dashrightarrow D_{j+k} \cdot \\ \dashrightarrow D_j + D_k \cdot \end{cases} \qquad (2.9)$$

where P_j represents a growing free-radical chain of length j, D_j is an inactive chain of length j, and M is the monomer. Because only a few individual chains are initiated at a given time and these grow to the desired length very quickly before termination, this type of polymerization can have significant amounts of high-molecular-weight material at relatively low conversion. It conversely implies that, by maintaining high reaction temperature, one can convert the initiator to the initiator radical rather faster and keep the average molecular weight low.

Species present during the polymerization can act as chain transfer agents. Solvent and monomer are involved. They enter the propagation step by giving up a hydrogen atom to the growing polymer to terminate it, while initiating a new chain:

4. Chain Transfer to Monomer.

$$P_n^{\cdot} + M \dashrightarrow D_n + P_1^{\cdot}. \qquad (2.10)$$

5. Chain Transfer to Solvent.

$$P_n^{\cdot} + S \dashrightarrow D_n + S^{\cdot}. \qquad (2.11)$$

Usually steps 4 and 5 keep the involved polymer chains from growing further, and consequently decrease the average chain length. Since the existence of solvent diminishes the

number of molecules of the growing polymer species, it causes the monomer conversion rate to become sluggish.

Chain polymerization can be obtained not only by the free-radical method, but also by the various ionic methods (i.e., cationic, anionic) depending on the reactivity of monomers to polymerization and initiator employed. More details on these topics are found in Rodriguez(1989).

In this dissertation, we focused on polymerization which uses the free radical chain growth mechanism.

2.2 Polymer Characterization

A great many useful and interesting material properties, uniquely associated with polymeric materials, are a consequence of their average molecular weight and MWD. For example, we note that the tensile strength of a polymer can increase quite rapidly with molecular weight, and for many materials the processing characteristics depend critically on the molecular weight distribution. For instance, a low molecular weight fraction may plasticize the material and facilitate shaping by extrusion. In addition there are a number of other properties which are strongly influenced by the MWD.

The distribution of molecular weights in a polymeric material may be represented as a density function or "differential" distribution. For example, P_n, the moles of polymer of chain length n, is shown as a discrete

differential distribution in Figure 2.1. Although there are no values of P_n except for integer values of n, a smooth curve results from the numerous points (often more than 10^5) on the graph. Thus we shall use a continuous graph to represent the differential distributions in our discussions.

There are a number of differential distributions which fall under the generic title "molecular-weight distributions." The plot of P_n versus n shown in Figure 2.1 is more precisely the "number chain length distribution" (NCLD), because it represents the chain length distribution with respect to the number of molecules. Similarly, the "weight chain length distribution" (WCLD), shown in Figure 2.2, represents the chain length distribution with respect to the weight of the molecules, $MW_n nP_n$. The NCLD reflects the predominant number of small chain-length species while the WCLD reflects the importance of the few large chain-length species. Thus, the average value in Figure 2.1 is shifted to the right in Figure 2.2.

Let us consider the very simple mechanism which might arise in anionic chain polymerization with very rapid initiation

$$P_n + M_1 \text{ -------> } P_{n+1} \qquad n=1,2,... \qquad (2.12)$$

where P_n represents the growing polymer chain and M_1 is the monomer. The polymer material balances for a reactor yield the equations

$$dP_1/dt = - k_p M P_1 \qquad P_1(0)=P_{1o} \qquad (2.13)$$

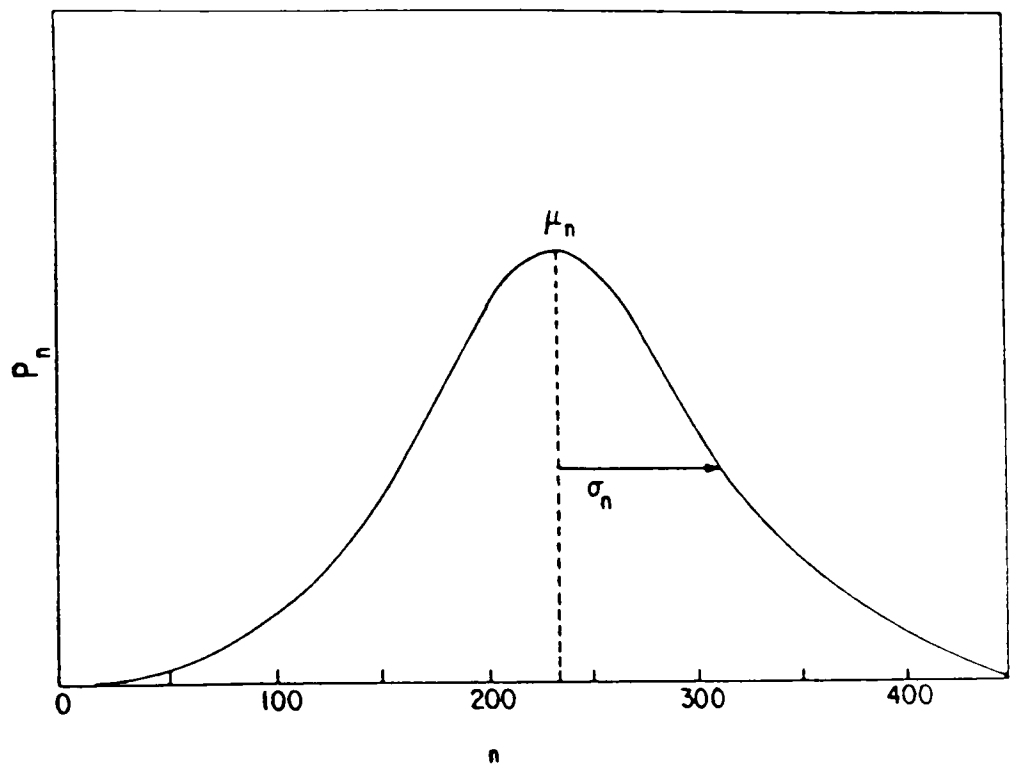


Figure 2.1
Number Chain Length Distribution

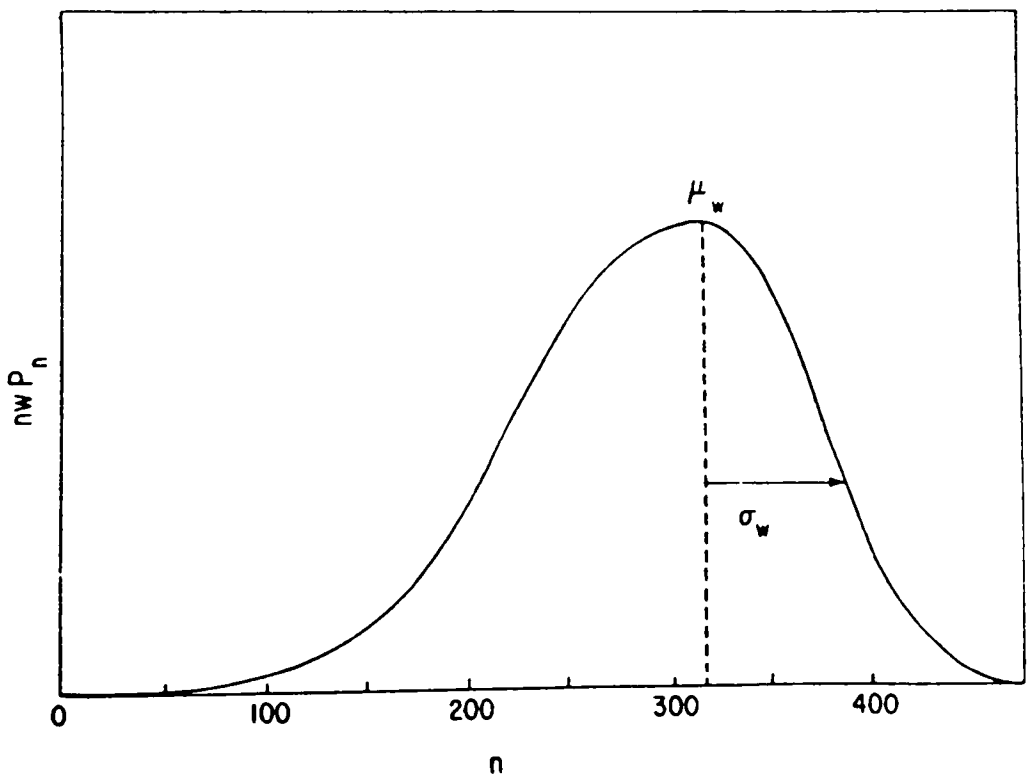


Figure 2.2
Weight Chain Length Distribution

$$dP_n/dt = - k_p M (P_n - P_{n-1}) \qquad P_n(0)=0 \qquad (2.14)$$

with the additional monomer balance

$$dM/dt = - k_p M \Sigma_{\text{for all } n} P_n \qquad M(0)=M_1 \ . \qquad (2.15)$$

The kinetic scheme is very simple, but has the essential features necessary to illustrate the techniques we shall be discussing.

One very obvious approach to obtain the MWD is to numerically integrate the modeling equations to produce $P_n(t)$. There have been a number of applications of this approach in a wide variety of polymer system. Liu and Amundson (1961) and Hamielec et al. (1967) applied this approach to the styrene polymerization. However, the computational burden of direct integration can quickly grow to an unmanageable proportion because of the very small integration step size and the number of variables (i.e., 10^5 even more) which need to be tracked.

It is very often difficult to measure the differential molecular weight distribution of a polymer, but relatively easy to measure some of the molecular-weight distribution moments through viscosity for instance. Therefore, these moments are often used to characterize the polymer.

Ray (1972) and Shinnar and Katz (1972) reviewed some important mathematical modeling techniques for polymerization, i.e., generating functions, the continuous variable approximation, the statistical method, and the

method of moments. In this work, we have selected the method of moments as a standard approach for the modeling of MWD. We also introduced a different approach from that which has been used in previous work. That is the class method. Both are thoroughly discussed in Chapter IV.

2.3 Temperature Control of Batch Reactor

Batch processing is quite different from continuous processing. Startups and shutdowns are normal procedures in batch processes. These processes either move from one steady-state to another or never reach steady-state at all. More generally, a wide range of operations are carried out to produce a number of chemicals in a single vessel. By virtue of their volume to heat-transfer area, batch exothermic reactors tend to be unstable, requiring tight temperature control to prevent thermal runaway. Shinskey (1992) pointed out that typically, the batch reactors are characterized as: (1) the heat generated by the reaction changes with setpoint, recipe and conversion; and (2) the reactor temperature response can change with batch size, agitation, viscosity and fouling.

The rate of heat released as the reaction begins can become very large very quickly, and the reaction may become unstable and cause the temperature to "run away" if the heat generated exceeds the cooling capacity of the reactor. For this time varying characteristic, the thermal control of

batch reactors is not easily accomplished by a PID controller, and remains a difficult and up-to-date challenge.

A lot of work about thermal control of batch reactor have been reported (Marroquin and Luyben, 1973; Kiparissides and Shah, 1983; Jutan and Uppal, 1984; Cluett et al., 1985; Takamastu et al., 1986; Cott and Macchietto, 1989; Kravaris et al., 1987; Rostein and Lewin, 1990; Hodgson and Clarke, 1984; Merkle and Lee, 1989; and many more). It has to be noticed that the temperature control of the reactor depends mainly on the heating-cooling system of the reactor. Three different types of heating-cooling systems can be distinguished in these works (Rafalimanana et al., 1992):

(1) the first type for which the thermal control is ensured by changing the inlet temperature of an intermediate fluid flowing inside the jacket which surrounds the reactor.

(2) the second type which consists of a jacketed reactor equipped with an inside coil and for which the temperature control is ensured by acting on both variables: the flow rate of cooling water flowing in the immersed coil and the flow rate of heating fluid flowing inside the jacket. Most often one of these two flowrates is fixed. For example, the hot fluid flowrate is set to a mean value and only the cooling water flow rate is controlled.

(3) the third type called the alternate heating-cooling system which uses alternatively and directly cold or hot utility fluids (i.e., cooling water or steam) available at the plant at given temperatures and for which the temperature control is realized by first choosing the "right" fluid and then by acting on the flowrate of this fluid flowing inside the jacket.

More than 90% of industrial reactors are equipped with the third type, the alternate heating-cooling system. However, most of the studies mentioned above utilized the first type of heating-cooling system (Marroquin and Luyben, 1972; Kiparissides and Shah, 1983; Cluett et al., 1985; Takamatsu et al., 1986; Cott and Macchietto, 1989; Rostein and Lewin, 1990).

Every batch reactor goes through an initial heat-up stage. This stage is quite important because any temperature overshoot could cause temperature runaway and slow heat-up delays the completion of the batch. The strategy commonly used is the on-off type and consist in applying maximum heating until the reactor temperature is within a specified number of degrees of the setpoint and switching to maximum cooling to bring the rate of temperature change to zero when the temperature has reached its final desired set point. At this point, standard feedback controllers can be switched on and used to maintain the temperature. The most commonly used strategy of this type in industry is the dual-mode

controller of Shinskey and Weinstein (1965), which uses a standard PID controller for maintaining temperature. It is further discussed by Liptak (1986). The main problem with approaches of this type is that the optimal switching criterion from heating to cooling, usually based on the reactor temperature, is determined *a priori* and is therefore only valid for a specific range of operating conditions. Because heat-up proceeds in an open-loop manner and no feedback from the reactor is used, there is no allowance for modeling errors or for changes in process parameters. The net result is that these strategies lack robustness, and any deviation in the operating conditions from those used to tune the controller may result in significantly poorer control performance.

Jutan and Uppal (1984) used a combined feedforward-feedback servo control scheme. Their cooling-heating system belongs to the Type 2. The feedforward part inferentially predicts the amount of heat generated using measured reactor and jacket temperatures. The feedback section corrects the control action determined by the feedforward loop. Their controller outperformed PID type controller for a pilot scale reactor simulator, but there is the presence of significant overshoot in the reactor temperature. This effect may be attributed to the linearization necessary to implement the feedback control action.

The use of adaptive control algorithms would appear to offer a promising approach in this direction. A recent paper by Rafalimanana et al. (1992) is typical of such attempts. They used a single adaptive control algorithm based on an ARIMA (Auto-Regressive Integrated Moving Average) type time series model for both heat-up and temperature maintenance. The changes of the process are detected by identifying the model parameters using a typical recursive least square method. The model is quite empirical and selection of model parameters can be very significant to successful control. A very similar approach has been tried for a PVC reactor by Kiparissides and Shah (1983), Cluett et al. (1985) and for a batch evaporative crystallizer by Febotte and Klein (1993). They used a single adaptive control algorithm for both heat-up and temperature maintenance but found that the algorithm did not handle the sharp change from the heat-up mode to the temperature maintenance mode very well. The significant computational burden for parameter estimation dictates that a relatively slow sampling rate be used. Infrequent sampling, however, diminishes disturbance rejection capabilities.

Wright and Edgar (1993) used a nonlinear model predictive control (NMPC) based on a phenomenological model for control of a continuous-flow fixed-bed, water-gas shift reactor. Nonlinear Programming (NLP) was formulated and solved for control horizon (manipulated variables) such that

process-predicted values were minimized just like DMC type controllers. The NLP was solved at each sampling interval. The solution for the NLP is computationally very expensive.

Juba and Hamer (1986) applied Nonlinear Process Model-Based Control (NPMBC) to the control of a batch reactor for a highly exothermic polymerization reaction. Using an energy balance on the reactor and an estimate of the activation energy of the reaction, they formulated a model for the heat generated by the reaction mixture. However, because of the difficulty and complexity of deriving an accurate heat transfer model, the heat removal system was modeled empirically. Their NPMBC equations comprise both phenomenological and empirical model approaches as they sought to use the simplest model that was sufficiently accurate to model the phenomena. They found that a steady-state (asymptotic) model of the process was sufficient. With their controller, they found that they were able to obtain reliable temperature control while PID control experienced temperature runaway. The authors state that the use of time-varying nonlinear process model is a key element in the further advance of batch reactor control.

Cott and Macchietto approached Juba and Hamer's problem by trying to identify the key process parameter, which is Q/UA, from the reactor energy balance. From this information, the jacket temperature setpoint is calculated, and a PID controller is used to control the jacket

temperature. They used the Type 1 cooling-heating system. It worked excellently for a small size reactor. For a commercial scale reactor, the sluggish PID algorithm could lead to a control problem like an integral windup.

Lee and Sullivan (1987) have presented a method for implementing NPMBC that has a proportional type and integral-type action. Their approach is called Generic Model Control (GMC) and is a single-step control law that usually requires the solution of a system of algebraic equations for its implementation. For instance, when Cott and Macchietto calculate the jacket temperature setpoint from the reactor energy balance model, it can be done within the GMC law. Lee and Sullivan show that GMC is similar in many ways to single-loop PI control, feedforward and decoupling control, time horizon matrix controllers, and IMC; but GMC uses a dynamic process model directly without any linearization steps. Since GMC is both functional and simple, GMC will be used for the research proposed here.

2.4 Optimization of a Polymerization Process

Batch processes are often characterized by the frequent repetition of production of small-volume, high value material. Consequently, such processes are required to carefully follow well-defined recipes in order to satisfy certain quality, safety and performance requirements. Optimal operation, i.e., optimal feed addition or

temperature trajectory, is usually determined before start-up based on reaction kinetics alone, typically using the "maximum principle" (Sacks et al., 1972; Chen and Lin, 1980; Thomas and Kiparissides, 1984). These open-loop optimizations are appropriate in situations where the process model is accurately known and no batch-to-batch variations are expected. A comprehensive summary of these works is given in Rippin (1983).

If the model is not accurately known, but no significant batch-to-batch variations are expected, the strategies proposed by Filippi-Bossy and co-workers (1989) can be applied. Both approaches use information from the previous batch to improve the feed or temperature policies for the next run.

When the model is appropriately known but the parameters undergo significant changes from batch to batch, open-loop optimization schemes together with an on-line parameter or state estimator can be used. The problems with this approach are: (1) The standard measurements (flow, temperature, pressure) do not contain the same pertinent information for estimation. Laboratory analysis or additional "on-line" measurements introduce a large time delay compared to the relatively short reaction time. (2) Because of the relatively low volumes produced, the expenses for the development and implementation of on-line optimization is often economically not justified. (3) The

extent to which the final outcome can be influenced (degree of controllability) and thereby the possibility of batch reactor performance improvement decreases as the batch progresses so that corrective action needs to be taken early on. Therefore, the estimator has to converge quickly. (4) Unlike continuous reactors, batch reactors require models with a wider region of validity.

Although on-line optimization appears not easily applicable to batch reactors, increasing quality and environmental requirements call for the development of improved operational strategies. On-line estimation of parameters and states, which represents one of the main steps towards the optimization of batch reactors, has been the subject of several very promising theoretical and practical research studies (Meadows and Rawlings, 1990; Bonvin et al., 1993). They have shown their results for simple chemical reactions so that the on-line estimation can be achieved in a short period of time. It is very doubtful that this idea is applicable for the more complicated polymerization process.

Therefore, it seems that the off-line optimization technique still could be a promising method for the polymerization process. A new type of off-line process optimization is needed to improve the drawback which inherently occurs with the kinetic model dependent optimization technique. Since the trajectory obtained by

such a way essentially ignores constraints arising from the system operation, implementation of the temperature trajectory is affected by manipulation of final control elements. The actual choice of the desired trajectory is therefore also dependent on the physical limitations of the processing unit, a fact that has not been given sufficient attention in the past. Lewin and Lavie (1990) found that an optimum trajectory obtained without considering this aspect could be disastrous even for a slight fouling factor change. They took this aspect into account for determining feasible and safe start-up heating time.

If a rigorous polymerization reactor simulator is available, one can see how the operation recipe (feed, temperature) influences the product specification and the batch time, etc., as we change them every time we run the simulator. After many trials, we can decide the best recipes which meet the product and operation objective as closely as possible, and does not violate the system constraint. This idea can be automated by having a proper optimization technique search for the optimum point.

CHAPTER III

PROCESS DESCRIPTION

Two different polymerization reactor processes are used in this study. The first one is a pilot scale semi-batch reactor which produces a specialty emulsion polymer. The process description is prepared by S.C. Johnson & Sons Company and given as an Industrial Challenge Problem (Chylla and Hasse, 1990), "Temperature Control of a Semibatch Polymerization Reactor." While the challenge problem expresses the severe control difficulties of process nonlinearity, it is relatively simple. The process details are described in the first four sections in Appendix A. The temperature control strategy developed in this study is initially tested for this very ideal process.

The second polymerization process is a commercial scale batch reactor which has the same system configuration as the SC Johnson & Sons reactor, but is different in size, kinetics, and batch recipes. Realistic polymerization reaction kinetics are given so that one can perform the dynamic modeling of the molecular weight distribution. The process optimization is studied for the polymerization reaction in this industrial size reactor. Based upon the rigorous process description for the industrial size reactor, a reactor process simulator is developed. The details are presented in the following section.

3.1 Polymerization Reaction Mechanism

A free-radical solution homopolymerization system described by Chiu et al. (1983) and Baillagou and Soong (1985) is chosen as the process. The typical three step polymerization mechanism, initiation-propagation-termination, as shown in Figure 3.1 is proposed with possible chain transfers. The mechanistical kinetic equations are described in Figure 3.2, and rate constant expression and kinetic parameter data are presented in Figures 3.3 and 3.4.

A problem occurring in most addition polymerizations is the reduction of the free volume of the mixture with conversion. This phenomena, referred to as the gel effect, was discovered by Trommsdorff (1945). The conversion rate is accelerated due to a decrease in the rate of termination. Because the termination steps involve reactions between growing polymers, they are more severely diffusion controlled than the propagation reaction. As the free volume of the polymerizing mixture decreases with conversion, the mobility of the growing polymers is severely reduced, resulting in the onset of the gel effect.

Although the termination rate expression which reflects the chain-length of a growing polymer is desirable for a better representation of the process, it was decided to use chain-length independent termination rate expression as proposed by Chiu et al. and Baillagou and Soong. The same

$I \xrightarrow{k_d} 2R^{\cdot}$	Initiation	(3.1)
$R^{\cdot} + M \xrightarrow{k_i} P_1^{\cdot}$	Initiation	(3.2)
$P^{\cdot}_n + M \xrightarrow{k_p} P^{\cdot}_{n+1}$	Propagation	(3.3)
$P^{\cdot}_n + M \xrightarrow{k_{tm}} D_n + P^{\cdot}_1$	Chain transfer to Monomer	(3.4)
$P^{\cdot}_n + S \xrightarrow{k_{ts}} D_n + S^{\cdot}$	Chain transfer to Solvent	(3.5)
$P^{\cdot}_n + P^{\cdot}_m \xrightarrow{k_{td}} D_n + D_m$	Termination by Disproportionation	(3.6)
$P^{\cdot}_n + P^{\cdot}_m \xrightarrow{k_{tc}} D_{m+n}$	Termination by Combination	(3.7)

where

I	is the initiator concentration.
$R^{\cdot}$	is the primary initiator radical concentration.
M	is the monomer concentration.
S	is the solvent concentration.
$P^{\cdot}_j$	is the growing polymer radical with degree of polymerization j.
D_j	is the dead polymer with degree of polymerization j.

Figure 3.1
Proposed Free-Radical Polymerization Mechanism

For Initiator:

$$\frac{1}{V}\frac{d(IV)}{dt} = -k_d I. \qquad (3.8)$$

$$\frac{1}{V}\frac{d(RV)}{dt} = 2fk_d I - k_i RM. \qquad (3.9)$$

For Growing Polymer:

$$\frac{1}{V}\frac{d(P_1 V)}{dt} = k_i RM - k_p MP_1 + (k_{tm}M + k_{ts}S)(P - P_1) - (k_{tc}+k_{td})P_1 P. \qquad (3.10)$$

$$\frac{1}{V}\frac{d(P_n V)}{dt} = k_p M(P_{n-1} - P_n) - (k_{tm}M + k_{ts}S)P_n - (k_{tc}+k_{td})P_n P. \qquad (3.11)$$

$$\frac{dP}{dt} = -\frac{P}{V}\frac{dV}{dt} + k_i RM - (k_{tc}+k_{td})P^2. \qquad (3.12)$$

For Monomer and Dead polymer:

$$\frac{1}{V}\frac{d(MV)}{dt} = -k_i RM - (k_p + k_{tm})PM. \qquad (3.13)$$

$$\frac{1}{V}\frac{d(D_n V)}{dt} = k_{tm}P_n M + k_{ts}P_n S + k_{td}P_n P + \frac{1}{2}k_{tc}\sum_{m=1}^{n-1} P_m P_{n-m}. \qquad (3.14)$$

For Solvent:

$$\frac{1}{V}\frac{d(SV)}{dt} = -k_{ts}SP. \qquad (3.15)$$

where

V is the system volume. $V = V_0(1+\epsilon x)$, $\epsilon = (d_m - d_p)/d_p$.

$0 \le f \le 1$ is an efficiency factor to account for other simultaneous processes consuming R.

$P = \sum_{n=1}^{\infty} P_n$.

Figure 3.2
Mechanistical Kinetic Equations
for the Proposed Free-Radical Mechanism

Initiation:

$$k_d = k_d^0 \exp(-E_d/RT_r). \quad (3.16)$$

Propagation:

$$k_p = \frac{k_{p0}}{1+\frac{\lambda_0 k_{p0}}{k_{\theta p} D}}. \quad (3.17)$$

$$k_{p0} = k_{p0}^0 \exp(-E_{p0}/RT_r). \quad (3.18)$$

$$k_{\theta p} = k_{\theta p}^0 \exp(-E_{\theta p}/RT_r). \quad (3.19)$$

Chain transfer:

$$k_{tm} = k_{tm}^0 \exp(-E_{tm}/RT_r). \quad (3.26)$$

$$k_{ts} = k_{ts}^0 \exp(-E_{ts}/RT_r). \quad (3.27)$$

Termination:

$$k_t = \frac{k_{t0}}{1+\frac{\lambda_0 k_{t0}}{k_{\theta t} D}}. \quad (3.20)$$

$$k_{t0} = k_{t0}^0 \exp(-E_{t0}/RT_r). \quad (3.21)$$

$$k_{\theta t} = I_0 k_{\theta t}^0 \exp(-E_{\theta t}/RT_r). \quad (3.22)$$

$$k_t = k_{tc} + k_{td}. \quad (3.23)$$

$$k_{tr} = \frac{k_{td}}{k_{tc}}. \quad (3.24)$$

$$k_{tr} = k_{tr}^0 \exp(-E_{tr}/RT_r). \quad (3.25)$$

where

$$D = \exp\left(\frac{2.3(1-\phi_p)}{A+B(1-\phi_p)}\right). \quad (3.28)$$

$$\phi_p = \frac{\frac{\mu_1}{d_p}}{\frac{\mu_1}{d_p}+\frac{M(MW_m)}{d_m}+\frac{S(MW_s)}{d_s}}. \quad (3.30)$$

$$A=0.168-8.21*10^{-6}(T_r-T_{gp})^2. \quad (3.29)$$

B=0.03
T_{gp}=378.2 °K

Figure 3.3
Rate Constant Expressions for
Free Radical Polymerization Model

k_d^0 = 1.0533E15	s^{-1}.	E_d = 1.2877E5	KJ $kgmole^{-1}$.
k_{p0}^0 = 4.9167E5	$m^3\ kgmol^{-1}s^{-1}$.	E_{p0} = 1.8283E4	KJ $kgmole^{-1}$.
$k_{\theta p}^p$ = 3.0233E13	s^{-1}.	$E_{\theta p}$ = 1.1700E5	KJ $kgmole^{-1}$.
k_{tm}^0 = 4.6610E9	$m^3\ kgmol^{-1}s^{-1}$.	E_{tm} = 7.4479E4	KJ $kgmole^{-1}$.
k_{ts}^0 = 4.9658E8	$m^3\ kgmol^{-1}s^{-1}$.	E_{ts} = 6.6163E4	KJ $kgmole^{-1}$.
k_{t0}^0 = 9.8000E7	$m^3\ kgmol^{-1}s^{-1}$.	E_{t0} = 2.9442E3	KJ $kgmole^{-1}$.
$k_{\theta t}^0$ = 1.4540E20	$m^3\ kgmol^{-1}s^{-1}$.	$E_{\theta t}$ = 1.4584E5	KJ $kgmole^{-1}$.
k_{tr}^0 = 2.5280E3		E_{tr} = 1.7178E4	KJ $kgmole^{-1}$.

Figure 3.4
Kinetic Data for
the Free Radical Polymerization Model.

termination rate is applied over the entire growing polymer species regardless of their chain length. The kinetic expressions for propagation and termination, however, accounts for the gel effect at high conversion.

During the study, we encountered a problem with the literature-proposed kinetic expression for k_t, the termination step rate constant because of the limitation that it imposes. The k_t is expressed by a combination of a kinetic limiting term and a mass transfer limiting term. That is:

$$\frac{1}{k_t} = \frac{1}{k_{t_0}} + \frac{1}{k_D}. \qquad (3.31)$$

or

$$k_t = \frac{k_{t_0}}{1+\frac{k_{t_0}}{k_D}}. \qquad (3.32)$$

where k_D, the mass transfer rate constant, is supposed to be a function of polymer molecular weight, diffusivity and temperature. Rewriting Eq.(3.32) per the Chiu et al. reference gives

$$k_t = \frac{k_{t_0}}{1+\frac{\lambda_0 k_{t_0}}{k_{\theta_t} D}}. \qquad (3.33)$$

where k_D is expressed in terms of the diffusivity of the batch contents. The diffusivity is then expressed in terms of polymer volume fraction and temperature. This seems reasonable. However, the molecular weight dependence of k_D

is expressed as a function of I_o, initiator mass at the beginning :

$$k_{\theta_t} = I_0 k_{\theta_{t_0}} \exp\left(-\frac{E_{\theta_t}}{RT_r}\right). \qquad (3.34)$$

This does not seem fundamentally correct. How does the initial initiator concentration affect mass transfer limited polymerization reaction rate into the run? We believe that Eq.(3.34) is an attempt to account for polymer molecular weight on the mass transfer limit to the reaction rate. Increasing molecular weight increases viscosity and thereby decreases the reaction rate. Given a single batch recipe, the greater I_o is, the more polymer chains are simultaneously started and the lower the average molecular weight will be at any value of conversion. It appears that the originator intended to account for the true molecular weight mechanism, indirectly, by correlating k_t to I_o. However, this must be incorrect. Consider seeding a batch of monomer with a few growing polymer free radicals but with zero initiator present. Eq.(3.33) suggests that the termination rate will be zero. Eq.(3.34) is quite empirical and lacks a mechanistical justification. In addition, it does not even account for the initial mass of solvent, and for the case where the initiator is being fed in during the run. Simply by Eq.(3.34), one cannot describe the molecular weight dependence of k_p.

Since developing a new realistic expression for k_D is not our objective, we will continue to use the given kinetic expression to explore control strategies. However, we need a value for k_t. Currently we are using a reasonable fixed value for the I_o term in Eq.(3.34). This means that our analysis will not consider the molecular weight dependence on k_D. While the average molecular weight is dependent on the initial mass of the initiator, k_t seems negligibly dependent on the initial mass of initiator. Some related results are found in Section 7.1 of Chapter VII, which support our current strategy of neglecting the molecular weight dependence of k_D, and proceeding with control and optimization studies.

The physical properties, heat transfer coefficient correlation, reactor and jacket side energy balances are presented in Figure 3.5. The correlations for wall viscosity and film heat transfer coefficient are adopted from the S.C. Johnson & Sons reactor problem statement (Appendix A) and reasonably modified to fit this reactor system.

The differential equations shown in Figures 3.1 to 3.5 constitute the mathematical model of the reactor. Solved simultaneously, they give us the monomer conversion, MWD and both reactor and jacket temperatures. These equations are used to simulate the dynamics of the polymerization reactor. This is a typical nonlinear and nonstationary process.

d_i = 915 kg m^{-3}. MW_i = 164.21 kg $kgmole^{-1}$.
d_m = 889.5 kg m^{-3}. MW_m = 100.12 kg $kgmole^{-1}$.
d_s = 820.0 kg m^{-3}. MW_s = 90.14 kg $kgmole^{-1}$.

C_i = 1.3 KJ/Kg/K.
C_m = 2.2 KJ/Kg/K.
C_s = 3.1 KJ/Kg/K.
C_p = 2.2 KJ/kg/K.

$-\Delta H_p$ = 75,140 KJ $kgmole^{-1}$.

$$\mu_w = 5.0\exp(8f_p)6.0^{[5.0(\alpha 1-1.7)]}. \qquad (3.35)$$

where $\alpha 1=500/(T_w+273)$.
$T_w=(T_r+T_j)/2.0$.
f_p=polymer mass fraction.

$$h_i = 0.8\exp(-0.03\mu_w). \qquad (3.36)$$

h_o = infinite.

The reactor side energy balance equation:

$$\frac{d(\sum m_i Cp_i T_r)}{dt} = (\dot{m}_M Cp_M + \dot{m}_S Cp_S + \dot{m}_I Cp_I)(T_{amb} - T_r) + k_p \lambda_0 n_M (-\Delta H_p)$$

$$-\frac{1}{1/h_i+1/h_f}\cdot A\cdot(T_r-T_j) - (UA_{loss})(T_r-T_{amb}). \qquad (3.37)$$

The jacket side energy balance equation:

$$(V_j \rho_{w,j} Cp_{w,j})\frac{dT_j}{dt} = \dot{v}_w \rho_w C_{p,w}(T_w - T_j) + \dot{m}_s(\lambda_s + C_{p,w}(T^a - T_j)) +$$

$$- \dot{m}_s C_{p,w} T_j + \frac{1}{1/h_i+1/h_f} A(T_r - T_j). \qquad (3.38)$$

Figure 3.5
Physical Parameters and Energy Balance
for the Free Radical Polymerization Model

As proposed in the literature the dimensional units of recipe variables (i.e., I, S, λ_k, μ_k) in Figures 3.2 and 3.3 are molar concentration, kgmole/m³. However, we find that it has been easier to describe the variables if one lets them have kgmole units instead of kgmole/m³, especially considering a semi-batch operation case in which materials are being fed during the operation. The following two modifications need to be made to work with the kgmole unit:

1. Divide the rate constants, k_t, k_p, k_{ts}, k_{tm} by the system volume (m³).

2. Modify the rate constant expressions for k_p, k_t to be:

$$k_t = \frac{k_{t_0}}{1+\frac{\lambda_0 k_{t_0}}{k_{\theta_t} DV_{sys}}}. \qquad (3.39)$$

$$k_p = \frac{k_{p_0}}{1+\frac{\lambda_0 k_{p_0}}{k_{\theta_p} DV_{sys}}}. \qquad (3.40)$$

3.2 Reactor Schematic

The reactor schematic and sensor-control element are shown in Figure 3.6. A stirred tank reactor with a working volume of 15 m³(529.7 ft³) is used to make a solution polymer in a batch process. Sufficient agitation is provided such that the reactor can be assumed to be well-mixed. However, mechanical energy generated by the agitator is neglected. The heating-cooling system of this reactor

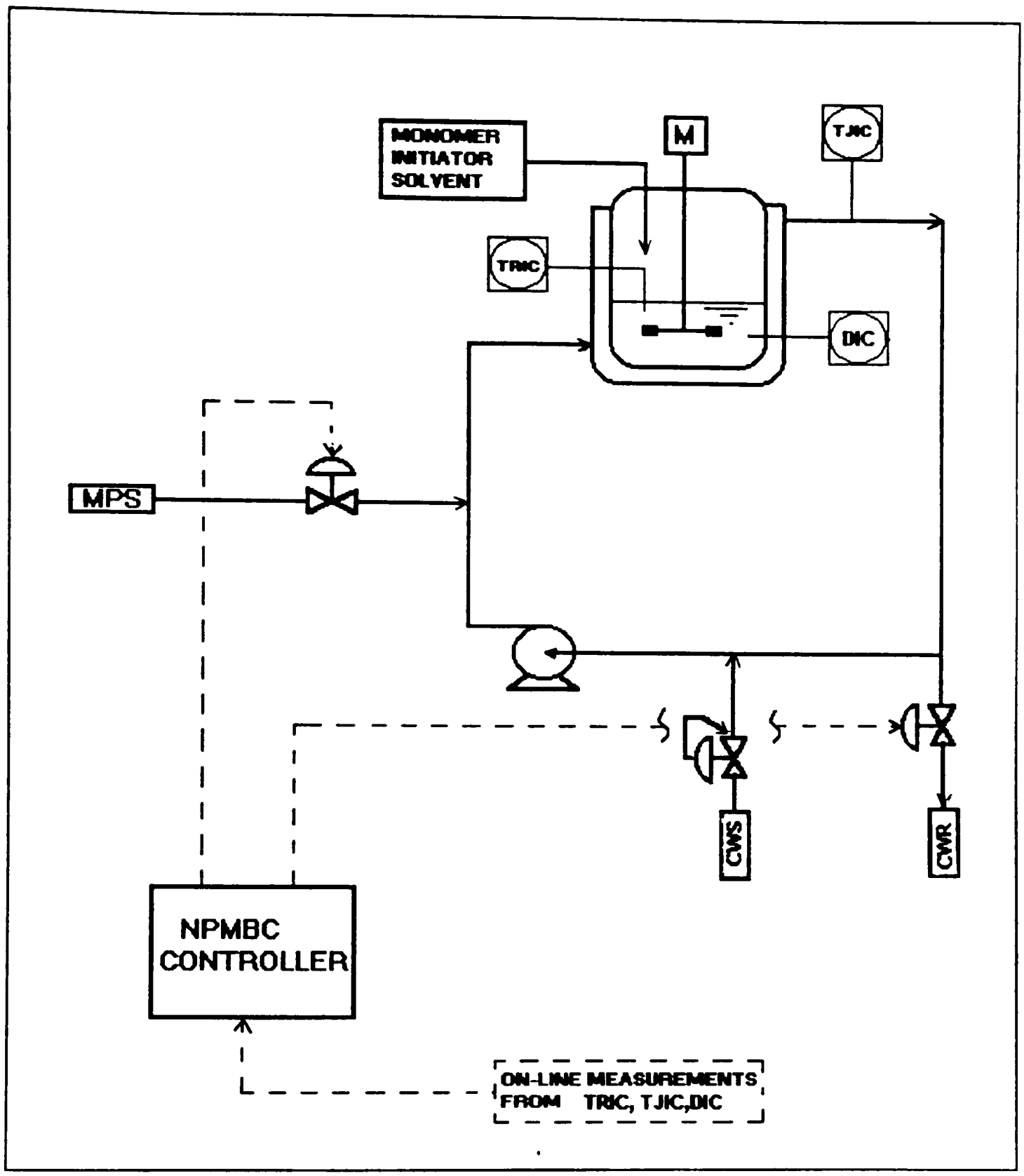


Figure 3.6
Reactor Schematic and Control Element
(Industrial Scale Reactor)

belongs to the Type 3 as described in Section 2.4 of Chapter II. The reactor jacket has a volume of 2.3 m^3(81.2 ft^3) and special internals to facilitate a very high jacket side heat transfer coefficient. The recirculation line volume is assumed to contain 1.5 m^3(53 ft^3) for a total volume of 3.8 m^3(134.2 ft^3) in the recirculation loop. The reactor and recirculation loop data are given in Table 3.1. The three measurable data in this process are the reactor temperature, jacket temperature and mixture density. The density can be readily translated into the conversion by means of empirical relations so that the unreacted amount of monomer in the reactor becomes measurable. The geometrical data for the reactor do not represent any one specific reactor system, but are chosen to mimic industrial practice as closely as possible.

3.3 Process Noise and Disturbances

Process data are subject to variations for several reasons. Accordingly, measured process values respond with an inherent variability even when the manipulated process inputs are unchanged. These process variations may be classified as noise and disturbances. Sources of noise are: (1) Electric interference on transmission lines due to static electricity, sparks, motors, etc. (2) Real but short lived process events due to flow turbulence on a differential-pressure cell, nonideal mixing, etc. Sources of

Table 3.1
Reactor and Recirculation Loop Data

DESCRIPTION	UNITS	VALUE
Reactor Diameter	m	2.5
Reactor Height	m	3.0
Jacket Volume	m^3	2.3
Circulation line volume	m^3	1.5
Steam (saturated) pressure	barg	10.0
Jacket side pressure	barg	4.0
Chilled water temperature	°C	5.0
Ambient air temperature	°C	20.0
CW valve $Cv_{selected}$	-	120
Steam valve $Cv_{selected}$	-	30
Reactor fouling factor	m^2°C/kw	0.1
environmental heat loss factor, $(UA)_{loss}$	kw/°F	0.2

disturbances are: (1) Real process influences due to environmental effects (ambient conditions, heat losses, inlet temperatures, %RH of intake air, etc.), (2) Real process influences due to up-stream changes (composition drifts due to a controller bringing composition back to setpoint after an upset, inlet flow rate changes in response to recycle of waste etc.), (3) Apparent process changes due to sensor/transmitter drifts (moisture build-up in one side of a dp cell, ambient temperature caused calibration drift, fouling on a sensor window, static electricity, etc.).

Noise is a short-term interference which is added to the time-averaged process signal. "Short-term" meaning that the noise at each sampling is independent of previous noise. Process simulators based on the solution of differential equations generate time-averaged (deterministic) outputs. Since process knowledge is gained by measurements which are noisy, any measurable simulated output should have noise added to it.

Disturbances are auto-correlated: the previous value influences the current value. If a cloud covered the sun and cast a shadow on a distillation tower which increased the ambient heat losses at the last sampling, heat losses will likely be high this sampling.

Figure 3.7 describes how noise and disturbances are included in simulators. The disturbed process outputs reflect disturbances, and are used as inputs for downstream

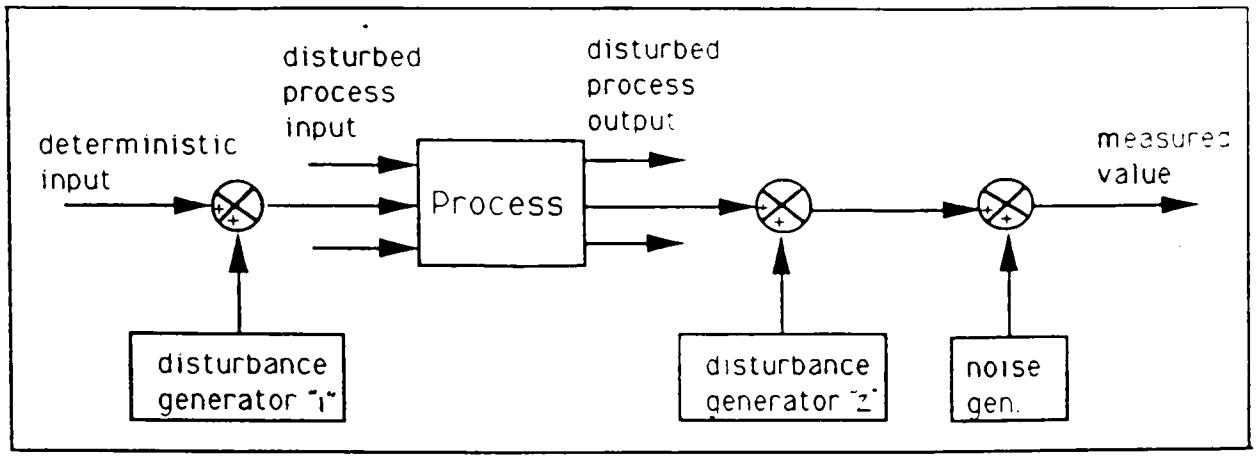


Figure 3.7
Schematic of Noise and Disturbance

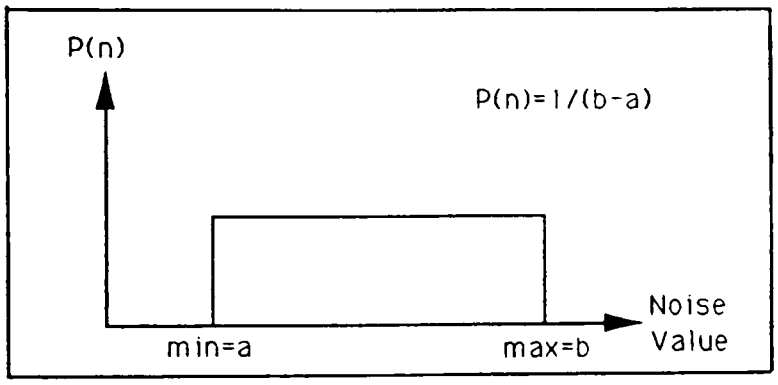


Figure 3.8
Uniform Distributed Noise

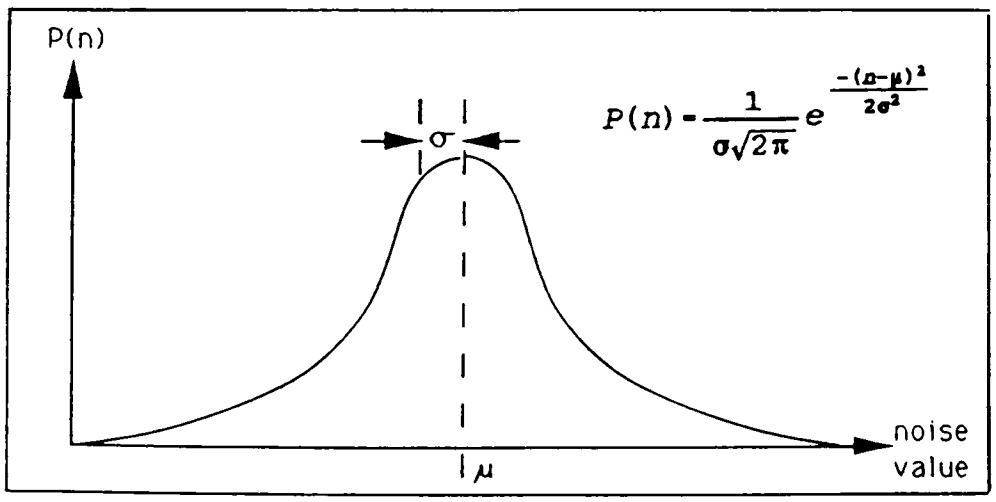


Figure 3.9
Gaussian (Normal) Distributed Noise

processes. The disturbed process inputs can either be disturbed process outputs from upstream or deterministic inputs with an added disturbances. For the polymerization reactor case, disturbance generator "1" includes initiator reactivity and heat transfer fouling. Disturbance generator "2" includes calibration drifts on all measurements including material inputs. The following two sub-sections describe how to simulate noise and disturbances.

3.3.1 Noise Generation

If there is a single random event causing noise, noise will be a uniformly distributed random variable (Figure 3.8). If there are many, small, random, independent influences, noise will be a normally or Gaussian (Normal) distributed random variable (Figure 3.9). Either situation is ideal. But, the Gaussian idealizations come close to approximating real noise. To generate Gaussian distributed noise, one can use (Knuth, 1981):

$$x_G = (-2 \cdot \ln(r_1))^{1/2} \sin(2\pi \cdot r_2). \qquad (3.41)$$

where r_1 and r_2 are uniformly distributed random numbers with $0 < r_1, r_2 \leq 1$. x_G is Gaussian distributed with mean of zero and variance of 1.

To generate any specific Gaussian noise of each measuring component, its standard deviation shall be multiplied by x_G.

$$noise = \sigma \cdot x_G. \qquad (3.42)$$

For the S.C. Johnson & Sons reactor simulation, the standard deviation of each measurement is chosen as follows:

σ = (1% of actual value)/5.0, for monomer mass

σ = 0.2 °F, for temperature

σ = (3 % of actual value)/5.0, for steam or dump water flow rate

σ = 0.005 lb/min, for feed flow rate.

Figure 3.10 shows a typical noise sequence for temperature generated by the S.C. Johnson & Sons reactor simulator. For the industrial reactor simulation, the standard deviation of each measurement is chosen as follows:

σ = 0.00002(moles of monomer)^2/5.0, for monomer mass.

σ = 0.2 °C, for temperature.

σ = 0.005(flow rate, kg/sec)^2/5.0, for steam flow rate.

σ = 0.00005(flow rate, m^3/sec)^2 /5.0, for CW flow rate.

σ = 0.00007(flow rate, kg/min)^2 /5.0, for solvent flow rate.

σ = 0.00004(flow rate, kg/min)^2 /5.0, for initiator flow rate.

3.3.2 Disturbance Generation

A moving average (MA) (Actually a weighted sum of the past k noise values) is defined as

$$q_i = n_i + b_1 n_{i-1} + b_2 n_{i-2} + \ldots + b_k n_{i-k}. \qquad (3.43)$$

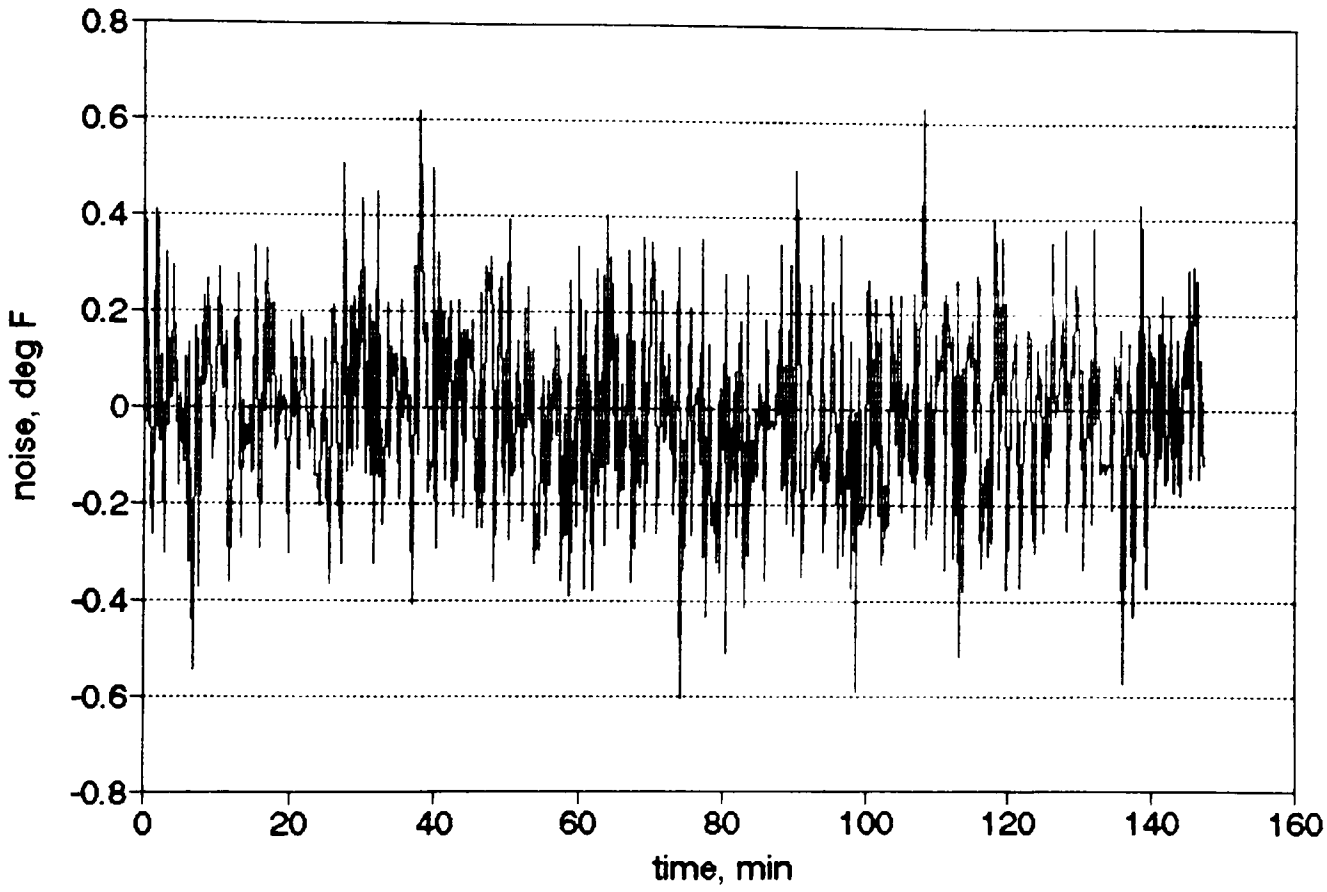


Figure 3.10
Noise on Temperature Measurement
noise=$\sigma[(-2\ln(r_1))^{0.5}\ \sin(2\pi r_2)]$
where $r_{1,2}$=independent, random, uniform
$0<r_{1,2}\leq 1$
σ=0.2 °F

Eq.(3.43) indicates that the influence of past values of an event persists up to k samplings. Usually, for simulation, $k \leq 3$, rarely is $k > 5$. Since the influence of an event should decay in time, usually $b_k < b_{k-1} < \ldots < b_2 < b_1$. All b's should be positive.

Autoregressive (AR) behavior means that the rate of change of a variable depends on the previous value of the variable and is driven by noise. Mechanistically this can be described as

$$\tau_p \frac{dq}{dt} + q = k \cdot n. \tag{3.44}$$

Eq.(3.44) represents a first-order process with noise as a forcing function. When it is discretized

$$\tau_p \frac{q_i - q_{i-1}}{T} + q_i = k \cdot n_i. \tag{3.45}$$

and rearranged

$$q_i = a_1 \cdot q_{i-1} + c \cdot n_i. \tag{3.46}$$

Eq.(3.46) represents a first-order autoregressive process. In the special case where $a_1 = 1$, it is a random walk process. If $|a| > 1$ the disturbance process is unstable. If $a > 0$, the q's are positively correlated: a positive value of q_{i-1} implies a positive value of q_i. If $a < 0$, successive values of q tend to oscillate '+' and '-'.

Disturbances can arise from higher order autoregressive processes. In general:

$$q_i = a_1 q_{i-1} + a_2 q_{i-2} + \ldots + a_l q_{i-l} + c \cdot n_i . \qquad (3.47)$$

Usually, for simulation, $l \leq 3$. Usually $0 < a_j < 1$ and $a_1 < a_{1-1} < \ldots < a_2 < a_1$.

Box and Jenkins (1976) report that the most parsimonious representation of a stochastic process is an autoregressive moving average (ARMA) combination:

$$q_i = \underline{a} \cdot \underline{q} + \underline{b} \cdot \underline{n} . \qquad (3.48)$$

where the order l equals the order k. They find that the l+k order would usually be 2 to 4 for adequate modeling. Usually, a first order AR, Eq.(3.46), with either Gaussian or uniform noise as a driver is both flexible and characteristic of a wide range of observed behavior. Adjustment of a_1 affects the cycle period and the amplitude (persistence of old disturbance effects). Adjustment of c predominantly affects the drift amplitude.

Thus, in this work the sensor/transmitter drift is simulated as:

$$q_i = a_1 \cdot q_{i-1} + c \cdot n . \qquad (3.49)$$

For the S.C. Johnson & Sons reactor simulation, the selection of a_1 and c for each measuring component is chosen as follows:

$a_1 = 0.95$, $c = 2.0 * 10^{-6}$, for monomer mass measurement drift

$a_1 = 0.99$, $c = 0.4$, for temperature measurement drift

$a_1 = 0.90$, $c = 4.0 * 10^{-3}$, for flow rate measurement drift.

Figure 3.11 demonstrates a typical sensor/transmitter drift for temperature obtained from the S.C. Johnson & Sons reactor simulator. Note that the graph shows a drift period of approximately 50 minutes and a net calibration drift of about 2 °F during a 150 minute batch reaction period. For the industrial reactor simulation, the selection of a_1 and c for each measuring component is chosen as follows:

a_1= 0.99, c=1.0*10^{-7}, for monomer mass measurement drift

a_1= 0.99, c=0.2, for temperature measurement drift

a_1= 0.95, c=0.2, for solvent flow rate measurement drift.

a_1= 0.95, c=0.01, for initiator flow rate measurement drift.

a_1= 0.95, c=5*10^{-3}, for steam flow rate measurement drift.

a_1= 0.95, c=1*10^{-4}, for CW flow rate measurement drift.

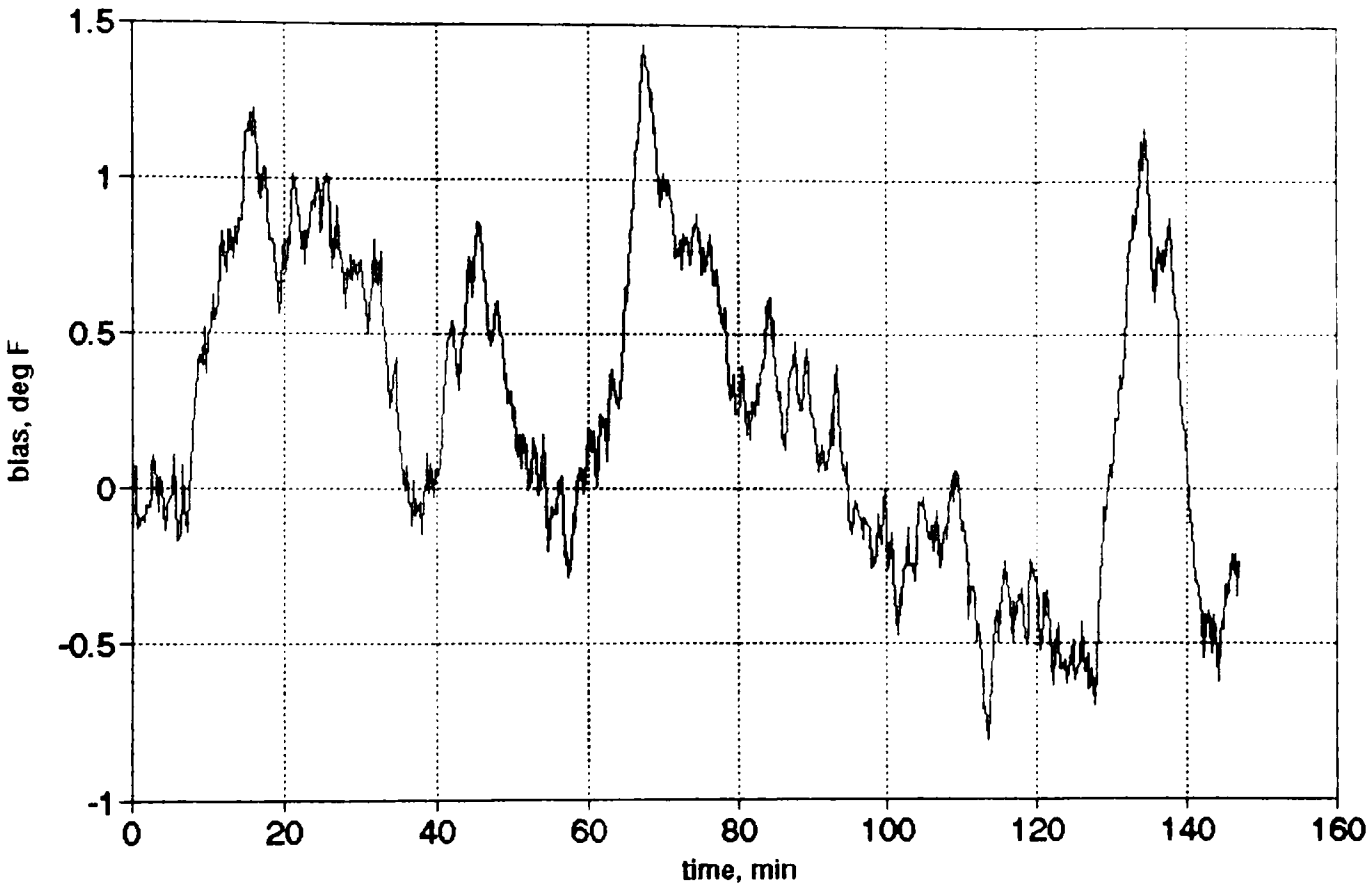


Figure 3.11
Drift on Temperature Measurement
$DR_i=0.99*DR_{i-1}+0.3(r_i-0.5)$
where r_i=independent, random, uniform
$0 < r_i \leq 1$

CHAPTER IV

MATHEMATICAL ANALYSIS OF POLYMERIZATION KINETICS

The special nature of polymerization reactions requires the use of special mathematical techniques in order to solve the set with a seemingly infinite number of differential equations which describe the molecular weight distribution. In this chapter, the standard MWD modeling technique, the method of moments and the new technique called the class method are discussed.

4.1 Method of Moments

The method of moments is a standard approach which offers computational simplicity. The various molecular-weight and chain-length distributions can be characterized by their moments, and the physical significance of each of the leading moments is very helpful in forming the modeling equations. Although this process simulation approach does not provide an actual shape of the molecular weight distribution (MWD), it still gives useful information about the MWD (for instance, average, variance, skewness and kurtosis of the MWD). The use of the method of moments is computationally very effective because only several leading moment variables are required to explain the commercially

relevant characteristics of the MWD (average molecular weight and polydispersity index).

Section 4.1 shows how to predict the molecular weight distribution of the resulting polymer at various reaction stages without directly solving the mechanistical kinetic equations. Further, it shows how certain control action can improve polymerization results. The first part, 4.1.1, explains the distribution function and their moments. The second part shows how to apply the "generating function" to transform the kinetic equations into the moment equations. It is illustrated for an idealized simple polymerization process, and the technique is applied for the polymerization mechanism described in Chapter III. The application results including how the distribution parameters (average molecular weight, variance, skewness, kurtosis), and reaction properties (conversion and reaction rate) change with various control choices (operating temperatures, recipe loadings, and initiator manipulations) are presented in Section 7.2 of Chapter VII.

4.1.1 Distribution Functions and Their Moments

All synthetically produced high molecular weight compounds are obtained in a mixture of molecular sizes differing from each other by the number of monomer units incorporated in the polymer chain. The introduction of

statistical methods is a reasonable response to this state of affairs.

The following analysis is excerpted from Biesenberger (1983).

Consider a property that is distributed among a population, and let the variable 'n' be a measure of the magnitude of that property. The distribution of n is described by a frequency function f_n when n is a discrete variable and by a density function f(n) when n is continuous. Thus f_n, or f(n)dn, represents the portion (number, weight, volume, fraction, etc.) of the population whose property is precisely n (discrete) or whose property lies between n and n+dn (continuous). Since molecular weights have discrete values, what follows uses the frequency function analysis.

Distributions are generally characterized by their shape and breadth and by some measure of the "central" value of n. These characteristics may be computed systematically through the use of moments. Two types of moments are commonly employed. One type is defined about the origin (n=0) of the distribution and the other about its mean value ($\bar{n}$). The k^{th} moment about the origin is defined as

$$\mu_k = \sum_{n=1}^{\infty} n^k f_n. \qquad (4.1)$$

and the k^{th} moment about the mean is defined as

$$\sigma_k = \sum_{n=1}^{\infty} (n-\bar{n})^k f_n. \tag{4.2}$$

where k is an integer and f signifies that the distribution has been normalized through division by the population size

$$f_n = \frac{f_n}{\sum_{n=1}^{\infty} f_n}. \tag{4.3}$$

Thus μ_k represents the k-th moment of the normalized distribution about the origin.

Moments σ_k and μ_k are related in general by

$$\sigma_k = \sum_{j=0}^{k} \binom{k}{j} (-1)^{(k-j)} \mu_j \mu_1^{(k-j)}. \tag{4.4}$$

where $\binom{k}{j}$ is the binomial coefficient.

Three examples of these equations are:

$$\sigma_2 = \mu_2 - \mu_1^2. \tag{4.5}$$

$$\sigma_3 = \beta_3 - 3\beta_2\beta_1 + 2\beta_1^3. \tag{4.6}$$

$$\sigma_4 = \beta_4 - 4\beta_3\beta_1 + 6\beta_2\beta_1^2 - \beta_1^4. \tag{4.7}$$

Note that the 2nd moment about the average, σ_2, is identical to the variance, σ^2. But, in general, $\sigma_k \neq \sigma^k$.

Several commonly used characteristics follow. One is μ_0, which measures population size. Another is the mean

$$\beta_1 = \frac{\mu_1}{\mu_0} = \bar{n} = n_{avg}. \tag{4.8}$$

which represents a central value of n. A characteristic of distributions that is of considerable importance to us is their breadth or dispersion. Such a characteristic is the variance σ^2 or, equivalently, the standard deviation, SD. Both measure the mean deviation of a distribution about its mean value. Thus,

$$\sigma_2 = \sigma^2 = \overline{(n-\bar{n})^2} = \overline{n^2} - \bar{n}^2 = var\ n. \tag{4.9}$$

$$SD = (var\ n)^{1/2} = \sigma^1 = \sqrt{\sigma_2} \neq \sigma_1. \tag{4.10}$$

The shape of a distribution, specifically its asymmetry, is measured by skewness,

$$SK = \frac{\sigma_3}{(SD)^3}. \tag{4.11}$$

and specifically its peakedness is measured by kurtosis.

$$KU = \frac{\sigma_4}{(SD)^4} - 3. \tag{4.12}$$

Thus, if we calculate moments up to the 4th (μ_4) of the distribution, we will be able to get these four statistics of the number distribution. For the weight distribution, one additional moment μ_5 is required. For the ideal normal (Gaussian) distribution, both SK and KU are zero. The formulae for each of these four leading moments of molecular weight distribution are derived and presented in terms of the average chain length in Appendix C.

4.1.2 Application of Generating Functions to Moment Calculations

The following analysis is excerpted from Goodrich (1967): An approach which has wide use in polymer kinetics and which can be used to produce key features of the molecular weight distribution (MWD) dynamics is the generating function technique. The technique allows one to perform the indicated series summations in Eq.(4.1) without requiring each value of the polymer species. One should define a generating function for the molecular weight distribution, and extract all the moments from this. For example, if we are dealing with a homopolymerization with polymer concentration P_n, the generating function

$$G(s,t) = \sum_{n=1}^{\infty} s^n P_n = sP_1(t) + s^2P_2(t) + s^3P_3(t) + s^4P_4(t) \ldots \qquad (4.13)$$

will serve to generate all of the number-moments. Here P_n is the frequency function, f_n.

The number-moments μ_k have a relationship with the generating function

$$[(s\frac{\partial}{\partial s})^k G(s,t)]|_{s=1} = \sum_{n=1}^{\infty} n^k P_n(t) = \mu_k. \qquad (4.14)$$

Because sums of this sort are required in the evaluation of distribution moments, a knowledge of the generating function enables the investigator to calculate the moments of a distribution even without possessing an explicit formula for the distribution itself.

A second useful property of the generating function is its ability to order convolution sums. We shall require in our work expressions of the type

$$\Sigma P_j P_{x-j} \qquad (4.15)$$

known as convolution sums, and by squaring both sides of Eq.(4.13) and rearranging the result as a power series in s, Eq.(4.15) is identical to the coefficient of s^n in $G^2(s,t)$.

Consider a simple system:

$$\frac{dp_1}{dt} = -P_1 \qquad P_1(0)=P_{1_0}. \qquad (4.16)$$

$$\frac{dp_n}{dt} = -P_n + P_{n-1} \qquad P_n(0)=P_{1_0}. \qquad (4.17)$$

P_{n_0}

The generating function for this system is defined as Eq.(4.13). The equations for G(s,t) in the batch reactor for the simple system, Eqs.(4.16) and (4.17) can be obtained by multiplying each equation by s^n and summing over n to yield

$$\frac{dG(s,t)}{dt} = (s-1)G(s,t) \qquad G(s,0) = sP_{1_0}. \tag{4.18}$$

Or, simply the system can be transformed into the generating function form by taking the derivative of Eq.(4.13) and then inserting Eq.(4.17):

$$\frac{dG(s,t)}{dt} = \frac{d}{dt}\sum_{n=1}^{\infty} s^n P_n = s\sum_{n=1}^{\infty}\frac{dP_n}{dt} = -\sum_{n=1}^{\infty} s^n P_n + \sum_{n=1}^{\infty} s^n P_{n-1}$$

$$= -\sum_{n=1}^{\infty} s^n P_n + s\sum_{n=1}^{\infty} s^{n-1} P_{n-1} = (s-1)G(s,t). \tag{4.19}$$

The differential equation for μ_0 results from setting s=1 in Eq.(4.18).

$$\frac{d\mu_0}{dt} = 0 \qquad \mu_0(0) = P_{1_0}. \tag{4.20}$$

The differential equations for μ_1, μ_2 and μ_3 result from differentiation with respect to s on both sides of Eq.(4.18).

$$\frac{d\mu_1}{dt} = \frac{\partial}{\partial t}\left[\frac{\partial G(s,t)}{\partial s}\right]_{s=1} = \left[G(s,t) + (s-1)\frac{\partial G(s,t)}{\partial s}\right]_{s=1} = G(1,t) = \mu_0. \tag{4.21}$$

$$\frac{d\mu_2}{dt} = \frac{\partial}{\partial t}\left[\frac{\partial G(s,t)}{\partial s} + \frac{\partial^2 G(s,t)}{\partial s^2}\right]_{s=1} = \left[\frac{d\mu_1}{dt} + 2\frac{\partial G(s,t)}{\partial s} + (s-1)\frac{\partial^2 G(s,t)}{\partial s^2}\right]_{s=1} = \mu_0 + 2\mu_1. \tag{4.22}$$

$$\frac{d\mu_3}{dt} = \frac{\partial}{\partial t}\left[\frac{\partial G(s,t)}{\partial s} + 3\frac{\partial^2 G(s,t)}{\partial s^2} + \frac{\partial^3 G(s,t)}{\partial s^3}\right]$$

$$= \left[-2\frac{\partial \mu_1}{dt} + 3\frac{\partial \mu_2}{\partial} t + 3\frac{\partial^2 G(s,t)}{\partial s^2} + (s-1)\frac{\partial^2 G(s,t)}{\partial s^3}\right]_{s-1} = \mu_0 + 3\mu_1 + 3\mu_2. \qquad (4.23)$$

Similarly, the differential equations for higher order can be derived.

Since we know the first four moments (μ_0, μ_1, μ_2 and μ_3) of the distribution, we can calculate the number average chain length and variance of the chain length distribution

$$X_N = \frac{\mu_1}{\mu_0}. \qquad (4.24)$$

$$\sigma_N^2 = \frac{\mu_2}{\mu_0} - \left(\frac{\mu_1}{\mu_0}\right)^2. \qquad (4.25)$$

and also weight average parameters

$$X_W = \frac{\mu_2}{\mu_1}. \qquad (4.26)$$

$$\sigma_W^2 = \frac{\mu_3}{\mu_0} - \left(\frac{\mu_2}{\mu_1}\right)^2. \qquad (4.27)$$

In the above equations N and W are not integer subscript notations. They denote number- and weight-average properties. The number-average property reflects the predominant number of small chain length species while the weight-average property reflects the importance of the large chain length species. Therefore, theoretically the weight-average chain length is always greater than the

number-average chain length. The bigger the difference between these two values, the more the distribution broadens, and it is measured by number-Polydispersity Index (PDI) which is defined as:

$$PDI_N = \frac{X_N}{X_N}. \tag{4.28}$$

The PDI is widely used as a standard measure of breadth of distribution instead of variance because X_w and X_N are readily measurable.

These mathematical methods are applied to the free-radical homopolymerization mechanism described in Chapter III.

The generating function for the growing polymer radicals and the dead polymer species are defined as

$$G(s,t) = \sum_{n-1}^{\infty} s^n P_n(t), \tag{4.29}$$

$$F(s,t) = \sum_{n-1}^{\infty} s^n D_n(t), \tag{4.30}$$

respectively.

By inserting the $P_n(t)/dt$ expression into the differential form of Eq.(4.29):

$$\frac{dG(s,t)}{dt} = s^n \sum_{n-1}^{\infty} \frac{dP_n}{dt}. \tag{4.31}$$

the dP_n/dt term is transformed into the generating function form, $dG(s,t)/dt$. Similarly, the dD_n/dt term is also transformed into the generating function form, $dF(s,t)/dt$. Now the moments differential equations, $d\lambda_k/dt$ for growing polymer radical, and $d\mu_k/dt$ for dead polymer, will be obtained by the method explained earlier. The moment differential equations have been derived for up to the fifth moment for each. The detailed derivations and results are presented in Appendix D. To keep the units consistent as discussed in Chapter III, neglect the first term (volume contraction term) in the RHS of Eqs.(D.79) to (D.92) in Appendix D. Solving the derived moments' equations in Appendix D will provide the dynamic behavior of each moment, and consequently MWD parameters including mean, variance, skewness and kurtosis are readily obtained using Eqs.(4.8), (4.9), (4.11) and (4.12).

4.2 Class Method

There is no technical problem to solve the kinetic equations presented in Figures 3.1 to 3.4, but it is extremely time consuming. A direct approach to solve those differential equations has been made for a polymerization process whose average chain length is relatively short. That is not feasible to the high average chain length polymerization. The main advantage of computing the entire MWD versus the leading moments of the distribution is where

skewed or bimodal MWD's arise, cases which leading moments cannot adequately describe. In this section, a novel MWD modeling technique called the class method is presented. This approximate method computes the entire MWD without variable transformation. It is related to energy partitioning of neutrons in a nuclear reactor to obtain the energy population distribution.

The method categorizes the polymer species into molecular weight classes according to the sizes of species and all species in the class are assumed to behave the same as the average weight molecule in each class. To illustrate by assuming there exist 120,000 possible growing polymer species at the end of reaction for the system shown in Figures 3.1 to 3.5. That means the maximum chain length obtainable of growing polymer is 120,000. If one divides the species into 300 classes, each class will have 400 species. Therefore,

$$
\begin{aligned}
{}_1P &= P_1 + P_2 + P_3 + \ldots P_{201} \ldots + P_{400} = 400 \langle P \rangle_{\text{class 1}} \\
{}_2P &= P_{401} + P_{402} + P_{403} + \ldots P_{601} \ldots + P_{800} = 400 \langle P \rangle_{\text{class 2}} \\
{}_3P &= P_{801} + P_{802} + P_{803} + \ldots P_{1001} \ldots + P_{1200} = 400 \langle P \rangle_{\text{class 3}} \\
{}_4P &= P_{1201} + P_{1202} + P_{1203} + \ldots P_{1401} \ldots + P_{1600} = 400 \langle P \rangle_{\text{class 4}} \\
&\vdots \qquad\qquad \vdots \qquad\qquad \vdots \qquad\qquad \vdots \qquad\qquad \vdots \\
{}_{299}P &= P_{119201} + P_{119202} + P_{119203} + \ldots P_{119401} \ldots + P_{119600} = 400 \langle P \rangle_{\text{class 299}} \\
{}_{300}P &= P_{119601} + P_{119602} + P_{119603} + \ldots P_{119801} \ldots + P_{120000} = 400 \langle P \rangle_{\text{class 300}}
\end{aligned}
$$

where P_i represent the moles of polymer of degree of polymerization i and ${}_nP$ represents the moles of polymer in

the n^{th} class and it is assumed that P_i ($400n+1 \leq i \leq 400n+400$) behaves identically to $P_{400n+201}$.

Now the mass balance equations on each of the growing polymer species (Eq.(3.10),(3.11) in Figure 3.2)

$$\frac{dP_1}{dt} = k_i RM - k_p M P_1 + (k_{tm}M + k_{ts}S)(P - P_1) - k_t P P_1.$$

$$\frac{dP_n}{dt} = k_p M(P_{n-1} - P_n) - (k_{tm}M + k_{ts}S) P_n - k_t P P_n.$$

where n = 2, 3, 4, ... ,120000

are transformed into set of mass balance equations on each of the growing polymer classes:

$$\frac{d\ {}_1P}{dt} = k_i RM - k_p M \frac{{}_1P}{400} + (k_{tm}M + k_{ts}S)(P - {}_1P) - k_t\ {}_1P P. \qquad (4.32)$$

$$\frac{d\ {}_nP}{dt} = k_p M\left(\frac{{}_{n-1}P}{400} - \frac{{}_nP}{400}\right) - (k_{tm}M + k_{ts}S)\ {}_nP - k_t P\ {}_nP. \qquad (4.33)$$

where n = class 2, 3, 4, ... , 300

Similarly transforming the mass balance equation on each of the dead polymer species, Eq.(3.14), gives

$$\frac{d\ {}_nD}{dt} = (k_{tm}M + k_{ts}S)\ {}_nP + k_{td}\ {}_nP P + \frac{1}{2}k_{tc}\sum_{m=1}^{n-1}\ {}_mP\ {}_{n-m}P. \qquad (4.34)$$

By solving the differential equations (4.32), (4.33) with Eq.(4.34), one will get the moles of each class of growing polymer and dead polymer. The resulting dead polymer, ${}_nD$, will have twice as many classes because of termination by

combination. The moments can also be calculated from the moles of each class:

$$\lambda_i = \sum_{n=1}^{n_{max}\ of\ GP} (n*400-200)^i \cdot {}_nP \qquad for\ i=0,1,2,3,... \qquad (4.35)$$

$$\mu_i = \sum_{n=1}^{n_{max}\ of\ DP} (n*400-200)^i \cdot {}_nD \qquad for\ i=0,1,2,3,... \qquad (4.36)$$

The values in the parentheses represent the average chain length in the n^{th} class.

The moments calculated by Eqs.(4.35),(4.36) will deviate from those obtained by moment model because of the MWD discretization and the uniformity assumption for each species in each class. The fewer the number of classes, the lower is the computation load. However, such coarse discretization will lose accuracy. Therefore, an issue on the class method is how to balance the competing desires of obtaining accurate molecular weight distribution versus practical utility. There are no guidelines of how to predict the maximum chain length obtainable and how to choose a reasonable number of classes. One should try to find reasonable values for the maximum chain length and the number of classes such that $\lambda_1+\mu_1$ as calculated by Eq.(4.35) and (4.36) is close to the total moles of monomer loaded into the reactor.

CHAPTER V

THE TEMPERATURE CONTROL STRATEGY FOR THE EXOTHERMIC BATCH REACTOR

5.1 Introduction to NPMBC control

The control problem concerns the temperature control of a batch reactor and the design of a controller capable of maintaining the desired reaction temperature throughout the batch reaction time. The controller begins regulating the jacket steam and cold water flow to maintain the reactor temperature immediately after the start-up. Cold monomer or solvent enters the reactor as pre-scheduled. Cold inflow and ambient losses would require heat, but the heat of polymerization requires a net cooling. As the reaction proceeds, the heat duty changes and the steam or water rates must be continuously adjusted. Further, as the reaction proceeds, the heat transfer coefficient changes requiring a change in the dynamics of valve action.

Conventional split-range cascaded PID control gives a sluggish control response because shifting of the slave output signal to either of the final control elements lags behind the master output. The switch becomes possible only after the sign of the slave controller actuating error changes. The sign change is sluggish due to the reset windup on the master controller (Choi et al., 1992). Since the conventional control cannot accommodate the

nonlinear/nonstationary controller performance, it is insufficient to meet the control performance criteria which is to maintain the reactor temperature within ±1°C from setpoint.

While PID and linear model-based controllers essentially ignore material and energy balance information, nonlinear process-model-based control (NPMBC) strategies use this information directly. In NPMBC, the process is controlled by using models based on, but not exactly the same as the process. By comparison to linear and PID-type approaches the NPMBC action is "intelligent" and has shown benefits in uniformity, disturbance rejection, and setpoint tracking (Rhinehart, 1994). All of which translate into better process economics.

The basic concept of NPMBC (Rhinehart, 1994) is illustrated in Figure 5.1 where the "controller," I, uses a model of the process to calculate a value for the manipulated variable which should make the process behave in a desired way. That calculated value is then implemented. The controller does not have PID components; and, in general, there is only one tuning parameter, the choice of how fast the process should move to the setpoint. The symbol, I, represents "inverse." In a normal modeling approach, you set the input and watch how the output responds. By contrast, NPMBC determines what input causes a set output response, the inverse.

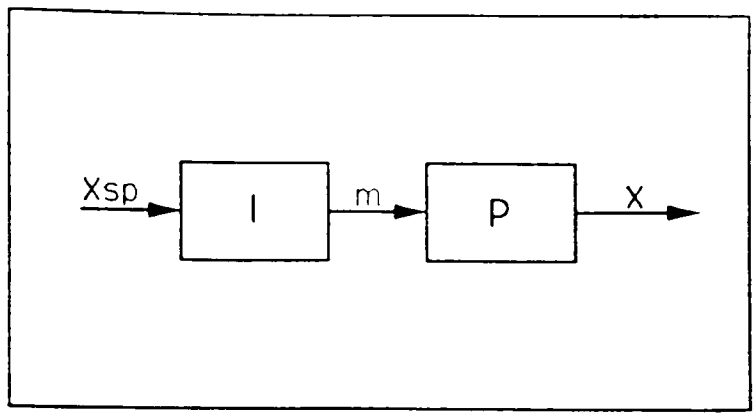


Figure 5.1
The NPMBC Concept, Model Inverse and Process

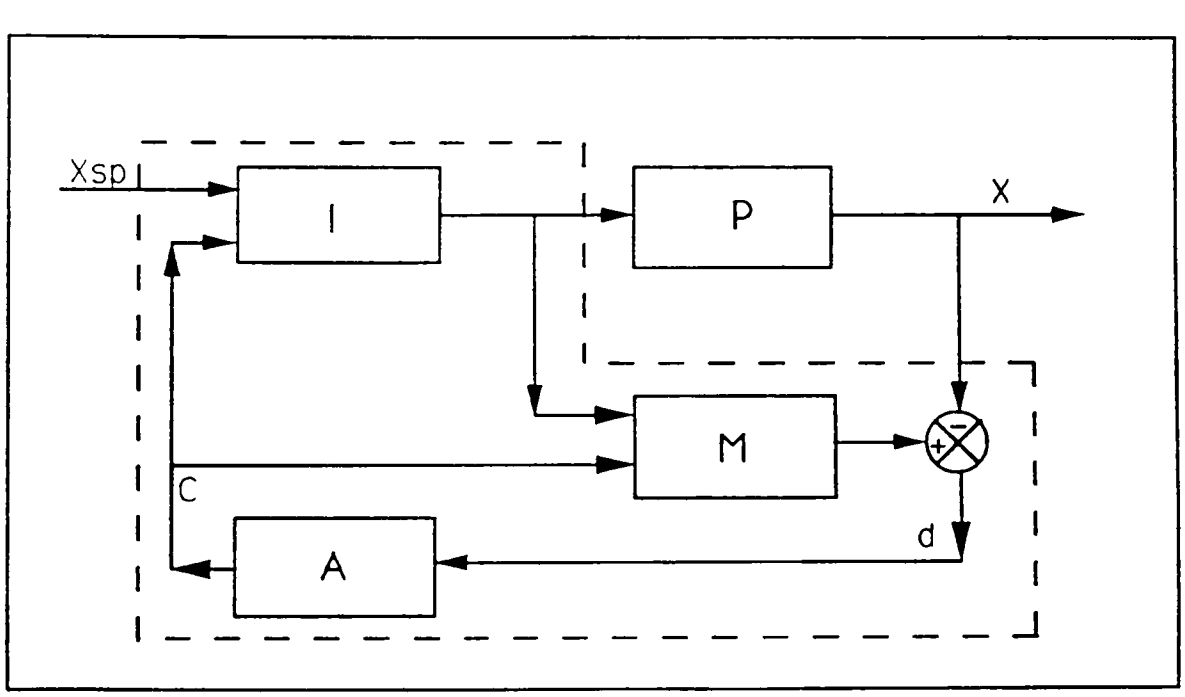


Figure 5.2
The Three NPMBC Functions, Inverse, Model, and Adjustment

If the model is perfect and there are no constraints, the open-loop structure of Figure 5.1 will work. However, practicable controller models are not perfect, and some form of feedback correction is required. The conventional feedback approach is illustrated in Figure 5.2, where the difference, d, between the model, M, and the process output is monitored and used to adjust, A, a controller feature, c, which is usually either a setpoint bias or a model coefficient. The controller is all three I, M and A functions enclosed by the dashed line, not simply I.

The controller model will have adjustable parameters as many as the number of on-line process measurements, and which are periodically updated by Incremental Model Parameterization On-Line (IMPOL) (Rhinehart and Riggs, 1990) using on-line measurements. The parameterized model, then can be useful for control action (the inverse) which determines, for this particular case, the amounts of heating or cooling required in terms of the steam flow rate or cooling water make-up rate. Primarily, the three adjustable model parameters for the industrial scale reactor case are the total moles of growing polymer species (λ_o), heat transfer coefficients (U) and environmental heat loss (EH_{loss}). The selection guide for the adjustable parameter is discussed in the subsequent section.

The IMPOL method for model parameterization uses a one-step process/model mismatch and the modeled output/parameter

sensitivity to calculate an incremental adjustment to the model parameters at each reparameterization interval. A dynamic process model is used to determine the output/parameter sensitivity. This method promises the advantages of computational speed, upsetless process parameterization, conceptual simplicity, continuous adaptation and phenomenological validity.

In a recent approach to nonlinear model-based control, Lee and Sullivan (1988) have introduced generic model control (GMC). In it a setpoint/process error integral is added to corresponding transient conservation equations in the nonlinear process model to accommodate for mismatch between reparameterization steps. The GMC law with IMPOL is used in this study.

5.2 Controller

5.2.1 Model Adjustment Principle

The first controller function is to adjust the model to keep it true to the process. This work uses the incremental model parameter adjustment of Rhinehart and Riggs (1991).

Consider a nonstationary SISO process described by the following ordinary differential equation:

$$\frac{dy}{dt} - f(y,u,\boldsymbol{d},\boldsymbol{p}) . \qquad (5.1)$$

where y is the dependent (output) variable, u is an independent (input) variable, **d** is a vector of disturbances (measured and unmeasured) and **p** is a vector of parameters of the process. For example, **p** can contain diffusion coefficients, leak rates, reaction rate coefficients and heat transfer coefficients. In general, these are nonstationary. Eq.(5.1) cannot be rigorously known.

For process control and on-line optimization, it is often desirable to represent the process with an approximate nonlinear model. To be computationally rapid, the approximate model should be considerably simpler than the real process; but, to be functionally useful, it should mechanistically account for the dominant phenomena of the process. Such an approximate model is represented as:

$$\frac{d\bar{y}}{dt} = \bar{f}(\bar{y},\bar{u},\bar{\mathbf{d}},\bar{\mathbf{p}},\phi) . \qquad (5.2)$$

where the overbars represent approximate model characteristics. For example, $\bar{d}$ would be the measurable disturbances and ϕ would be the adjustable model parameter. Note that the input variable $\bar{u}$ is not the same for both the process and the approximate model. The process "knows" what the value of u is, while the model uses a filtered measured value. The adjustable model parameter ϕ must be selected by the modeler, and the criteria (Rhinehart and

Riggs, 1991) are that: (1) the model output should be realistically sensitive to ϕ, and (2) ϕ should phenomenologically represent either a major modeling uncertainty or a nonstationary process parameter(s) which is (3) known to change with time. Such a selection criteria requires both process and modeling expertise.

The model is approximate, the process is nonstationary, and there are unmeasured disturbances. Consequently, a stationary value for ϕ is not likely to keep the approximate model "true" in an input-output sense as process conditions change. The approach taken here is to change ϕ with the objective that the output of the model matches the process, given the inputs. A direct measure of how well the approximate model is working is the difference between the measured process output y and the predicted value $\bar{y}$. At a particular reparameterization time t, this difference ϵ, is given by

$$\epsilon_t = y_t - \bar{y}_t. \tag{5.3}$$

Ideally ϵ_t is zero, and for a noisy process y_t would be a filtered value. The value of $\bar{y}_t$ can be calculated from Eq.(5.2) using the 4th-order Runge-Kutta method.

The reparameterization objective is to find ϕ such that ϵ_t becomes zero. Applying Newton's method, ϕ, the root of Eq.(5.3) is given by:

$$\phi_{new} = \phi_{old} - \frac{\epsilon_t}{\frac{\partial e}{\partial \phi}|_t}. \qquad (5.4)$$

Now restating $\bar{y}_t$, as discretized from Eq.(5.2):

$$\bar{y}_t \approx y_{t-\Delta t} + \Delta t \cdot \bar{f}(t-\Delta t, y_{t-\Delta t}, \bar{u}, \bar{d}, \bar{p}, \phi). \qquad (5.5)$$

and applying Eq. (5.3) and (5.5) sequentially yields:

$$\frac{\partial e}{\partial \phi}|_t - 0 - \frac{\partial \bar{y}}{\partial \phi}|_t - 0 - 0 - \Delta t \cdot \frac{\partial \bar{f}}{\partial \phi}|_{t-\Delta t} - - \Delta t \frac{\partial \bar{f}}{\partial \phi}|_{t-\Delta t}. \qquad (5.6)$$

Then, for a single step adjustment to ϕ:

$$\phi|_t - \phi|_{t-\Delta t} + \frac{\epsilon_t}{\Delta t \frac{\partial \bar{f}}{\partial \phi}|_{t-\Delta t}}. \qquad (5.7)$$

Eq.(5.7) is a single step adjustment to ϕ, and applying it once at each reparameterization interval will progressively reparameterize the model and incrementally adjust it toward being an accurate input-output representation of the process. Parameter values usually change slowly in comparison to data sampling and control periods. Consequently, iterative application of Eq.(5.7) at each sampling to find the exact value of ϕ is probably unnecessary.

Because Newton's method is a linearization, in certain cases Eq.(5.7) overestimates the changes in the parameter. In other cases, it underestimates them. Therefore, a

relaxation coefficient α is incorporated to adjust the change:

$$\phi|_t - \phi_{t-\Delta t} + \alpha \frac{e_t}{\Delta t \frac{\partial \bar{f}}{\partial \phi}|_{t-\Delta t}} . \qquad (5.8)$$

In fact, if α is too large, ϕ will tend to oscillate. If α is too small, sluggish tracking will result. In this work, I used the ideal $\alpha = 1$.

Since $\bar{f}$ represents $d\bar{y}/dt$, Rhinehart and Riggs term Eq.(5.8) the derivative IMPOL method. Its application to the specific reactor is described next.

5.2.2 Model Adjustment of the Controller for the S.C. Johnson & Sons Reactor

Model adjustment of the controller for the S.C. Johnson & Sons reactor is presented in Appendices A and B. Appendix A contains the application of two parameter model and its adjustment. Appendix B contains the application of the three parameter model and its adjustment.

5.2.3 Model Adjustment of the controller for the Industrial Size Reactor

Looking back at the process dynamic equations in Figures 3.1 to 3.5, there are several candidates for the adjustable parameters as shown in Table 5.1.

The number of adjustable parameters which will be used for model adaptation depends on the number of process

Table 5.1
List of Possible Adjustable Parameters

Symbol	Description
λ_0	total moles of growing polymer species
D	molecular diffusivity of growing polymer
k_p	propagation rate constant
U	overall heat transfer coefficient
f_d	reactor fouling factor
EH_{loss}	heat loss to environment

measurements. Since the mixture density (d_{mix}), and the reactor and jacket temperatures (T_r and T_j) are the three measurable data which represents the process conditions, three adjustable parameters should be selected and three equations (monomer mass balance, reactor side energy balance equation, jacket side energy balance equation) should be used for reparameterization. As three adjustable parameters, λ_0, U, and EH_{loss}, are chosen because they met all the criteria given in Section 5.2.1. The approximate control model which is equivalent to Eq.(5.2) can be described as follow (Note the bold-faced letters for adjustable parameters):

$$\frac{d\bar{n}_M}{dt} = \frac{\dot{m}_M}{MW_M} - 2\eta k_d \cdot n_I - (k_p + k_{tm})\,\boldsymbol{\lambda_0}\cdot\bar{n}_M. \qquad (5.9)$$

$$\frac{d(\sum m_1 Cp_1 \bar{T}_r)}{dt} - (\dot{m}_M Cp_M + \dot{m}_S Cp_S + \dot{m}_I Cp_I)(T_{amb} - \bar{T}_r) + k_p \boldsymbol{\lambda_0}\cdot\bar{n}_M(-\Delta H_p) - \boldsymbol{U}A(\bar{T}_r - \bar{T}_j) - \boldsymbol{EH_{loss}}. \qquad (5.10)$$

$$(V_j \rho_{w,j} Cp_{w,j})\frac{d\bar{T}_j}{dt} = \dot{v}_w \rho_w C_{p,w}(T_w - \bar{T}_j) + \dot{m}_s(\lambda_s + C_{p,w}(T^\sigma - \bar{T}_j))$$

$$- \dot{m}_s C_{p,w}\bar{T}_j + \boldsymbol{U}A(\bar{T}_r - \bar{T}_j). \qquad (5.11)$$

One can easily notice the mismatch between the process dynamic model and this control model. For instance, U in Eqs.(5.10) and (5.11) replaces the overall heat transfer coefficient, $1/(1/h_i + 1/h_f)$, in the process energy balance (Eqs.(3.33) & (3.34)). Furthermore the reactor side energy balance (Eq.(5.10)) does not retain the environmental heat

loss term. In addition to the mismatch between the process and control model, the control model parameters are chosen such that the mismatches exist. They are listed in Table 5.2. Eqs.(5.9), (5.10) and (5.11) shall be used as the conversion and temperature predictor by which the controller obtains the difference between the measured process output n_M, T_r, T_j and predicted value $\bar{n}_M$, $\bar{T}_r$ and $\bar{T}_j$ as described in Eq.(5.3). Note that these prediction equations contain adjustable parameters which are continuously updated every reparameterization step. This set of ordinary differential equations are solved by the 2nd order Runge-Kutta method between the previous and current reparameterization steps. The reparameterization objective is to find λ_o, U and EH_{loss} such that $\epsilon_{t,M}$, $\epsilon_{t,r}$ and $\epsilon_{t,j}$ become zero. By restating $\bar{n}_M$, $\bar{T}_r$ and $\bar{T}_j$ is discretized form as in Eq.(5.5) one can get three objective functions shown below:

$$\epsilon_{t,M} = n_{M,t} - \bar{n}_{M,t} - n_{M,t} - n_{M,t-\Delta t} - \Delta t \left(\frac{\dot{m}_M}{MW_M} - 2\eta k_d n_I - (k_p + k_{tm}) \lambda_0 \bar{n}_M \right) . \qquad (5.12)$$

$$\epsilon_{t,r} = T_{r,t} - \bar{T}_{r,t} - T_{r,t} - T_{r,t-\Delta t} - \Delta t \left[(\dot{m}_M Cp_M + \dot{m}_S Cp_S + \dot{m}_I Cp_I) (T_{amb} - T_{r,t-\Delta t}) \right.$$

$$\left. + k_p \lambda_0 n_{m,t-\Delta t} (-\Delta H_p) - U\bar{A}_{t-\Delta t} (T_{r,t-\Delta t} - T_{j,t-\Delta t}) - EH_{loss} \right] . \qquad (5.13)$$

Table 5.2
The List of Process-Model Mismatches

ITEM	PROCESS MODEL	CONTROL MODEL	ITEM	PROCESS MODEL	CONTROL MODEL
Kinetic Data			**Physical Property**		
k_d^0	$1.0533*10^{15}$	$1.0*10^{15}$	d_i	915.0	910.0
k_{p0}^0	$4.9167*10^5$	$5.0*10^5$	d_m	889.5	880.0
$k_{\theta p}^0$	$3.0233*10^{13}$	$3.0*10^{13}$	d_s	820.0	810.0
k_{tm}^0	$4.6610*10^9$	$4.6*10^9$	d_p	1200.0	1190.0
k_{ts}^0	$4.9658*10^8$	$4.9*10^8$	d_w	1000.0	995.0
k_{t0}^0	$9.8000*10^7$	$9.8*10^7$	MW_i	164.21	163.0
$k_{\theta t}^0$	$1.4540*10^{20}$	$1.4*10^{20}$	MW_m	100.12	99.0
k_{tr}^0	$2.5280*10^3$	$2.5*10^3$	MW_s	90.14	89.0
E_d	$1.2877*10^5$	$1.2*10^5$	$C_{p,i}$	1.2	1.3
E_{p0}	$1.8283*10^4$	$1.8*10^4$	$C_{p,m}$	2.0	2.2
$E_{\theta p}$	$1.1700*10^5$	$1.1*10^5$	$C_{p,s}$	2.5	2.6
E_{tm}	$7.4479*10^4$	$7.4*10^4$	$C_{p,p}$	2.0	2.2
E_{ts}	$6.6163*10^4$	$6.6*10^4$	$C_{p,w}$	4.2	4.3
E_{t0}	$2.9442*10^3$	$2.9*10^3$	ΔH_p	75140.0	76000.0
$E_{\theta t}$	$1.4584*10^5$	$1.4*10^5$	T_{gp}	378.2	377.0
E_{tr}	$1.7178*10^4$	$1.7*10^4$	λ_s	2015.0	2010.0
Geometric Data & etc.			$T_{cw,i}$	5.0	4.9
ID_r	2.5	2.45	$T_{r,i}$	20.0	19.5
A_{btm}	0.167	0.162	$T_{j,i}$	21.0	20.5
V_j	2.3	2.25			
V_{cl}	1.5	1.45			

$$\epsilon_{t,j} = T_{j,t} - \overline{T}_{j,t} - T_{j,t} - T_{j,t-\Delta t} - \Delta t[\dot{v}_w \rho_w (T_w - T_{j,t-\Delta t}) +$$

$$\dot{m}_s \lambda_s + U\overline{A}_{t-\Delta t}(T_{r,t-\Delta t} - T_{j,t-\Delta t})]. \qquad (5.14)$$

where Δt = reparameterization interval.

Since we have three equations and three unknowns, Newton's method can solve them. The roots of equations are obtained by:

$$\begin{bmatrix} \frac{\partial \epsilon_{m,t}}{\partial \lambda_0} & \frac{\partial \epsilon_{m,t}}{\partial U} & \frac{\partial \epsilon_{m,t}}{\partial EH_{loss}} \\ \frac{\partial \epsilon_{r,t}}{\partial \lambda_0} & \frac{\partial \epsilon_{r,t}}{\partial U} & \frac{\partial \epsilon_{r,t}}{\partial EH_{loss}} \\ \frac{\partial \epsilon_{j,t}}{\partial \lambda_0} & \frac{\partial \epsilon_{j,t}}{\partial U} & \frac{\partial \epsilon_{j,t}}{\partial EH_{loss}} \end{bmatrix} \begin{bmatrix} \Delta\lambda_0 \\ \Delta U \\ \Delta EH_{loss} \end{bmatrix} = - \begin{bmatrix} \epsilon_{m,t} \\ \epsilon_{r,t} \\ \epsilon_{j,t} \end{bmatrix}. \qquad (5.15)$$

If one goes through the steps in Eq.(5.6), it yields:

$$\Delta t \begin{bmatrix} \frac{\partial \overline{f}_{n_N,t-\Delta t}}{\partial \lambda_0} & \frac{\partial \overline{f}_{n_N,t-\Delta t}}{\partial U} & \frac{\partial \overline{f}_{n_N,t-\Delta t}}{\partial EH_{loss}} \\ \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial \lambda_0} & \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial U} & \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial EH_{loss}} \\ \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial \lambda_0} & \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial U} & \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial EH_{loss}} \end{bmatrix} \begin{bmatrix} \Delta\lambda_0 \\ \Delta U \\ \Delta EH_{loss} \end{bmatrix} - \begin{bmatrix} \epsilon_{m,t} \\ \epsilon_{r,t} \\ \epsilon_{j,t} \end{bmatrix}. \qquad (5.16)$$

Then, for a single step adjustment to λ_o, U & EH_{loss} incorporating with relaxation coefficient α:

$$\Delta\lambda_0 = \frac{\alpha(-\epsilon_{m,t} \cdot \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial U} \cdot \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial EH_{loss}})}{DET}. \qquad (5.17)$$

$$\Delta\lambda_0 = \lambda_{0_t} - \lambda_{0_{t-\Delta t}}. \qquad (5.18)$$

$$\Delta U = \frac{\alpha\left(-e_{j,t}\cdot\frac{\partial f_{r,t-\Delta t}}{\partial EH_{loss}}\cdot\frac{\partial f_{n_M,t-\Delta t}}{\partial \lambda_0}\right)}{DET}. \tag{5.19}$$

$$\Delta U = U_t - U_{t-\Delta t}. \tag{5.20}$$

$$\Delta EH_{loss} = \frac{\alpha\left(e_{j,t}\frac{\partial f_{n_M,t-\Delta t}}{\partial \lambda_0}\frac{\partial f_{r,t-\Delta t}}{\partial U} + e_{m,t}\frac{\partial f_{r_t-\Delta t}}{\partial \lambda_0}\frac{\partial f_{j_t-\Delta t}}{\partial U}\cdot -e_{r,t}\frac{\partial f_{j_t-\Delta t}}{\partial U}\frac{\partial f_{n_M,t-\Delta t}}{\partial \lambda_0}\right)}{DET}. \tag{5.21}$$

$$\Delta EH_{loss} = EH_{loss,t} - EH_{loss,t-\Delta t}. \tag{5.22}$$

$$DET = \Delta t\left(-\frac{\partial f_{m,t-\Delta t}}{\partial \lambda_0}\cdot\frac{\partial f_{r,t-\Delta t}}{\partial EH_{loss}}\cdot\frac{\partial f_{j,t-\Delta t}}{\partial U}\right). \tag{5.23}$$

The partial derivatives in Eqs. (5.17), (5.18) and (5.19) are defined as follows:

$$\frac{\partial f_{m,t-\Delta t}}{\partial \lambda_0} = -\overline{n}_{M,t-\Delta t}. \tag{5.24}$$

$$\frac{\partial f_{m,t-\Delta t}}{\partial U} = 0. \tag{5.25}$$

$$\frac{\partial f_{m,t-\Delta t}}{\partial EH_{loss}} = 0. \tag{5.26}$$

$$\frac{\partial f_{r,t-\Delta t}}{\partial \lambda_0} = \frac{\overline{n}_{M,t-\Delta t}(-\Delta H_p)}{\sum(\overline{m}_{l,t-\Delta t}Cp_l)}. \tag{5.27}$$

$$\frac{\partial f_{r,t-\Delta t}}{\partial U} = -\frac{\overline{A}_{t-\Delta t}(T_{r,t-\Delta t}-T_{j,t-\Delta t})}{\sum(\overline{m}_{l,t-\Delta t}Cp_l)}. \tag{5.28}$$

$$\frac{\partial f_{r,t-\Delta t}}{\partial EH_{loss}} = \frac{-1}{\sum(\bar{m}_{l,t-\Delta t} Cp_l)}. \tag{5.29}$$

$$\frac{\partial f_{j,t-\Delta t}}{\partial U} = \frac{\bar{A}_{t-\Delta t}(T_{r,t-\Delta t} - T_{j,t-\Delta t})}{\sum(\bar{m}_{l,t-\Delta t} Cp_l)}. \tag{5.30}$$

$$\frac{\partial f_{r,t-\Delta t}}{\partial EH_{loss}} = \frac{-1}{\sum(\bar{m}_{l,t-\Delta t} Cp_l)}. \tag{5.31}$$

$$\frac{\partial f_{j,t-\Delta t}}{\partial \lambda_0} = 0. \tag{5.32}$$

5.2.4 NPMBC Control Law

The development and on-line parameterization of the reactor temperature control model has been described. How is this model used for control?

Consider the reactor temperature at $T_{r,o}$. A set point change to a different temperature $T_{r,sp}$ is desired. This could represent either a set point change or an unmeasured disturbance rejection. Further assume that $T_{r,sp}$ must be attained in some time interval τ. Assuming that dT_r/dt remains constant during the time interval, a forward difference approximation of dT_r/dt can be made:

$$\frac{dT_r}{dt} = \frac{T_{r,sp} - T_r}{\tau}. \tag{5.33}$$

Substituting the right hand side of Eq.(5.10) for the derivative, Eq.(5.33) can be solved simultaneously to determine the control action which is to calculate the

jacket temperature set point $T_{j,sp}$:

$$T_{j,sp}=T_r+\frac{\Sigma m_l Cp_l (T_{r,sp}-T_r)}{\tau}+(\dot{m}_I Cp_I+\dot{m}_S Cp_S+\dot{m}_M Cp_M)(T_{amb}-T_r)+\frac{k_p}{V_{sys}}\lambda_0 n_M \Delta H_p - EH_{loss}. \quad (5.34)$$

Similarly, assume that $T_{j,sp}$ must be attained in some time interval τ. Again a forward difference approximation of dT_j/dt can be made:

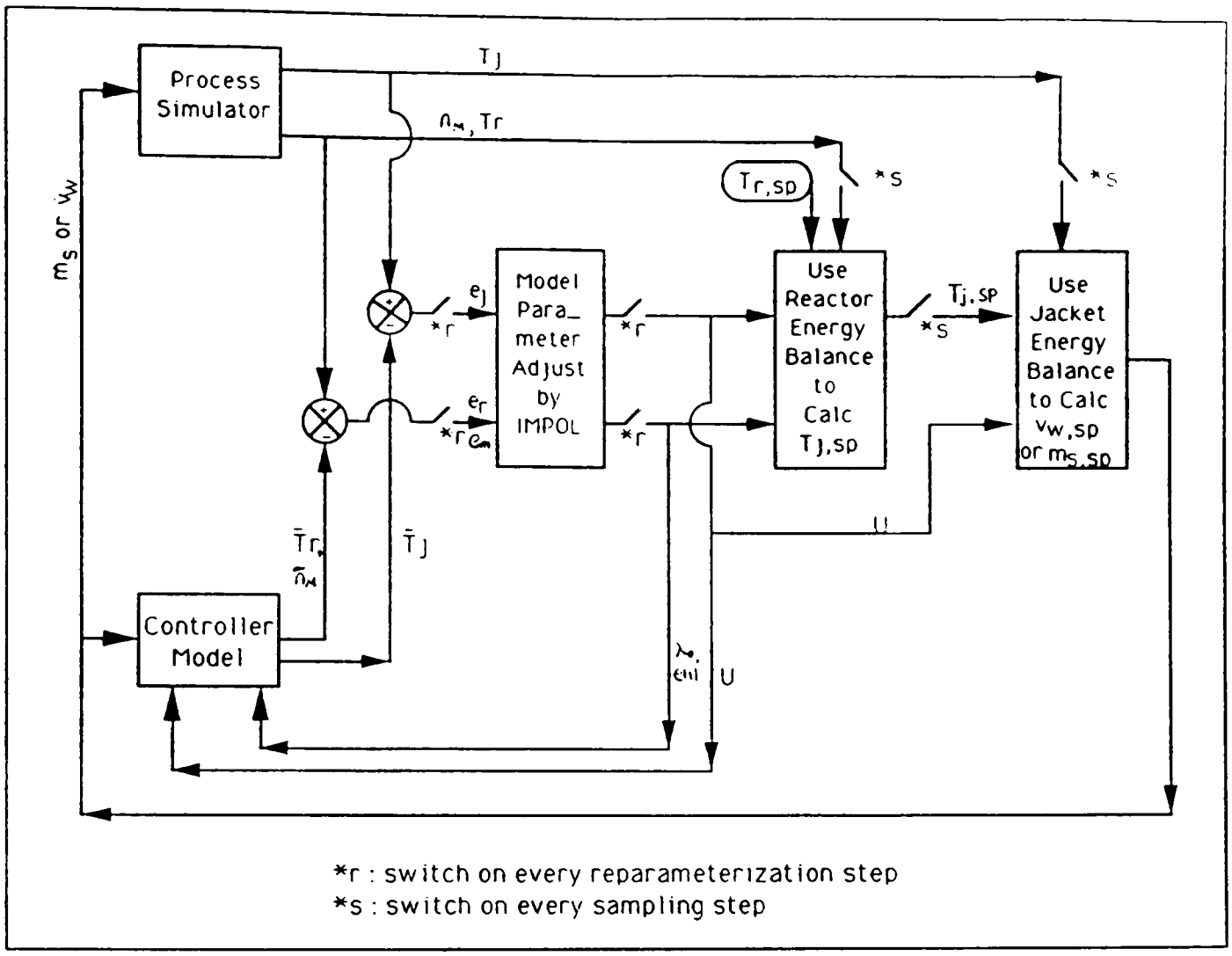
$$\frac{dT_j}{dt}=\frac{T_{j,sp}-T_j}{\tau}. \quad (5.35)$$

Substituting the right-hand side of Eq.(5.11) for the derivative, Eq.(5.35) can be solved to determine the control action which is to calculate the dump rate, $\dot{V}_{w,sp}$ or steam flow rate, $\dot{m}_{s,sp}$.

$$\dot{m}_{s,sp}=\frac{V_j Cp_w \rho_w (T_{j,sp}-T_j)/\tau - UA(T_r-T_j)}{\lambda_s + Cp_w (T^a-T_j)}. \quad (5.36)$$

$$\dot{V}_{w,sp}=\frac{V_j Cp_w \rho_w (T_{j,sp}-T_j)/\tau - UA(T_r-T_j)}{Cp_w \rho_w (T_{cw}-T_j)}. \quad (5.37)$$

If τ is a short time period, the process will be driven to the set point relatively quickly. If τ is large, the process response will be slower. Therefore, τ is a tuning parameter of the controller. It is a measure of the time to move to target. A separate PI flow controller is implemented for flow regulating to target. Figure 5.3 shows the block diagram for the NPMBC control structure.



Figrue 5.3
Block Diagram for NPMBC Control of Batch Reactor

5.2.5 Offset Compensation Using GMC

Because the process changes, the model parameters will change slightly over time. Otherwise, some process model mismatch will develop and result in controlled system offset. Offset can be removed by reparameterization. However, a temporary means of removing the steady-state offset in between parameterizations is to add an integral term to the control action equation. If there is process offset, the integral term accumulates and produces a response that eliminates the offset. This is the control law developed by Lee and Sullivan (1988) called generic model control. Thus control action with temporary offset correction shall be obtained from the generic model control law:

$$f(y,u,\bar{d},\bar{k})+K_1(y_0-y_{sp})+K_2\int_0^t(y_0-y_{sp})\,dt-0. \qquad (5.38)$$

Note that the integral term is reset to zero whenever the model parameters are updated.

5.3 Measure of Temperature Controllability

The temperature could be uncontrollable not because of poorly performing controllers but because of a constraint on the heat transfer. The constraint, which can be termed as maximum available heat transfer rate, is determined by the current heat transfer coefficient, heat transfer area and inlet cooling medium temperature. The maximum available heat

transfer rate changes with time because of the nonstationary characteristic of the overall heat transfer coefficient and heat transfer area.

In NPMBC for reactor temperature control, one can take advantage of the on-line estimated model parameters, ΣP_n ($=\lambda_o$) and the overall heat transfer coefficient (U), to monitor the temperature controllability which is defined as the ratio of the current heat generation rate to the current maximum available heat transfer rate:

$$Controllability = \frac{\frac{k_p}{V_{sys}} \lambda_0 n_M \Delta H_p}{UA(T_{cw}-T_r)}. \qquad (5.39)$$

It is normally considered uncontrollable if the ratio becomes greater than 0.5.

5.4 Filtering

The objective of a filter is to 'see' the process variable value through the process noise. In Figure 5.4 for example, if dots represent the noisy value, the dashed line would be the desired filtered value. The desired characteristics of the filtered value are that (1) it does not respond to noise (it does not indicate a process change when the process is noisy but stationary); (2) it reports a representative value of the process variable; and (3) it changes the filtered value rapidly when the process makes a real change.

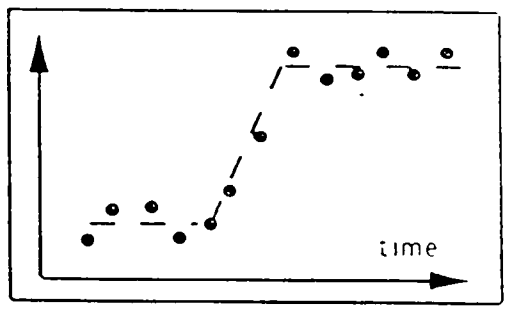


Figure 5.4
The Desired
Filtered Value

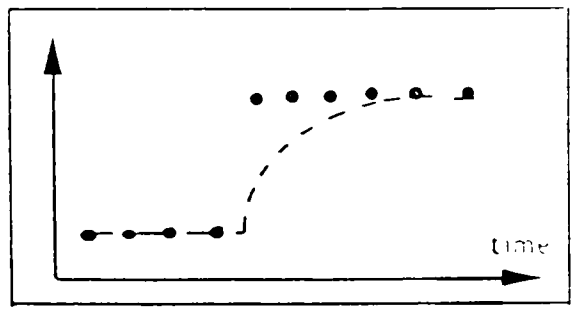


Figure 5.5
The Lagged
Filtered Value

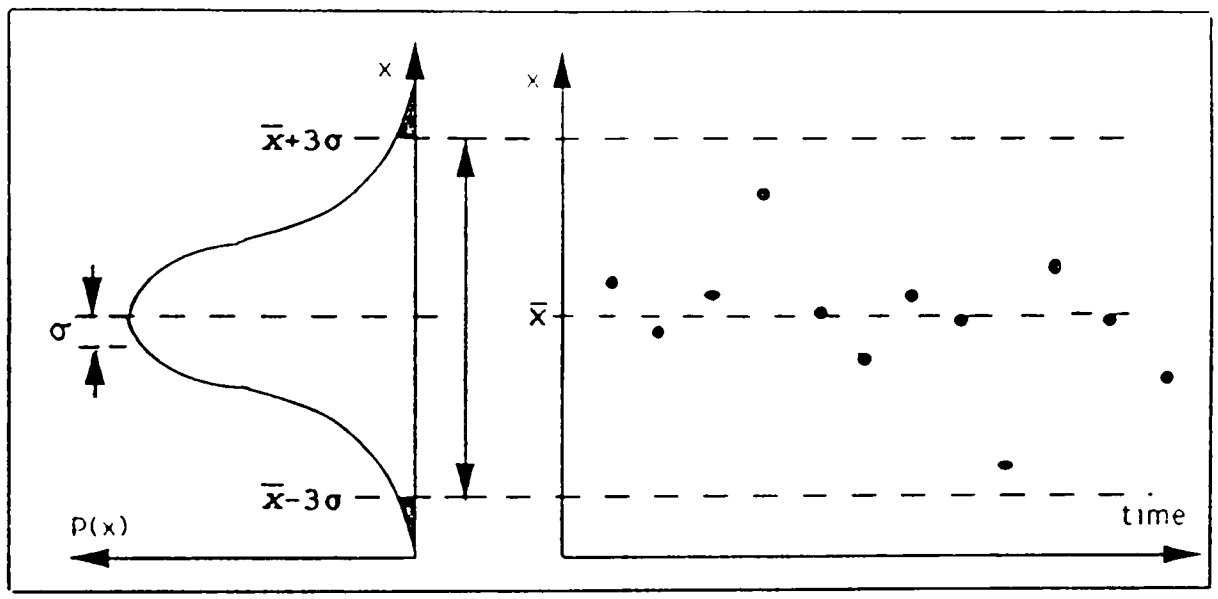


Figure 5.6
Probability Density Distribution of Stable Process

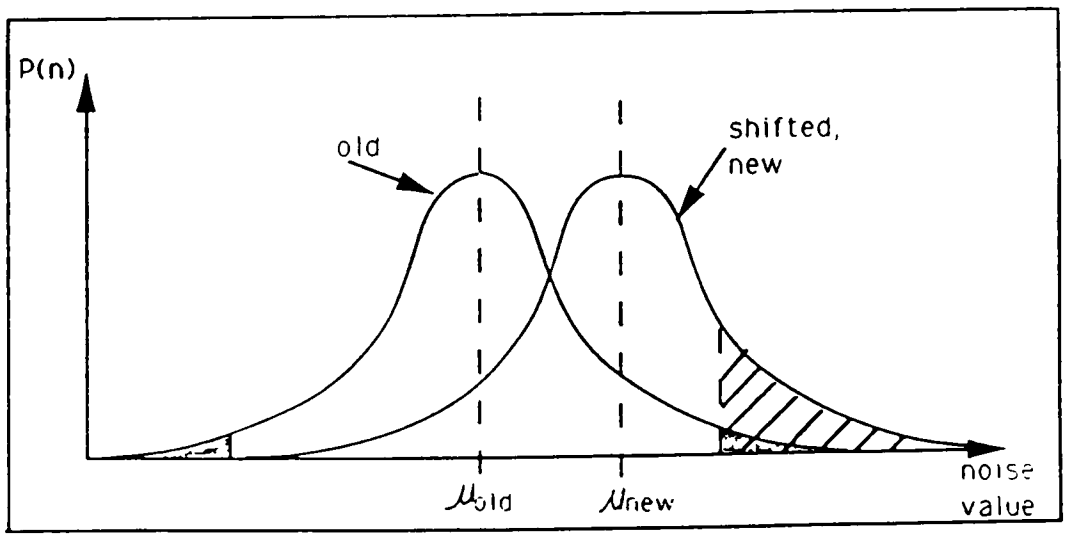


Figure 5.7
A Process Shift

5.4.1 Conventional Filter

Conventional filters are averaging type devices such as the first-order filter.

$$x_{f_i} = (1-e^{-T/\tau_f})x_i + (e^{-T/\tau_f})x_{f_{i-1}}. \qquad (5.40)$$

It is also called an exponentially weighted moving average(EWMA) when using λ for e^{-T/τ_f}

$$EWMA_i = (1-\lambda)\cdot EWMA_{i-1} + \lambda \cdot x_i. \qquad (5.41)$$

By averaging, objectives (1) and (2) are met. However, averaging causes the filtered signal to lag behind real changes (Figure 5.5). Thus objective (3) is not met.

5.4.2 SPC-based Filter

So, reconsider filtering. A stationary process with the probability density distribution (histogram) is shown in Figure 5.6. Most points are close to the average, but a few are far away. If the process is Gaussian (normally) distributed, 99.73% of the data will fall within the $\mu \pm 3\sigma$ range centered on μ. If the process is stationary, then there is only a 100%-99.73% = 0.27% chance of a data point appearing in the shaded tail area (Figure 5.6).

Alternatively, if the process has shifted to a new mean as shown in Figure 5.7, there is a much greater chance of finding a data value in the old tail area. So, if a point is

experienced outside of the $\mu \pm 3\sigma$ range it was because either (a) the process was stationary and it was one of those rare, but probable, 0.27% chance events or (b) the process shifted and there is a high percentage chance of seeing such a data point. If a data is outside of the old 3σ limits, it is probable to accept case b.

Rhinehart (1992) developed a technique to measure σ on line, to trigger the need for a filtered variable change, and to determine the magnitude of the change. One of the perspectives, fundamental to statistical process control (SPC) is to accept inherent process variability and to make changes only when there is a high (about 99.7%) statistical confidence that a change is justified. An SPC-manipulated variable would not respond to such normal variability. However, SPC rules direct neither how much change nor which process variables should be changed. Rhinehart proposed to credit the cumulative deviation as occurring at an average rate and to adjust the filtered value by that average. Applying such a philosophy to a Process Variable (PV), one would retain the previous SPC-filtered value until there was statistically sufficient evidence that the PV value had changed. A simple CUSUM procedure can be that indication. Further, once a significant change in level had been indicated, the SPC-filtered value can be deterministically adjusted by the CUSUM value.

There are various forms for the CUSUM. For this work it is useful to use one of the simplest forms and to define CUSUM as cumulative sum of the number of standard deviations that the PV has deviated from the SPC-filtered value:

$$CUSUM_i = CUSUM_{i-1} + (x_i - x_{spc_i}) / \hat{\sigma}_{pv_i} \qquad (5.42)$$

where x_{spc} is the SPC-filtered value of the PV, and $\hat{\sigma}_{pv_i}$ is the approximately calculated standard deviation of PV (for more detail derivation, see Rhinehart, 1992).

$$\hat{\sigma}_{pv_i} = \sqrt{\rho_{f_i}^2 / 2} \qquad (5.43)$$

where $\rho_{f_i}^2$ is a measure of the variance on x_i and calculated by:

$$\rho_{f_i}^2 = \frac{1}{N-1}(x_i - x_{i-1})^2 + \left(\frac{N-2}{N-1}\right)\rho_{f_{i-1}}^2. \qquad (5.44)$$

If the process is stationary, CUSUM will fluctuate about a value of zero with a variance of k, where k is the number of samples since the last change in x_{spc}. However, CUSUM could have an absolute value of greater than 3 ($\pm 3\sigma_{pv}$ cumulative deviation) by any one of several mechanisms. For example, after a period normal, near-zero behavior one $(x - x_{spc})/\hat{\sigma}_{pv}$ contribution could exceed the ±3 value. Under normal variability, there is a 0.27% chance of this happening. Alternately, if there was a real PV change, then

there is a much higher chance of this happening. When |CUSUM| > 3, elementary SPC philosophy accepts that something real has happened and that action is justified. Note that this trigger for action is based on an improbable process measurement. It rejects the null hypothesis of $x_i=x_{spc,i}$. It is not based on CUSUM, a nonstationary random variable for which the trigger would be $3\sqrt{k}$.

Summarizing the procedure:

Measure or calculate x_i, then obtain a measure of the variance on x_i.

$$\rho_{f_i}^2 = \frac{1}{N-1}(x_i-x_{i-1})^2+(\frac{N-2}{N-1})\rho_{f_{i-1}}^2.$$

From it the standard deviation,

$$\sigma_i = \sqrt{\rho_{f_i}^2/2}.$$

Accumulate the CUSUM.

$$CUSUM_i = CUSUM_{i-1}+\frac{x_i-x_{SPC}}{\sigma_i}.$$

Count.

$$K = K+1.$$

If the deviation is large

$$If\ \ (\ CUSUM_i > trigger\ )\ \ THEN$$

then adjust the filtered value

$$x_{spc} = x_{spc}+CUSUM_i\cdot\sigma_i/K.$$

and reinitialize terms

$$CUSUM_i = 0.$$

$$K = 0.$$

One tunes the filter by choosing the trigger value. If the trigger is 2 then action will occur sooner, but it will be more influenced by noise (5% of data in a normal steady-state process will have a $\pm 2\sigma$ violation). If the trigger is 4, then it will be less influenced by noise (only 0.006% of individual will violate the $\pm 4\sigma$ limit) but it will wait longer to take action. While a trigger of 3 is conventional, SPC limits equivalent from 2 to 4 sigmas are subjectly chosen to balance responsiveness and false alarms. This work used a trigger of 3.

This new SPC-based filter is not limited to use for process measurements, it can be applied to any calculated values which are based on unfiltered data. For the S.C. Johnson & Sons reactor control case, model parameter values calculated with noisy measured data are CUSUM filtered. The CUSUM of each parameter is independently triggered for this case. The results of SPC based-filter application to the S.C. Johnson & Sons reactor is as good as the conventional averaging type filter even if it has not shown any clear superiority. These results are presented in Appendix B.

CHAPTER VI

PROCESS OPTIMIZATION

The operating factors, like temperature setpoint trajectory and recipe loading and its schedule, influence the product properties and operating cost. These factors interact with each other. For instance, lower operating temperature gives higher average molecular weight, but it will take a longer time for a batch. Also, more solvent allows a better heat removal, but it broadens the molecular weight distribution by chain transfer and takes a longer reaction time. Thus, an optimum operating condition should be maintained throughout a batch time such that the product gets as close as possible to the desired specification, and simultaneously minimizes operating costs. The optimum temperature condition should not be a constant throughout a batch time because the process condition keeps changing. The entire batch reaction time, therefore, is divided into several stages according to the conversion (not divided by the time because the batch time for 100% conversion varies with choice of operating condition), and an optimization problem is created to find the optimum for decision variables for each stage. If one divides the reaction time into three stages, for instance, 0 to 40% conversion for stage I, 40% to 70% for stage II, 70% to 97% for stage III, and if monomer loading at the start-up and initiator loading

schedule, FI_I, FI_{II} and FI_{III} are prefixed, there will be three decision variables on the temperature setpoint, T_I, T_{II} and T_{III} and three decision variables on the solvent loading schedule FS_I, FS_{II} and FS_{III}. A typical optimization scheme is shown in Figure 6.1. The reactor simulator with the NPMBC controller "on", is run, and the objective function value is evaluated at 97% of conversion. For this study, the objective function has a form of:

$$OF = w_1(X_n - X_{n_{target}})^2 + w_2(PDI-1)^2 + w_3(TT97\%-0)^2 + w_4(\Sigma FS_i-0)^2. \qquad (6.1)$$

Eq.(6.1) reflects the desires of meeting a target average chain length, Xn, minimizing polydispersity index, reaction time, TT97% and the total amount of solvent supplied, ΣFS_i. The last term may not be included because the second and the third terms contribute to minimize the solvent loading. Even if the last term is deleted, one could make up the effect by assigning more weighting on the second and the third term. Here the process part of the reactor simulator is based on the moment method. One should carefully judge the weighting factors, $w_{1,2,3,4}$ considering the importance of each contribution in Eq.(6.1). The optimizer will search for the optimum decision variables which minimize the objective function value. The objective function can also reflect the desires for a specific MWD. If one wants the first 10 classes and last 300 classes out of, for instance, a total of 600 classes to have a certain fraction each, one should

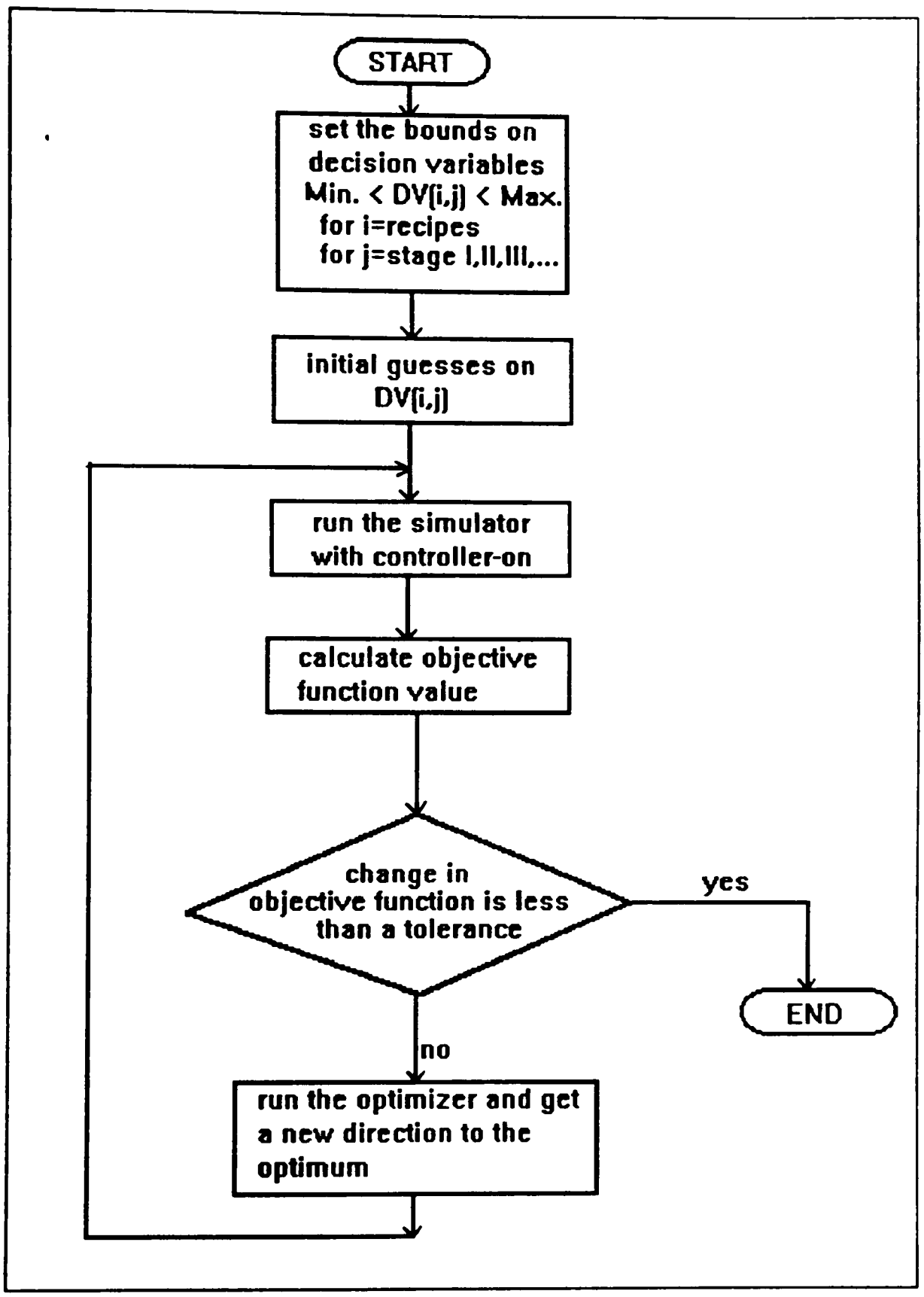


Figure 6.1
The Optimization Routine by Complex Method

use the class method to model MWD in the reactor simulator. In this case Eq.(6.1) shall be modified as

$$OF = w_1\left(\frac{\sum_{low\ tail} \bar{n}\cdot_n D}{\sum_{all\ n} \bar{n}\cdot_n D} - LF_{target}\right)^2 + w_2\left(\frac{\sum_{high\ tail} \bar{n}\cdot_n D}{\sum_{all\ n} \bar{n}\cdot_n D} - HF_{target}\right)^2 +$$

$$w_3(TT97\%-0)^2 + w_4(\Sigma FS_i-0)^2. \qquad (6.2)$$

The last term in Eq.(6.2) may not be included because the third term contributes to minimize the solvent loading. Even if the last term is deleted, one could make up the effect by assigning more weighting on the third term.

The maximum possible heat removal rate by the cooling jacket during the batch reaction time is limited by the temperature of the chilled water supplied (T_{cw}), the overall heat transfer coefficient (U) and the heat transfer area (A). So, it is calculated by $U*A*(T_r-T_{cw})$. It is possible, during the optimizer search, for certain values of decision variables to lead to insufficient solvent supply or very high reaction temperature. Consequently this condition causes a very high mixture viscosity or a very high reaction rate so that the heat generation rate is higher than the maximum possible heat removal rate. If so, temperature controllability drops and an excursion of reactor temperature occurs. It is a general practice for manipulated variables to have twice the normally required capacity. Here, the reactor cooling system must have a ratio of heat

generation rate over the maximum possible heat removal rate of less than 1/2. This rule is employed in the optimization program to act as an indirect constraint to the optimizer.

As an optimizer, an IMSL(1982) subroutine, DBCPOL is employed. DBCPOL, a constraint optimization technique, minimizes a function of N variables subject to bounds on the variables using a direct search "complex" algorithm. Since DBCPOL uses only function-value information at each step to determine a new approximate minimum, it could be inefficient compared to other methods which take into account derivative information at each iteration. The gradient method, however, was difficult to employ because at a certain set of decision variables, the gradient calculation by the finite difference method requires the functional evaluation by the process simulator which is often impossible due to the constraint on the reactor cooling system. When it happens with the complex method, the program assigns a reasonably big penalty value to the current decision variables and return it to the optimizer. By doing so, one keeps the optimizer from moving in that direction again.

In the complex algorithm, a set of 2*N points in an N-dimensional space is called a complex. The method is based on function comparison. It starts with 2n points x_1, x_2, ... , x_{2n}. At each iteration, a new point is generated to replace the worst point x_j, which has the largest function value among these 2n points. The new point is constructed by

the following formula:

$$x_k = c + \alpha(c - x_j). \qquad (6.3)$$

where $c=\frac{1}{2n-1}\Sigma_{i\neq j} x_i$ α ($\alpha>0$) is the reflection coefficient.

When x_k is a best point, that is, when $f(x_k) \leq f(x_i)$ for i=1, ... , 2n, an expansion point is computed $x_e = c + \beta(x_k - c)$, where β ($\beta > 1$) is called the expansion coefficient. If the new point is a worst point, then the complex would be contracted to get a better new point. If the contraction step is unsuccessful, the complex is shrunk by moving the vertices half-way toward the current best point. Whenever the new point generated is beyond the bound, it will be set to the bound. This procedure is repeated until one of the following stopping criteria

$$\textit{Criterion 1:} \quad f_{best}-f_{worst} \leq \epsilon_f(1 + |f_{best}|).$$

$$\textit{Criterion 2:} \quad \sum_{i=1}^{2n}(f_i - \frac{\sum_{j=1}^{2n} f_i}{2n})^2 \leq \epsilon_f.$$

is satisfied. Where $f_i=f(x_i)$, $f_i=f(x_j)$, and ϵ_f is a tolerance and is given by $1*10^{-5}$.

CHAPTER VII

RESULTS AND DISCUSSION

This chapter covers (1) review of the Challenge Problem results and the results and discussion of (2) modeling of molecular weight distribution, (3) application of NPMBC strategy for the industrial scale reactor, whose process described in Chapter III, (4) optimization for polymerization process.

The controller uses a model of the process. But, in real life the process is not known exactly, so the controller model would necessarily be different from the process. Here, though, our process is simulated; so, we exactly know it. To simulate real life we intentionally made the controller model different from the process simulator. This study uses two reactor simulators, the S.C Johnson and Sons reactor process in Appendix A and the industrial scale reactor process in Chapter III. Table 7.1 shows how each of controller models used differs from the simulator. Note that each of the following result is based on the specific type of model given in Table 7.1

7.1 Review of the Challenge Problem Results

The challenge problem expresses some severe control issues (nonlinear, nonstationary), yet it specifies a very simple first order reaction mechanism: monomer becomes

Table 7.1
Summary of Process/Control Models Used

APLICATION		POLYMERIZATION PROCESS USED FOR REACTOR SIMULATOR	PROCESS MODEL	TEMPERATURE CONTROL MODEL
MWD Model	Moment	Baillagou & Soong, 1985	PM1*1	---
	Class	Baillagou & Soong, 1985	PM2*2	CM1*8,9
Temperature Control		Chylla & Hasse, 1990	PM3*3	CM2*9
		Chylla & Hasse, 1990	PM4*4	CM3*9
		Baillagou & Soong, 1990	PM5*5	CM4*8,9
Optimization		Baillagou & Soong, 1990	PM6*6 PM7*7	CM4*8,9

Remarks
*1: The process described in Section 3.1 is incorporated with isothermal operation and with moment equations as MWD modleing. No reactor/jacket configurations are included.
*2: The process described in Section 3.1 is incorporated with class balance equation as MWD modeling. However, the reactor/jacket configuration data in section 3.2 are modified for a pilot scale.
*3: The original S.C. Johnson and Sons reactor process is incorporated.
*4: The original S.C. Johnson and sons reactor process is modified. Conversion is considered as measurable. Noisy process measurement is incorporated.
*5: The process described in sections 3.1 & 3.2 is incorporated except for the polymer species balance equations. Noisy process measurement is incorporated.
*6: The process described in section 3.1 & 3.2 is incorporated with moment balance equations as the MWD model. PI flow controller/valve dynamics, however, are not included. It is assumed that steam/CW flow rate reaches a new setpoint as it is changed. Recipe loading is performed instantaneously when new stage begins.
*7: Same as the *6 condition except class balance equations used as the MWD modeling.
*8: MWD modeling equations are not incorporated.
*9: The control model was designed with several forms of mismatches over the process model.

polymer. There is no independent measure of reaction conversion. Further, there is no noise or lags in valve action, and both of these process realities confound control action.

The controller uses a phenomenological dynamic process model to make control decisions. Intentionally, the model was designed with several forms of mismatch over the process. The results of application of NPMBC strategy for the S.C. Johnson & Sons Reactor (the AIChE Challenge Problem) are presented in Appendices A & B.

In Appendix A, the problem is solved by exactly following the given statements, and the control results are presented. The simulation results are based on "PM3" and "CM2" in Table 7.1 as a process and a control model respectively. The entire 2-1/2 hours process and control (and much I/O) simulation took about 3 minutes execution time on a 25MHz 386 based PC. The implication is that the control effort would require less than 0.5% of controller time in a real-time run. The control could easily be executed in contemporary micro-processor based programmable controllers.

Simulated control of the batch polymerization challenge problem shows that NPMBC can easily meet the performance criteria. Conventional cascade strategy cannot because a sluggish control response is caused by shifting of the slave output to either of the final element which lags behind the

master output due to the reset windup on the master controller. Even when an anti-windup technique called "Internal Reset Feedback" is used, PID based control cannot accommodate the nonlinear/nonstationary controller performance. Conventional control is still not sufficient to meet the control performance criteria which is to maintain reactor temperature within 1°F from setpoint.

NPMBC control is robust to a wide range of user choices in tuning parameters, choice of adjustable model parameters, choice of sampling and parameter adjustment interval, and intentional process/model mismatch. In this work the controller model was adjusted incrementally at each parameterization interval. Parameterizing the multivariable nonlinear model on-line is effective and simple both computationally and implementationally.

In Appendix B, some modifications are made to the original problem statement. They include an additional process measurement of the monomer conversion, and noisy process measurements. Both the conventional filtering and the SPC-based filtering developed in Section 5.4 are implemented, and their results are presented. The simulation results presented in Appendix B are based on "PM4" and "CM3" in Table 7.1 as a process and a control model, respectively.

NPMBC met the controlled system performance criteria with the influence of noisy and drifting process measurement data. The controller demonstrates its robustness when the

process noise is filtered by either the CUSUM-filter or the conventional first-order filter. A simple and computationally inexpensive method for obtaining an on-line measure of process variance has been employed. A CUSUM procedure is offered as a means to filter a process variable in a manner consistent with the SPC philosophy. The technique is illustrated as applied to the NPMBC output, and demonstrates a good balance of filtered model parameters and the process response. However, in this work it shows no control performance advantages over a standard first-order filter.

The framework of the controller model now reflects the additional process measurement, the monomer conversion. Even so, the model still retains the many forms of mismatches over the process as detailed in "CM2" in Table 7.1.

Here, it is shown that NPMBC control is implemented not only for the stage where reaction is governing the process, but also for the initial heat-up stage. Hence, the entire batch sequence is run by a consistent control strategy even though the control objectives differ in each stage. Total batch control (start-up to finishing) can be achieved with a single control strategy. The integral part of the GMC law provides a second-hand adjustment to the controller in between the reparameterization steps.

The control strategy works in spite of the confounding effects of noise and measurement drift. The temperature

control objectives are met. The model parameters retain phenomenologically valid values. The system performance was sensitive to the choice of tuning parameters, but not to the trigger values.

7.2 Moment Model Results and Discussion

The solutions of the differential equations in Appendix D allow one to calculate the conversion, the moments of the growing polymer and the moments of the dead polymer as the reaction proceeds, and consequently the distribution parameters (average, variance, skewness and kurtosis).

The reactor simulator used to generate the results in this section adopted "PM1" in Table 7.1 as a process model. There is no controller. The simulator is held at isothermal conditions.

Factors which affect the shape of MWD are temperature, each recipe loading and the loading schedule. Although I have not presented extensive simulation results for the entire spectrum of these factors, the results shown here will guide in deciding optimum operating conditions which gives a desirable combination of MWD parameters, conversion, and reaction rate.

The moment-model-based process simulator is run for a base case. It is then compared with cases which are run for different operating conditions. The base case is chosen as:

Initiator at t=0 : 0.01548 kgmole

Monomer at t=0 : 8.43769 kgmole

Solvent at t=0 : 0.0 kgmole

Temperature : 70 deg C

Run period : 180 minutes.

The results of the base case run are shown in Figures 7.1 to 7.24. Figure 7.1 shows the conversion versus time graph. A very sharp increase in conversion is observed during the time period 50 to 60 min. The "end of run" call is made at 97% of monomer conversion. Figure 7.2 shows the sum (kgmole) of growing polymer, ΣP_n ($=\lambda_o$) versus time graph. It starts to increase at time point 63 min (75% conversion) because the high viscosity of the mixture begins to inhibit the mobility of long chain growing polymers and make them difficult to be terminated, and while initiator keeps creating small chain growing polymer. Since short chain growing polymers will preferentially find the residual monomer, low molecular weight polymers dominate the system at the end of the run. Thus, number-AMW decreases very sharply at this conversion stage as shown in Figure 7.3. No such dramatic change occurs on weight-AMW as shown in Figure 7.4 because weight-AMW reflects the importance of long-chain polymer species. The decrease of number-AMW is reflected by the rapid increase of PDI in Figure 7.5. Also, note that maximum molecular weight is observed at a conversion of

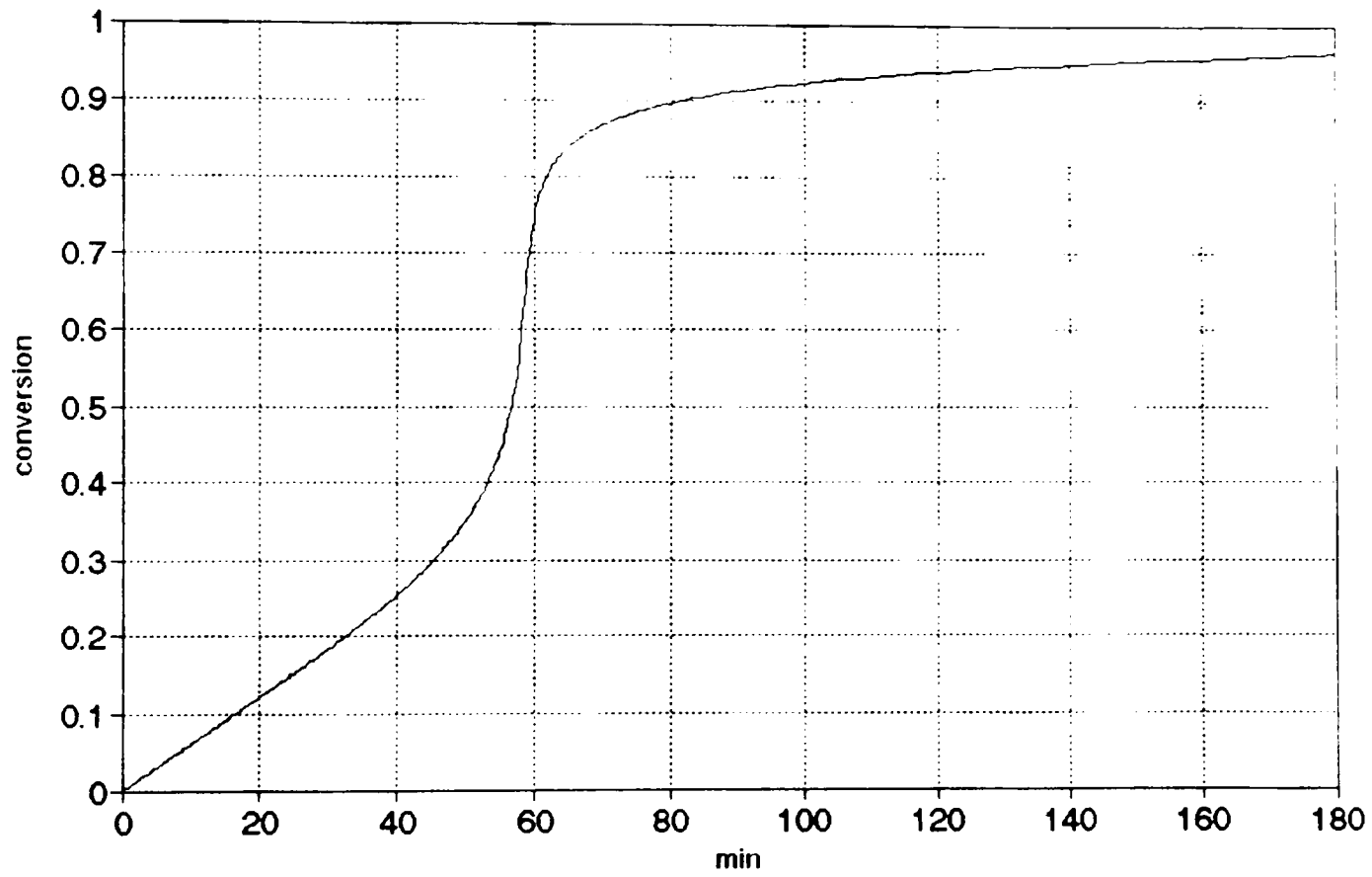


Figure 7.1
Conversion versus Time (Base Case)

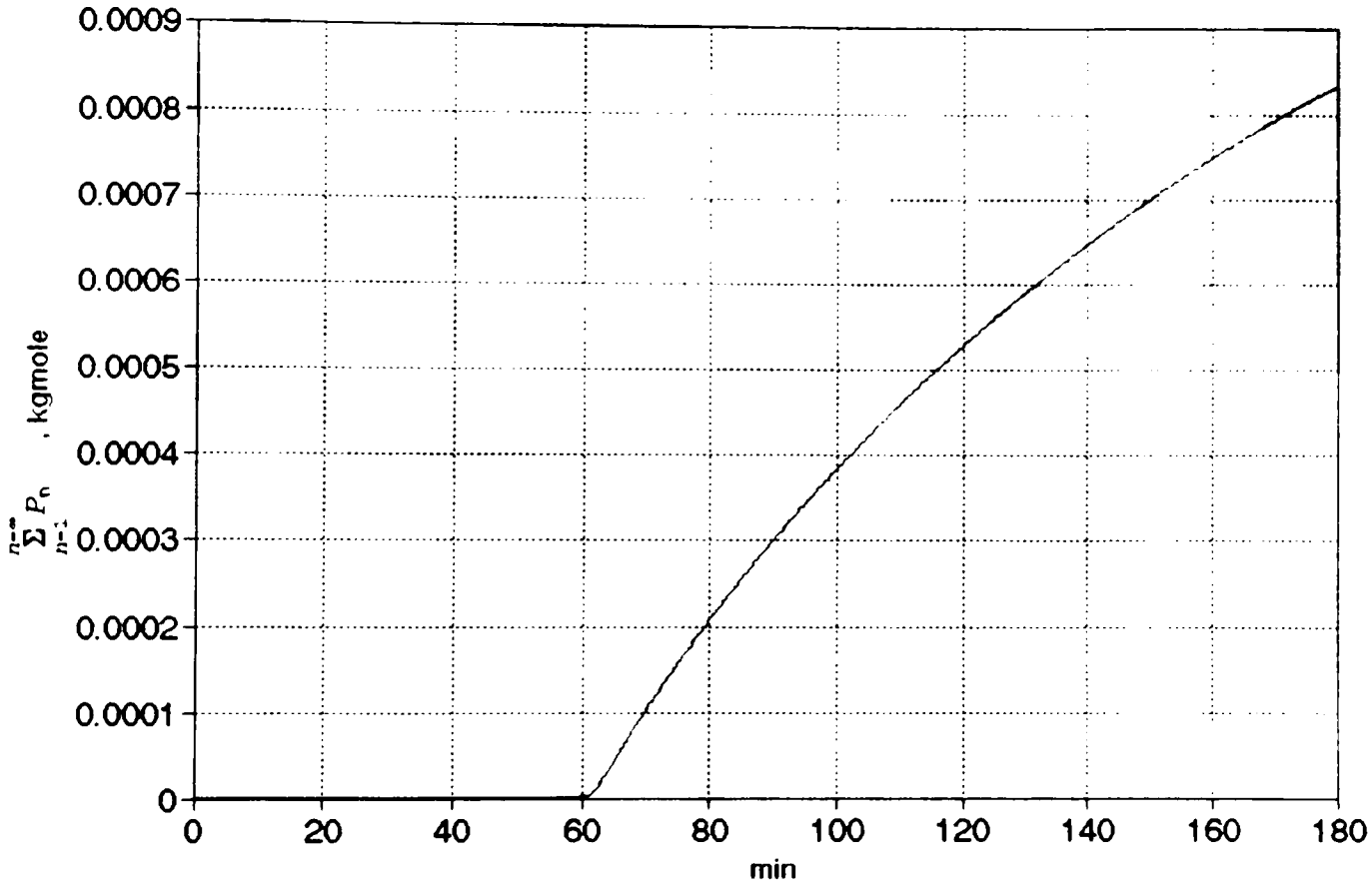


Figure 7.2
ΣP_n (λ_o) versus Time (Base Case)

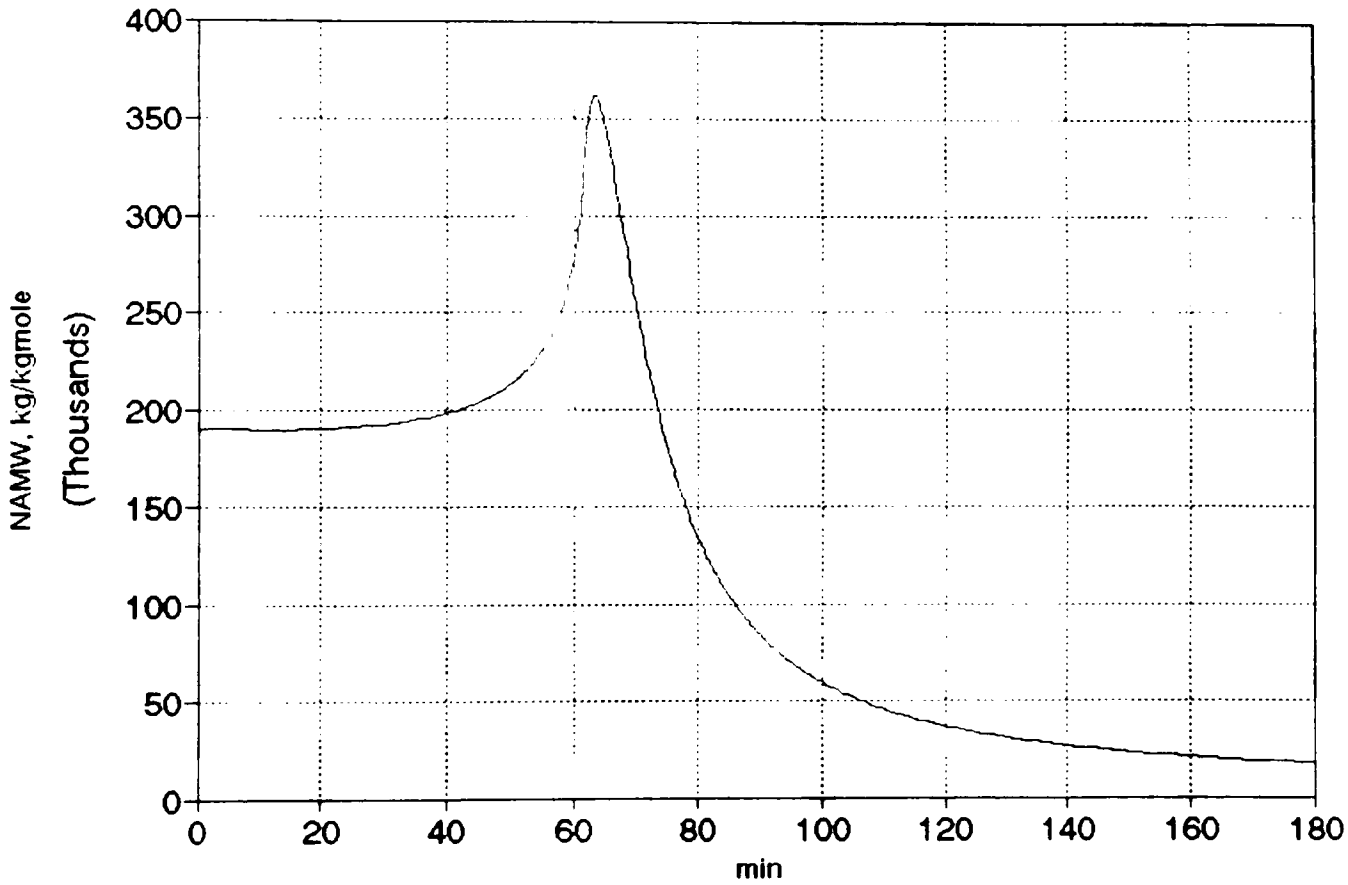


Figure 7.3
Number-AMW versus Time (Base Case)

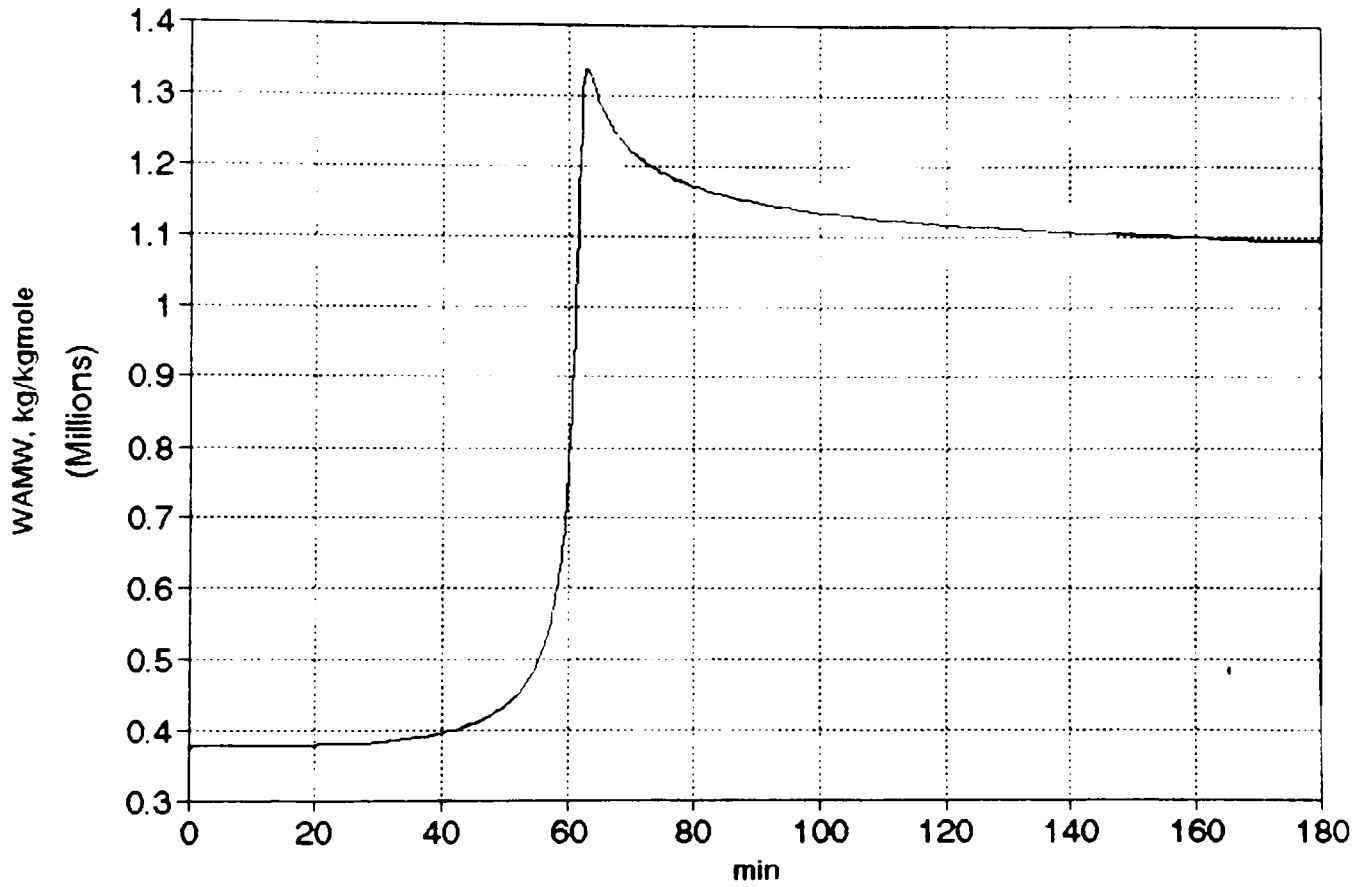


Figure 7.4
Weight-AMW versus Time (Base Case)

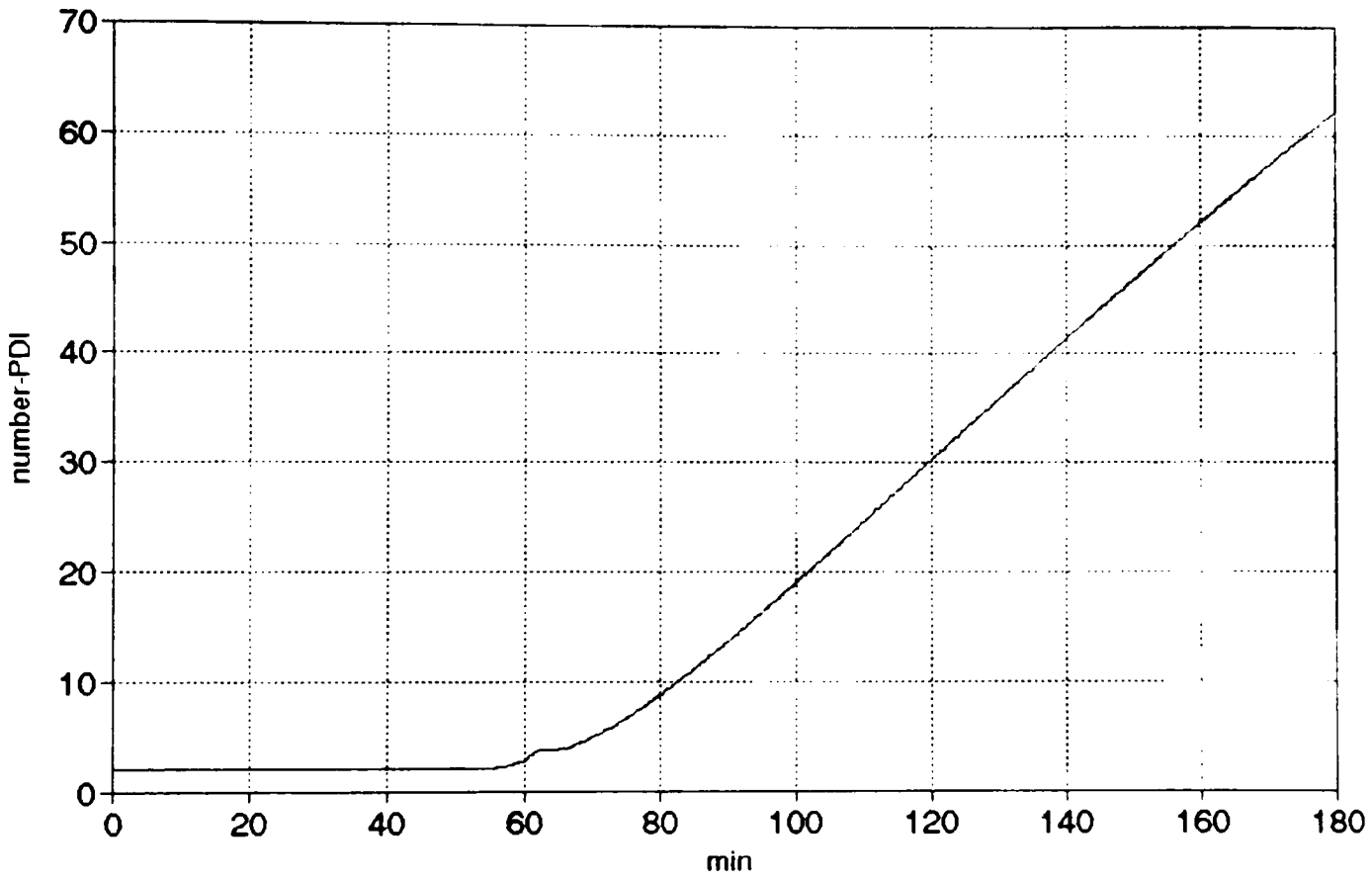


Figure 7.5
Number-PDI versus Time (Base Case)

about 80%. Number and weight-SD versus time graphs are shown in Figures 7.6 and 7.7, respectively. Note that they increase sharply at the same time that PDI begins to increase. Then the number-SD gradually decreases due to the dominating increase in low molecular weight species. Figures 7.8 and 7.9 show the number and weight-skewness. The positive skewness suggests that MWD is severely skewed with a high molecular weight tail at the end of the run. Figure 7.10 and Figure 7.11 show the number and weight-kurtosis, respectively. The positive kurtosis suggests that the peak of MWD is very sharp at the end of run.

The base case is run for different temperatures, 60 °C and 80 °C. The results are shown in Figures 7.12 to 7.14. The results suggest that high temperature is favorable for faster conversion while it is not desirable for high molecular weight and low PDI. It is due to the reason that, at high temperature, the initiator decomposes into initiator radicals in excess and consequently convert more monomer into growing polymer seeds. Therefore, there are more chances for the termination reaction to occur between the short chain-length growing polymer radicals, and results in low number-AMW and high number-PDI.

Figure 7.15 demonstrates the influence of initiator loading to the average MW. The higher initiator loading creates more initiator radicals and provides more chances for monomer to add to the initiator radicals rather than to

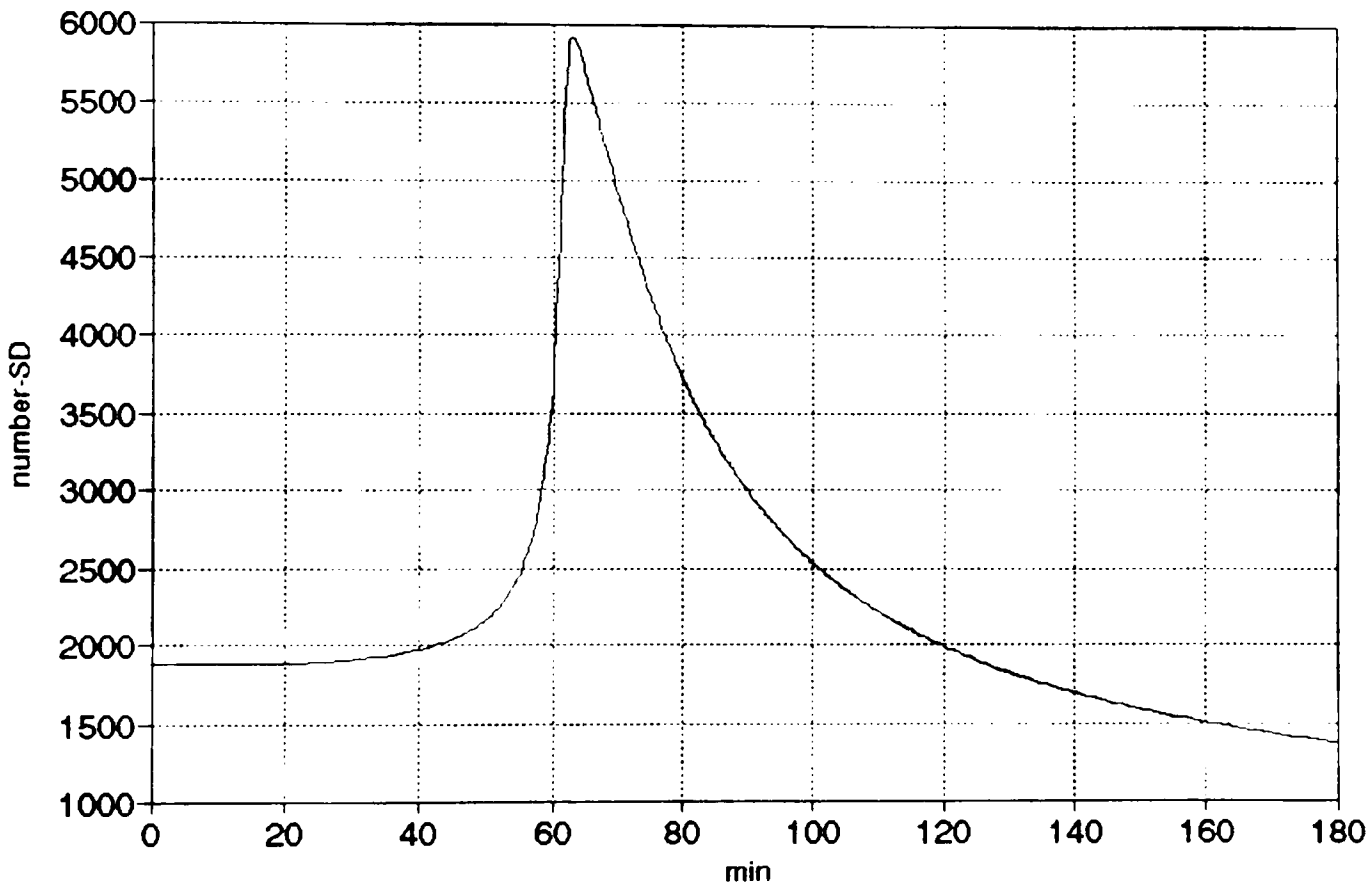


Figure 7.6
Number-SD versus Time (Base Case)

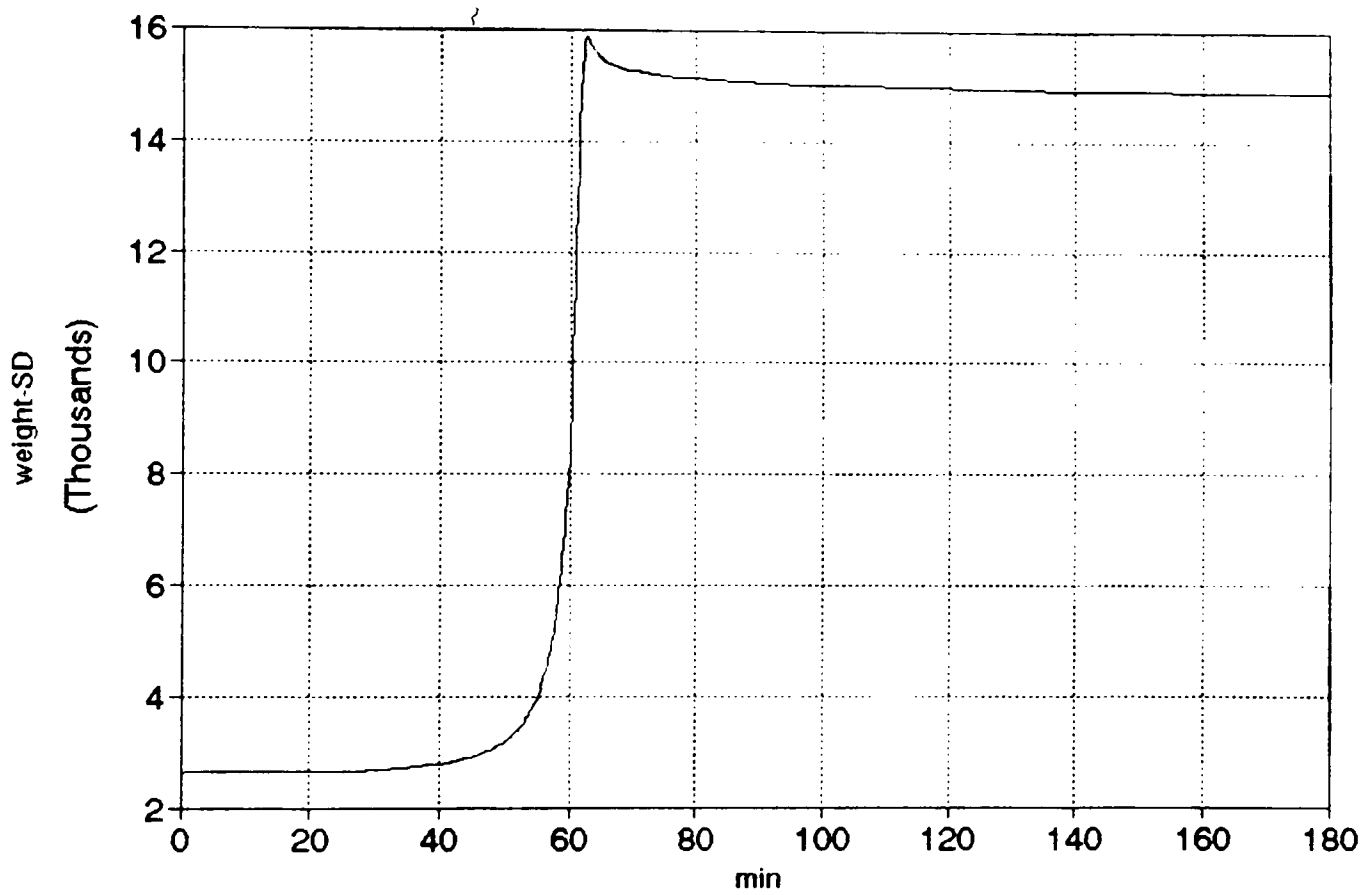


Figure 7.7
Weight-SD versus Time (Base Case)

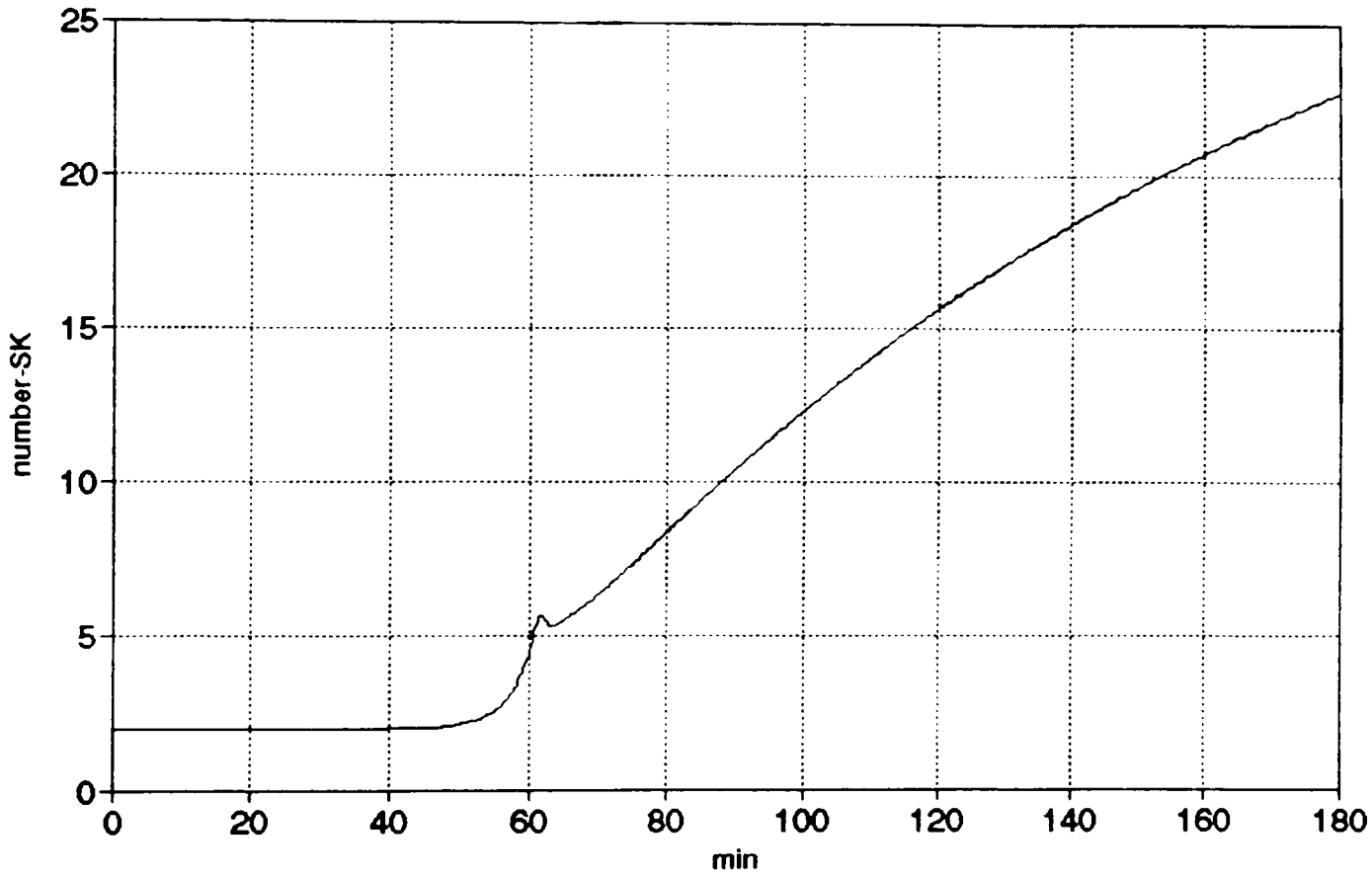


Figure 7.8
Number-SK versus Time (Base Case)

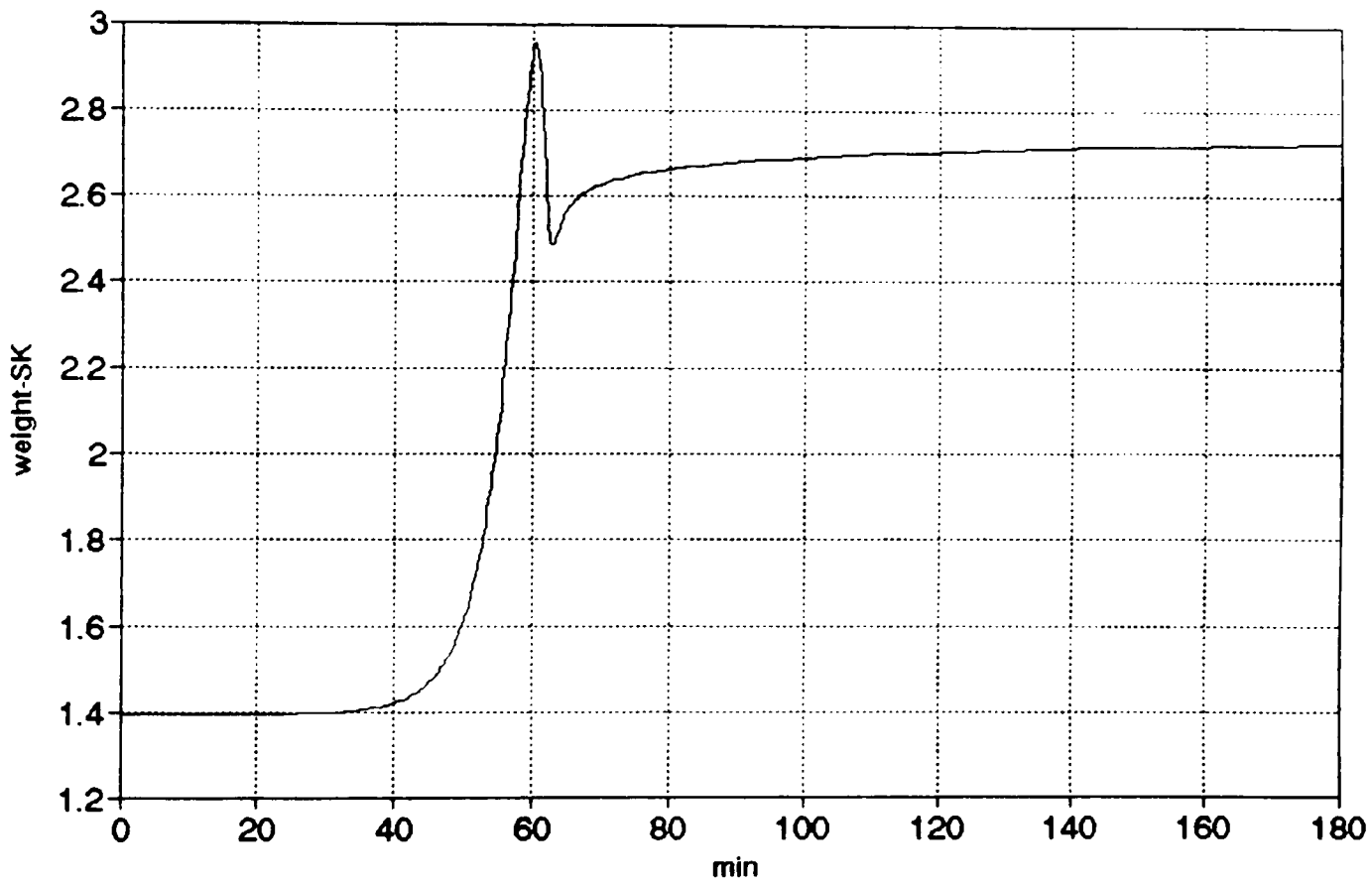


Figure 7.9
Weight-SK versus Time (Base Case)

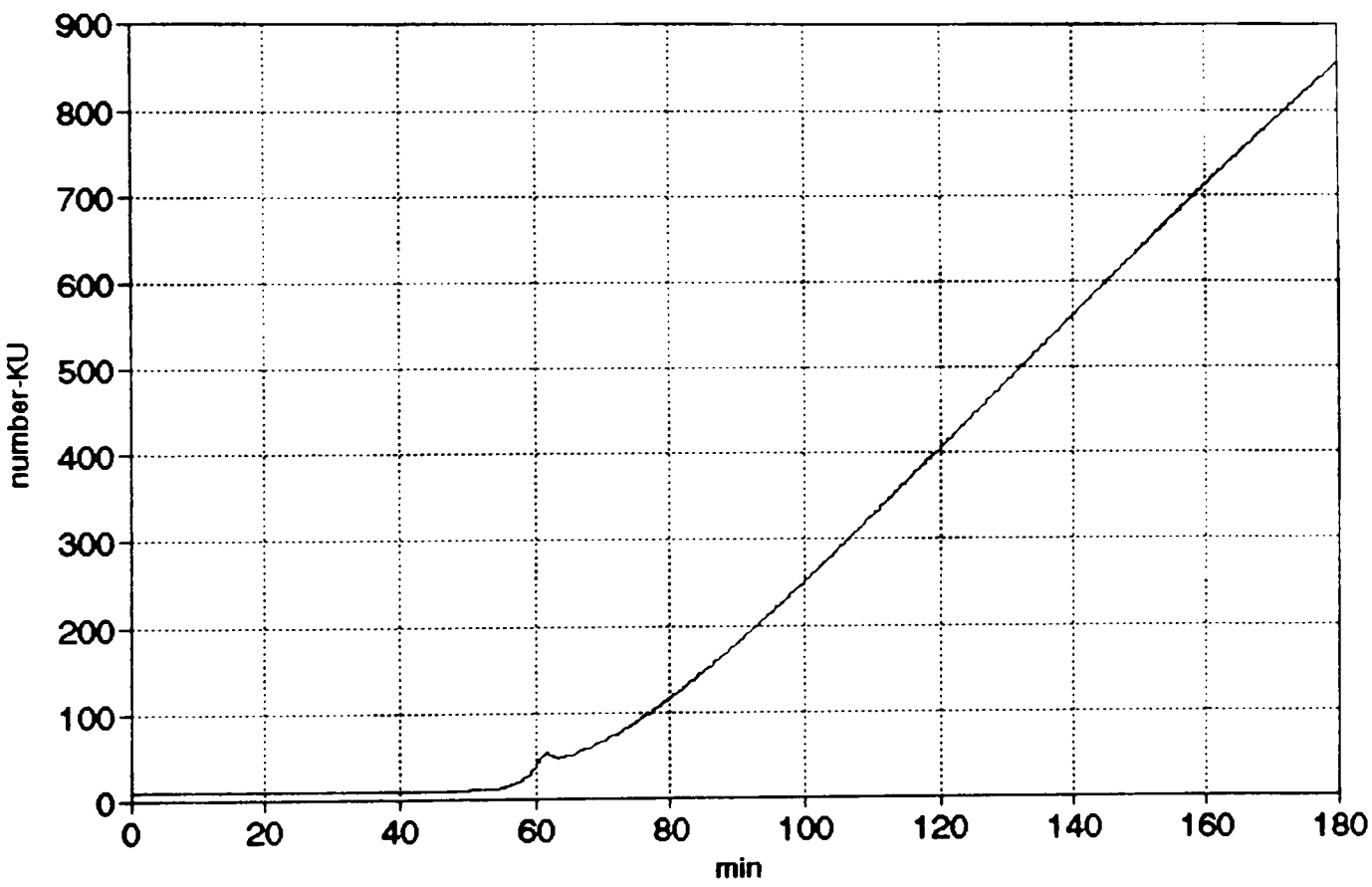


Figure 7.10
Number-KU versus Time (Base Case)

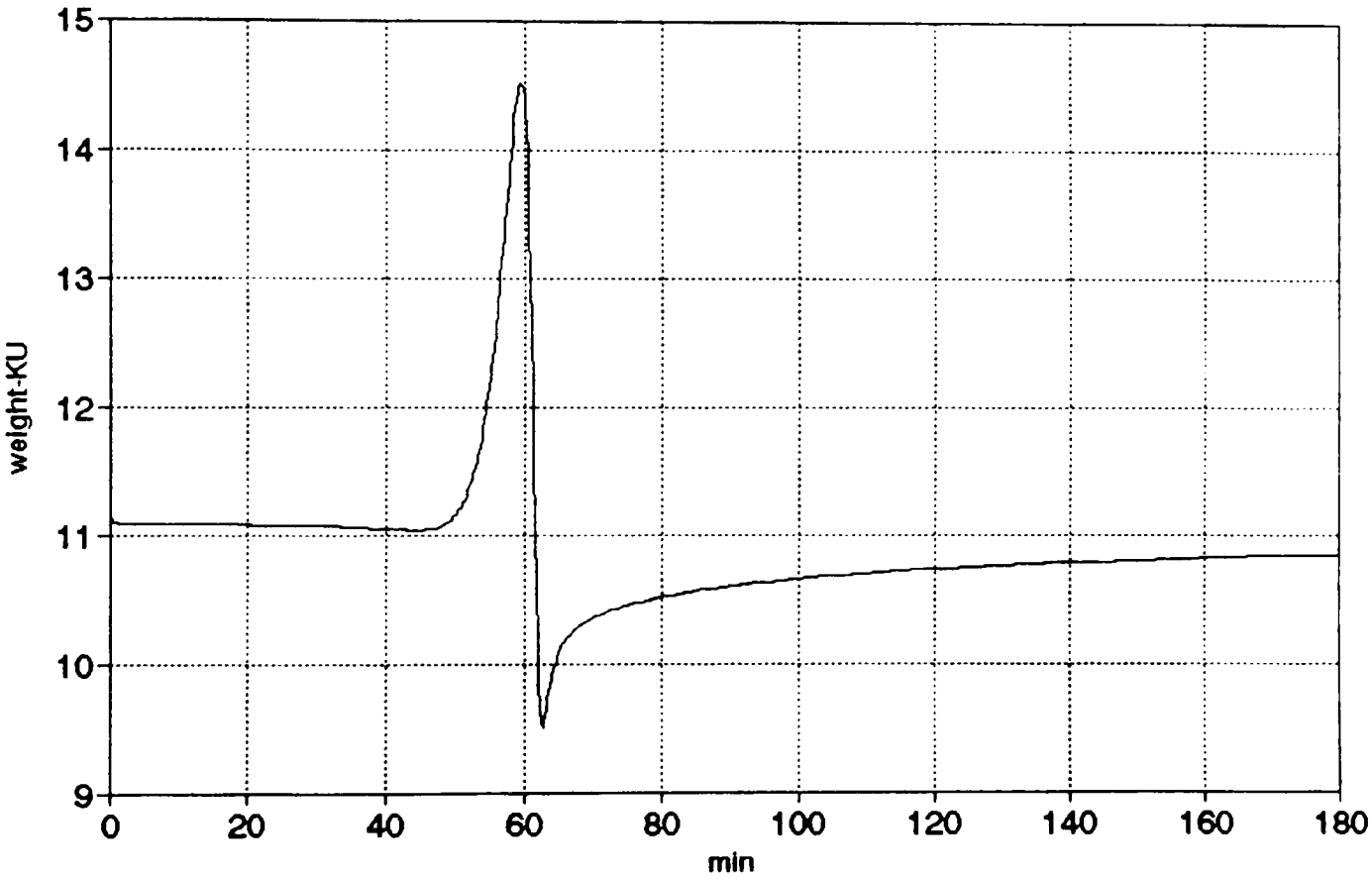


Figure 7.11
Weight KU versus Time (Base Case)

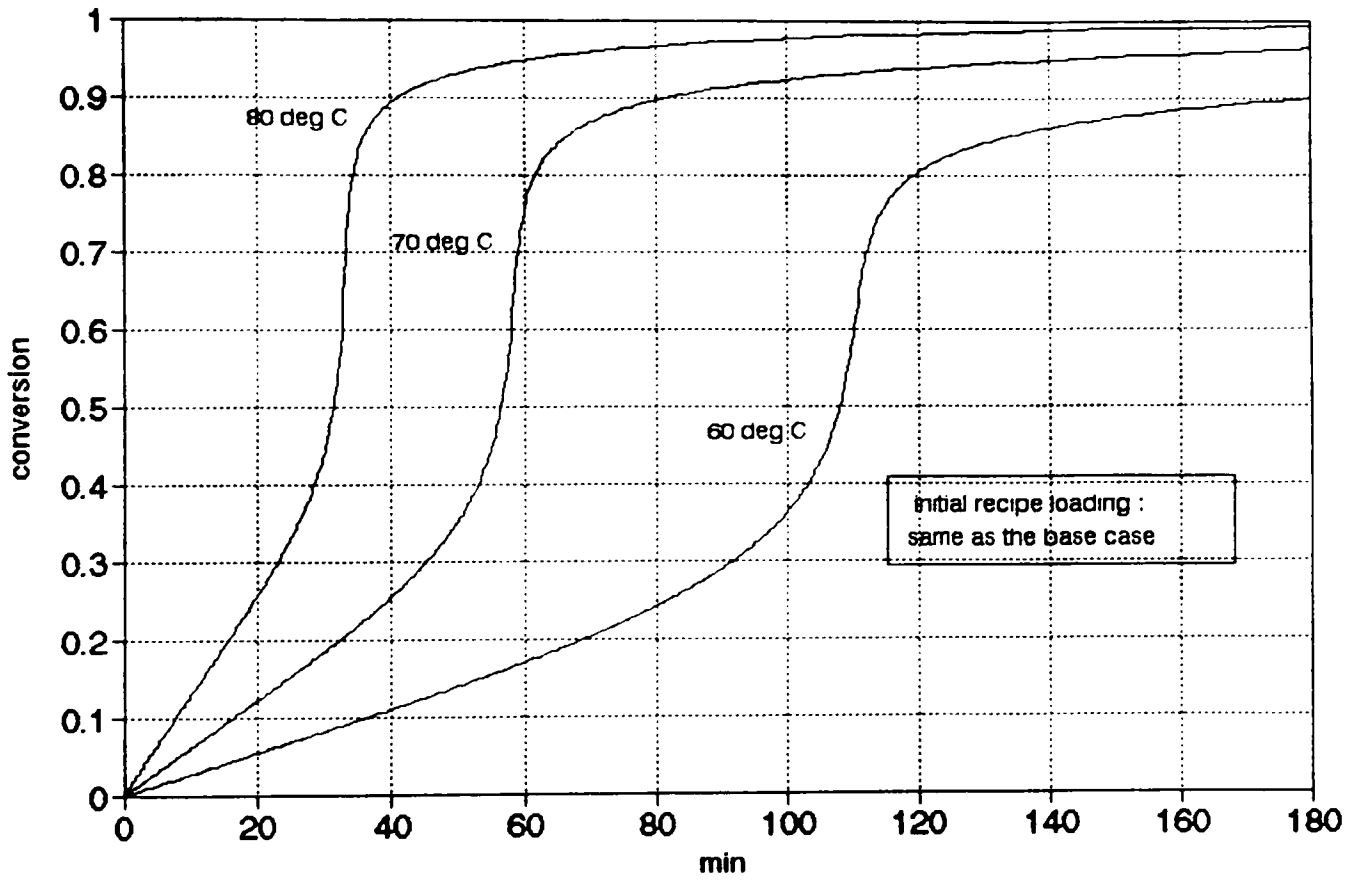


Figure 7.12
Temperature Effect to Conversion

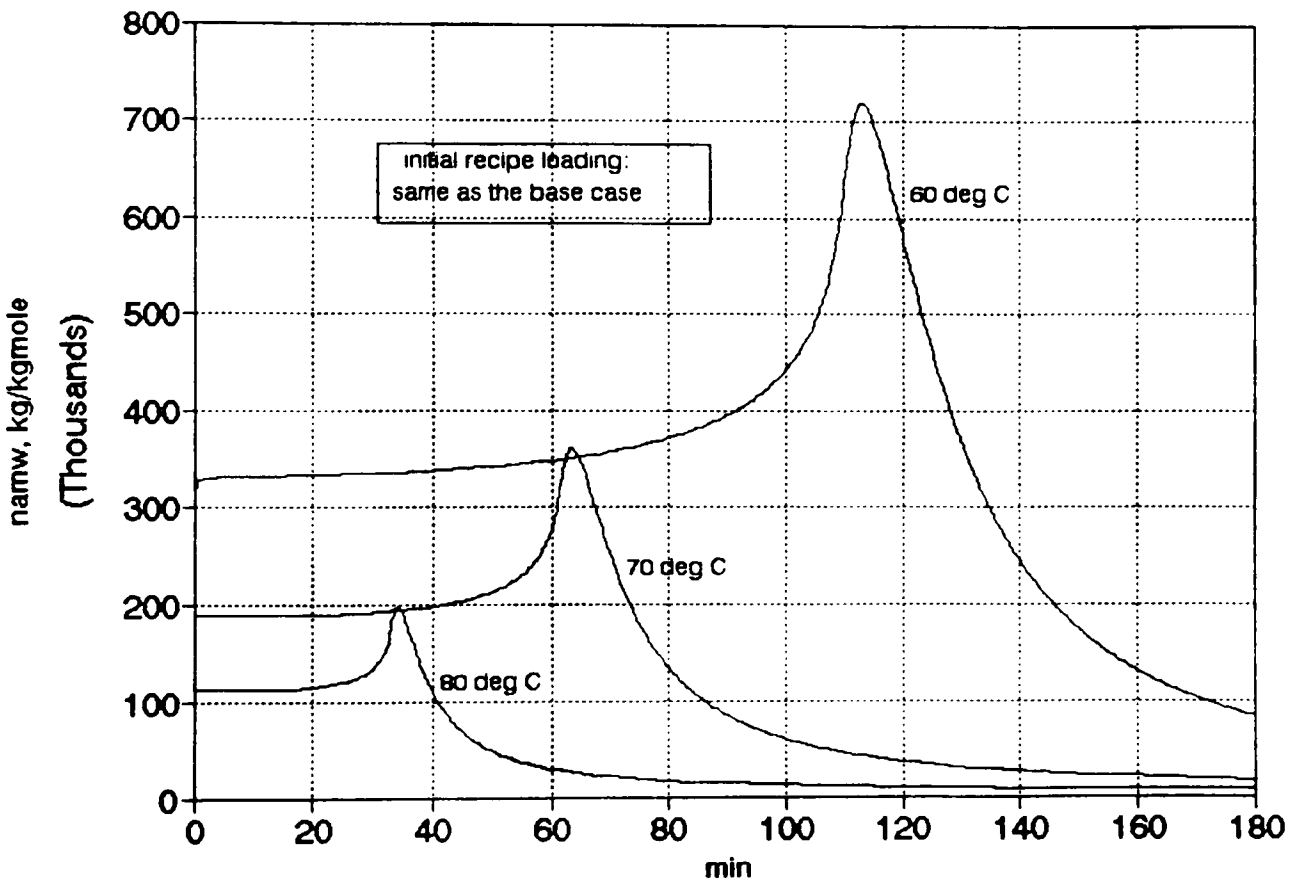


Figure 7.13
Temperature Effect to NAMW

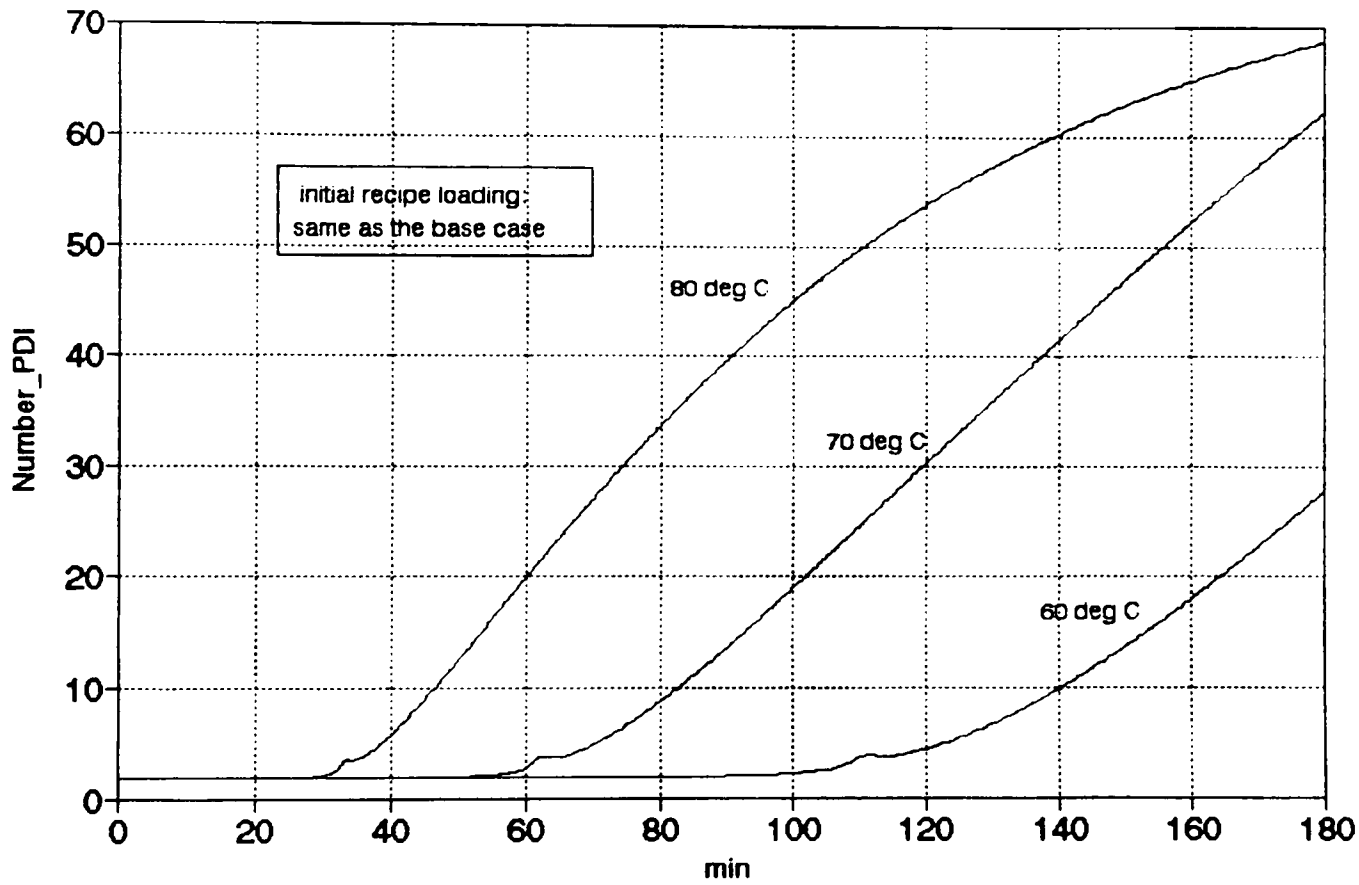


Figure 7.14
Temperature Effect to Number-PDI

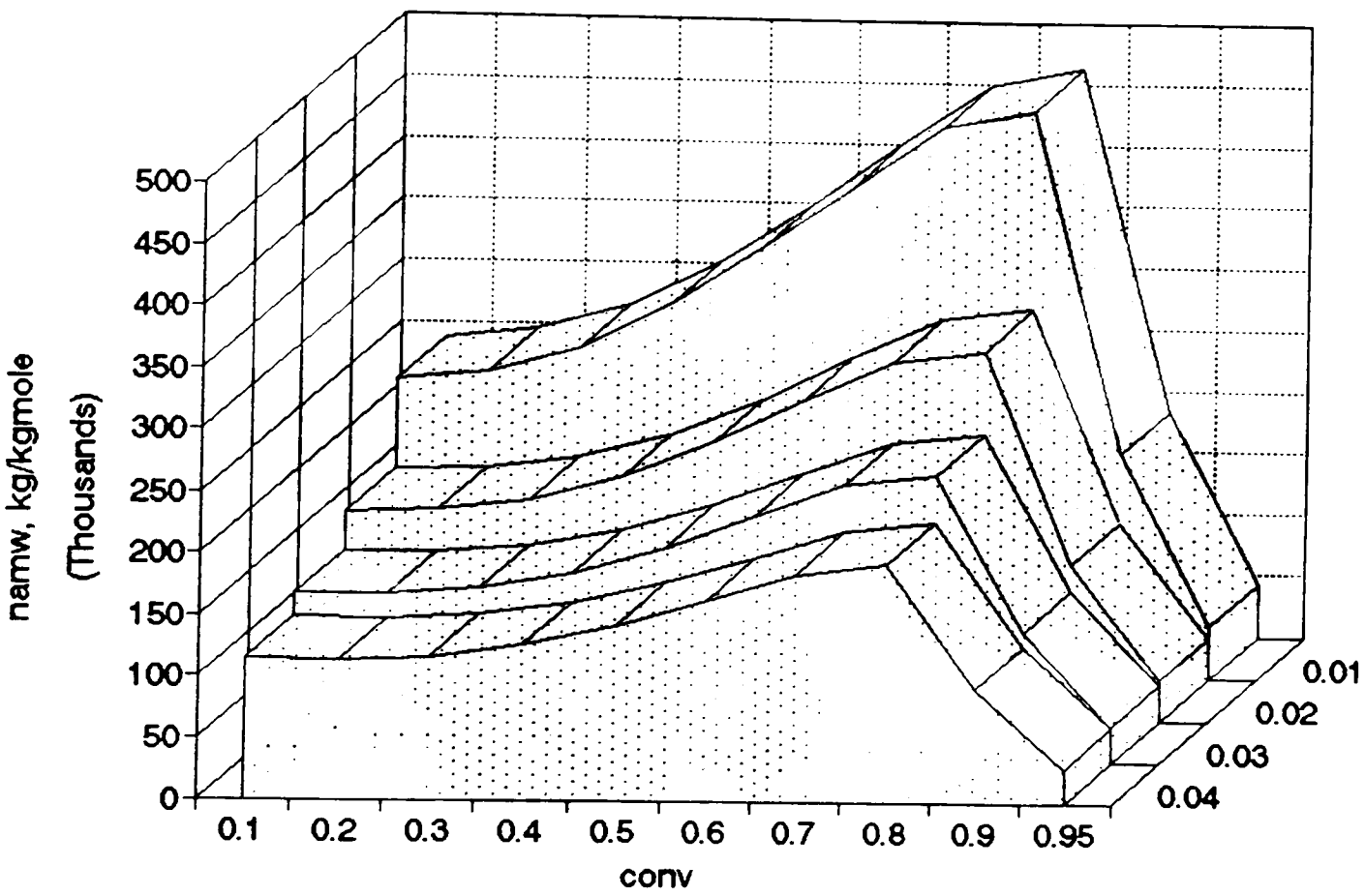


Figure 7.15
Effect of Initiator Loading to NAMW
(T=70 deg C)

add to the growing polymer radicals. Thus, short chain growing radicals prevail in the system just like the case of high temperature operation.

The solvent in the mixture increases the chain transfer and terminates the growing polymer before they grow further and results in lower number-AMW and also slower monomer conversion. The effects of solvent are illustrated in Figures 7.16 and 7.17.

Figures 7.18 to 7.24 suggest an important control feature. If one can kill the initiator when a high conversion, say 0.8, is reached, the MWD is significantly improved even though the conversion rate drops after the initiator is killed. However, without the initiator, the conversion continues to increase by the chain transfer to monomer. The conversion at the end of three hour run is about 94%. This is only about 2.5% less than that of the base case run. Figure 7.19 shows ΣP_n versus time graph. It shows a remarkable comparison with Figure 7.2. After the initiator is killed, ΣP_n rapidly decays to zero. Figure 7.20 indicates almost 16 times increase in number-average MW over the base case at the end of run. Number-PDI (Figure 7.21) at the end of run is decreased 94% over the base case. Number-SD (Figure 7.22) is increased about 450% at the end of run over the base case. Number-SK (Figure 7.23) and Number-KU are only 22% and 5% of the base case at the end of run over the base case.

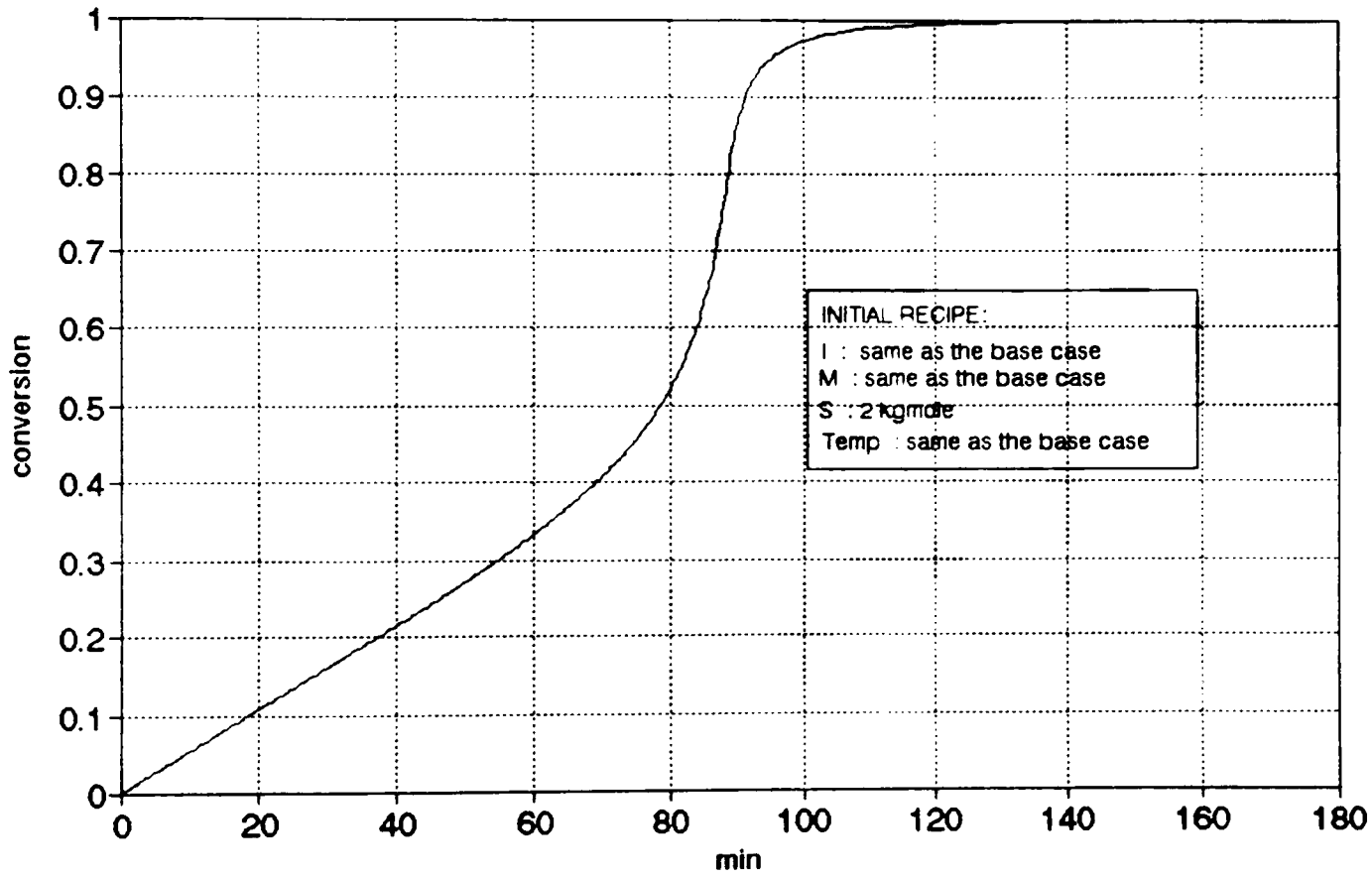


Figure 7.16
Conversion versus Time (with Solvent)

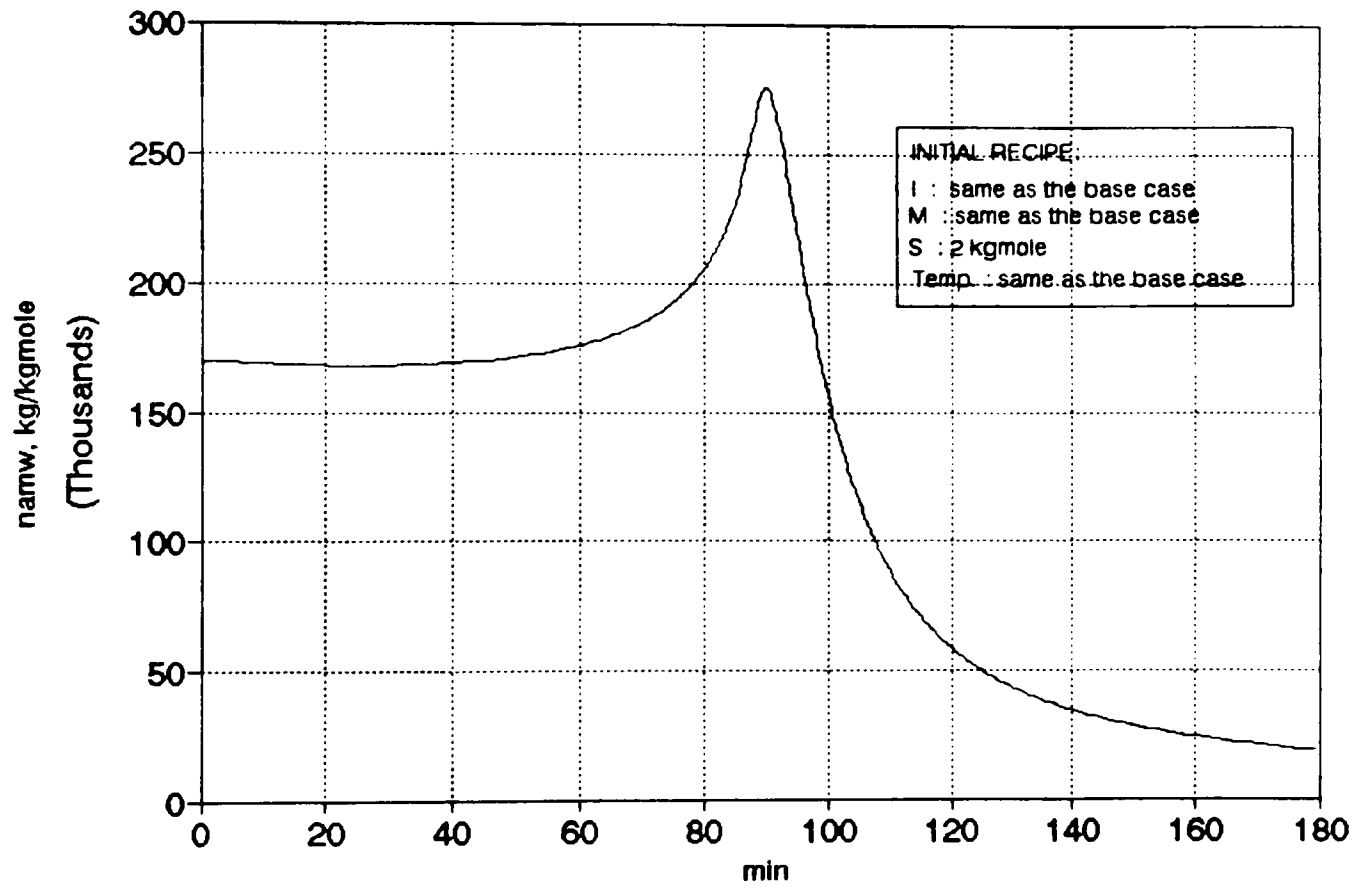


Figure 7.17
Number-AMW versus Time (with Solvent)

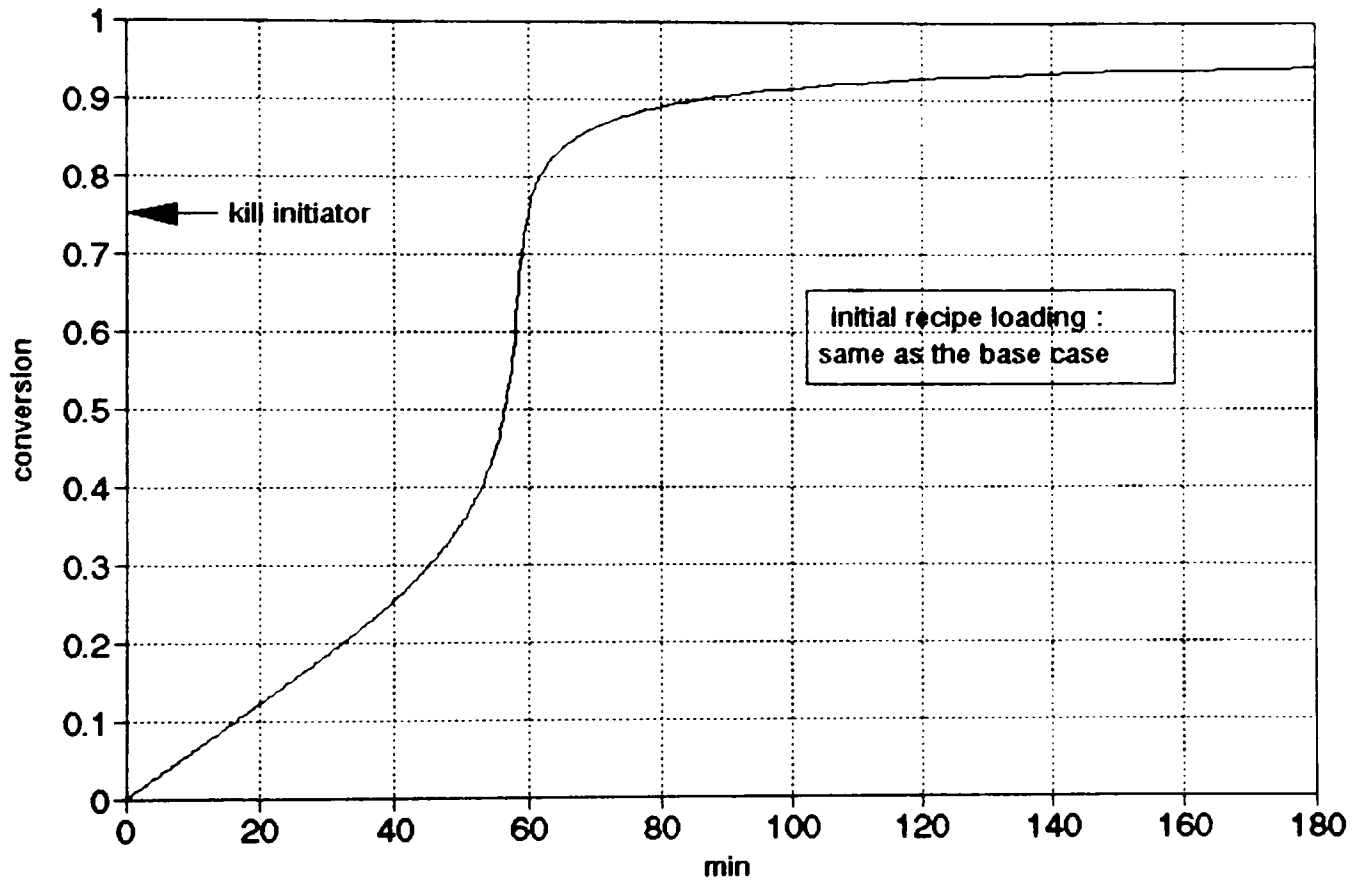


Figure 7.18
Conversion versus Time
(initiator is killed when x > 0.75)

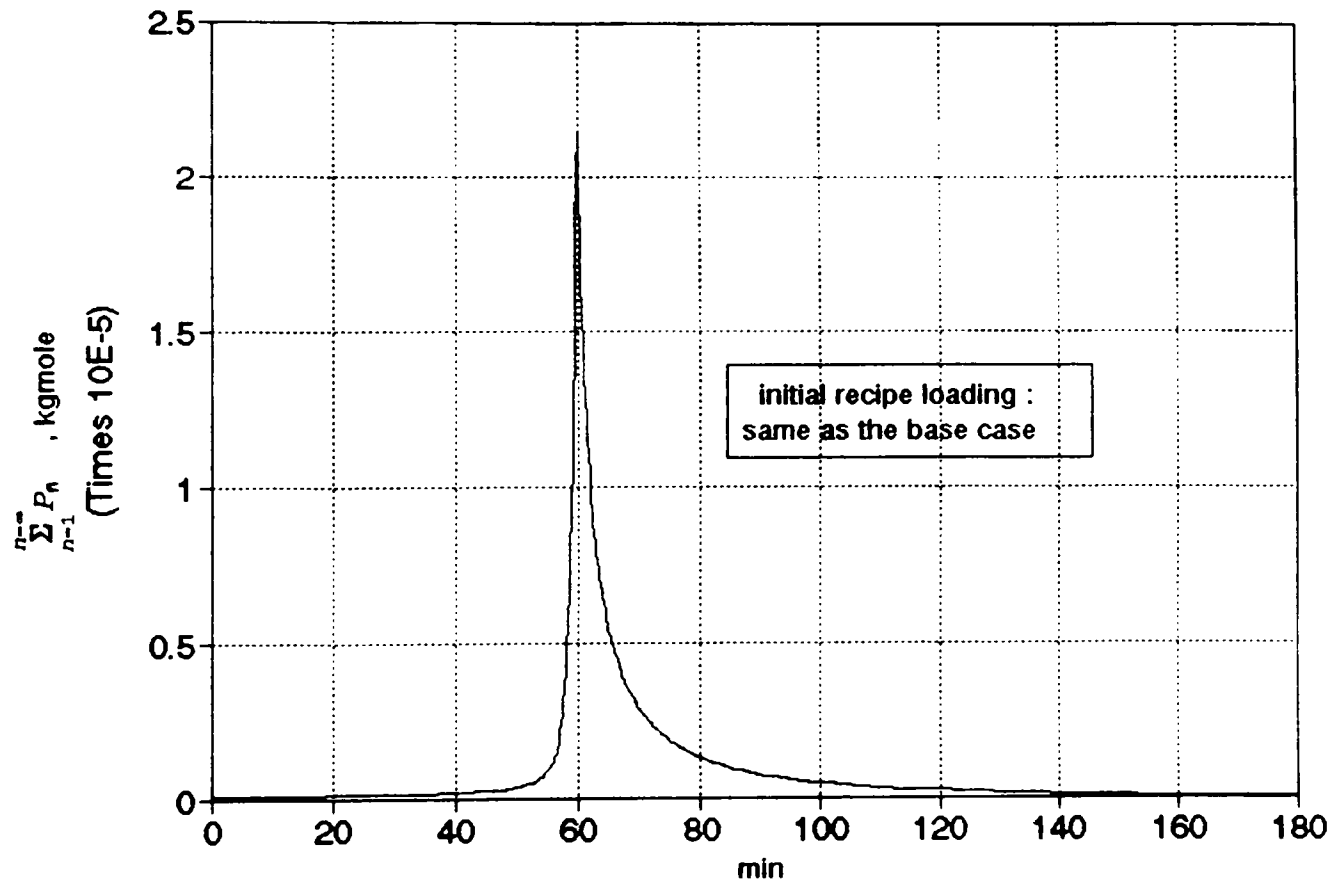


Figure 7.19
ΣP_n (λ_o) versus Time
(initiator is killed when x > 0.75)

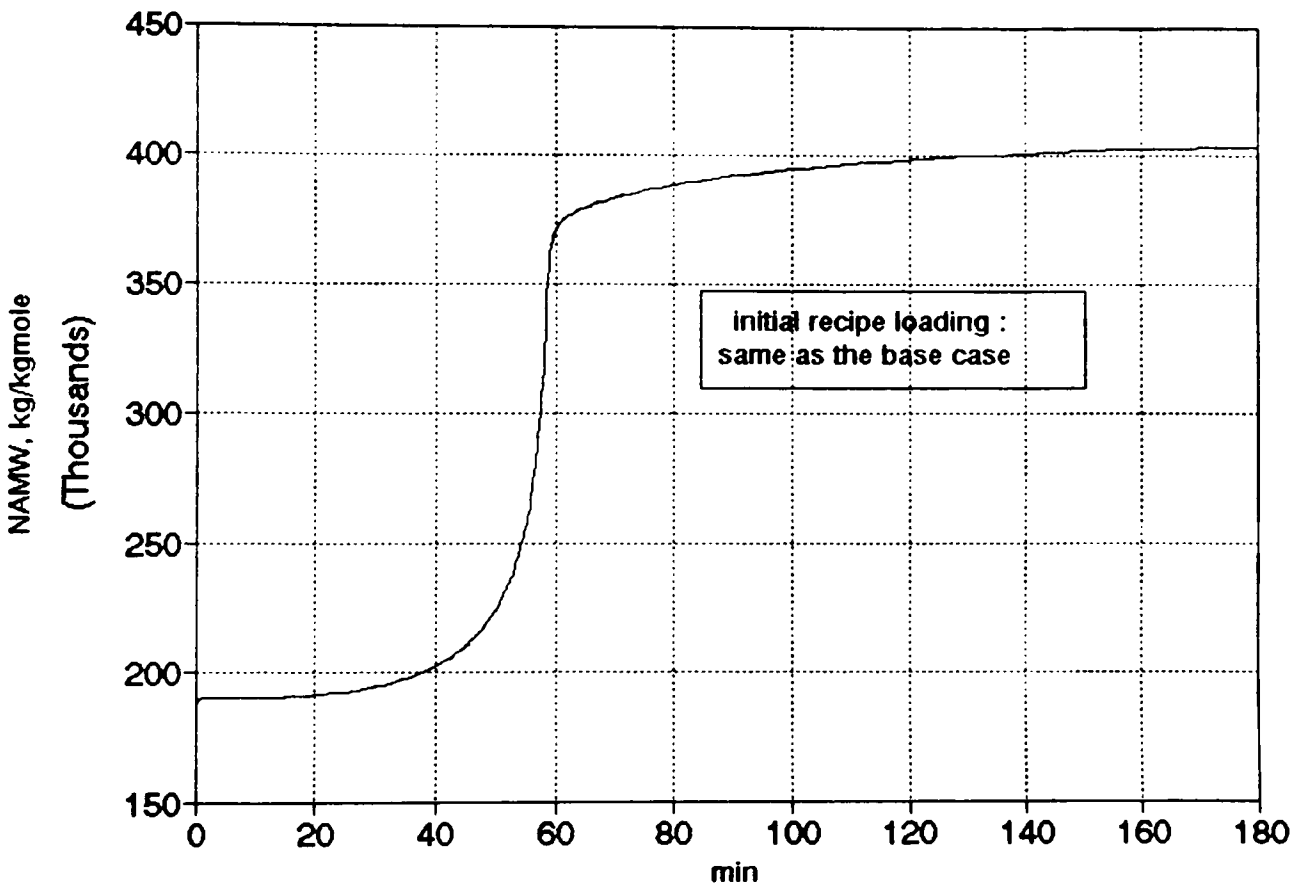


Figure 7.20
Number-AMW versus Time
(initiator is killed when x > 0.75)

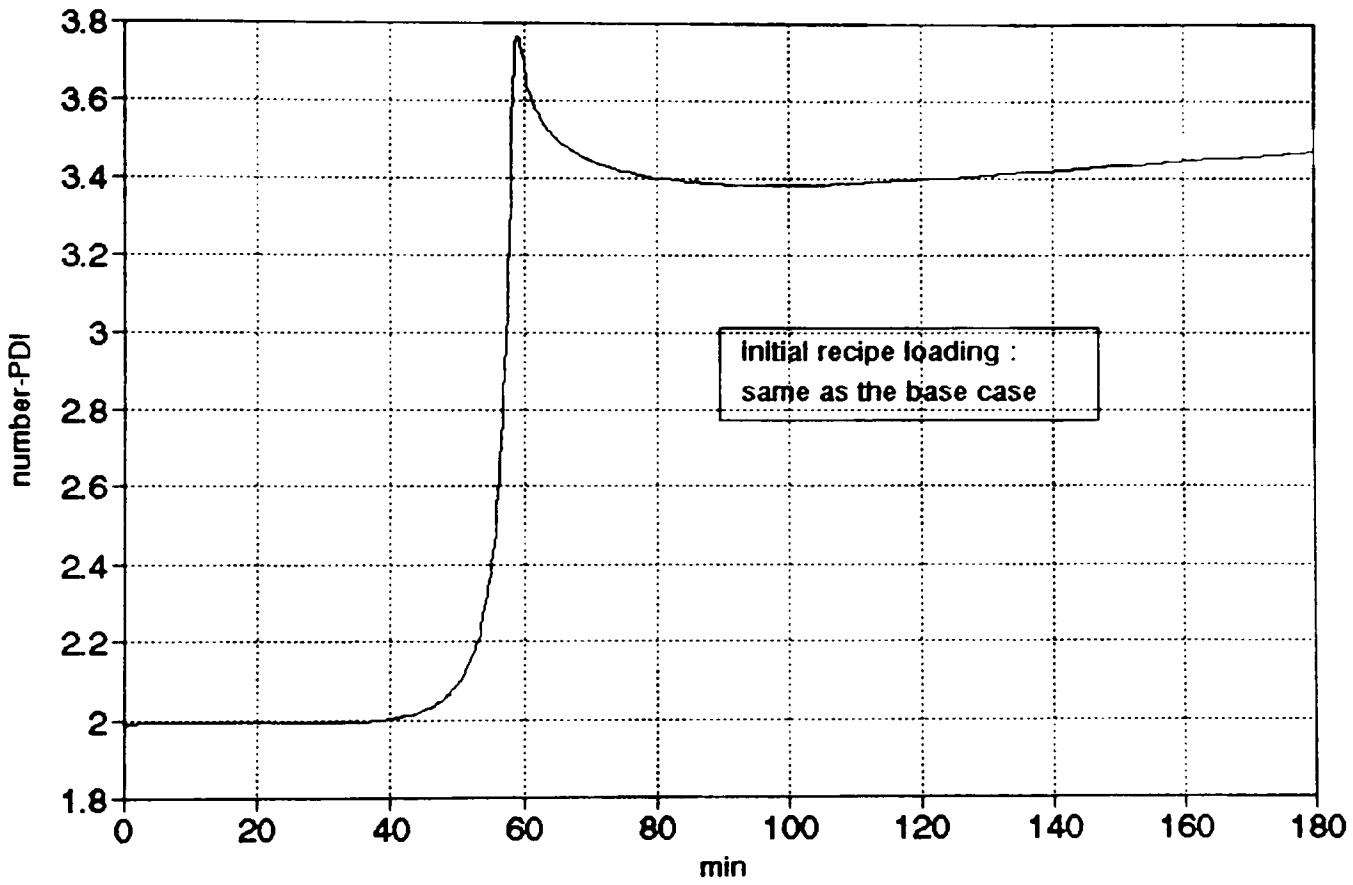


Figure 7.21
Number-PDI versus Time
(initiator is killed when x > 0.75)

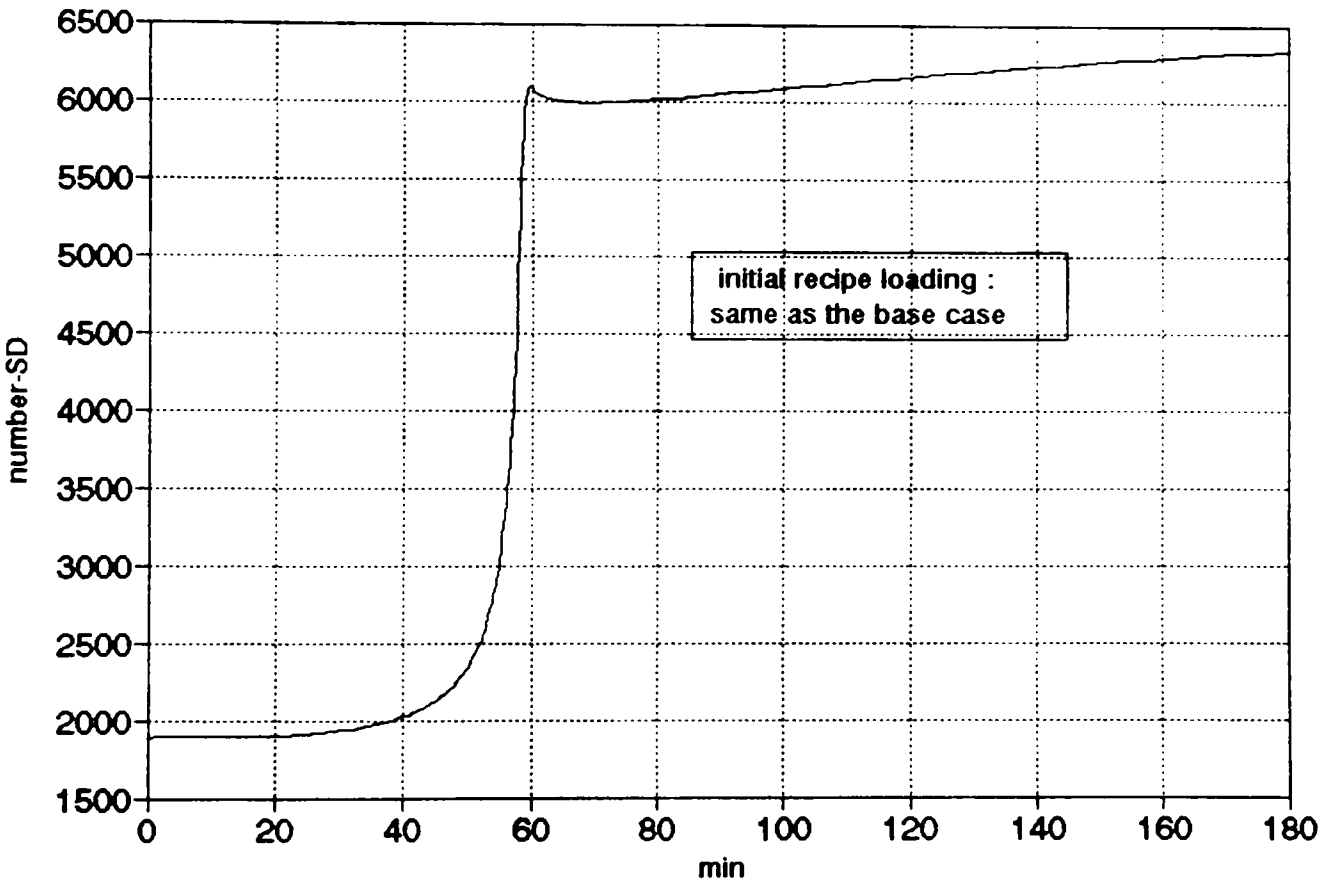


Figure 7.22
Number-SD versus Time
(initiator is killed when x > 0.75)

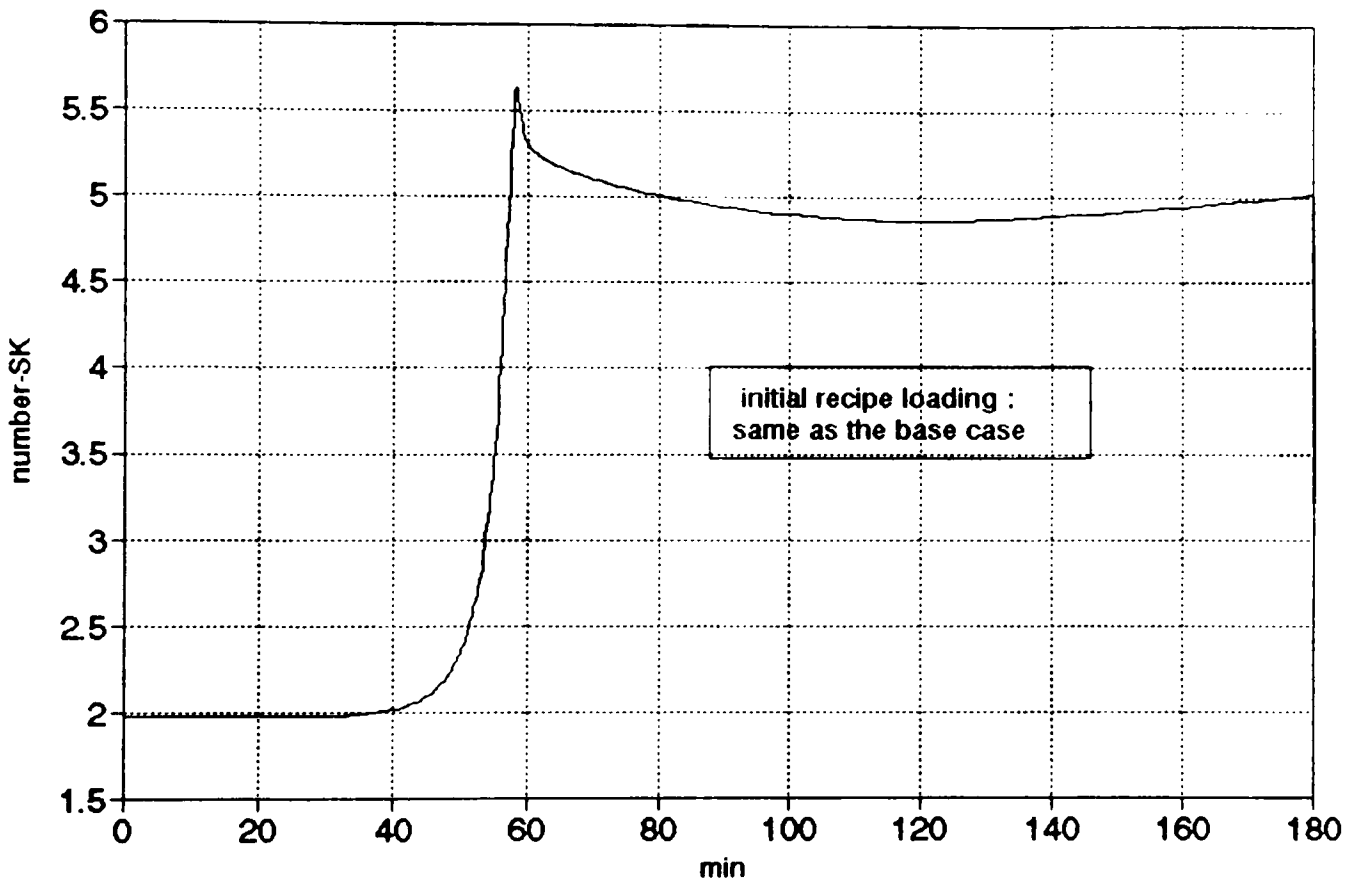


Figure 7.23
Number-SK versus Time
(initiator is killed when x > 0.75)

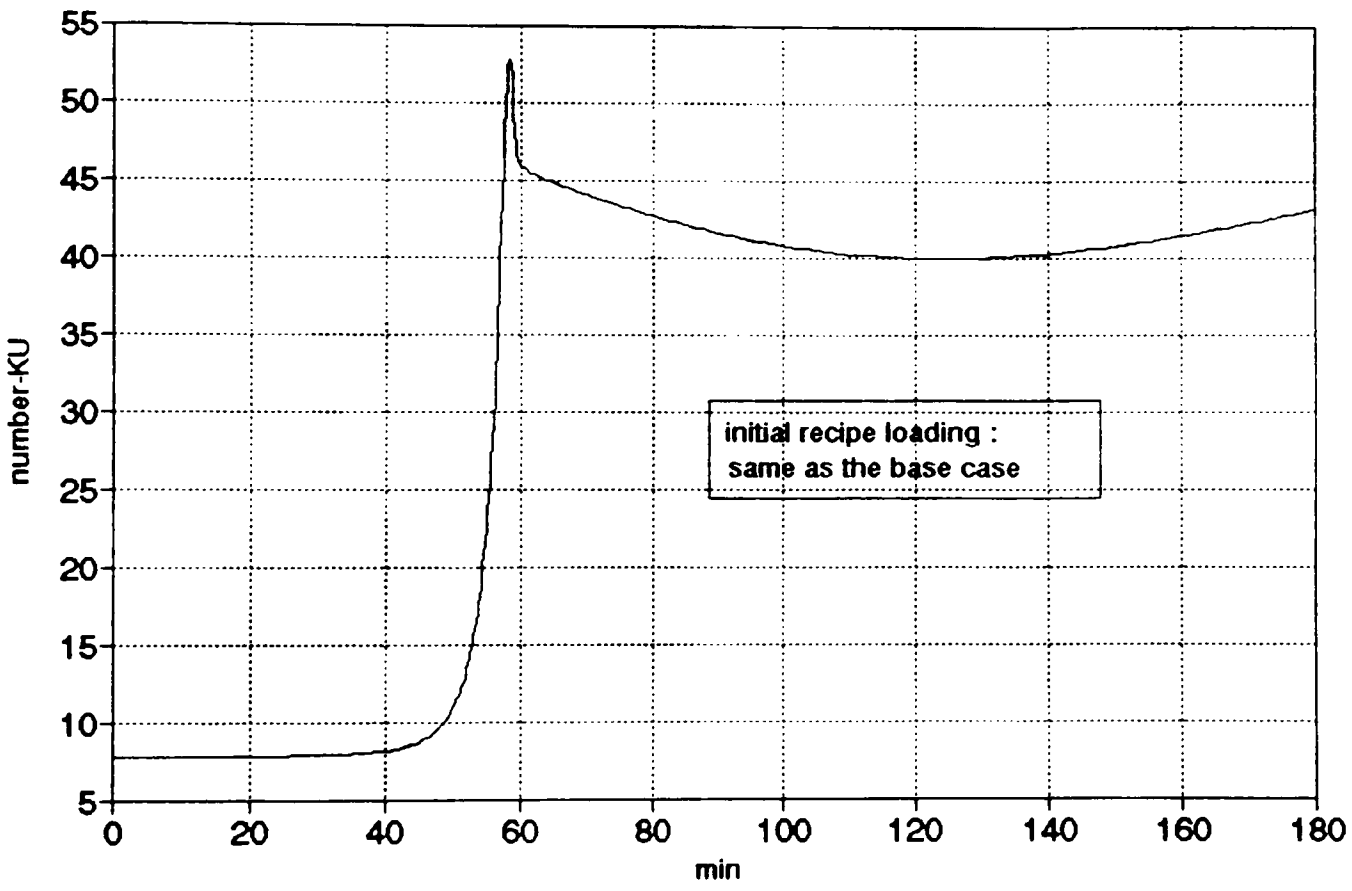


Figure 7.24
Number-KU versus Time
(initiator is killed when x > 0.75)

The timing of killing the initiator is important. If it is too early, ΣP_n becomes so small that lower monomer conversion will be obtained. Conversely, if it is too late, ΣP_n becomes to be dominated by short chains and big enough to remain relatively high amount after killing the initiator. Therefore, the average MW drops very rapidly. It is not recommended to maintain a high temperature, say 80 °C, after killing the initiator because the termination reaction proceeds more rapidly than the chain transfer to monomer and monomer conversion is essentially stopped due to the lack of growing polymers.

Solving the differential equations in Figure 3.2 to get MWD would be extremely tedious because all the species of growing polymer and dead polymer should be calculated individually. It also causes a problem of memory allocation in a PC-DOS environment, because each of thousands of species must take an 8 bytes of double precision variable. The single precision floating point calculation does not provide sufficient accuracy for this case. Due to these reasons, these mechanistical model hardly can be used as a process model for a control study of this polymerization reaction. The calculation result of this rigorous model, however, will give the actual MWD. In comparison, the moment model requires only 6 variables to describe the dynamic behavior of the MWD. Without any question, it has a remarkable advantage over using rigorous model. The problem

with the moment-model-based process simulator is that it cannot predict, for instance, a bimodal type of distribution. This problem could be overcome using the class method.

7.3 Class Method Results and Discussion

The reactor simulator adopted "PM2" in Table 7.1 as a process model. Noise-free process measurements are used to provide an equal basis for any comparison made on the results. The results presented here are obtained while the reactor temperature is under control by NPMBC. The NPMBC adopted "CM1" in Table 7.1 as a controller model. The recipe used is:

	Tr_{sp}(°C)	Initiator supply rate (kg/min)	Solvent supply rate (kg/min)	Monomer supplyrate (kg/min)
Stage I :	70	0.73	60.0	0.0
Stage II :	60	0.36	35.0	0.0
Stage III:	60	0.0	10.0	0.0

The 200 kg of monomer is preloaded, and no monomer is added during the batch time. Initiator and solvent is fed into the reactor for 1 minute immediately after the new stage begins.

The solutions of Eqs.(4.32 to 4.34) provide moles of each class of polymer. By multiplying the mid-chain length of the class number, n, with ${}_nP$ or ${}_nD$ (λ_1 and μ_1 by Eq.(4.35) and Eq.(4.36) respectively), one gets the moles of monomer

converted into n^{th} class of growing or dead polymer. Therefore, the sum of λ_1 and μ_1 at the end of the reaction should be equal to the total moles of monomer entered the reactor.

The differential equations, Eqs.(4.32-4.34), are solved using three different numerical integrators, 4th-order Runge-Kutta, 2nd-order Runge-Kutta and Euler's. As shown by the columns C1, C2 and C3 in Table 7.2, the calculation result does not show any significant difference among three numerical integrators. It suggests that the 4th-order Runge-Kutta is not necessary to perform an accurate calculation. For the step size of 0.05 seconds, the 2nd-order Runge-Kutta and Euler's results are as accurate as the 4th-order Runge-Kutta method. It took about 20 hours of computer time to get 97% conversion when the 2nd-order Runge-Kutta method was used to integrate 150 differential equations of growing polymer classes and 300 differential equations of dead polymer classes with a 486 33 MHz PC using a 0.05 second integration time step.

As indicated by the values of yielded distribution parameters and $\lambda_1+\mu_1$ (amount of monomer converted to polymer) of columns C3, C4 and C5 in Table 7.2, the class method result gets closer to the true value which is obtained by the method of moments with the increase of the number of classes. That is an evidence that the class method can effectively generate the chain length distribution which

Table 7.2
The Influence of Choice of Numerical Integrator, Its Stepsize and Number of Classes to the Class Method Performance

		M	C1	C2	C3	C4	C5
X=0.1	Xn	1638.6	2161.3	2161.3	2161.3	1923.0	1802.6
	Xw	3823.2	4546.9	4546.9	4546.9	4158.9	3957.3
	nPDI	2.271	2.104	2.104	2.104	2.163	2.195
	nSK	5.555	3.190	3.190	3.190	3.537	3.771
	nKU	146.78	30.331	30.331	30.331	39.303	45.67
	$\mu_1+\lambda_1$	0.2	0.2613	0.2613	0.2613	0.229	0.2149
X=0.3	Xn	2038.4	2517.0	2517.0	2517.0	2278.3	2158.1
	Xw	4361.3	5176.3	5176.3	5176.3	4766.5	4554.1
	nPDI	2.14	2.057	2.057	2.057	2.092	2.110
	nSK	3.067	2.426	2.426	2.426	2.516	2.581
	nKU	40.492	13.881	13.881	13.881	15.724	17.040
	$\mu_1+\lambda_1$	0.6	0.7458	0.7458	0.7458	0.6746	0.639
X=0.5	Xn	2512.3	2988.9	2988.9	2988.9	2751.7	2631.8
	Xw	5753.3	6513.7	6513.7	6513.7	6133.3	5937.0
	nPDI	2.290	2.179	2.179	2.179	2.229	2.256
	nSK	2.919	2.606	2.606	2.606	2.685	2.734
	nKU	20.412	13.144	13.144	13.144	13.952	14.465
	$\mu_1+\lambda_1$	1.0	1.1907	1.1907	1.1907	1.0955	1.0477
X=0.7	Xn	2966.0	3438.4	3438.4	3438.4	3203.9	3084.7
	Xw	7194.9	7886.5	7886.5	7886.5	7541.2	7362.0
	nPDI	2.426	2.294	2.294	2.294	2.354	2.387
	nSK	2.992	2.765	2.765	2.765	2.840	2.884
	nKU	17.052	13.684	13.684	13.684	14.302	14.663
	$\mu_1+\lambda_1$	1.4	1.6261	1.6261	1.6261	1.5146	1.4553
X=0.9	Xn	3133.7	3600.9	3600.9	3600.9	3368.2	3250.7
	Xw	7423.9	8133.5	8133.5	8133.5	7778.2	7596.6
	nPDI	2.369	2.259	2.259	2.259	2.309	2.337
	nSK	2.824	2.648	2.648	2.648	2.707	2.741
	nKU	15.022	12.563	12.563	12.563	13.019	13.282
	$\mu_1+\lambda_1$	1.8	2.0696	2.0696	2.0696	1.9361	1.8686
X=0.95	Xn	3028.1	3486.6	3486.6	3486.6	3260.6	3144.2
	Xw	7423.2	7936.2	7936.2	7936.2	7593.5	7414.0
	nPDI	2.392	2.276	2.276	2.276	2.329	2.358
	nSK	2.887	2.703	2.703	2.703	2.764	2.800
	nKU	15.664	13.072	13.072	13.072	13.56	13.856
	$\mu_1+\lambda_1$	1.9	2.1948	2.1948	2.1948	2.0458	1.9728

Table 7.2 (Continued)

where

		numerical integrator	step size	number of classes	number of species/class
M	gpm	4-th order RK	0.005	-	-
	dpm	4-th order RK	0.005	-	-
C1	gpc	4-th order RK	0.05	75	800
	dpc	4-th order RK	0.05	150	800
C2	gpc	2-nd order RK	0.05	75	800
	dpc	2-nd order RK	0.05	150	800
C3	gpc	euler	0.05	75	800
	dpc	euler	0.05	150	800
C4	gpc	2-nd order RK	0.02	150	400
	dpc	2-nd order RK	0.02	300	400
C5	gpc	2-nd order RK	0.02	300	200
	dpc	2-nd order RK	0.02	600	200

where gpm = moment pf growing polymer
dpm = moment of dead polymer
gpc = class of growing polymer
dpc = class of dead polymer
RK = Runge-Kutta numerical integrator
Xn = number average chain length
Xw = weight average chain length
PDIn= number polydispersity index
SKn = number skewness
KUn = number kurtosis
$\mu_1+\lambda_1$ = moles of monomer converted to polymer

mimics the true one. The results in Table 7.2 also indicate that the maximum chain length of 60,000 for the growing polymer was sufficiently high. If it was assumed smaller than a minimum necessary chain-length, it would not have improved the accuracy as the number of classes increased, keeping the same class interval. However, one should run the class method with lower maximum chain-length to see whether the maximum chain-length is chosen unnecessarily high. If the accuracy is not lost with the lower maximum chain-length, the current maximum chain-length is chosen unnecessarily high.

The dynamic development of the normalized weight chain length distribution which is equivalent to Figure 2.2 is obtained using the condition C5 in Table 7.2 at every 5% of conversion increment and is plotted in 3-dimension as shown in Figure 7.25. As we wanted, we can see how the CLD is forming as the conversion proceeds. The class method is also tested with isothermal operation case and without solvent. The same amount of monomer and initiator were preloaded as for the Case C5. The temperature at each stage was 70°C, 75°C and 80°C, respectively. The result is shown in Figure 7.26. Since no solvent is loaded into the system, the termination step is severely diffusion controlled and low chain-length polymer dominates the system at the end of run. The equivalent results were observed by the number-AMW and number-PDI in Section 7.2.

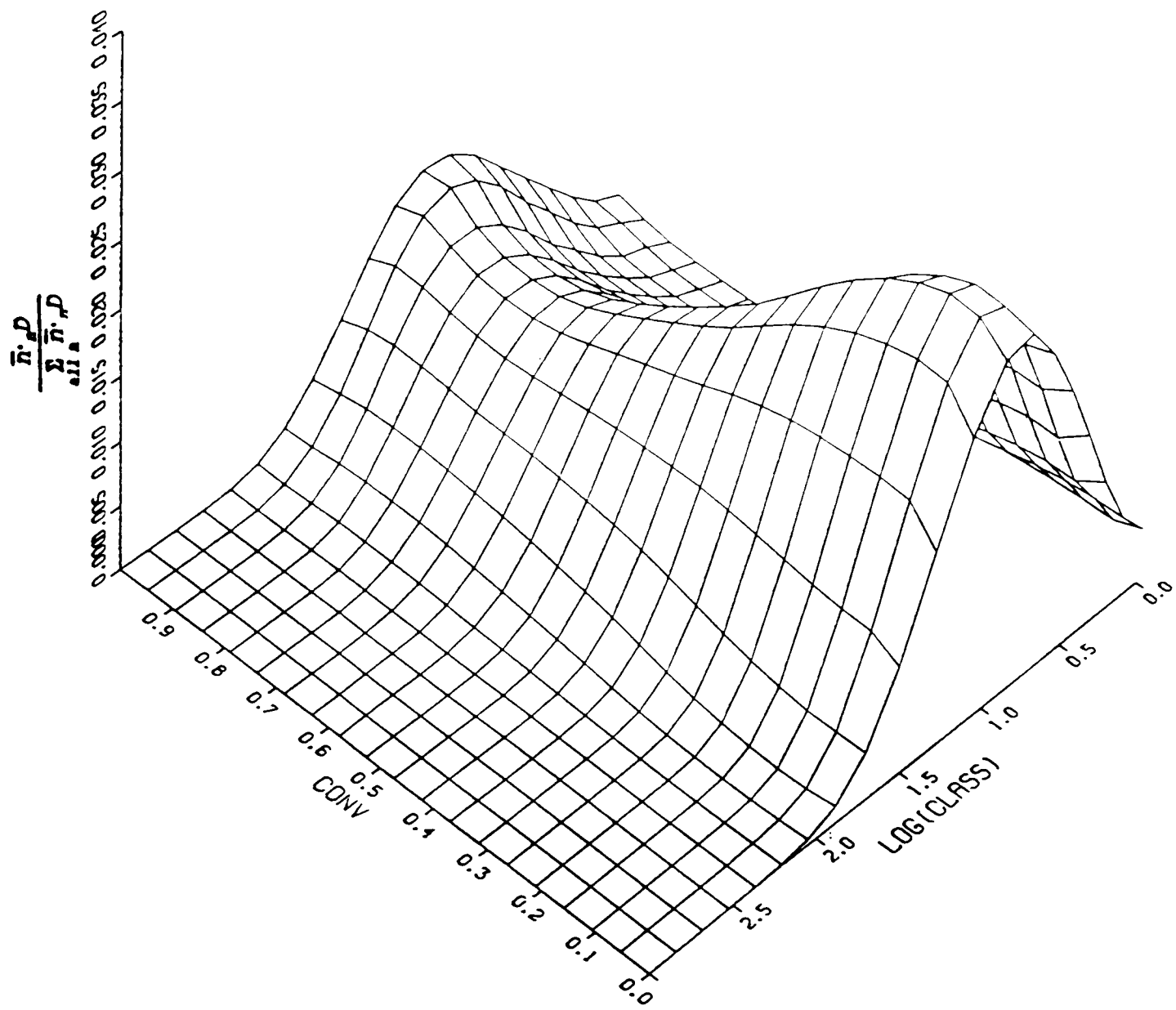


Figure 7.25
Weight Chain Length Distribution versus Conversion
(nonisothermal condition)

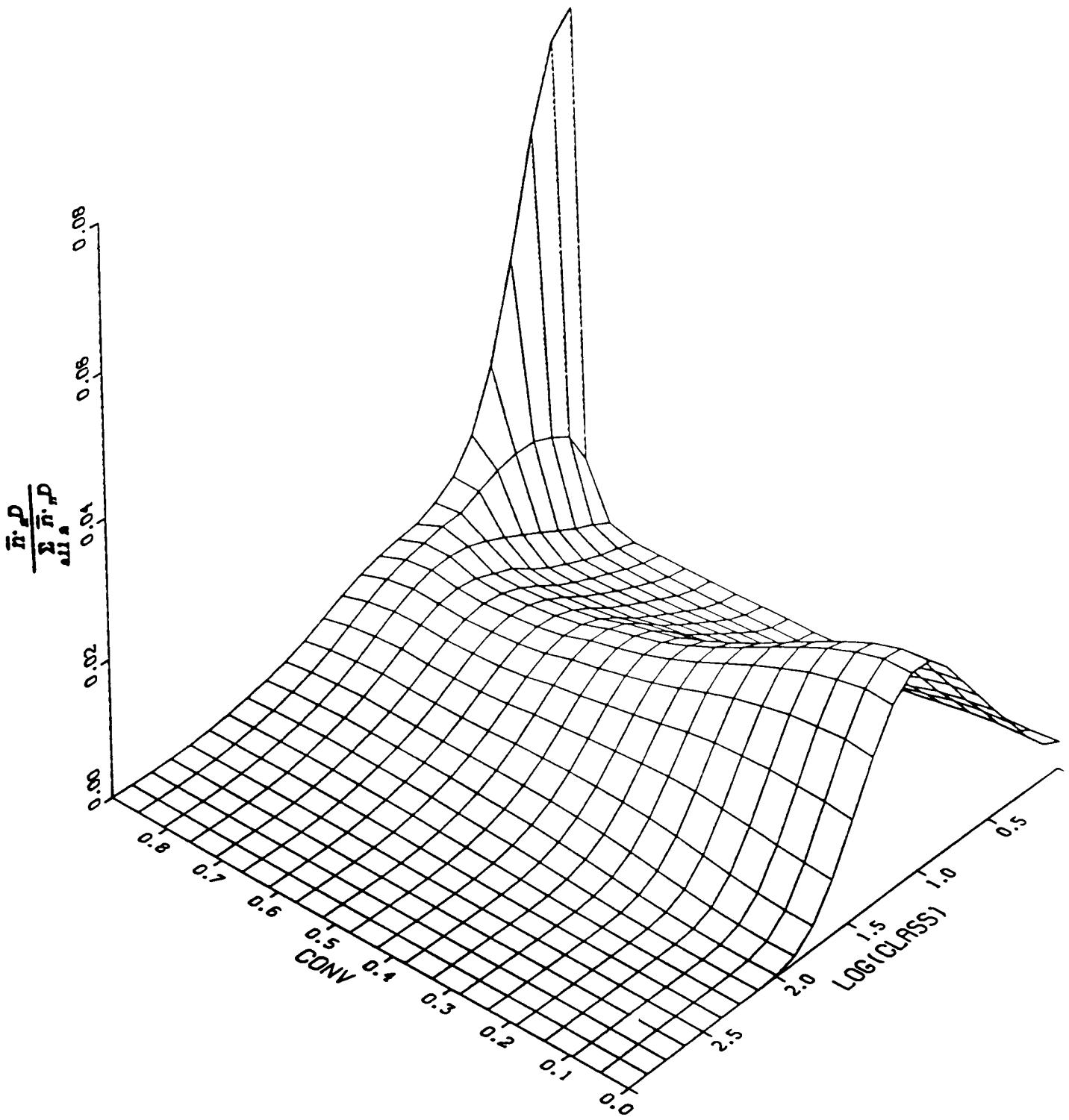


Figure 7.26
Weight Chain Length Distribution versus Conversion
(isothermal condition)

7.4 Temperature Control Results and Discussion

The temperature control of the process described in Figures 3.1 to 3.6 is demonstrated for the condition listed below:

	Tr_{sp}(°C)	Initiator supply rate (kg/min)	Solvent supply rate (kg/min)	Monomer supply rate (kg/min)
Stage I :	70	20.0	1700.0	0.0
Stage II :	60	7.0	750.0	0.0
Stage III:	60	0.0	300.0	0.0

The monomer 5000 kg is preloaded, and no monomer is added during the batch time. The initiator is fed into the reactor for 1 minute and solvent is fed into the reactor for 2 minutes respectively immediately after the new stage begins.

Controller conditions are as follows:

sampling interval = 10 sec.

reparameterization interval = 120 sec.

control action interval = 10 sec.

tuning constant, k1 = 0.0035

tuning constant, k2 = 0.0005

The reactor simulator adopted "PM5" in Table 7.1 as a process model. The NPMBC adopted "CM4" in Table 7.1 as a controller model. The measured monomer conversion is shown in Figure 7.27. The wall viscosity which directly affects

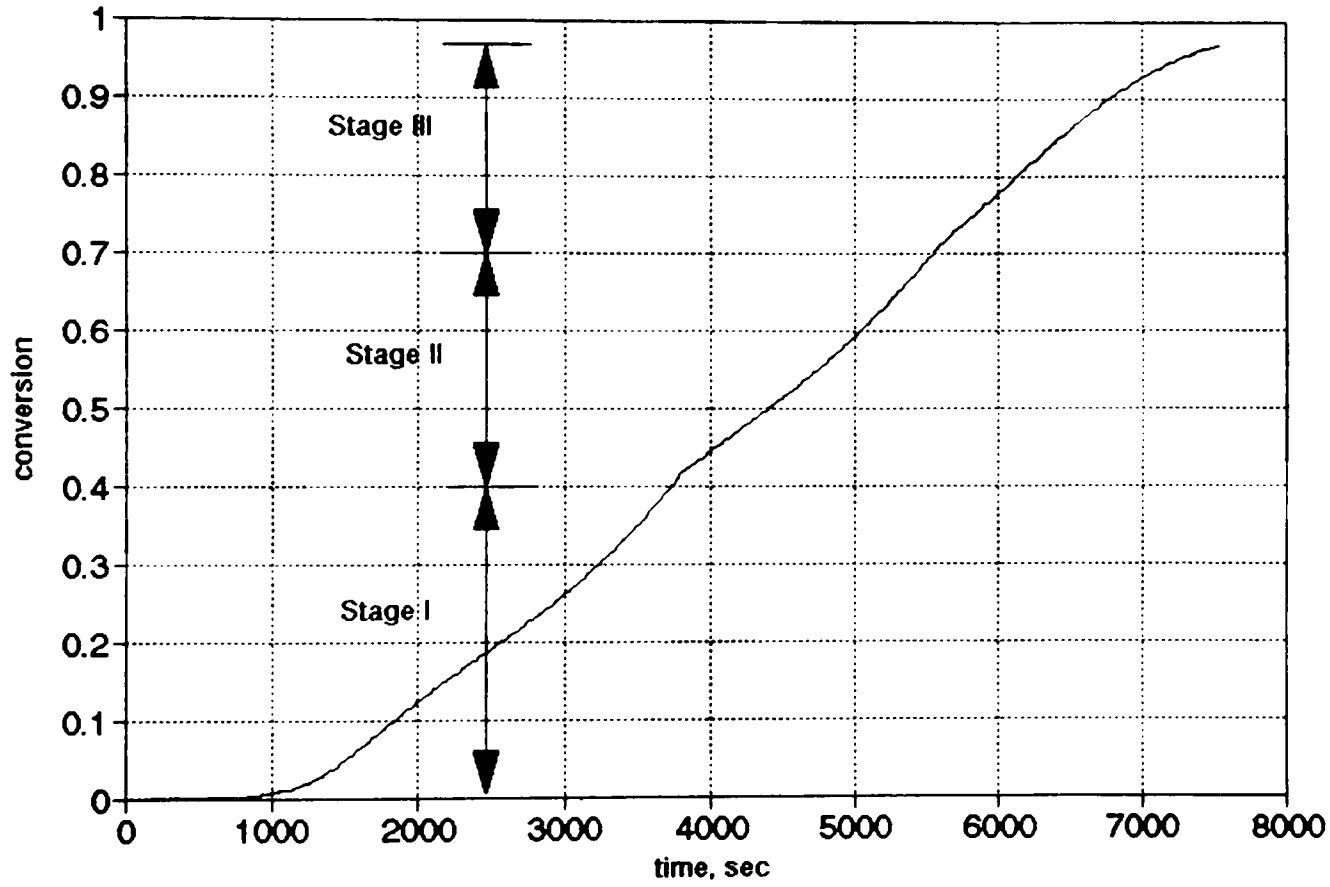


Figure 7.27
Monomer Conversion versus Time

the reactor side heat transfer coefficient is shown in Figure 7.28. Small humps at time periods of 3700 and 5500 sec are caused by the solvent addition to the reactor. The high wall viscosity at high conversion hurts the temperature controllability because heat transfer coefficient decreases due to the high viscosity. The wall viscosity drop after time period 7000 is caused by the jacket temperature rise. The model parameters, λ_o, U and EH_{loss} are shown in Figures 7.29, 7.30 and 7.31 along with corresponding process values. If there is no model mismatches between the process model and the control model, the model parameters are identical to the process values. For model reparameterization, the three on-line measured data are smoothed using a first-order filter. The filter weighting factors are 70% to currently measured value and 30% to previously filtered value.

Using the updated model parameters, control action calculates the required jacket temperature target using Eq.(5.34) as shown in Figure 7.32 which brings the reactor temperature to the target. Then the target can be used to calculate the required steam flow rate or chilled water flow rate using Eq.(5.36) or Eq.(5.37) as shown Figures 7.33 and 7.34 respectively. The excursion on the jacket temperature target at time period 200 in Figure 7.32 is caused by a low starting reactor temperature. Note that this initial heat-up is also done by NPMBC, not by manually. Similar excursion at time period 3700 and 5600 is caused by cold solvent entering

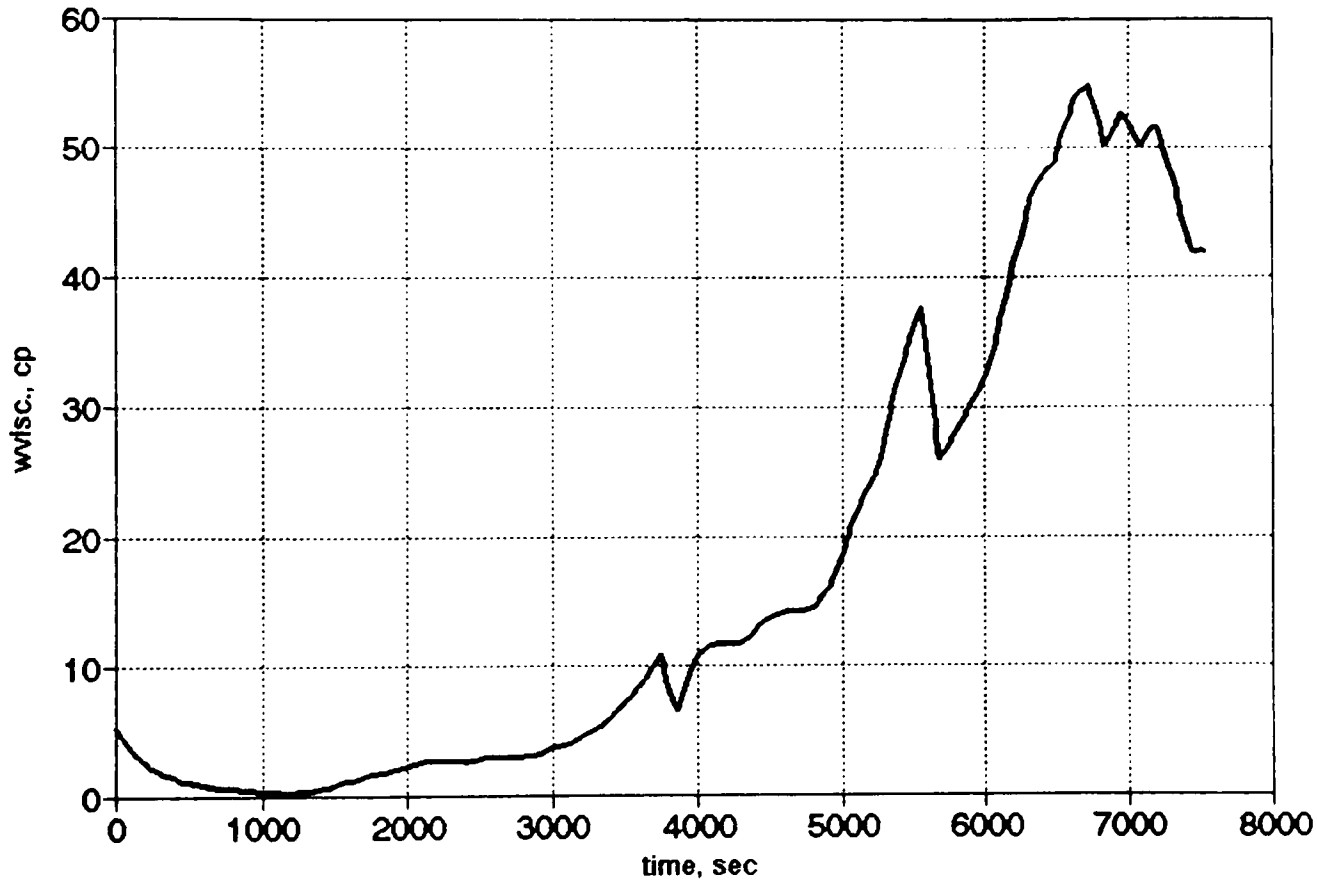


Figure 7.28
Wall Viscosity versus Time

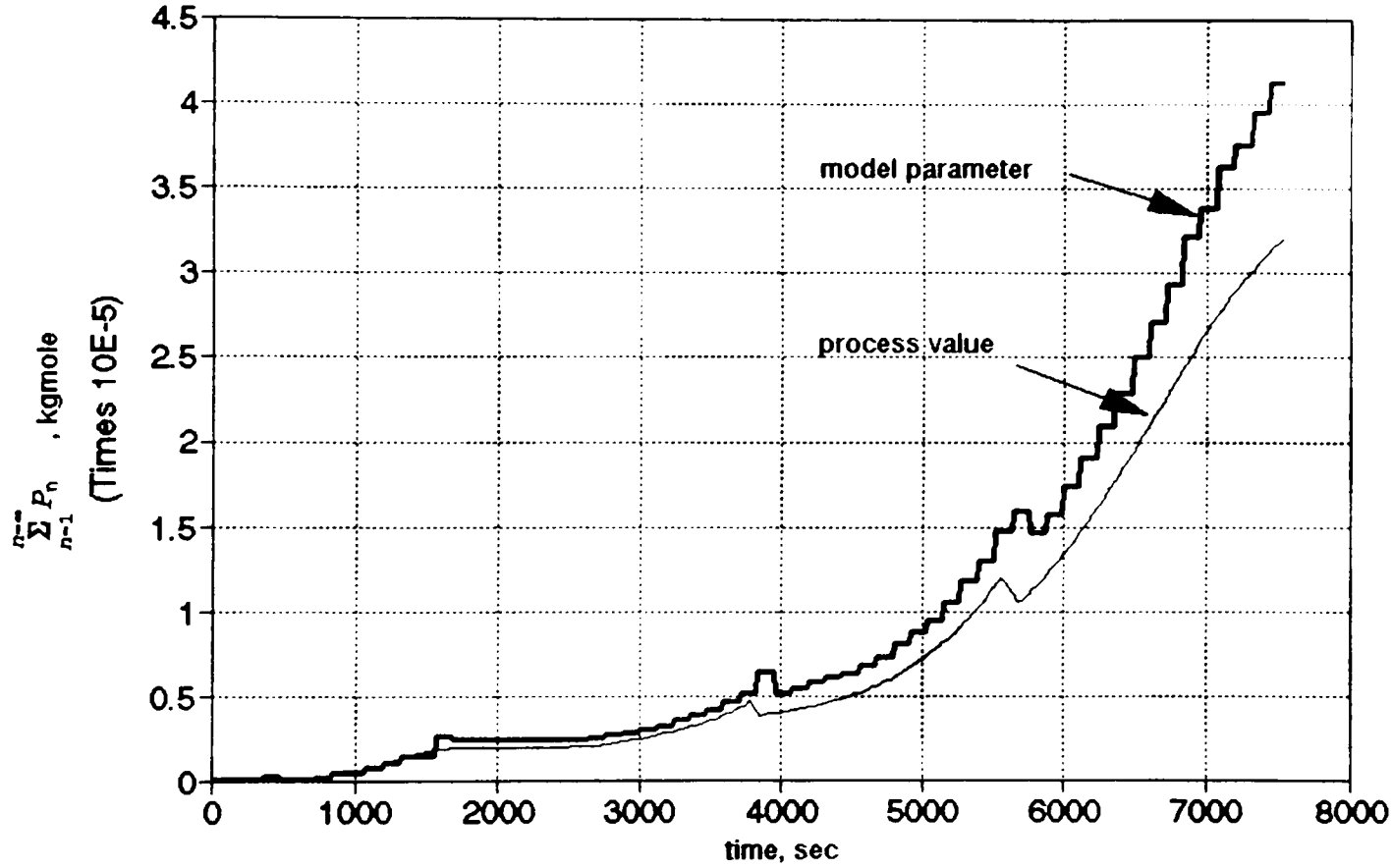


Figure 7.29
ΣP_n (λ_o) versus Time

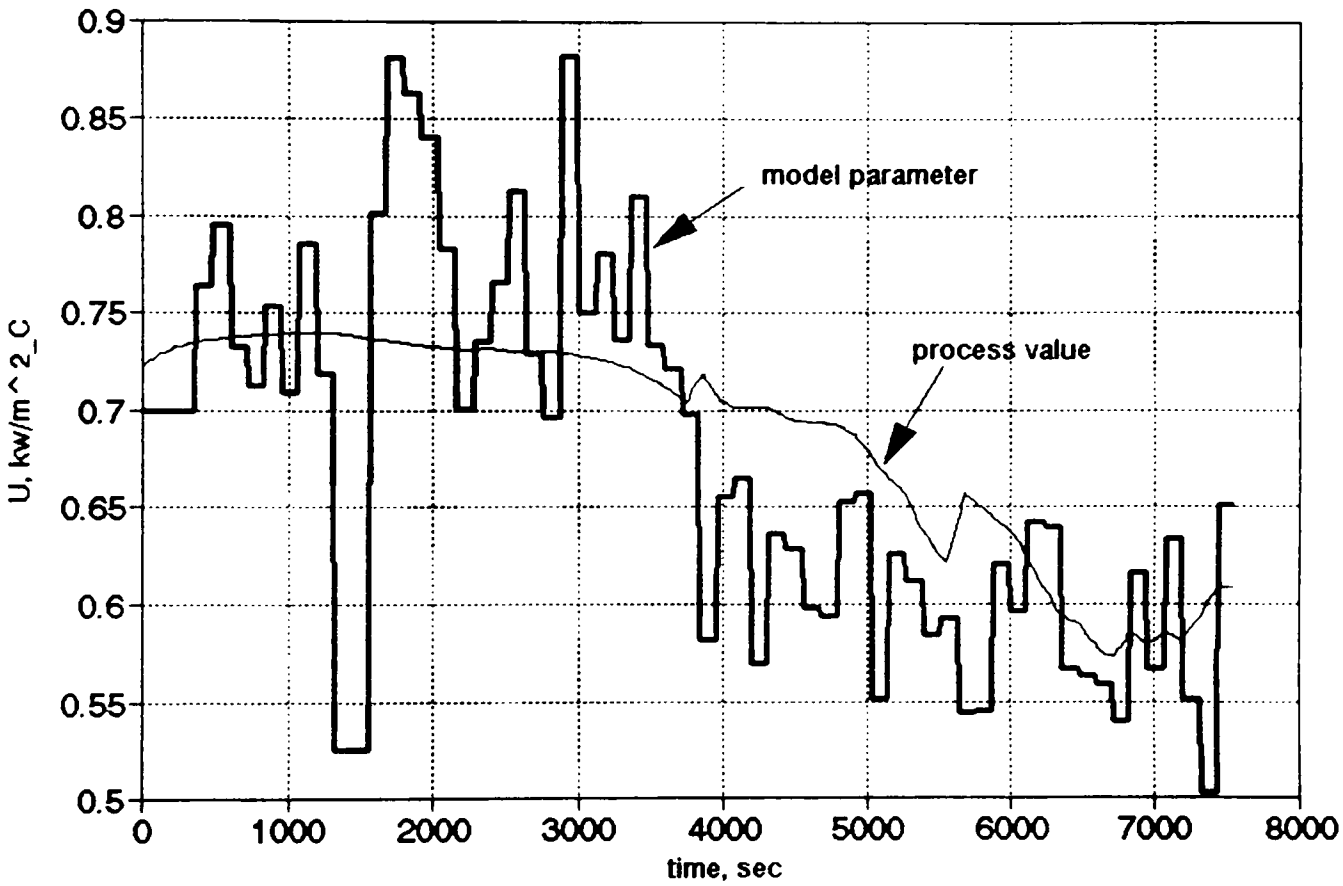


Figure 7.30
Overall Heat Transfer Coefficient versus Time

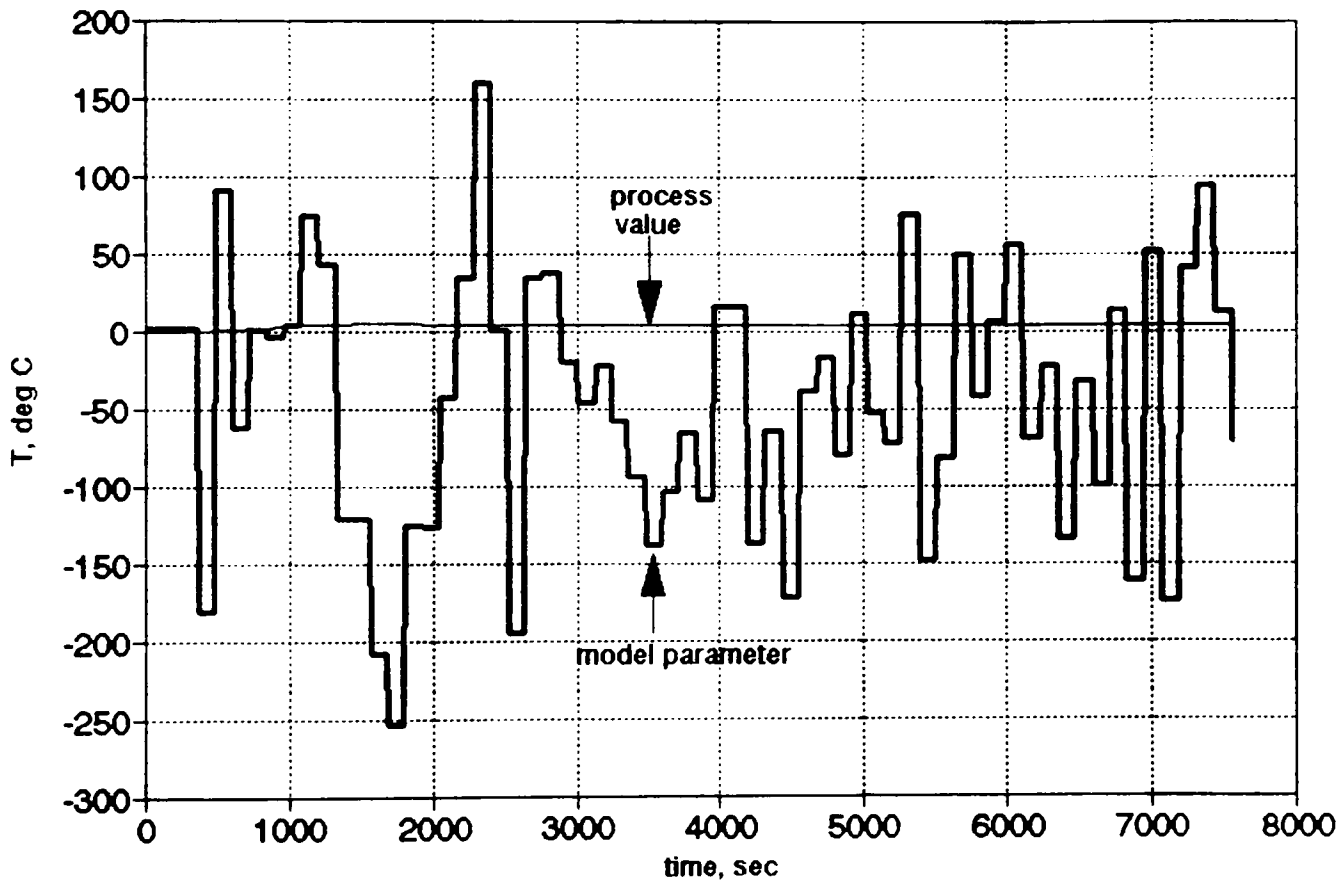


Figure 7.31
Environmental Heat Loss versus Time

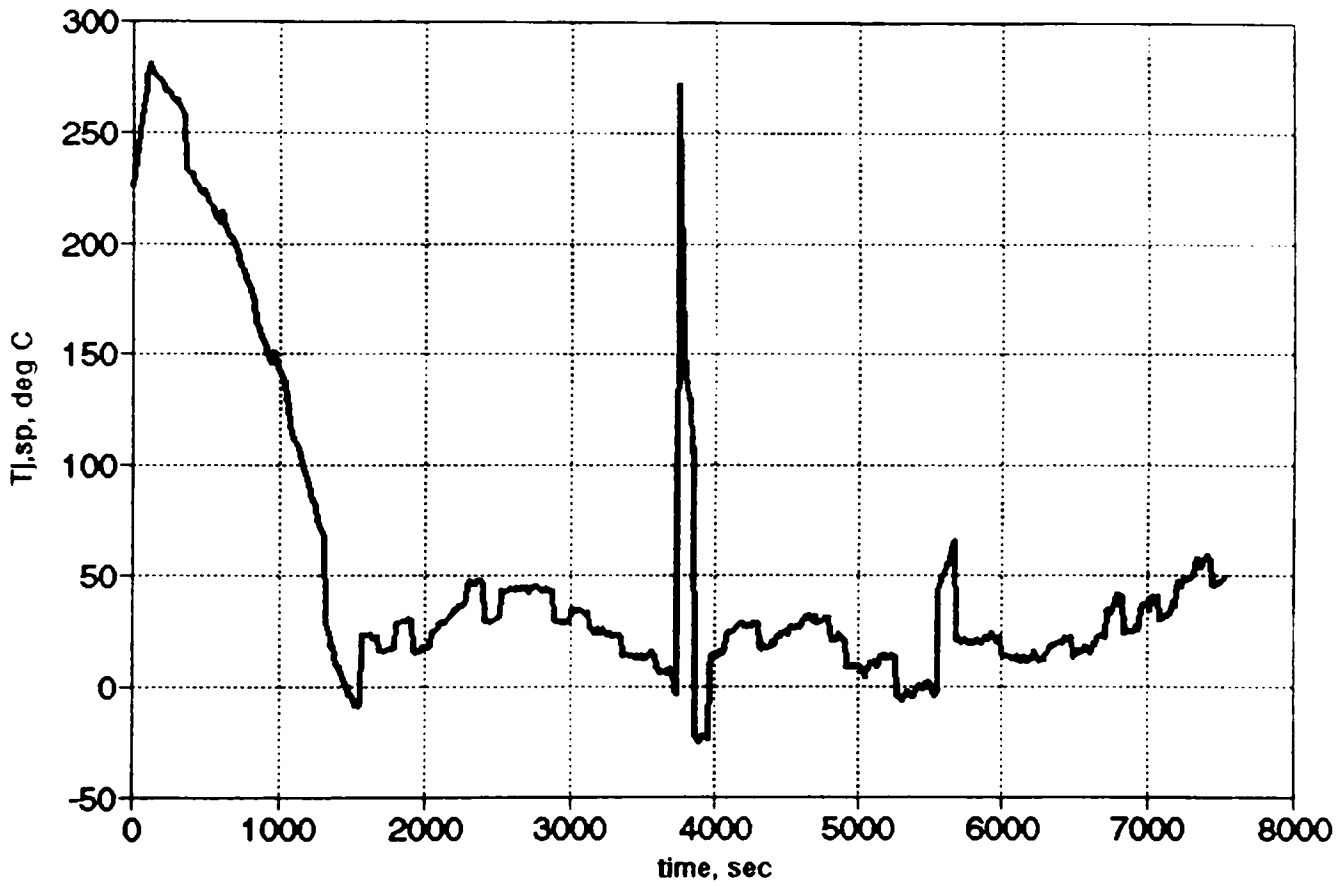


Figure 7.32
Control Action Calculation:
Jacket Temperature Setpoint versus Time

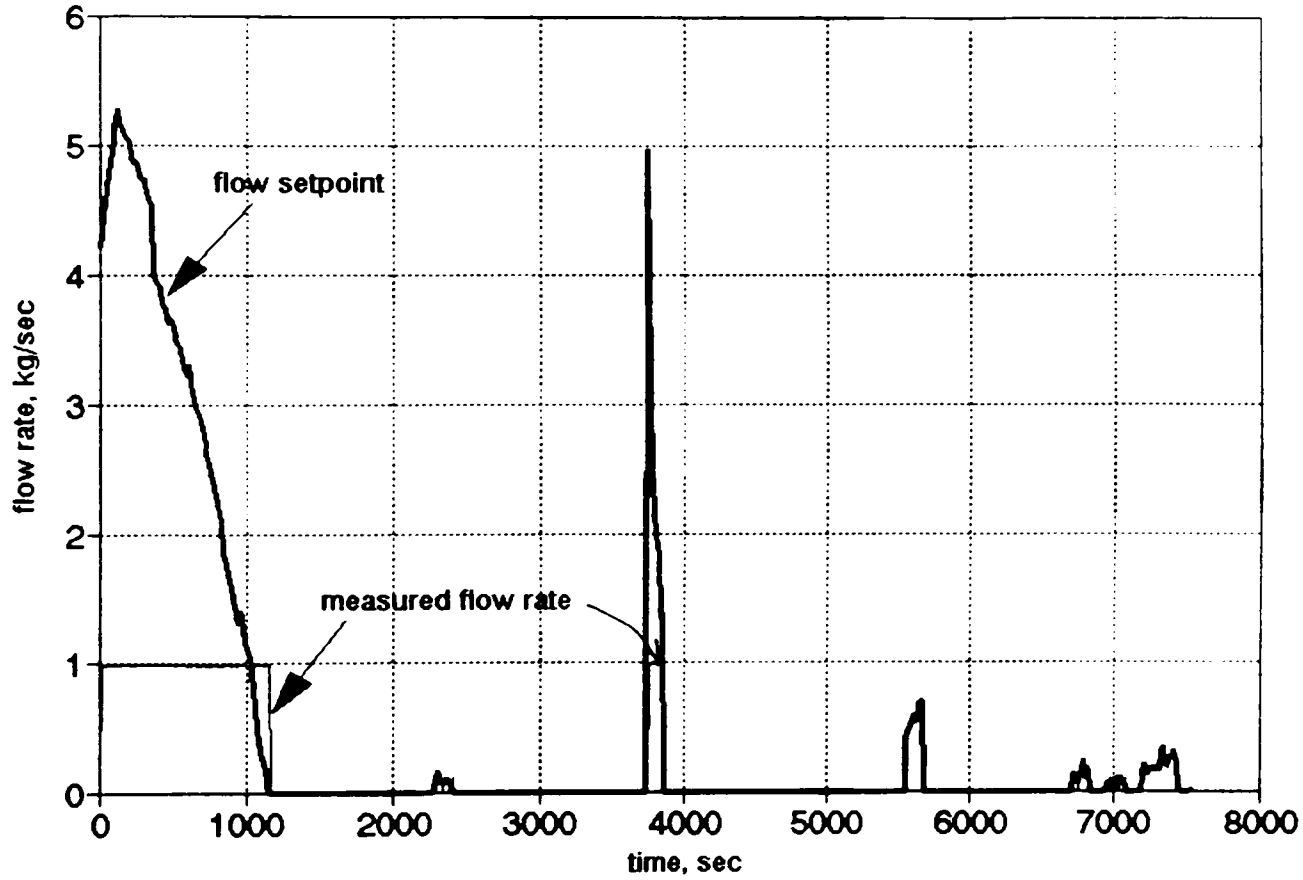


Figure 7.33
Control Action Calculation:
Required and Measured Steam Flow Rate versus Time

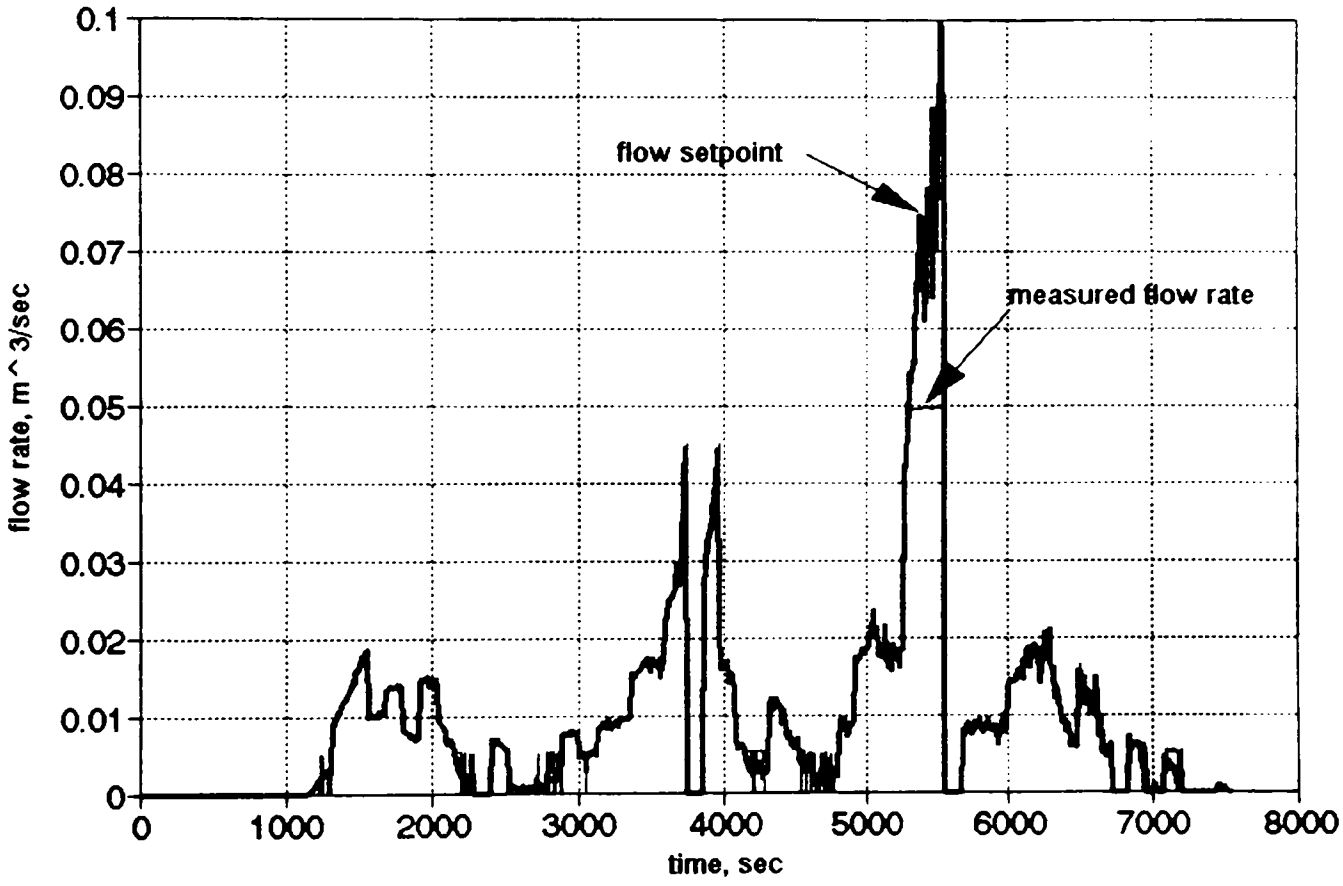


Figure 7.34
Control Action Calculation:
Required and Measured Chilled Water Flow Rate versus Time

the reactor. The excursion on steam flow rate shown in Figure 7.33 reflect these sequences. As shown in Figure 7.34, although the control action hits the constraint on the chilled water supply at time period 5300 due to the selected valve size and the availability of a pressure drop across the valve, and causes a sustained offset on the reaction temperature, the control action immediately returns to normal when stage III starts and more solvent is introduced into the reactor. There is no integral wind-up during the sustained offset period in this version of a NPMBC control, although an integral is used as a temporary means of removing the steady-state offset. It does not "wind up," because every time the model parameters are updated, the integral value is reset to zero.

The reactor and jacket temperature are shown in Figure 7.35. It took about 23 minutes to bring the reactor temperature to the setpoint after start-up. The sluggish response indicates a large process time constant which was caused by the large system volume. The reactor temperature is controlled within ±3 °C of the setpoint after it hits the setpoint at time period 1300. The optimum temperature of each stage as determined by the optimizer is different from stage to stage. The temperature setpoint change is performed in a ramp fashion from 3800 to 4300 seconds of time period. The ramp setpoint change is a rather common practice instead of step changes because the step setpoint change usually

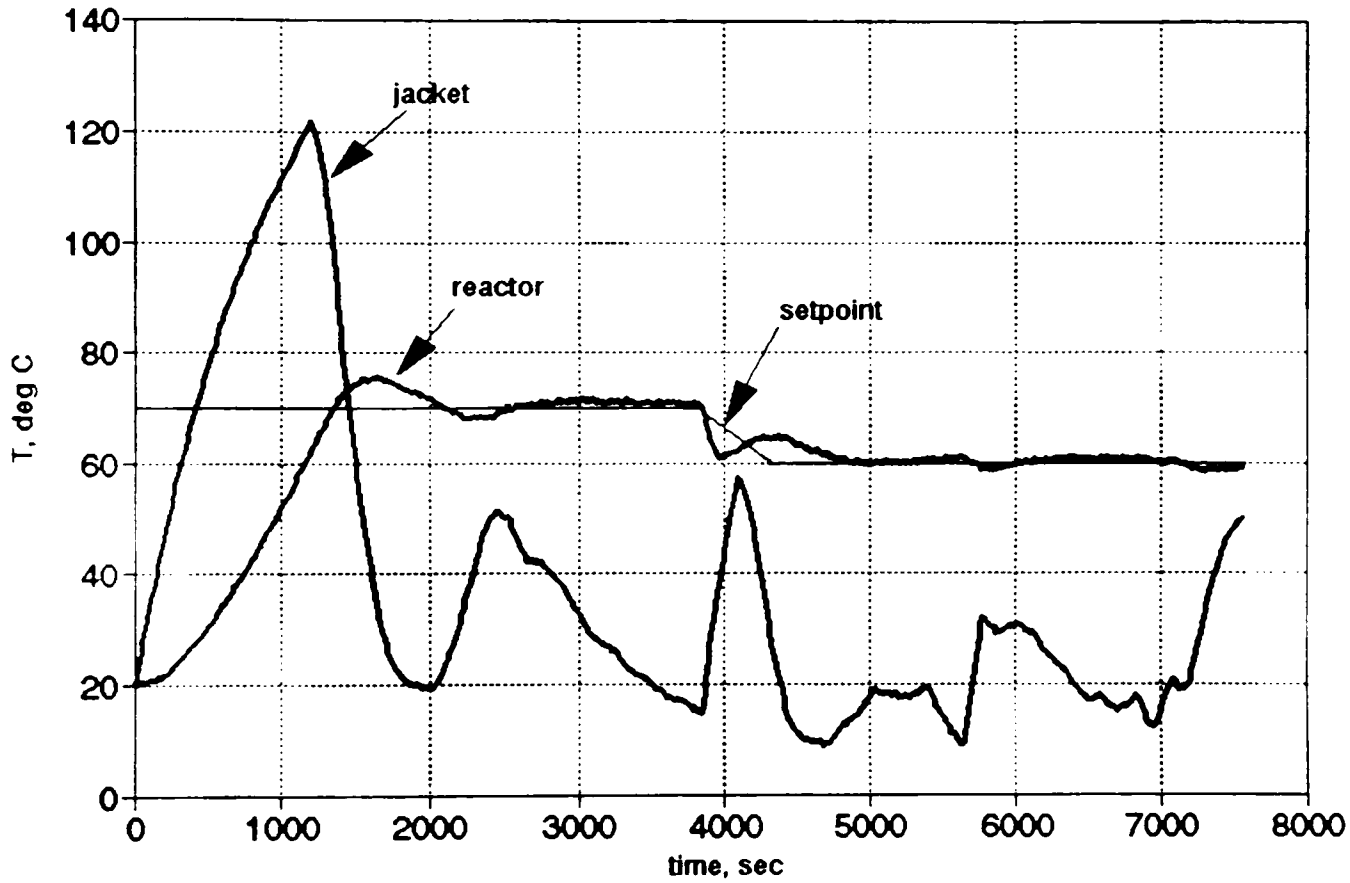


Figure 7.35
Reactor and Jacket Temperature versus Time

leads to a large overshoot on the response. The ramp change interval, 500 seconds, is reasonably chosen considering the sluggish process behavior. The measure of reactor temperature controllability is shown in Figure 7.36. It is defined as a ratio of the heat generation rate to the maximum possible heat transfer rate (see section 3.5). One can see that the controllability is improved at times of 3700 and 5300 seconds when the new stages start and the solvent is added to the system. Figure 7.36 suggests that the temperature controllability (predicted by controller) must be improved because the predicted value of $Q_g/Q_{trans,max}$ is above 0.5 after the second stage began. One should realize that the true possible value in Figure 7.36 would not be known on a real run. To improve the controllability, one might have to consider using even lower temperature of the cooling medium or enhancing the heat transfer coefficient by adding more solvent.

The entire 2 hours of process and control (and much I/O) simulation took about 30 minutes of execution time on a 33 MHz 486 based PC. The implication is that the control effort would require less than 25% of controller time in a real-time run. The control could easily be executed in contemporary micro-processor based programmable controller.

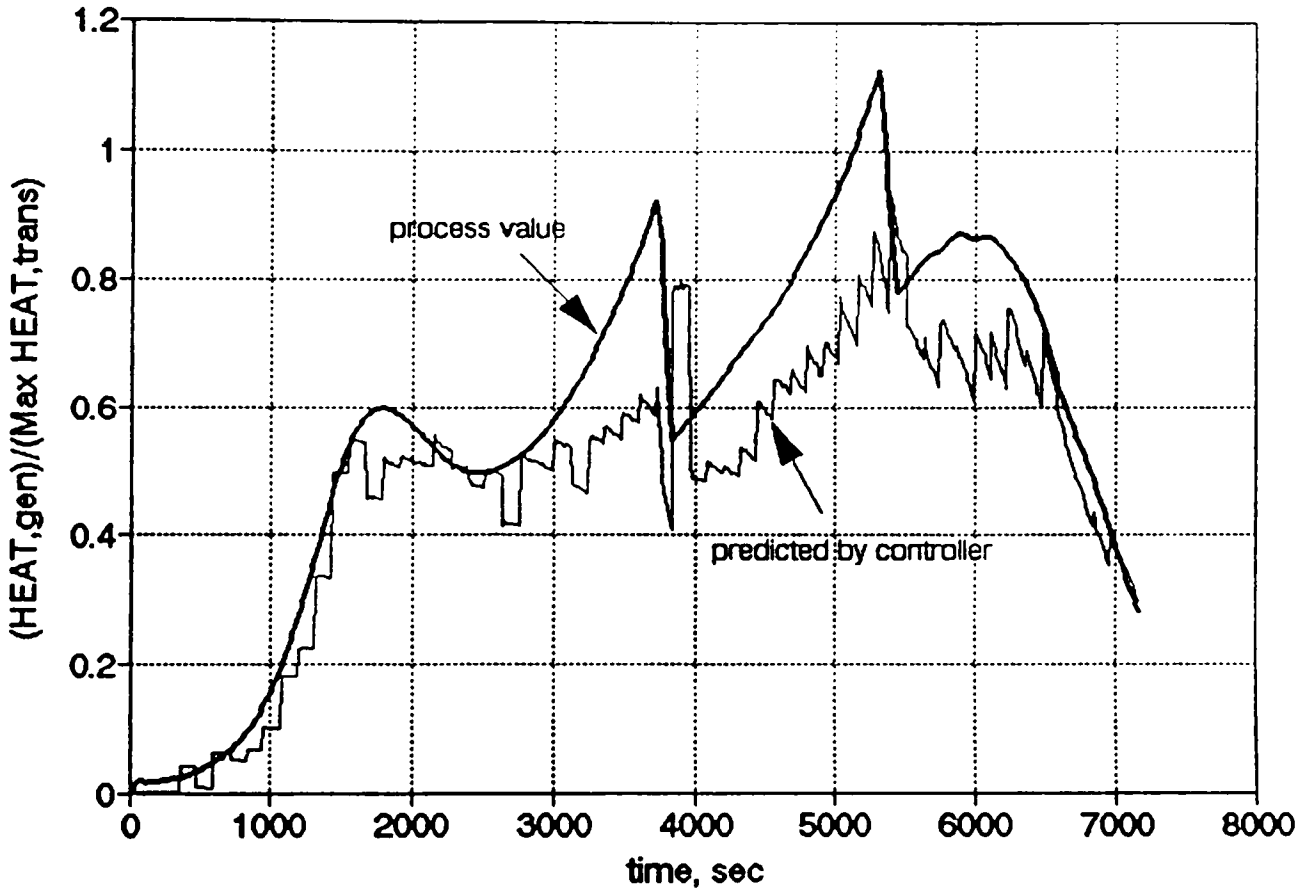


Figure 7.36
Measurement of Temperature Controllability

7.5 Optimization Results and Discussion

As a demonstration, the process described in Figures 3.1 to 3.6 and Table 3.1 is to be optimized. As discussed in Chapter VI, the reactor simulator is being run for the functional evaluation which is required by the optimizer. The reactor simulator adopted "PM6" or "PM7" in Table 7.1 as a process model according to the choice of MWD target. "CM4" is adopted as a control model.

The batch time is chosen to be divided into three stages,

$$\text{Stage I} : \quad 0.0 < x \leq 0.4$$

$$\text{Stage II} : \quad 0.4 < x \leq 0.7$$

$$\text{Stage III}: \quad 0.7 < x \leq 0.97$$

and 50 kgmole of monomer is loaded at the beginning and no monomer is added during the reaction. The initiator loading schedule is prefixed as:

$$FI_{I} = 0.12 \text{ kgmole}$$

$$FI_{II} = 0.04 \text{ kgmole}$$

$$FI_{III} = 0.0 \text{ kgmole}$$

Therefore three decision variables on solvent feeding (FS_{I}, FS_{II}, FS_{III}) and three decision variables on temperature setpoint (T_{I}, T_{II}, T_{III}) are to be searched by the optimizer. The complex method requires bounds to be input for each of the decision variables. They are set as follows:

$$30.0 < FS_{I} < 60.0$$

$$15.0 < FS_{II} < 50.0$$

$$0.0 < FS_{III} < 10.0$$

$$55.0 < T_{I} < 80.0$$

$$55.0 < T_{II} < 80.0$$

$$55.0 < T_{III} < 80.0 \ .$$

These bounds are reasonably set considering that:
(1) Smaller amounts of solvent could lead to temperature uncontrollability; (2) Bigger amounts of solvent slow down the conversion rate and reduce average molecular weight; (3) Lower temperature slows down the conversion rate; (4) Higher temperature could lead to a temperature runaway and reduce the average molecular weight.

From the lower and upper solvent loading bounds for each period, the lower and upper total solvent feeding are bounded as

$$45.0 < \Sigma FS_{i} < 120.0 \ .$$

A total of four cases are tested as presented in Table 7.3. The first three cases used the moment model as a polymerization model ("PM6" in Table 7.1), and the class model ("PM7" in Table 7.1) is used for the last case. So, Cases I,II and III used Eq.(6.1) as the objective function, and Eq.(6.2) is used as a objective function for Case IV.

The starting values for the decision variables are the same for all cases. They are:

$FS_{I} = 58.0$ kgmole $\qquad T_{I} = 66.0$ °C

$FS_{II} = 38.0$ kgmole $\qquad T_{II} = 70.0$ °C

Table 7.3
Summary of The Optimization Result

For the CASE I

target	weight	start	end
X_n=2500.0	w1 = 0.1	f1 =0.16615*10^6	f1 =0.275859*10^1
PDI=0.0	w2 = 2000.0	f2 =0.11663*10^5	f2 =0.986066*10^4
TT97%=0.0	w3 = 0.1	f3 =0.73351*10^5	f3 =0.252280*10^4
ΣFS_i=0.0	w4 = 0.5	f4 =0.49005*10^4	f4 =0.218183*10^4
		sum=0.25606*10^6	sum=0.14568*10^5

Optimum decision variables are:

FS_I = 0.430063*10^2	T_I = 65.6653
FS_{II} = 0.225576*10^2	T_{II} = 59.7899
FS_{III} = 0.494053*10^0	T_{III} = 55.7795

The results with these optimum values are:
Xn=2505 PDI= 2.22 TT97%= 9530 sec ΣFS_i=.6606*10^2 kgmole

For the CASE II
Target chain-length is 500 lower than that of CASE I.
Others remain the same as the CASE I.

target	weight	start	end
X_n=2000	w1 = 0.1	f1= 0.62251*10^5	f1= 0.32281*10^1
PDI=0.0	w2 = 2000.0	f2= 0.11663*10^5	f2= 0.91279*10^4
TT97%=0.0	w3 = 0.1	f3= 0.73351*10^4	f3= 0.21805*10^4
ΣFS_i=0.0	w4 = 0.5	f4= 0.49005*10^4	f4= 0.22910*10^4
		sum=0.86149*10^5	sum=0.13602*10^5

Optimum decision variables are:

FS_I = 0.434836*10^2	T_I = 66.706
FS_{II} = 0.240663*10^2	T_{II} = 64.8393
FS_{III} = 0.140107*10^0	T_{III} = 57.8048

The result with these optimum values are:
Xn=1994 PDI= 2.14 TT97%= 8860 sec ΣFS_i=.6769*10^2 kgmole

Table 7.3 (Continued)

For the CASE III

2nd weighting factor is halved comparing to CASE II.
Others remain the same as the CASE II.

target	weight	start	end
X_n=2000	w1 = 0.1	f1= $0.62251*10^5$	f1=$0.469162*10^2$
PDI=0.0	w2 = 1000.0	f2= $0.58317*10^4$	f2=$0.479098*10^4$
TT97%=0.0	w3= 0.3	f3= $0.22005*10^5$	f3=$0.663053*10^4$
ΣFS_i=0.0	w4 = 0.5	f4= $0.49005*10^4$	f4=$0.240724*10^4$
		sum=$0.94988*10^5$	sum=$0.13876*10^5$

Optimum decision variables are:

FS_I = $0.422520*10^2$	T_I = 65.6053
FS_{II} = $0.242605*10^2$	T_{II} = 65.6447
FS_{III} = $0.287395*10^1$	T_{III} = 60.3921

The results with these optimum values are:
Xn=1978 PDI= 2.19 TT97%= 8920 sec ΣFS_i=$.6939*10^2$ kgmole

For the CASE IV

target
LF = 20 wt% for the first 10 classes out of total 300 classes (400 species/class).
UF = 1 wt% for the last 260 classes out of total 300 classes (400 species/class).
TT97% = 0.0
ΣFS_i=0.0

weight	start	end
w1 = $2.0*10^5$	f1=$0.16272*10^4$	f1=$0.4205*10^2$
w2 = $7.0*10^7$	f2=$0.17500*10^4$	f2=$0.2520*10^2$
w3 = 0.1	f3=$0.73351*10^4$	f3=$0.2016*10^4$
w4 = 0.5	f4=$0.49005*10^4$	f4=$0.2238*10^4$

Optimum decision variables are:

FS_I = $0.512*10^2$	T_I = 71.8
FS_{II} = $0.150*10^2$	T_{II} = 63.9
FS_{III} = $0.730*10^0$	T_{III} = 55.0

The results with these optimum values are:
LF=0.2145 HF=0.0106 TT97%=8520 sec. ΣFS_i=66.9 kgmole

FS_{III} = 3.0 kgmole T_{III} = 56.0 °C

The results with these initial decision variables are:

Xn=1211 PDI =2.42 TT97%=16250 sec ΣFS_i=99 kgmole.

The product target values, weighting factors (w1,w2,w3,w4), starting and final individual objective function values (f1,f2,f3,f4) and combined objective function value (f1+f2+f3+f4), and the optimium decision variables are listed below.

It took about 72 hours to run Cases I, II, III on an IBM RS 6000/320H computer. It took about 20 days to run the Case IV on the same computer. The complex method is not really fast in searching for the optimum point, but it is robust.

When the optimum is reached, one should check whether any of the decision values hit its bound or not. If it was hit, one should consider expanding the current bounds and rerunning the optimization program again. The optimum was not on constraints in Cases I, II, III. The constraints of FS_{II} and T_{III}, however, were hit in Case IV. The corresponding variable's bound should be expanded, and the optimization program should be rerun. Furthermore, there is no guarantee for the optimum point to be a global optimum. However, the fact that various initial guesses gave the same optimum provides knowledge that the surface is well behaved and confidence that the optimum point obtained could be a global optimum.

Although, in this demonstration, $S_{I,II,III}$ and $T_{I,II,III}$ are selected as decision variables and the batch "time" is divided into three conversion stages, it would be also possible to consider initiator or monomer as additional decision variables and to divide the batch time into more than three stages.

Using the optimized $T_{I,II,III}$ and $FS_{I,II,III}$ variables for Case II, the optimization program was run to find the optimum values for the initiator loading, I_I, I_{II}, I_{III}. The same weighting factors are used as in Case II. The starting values of these variables are those used as fixed variables for Case II. No significant reduction in the objective function values is found even after 100 iterations. The values of these variables at the last run are:

$FI_I = 0.119649*10^{0}$,

$FI_{II} = 0.401266*10^{-1}$,

$FI_{III} = 0.228377*10^{-3}$.

These values virtually have not been changed from the starting values. This implies that the optimum of decision variables in Case II are found such that those fixed and decision variables altogether become the optimum.

The optimization results have been varied with the choices of product and operation target. One should also realize that the results presented here could have been different if a different type of controller were employed for the temperature control.

It has been tested whether the optimum decision variables calculated by the optimizer can reproduce the product and operation target that the optimizer tried to meet when the realistic reactor simulator was run with them. Here the realistic reactor simulator adopted "PM5" and "CM4" as a process and control model, respectively, along with MWD modeling equations so that, at the end of run, MWD can be identified. The realistic reactor simulator was run with the optimum of decision variables for Case I, and it gives

X_n=2541 PDI=2.221 TT97%=9470 sec.

No significant deviation from the optimizer results is found although they have been obtained by the realistic reactor simulator. Furthermore this test has also been performed with an additional source of simulated measurement error: monomer is loaded by 5% less than the measured value, which means that measured is 50 kgmole (used for optimization) but actual loaded amount is 47.5 kgmole. It gives

X_n= 2424, PDI=2.224 TT97%=9850 sec.

Similarly, when solvent is loaded at each stage by 5% less than the measured value, it gives

X_n= 2474 PDI=2.292 TT97%=8470 sec.

Similarly, when initiator is loaded at each stage by 5% less than the measured value, it gives

X_n= 2636 PDI= 2.212 TT97%= 10110 sec.

These results imply that the off-line optimized batch properties are not extremely sensitive to several common errors in manufacturing implementation of the recipe.

CHAPTER VIII

CONCLUSION AND RECOMMENDATION

8.1 Conclusion

A set of kinetic differential equations for a postulated free radical homopolymerization is transformed into a set of moment differential equations using the generating function technique. The process simulator based on the moments for modeling of the molecular weight distribution (MWD) provides dynamic behavior of key MWD parameters, average, variance, skewness and kurtosis.

The class method as a polymerization modeling approach enables a visualization of the entire molecular weight distribution which is not possible with the method of moments. The class method groups adjacent species into classes, reduces the number of differential equations which must be solved, and yields moments value close to the true "moment" method. The accuracy of class method is greatly dependent on how finely the classes are discretized. The finer the class discretization, the closer to the true moments it will approach. This takes, however, more computation time.

It is demonstrated that NPMBC control works for temperature control of both the S.C. Johnson & Sons reactor and the industrial scale rigorous kinetics reactor which are both characterized by highly nonlinear and nonstationary

process behavior. Noise and disturbances are added to the measurable process variables to make the simulator more realistic.

The controller is found to be robust when the process noise is filtered by either the CUSUM-filter or the conventional first-order filter. For the S.C. Johnson & Sons reactor control, it was demonstrated that CUSUM filtering can be performed not only on the PV but also to the MV and controller model parameters. Although the new filtering technique, the SPC-based filter, did not outperform the conventional filter, it suggests that this new way of filtering works as well.

The NPMBC was effective for the sluggish process of industrial scale reactor.

The split-range cascade PID control, is also implemented and tested for the S.C. Johnson & Sons reactor. Since the conventional controller cannot accomodate the nonlinear/nonstationary controller performance, it fails to meet the control performance criteria.

The process optimizer, based on the complex method, used the reactor simulator for function evaluation, while the reactor simulator was being controlled by the NPMBC controller. Either the method of moments or the class method can be employed for the process optimization depending on the user's product and operational objectives. The complex

method as an optimizer searches the optimum point slowly but it is robust.

8.2 Recommendation

The class method is a new approach in the area of modeling of entire molecular weight distribution. To draw attention on this technology, it might be necessary to present a performance comparison with other techniques such as the continuous variable approximation. The performance can be compared by measuring the time required to give the same accuracy, how closely the yielded moments match to the true moments.

Although, in this study, a chain-length independent termination rate constant is used in the polymerization kinetics, it seems important that the continuation of the class method study should demonstrate its capability of handling the chain-length dependent termination rate constant because later in the reaction termination is strongly diffusion-controlled and therefore strongly dependent upon the sizes of the chain length of growing polymer. One might encounter a difficulty with the application of the method of moments since the chain length dependent term $k_t(n,m)$ introduces severe nonlinearities into the equations.

The film heat transfer coefficient, h_i, correlation used for the industrial scale reactor does not necessarily

represent the actual heat transfer of this reactor system because it was originally formulated suitable for the S.C. Johnson & Sons reactor problem. It has been modified to provide a reasonable overall heat transfer coefficient, U, for the industrial scale reactor. If it were modified such that a higher heat transfer coefficient can be obtained with less amount of solvent, a bimodal type distribution could be obtained in Figure 7.25 due to the gel effect. Continuation of this simulation study might need to consider this point.

It might be worth investigating the utility of decision variables in the optimization so that one can judge which set of decision variables are most influential on the product and operation target.

The batch time is divided into three stages in the optimization study. It is also worth trying both a smaller and larger stage division, and watch how the optimization result is affected.

The simulation study shows that the optimization and the control techniques on the polymerization process are very promising. The next step in this research should be an experimental demonstration. If the kinetic constants vary significantly from batch to batch, the optimum recipe determined by off-line optimization will be no longer valid. If this is the case, one should apply an on-line optimization technique to the run whose optimum policy is predetermined by off-line optimization.

REFERENCES

Agarwal, M., Bonvin, D., "State Estimation Accounting for Uncertainty in Decoupled Parameter Estimation," AIChE Annual Meeting, San Francisco, 1989.

Amundson, N.R., Zeman, R.J., "Continuous Polymerization Models-I and II," Chem. Eng. Sci., 1965, Vol.20, pp.331-361 and pp.637-664.

Baillagou, P.E., Soong, D.S., "Major Factors Contributing to the Nonlinear Kinetics of Free-Radical Polymerization," Chem. Eng. Sci., Vol.40, No.1, 1985.

Balke, S.T. and Hamielec, A.E.,"Bulk Polymerization of Methyl Methacrylate," J. Appl. Polym. Sci., 17, 905, 1973.

Biesenberger,J.A., Sebastian, D.H., Principles of Polymerization Engineering, Wiley-Interscience, New York, pp.681-684, 1983.

Bogunjoko, J.S.T., Brooks, B.W., "Molecular Weight Distributions of Poly(methyl methacrylate) Produced at High Viscosities," Macromol. Chem., 184, 1603, 1983.

Box, G.E.P., Jenkins, G.M., Time Series Analysis, Forecasting and Control, Revised, Holden Day, Oakland, CA, 1976.

Chen, S., Lin, K.,"Minimum end time Policies for Batchwise Radical Chain Polymerization-II," Chem. Eng. Sci., Vol. 35, pp.2325, 1980.

Chiu, W.Y., Carratt, G.M., Soong, D.S., "A Computer Model for the Gel Effect in Free-Radical Polymerization," Macromolecules, 16,348-357, 1983.

Choi, J.Y., Rhinehart, R.R., Riggs, J.B., "Nonlinear Model-Based Control of a Batch Polymerization Reactor," AIChE Annual Meeting, Miami, FL, 1992.

Chylla, R.W., Haase, D.R.,"AIChE Industrial Challenge Problem: Temperature Control of Semibatch Polymerization Reactors," Presented AIChE Annual Meeting, Chicago, IL., 1990.

Cluett, W.R., Shah,S.L., Fisher, D.G., "Adaptive Control of a Batch Reactor," Chem. Eng. Commun., Vol.38, 1985.

Cott, B.J., Macchietto, S., "Temperature Control of Exothermic Batch Reactors Using Generic Model Control," Ind. Eng. Chem. Res., Vol.28, No.8, 1989.

Coyle D.J., Tulig, T.J., Tirrell, M., "Finite Element Analysis of High Conversion Free-Radical Polymerization," Ind. Eng. Chem. Fundam., Vol.24, No.3, 1985.

Fevotte, G., Klein, J.P., "Predicitve Temperature Tracking for Batch Evaporative Crystallizers: The Partial State Model References Approach," Journal of Process Control, Vol.3, No.4, 1993.

Friedman, Y.Z., "Model-Based -Control of Crude Product Qualities," Hydrocarbon Processing, Vol.73, No.2, 1994.

Gill, P.E., Murray, W., Wright, M.H., Practical Optimization, Academic Press, New York, pp.205, 1981.

Goodrich, F.C., "The Numerical Analysis and Kinetic Interpretation of Molecular Weight Distribution," in Chapter F of Polymer Fractionation, Editor, Cantow, M.J.R, Academic Press, New York, 1967.

Hamielec, A.E., Hodgins, J.W., Tebbens, K., "Polymer Reactors and Molecular Weight Distribution:Part II. Free Radical Polymerization in a Batch Reactor," AIChE Journal, November 1967.

IMSL Math/Library Volume 3, Chapter 8-10, 785-1152, IMSL Inc., Houston, TX, 1987.

Juba, M.R., Hamer, W., "Progress and challenges in Batch Process Control," In Proceedings of the Third International Conference on Chemical Process Control; Morari, M, McAvoy, T., (Eds.); Elsevier Science Publishers, New York, 1986.

Jutan, A., Upal, A., "Combined Feedforward-feedback Servo-Control Scheme for an Exothermic Batch Reactor," Ind. Engng. Chem. Process Des. Dev., 23, 1984.

Kiparissides, C., Shah, S.L., "Self-Tuning and Stable Adaptive Control of a Batch Polymerization Reactor," Automatica, 19, 1983.

Knuth, D.E., "Seminumerical Algorithms," in The Art of Computer Programming, 2nd ed., Vol.2, pp.9, Addison-Wesley, Reading, MA.

Kravaris, C. and Wright, J.F., Carrier, J.F., "Nonlinear Controllers for Trajectory Tracking in Batch Processes," Computers and Chem. Eng., 13 1989.

Lee, P.C., Sullivan, G.R., "Generic Model Control-GMC," Computers and Chemical Engineering, Vol.12, No.6, 1988.

Lenz, R.W., Organic Chemistry of Synthetic High Polymers, Wiley-Interscience, pp.53-565, New York, 1967.

Lewin, D.R., Lavie, R., "Designing and Implementing Trajectories in an Exothermic Batch Chemical Reactor," Ind. Eng. Chem. Res., Vol.29, No.1, 1990.

Liptak, B.G., "Controlling and Optimizing Chemical Reactors," Chemical Engineering, May 26, 1986.

Mahuli, S, Rhinehart, R.R., Riggs, J.B., "Experimental Demonstration of Nonlinear Model-Based In-Line Control of pH," Journal of Process Control, Vol.2, No.3, 1992.

Mahuli, S, Rhinehart, R.R., Riggs, J.B., "pH Control Using a Statistical Technique for Continuous On-Line Model Adaptation," Computers Chem. Engng., Vol.17, No.4, 1993.

Marroquin, G., Luyben, W.L., "Practical Control Studies of Batch Reactors Using Realistic Mathematical Models," Chem. Eng. Sci., 28, 1973.

Meadows, E.S., Rawlings, J.B., "Simultaneous Model Identification and Control of a Semi-Batch Chemical Reactor," Proceedings ACC90, San Diego, CA, 1990.

Merkle, J.E., Lee, W-K, "Adaptive Strategies for Automatic Startup and temperature Control of Batch Processes," Computers Chem. Eng., 13, 1989.

Natarajan, S., Rhinehart, R.R.,"Implementation Techniques for an In-Line pH Controller," America Control Conference, Baltimore, MD, 1994.

Nelder, J.A., Mead, R., "A Simplex Method for Function Minimization," Computer Journal, pp.308, 1965.

Pandit, H.G., Rhinehart, R.R., "Experimental Demonstration of Nonlinear Model-Based Distillation Control," American Control Conference Proceedings, Chicago, IL, 1992.

Paruchuri, V.P., Rhinehart, R.R., "Experimental Demonstration of Nonlinear Model-Based -Control of a Heat Exchanger," ACC Conference, Baltimore, MD, 1994.

Paruchuri, V.P., Rhinehart, R.R., "Experimental Demonstration of a Process-Model-Based Flow Controller," Advances in Instrumentation and Control, Vol. 48, pp 529, 1993.

Rafalimanana, A., Cabassud, M., Le Lann, M.V., Casamatta, G., "Adaptive Control of a Multipurpose and Flexible Simi-Batch Pilot Plant Reactor," Compu. Chem. Eng., Vol.16, No.9, 1992.

Ray, H.W., "On the Mathematical Modeling of Polymerization Reactors," J. of Macromol. Sci., C8, 1, 1972.

Ray, H.W., Laurence, R.L., "IX. Polymerization Reaction Engineering," in Chemical Reactor Theory, A Review, Editor, Lapidus, L. and Amundson, N.R., Prentice-Hall, Englewood Cliffs, NJ, 1977.

Reklaitis, G.V., Ravindran, A., Ragsdell, K.M., Engineering Optimization, John Wiley and Sons, New York, 1983.

Rhinehart, R.R., "Process-Model-Based Control of Wastewater pH Neutralization," InTECH, Vol.37, No 7, 1990.

Rhinehart, R.R., "Model-Based Control," in Chapter II. Process Control And Analysis of Instrument for Engineers' Handbook, Editor, Liptak, 3rd Ed., Chilton, Rodnor, PA, in press, 1994.

Rhinehart, R.R., "A Self-Tuning Filter and Process Changes Identified for Continuous Variables," Advances in Instrumentation and Control, Vol.45, Part 2, 1990.

Rhinehart, R.R., "A CUSUM Type on-line filter, Process Control and Quality," Elsevier Science Publishers B.V., Amsterdam, February 1992.

Rhinehart, R.R., Riggs, J.B., "Two Simple Methods for On-Line Incremental Model Parameterization," Computers and chemical Engineering, Vol.15, No 3, 1991.

Rhinehart, R.R., Felder, R.M., Ferrell, J.K., "Process-Model Based Control of a Pilot-Scale Flidized Bed Coal Gasification Reactor," Advances in Instrumentation and Control, Vol.43, Part 1, 1988,

Rhinehart, R.R., Choi, J.Y., "Process-Model Based Control of a Wastewater pH Neutralization," Advances in Instrumentation and Control, Vol.43, Part 1, 1988.

Riggs, J.B., "Nonlinear Process-Model-Based Control for a Propylene Sidestream Draw Column," I&ECR, Vol.29, No.11, 1990.

Riggs, J.B., Watts, J., Beauford, M., "Model-Based Control Streamlines Process," Control, July 1991.

Rodriguez, F., Principles of Polymer Systems, (3rd ed.), Hemisphere Publishing Co., pp.86-94, New York, 1989.

Ruppen, D., Rippin, D.W.T., Bonvin, D., "An On-Line Optimization Strategy For Fast Batch and Semi-Batch Reaction Systems," Proceedings of IFAC; Advanced control of Chemical Processes, Toulouse, France, 1991.

Sacks, M.E., Lee, S., Biesenberger, J.A., "Optimal Policies for Batch Chain Addition Polymerization," Chem. Eng. Sci., Vol.27, 1972.

Shinnar, R., Katz, S., " Polymerization Kinetics and Reactor Design," in "Chemical Reactor Engineering," Advances in Chemistry, Vol.109, American Chemical Society, Washington D.C., 1972.

Shinskey, F.G., Weinstein, J.L., "A Dual-Mode Control System for a Batch Exothermic Reactor," 20th Annual ISA Conference, Los Angeles, CA, 1965.

Subawalla, H., Rhinehart, R.R., "Experimental Comparison of Model-Based and Conventional Pressure Control for a Plasma Reactor," America Control Conference, Baltimore, 1994.

Takamatsu, T.S., Okada, Y., "Design of Adaptive/Inferential Control System and its Application to Polymerization Reactors," Ind. Eng. Chem. Process Des. Dev., 25, 1986.

Taylor, T.W., Gonzales V., Jensen K.F., "Modeling and Control of Molecular Weight Distribution in Methyl Methacrylate Polymerization," Polymer Reaction Engineering, 261-273, 1986.

Zhao, H., Isaacs, S.H., Soeberg, H., Kummel, M., "Nonlinear Optimal Control of an Alternating Activated Sludge Process in a Pilot Plant," J. of Proc. Cont., Vol.4, No.1, 1994.

APPENDIX A

AIChE INDUSTRIAL CHALLENGE PROBLEM: NPMBC APPLICATION

A.1 Summary

The AIChE Industrial Challenge Problem (Chylla and Hasse, 1990 Annual Meeting Paper 24 c), "Temperature Control of Semibatch Polymerization Reactor," given by Chylla and Haase, of S.C. Johnson and Sons, Inc. has been solved by Nonlinear Process-Model-Based Control (NPMBC) which is described in Chapter V. This control problem concerns the temperature control of a reactor used to make specialty emulsion polymers and is to design a controller capable of maintaining the desired reaction temperature throughout the batch reaction time. The controller works throughout a wide range of tuning values, scan periods, and choices of feedback mechanism: It is robust to implementation choices.

The jacketed semi-batch emulsion polymerization reactor can either be steam heated or water cooled. The objective is to develop a single water/steam controller which will keep the reactor at a setpoint temperature throughout the batch campaign and inspite of recipe and disturbance changes. The process schematic is illustrated in Figure A.1. The water-filled, jacketed batch reactor is first heated by steam injection. When the reactor contents reach 180°F, the controller begins regulating jacket steam and cold water flow to maintain the reactor at 180°F$\pm$1°F, also monomer feed begins at a constant rate for 70 minutes. Cold monomer inflow and ambient losses would require heat, but the heat of polymerization requires a net cooling. As the reaction

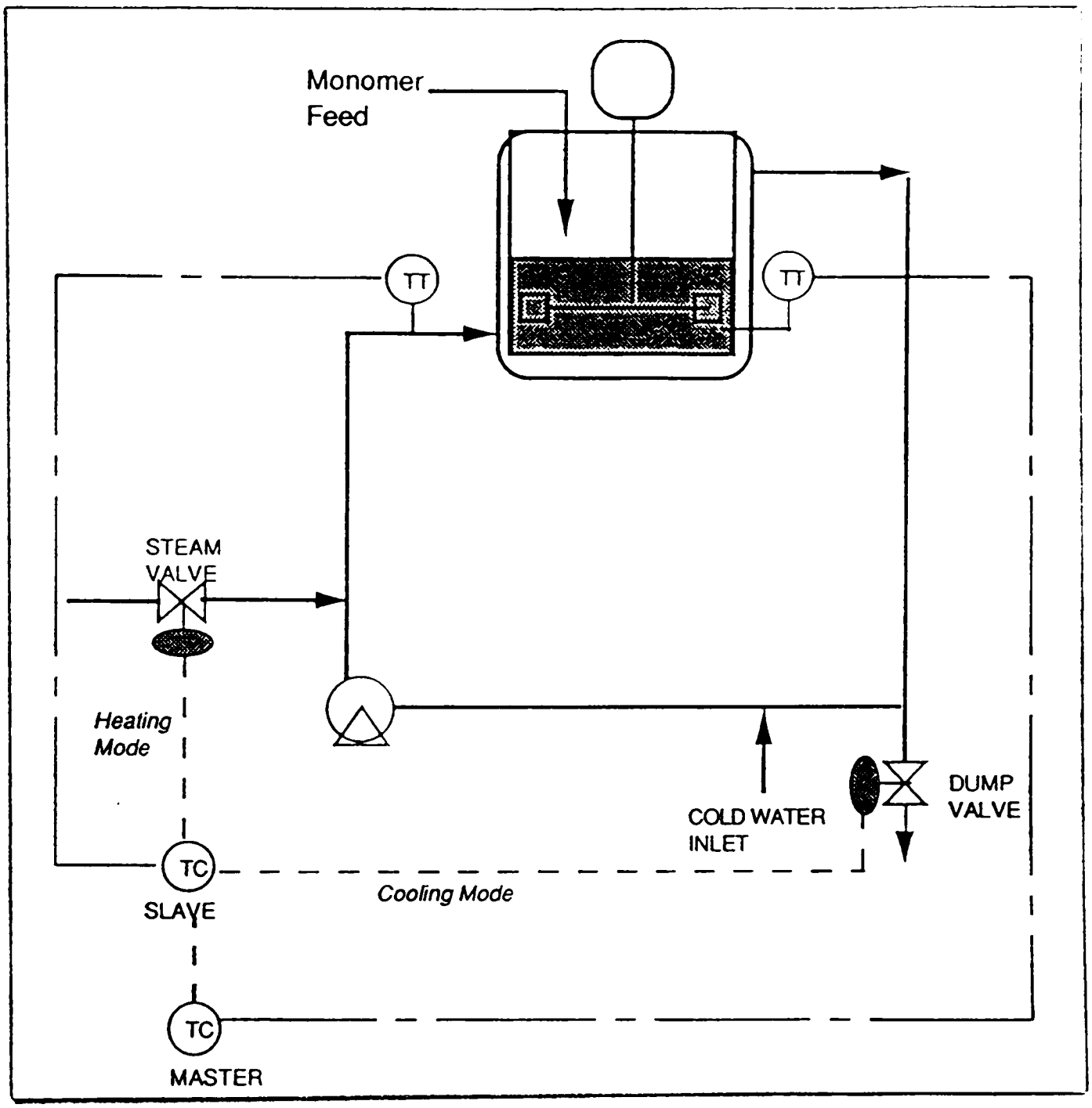


Figure A.1
Reactor Schematic and Control Element
(SC Johnson & Sons Reactor)

proceeds, the heat duty changes and the steam or water rates must be continuously adjusted. Further, as the reaction proceeds, the heat transfer coefficient changes requiring a change in the dynamics of valve action. The process is nonlinear and nonstationary.

Conventional cascaded PID control gives a sluggish control response because shifting of the slave output signal to either of the final control elements lags behind the master output. The switch becomes possible only after the sign of the slave controller actuating error changes. The sign change is sluggish due to the reset windup on the master controller. Even when an anti-windup technique called "Internal Reset Feedback" is used, conventional control can not accommodate the nonlinear/nonstationary controller performance. Conventional control is still not sufficient to meet the control performance criteria.

While PID and linear model-based controllers essentially ignore material and energy balance information, nonlinear process-model-based control (NPMBC) strategies use this information directly. In NPMBC, the process is controlled by using models based on, but not exactly the same as the process. By comparison to linear and PID-type approaches the NPMBC action is "intelligent" and has shown benefits in uniformity, disturbance rejection, and setpoint tracking. All of which translate into better process economics.

This Appendix describes control results for a variety of process conditions. The open-loop response data are also included to contrast the process behavior without control. Results of conventional cascaded PID control is also presented as a comparison with the NPMBC control strategy.

A.2 Process Description

A stirred tank reactor with a working volume of 30 gallons is used to make specialty emulsion polymers in a semi-batch processes. Sufficient agitation is provided such that the reactor can be assumed to be well-mixed. However, mechanical energy generated by the agitator is assumed to be negligible. The reactor jacket has a volume of 5.7 gallons and spiral-cut internals to facilitate very high jacket side heat transfer coefficient. The recirculation line volume is calculated to contain 3.7 gallons (considering 35 ft of 1.61" ID line) for a total of 9.4 gallons of recirculation loop volume. As shown in the reactor and control schematic (Figure A.1), PID controllers are implemented on both the master and slave loops. The new NPMBC control strategy will replace both of the controllers. The sensors and final elements will be kept as is.

The polymerization reaction is subject to several nonstationary effects. These include batch-to-batch variations in polymer reactivity and fouling factor, a change in internal heat transfer coefficient during

polymerization as viscosity changes, and a change in required cooling rate as the reaction initiates then depletes. Phenomenological expressions for kinetics and several constitutive relations for viscosity and heat transfer coefficient are given in the S.C. Johnson & Son, challenge problem statement. Detailed geometric data of the reactor and recirculation loop, and of physical property of reaction components and products recipes are also given (see Tables A.1 to A.3).

The two measurable data in this process are reactor temperature and jacket temperature. The two control valves, for steam supply and circulation water dump, respectively, are final control elements. At the initial stage, the steam valve will be fully opened manually to raise the initial charge temperature up to the set point. Then, the controller mode will be switched to automatic so that the model-based control starts. At this time the jacket temperature is 280 °F and for the first several automatic control actions the dump valve will be fully open.

The maximum flow rate obtainable across the valve when valve is fully open depends on the valve inlet/outlet pressure, fluid physical property and selected C_v. Regardless of valve type, linear or equal percentage, when the valve is wide open, the required (calculated) C_v equals the selected C_v. Control valve sizing formula for subcritical liquid (water) flow is:

Table A.1
Reactor and Recirculation Loop Data

DESCRIPTION	UNITS	VALUE
Reactor Diameter	inches	19.5
Reactor Height	inches	21.0
Working Volume	gallons	30.0
Jacket Volume	gallons	5.7
Jacket Bottoms Area	ft^2	1.8
Jacket Sides Area	ft^2	5.23*height
Water Recirculation Rate	gal/min	15.0
Steam Pressure	psig	150.0
Water Inlet Pressure	psig	75.0
Pump Discharge Pressure	psig	60.0
Water Inlet Temperature (Winter)	°F	42
Water Inlet Temperature (Winter)	°F	70
Ambient Air Temperature (Winter)	°F	45
Ambient Air Temperature (Summer)	°F	90

Control Valves

Dump Valve: Equal Percentage C_v=0.42 2% hysteresis
Steam Valve: Equal Percentage C_v=0.42 2% hysteresis

Reactor Heat Loss

$(UA)_{loss}$ = 10.74 btu/h°F

Table A.2
Product A Data

PRODUCT A DATA			
Arrhenious Prexponential	k_0	min^{-1}	3300
Heat of Polymerization	$-\Delta H_p$	btu/lbmol	30160
Monomer Molecular Weight	MW_M	lb/lbmol	104
Monomer Specific Heat	C_{pM}	btu/lb_F	0.4
Water Specific Heat	C_{pW}	btu/lb_F	1.0
Solids Specific Heat	C_{pS}	btu/lb_F	0.75
Monomer Specific Gravity	ρ_M		0.9
Solids Specific Gravity	ρ_S		1.04
Water Specific Gravity	ρ_W		1.0
Initial Reactor Charge	Solids (t=0)	lbs	119.0
Initial Fraction Solids	f_{solids}°		.208
Initial Fraction Water	f_{water}°		.792
Initial Fraction Monomer	f_{mon}°		0.0
Reaction Temperature Setpt	T_s	°F	180.0

Product A Recipe

1) Place initial charge of solids and water into the reactor at ambient temperature.
2) Raise temperature of the initial charge to the reaction temperature setpoint.
3) Feed pure monomer into the reactor at 1.0 lb/min for 70 minutes.
4) After the feed addition period is complete, hold at reaction temperature for 60 minutes.

Table A.3
Product B Data

PRODUCT B DATA			
Arrhenious Prexponential	k_0	min^{-1}	1200
Heat of Polymerization	$-\Delta H_p$	btu/lbmol	28200
Monomer Molecular Weight	MW_M	lb/lbmol	106
Monomer Specific Heat	C_{pM}	btu/lb_F	0.4
Water Specific Heat	C_{pW}	btu/lb_F	1.0
Solids Specific Heat	C_{pS}	btu/lb_F	0.75
Monomer Specific Gravity	ρ_M		0.9
Solids Specific Gravity	ρ_S		1.04
Water Specific Gravity	ρ_W		1.0
Initial Reactor Charge	Solids (t=0)	lbs	93.3
Initial Fraction Solids	f_{solids}°		.204
Initial Fraction Water	f_{water}°		.796
Initial Fraction Monomer	f_{mon}°		0.0
Reaction Temperature Setpt	T_s	°F	176.0

Product B Recipe

1) Place initial charge of solids and water into the reactor at ambient temperature.
2) Raise temperature of the initial charge to the reaction temperature setpoint.
3) Feed pure monomer into the reactor at 0.8 lb/min for 60 minutes.
4) After the feed addition period is complete, hold at reaction temperature for 30 minutes.
5) Feed pure monomer into the reactor at 0.8 lb/min for 40 minutes.
6) After the second feed addition period is complete, hold at reaction temperature for 45 minutes.

$$C_{v,req} = Q_w\sqrt{\left(\frac{SG}{\Delta p}\right)}. \qquad (A.1)$$

control valve sizing formula for saturated steam is: For subcritical flow,

$$C_{v,req} = \frac{w_s}{2.1\sqrt{\Delta p(p_1+p_2)}}. \qquad (A.2)$$

For critical flow,

$$C_{v,req} = \frac{w_s}{1.83C_f p_1}. \qquad (A.3)$$

where:

C_f = Critical flow factor,

p_1 = Upstream pressure, psia,

p_2 = Downstream pressure, psia,

Δp = Pressure drop, p_1-p_2, psi,

Q_w = Water flow rate, gpm,

w_s = Steam flow rate, pound per hour.

Thus for dump valve,

$$Q_w = 0.42\sqrt{\frac{\Delta p}{SG}} = 0.42\sqrt{\frac{60}{1}} = 3.2533 \ gpm.$$

For steam valve, formula for critical flow should be applied because $\Delta p \geq 0.5C_f p_1$. However since C_f is not given and Δp is not so greater than $0.5p_1$, we used formula 2. Thus for

the steam valve,

$$w = 0.42 \cdot 2.1 \sqrt{\Delta p (p_1 + p_2)} = 0.42 \cdot 2.1 \sqrt{90(164.7+74.7)} = 129.4 \ lb/hr.$$

(Note: Control valve sizing formula is adopted from "Masoneilan Handbook for Control Valve Sizing.")
The inlet pressure of the steam valve is taken as steam supply pressure of 150 psi. The steam valve outlet pressure is approximated as a pump discharge pressure of 60 psi because dynamic loss from pump discharge to steam valve outlet is negligible compared to pump discharge pressure of 60 psi (assuming jacket volume and recirculation loop line are sized big enough to ignore the friction loss over the entire recirculation loop). The pressure of the recirculation loop is supposed to be balanced immediately by the back-pressure regulating valve when the dump valve opens and discharges water. They are calculated to be 2.16 lb/min for the steam valve and 3.25 gpm for the dump valve.

A.3 Process Equations

An extremely simple kinetic model is given in the problem statement. The model description does not describe the chemical composition, particle size distribution, nucleation or initiation mechanism. The simple material balance is given in terms of moles of monomer:

$$\frac{dn_M}{dt} = F_M - R_p. \qquad (A.4)$$

$$F_M = \frac{\dot{m}_M}{MW_M} . \qquad (A.5)$$

$$R_P = i(k \cdot n_M) . \qquad (A.6)$$

$$k = k_0 \exp\left(-\frac{6400}{T_r+460}\right) \mu^{0.4} . \qquad (A.7)$$

The conversion is modeled using first order kinetics. The only modification is the impurity factor "i" preceding the kinetic constant of Eq.(A.6). The impurity factor takes on a random value between 0.8 and 1.2 for any particular batch. By this way, the model represents the apparent fluctuation in monomer kinetics caused by batch-to-batch variation in reactive impurities. The energy balance around the reactor is described in Eq.(A.8).

$$\frac{d(\sum m_l Cp_l T_r)}{dt} = \dot{m}_M Cp_M (T_{amb} - T_r) + R_p (-\Delta H_p)$$

$$- \frac{1}{1/h_i + 1/h_f} \cdot A \cdot (T_r - T_j) - (UA_{loss}) (T_r - T_{amb}) . \qquad (A.8)$$

The energy balance around the jacket is described in Eq.(A.9). For cooling, there will be no flow on $\dot{m}_s$. For heating $\dot{v}_w$ shall be zero.

$$(V_j\rho_{w,j}Cp_{w,j})\frac{dT_j}{dt} - \dot{v}_w\rho_w C_{p,w}(T_w-T_j) + \dot{m}_s(\lambda_s+C_{p,w}(T^o-T_j)) +$$

$$-m_s C_{p,w}T_j + \frac{1}{1/h_i+1/h_f}A(T_r-T_j)\,. \qquad (A.9)$$

In Eq.(A.9), V_j is the sum of the jacket volume and recirculation loop volume. The set of differential equations are solved simultaneously by the 4th-order Runge-Kutta algorithm with a step size controller chosen such that the maximum relative change in dependent variables is 5% per time increment.

The viscosity of the reactor contents is given as a function of both solid fraction in the reacting mixture and its temperature. Eqs.(A.10) and (A.11) are viscosity correlations for Product A and Product B.

$$\mu = 0.052\cdot e^{(16.4f)}10^{[2.3(A-1.563)]} \quad cp. \qquad (A.10)$$

$$\mu = 0.032\cdot e^{(19.2f)}10^{[2.3(A-1.563)]} \quad cp. \qquad (A.11)$$

where A=1000/(T+460)

The solid mass fraction 'f' is readily calculated by Eq.(A.12) once one knows initial charge, total monomer fed and unreacted monomer left in the reactor.

$$f = \frac{solid}{initial\ charge+unreacted\ monomer+solid}. \qquad (A.12)$$

The monomer remaining in the reactor is obtained by solving Eq.(A.4).

The reactor-side film heat transfer coefficient is given as a function of wall viscosity in Eq.(A.13).

$$h_i = 143.4 \cdot \exp(-5.13e^{-3} \cdot \mu_{wall}). \qquad (A.13)$$

The wall viscosity, μ_{wall}, is determined by Eq.(A.10) or (A.11) evaluated at the wall temperature T_{wall} which is simply the arithmetic mean of reactor and jacket temperature ($T_w=(T_r+T_j)/2$). Although the overall heat transfer coefficient, U, should be calculated by $1/U = \Sigma R_j = R_i + R_{wall} + R_o + R_{f,i} + R_{f,o}$, the problem statement allows neglecting R_{wall}, R_o and $R_{f,o}$ and uses the simplified Eq.(A.14).

$$U = \frac{1}{\frac{1}{h_i}+\frac{1}{h_f}}. \qquad (A.14)$$

During successive batches of the same product, h_f should change as shown in Table A.4.

Table A.4
Reactor Side Fouling Factor (Unit:hr.ft^2.F/btu)

Batch	1	2	3	4	5
$1/h_f$	0.000	0.001	0.002	0.003	0.004

The total effective heat transfer area in this jacketed reactor is the sum of jacket bottom area and reactor inside wetted area.

$$A_e = A_{jb} + A_h. \tag{A.15}$$

$$A_e = (1.8 + 5.23 \cdot h_l). \tag{A.16}$$

$$h_l = h_r \cdot \frac{w_b/\rho_b}{V_R}. \tag{A.17}$$

The density of batch is calculated as follows.

$$\rho_b = \frac{1}{\Sigma(w_i/\rho_i)}. \tag{A.18}$$

where w_i is a weight fraction of component i in the reacting mixture. Since w_b and ρ_b are time dependent, h_l becomes a function of time.

A.4 Uncontrolled Response

Open-loop (uncontrolled) responses show the process characteristics visually. The data shown in this section is based on product recipe A (see Table A.2), reactor fouling factor 0.0025 hr_ft²_F/btu and impurity factor 1.2.

To generate the open-loop (uncontrolled) response, jacket temperature is first raised so that the reactor temperature reaches its set point, and at the moment monomer feed starts. Figure A.2 shows the uncontrolled reactor and jacket temperature response. At the initial stage, the

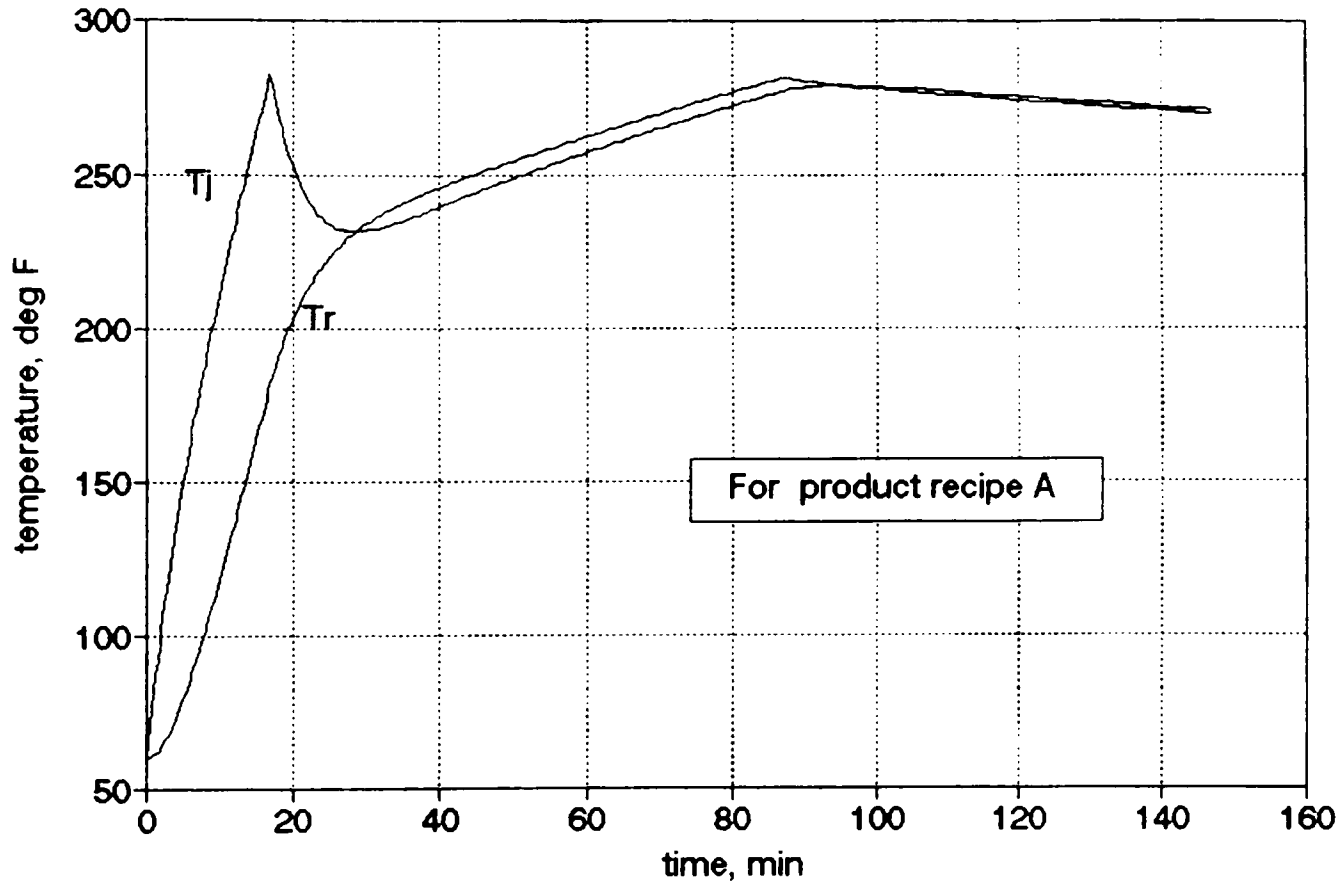


Figure A.2
Temperature versus Time
(Uncontrolled Response)

reactor temperature reaches its 180°F set point when the jacket temperature is just above 280 °F. The reactor temperature keeps increasing until monomer feed stops. Even without an external heat supply, the reactor temperature increases by about 100 degrees by the heat of polymerization. Figure A.3 shows monomer conversion versus time. The fast conversion also causes the fast viscosity increase shown in Figure A.4. At the end of the monomer feed period (time point at 85 minutes), the viscosity has been increased 20 times over its value at the initial feed stage. Figure A.5 shows the opposite behavior of the overall heat transfer coefficient. Since the overall heat transfer coefficient directly affects the amount of heat transferred from reactor to jacket, its behavior simply indicates the nonlinearity of the process gain. A linear change of effective heat transfer area is shown in Figure A.6.

A.5 Controller

The model adjustment principle is described in Chapter V. Looking back the process dynamic equations, Eqs.(A.4) to (A.9), there are several possible adjustable parameters as shown in Table A.5. The number of adjustable parameters which will be used for model adaptation depends on the number of process measurements. Since T_r and T_j are the two measurable data which represents the process conditions, two adjustable parameters should be selected and two equations

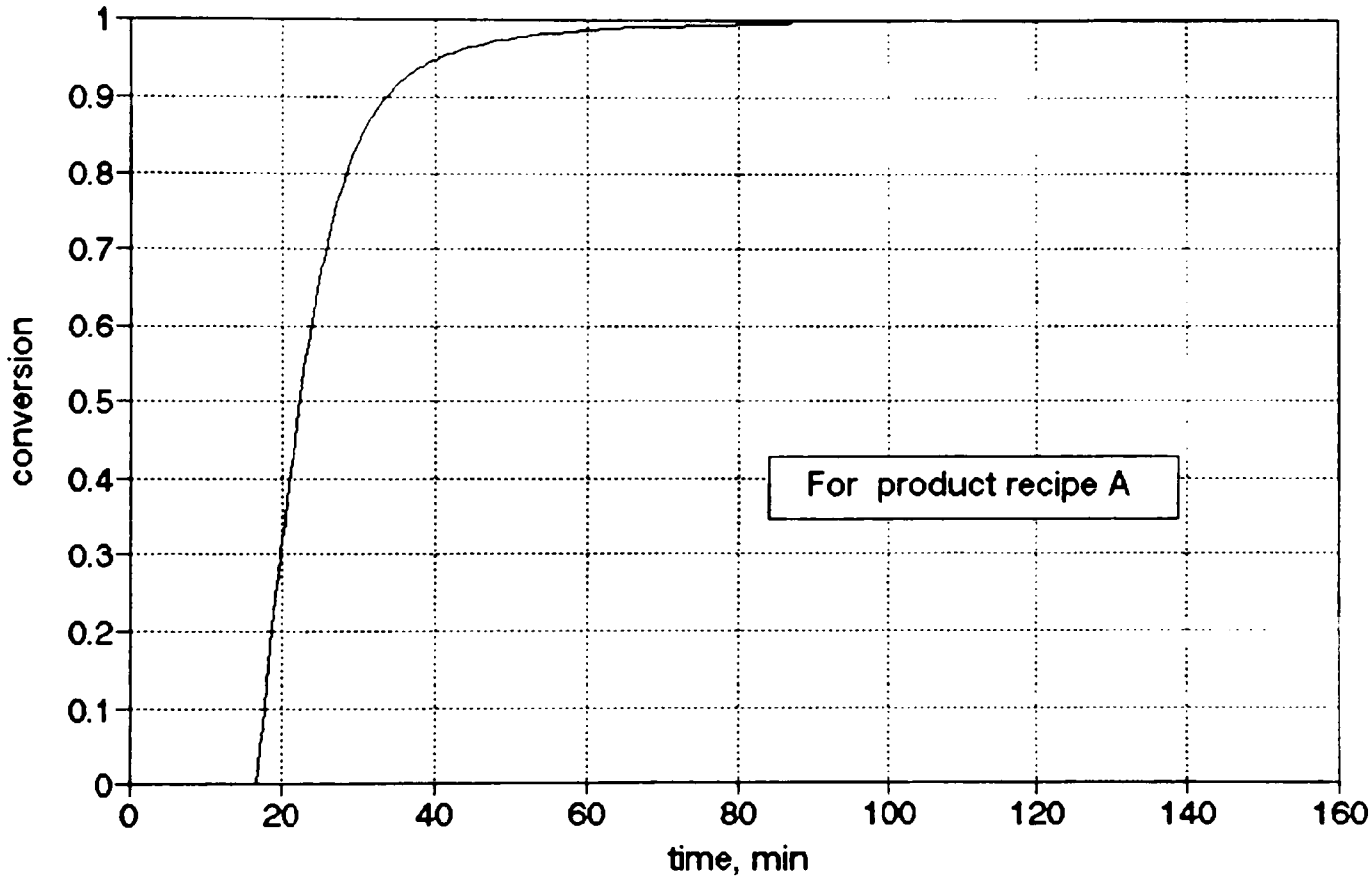


Figure A.3
Conversion versus Time
(Uncontrolled Response)

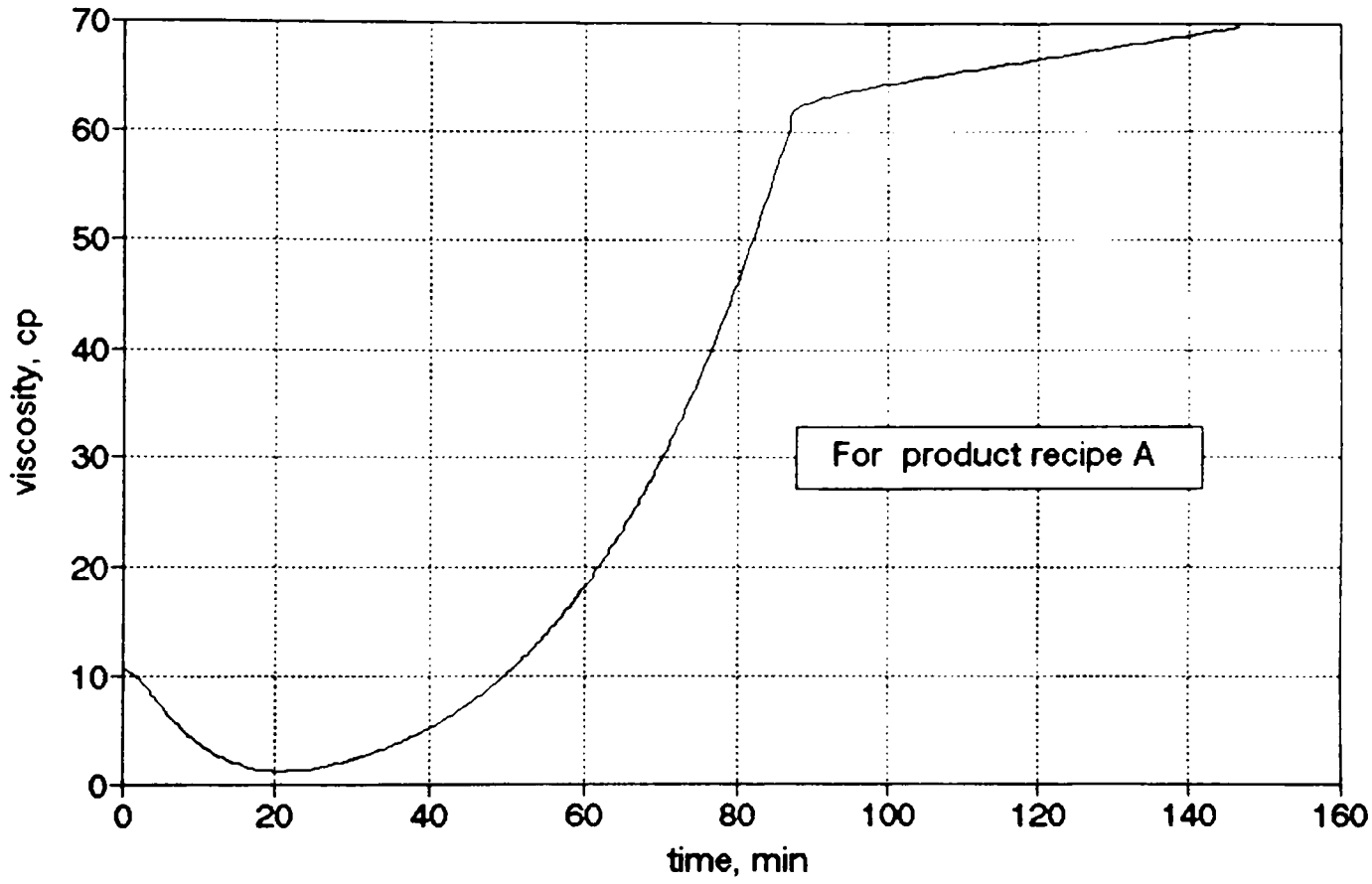


Figure A.4
Viscosity versus Time
(Uncontrolled Response)

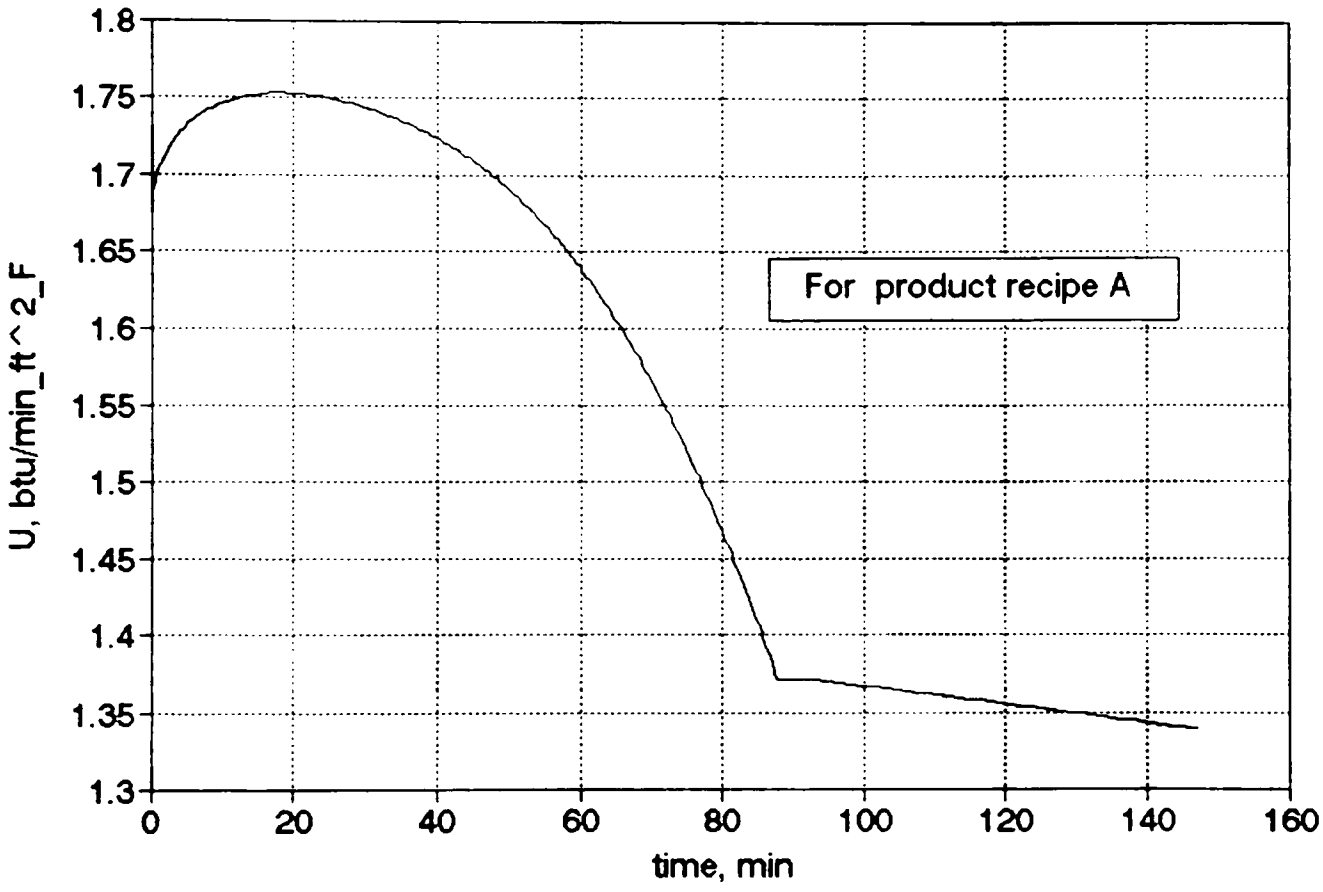


Figure A.5
Overall Heat Transfer Coefficient versus Time
(Uncontrolled Resposne)

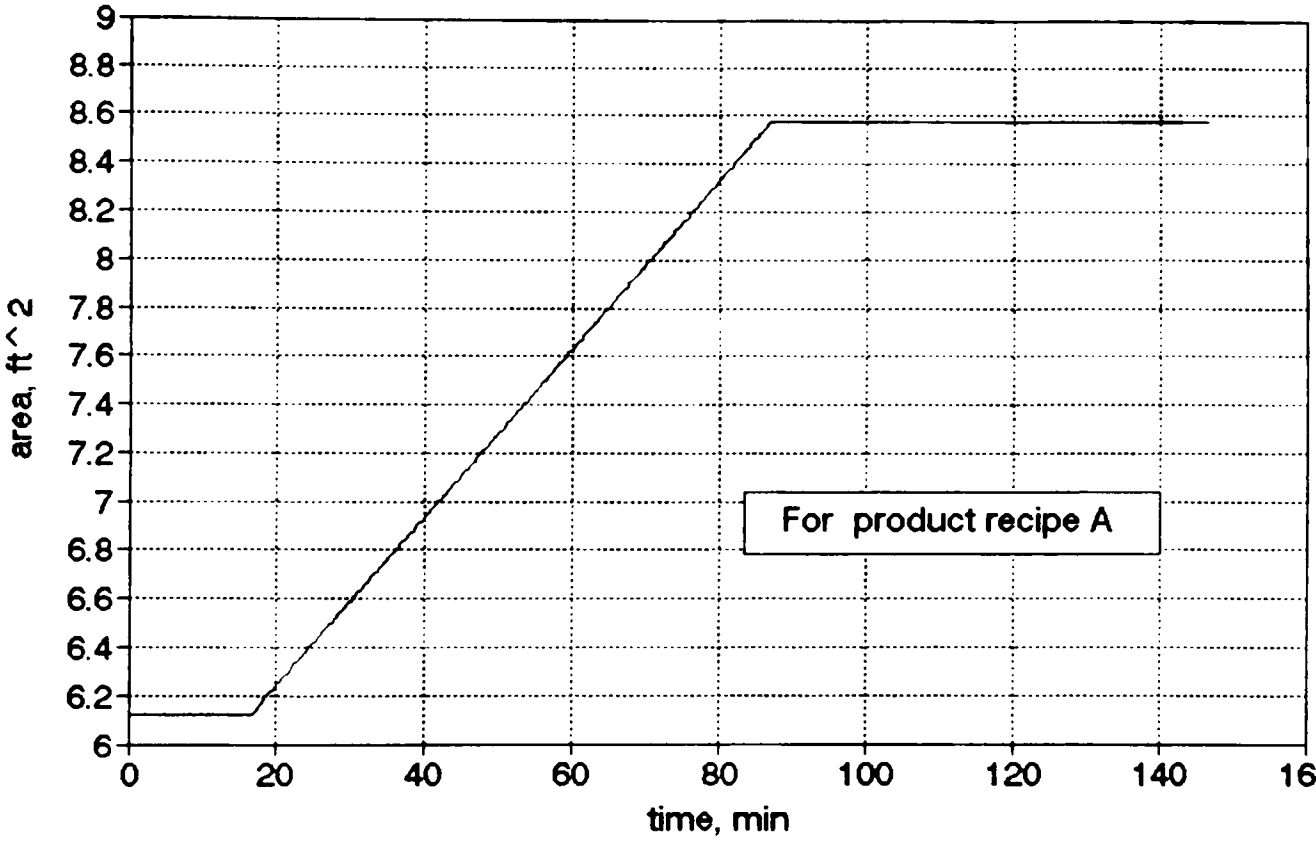


Figure A.6
Effective Heat Transfer Area versus Time

Table A.5
List of Possible Adjustable Parameters

Symbol	Description
k_0	Preexponential term of reaction rate constant
k	Reaction rate constant
f_d	Fouling factor
U	Overall heat transfer coefficient
EH_{Loss}	Environmental heat loss
n_M	Moles of monomer remained in reactor

(reactor side energy balance equation & jacket side energy balance equation) should be used for reparameterization. As a base case two adjustable parameters, k and U, are chosen even though there are other choices of sets of adjustable parameters. These include k_o and f_d, EH_{loss} and f_d, n_m and U. The approximate control model which is equivalent to Eq.(5.2) can be described as follows (note the bold-faced letters for adjustable parameters):

$$\frac{d\bar{n}_M}{dt} = \frac{\dot{m}_M}{MW_M} - \boldsymbol{k}\bar{n}_M. \tag{A.19}$$

$$\frac{d(\sum \bar{m}_l Cp_l \bar{T}_r)}{dt} = \dot{m}_M Cp_M (T_{amb} - \bar{T}_r) + \boldsymbol{k}\bar{n}_M(-\Delta H_p) - \boldsymbol{U}A(\bar{T}_r - \bar{T}_j). \tag{A.20}$$

$$(V_j \rho_{w,j} Cp_{w,j}) \frac{d\bar{T}_j}{dt} = \dot{v}_w \rho_w C_{p,w}(T_w - \bar{T}_j) + \dot{m}_s(\lambda_s + C_{p,w}(T^o - T_j))$$

$$- \dot{m}_s C_{p,w} T_j + \boldsymbol{U}A(\bar{T}_r - \bar{T}_j). \tag{A.21}$$

One can easily notice the mismatch between the process dynamic model and this control model. For instance, k in Eq.(A.19) replaces the reaction rate constant term, $ik_o\exp(-E_a/RT)\mu^{0.4}$, in the process mass balance (Eq.(A.4)), and U in Eqs.(A.20) and (A.21) replaces the overall heat transfer coefficient, $1/(1/h_i+1/h_f)$, in the process energy balance (Eqs.(A.8) and (A.9)). Furthermore the reactor side energy balance, Eq.(A.20) does not retain the environmental heat loss term.

Eqs.(A.19), (A.20) and (A.21) shall be used as a temperature predictor by which the controller obtains the difference between the measured process output T_r and T_j and predicted value $\bar{T}_r$ and $\bar{T}_j$ as described in Eq.(5.3). Note that these prediction equations contain adjustable parameters which are continuously updated every reparameterization step. This set of ODEs are solved by the 4th order Runge-Kutta method between the previous and current reparameterization steps. The step size is adjusted by the same p-value method as used for process dynamics. The reparameterization objective is to find k and U such that $\epsilon_{t,r}$ and $\epsilon_{t,j}$ become zero. Restating $\bar{T}_r$ and $\bar{T}_j$ as discretized form as in Eq.(5.5) one can get two objective functions shown below:

$$\epsilon_{t,r} = T_{r,t} - \bar{T}_{r,t} = T_{r,t} - T_{r,t-\Delta t} - \Delta t\,[\dot{m}_M Cp_M (T_{amb} - T_{r,t-\Delta t}) + k \bar{n}_{M,t-\Delta t}(-\Delta H_p) - U\bar{A}_{t-\Delta t}(T_{r,t-\Delta t} - T_{j,t-\Delta t})]. \quad (A.22)$$

$$\epsilon_{t,j} = T_{j,t} - \bar{T}_{j,t} = T_{j,t} - T_{j,t-\Delta t} - \Delta t\,[\dot{v}_w \rho_w C_{p,w}(T_w - T_{j,t-\Delta t}) + \dot{m}_s(\lambda_s + C_{p,w}(T^o - T_j)) - \dot{m}_s C_{p,w} T_j + U\bar{A}_{t-\Delta t}(T_{r,t-\Delta t} - T_{j,t-\Delta t})]. \quad (A.23)$$

where Δt = reparameterization interval.

Since we have two equations and two unknowns, Newton's method can solve them. The roots of equations are obtained

by:

$$\begin{bmatrix} \frac{\partial e_{r,t}}{\partial k} & \frac{\partial e_{r,t}}{\partial U} \\ \frac{\partial e_{j,t}}{\partial k} & \frac{\partial e_{j,t}}{\partial U} \end{bmatrix} \begin{bmatrix} \Delta k \\ \Delta U \end{bmatrix} = - \begin{bmatrix} e_{r,t} \\ e_{j,t} \end{bmatrix}. \tag{A.24}$$

If one goes through the steps in Eq.(5.6), it yields:

$$\Delta t \begin{bmatrix} \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial k} & \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial U} \\ \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial k} & \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial U} \end{bmatrix} \begin{bmatrix} \Delta k \\ \Delta U \end{bmatrix} = \begin{bmatrix} \epsilon_{r,t} \\ \epsilon_{j,t} \end{bmatrix}. \tag{A.25}$$

Then, for a single step adjustment to k and U incorporating with relaxation coefficient α:

$$\Delta k = \frac{\alpha \left(e_{r,t} \cdot \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial U} - e_{j,t} \cdot \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial U} \right)}{\frac{\partial \overline{f}_{r,t-\Delta t}}{\partial k} \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial U} - \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial U} \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial k}}. \tag{A.26}$$

$$\Delta k = k_t - k_{t-\Delta t}. \tag{A.27}$$

$$\Delta U = \frac{\alpha \left(e_{j,t} \cdot \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial k} - e_{r,t} \cdot \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial k} \right)}{\frac{\partial \overline{f}_{r,t-\Delta t}}{\partial k} \cdot \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial U} - \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial U} \cdot \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial k}}. \tag{A.28}$$

$$\Delta U = U_t - U_{t-\Delta t}. \tag{A.29}$$

The partial derivatives in Eqs.(A.26) and (A.29) are defined as follows:

$$\frac{\partial \overline{f}_{r,t-\Delta t}}{\partial k} = \frac{\overline{n}_{M,t-\Delta t}(-\Delta H_p)\Delta t}{\sum(\overline{m}_{l,t-\Delta t} Cp_l)} \tag{A.30}$$

$$\frac{\partial \bar{T}_{r,t-\Delta t}}{\partial U} = \frac{-\bar{A}_{t-\Delta t}(T_{r,t-\Delta t}-T_{j,t-\Delta t})\Delta t}{\sum(\bar{m}_{i,t-\Delta t}Cp_i)}. \qquad (A.31)$$

$$\frac{\partial \bar{T}_{j,t-\Delta t}}{\partial k} = 0. \qquad (A.32)$$

$$\frac{\partial \bar{T}_{j,t-\Delta t}}{\partial U} = \frac{\bar{A}_{t-\Delta t}(T_{r,t-\Delta t}-T_{j,t-\Delta t})\Delta t}{V_j \rho_{w,j} Cp_{w,j}}. \qquad (A.33)$$

If one chooses a different set of adjustable parameters, the functional form of Eqs.(A.22) and (A.23) will be slightly different. However, the reparameterization procedure still remains the same.

NPMBC control action and offset compensation by GMC law are implemented as expressed in Chapter V.

A.6 Simulation Results

Control performance is compared between conventional PID and NPMBC control. However, for NPMBC control, simulations are also carried out with a range of controller models and process and control conditions.

The original problem statement challenges control of two recipes. However, our results only consider product recipe A, because an inherent problem in the process for recipe B makes end-of-run control impossible. At the beginning of the second recipe feed B (110 minute) the reactor viscosity becomes so high that the overall heat transfer coefficient effectively becomes zero. These

phenomena are illustrated in Figures A.7 and A.8. Accordingly, any control action fails to remove the heat accumulating in the reactor and the reactor temperature gets higher (maximum 240 °F). Temperature trends shown in Figure A.9 explain that phenomena. Please note, however, that until that point, reactor temperature is controlled.

A.6.1 PID results

Conventional PID control is implemented in both the master and slave controllers. An anti-windup technique, "internal reset feedback," is used to reject the master reset windup. Each of Figures A.10 and A.11 shows master and slave controller output, respectively. Final control elements are actuated by splitting the range of slave controller output. The dump valve reversely acts at 0 to 50% of the slave controller output while the steam valve acts at 50 to 100%. Tuning parameters used are:

Master controller: K_c=3.3 τ_i=2.5 τ_d=0.4,

Slave controllers: K_c=2.6 τ_i=3.5 τ_d=0.3,

Just after the monomer feed stops (85 minutes), reactor temperature dropped about 4 °F(Figure A.12). The maximum absolute value of the error is observed to be 23.64°F, and the integral performance measures are:

IE=1627.128 °F min,

IAE=3781.334 °F min,

ISE=43404.851 °F² min.

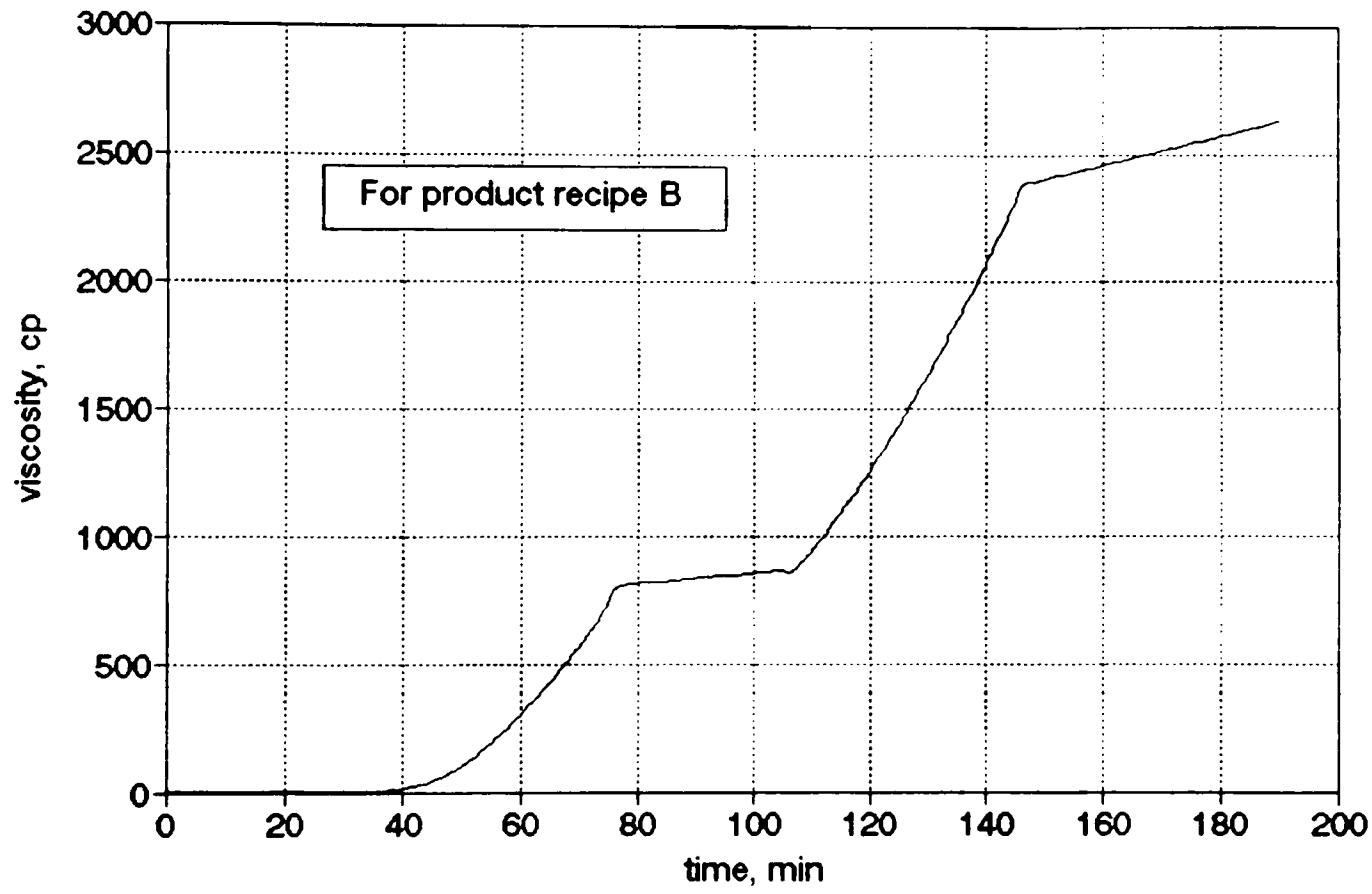


Figure A.7
Viscosity versus Time
(Controlled Response)

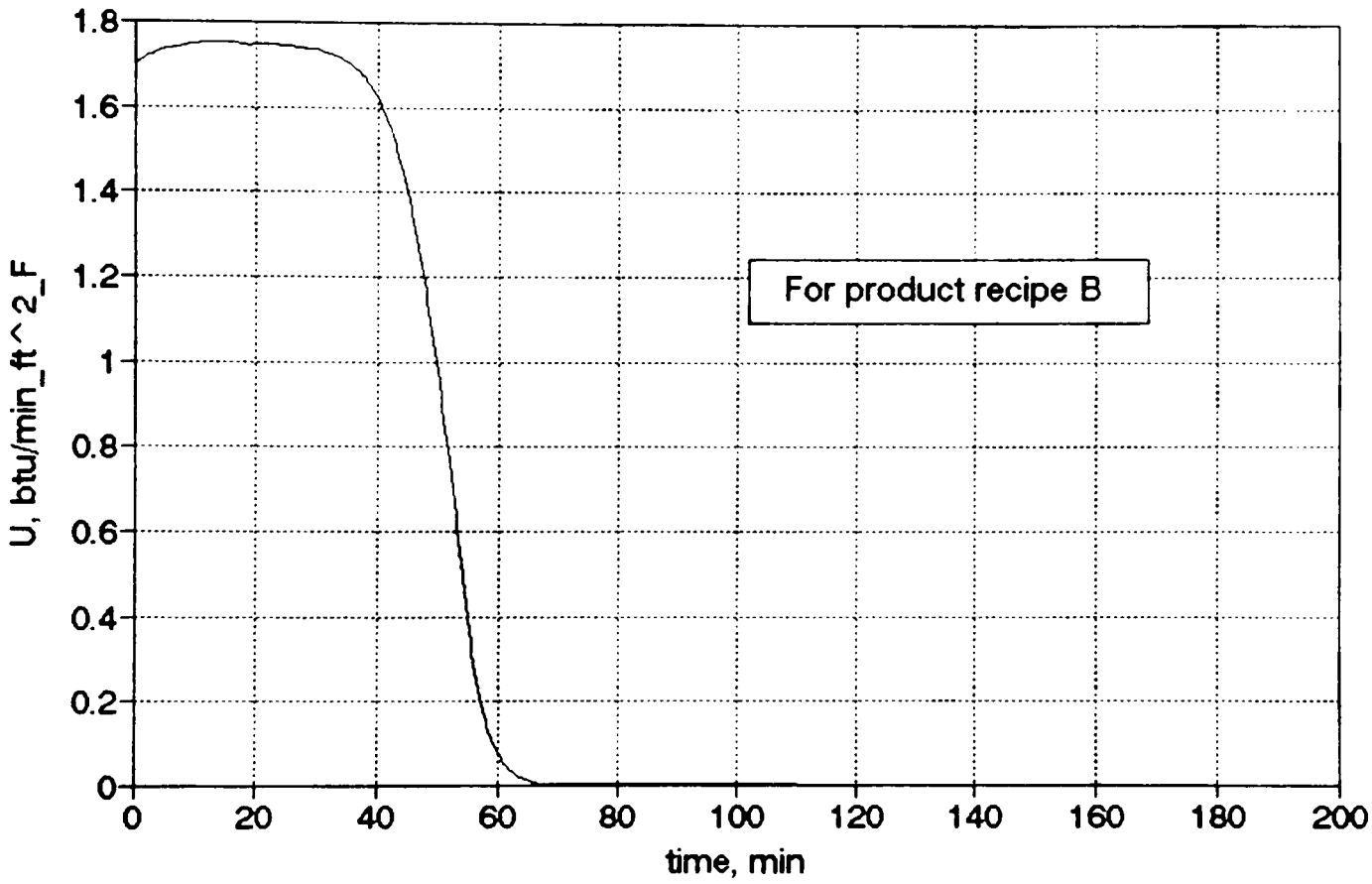


Figure A.8
Overall Heat Transfer Coefficient versus Time
(Controlled Response)

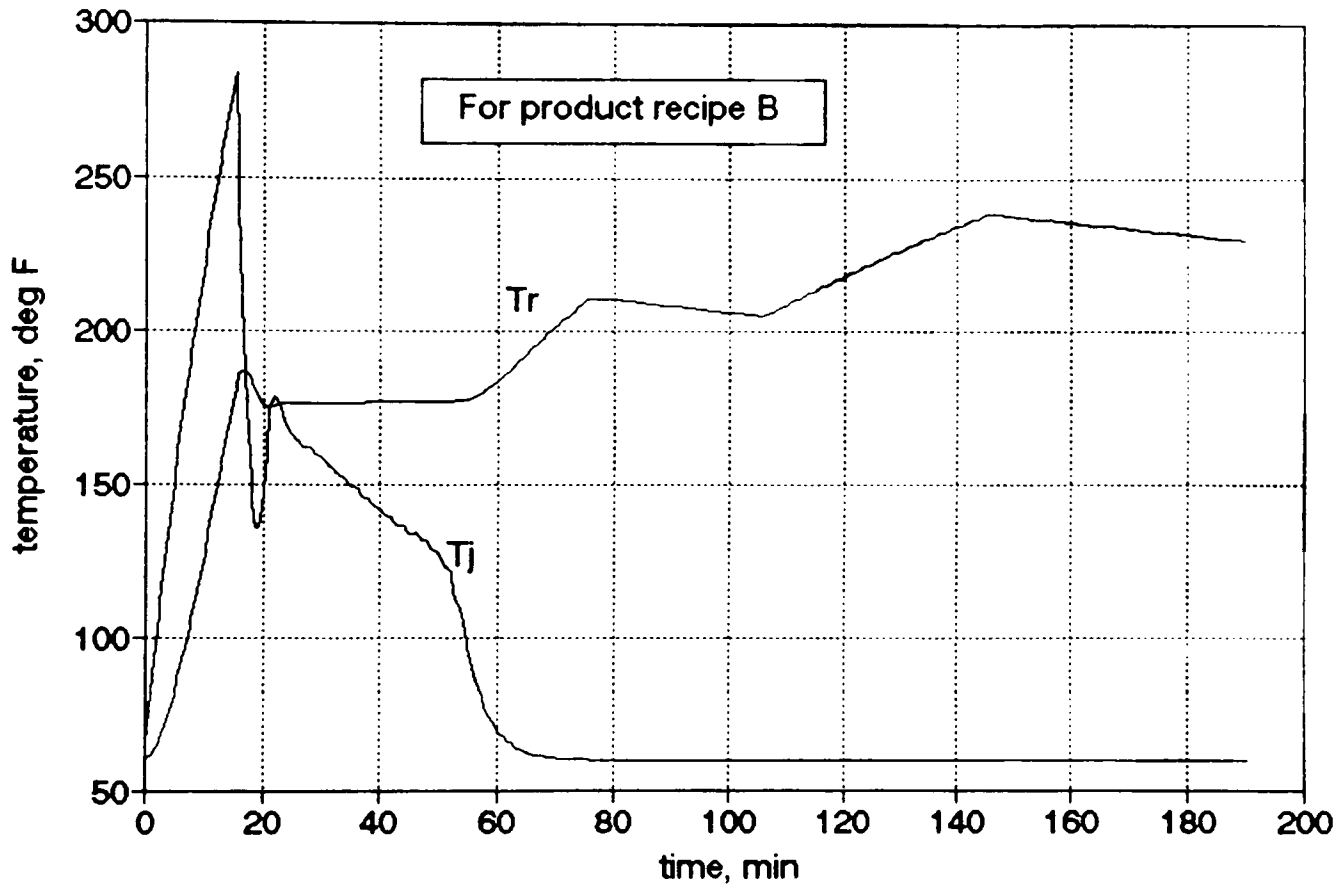


Figure A.9
Temperature versus Time
(NPMBC Control Response)

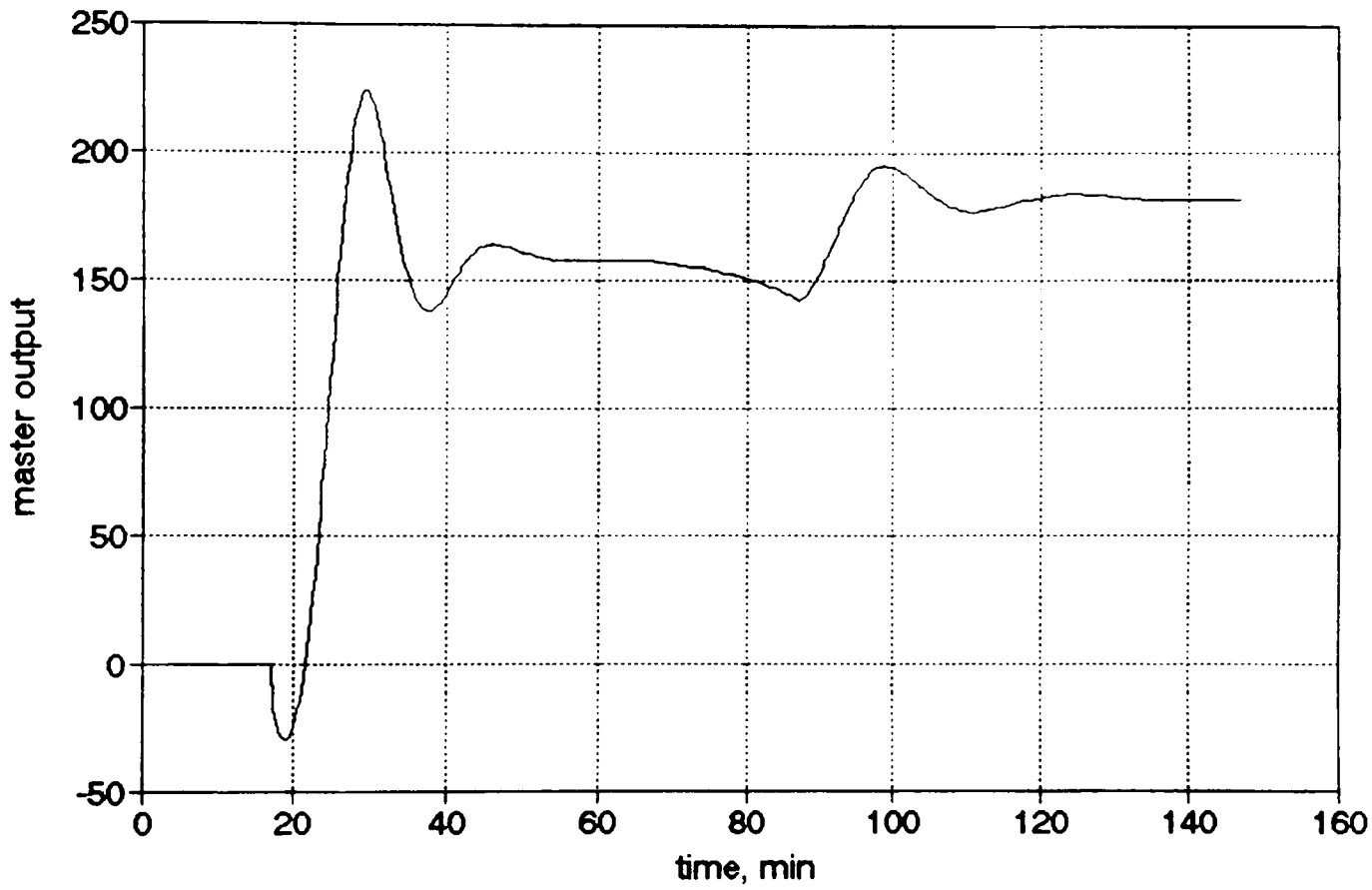


Figure A.10
Master Controller Output versus Time
(Cascade PID Control)

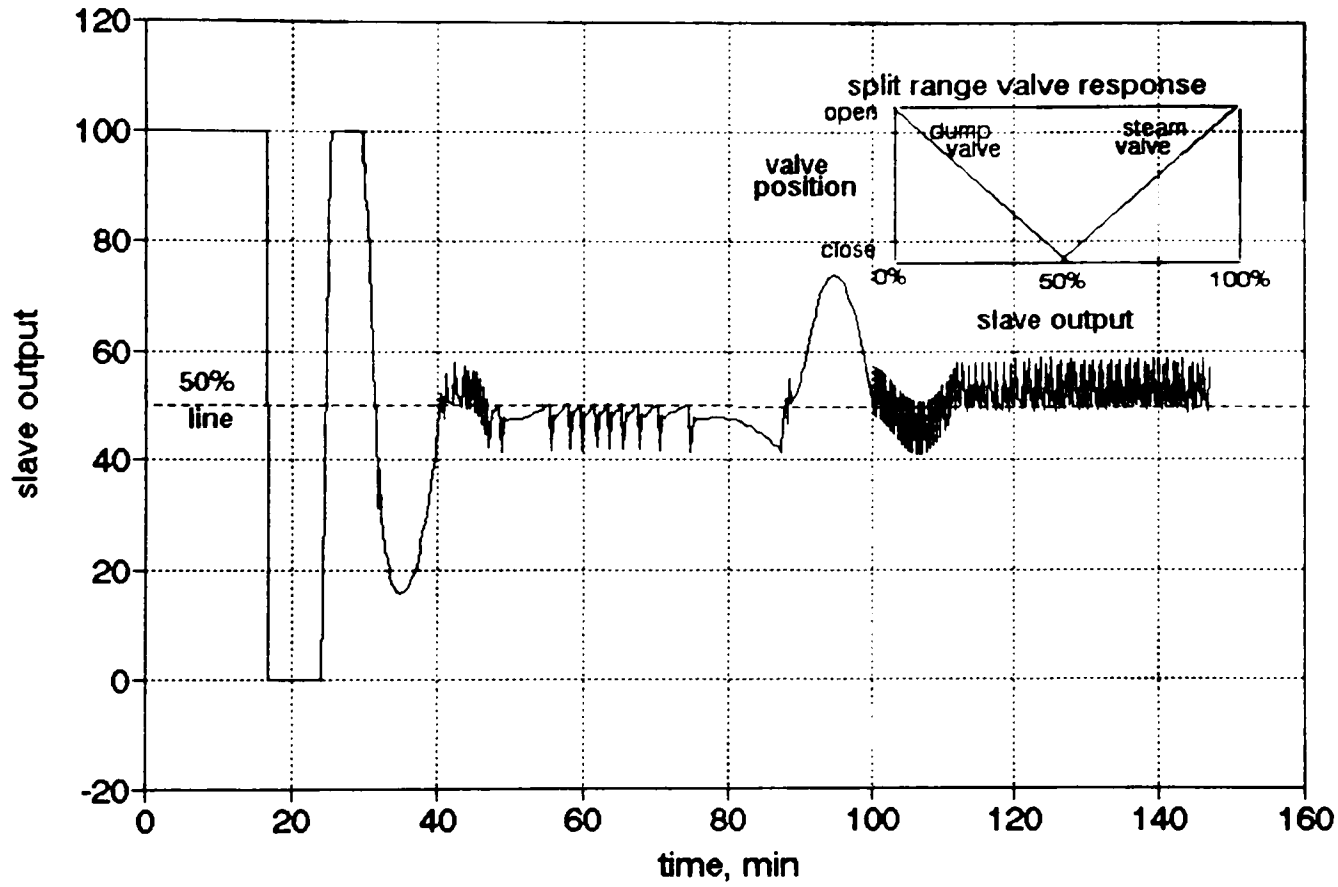


Figure A.11
Slave Controller Output versus Time
(Cascade PID Control)

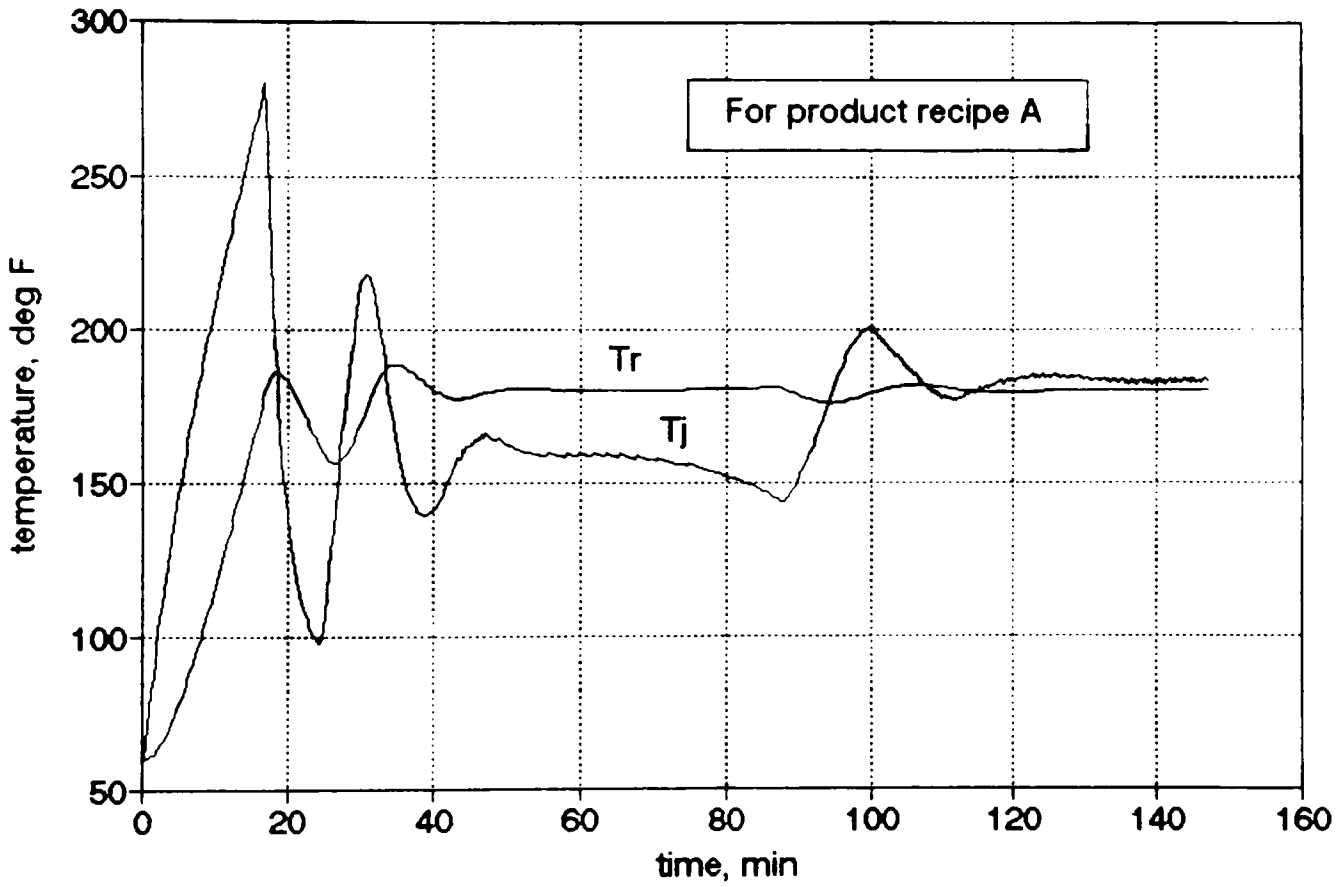


Figure A.12
Temperature versus Time
(Cascade PID-Split Range Control Response)

Conventional control gives some undersirable results. Early in the run the slave controller output does not oscillate, but as conditions change the controller gain becomes too high. Note the valve oscillations at the mid run time. Further note that the steam and cold water both are active at the 100 minute period.

A.6.2 Base Case NPMBC Results

There are many controller and process factors which could affect the controlled system performance. These are listed below along with the base case values. The base case provides a reference for evaluating how controlled system response differs when the factors are changed.

The base case is:

- Controller part:

  adjustable parameter set 1 (k and U)

  sampling interval = 0.25 min

  reparameterization interval = 2.0 min

  tuning parameter K1 = 2.0

  tuning parameter K2 = 0.1

  relaxation coeff. for parameter tracking = 1.0

  $C_{p,monomer}$=0.45 (112.5 % of actual value)

  $C_{p,water}$ =1.05 (105 % of actual value)

  $C_{p,solid}$ =0.8 (107 % of actual value).

- Process part:

  impurity factor of reacting mixture = 1.2

  reactor side fouling factor = 0.0025 hr_ft^2_F/btu

  ambient temperature = 45 °F

  water supply temperature = 60 °F

  initial reactor temperature = 60 °F

  initial jacket temperature = 60 °F

  other process conditions are same as Tables A.1 and 2.

At the start of a run, the jacket receives the maximum steam rate (2.1577 lb/min) until the reactor temperature reaches its set point of 180°F. At the next sampling period, monomer feed starts, the steam valve is closed, and NPMBC control. From this point on automatic, control first calculates the jacket temperature set point and from this either the steam inlet rate or cold water dump rate. The steam or water rate are cascaded to instantaneously acting PI flow controllers. Start-up parameter values are k = 0.1min^{-1} and U = 2.0btu/min_ft^2_F.

Figures A.13 and A.14 illustrate the base case parameter tracking. When the amount of monomer in the reactor mixture is small, the heat of reaction is negligible when compared to other terms in the reactor energy balance. Because of process/model mismatch the reaction rate can actually be parameterized as a negative value. This happens

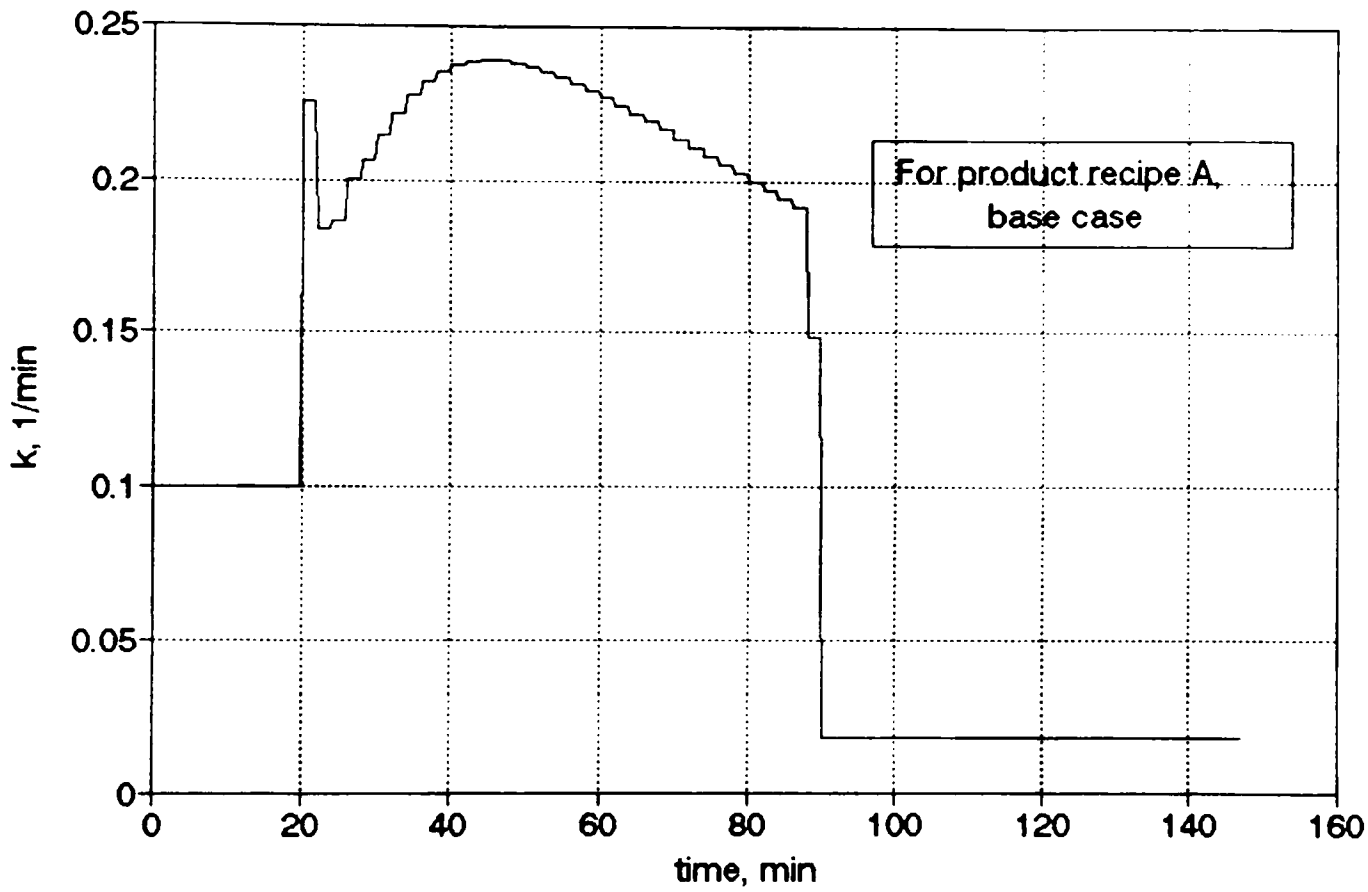


Figure A.13
Adjustable Parameter k versus Time

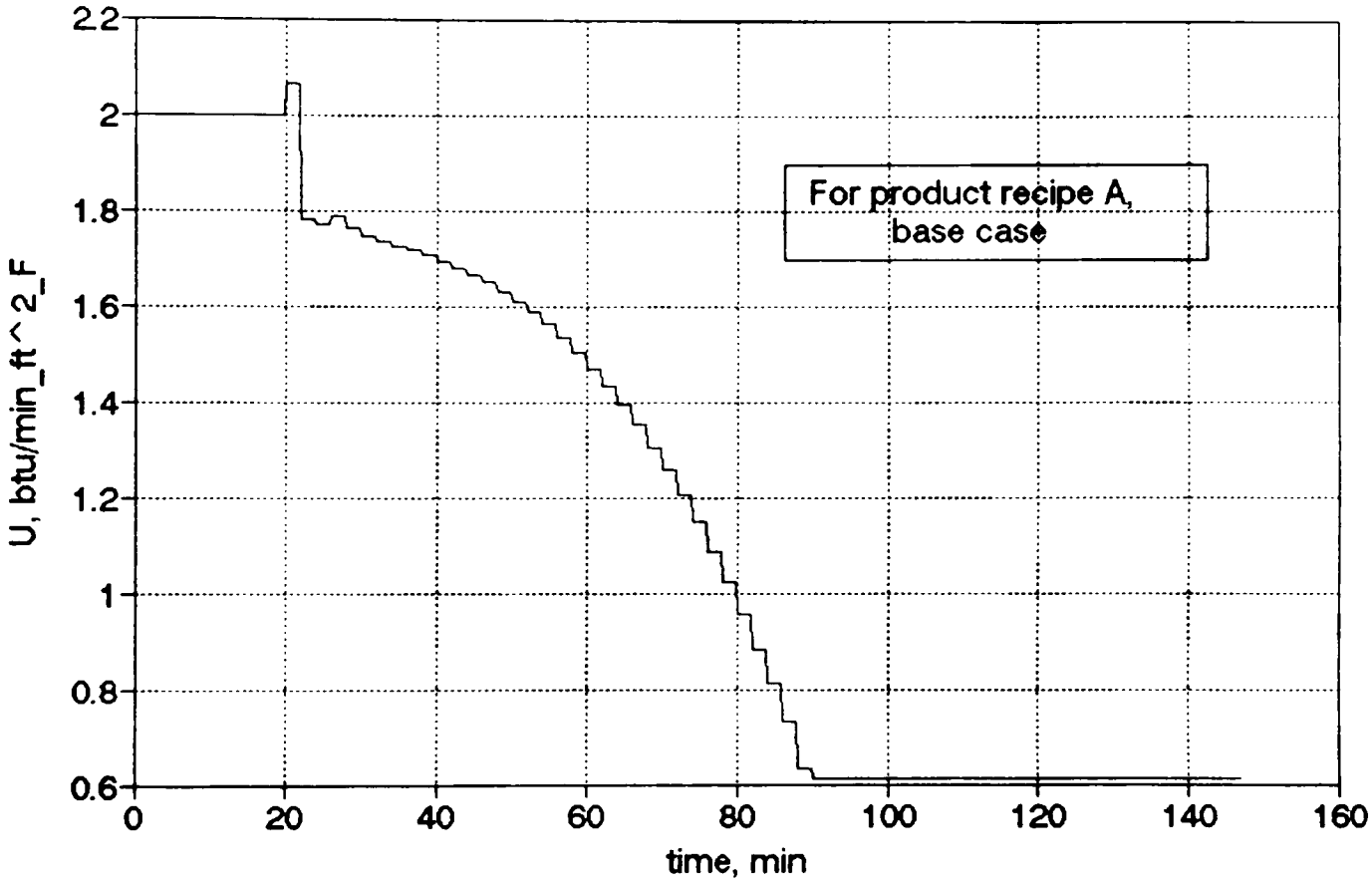


Figure A.14
Adjustable Parameter U versus Time

both at the beginning and end of the reactor campaign. When this ocurrs the controller model keeps the previous values of k and U. Figure A.13 indicates that phenomena and also shows parameter tracking during the bulk of the run. Parameter adjustment is the main feedback mechanism and also helps adjust the model to account for process gain changes. Any offset not removed by parameter adjustment will be removed by the integral term of the GMC law.

Figure A.15 illustrates the jacket temperature set point which is obtained from the reactor control calculation. In response to this jacket temperature target, the jacket control calculation determines the manipulated variable value. The dump flow and steam supply are shown in Figures A.16 and A.17.

The control performance, the reactor temperature trend, is shown in Figure A.18. The temperature meets the control objective, it deviates from the setpoint less than only 1 °F (except for the first five minutes of start up). The time-integral performance measures are as follows:

IE = -58.664 °F min

IAE = 75.154 °F min

ISE = 290.497 °F min

Figure A.19 demonstrates a visual comparison between PID and NPMBC response. NPMBC's superiority over the conventional PID's is clearly depicted on that graph.

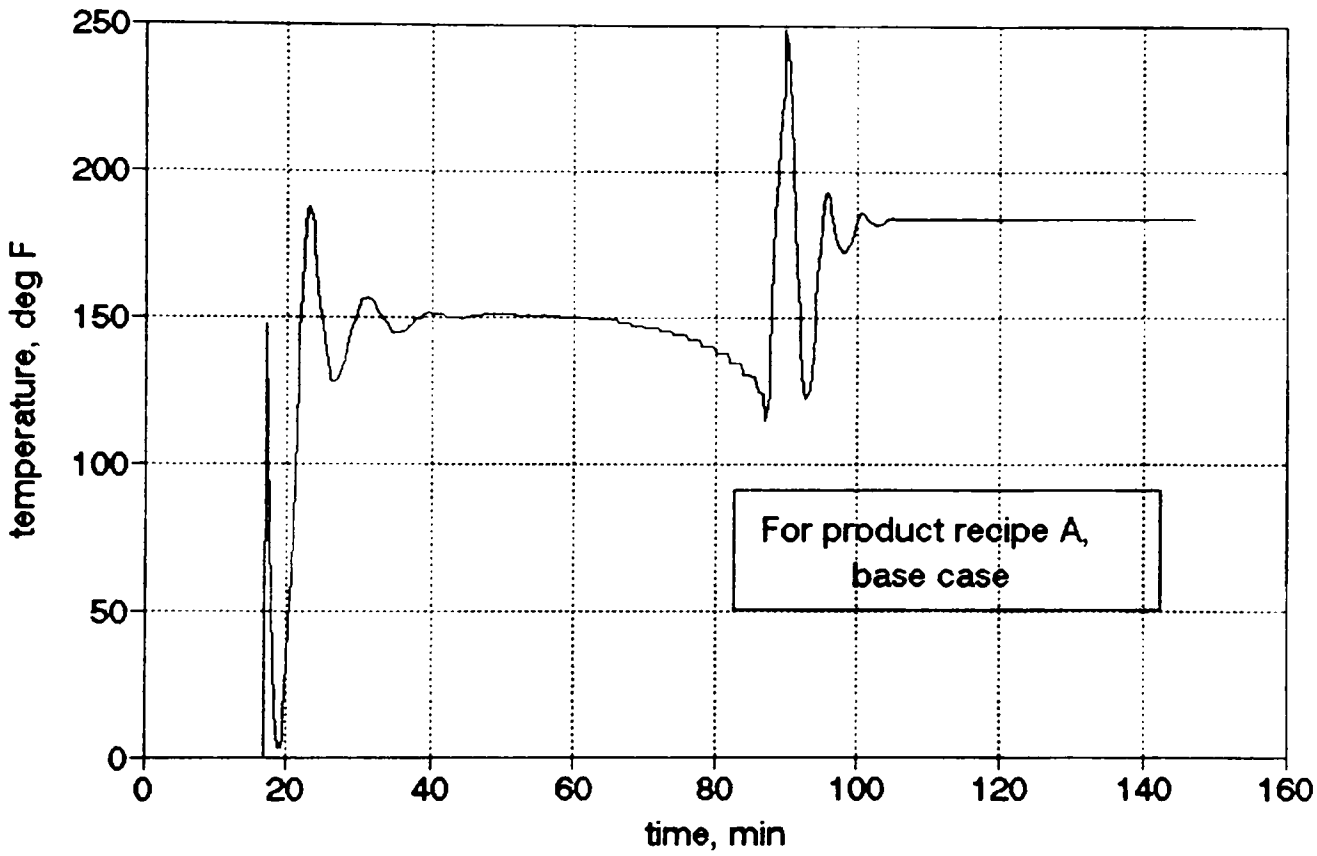


Figure A.15
Control Action Calculation:
Jacket Temperature Setpoint versus Time

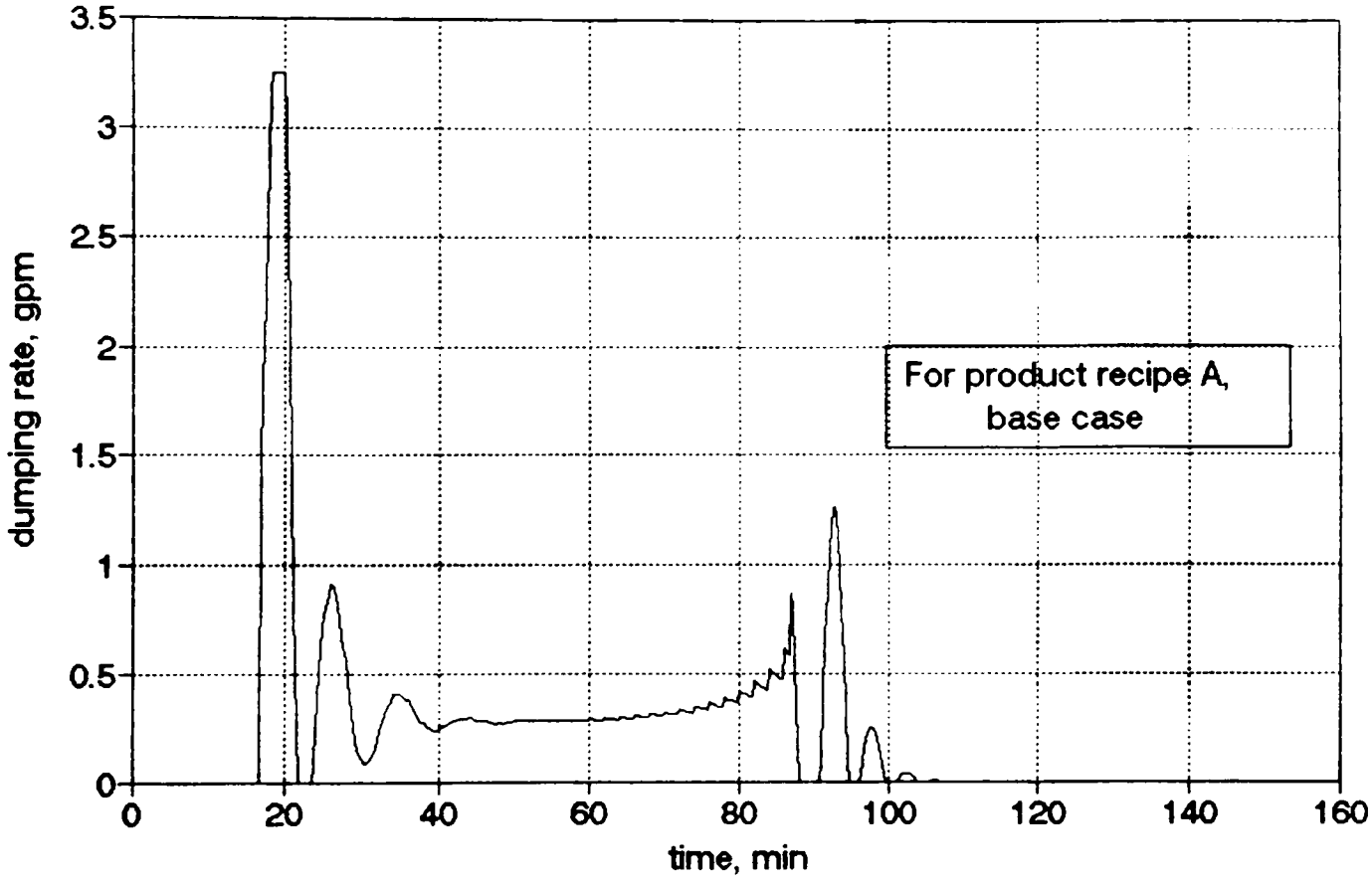


Figure A.16
Control Action Calculation:
Required Cooling Water Flow Rate versus Time

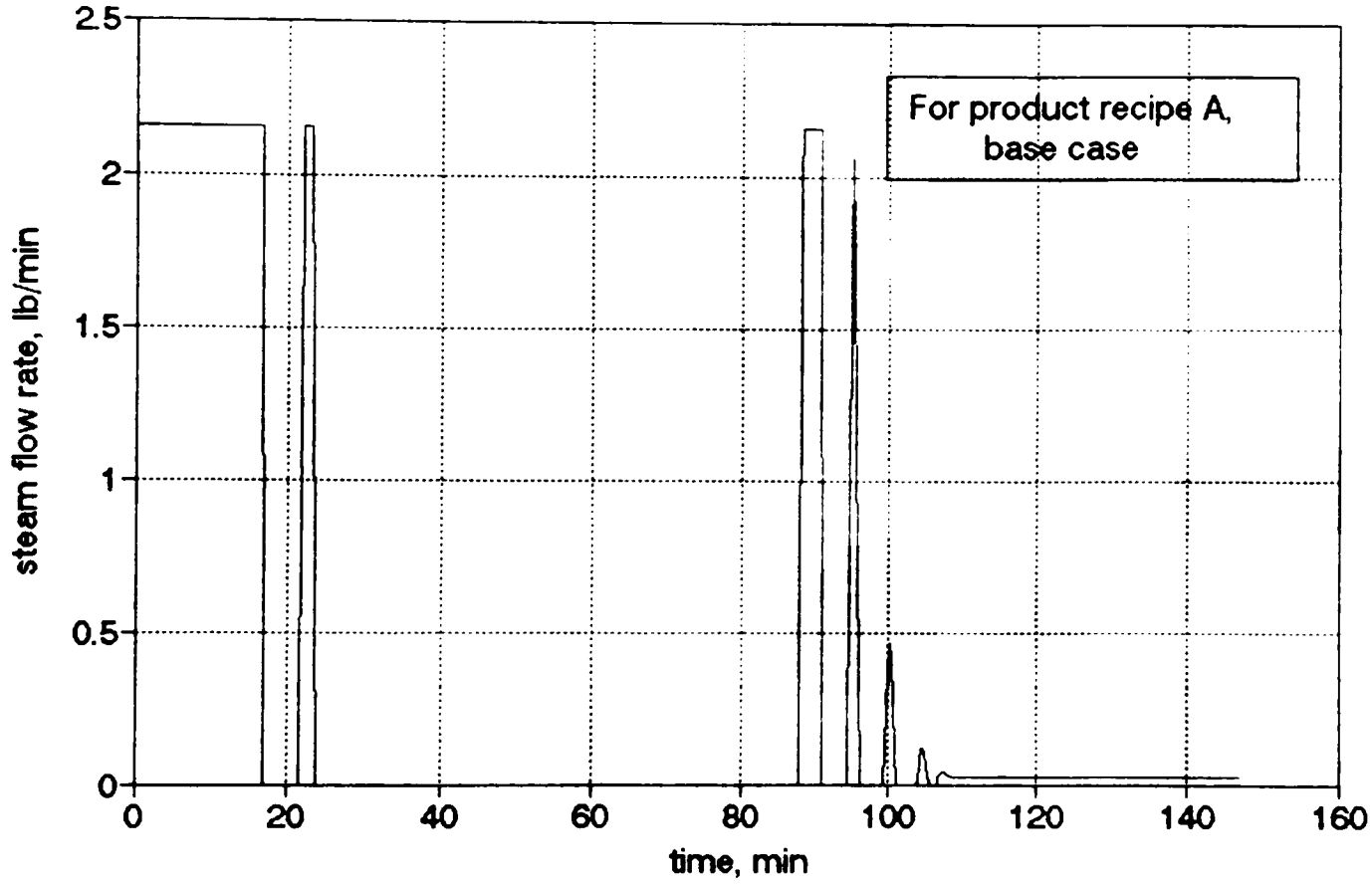


Figure A.17
Control Action Calculation:
Required Steam Flow Rate versus Time

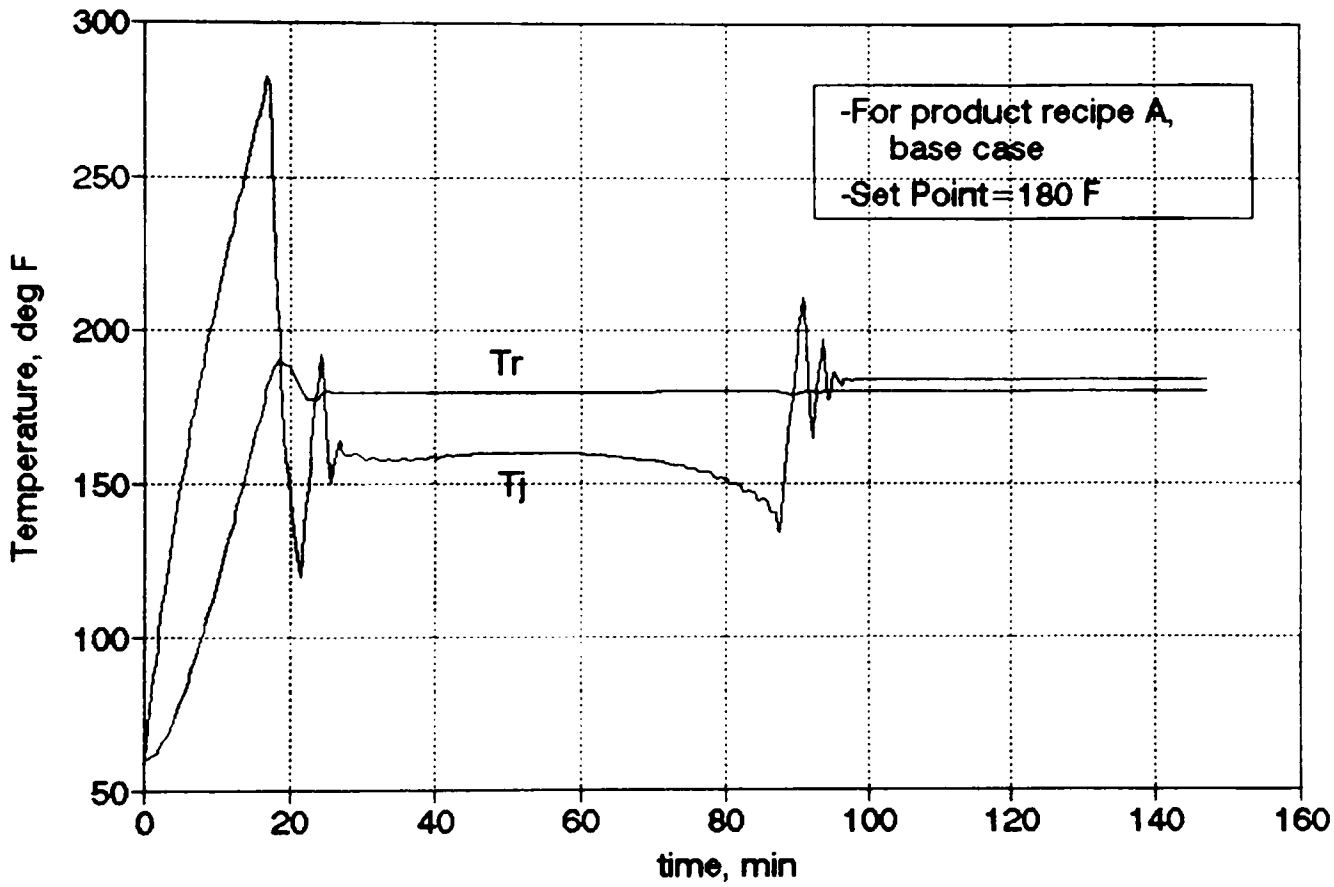


Figure A.18
Temperature versus Time (NPMBC Response)

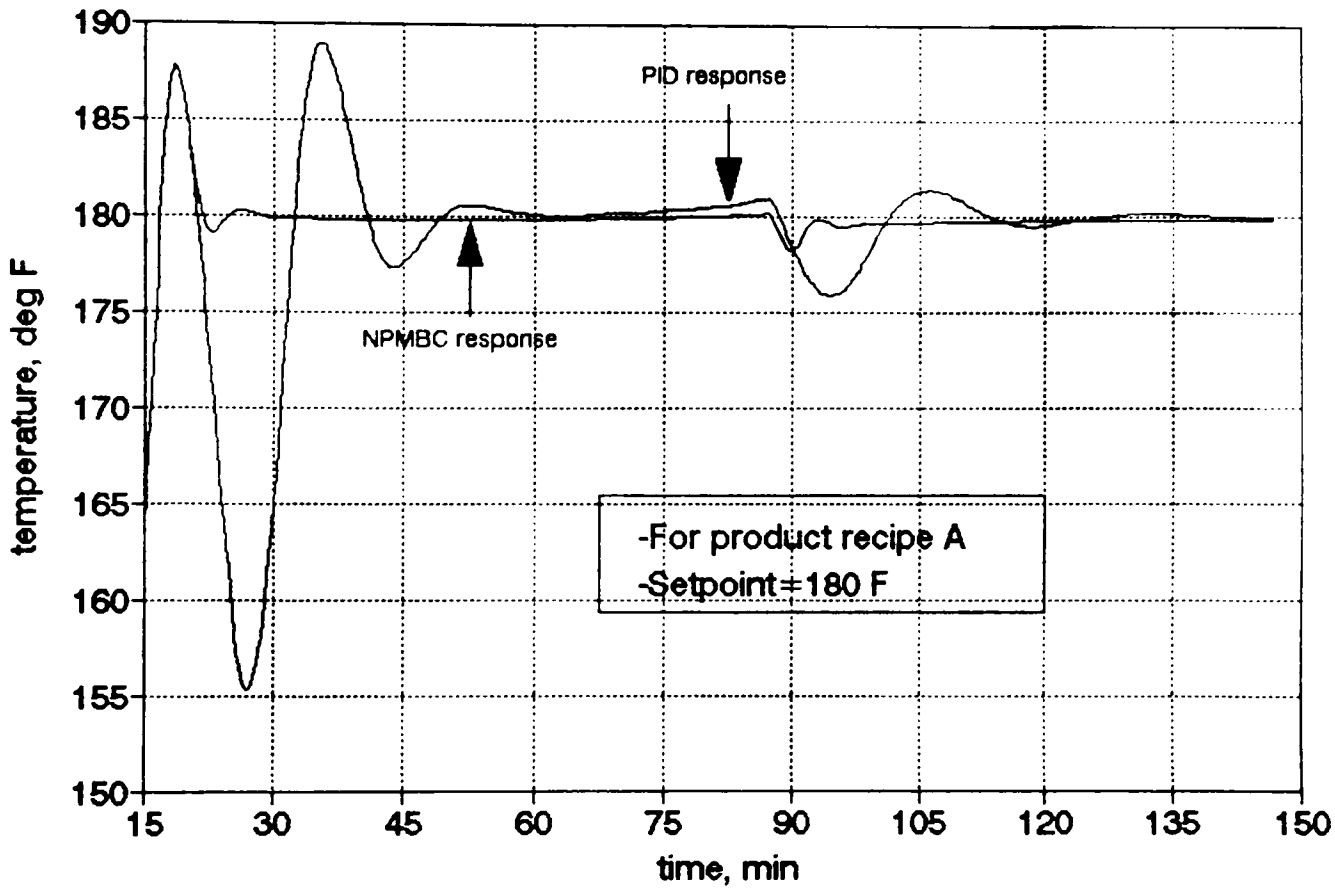


Figure A.19
Comparison Between Cascade PID and NPMBC Response
on Reactor Temperature

A.6.3 Simulation results for other cases

Simulation results for other values of controller model and process parameters and conditions are now compared to the base case in terms of time-integral performance (TIP) measures. The results are presented in Tables A.6 to A.10. Those five tables contain a lot of data, but the general observations and conclusions are summarized below:

The major NPMBC controller user option is the choice of adjustable parameters. We chose to evaluate four sets. The base case controller used k and u, the other three sets are k_o & f_d, EH_{loss} & f_d and n_M & U. For adjustable parameter set 2, k_o & f_d, the control model uses:

$$k = k_0 \exp(-7800/(T+460))\mu^{0.0},$$

$$h_i = 135.0\exp(-6.0e\text{-}3\mu_w).$$

$C_{p,i}$ is same as the base case.

For adjustable parameter set 3, EH_{loss} & f_d, the control model uses:

$$k = k_0 \exp(-6400/(T+460))\mu^{0.4},$$

$$h_i = 135.0\exp(-6.0e\text{-}3\mu_w).$$

$C_{p,i}$ is same as the base case.

For adjustable parameter set 4, n_M & U, the control model uses:

Table A.6
Time Integral Performance 1
(* : indicate the base case)

Product Recipe A Repara. Step=2.0, Sampling Step=0.25 Process conditions are the same as the base case				
	Set 1 k & U	Set2 k_0 & f_d	Set 3 EH_{loss} & f_d	Set4 m_M & U
K_1=1.0	K_2=0.1			
MaxAE	17.377	16.824	16.611	17.472
IE	-205.102	-189.145	-182.477	-141.982
IAE	229.772	213.761	195.065	261.527
ISE	1773.278	1542.319	1437.609	1754.715
K_1=1.0	K_2=1.0			
MaxAE	16.912	16.387	16.183	17.003
IE	-195.572	-144.039	-55.526	-155.129
IAE	220.676	172.735	223.733	226.253
ISE	1636.751	1228.417	1519.192	1569.522
K_1=2.0	K_2=0.1			
MaxAE	10.111*	9.807	9.760	10.132
IE	-58.664*	-53.952	-51.053	-29.899
IAE	75.154*	74.358	93.778	93.432
ISE	290.497*	274.733	288.572	302.366
K_1=2.0	K_2=1.0			
MaxAE	10.069	9.773	9.732	10.088
IE	-56.776	-47.009	-47.782	-36.527
IAE	74.825	75.841	100.684	85.365
ISE	287.745	287.662	300.976	293.100
K_1=3.0	K_2=0.1			
MaxAE	8.147	8.077	8.078	8.145
IE	-34.152	-30.704	-20.908	-15.504
IAE	99.902	49.526	75.701	63.625
ISE	165.381	165.138	189.754	172.592
K_1=3.0	K_2=1.0			
MaxAE	8.147	8.077	8.078	8.145
IE	-33.341	-29.347	-25.419	-19.095
IAE	50.051	49.673	73.546	59.332
ISE	165.553	167.316	186.687	170.090

Table A.7
Time Integral Performance 2

Product Recipe A
Repara. Step=0.25, Sampling Step=0.25
Process conditions are the same as the base case

	Set 1 k & U	Set2 k_0 & f_d	Set 3 EH_{loss} & f_d	Set4 m_M & U
K_1=1.0	K_2=0.1			
MaxAE	16.699	16.588	16.587	16.492
IE	-188.544	-162.747	-260.858	-186.147
IAE	208.529	190.522	272.979	206.691
ISE	1527.785	1436.546	1632.835	1492.857
K_1=1.0	K_2=1.0			
MaxAE	16.690	16.162	16.161	16.463
IE	-191.394	-55.426	-126.961	-188.849
IAE	207.010	222.416	293.474	205.043
ISE	1526.439	1514.340	1700.590	1488.277
K_1=2.0	K_2=0.1			
MaxAE	9.873	9.759	9.755	10.118
IE	-56.385	-51.092	-89.508	-57.581
IAE	70.159	66.184	104.185	71.072
ISE	269.571	259.144	310.129	287.191
K_1=2.0	K_2=1.0			
MaxAE	9.866	9.733	9.728	10.086
IE	-56.392	-41.353	-73.268	-57.426
IAE	70.166	64.365	117.164	70.930
ISE	269.571	268.866	342.955	284.814
K_1=3.0	K_2=0.1			
MaxAE	8.147	8.077	8.078	8.145
IE	-32.901	-31.321	-51.966	-32.933
IAE	47.226	46.232	77.350	47.124
ISE	165.080	161.683	194.117	166.367
K_1=3.0	K_2=1.0			
MaxAE	8.147	8.077	8.078	8.145
IE	-33.006	-26.216	-46.495	-33.093
IAE	47.333	41.936	82.635	46.969
ISE	165.187	160.672	208.234	165.361

Table A.8
Time Integral Performance 3

Product Recipe A
Repara. Step=5.0, Sampling Step=0.25
Process conditions are the same as the base case

	Set 1 k & U	Set2 k_p & f_d	Set 3 EH_{loss} & f_d	Set4 m_v & U
K_1=1.0	K_2=0.1			
MaxAE	17.425	16.869	16.613	17.543
IE	-221.111	-191.009	-177.879	-164.127
IAE	241.827	226.097	190.460	251.156
ISE	1854.765	1649.668	1452.327	1865.444
K_1=1.0	K_2=1.0			
MaxAE	16.921	16.399	16.183	17.024
IE	-197.278	-105.735	-85.144	-153.266
IAE	224.229	172.026	212.630	214.574
ISE	1568.832	1318.632	1503.663	1520.207
K_1=2.0	K_2=0.1			
MaxAE	10.111	9.807	9.761	10.132
IE	-60.750	-55.259	-57.375	-40.444
IAE	76.864	73.400	65.884	90.380
ISE	291.740	272.666	260.954	310.517
K_1=2.0	K_2=1.0			
MaxAE	10.069	9.773	9.732	10.088
IE	-52.379	-48.057	-40.979	-36.865
IAE	73.912	70.109	67.488	83.151
ISE	287.336	264.877	265.119	297.419
K_1=3.0	K_2=0.1			
MaxAE	8.147	8.077	8.078	8.145
IE	-36.052	-32.027	-33.283	-20.826
IAE	51.503	50.494	43.927	60.092
ISE	166.892	167.516	161.826	175.261
K_1=3.0	K_2=1.0			
MaxAE	8.147	8.077	8.078	8.145
IE	-32.159	-28.403	-23..093	-19.732
IAE	50.867	48.191	47.037	56.973
ISE	168.194	164.291	169.350	171.144

Table A.9
Time Integral Performance 4

Product Recipe A
Repara. Step=2.0, Sampling Step=0.125
Process conditions are the same as the base case

	Set 1 k & U	Set2 k_0 & f_d	Set 3 EH_{loss} & f_d	Set4 m_M & U
K_1=1.0	K_2=0.1			
MaxAE	16.581	16.053	15.796	16.689
IE	-199.921	-173.781	-153.123	-142.035
IAE	223.487	197.590	175.193	247.529
ISE	1634.210	1356.551	1274.845	1595.518
K_1=1.0	K_2=1.0			
MaxAE	15.577	15.123	14.896	15.681
IE	-171.719	-126.314	-25.514	-153.779
IAE	194.581	170.385	211.511	195.085
ISE	1259.002	1086.764	1537.521	1297.620
K_1=2.0	K_2=0.1			
MaxAE	9.388	9.068	9.001	9.419
IE	-54.977	-49.998	-55.071	-28.799
IAE	70.419	68.338	60.921	86.830
ISE	243.756	225.481	214.437	255.019
K_1=2.0	K_2=1.0			
MaxAE	9.253	8.958	8.899	9.282
IE	-52.633	-43.730	-42.726	-38.589
IAE	68.494	62.681	69.595	73.507
ISE	235.488	219.290	215.469	237.195
K_1=3.0	K_2=0.1			
MaxAE	7.258	7.166	7.160	7.260
IE	-29.743	-27.050	-27.455	-11.943
IAE	43.425	43.735	36.066	54.997
ISE	124.171	123.757	118.563	128.662
K_1=3.0	K_2=1.0			
MaxAE	7.255	7.163	7.157	7.257
IE	-29.230	-24.097	-9.574	-18.273
IAE	43.094	41.261	45.520	48.111
ISE	124.361	124.091	130.677	125.348

Table A.10
Time Integral Performance 5

Product Recipe A
Repara. Step=2.0, Sampling Step=0.25
Process conditions are the same as the base case except f_d=0.004

	Set 1 k & U	Set2 k_0 & f_d	Set 3 EH_{loss} & f_d	Set4 m_M & U
K_1=1.0	K_2=0.1			
MaxAE	17.980	17.285	17.233	17.953
IE	-196.462	-179.549	-193.906	-187.013
IAE	225.120	206.752	204.232	217.169
ISE	1863.644	1630.487	1605.755	1775.665
K_1=1.0	K_2=1.0			
MaxAE	17.575	16.904	16.843	17.550
IE	-189.099	-150.393	-99.140	-178.153
IAE	217.286	181.872	215.885	212.080
ISE	1731.444	1477.547	1600.719	1654.875
K_1=2.0	K_2=0.1			
MaxAE	10.197	9.889	9.830	10.233
IE	-56.254	-53.347	-53.771	-54.956
IAE	79.134	73.634	66.038	73.600
ISE	327.314	296.950	287.225	321.252
K_1=2.0	K_2=1.0			
MaxAE	10.137	9.845	9.796	10.161
IE	-55.991	-50.610	-21.158	-54.298
IAE	78.797	70.682	88.975	74.193
ISE	324.635	293.328	309.684	318.270
K_1=3.0	K_2=0.1			
MaxAE	7.996	7.885	7.877	7.999
IE	-31.436	-29.668	-22.814	-30.581
IAE	49.988	49.378	76.716	46.533
ISE	172.204	168.721	191.680	168.087
K_1=3.0	K_2=1.0			
MaxAE	7.996	7.885	7.877	7.999
IE	-31.161	-27.348	-19.374	-29.442
IAE	50.966	50.911	65.364	49.912
ISE	175.402	176.629	184.087	176.731

$$k = k_0 \exp(-6400/(T+460))\mu^{0.4}.$$

$C_{p,i}$ is same as the base case.

Controller parameter set #3 was not a good choice. EH_{loss} and f_d were too related, they could not 'explain' the several process behaviors, and occasionally resulted in diverging values. Otherwise, considering the TIPs in Tables A.6 to A.10, there are only minor differences in the controlled system performance due to the choice of adjustable parameters. This performance is robust to user choice.

Similarly, there are remarkable differences due to the selection of values of tuning parameters, K_1 which determines the agressiveness of control action. For instance, consider adjustable parameter set 1 in Table A.6. When K_1 is increased from 1 to 2 while K_2=0.1, IAE decreased about 67% and ISE decreased 84%. No significant differences are found due to the choice of K_2. However, one must note that K_2 should be selected such that the error integral not to give unnecessary agressiveness on the control action. For example, adjustable parameter set 2 and 3 in Table A.7 for K_1 =1.0 does not provide better performance when K_2 is increased from 0.1 to 1.0.

When the reparameterization interval is changed from the base case of 2.0 minutes to either 0.25 minutes

(Table A.7) or to 5.0 minutes (Table A.8), TIP performances again show minor influence.

The sampling interval could be an influential controller parameter. When all other conditions the same as the base case, IAE decreased 7% and ISE decreased 16% (Table A.9) with more rapid sampling.

The results in Table A.10 are obtained with the same condition as Table A.6 except the fouling factor is increased 60%. The severe process condition change does not deteriorate the control performance.

The base case controller was selected as the "best" in a TIP sense with infrequent control and parameterization action. It is an aggressive controller. Figure A.20 illustrates that the control action becomes less aggressive when the tuning parameter K1 is changed from 2.0 (base case) to 1.0. As one can see, the dumping rate at time period 23 is only half of the base case. Further, Figure A.21 illustrates that reparameterization interval choice can also influence the over control behavior which is represented by slight "ringing" on the control action between time period 60 to 90. When the interval is 0.25 minutes and K1=1.0, the oscillatory behavior disappears.

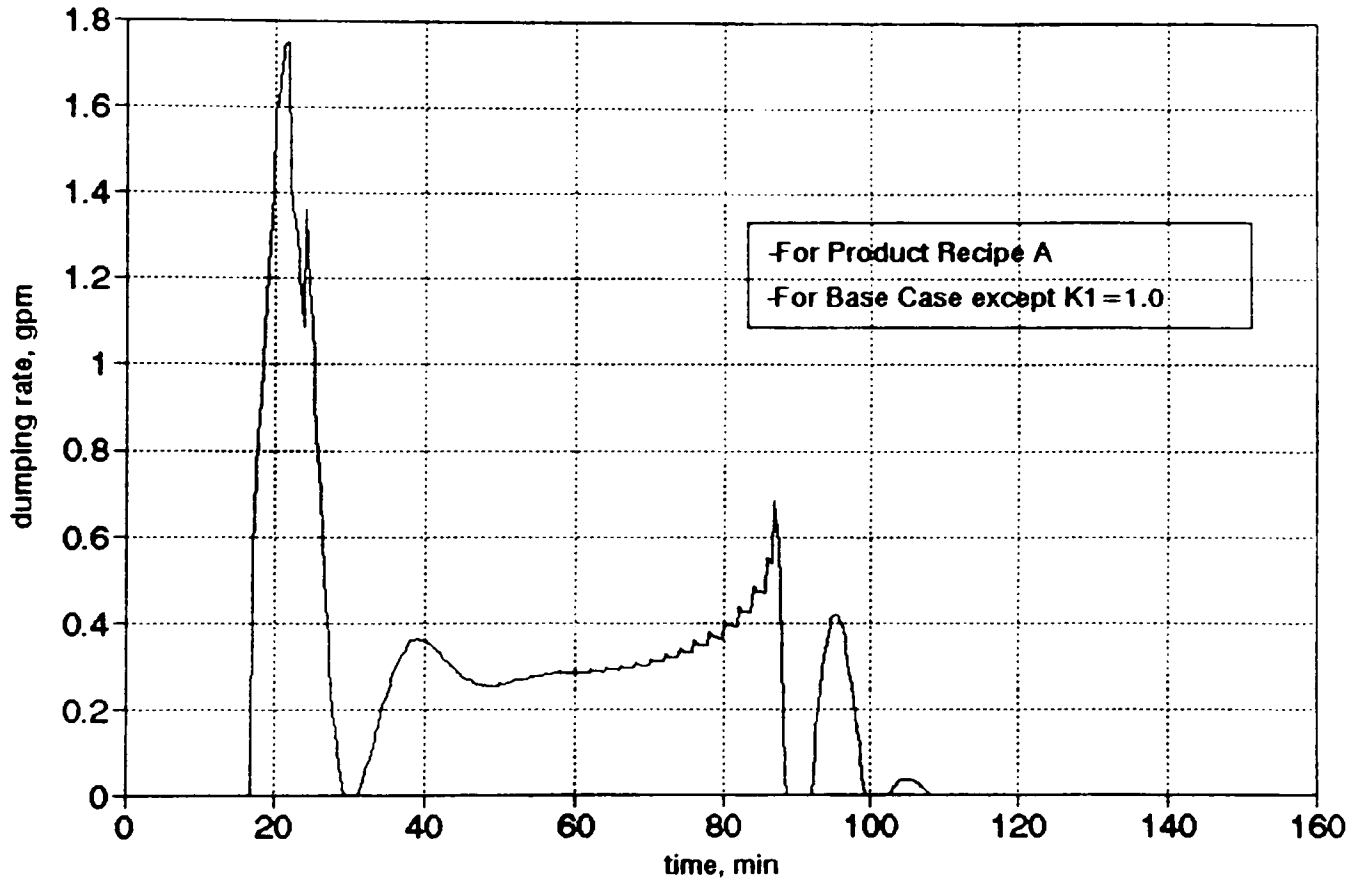


Figure A.20
Control Action Calculation:
Required Cooling Water Flow Rate versus Time

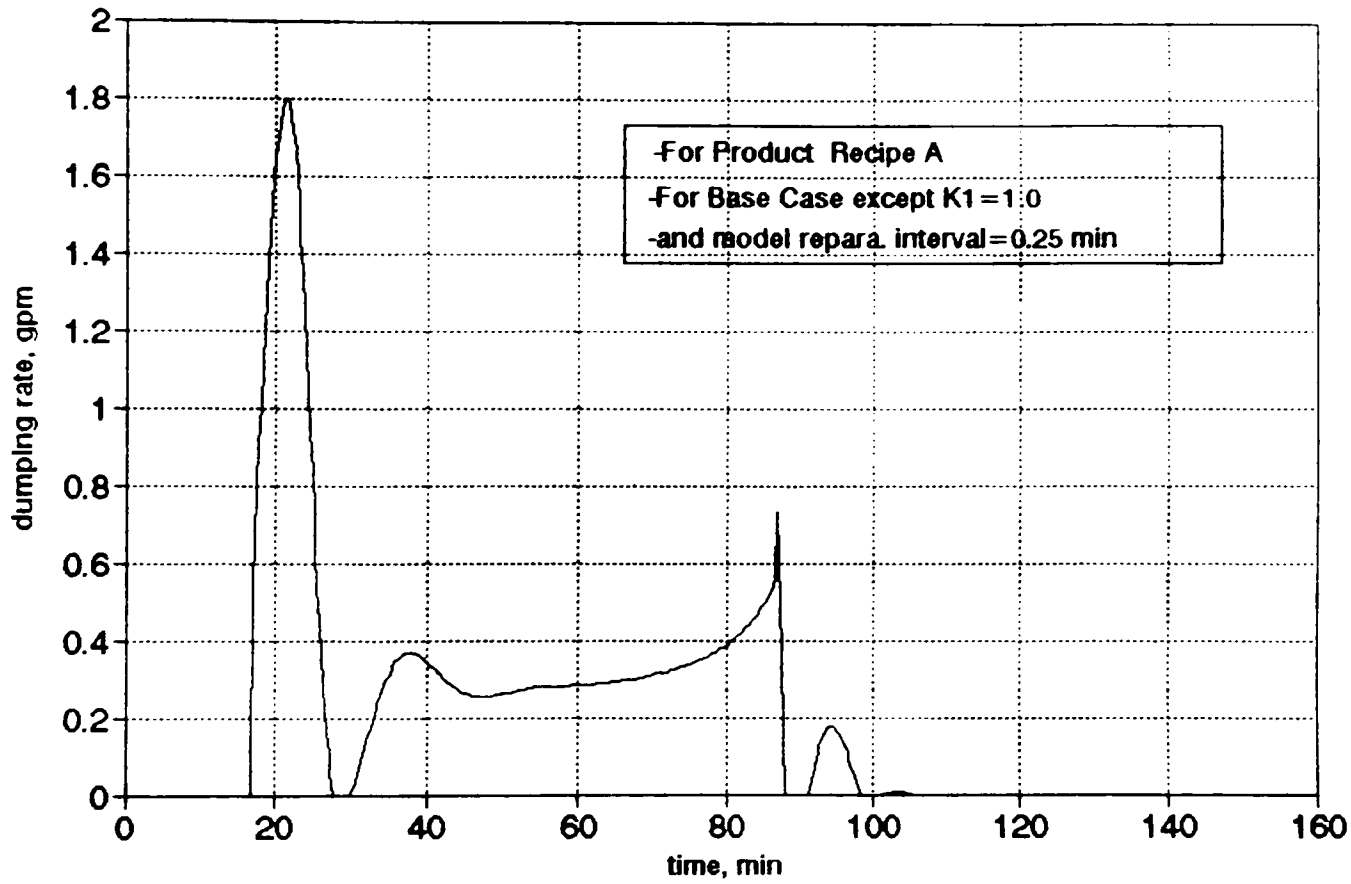


Figure A.21
Control Action Calculation:
Required Cooling Water Flow Rate versus Time

APPENDIX B

AIChE CHALLENGE PROBLEM:

NPMBC/SPC-BASED FILTER APPLICATION

B.1 Summary

The AIChE Industrial Challenge Problem, "Temperature Control of Semibatch Polymerization Reactor", given by S.C. Johnson and Son, Inc. has been solved, and the results are attached in Appendix A. While the challenge problem expresses the severe control difficulties of process nonlinearity, it was relatively idealized. Here as a continuing investigation into the problem, I have incorporated three features to improve realism: start-up control, measurement of monomer conversion, and noise and drifts on process measurements. I now consider density as a measurable process variable, from which the mass of monomer in the reactor can be estimated.

Noise and calibration drifts are classic confounding effects in control. Most chemical and measurement process are influenced by many small, random, independent environmental disturbances. In fact, if a controller takes action on such transients it will unnecessarily induce process upsets and will increase the process variability. An on-line filter must be employed to remove noise from a process variable. This work investigates various statistically-based and standard filters. These include a CUSUM-type on-line filter which removes noise from a measured process variable in a quality control way. Details are included in this report. The filter options discussed here include: (1) Independent or coordinated CUSUM

triggering on each parameter, (2) applying the CUSUM filter to the process variables or model parameters, (3) reintializing the control model on filtered or noisy data, and (4) choice of the CUSUM trigger value.

In these simulations we demonstrate that NPMBC works with noisy data. Noise and drift are generated as discussed in CHAPTER V and added to the deterministic data. While the noise and disturbances generally degrade the control performance, control objectives are still met. Each of the several filter types work.

The use of a third (density) measurement improves the on-line estimate of conversion, making accurate on-line optimization of production rate appear feasible. Total batch control (NPMBC from start-up to shut-down) is implemented.

The relevent modifications of the controller and control model are presented.

B.2 Controller

As we can assume to measure the monomer content in the reactor, the controller can have one more adjustable parameter than was allowed in our initial work. We choose to include the environmental heat loss term, EH_{loss}, as a new model parameter. There may be other valid choices. Thus the approximate control model which is equivalent to Eq.(5.2) can be described as follows:

$$\frac{d\bar{n}_M}{dt} = \frac{\dot{m}_M}{MW_M} - k\bar{n}_M. \qquad (B.1)$$

$$\frac{d(\sum \bar{m}_i Cp_i \bar{T}_r)}{dt} = \dot{m}_M Cp_M (T_{amb} - \bar{T}_r) + k\bar{n}_M (-\Delta H_p) - UA(\bar{T}_r - \bar{T}_j) + EH_{loss}. \qquad (B.2)$$

$$(V_j \rho_{w,j} Cp_{w,j}) \frac{d\bar{T}_j}{dt} = \dot{v}_w \rho_w (T_w - \bar{T}_j) + \dot{m}_s \lambda_s + UA(\bar{T}_r - \bar{T}_j). \qquad (B.3)$$

Since we added a new parameter to the reactor energy balance of the control model, the mismatch between the real process and the control model is reduced.

As the NPMBC control is carried out for the entire time period (total batch control), two reparameterization objectives become reasonable. Among the three operation stages, initial heatup, monomer feeding and holding, most polymerization reaction occurs during the feeding period. Thus the model parameter, k, is important and shall be reparameterized only on that period. The overall heat transfer coefficient, U, and EH_{loss} will be adjusted in all three periods. So the objective is to calculate k, u and EH_{loss} such that ϵ_{mM}, ϵ_{tr} and ϵ_{tj} become zero. After the feed period ends, the latest k value calculated will be kept. In other periods, U and EH_{loss} shall be reparameterized such that ϵ_{tr} and ϵ_{tj} become zero.

For the initial heatup and final holding period, one can get two objective functions shown below:

$$\epsilon_{r,t} = T_{r,t} - \overline{T}_{r,t} - T_{r,t} - T_{r,t-\Delta t} - \Delta t\,[\dot{m}_M Cp_M (T_{amb} - T_{r,t-\Delta t})$$

$$- \overline{UA}_{t-\Delta t} (T_{r,t-\Delta t} - T_{j,t-\Delta t}) - \overline{EH}_{loss}] \,. \qquad (B.4)$$

$$\epsilon_{j,t} = T_{j,t} - \overline{T}_{j,t} - T_{j,t} - T_{j,t-\Delta t} - \Delta t\,[\dot{v}_w \rho_w (T_w - T_{j,t-\Delta t}) +$$

$$\dot{m}_s \lambda_s + \overline{UA}_{t-\Delta t} (T_{r,t-\Delta t} - T_{j,t-\Delta t})] \,. \qquad (B.5)$$

where Δt = reparameterization interval.

Since we have two equations (not severely nonlinear) and two unknowns, Newton's method can solve them. The roots of equations are obtained by:

$$\begin{bmatrix} \frac{\partial e_{r,t}}{\partial U} & \frac{\partial e_{r,t}}{\partial EH_{loss}} \\ \frac{\partial e_{j,t}}{\partial U} & \frac{\partial e_{j,t}}{\partial EH_{loss}} \end{bmatrix} \begin{bmatrix} \Delta U \\ \Delta EH_{loss} \end{bmatrix} = - \begin{bmatrix} e_{r,t} \\ e_{j,t} \end{bmatrix}. \qquad (B.6)$$

If one goes through the steps in Eq.(5.6), it yields:

$$\Delta t \begin{bmatrix} \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial U} & \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial EH_{loss}} \\ \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial U} & \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial EH_{loss}} \end{bmatrix} \begin{bmatrix} \Delta U \\ \Delta EH_{loss} \end{bmatrix} - \begin{bmatrix} e_{r,t} \\ e_{j,t} \end{bmatrix}. \qquad (B.7)$$

Then, for a single step adjustment to U and EH_{loss} incorporating with relaxation coefficient α:

$$\Delta U = \frac{\alpha \left(e_{r,t} \cdot \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial EH_{loss}} - e_{j,t} \cdot \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial EH_{loss}} \right)}{DET}. \qquad (B.8)$$

$$\Delta U = U_t - U_{t-\Delta t}. \tag{B.9}$$

$$\Delta EH_{loss} = \frac{\alpha (e_{j,t} \cdot \frac{\partial \bar{f}_{r,t-\Delta t}}{\partial U} - e_{r,t} \cdot \frac{\partial \bar{f}_{j,t-\Delta t}}{\partial U})}{DET}. \tag{B.10}$$

$$\Delta EH_{loss} = EH_{loss,t} - EH_{loss,t-\Delta t}. \tag{B.11}$$

$$DET = \Delta t \left(\frac{\partial \bar{f}_{r,t-\Delta t}}{\partial U} \frac{\partial \bar{f}_{j,t-\Delta t}}{\partial EH_{loss}} - \frac{\partial \bar{f}_{r,t-\Delta t}}{\partial EH_{loss}} \frac{\partial \bar{f}_{j,t-\Delta t}}{\partial U} \right). \tag{B.12}$$

The partial derivatives in Eqs.(B.8) and (B.10) are defined as follows:

$$\frac{\partial \bar{f}_{r,t-\Delta t}}{\partial U} = -\frac{\bar{A}_{t-\Delta t}(T_{r,t-\Delta t} - T_{j,t-\Delta t})}{\sum (\bar{m}_{l,t-\Delta t} Cp_l)}. \tag{B.13}$$

$$\frac{\partial \bar{f}_{r,t-\Delta t}}{\partial EH_{loss}} = \frac{-1}{\sum (\bar{m}_{l,t-\Delta t} Cp_l)}. \tag{B.14}$$

$$\frac{\partial \bar{f}_{j,t-\Delta t}}{\partial U} = \frac{\bar{A}_{t-\Delta t}(T_{r,t-\Delta t} - T_{j,t-\Delta t}}{\sum V_j \rho_{w,j} C_{p,w_j}}. \tag{B.15}$$

$$\frac{\partial \bar{f}_{j,t-\Delta t}}{\partial EH_{loss}} = 0. \tag{B.16}$$

For the monomer feeding period, one can get three objective functions shown below:

$$\epsilon_{t,M} = n_{M,t} - \bar{n}_{M,t} = n_{M,t} - n_{M,t-\Delta t} - \Delta t \left(\frac{\dot{m}_M}{MW_M} - k n_{M,t-\Delta t} \right) \tag{B.17}$$

$$\epsilon_{t,r} = T_{r,t} - \overline{T}_{r,t} = T_{r,t} - T_{r,t-\Delta t} - \Delta t\,[m_M Cp_M (T_{amb} - T_{r,t-\Delta t}) \cdot$$

$$+k\, n_{m,t-\Delta t}(-\Delta H_p) - \overline{UA}_{t-\Delta t}(T_{r,t-\Delta t} - T_{j,t-\Delta t}) - EH_{loss}] \qquad \text{(B.18)}$$

$$\epsilon_{t,j} = T_{j,t} - \overline{T}_{j,t} = T_{j,t} - T_{j,t-\Delta t} - \Delta t\,[\dot{v}_w \rho_w (T_w - T_{j,t-\Delta t}) +$$

$$\dot{m}_s \lambda_s + \overline{UA}_{t-\Delta t}(T_{r,t-\Delta t} - T_{j,t-\Delta t})]\,. \qquad \text{(B.19)}$$

where Δt = reparameterization interval.

Since we have three equations (not severely nonlinear) and three unknowns, Newton's method can solve them. The roots of equations are obtained by:

$$\begin{bmatrix} \frac{\partial \epsilon_{m,t}}{\partial k} & \frac{\partial \epsilon_{m,t}}{\partial U} & \frac{\partial \epsilon_{m,t}}{\partial EH_{loss}} \\ \frac{\partial \epsilon_{r,t}}{\partial k} & \frac{\partial \epsilon_{r,t}}{\partial U} & \frac{\partial \epsilon_{r,t}}{\partial EH_{loss}} \\ \frac{\partial \epsilon_{j,t}}{\partial k} & \frac{\partial \epsilon_{j,t}}{\partial U} & \frac{\partial \epsilon_{j,t}}{\partial EH_{loss}} \end{bmatrix} \begin{bmatrix} \Delta k \\ \Delta U \\ \Delta EH_{loss} \end{bmatrix} = - \begin{bmatrix} \epsilon_{m,t} \\ \epsilon_{r,t} \\ \epsilon_{j,t} \end{bmatrix}. \qquad \text{(B.20)}$$

If one goes through the steps in Eq.(5.6), it yields:

$$\Delta t \begin{bmatrix} \frac{\partial \overline{f}_{n_M,t-\Delta t}}{\partial k} & \frac{\partial \overline{f}_{n_M,t-\Delta t}}{\partial U} & \frac{\partial \overline{f}_{n_M,t-\Delta t}}{\partial EH_{loss}} \\ \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial k} & \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial U} & \frac{\partial \overline{f}_{r,t-\Delta t}}{\partial EH_{loss}} \\ \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial k} & \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial U} & \frac{\partial \overline{f}_{j,t-\Delta t}}{\partial EH_{loss}} \end{bmatrix} \begin{bmatrix} \Delta k \\ \Delta U \\ \Delta EH_{loss} \end{bmatrix} - \begin{bmatrix} \epsilon_{m,t} \\ \epsilon_{r,t} \\ \epsilon_{j,t} \end{bmatrix}. \qquad \text{(B.21)}$$

Then, for a single step adjustment to k, U and EH_{loss} incorporating with relaxation coefficient α:

$$\Delta k = \frac{\alpha\left(-\epsilon_{m,t}\cdot\frac{\partial f_{j,t-\Delta t}}{\partial U}\cdot\frac{\partial f_{r,t-\Delta t}}{\partial EH_{loss}}\right)}{DET}. \tag{B.22}$$

$$\Delta k = k_t - k_{t-\Delta t}. \tag{B.23}$$

$$\Delta U = \frac{\alpha\left(-\epsilon_{j,t}\cdot\frac{\partial f_{r,t-\Delta t}}{\partial EH_{loss}}\cdot\frac{\partial f_{n_M,t-\Delta t}}{\partial k}\right)}{DET}. \tag{B.24}$$

$$\Delta U = U_t - U_{t-\Delta t}. \tag{B.25}$$

$$\Delta EH_{loss} = \frac{\alpha\left(\epsilon_{j,t}\frac{\partial f_{n_M,t-\Delta t}}{\partial k}\frac{\partial f_{r,t-\Delta t}}{\partial U} + \epsilon_{m,t}\frac{\partial f_{r_t-\Delta t}}{\partial k}\frac{\partial f_{j_t-\Delta t}}{\partial U} - \epsilon_{r,t}\frac{\partial f_{j_t-\Delta t}}{\partial U}\frac{\partial f_{n_M,t-\Delta t}}{\partial k}\right)}{DET}. \tag{B.26}$$

$$\Delta EH_{loss} = EH_{loss,t} - EH_{loss,t-\Delta t}. \tag{B.27}$$

$$DET = \Delta t\left(-\frac{\partial f_{m,t-\Delta t}}{\partial k}\cdot\frac{\partial f_{r,t-\Delta t}}{\partial EH_{loss}}\cdot\frac{\partial f_{j,t-\Delta t}}{\partial U}\right). \tag{B.28}$$

The partial derivatives in Eqs.(B.22),(B.24) and (B.26) are defined as follows:

$$\frac{\partial f_{m,t-\Delta t}}{\partial k} = -\bar{n}_{M',t-\Delta t}. \tag{B.29}$$

$$\frac{\partial f_{m,t-\Delta t}}{\partial U} = 0. \tag{B.30}$$

$$\frac{\partial f_{m,t-\Delta t}}{\partial EH_{loss}} = 0. \tag{B.31}$$

$$\frac{\partial \bar{f}_{r,t-\Delta t}}{\partial k} = \frac{\bar{n}_{M,t-\Delta t}(-\Delta H_p)}{\sum(\bar{m}_{l,t-\Delta t}Cp_l)} \qquad (B.32)$$

$$\frac{\partial \bar{f}_{r,t-\Delta t}}{\partial U} = -\frac{\bar{A}_{t-\Delta t}(T_{r,t-\Delta t}-T_{j,t-\Delta t})}{\sum(\bar{m}_{l,t-\Delta t}Cp_l)}. \qquad (B.33)$$

$$\frac{\partial \bar{f}_{r,t-\Delta t}}{\partial EH_{loss}} = \frac{-1}{\sum(\bar{m}_{l,t-\Delta t}Cp_l)}. \qquad (B.34)$$

$$\frac{\partial \bar{f}_{j,t-\Delta t}}{\partial U} = \frac{\bar{A}_{t-\Delta t}(T_{r,t-\Delta t}-T_{j,t-\Delta t})}{\sum(\bar{m}_{l,t-\Delta t}Cp_l)} \qquad (B.35)$$

$$\frac{\partial \bar{f}_{r,t-\Delta t}}{\partial EH_{loss}} = \frac{-1}{\sum(\bar{m}_{l,t-\Delta t}Cp_l)}. \qquad (B.36)$$

$$\frac{\partial \bar{f}_{j,t-\Delta t}}{\partial k} = 0. \qquad (B.37)$$

As we discussed in the Filtering section of Chapter V, the three parameters calculated here will be CUSUM filtered before they are plugged into Eqs.(B.1), (B.2) and (B.3). Note that these differential equations shall be reinitialized at each reparametrization step with measured data, (noisy data not first-order filtered). The CUSUM of each parameter is independently triggered for this case. The trigger is chosen as 2. It means that a change in CUSUM filtered value for a parameter does not necessarily trigger the others. Independent triggering on each parameter is based on the philosophy that parameters are not intercorrelated each other. This observation was not rigorously analyzed. Contrarily, simultaneous triggering (if

one triggered, trigger others too) of parameters might be valid whenever values are strongly correlated.

The filtered parameters are used in the model; thereby, they influence control action. The control action calculates the jacket temperature setpoint and required amount of jacket heat addition or deletion. CUSUM-filtering is again used for the heat term. The trigger is chosen as 3.

B.3 Simulation Results and Discussion

This section presents the NPMBC control results for the noisy process. Except for noise and drifts the process parameters used are the same as the base case of Appendix A. Some of controller factors, however, are modified as follows:

reparameterization interval = 0.25 min,

tuning parameter k1 = 0.6 min^{-1},

tuning parameter k2 = 0.01 min^{-1},

relaxation coefficient for parameter tracking = 0.2,

Initial parameter values are given as $k=0^{min^{-1}}$,

$U=2.0$ Btu/min_ft^2°F and $EH_{loss}=20$.

Other operation schemes are still the same as mentioned in Appendix A.

Based on earlier findings, control is better with a quicker, reparameterization. Accordingly 0.25 min is reduced from our earlier 2 minute period. To make control action be

less influenced by noise, a lower k1 is adopted. The 0.6 reduction from 2.0 for controller "gain" means a mild control action, but it is adequate to keep reactor temperature within a tolerable range. The orginal reactor temperature maximum deviation from set point, 1 °F, will be no longer appropriate for this noisy process, because even the temperature sensor/transmitter drift is as high as 2.0 °F.

The relaxation coefficient α which was selected as 1.0 for the noise-free/2 min reparameterization case is now reasonably chosen as 0.2. This value was a first guess. It worked. No other values were investigated.

The model parameter k is being adjusted only during the monomer feeding period. It is set to zero before monomer feed starts and holds the latest value after the monomer feed ends. Figure B.1 shows calculated and CUSUM-filtered k values for the time period 50 to 60. By contrast, the model parameters, U and EH_{loss} are being adjusted for entire operation period. Figures B.2 and B.3 show calculated and CUSUM filtered U and EH_{loss} values for the same time period.

In Appendix A, the initial heatup was performed manually. The steam valve was kept full open until the reactor temperature reached 180 °F. Note that now, even initial heatup is carried out by NPMBC.

When the monomer is almost depleted and the process becomes nearly stationary, model reparameterization becomes

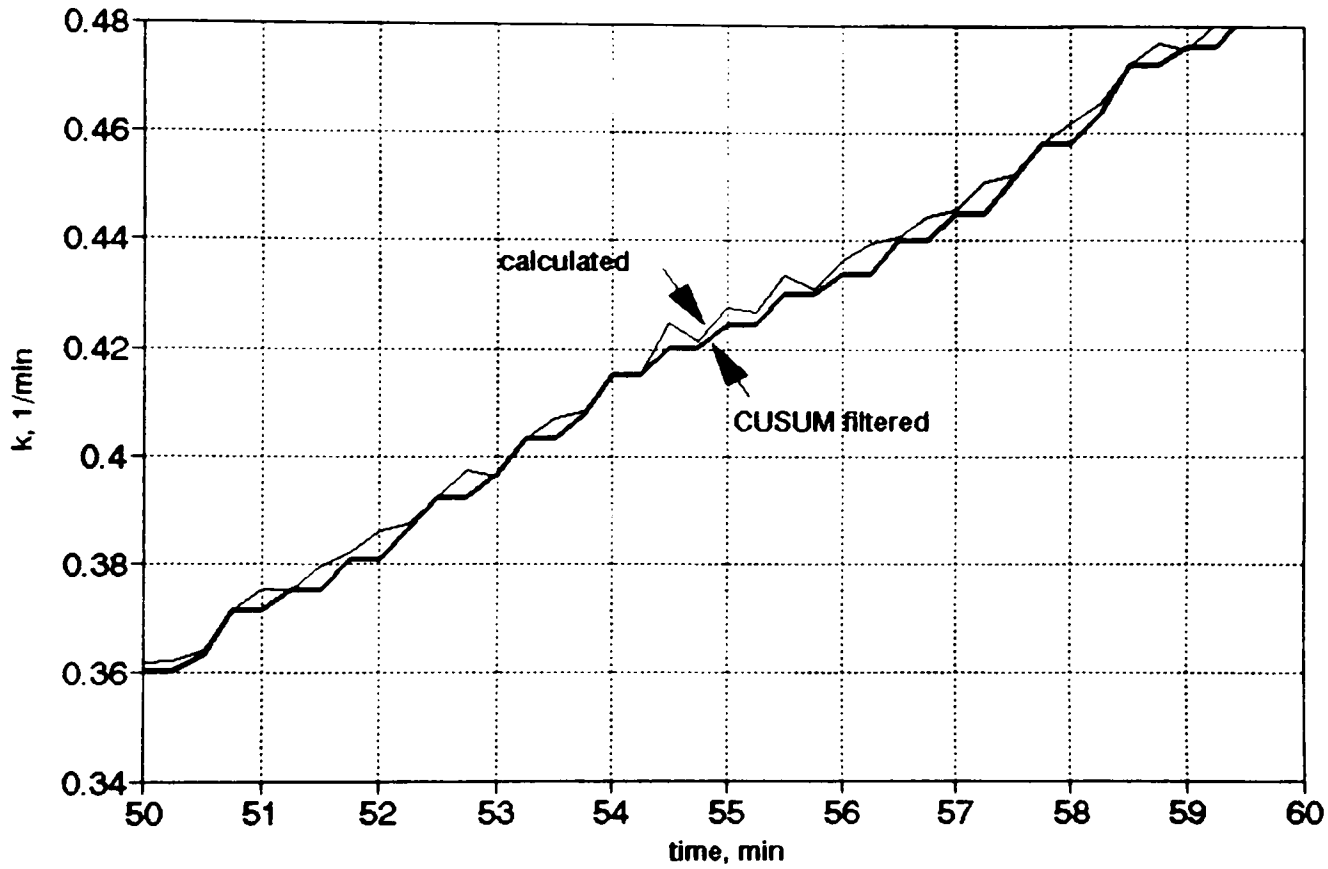


Figure B.1
Adjustable Parameter k versus Time

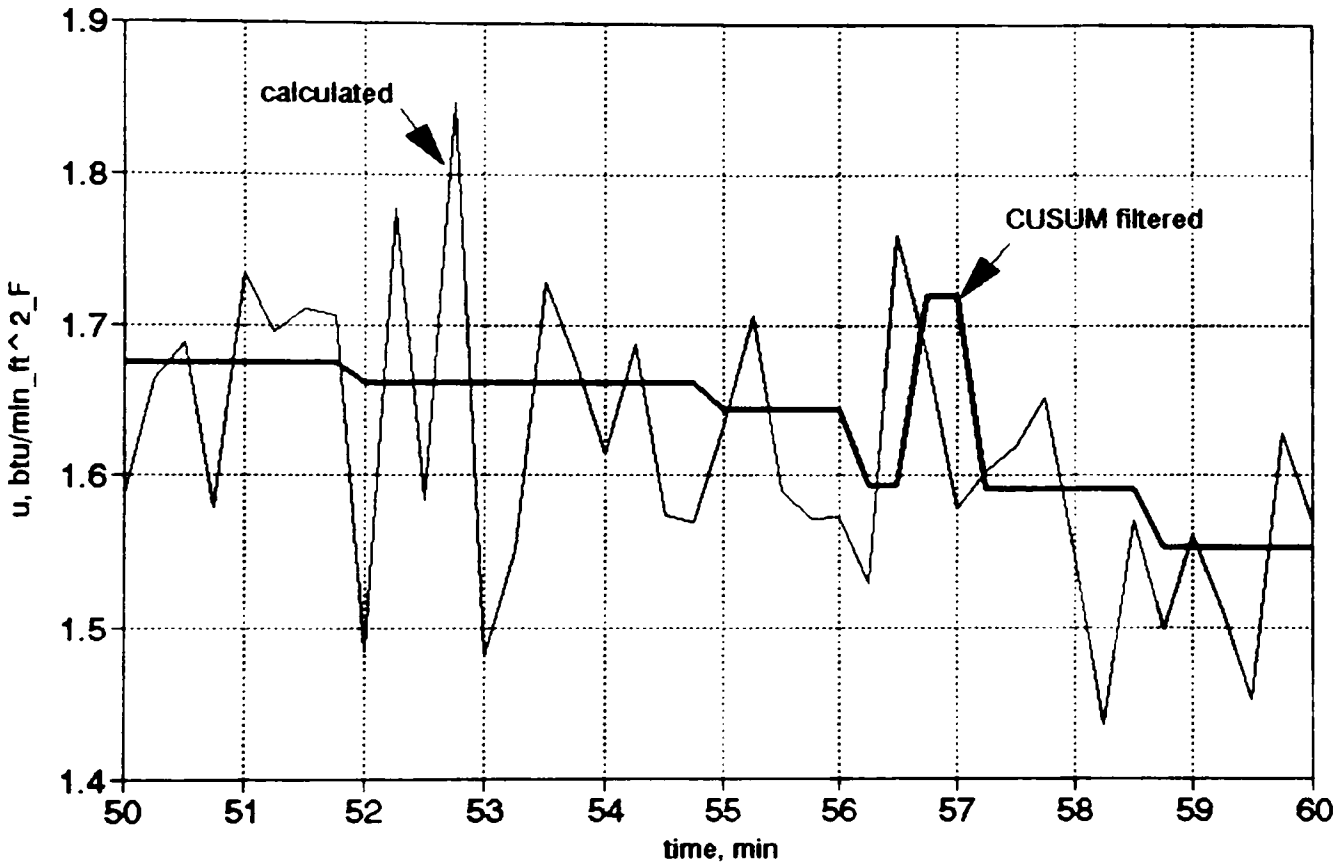


Figure B.2
Adjustable Parameter U versus Time

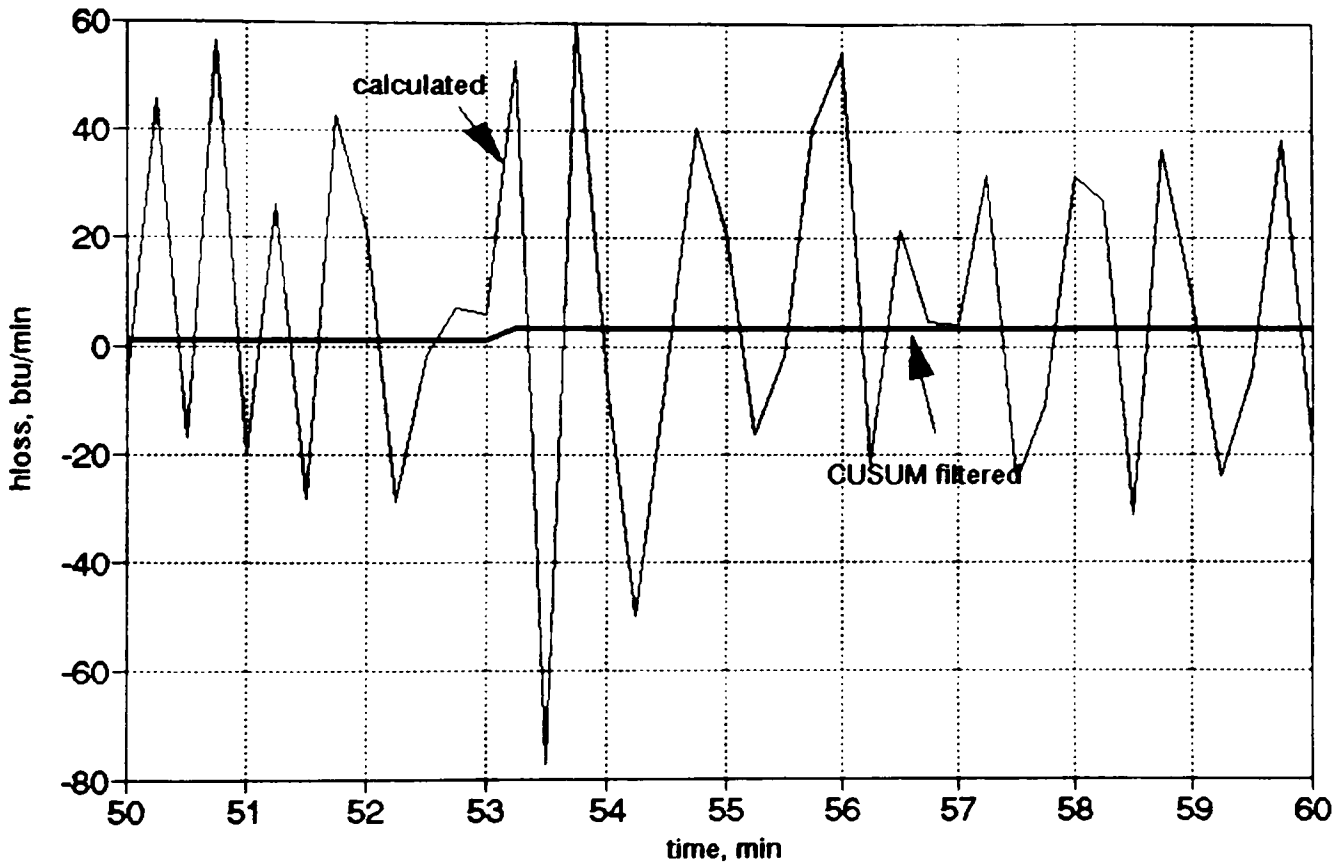


Figure B.3
Adjustable Parameter EH_{loss} versus Time

unnecessary and may even cause a process upset if adjustments are simply due to the noise and drifts. At this stage, the mean temperature difference (mtd) between reactor and jacket is small compared to other stages and very small compared to its noise. To prevent parameter adjustment due to low signal-to-noise ratio, the coefficient of variance of mtd,

$$CV_{mtd} = \frac{\sigma_{\Delta t_m}}{\Delta t_m} = \frac{\sqrt{\sigma_{T_r}^2 + \sigma_{T_j}^2}}{\Delta t_m}. \qquad (B.38)$$

is monitored. The reparameterization step is bypassed if the CV_{mtd} value is greater than 0.07. This value can be chosen by the control engineer. It means that the standard deviation of parameter values shall be maintained at least within 7% of the actual values. We did not investigate other numerical values or, even the need for such an action. Its function performed as desired.

As long as the model is updated by reparameterization, GMC integral action should not be necessary. However, in periods where reparameterization might be bypassed, the GMC integral would be necessary to remove offset caused by a possible model mismatch. Accordingly the GMC integral is rezeroed at each parameterization trigger. Otherwise it acts in the usual manner.

In Figures B.4.(a) and B.5.(a) during the time period of 0 to 30 minutes, the controller calculates a very high

jacket temperature setpoint and resulting heat load requirement. However, since the steam control valve is sized (see Process Description in Appendix A) to allow a maximum steam flow rate 2.16 lb/min, the flow rate is constrained. The amount of heat required is also CUSUM-filtered. The bold line in Figures B.5.(a) to B.5.(c) shows the CUSUM-filtered values of required heat. Here the trigger is chosen as 3. When the polymerization reaction is not the major event in the process, control action is easily influenced by noise even if required process data are filtered before they are used. Figure B.5.(c) demonstrates that phenomena after time period of 100 minutes. Figures B.6 and B.7 show the trend of the manipulated variables, dump rate and steam flow rate.

Figure B.8 shows the NPMBC reactor temperature response. Most of the time, it is still within the 1 °F tolerance level. Figure B.9 shows the corresponding response when all process measurements are filtered by the conventional first-order filter and no CUSUM-type filter is implemented. Its performance is equivalent to the CUSUM filtered case. One filter approach makes temperature slightly less variable during the feed period, the other during the holding period.

Measuring the conversion provides information to the controller and allows the mass balance of the controller model to closely track the actual mass (Figure B.10). One cannot tell a marked deviation from the actual curve for the

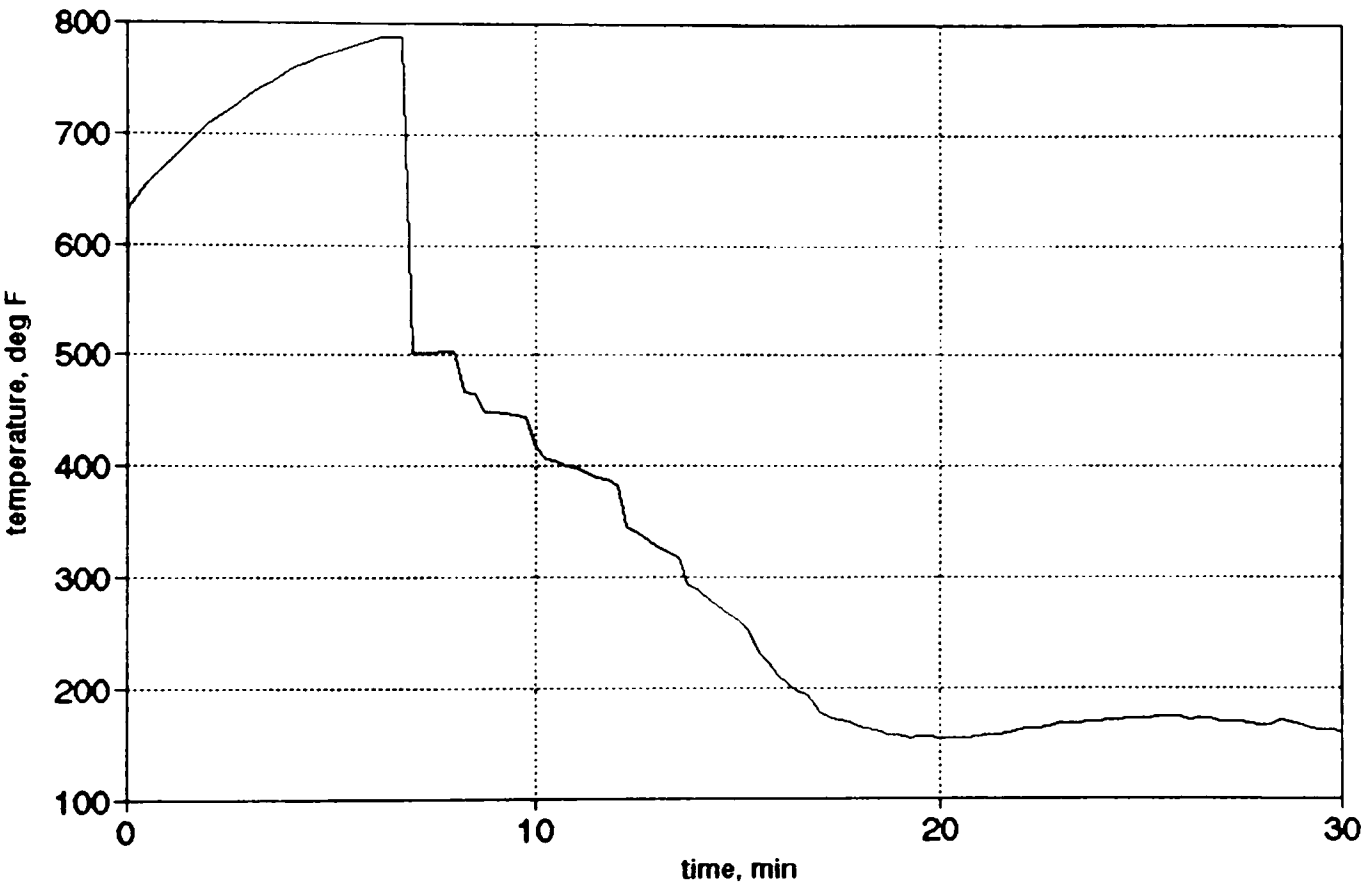


Figure B.4
Control Action Calculation:
Jacket Tmperature Setpoint versus Time
a. 0-30 min Start-up Period

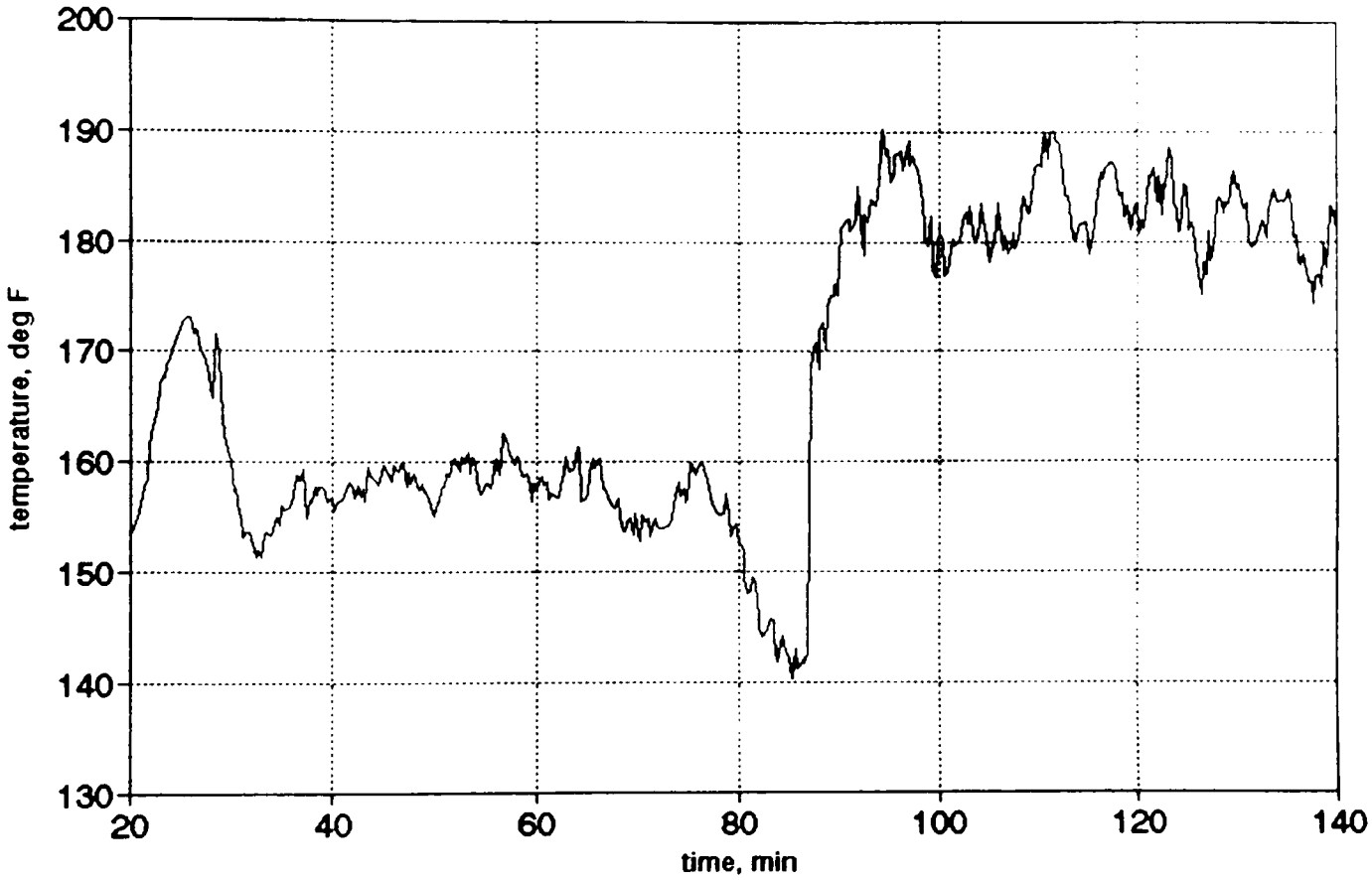


Figure B.4 (Continued)
Control Action Calculation:
Jacket Tmperature Setpoint versus Time
b. Reaction Period

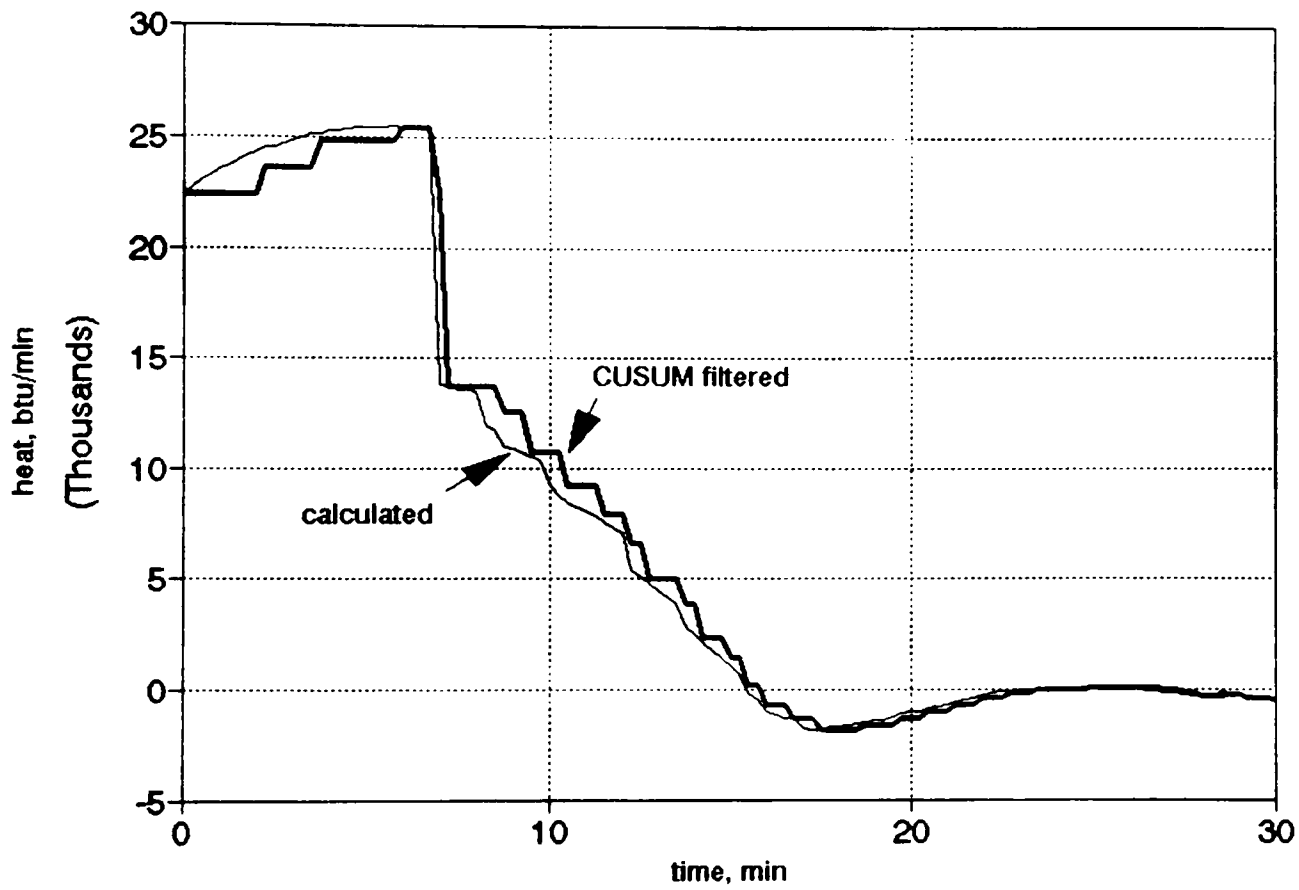


Figure B.5
Control Action Calculation:
Heat Required versus Time
a. 0-30 min Start-up Period

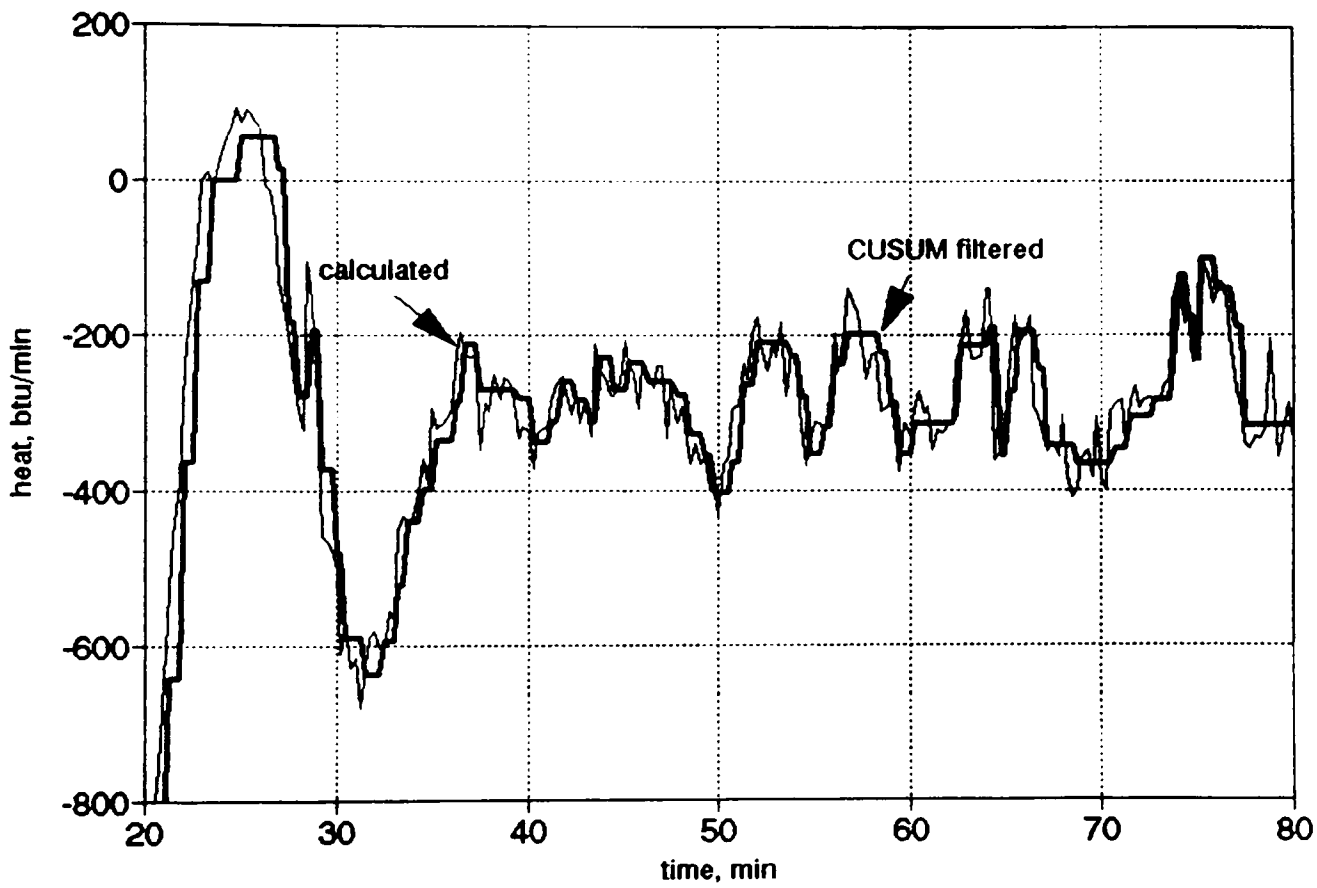


Figure B.5 (Continued)
b. Feed plus Reaction Period

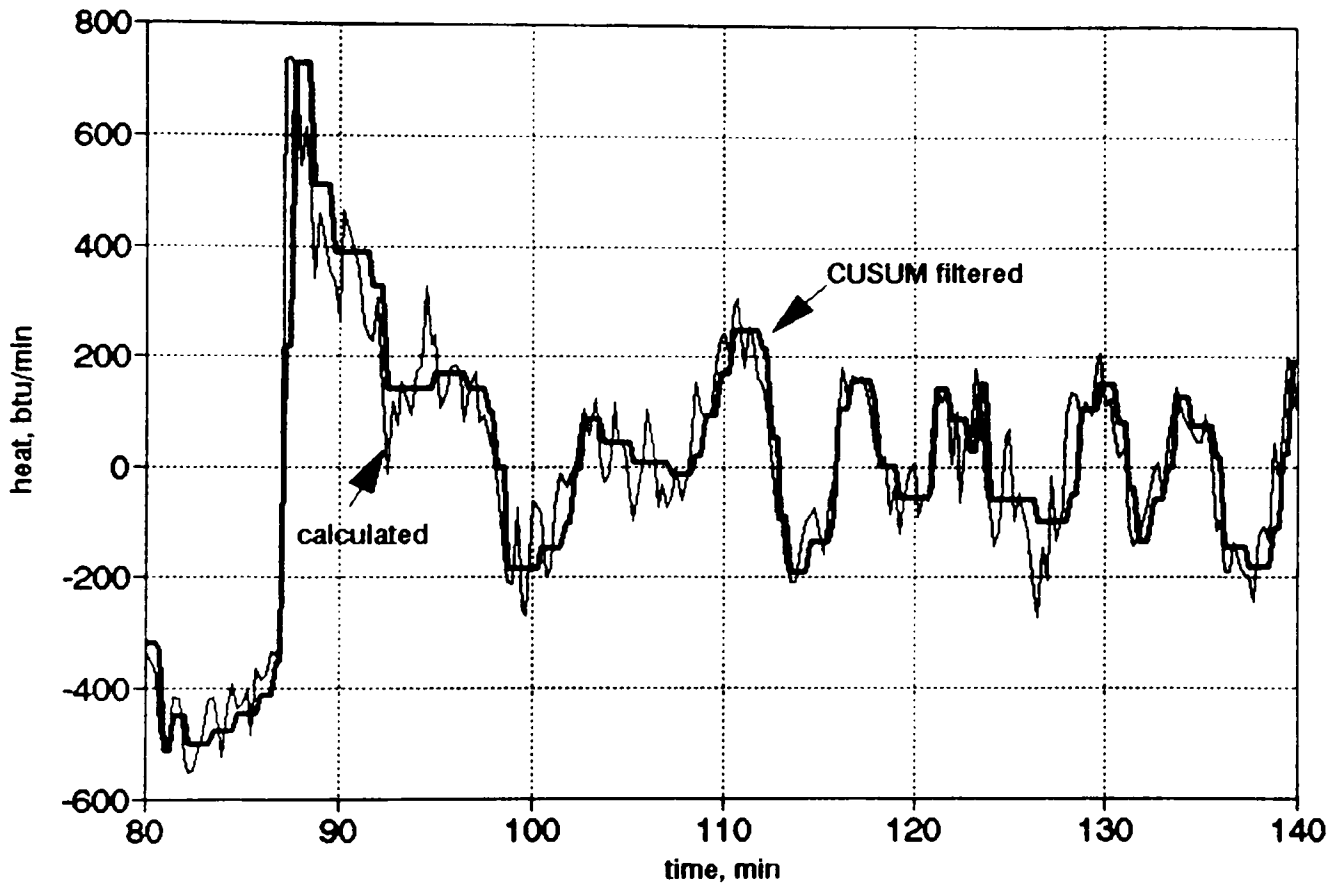


Figure B.5 (Continued)
Control Action Calculation:
Heat Required versus Time
c. Post-Feed Reaction Period

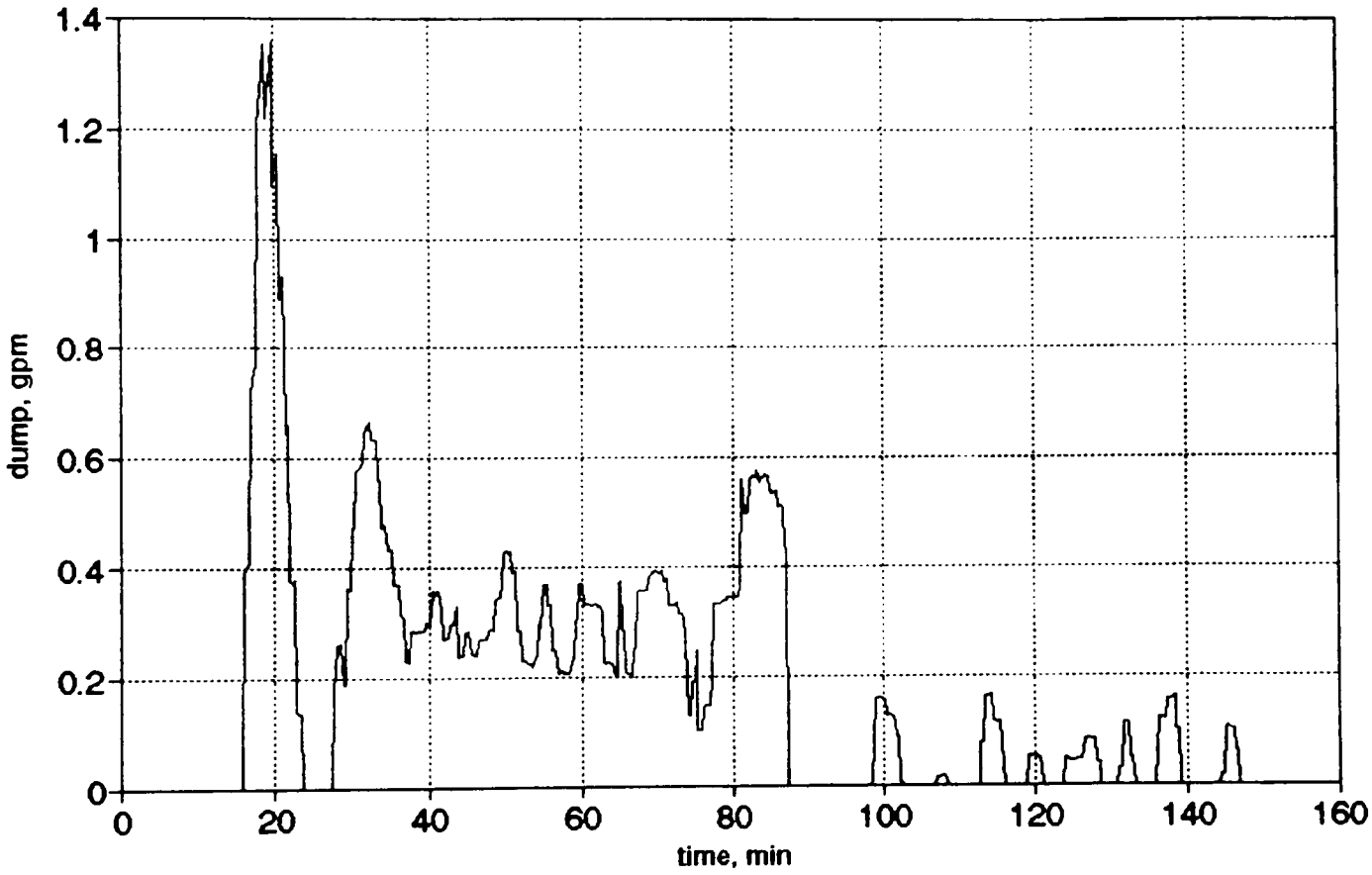


Figure B.6
Control Action Calculation:
Required Cooling Water Flow Rate versus Time

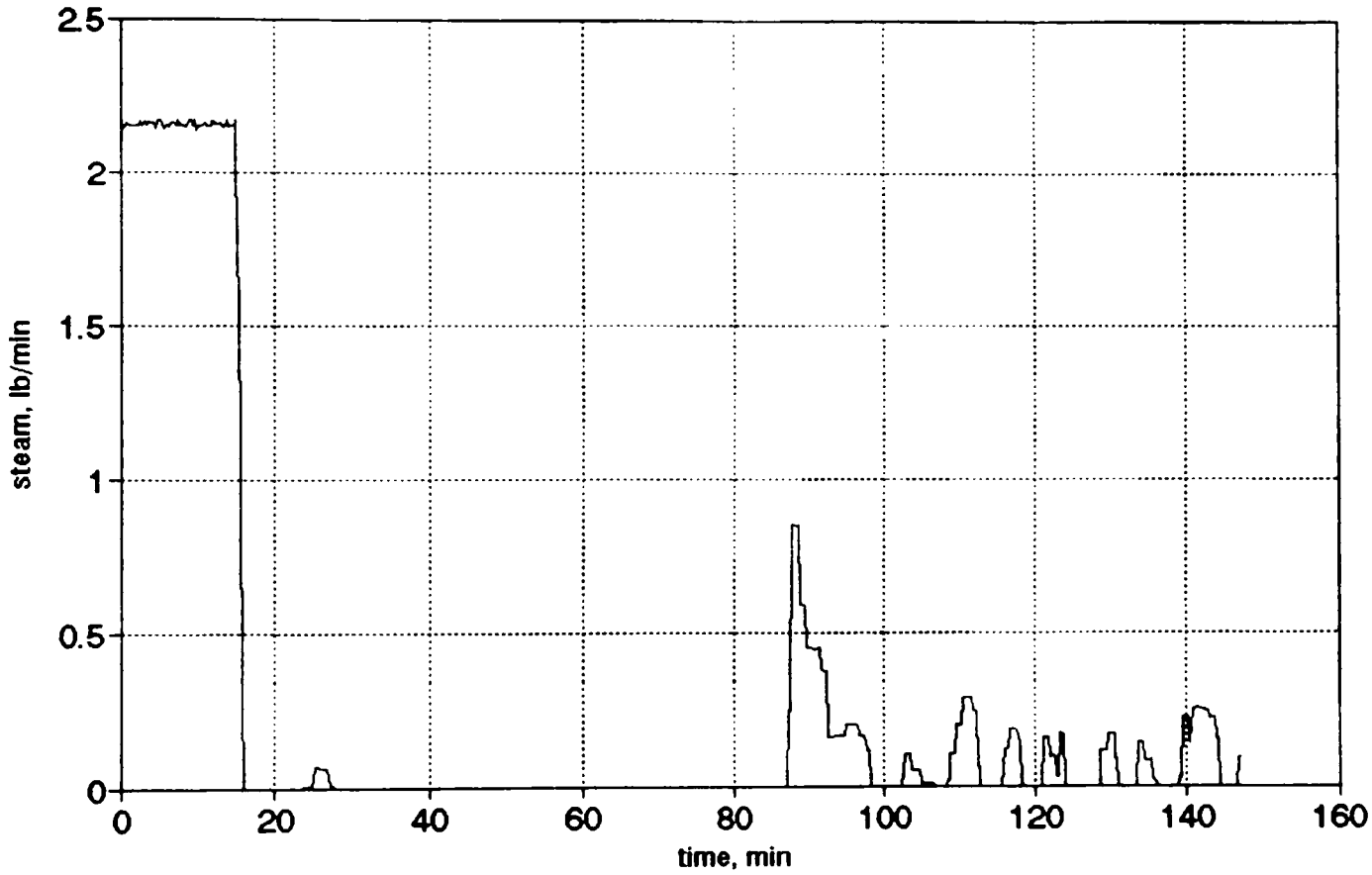


Figure B.7
Control Action Calculation:
Required Steam Flow Rate versus Time

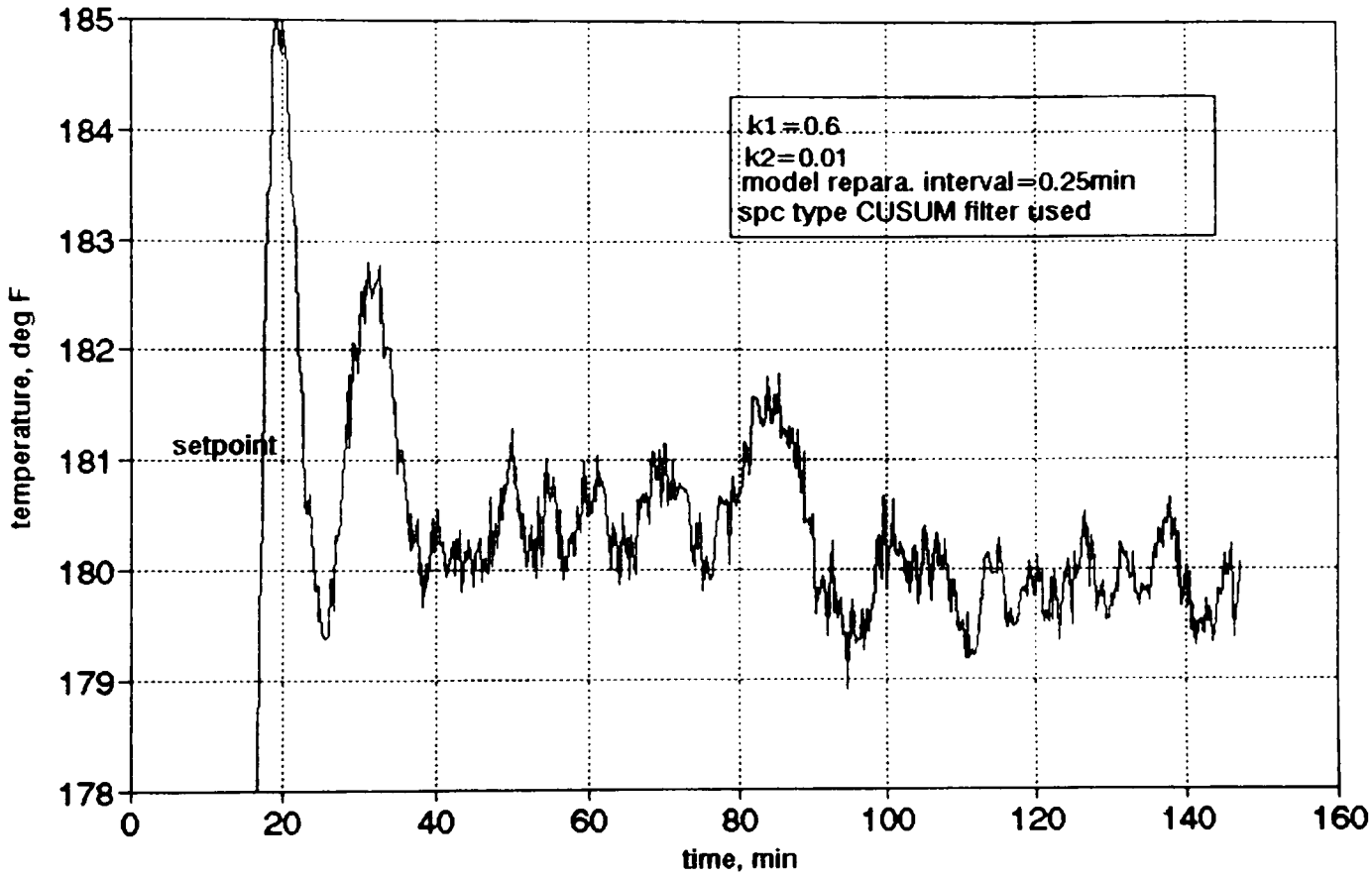


Figure B.8
Reactor Temperature versus Time
(NPMBC Response)

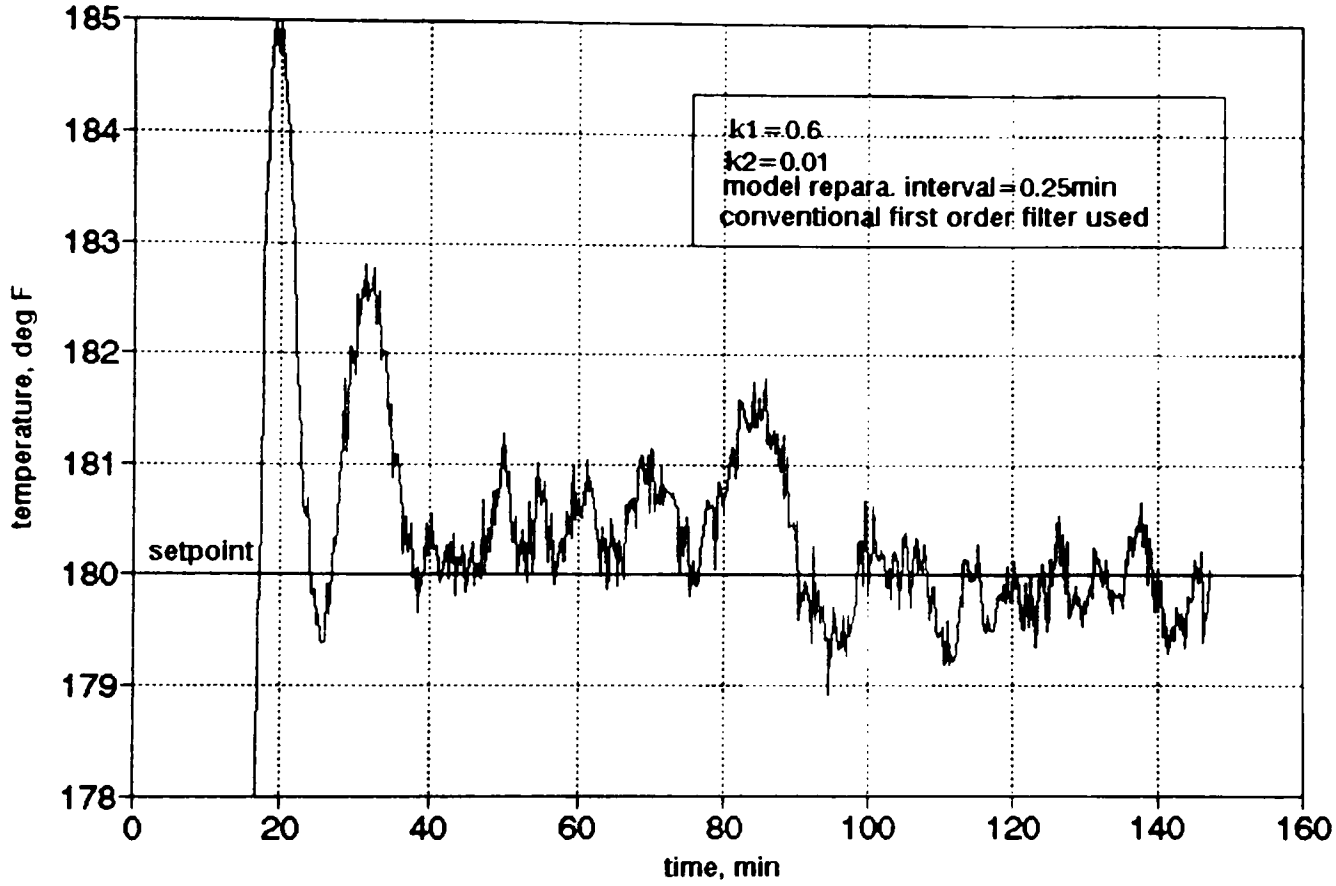


Figure B.9
Reactor Temperature versus Time
(NPMBC Response)

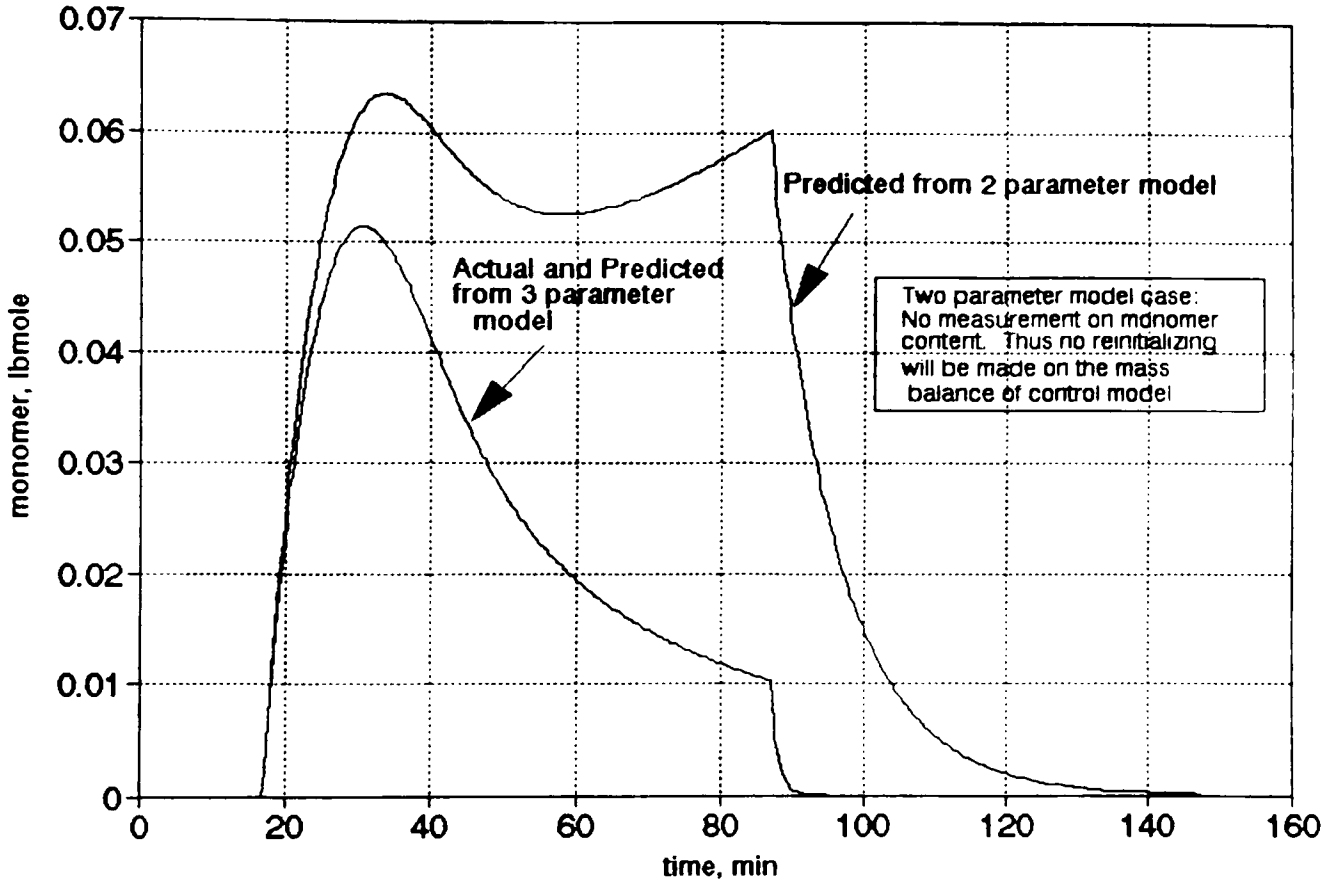


Figure B.10
Unreacted Monomer (Actual/Predicted) versus Time

0.25 min reparameterization interval case because the mass balance of the control model is being reinitialized every 0.25 min. However, for the control model which is based on no monomer measurement, an overestimation of monomer mass keeps propagating because a lower rate constant, k, is being calculated.

APPENDIX C

THE RELATIONSHIP OF MOLECULAR WEIGHT DISTRIBUTION PARAMETERS WITH NUMBER-MOMENT

As defined in Eq.(4.1) , k^{th} number-moment is

$$\mu_k = \sum_{all\ n} n^k D_n .$$

where k is an integer.

D_n is the number of molecules which have chain length of n.

Identities about average chain length :

$$X_N = \mu_1/\mu_0 . \qquad (C.1)$$

$$X_W = \mu_2/\mu_1 . \qquad (C.2)$$

$$X_Z = \mu_3/\mu_2 . \qquad (C.4)$$

$$X_{Z+1} = \mu_4/\mu_3 . \qquad (C.4)$$

$$X_{Z+2} = \mu_5/\mu_4 . \qquad (C.5)$$

As defined in Eq.(4.5), number-variance is

$$\sigma_N^2 = \hat{\mu}_2 - \hat{\mu}_1^2 .$$

Since $\hat{\mu}_k$ represents the k^{th} moment of the normalized distribution about the origin,

$$\sigma_N^2 = \frac{\mu_2}{\mu_0} - \left(\frac{\mu_1}{\mu_0}\right)^2 = \frac{\mu_2}{\mu_1}\frac{\mu_1}{\mu_0} - \left(\frac{\mu_1}{\mu_0}\right)^2 = X_W X_N - X_N^2 . \qquad (C.6)$$

The weight-variance is obtained by inserting one step higher order of average chain length into Eq.(C.6).

$$\sigma_W^2 = X_Z X_W - X_W^2. \qquad (C.7)$$

From Eqs.(4.6) and (4.10), number-skewness is

$$SK_N = \frac{\sigma_3}{SD^3} = \frac{\frac{\mu_3}{\mu_0} - 3\frac{\mu_2}{\mu_0}\frac{\mu_1}{\mu_0} + 2\left(\frac{\mu_1}{\mu_0}\right)^3}{\left[\frac{\mu_2}{\mu_0} - \left(\frac{\mu_1}{\mu_0}\right)^2\right]^{3/2}}. \qquad (C.8)$$

The number-PDI is defined as

$$PDI_N = \frac{X_W}{X_N} = \frac{\mu_2}{\mu_0}\frac{1}{X_N^2}. \qquad (C.9)$$

Rewriting Eq.(C.8) in terms of average chain length and PDI_N, gives

$$SK_N = \frac{X_Z X_W X_N - 3X_W X_N^2 + 2X_N^2}{(X_N^2 PDI_N - X_N^2)^{3/2}} = \frac{X_Z X_W - 3X_W X_N + 2X_N^2}{X_N^2 (PDI_N - 1)^{3/2}}. \qquad (C.10)$$

The weight-skewness is obtained by replacing average chain length and PDI_N with one step higher order of average chain length and $PDI_W (=X_Z/X_W)$.

Therefore,

$$SK_W = \frac{X_{Z+1} X_Z - 3X_Z X_W + 2X_W^2}{X_W^2 (PDI_W - 1)^{3/2}}. \qquad (C.11)$$

From Eqs.(4.7) and (4.10), number-kurtosis is

$$KU_N = \frac{\sigma_4}{SD^4} - 3 = \frac{\mu_4}{\mu_0} - 4\frac{\mu_1\mu_3}{\mu_0^2} + 6\frac{\mu_1^2\mu_2}{\mu_0^3} - \frac{\mu_1^4}{\mu_0^4} - 3. \qquad (C.12)$$

Rewriting Eq.(C.12) in terms of average chain length and PDI_N, gives

$$KU_N = \frac{X_{Z+1}X_ZX_WX_N - 4X_ZX_WX_N^2 + 6X_WX_N^3 - X_N^4}{X_N^4(PDI_N-1)^2} - 3 = \frac{X_{Z+1}X_ZX_W - 4X_ZX_WX_N + 6X_WX_N^2 - X_N^3}{X_N^3(PDI_N-1)^2} - 3. \qquad (C.13)$$

The weight-kurtosis is obtained by replacing the average chain length and PDI with one step higher order of average chain length and PDI. Therefore,

$$KU_W = \frac{X_{Z+2}X_{Z+1}X_Z - 4X_{Z+1}X_Z^2 + 6X_ZX_W^2 - X_W^3}{X_W^3(PDI_W-1)^2} - 3. \qquad (C.14)$$

APPENDIX D

THE DERIVATION OF MOMENT BALANCE EQUATIONS FOR THE PROPOSED MECHANISM

The kinetic equations in Figure 3.2 are converted into several leading moments equations. It is practical to assume that a quasi-steady state is valid for the initiator radical concentration. This transforming process starts with defining the following generating functions.

Generating Function for Growing Polymer, G(s,t).

$$G(s,t) = \sum_{n=1}^{\infty} s^n P_n(t) = sP_1 + s^2P_2 + s^3P_3 + \ldots. \qquad (D.1)$$

Generating Function for Dead Polymer, F(s,t).

$$F(s,t) = \sum_{n=1}^{\infty} s^n D_n(t) = sD_1 + s^2D_2 + s^3D_3 + \ldots. \qquad (D.2)$$

Up to the fifth moment, are the moment equations derived for both the growing and dead polymer. The details of the derivations are presented.

D.1 Conversion Equation

Differentiating system volume expression, $V=V_o(1+\epsilon x)$ with respect to time, gives

$$\frac{dV}{dt} = V_0\epsilon\frac{dx}{dt}. \qquad (D.3)$$

It is true if there is negligible amount of solvent and initiator in the system. Differentiate expression for the fractional monomer conversion, $x=(M_oV_o-MV)/(M_oV_o)$, with respect to time,

$$\frac{dx}{dt} = -\frac{1}{M_0 V_0}\frac{d(MV)}{dt}. \qquad (D.4)$$

Inserting Eq.(3.13) into Eq.(D.4), gives

$$\frac{dx}{dt} = -\frac{V}{M_0 V_0}\left[-k_i RM - (k_p + k_{tm})PM\right]. \qquad (D.5)$$

Since $MV/M_oV_o = 1-x$, Eq.(D.5) becomes

$$\frac{dx}{dt} = k_i R(1-x) + (k_p + k_{tm})(1-x)\lambda_0. \qquad (D.6)$$

Assuming $k_iR(1-x) << (k_p + k_{tm})(1-x)\lambda_o$, Eq.(D.6) can be rewritten as

$$\frac{dx}{dt} = (k_p + k_{tm})(1-x)\lambda_0. \qquad (D.7)$$

The physical interpretation is that the rate of monomer conversion into growing polymer seeds is much slower than the sum of rate of monomer consumption into polymer propagation and into the chain transfer to monomer.

D.2 Initiator Balance Equation

Eq.(3.8) can be rewritten as

$$\frac{1}{V}\left(I\frac{dV}{dt} + V\frac{dI}{dt}\right) = -k_d I. \qquad (D.8)$$

Inserting Eq.(D.3) into Eq.(D.8) and rearranging gives,

$$\frac{dI}{dt} = -k_d I - \frac{eI}{1+ex}\frac{dx}{dt}. \qquad (D.9)$$

Inserting Eq.(D.7) into Eq.(D.9) gives

$$\frac{dI}{dt} = -\frac{eI\lambda_0}{1+ex}(k_p+k_{tm})(1-x) - k_d I. \qquad (D.10)$$

D.3 Solvent Balance Equation

Applying the same procedure as for the initiator balance equation, gives

$$\frac{dS}{dt} = -\frac{eS\lambda_0}{1+ex}(k_p+k_{tm})(1-x) - k_{ts}S\lambda_0. \qquad (D.11)$$

D.4 Moment Balance Equations for The Growing Polymer

D.4.1 Zeroth Moment Balance Equation

Since $\lambda_o = \Sigma_{all\ n}\ P_n = P$,

$$\frac{d\lambda_0}{dt} = \frac{dP}{dt} = -\frac{P}{V}\frac{dV}{dt} + k_i RM - (k_{tc}+k_{td})P^2. \qquad (D.12)$$

and replacing $k_i RM$ with $2fk_d I$, gives

$$\frac{d\lambda_0}{dt} = -\frac{e\lambda_0^2}{1+ex}(k_p+k_{tm})(1-x) + 2fk_d I - (k_{tc}+k_{td})\lambda_0^2. \qquad (D.13)$$

D.4.2 The First Moment Balance Equation

The differential equation for generating function is

$$\frac{\partial}{\partial t}G(s,t) = \frac{d}{dt}\sum_{n=1}^{\infty} s^n P_n = s^n \sum_{n-1}^{\infty} \frac{dP_n}{dt}$$

$$= s^n \sum_{n-1}^{\infty} [-\frac{P_n}{V}\frac{dV}{dt} + k_p M(P_{n-1} - P_n) - (k_{tm}M + k_{ts}S) P_n - k_t P_n P]$$

$$= -\frac{1}{V}\frac{dV}{dt}\sum_{n-1}^{\infty} s^n P_n + k_p M(\sum_{n-1}^{\infty} s^n P_{n-1} - \sum_{n-1}^{\infty} s^n P_n)$$

$$- (k_{tm}M + k_{ts}S)\sum_{n-1}^{\infty} s^n P_n - k_t \lambda_0 \sum_{n-1}^{\infty} s^n P_n .$$

$$= -\frac{e\lambda_0}{1+ex}(k_p + k_{tm})(1-x)G(s,t) + k_p M(s-1)G(s,t) +$$

$$(k_{tm}M + k_{ts}S)G(s,t) - k_t \lambda_0 G(s,t) . \qquad (D.14)$$

According to Eq.(4.14),

$$\frac{d\lambda_1}{dt} = \frac{d}{dt}\frac{\partial G(s,t)}{\partial s}\Big|_{s=1} .$$

Therefore, differentiating Eq.(D.14) with respect to s, gives

$$\frac{d\lambda_1}{dt} = [-\frac{e\lambda_0}{1+ex}(k_p + k_{tm})(1-x)\frac{\partial G(s,t)}{\partial s} + k_p M[G(s,t) + (s-1)\frac{\partial G(s,t)}{\partial s}]$$

$$+ (k_{tm}M + k_{ts}S)\frac{\partial G(s,t)}{\partial s} - k_t \lambda_0 \frac{\partial G(s,t)}{\partial s}]_{s=1} . \qquad (D.15)$$

$$\frac{d\lambda_1}{dt} = -\frac{\epsilon\lambda_0\lambda_1}{1+\epsilon x}(k_p+k_{tm})(1-x)+k_pM\lambda_0+(k_{tm}M+k_{ts}S)\lambda_1-k_t\lambda_0\lambda_1. \qquad (D.16)$$

D.4.3 The Second Moment Balance Equation

According to Eq.(4.14),

$$\frac{d\lambda_2}{dt} = \frac{d}{dt}\left[\frac{\partial G(s,t)}{\partial s}+\frac{\partial^2 G(s,t)}{\partial s^2}\right]_{s-1} = \frac{d\lambda_1}{dt}+\frac{\partial^2}{\partial s^2}\frac{\partial G(s,t)}{\partial t}\Big|_{s-1}$$

$$= \frac{d\lambda_1}{dt}+\left\{\frac{-\epsilon\lambda_0}{1+\epsilon x}(k_p+k_{tm})(1-x)\frac{\partial^2 G(s,t)}{\partial s^2}+k_pM\left[\frac{\partial G(s,t)}{\partial s}+\frac{\partial G(s,t)}{\partial s}+(s-1)\frac{\partial^2 G(s,t)}{\partial s^2}\right]\right.$$

$$\left.+(k_{tm}M+k_{ts}S)\frac{\partial^2 G(s,t)}{\partial s^2}-k_t\lambda_0\frac{\partial^2 G(s,t)}{\partial s^2}\right\}_{s-1}$$

$$-\frac{-\epsilon\lambda_0\lambda_1}{1+\epsilon x}(k_p+k_{tm})(1-x)+k_pM\lambda_0+(k_{tm}M+k_{ts}S)\lambda_1-k_t\lambda_0\lambda_1$$

$$-\frac{\epsilon\lambda_0(\lambda_2-\lambda_1)}{1+\epsilon x}(k_p+k_{tm})(1-x)+2k_pM\lambda_1+(k_{tm}M+k_{ts}S)(\lambda_2-\lambda_1)-k_t\lambda_0(\lambda_2-\lambda_1)$$

$$= -\frac{\epsilon\lambda_0\lambda_2}{1+\epsilon x}(k_p+k_{tm})(1-x)+k_pM(\lambda_0+2\lambda_1)+(k_{tm}M+k_{ts}S)\lambda_2-k_t\lambda_0\lambda_2. \qquad (D.17)$$

D.4.4 The Third Moment Balance Equation

According to Eq.(4.14),

$$\frac{d\lambda_3}{dt} = \frac{d}{dt}\left[\frac{\partial G(s,t)}{\partial s}+3\frac{\partial^2 G(s,t)}{\partial s^2}+\frac{\partial^3 G(s,t)}{\partial s^3}\right]_{s-1}$$

$$= \frac{d\lambda_1}{dt}+3\left(\frac{d\lambda_2}{dt}-\frac{d\lambda_1}{dt}\right)+\frac{\partial^3}{\partial s^3}\frac{\partial G(s,t)}{\partial t}\Big|_{s-1}$$

$$= -2\frac{d\lambda_1}{dt}+3\frac{d\lambda_2}{dt}+\frac{\partial^3}{\partial s^3}\frac{\partial G(s,t)}{\partial t}\Big|_{s=1}$$

$$= -2\frac{d\lambda_1}{dt}+3\frac{d\lambda_2}{dt}+\{\frac{-e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)\frac{\partial^3 G(s,t)}{\partial s^3}+$$

$$k_pM[2\frac{\partial^2 G(s,t)}{\partial s^2}+\frac{\partial^2 G(s,t)}{\partial s^2}+(s-1)\frac{\partial^3 G(s,t)}{\partial s^3}]+$$

$$(k_{tm}M+k_{ts}S)\frac{\partial^3 G(s,t)}{\partial s^3}-k_t\lambda_0\frac{\partial^3 G(s,t)}{\partial s^3}\}_{s=1}$$

$$-\frac{2e\lambda_0\lambda_1}{1+ex}(k_p+k_{tm})(1-x)-2k_pM\lambda_0-2(k_{tm}M+k_{ts}S)\lambda_1+2k_t\lambda_0\lambda_1$$

$$-\frac{3e\lambda_0\lambda_2}{1+ex}(k_p+k_{tm})(1-x)+3k_pM(\lambda_0+2\lambda_1)+3(k_{tm}M+k_{ts}S)\lambda_2-3k_t\lambda_0\lambda_2$$

$$-\frac{e\lambda_0(\lambda_3-3\lambda_2+2\lambda_1)}{1+ex}(k_p+k_{tm})(1-x)+3k_pM(\lambda_2-\lambda_1)$$

$$+(k_{tm}M+k_{ts}S)(\lambda_3-3\lambda_2+2\lambda_1)-k_t\lambda_0(\lambda_3-3\lambda_2+2\lambda_1)$$

$$-\frac{-e\lambda_0\lambda_3}{1+ex}(k_p+k_{tm})(1-x)+k_pM(\lambda_0+3\lambda_1+3\lambda_2)-(k_{tm}M+k_{ts}S)\lambda_3-k_t\lambda_0\lambda_3. \qquad (D.18)$$

D.4.5 The Fourth Moment Balance Equation

According to Eq.(4.14),

$$\frac{d\lambda_4}{dt}=\frac{\partial}{\partial t}[\frac{\partial G(s,t)}{\partial s}+7\frac{\partial^2 G(s,t)}{\partial s^2}+6\frac{\partial^3 G(s,t)}{\partial s^3}+\frac{\partial^4 G(s,t)}{\partial s^4}]_{s=1}$$

$$= \frac{d\lambda_1}{dt} + 7\left(\frac{d\lambda_2}{dt} - \frac{d\lambda_1}{dt}\right) + 6\left(\frac{d\lambda_3}{dt} - 3\frac{d\lambda_2}{dt} + 2\frac{d\lambda_1}{dt}\right) + \frac{\partial^4}{\partial s^4}\frac{\partial G(s,t)}{\partial t}\Big|_{s=1}$$

$$= 6\frac{d\lambda_1}{dt} - 11\frac{d\lambda_2}{dt} + 6\frac{d\lambda_3}{dt} + \frac{\partial^4}{\partial s^4}\frac{\partial G(s,t)}{\partial t}\Big|_{s=1}$$

$$= \frac{6d\lambda_1}{dt} - 11\frac{d\lambda_2}{dt} + 6\frac{d\lambda_3}{dt} + \left\{\frac{-e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)\frac{\partial^4 G(s,t)}{\partial s^4} + \right.$$

$$k_p M\left[3\frac{\partial^3 G(s,t)}{\partial s^3} + \frac{\partial^3 G(s,t)}{\partial s^3} + (s-1)\frac{\partial^4 G(s,t)}{\partial s^4}\right] +$$

$$\left. + (k_{tm}M + k_{ts}S)\frac{\partial^4 G(s,t)}{\partial s^4} - k_t\lambda_0\frac{\partial^4 G(s,t)}{\partial s^4}\right\}_{s=1}$$

$$- \frac{-6e\lambda_0\lambda_1}{1+ex}(k_p+k_{tm})(1-x) + 6k_pM\lambda_0 + 6(k_{tm}M+k_{ts}S)\lambda_1 - 6k_t\lambda_0\lambda_1$$

$$+ \frac{11e\lambda_0\lambda_2}{1+ex}(k_p+k_{tm})(1-x) - 11k_pM(\lambda_0+2\lambda_1) - 11(k_{tm}M+k_{ts}S)\lambda_2 + 11k_t\lambda_0\lambda_2$$

$$- \frac{6e\lambda_0\lambda_3}{1+ex}(k_p+k_{tm})(1-x) + 6k_pM(\lambda_0+3\lambda_1+3\lambda_2) - 6(k_{tm}M+k_{ts}S)\lambda_3 - 6k_t\lambda_0\lambda_3$$

$$- \frac{e\lambda_0(\lambda_4-6\lambda_3+11\lambda_2-6\lambda_1)}{1+ex}(k_p+k_{tm})(1-x) + 4k_pM(\lambda_3-3\lambda_2+2\lambda_1)$$

$$+ (k_{tm}M+k_{ts}S)(\lambda_4-6\lambda_3+11\lambda_2-6\lambda_1) - k_t\lambda_0(\lambda_4-6\lambda_3+11\lambda_2-6\lambda_1)$$

$$- \frac{-e\lambda_0\lambda_4}{1+ex}(k_p+k_{tm})(1-x) + k_pM(\lambda_0+4\lambda_1+6\lambda_2+4\lambda_3) - (k_{ts}S+k_{tm}M)\lambda_4 - k_t\lambda_0\lambda_4. \qquad (D.19)$$

D.4.6 The Fifth Moment Balance Equation

According to Eq.(4.14),

$$\frac{d\lambda_5}{dt} = \frac{d}{dt}\left[\frac{\partial G(s,t)}{\partial s} + 15\frac{\partial^2 G(s,t)}{\partial s^2} + 25\frac{\partial^3 G(s,t)}{\partial s^3} + 10\frac{\partial^4 G(s,t)}{\partial s^2} + \frac{\partial^5 G(s,t)}{\partial s^5}\right]_{s=1}$$

$$-\frac{d\lambda_1}{dt} + 15\left(\frac{d\lambda_2}{dt} - \frac{d\lambda_1}{dt}\right) + 25\left(\frac{d\lambda_3}{dt} - 3\frac{d\lambda_2}{dt} + 2\frac{\lambda_1}{dt}\right) + 10\left(\frac{d\lambda_4}{dt} - 6\frac{d\lambda_3}{dt} + 11\frac{d\lambda_2}{dt} - 6\frac{d\lambda_1}{dt}\right) + \frac{\partial^5}{\partial s^5}\frac{\partial G(s,t)}{\partial t}\Big|_{s=1}$$

$$= -24\frac{d\lambda_1}{dt} + 50\frac{d\lambda_2}{dt} - 35\frac{d\lambda_3}{dt} + 10\frac{d\lambda_4}{dt} + \frac{\partial^5}{\partial s^5}\frac{\partial G(s,t)}{\partial t}\Big|_{s=1}$$

$$= -24\frac{d\lambda_1}{dt} + 50\frac{d\lambda_2}{dt} - 35\frac{d\lambda_3}{dt} + 10\frac{d\lambda_4}{dt} + \left\{\frac{-e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)\frac{\partial^5 G(s,t)}{\partial s^5} + \right.$$

$$\left. k_p M\left[4\frac{\partial^4 G(s,t)}{\partial s^4} + \frac{\partial^4 G(s,t)}{\partial s^4} + (s-1)\frac{\partial^5 G(s,t)}{\partial s^5}\right] + (k_{tm}M + k_{ts}S)\frac{\partial^5 G(s,t)}{\partial s^5} - k_t\lambda_0\frac{\partial^5 G(s,t)}{\partial s^5}\right\}_{s=1}$$

$$= -24\frac{e\lambda_0\lambda_1}{1+ex}(1-x)(k_p+k_{tm}) - 24k_pM\lambda_0 + 24(k_{ts}S+k_{tm}M)\lambda_1 + 24k_t\lambda_0\lambda_1$$

$$-50\frac{e\lambda_0\lambda_2}{1+ex}(1-x)(k_p+k_{tm}) + 50k_pM(\lambda_0+2\lambda_1) - 50(k_{ts}S+k_{tm}M)\lambda_2 - 50k_t\lambda_0\lambda_2$$

$$+35\frac{e\lambda_0\lambda_3}{1+ex}(1-x)(k_p+k_{tm}) - 35k_pM(\lambda_0+3\lambda_1+3\lambda_2) + 35(k_{ts}S+k_{tm}M)\lambda_3 + 35k_t\lambda_0\lambda_3$$

$$-10\frac{e\lambda_0\lambda_4}{1+ex}(1-x)(k_p+k_{tm}) + 10k_pM(\lambda_0+4\lambda_1+6\lambda_2+4\lambda_3) - 10(k_{ts}S+k_{tm}M)\lambda_4 - 10k_t\lambda_0\lambda_4$$

$$-\frac{e\lambda_0(\lambda_5-10\lambda_4+35\lambda_3-50\lambda_2+24\lambda_1)}{1+ex}(1-x)(k_p+k_{tm}) + 5k_pM(\lambda_4-6\lambda_3+11\lambda_2-6\lambda_1)$$

$$+ (k_{tm}M+k_{ts}S)(\lambda_5-10\lambda_4+35\lambda_3-50\lambda_2+24\lambda_1)-k_t\lambda_0(\lambda_5-10\lambda_4+35\lambda_3-50\lambda_2+24\lambda_1)$$

$$--\frac{e\lambda_0\lambda_5}{1+ex}(1-x)(k_p+k_{tm})+k_pM(\lambda_0+10\lambda_2+10\lambda_3+5\lambda_4)-(k_{ts}S+k_{tm}M)\lambda_5-k_t\lambda_0\lambda_5. \qquad (D.20)$$

D.5 The Moment Balance Equations of the Dead Polymer

D.5.1 The Zeroth Mpment Balance Equation

The differential equation for generating gunction is

$$\frac{\partial}{\partial t}F(s,t) - \frac{d}{dt}\sum_{n-1}^{\infty} s^n D_n - s^n\sum_{n-1}^{\infty}\frac{dD_n}{dt}$$

$$-s^n\sum_{n-1}^{\infty}[-\frac{D_n}{V}\frac{dV}{dt}+(k_{tm}M+k_{ts}S)P_n+k_{td}P_nP+\frac{1}{2}k_{tc}\sum_{m-1}^{n-1}P_mP_{n-m}]$$

$$--\frac{e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)\sum_{n-2}^{\infty}s^nD_n+(k_{tm}M+k_{ts}S)\sum_{n-2}^{\infty}s^nP_n+k_{td}P\sum_{n-2}^{\infty}s^nP_n+\frac{1}{2}k_{tc}\sum_{n-2}^{\infty}s^n\sum_{m-1}^{n-1}P_mP_{n-m}$$

$$-\frac{-e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)F(s,t)+(k_{tm}M+k_{ts}S)G(s,t)+k_{td}\lambda_0G(s,t)+\frac{1}{2}k_{tc}\sum_{n-2}^{\infty}s^n\sum_{m-1}^{n-1}P_mP_{n-m}. \qquad (D.21)$$

According to Eq.(4.14),

$$\frac{d\mu_0}{dt}=\frac{\partial}{\partial t}F(s,t)|_{s-1}=\frac{-e\lambda_0\mu_0}{1+ex}(k_p+k_{tm})(1-x)+(k_{tm}M+k_{ts}S)\lambda_0+k_{td}\lambda_0^2+\frac{1}{2}k_{tc}\sum_{n-2m-1}^{\infty}\sum^{n-1}P_mP_{n-m}. \qquad (D.22)$$

Note that $\sum_{n-2m-1}^{\infty}\sum^{n-1} P_m\ P_{n-m} = \lambda_0^2$. (D.23)

Inserting Eq.(D.23) into Eq.(D.22), gives

$$\frac{d\mu_0}{dt} = \frac{-e\lambda_0\mu_0}{1+ex}(k_p+k_{tr})(1-x)+(k_{tm}M+k_{ts}S)\lambda_0+k_{td}\lambda_0^2+\frac{1}{2}k_{tc}\lambda_0^2. \qquad (D.24)$$

D.5.2 The First Moment Balance Equation

According to Eq.(4.14),

$$\frac{d\mu_1}{dt} = \frac{d}{dt}\frac{\partial F(s,t)}{\partial s}\Big|_{s=1} = \Big[\frac{-e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)\frac{\partial F(s,t)}{\partial s}+$$

$$+(k_{tm}M+k_{ts}S)\frac{\partial G(s,t)}{\partial s}+k_{td}\lambda_0\frac{\partial G(s,t)}{\partial s}+\frac{1}{2}k_{tc}\sum_{n=2}^{\infty}ns^{n-1}\sum_{m=1}^{n-1}P_mP_{n-m}\Big]_{s=1}. \qquad (D.25)$$

Note that $\sum_{n=2}^{\infty}n\sum_{m=1}^{n-1}P_mP_{n-m} = 2\lambda_0\lambda_1.$ (D.26)

Inserting Eq.(D.26) into Eq.(D.25), gives

$$\frac{d\mu_1}{dt} = \frac{-e\lambda_0\mu_1}{1+ex}(k_p+k_{tm})(1-x)+(k_{tm}M+k_{ts}S)\lambda_1+(k_{tc}+k_{td})\lambda_0\lambda_1. \qquad (D.27)$$

D.5.3 The Second Moment Balance Equation

According to Eq.(4.14),

$$\frac{d\mu_2}{dt} = \frac{d}{dt}\Big[\frac{\partial F(s,t)}{\partial s}+\frac{\partial^2 F(s,t)}{\partial s^2}\Big]_{s=1} = \frac{d\mu_1}{dt}+\frac{\partial^2}{\partial s^2}\frac{\partial F(s,t)}{\partial t}\Big|_{s=1}$$

$$= \frac{d\mu_1}{dt}+\Big[\frac{-e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)\frac{\partial^2 F(s,t)}{\partial s}+(k_{tm}M+k_{ts}S)\frac{\partial^2 G(s,t)}{\partial s^2}$$

$$+k_{td}\lambda_0\frac{\partial^2 G(s,t)}{\partial s^2}+\frac{1}{2}k_{tc}\sum_{n=2}^{\infty}(n^2-n)s^{n-2}\sum_{m=1}^{n-1}P_mP_{n-m}\Big]_{s=1}. \qquad (D.28)$$

Note that

$$\sum_{n-2}^{\infty} n^2 \sum_{m-1}^{n-1} P_m P_{n-m} = 2(\lambda_0\lambda_2+\lambda_1^2). \qquad (D.29)$$

Inserting Eqs.(D.26), (D.27) and (D.29) into Eq.(D.28), gives

$$\frac{d\mu_2}{dt} = \frac{-e\lambda_0\mu_1}{1+ex}(k_p+k_{tm})(1-x)+(k_{tm}M+k_{ts}S)\lambda_1+(k_{tc}+k_{td})\lambda_0\lambda_1$$

$$-\frac{e\lambda_0(\mu_2-\mu_1)}{1+ex}(k_p+k_{tm})(1-x)+(k_{tm}M+k_{ts}S)(\lambda_2-\lambda_1)+k_{td}\lambda_0(\lambda_2-\lambda_1)+k_{tc}(\lambda_0\lambda_2+\lambda_1^2)-k_{tc}\lambda_0\lambda_1$$

$$-\frac{-e\lambda_0\mu_2}{1+ex}(k_p+k_{tm})(1-x)+(k_{tm}M+k_{ts}S)\lambda_2+(k_{tc}+k_{td})\lambda_0\lambda_2+k_{tc}\lambda_1^2. \qquad (D.30)$$

D.5.4 The Third Moment Balance Equation

According to Eq.(4.14),

$$\frac{d\mu_3}{dt} = \frac{d}{dt}\left[\frac{\partial F(s,t)}{\partial s}+3\frac{\partial^2 F(s,t)}{\partial s^2}+\frac{\partial^3 F(s,t)}{\partial s^3}\right]_{s-1}$$

$$= \frac{d\mu_1}{dt}+3\left(\frac{d\mu_2}{dt}-\frac{d\mu_1}{dt}\right)+\frac{\partial^3}{\partial s^3}\frac{\partial F(s,t)}{\partial t}\Big|_{s-1}$$

$$= -2\frac{d\mu_1}{dt}+3\frac{d\mu_2}{dt}+\left[\frac{-e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)\frac{\partial^3 F(s,t)}{\partial s^3}+(k_{tm}M+k_{ts}S)\frac{\partial^3 G(s,t)}{\partial s^3}\right.$$

$$\left.+k_{td}\lambda_0\frac{\partial^3 G(s,t)}{\partial s^3}+\frac{1}{2}k_{tc}\sum_{n-2}^{\infty}(n^3-3n^2+2n)s^{n-3}\sum_{m-1}^{n-1}P_m P_{n-m}\right]_{s-1} \qquad (D.31)$$

Note that

$$\sum_{n-2}^{\infty} n^3 \sum_{m-1}^{n-1} P_m P_{n-m} = 2(\lambda_0\lambda_3+3\lambda_2\lambda_2). \qquad (D.32)$$

Inserting Eqs.(D.26), (D.27), (D.29), (D.30), (D.32) into Eq.(D.31), gives

$$\frac{d\mu_3}{dt} - 2\frac{e\lambda_0\mu_1}{1+ex}(k_p+k_{tm})(1-x) - 2(k_{tm}M+k_{ts}S)\lambda_2 - 2(k_{tc}+k_{td})\lambda_0\lambda_1$$

$$-3\frac{e\lambda_0\lambda_2}{1+ex}(k_p+k_{tm})(1-x) + 3(k_{tm}M+k_{ts}S)\lambda_2 + 3(k_{tc}+k_{td})\lambda_0\lambda_2 + 3k_{tc}\lambda_1^2$$

$$-\frac{e\lambda_0(\lambda_3-3\lambda_2+2\lambda_1)}{1+ex}(k_p+k_{tm})(1-x) + (k_{tm}M+k_{ts}S)(\lambda_3-3\lambda_2+2\lambda_1)$$

$$+k_{td}\lambda_0(\lambda_3-3\lambda_2+2\lambda_1) + k_{tc}(\lambda_0\lambda_3+3\lambda_1\lambda_2) - 3k_{tc}(\lambda_0\lambda_2+\lambda_1^2) + 2k_{tc}\lambda_0\lambda_1$$

$$--\frac{e\lambda_0\mu_3}{1+ex}(k_p+k_{tm})(1-x) + (k_{tm}M+k_{ts}S)\lambda_3 + (k_{tc}+k_{td})\lambda_0\lambda_3 + 3k_{tc}\lambda_1\lambda_2. \qquad (D.33)$$

D.5.5 The Fourth Moment Balance Equation

According to Eq.(4.14),

$$\frac{d\mu_4}{dt} = \frac{d}{dt}\left[\frac{\partial F(s,t)}{\partial s} + 7\frac{\partial^2 F(s,t)}{\partial s^2} + 6\frac{\partial^3 F(s,t)}{\partial s^3} + \frac{\partial^4 F(s,t)}{\partial s^4}\right]$$

$$= \frac{d\mu_1}{dt} + 7\left(\frac{d\mu_2}{dt} - \frac{d\mu_1}{dt}\right) + 6\left(\frac{d\mu_3}{dt} - 3\frac{d\mu_2}{dt} + 2\frac{d\mu_1}{dt}\right) + \frac{\partial^4}{\partial s^4}\frac{\partial F(s,t)}{\partial t}\Big|_{s=1}$$

$$= 6\frac{d\mu_1}{dt} - 11\frac{d\mu_2}{dt} + 6\frac{d\mu_3}{dt} + \frac{\partial^4}{\partial s^4}\frac{\partial F(s,t)}{\partial t}\Big|_{s=1}$$

$$= 6\frac{d\mu_1}{dt} - 11\frac{d\mu_2}{dt} + 6\frac{d\mu_3}{dt} + \left[\frac{-e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)\frac{\partial^4 F(s,t)}{\partial s^4} + (k_{tm}M+k_{ts}S)\frac{\partial^4 G(s,t)}{\partial s^4}\right.$$

$$\left. + k_{td}\lambda_0\frac{\partial^4 G(s,t)}{\partial s^4} + \frac{1}{2}k_{tc}\bar{\sum_{n=2}}(n^4-6n^3-11n^2-6n)s^{n-4}\sum_{m=1}^{n-1}P_m P_{n-m}\right]_{s=1}. \qquad (D.34)$$

Note that

$$\sum_{n=2}^{\infty} n^4 \sum_{m=1}^{n-1} P_m P_{n-m} = 2(\lambda_0\lambda_4+4\lambda_1\lambda_3+3\lambda_2\lambda_2). \qquad (D.35)$$

Inserting Eqs.(D.26), (D.27), (D.29), (D.30), (D.32), (D.33), (D.35) into Eq.(D.34), gives

$$\frac{d\mu_4}{dt} = \frac{-6e\lambda_0\mu_1}{1+ex}(k_p+k_{tm})(1-x)+6(k_{tm}M+k_{ts}S)\lambda_1+6(k_{tc}+k_{td})\lambda_0\lambda_1$$

$$+11\frac{e\lambda_0\mu_2}{1+ex}(k_p+k_{tm})(1-x)-11(k_{tm}M+k_{ts}S)\lambda_2-11(k_{tc}+k_{td})\lambda_0\lambda_2-11k_{tc}\lambda_1^2$$

$$-6\frac{e\lambda_0\mu_3}{1+ex}(k_p+k_{tm})(1-x)+6(k_{tm}M+k_{ts}S)\lambda_3+6(k_{tc}+k_{td})\lambda_0\lambda_3+18k_{tc}\lambda_1\lambda_2$$

$$-\frac{e\lambda_0(\mu_4-6\mu_3+11\mu_2-6\mu_1)}{1+ex}(k_p+k_{tm})(1-x)+(k_{tm}M+k_{ts}S)(\lambda_4-6\lambda_3+11\lambda_2-6\lambda_1)+$$

$$k_{td}\lambda_0(\lambda_4-6\lambda_3+11\lambda_2-6\lambda_1)+k_{tc}(\lambda_0\lambda_4+4\lambda_1\lambda_3+3\lambda_2\lambda_2)-6k_{tc}(\lambda_0\lambda_3+3\lambda_1\lambda_2)+11k_{tc}(\lambda_0\lambda_2+\lambda_1^2)-6k_{tc}\lambda_0\lambda_1$$

$$-\frac{e\lambda_0\mu_4}{1+ex}(k_p+k_{tm})(1-x)+(k_{tm}M+k_{ts}S)\lambda_4+(k_{tc}+k_{td})\lambda_0\lambda_4+4k_{tc}\lambda_1\lambda_3+3k_{tc}\lambda_2\lambda_2. \qquad (D.36)$$

D.5.6 The Fifth Moment Balance Equation

According to Eq.(4.14),

$$\frac{d\mu_5}{dt} = \frac{d}{dt}\left[\frac{\partial F(s,t)}{\partial s}+15\frac{\partial^2 f(s,t)}{\partial s^2}+25\frac{\partial^3 F(s,t)}{\partial s^3}+10\frac{\partial^4 F(s,t)}{\partial s^4}+\frac{\partial^5 F(s,t)}{\partial s^5}\right]_{s=1}$$

$$-\frac{d\mu_1}{dt}+15\left(\frac{d\mu_2}{dt}-\frac{d\mu_1}{dt}\right)+25\left(\frac{d\mu_3}{dt}-3\frac{d\mu_2}{dt}+2\frac{d\mu_1}{dt}\right)$$

$$+ 10\left(\frac{d\mu_4}{dt}-6\frac{d\mu_3}{dt}+11\frac{d\mu_2}{dt}-6\frac{d\mu_1}{dt}\right)+\frac{\partial^5}{\partial s^5}\frac{\partial F(s,t)}{\partial t}\Big|_{s=1}$$

$$= -24\frac{d\mu_1}{dt}+50\frac{d\mu_2}{dt}-35\frac{d\mu_3}{dt}+10\frac{d\mu_4}{dt}+\frac{\partial^5}{\partial s^5}\frac{\partial F(s,t)}{\partial t}\Big|_{s=1}$$

$$= -24\frac{d\mu_1}{dt}+50\frac{d\mu_2}{dt}-35\frac{d\mu_3}{dt}+10\frac{d\mu_4}{dt}+\left[\frac{-e\lambda_0}{1+ex}(k_p+k_{tm})(1-x)\frac{\partial^5 F(s,t)}{\partial s^5}+(k_{tm}M+k_{ts}S)\frac{\partial^5 G(s,t)}{\partial s^5}\right.$$

$$\left.+ k_{td}\lambda_0\frac{\partial^5 G(s,t)}{\partial s^5}+\frac{1}{2}k_{tc}\sum_{n=2}^{\infty}(n^5-10n^4+35n^3-50n^2+24n)\,s^{n-5}\sum_{m=1}^{n-1}P_m P_{n-m}\right]_{s=1}. \qquad (D.37)$$

Note that

$$\sum_{n=2}^{\infty} n^5\sum_{m=1}^{n-1}P_m P_{n-m}=2(\lambda_0\lambda_5+5\lambda_1\lambda_4+10\lambda_2\lambda_3). \qquad (D.38)$$

Inserting Eqs.(D.26), (D.27), (D.29), (D.30), (D.32), (D.33), (D.35), (D.36), (D.38) into Eq.(D.37), gives

$$\frac{d\mu_5}{dt} = \frac{24e\lambda_0\mu_1}{1+ex}(k_p+k_{tm})(1-x)-24(k_{tm}M+k_{ts}S)\lambda_1-24(k_{tc}+k_{td})\lambda_0\lambda_1$$

$$-\frac{50e\lambda_0\mu_2}{1+ex}(k_p+k_{tm})(1-x)+50(k_{tm}M+k_{ts}S)\lambda_2+50(k_{tc}+k_{td})\lambda_0\lambda_2+50k_{tc}\lambda_1^2$$

$$+35\frac{e\lambda_0\mu_3}{1+ex}(k_p+k_{tm})(1-x)-35(k_{tm}M+k_{ts}S)\lambda^3-35(k_{tc}+k_{td})\lambda_0\lambda_3-105k_{tc}\lambda_1\lambda_2$$

$$+\frac{10e\lambda_0\mu_4}{1+ex}(k_p+k_{tm})(1-x)+10(k_{tm}M+k_{ts}S)\lambda^4+10(k_{tc}+k_{td})\lambda_0\lambda_4+40k_{tc}\lambda_1\lambda_3+30k_{tc}\lambda_2\lambda_2$$

$$-\frac{e\lambda_0(\mu_5-10\mu_4+35\mu_3-50\mu_2+24\mu_1)}{1+ex}(k_p+k_{tm})(1-x)+(k_{tm}M+k_{ts}S)(\lambda_5-10\lambda_4+35\lambda_3-50\lambda_2+24\lambda_1)$$

$$+k_{td}\lambda_0(\lambda_5-10\lambda_4+35\lambda_3-50\lambda_2+24\lambda_1)+k_{tc}(\lambda_0\lambda_5+5\lambda_1\lambda_4+10\lambda_2\lambda_3)$$

$$-10k_{tc}(\lambda_0\lambda_4+4\lambda_1\lambda_3+3\lambda_2\lambda_2)+35k_{tc}(\lambda_0\lambda_3+3\lambda_1\lambda_2)-50k_{tc}(\lambda_0\lambda_2+\lambda_1^2)+24k_{tc}\lambda_0\lambda_1$$

$$=-\frac{e\lambda_0\mu_5}{1+ex}(k_p+k_{tm})(1-x)+(k_{tm}M+k_{ts}S)\lambda_5+(k_{tc}+k_{td})\lambda_0\lambda_5+5k_{tc}\lambda_1\lambda_4+10k_{tc}\lambda_2\lambda_3. \qquad (D.39)$$

The derived moment equations are summarized as follows:

$$\frac{dI}{dt}=-k_dI-\frac{eI\lambda_0}{1+ex}(1-x)(k_p+k_{tm}).$$

$$\frac{dx}{dt}=(k_p+k_{tm})(1-x)\lambda_0.$$

$$\frac{dS}{dt}=\frac{-eS\lambda_0}{1+ex}(1-x)(k_p+k_{tm})-k_{ts}S\lambda_0.$$

$$\frac{d\lambda_0}{dt}=\frac{-e\lambda_0\lambda_0}{1+ex}(1-x)(k_p+k_{tm})+2fk_dI-(k_{tc}+k_{td})\lambda_0\lambda_0.$$

$$\frac{d\lambda_1}{dt}=\frac{-e\lambda_0\lambda_1}{1+ex}(1-x)(k_p+k_{tm})+k_pM\lambda_0-(k_{ts}S+k_{tm}M)\lambda_1-(k_{tc}+k_{td})\lambda_0\lambda_1.$$

$$\frac{d\lambda_2}{dt}=\frac{-e\lambda_0\lambda_2}{1+ex}(1-x)(k_p+k_{tm})+k_pM(\lambda_0+2\lambda_1)-(k_{ts}S+k_{tm}M)\lambda_2-(k_{tc}+k_{td})\lambda_0\lambda_2.$$

$$\frac{d\lambda_3}{dt}=\frac{-e\lambda_0\lambda_3}{1+ex}(1-x)(k_p+k_{tm})+k_pM(\lambda_0+3\lambda_1+3\lambda_2)-(k_{ts}S+k_{tm}M)\lambda_3-(k_{tc}+k_{td})\lambda_0\lambda_3.$$

$$\frac{d\lambda_4}{dt} = \frac{-e\lambda_0\lambda_4}{1+ex}(1-x)(k_p+k_{tm}) + k_pM(\lambda_0+4\lambda_1+6\lambda_2+4\lambda_3) - (k_{ts}S+k_{tm}M)\lambda_4 - (k_{tc}+k_{td})\lambda_0\lambda_4.$$

$$\frac{d\lambda_5}{dt} = \frac{-e\lambda_0\lambda_5}{1+ex}(1-x)(k_p+k_{tm}) + k_pM(\lambda_0+10\lambda_1+10\lambda_3+5\lambda_4) - (k_{ts}S+k_{tm}M)\lambda_5 - (k_{tc}+k_{td})\lambda_0\lambda_5.$$

$$\frac{d\mu_0}{dt} = \frac{-e\lambda_0\mu_0}{1+ex}(1-x)(k_p+k_{tm}) + (k_{tm}M+k_{ts}S)\lambda_0 + (k_{td}+\frac{1}{2}k_{tc})\lambda_0\lambda_0.$$

$$\frac{d\mu_1}{dt} = \frac{-e\lambda_0\mu_1}{1+ex}(1-x)(k_p+k_{tm}) + (k_{tm}M+k_{ts}S)\lambda_1 + (k_{td}+k_{tc})\lambda_0\lambda_1.$$

$$\frac{d\mu_2}{dt} = \frac{-e\lambda_0\mu_2}{1+ex}(1-x)(k_p+k_{tm}) + (k_{tm}M+k_{ts}S)\lambda_2 + (k_{td}+k_{tc})\lambda_0\lambda_2 + k_{tc}\lambda_1\lambda_1.$$

$$\frac{d\mu_3}{dt} = \frac{-e\lambda_0\mu_3}{1+ex}(1-x)(k_p+k_{tm}) + (k_{tm}M+k_{ts}S)\lambda_3 + (k_{td}+k_{tc})\lambda_0\lambda_3 + 3k_{tc}\lambda_1\lambda_2.$$

$$\frac{d\mu_5}{dt} = \frac{-e\lambda_0\mu_5}{1+ex}(1-x)(k_p+k_{tm}) + (k_{tm}M+k_{ts}S)\lambda_5 + (k_{tc}+k_{td})\lambda_0\lambda_5 + 5k_{tc}\lambda_1\lambda_4 + 10k_{tc}\lambda_2\lambda_3.$$

$$\frac{d\mu_4}{dt} = \frac{-e\lambda_0\mu_4}{1+ex}(1-x)(k_p+k_{tm}) + (k_{tm}M+k_{ts}S)\lambda_4 + (k_{tc}+k_{td})\lambda_0\lambda_4 + 4k_{tc}\lambda_1\lambda_3 + 3k_{tc}\lambda_2\lambda_2.$$