

**MULTIVARIABLE MODEL-BASED PREDICTIVE
CONTROL FOR INJECTION MOLDING**

MULTIVARIABLE MODEL-BASED PREDICTIVE
CONTROL FOR INJECTION MOLDING

By

HAIQIAN LU, B. ENG. (Mechanical Engineering)
Si Chuan University, Chengdu, Sichuan, China

A Thesis

Submitted to the School of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree

Master of Applied Science

McMaster University

© Copyright by Haiqian Lu, September 2007

MASTER OF APPLIED SCIENCE (2007)

McMaster University

(Chemical Engineering) Hamilton, Ontario

TITLE: Multivariable Model-based Predictive Control for Injection

Molding

AUTHOR: Haiqian Lu, B.Eng. (Si Chuan University, China)

SUPERVISOR: Dr. A. N. Hrymak

NUMBER OF PAGES: xiii, 139

Abstract

The rigorous quality criterion and intricate shapes of plastic injection molded parts require molders to improve process control systems in order to keep their competitive status in the market. In recent research, various advanced control algorithms are employed to develop in-line process controllers. In modern controllers design, in-mold process variables play a very important role in connecting machine variables and quality variables. Model-based predictive control (MPC) is used to investigate the controllability of cavity pressure and cavity temperature within a cycle or cycle-to-cycle.

The objective of the present work is to demonstrate a procedure to develop MPC controllers based on simulation results. Moldflow[®] was used to simulate the injection molding process for a thin-wall cell phone cover. Cavity pressure profiles and part surface temperature profiles were extracted to develop the dynamic model for controller design. Thermal analysis for the cooling stage was investigated by ANSYS[®] FEM software. Mold surface temperature profiles were used for controller design. Dynamic matrix control, a type of MPC control, was developed by using Matlab[®] MPC Toolbox.

A single-input/single-output MPC controller was developed to control cavity pressure in filling stage by manipulating injection flow rate. Simulation studies were then used to develop a MPC controller to implement a closed-loop control. The controller performed very well to control the pressure profile to trace the set-point, even with melt temperature or mold temperature change. Two MPC controllers were developed to control cavity surface cycle average temperature by manipulating coolant flow rate and

coolant temperature. Both controllers show good controllability for cycle average temperature control.

A two-input/two-output DMC controller was implemented to control cavity pressure and part surface temperature in the packing stage. Packing pressure and mold temperature were manipulated to trace the controlled profile set-points in each sampling time. Results shows that the controller was able to meet the set-point very well, for an unmeasured disturbance, based on a closed-loop test.

All the controllers were developed based on simulation results, which will have some differences with real production data. Therefore, the model parameter and controller tuning parameter should be validated and modified if needed before real-time application.

Acknowledgements

First, I wish to express my sincere appreciation to my supervisor, Dr. A.N. Hrymak, for his constant guidance, encouragement, enthusiasm, and support.

I am grateful to Dr. C.L.E. Swartz for giving me so many suggestion and guidance for my project.

Moreover, I would like to express my sincere thanks to Dr. Stephen W. Tullis, for allowing me to use their software; Ronald Hanson, for the valuable discussion about the FEM software; Zhiwen Chong and Alexei Nikandrov, for the indispensable discussions about process control. I would like to thank my colleagues in polymer process group and the staff in Chemical Engineering department and MMRI for their kindly support.

Most of all, I would like to thank my family: my lover Wei, my little boy Wendi and my parents. Your great support gives me the power to step forward all the time.

Table of Contents

Abstract	iii
Acknowledgements	v
List of Figures	x
List of Tables	xiii
1 Introduction	1
1.1 Injection Molding Process.....	1
1.2 Research Objective.....	4
1.3 Thesis Outline	5
2 Literature Survey	7
2.1 Controlled Variables and Manipulated Variables	8
2.2 Levels of Process Control	12
2.2.1 Level 1: Machine Variable Control	14
2.2.2 Level 2: In-mold Process Variable Control	15
2.2.2.1. Cavity Pressure Control	16
2.2.2.2. Nozzle Pressure Control.....	21
2.2.2.3. Melt Temperature Control.....	22
2.2.2.4. Mold Separation Control.....	25
2.2.3 Level 3: Quality Variable Control	27
2.3 Measurement Instrumentation.....	28
2.3.1 Cavity Pressure Sensors.....	28
2.3.2 Temperature Sensors.....	30

2.3.3	Fill Sensors.....	31
2.4	Advanced Control Algorithms	33
2.4.1	Artificial Neural Networks	33
2.4.2	Self-Tuning Regulator	36
2.4.3	Model Predictive Control (MPC).....	37
2.5	CAE Simulation Programs Applications	41
2.5.1	Molding Process Analysis Programs:	42
2.5.2	Mold Surface Temperature Transient Simulation.....	43
3	Model-Based Predictive Controller Development	46
3.1	Overview	46
3.2	Controller Structure Selection.....	51
3.2.1	Multivariable Process Control	51
3.2.2	Controlled and Manipulated Variables Selection	58
3.3	Simulation Process Preparation:	60
3.3.1	Part Selection	60
3.3.2	Material Selection	62
3.3.3	Mold and Coolant Selection.....	63
3.4	Injection Molding Process Simulation:	64
3.4.1	Model Preparation.....	64
3.4.2	Process Conditions.....	67
3.4.3	Simulation Result.....	68
3.5	FEM Thermal Simulation	69

3.5.1	Model Preparation.....	70
3.5.2	Analysis.....	74
3.6	MIMO Controller Development	76
3.6.1	Model Identification.....	76
3.6.2	MPC Controller Program.....	77
3.6.3	Closed-loop Controller Simulation.....	78
4	Simulation Results	81
4.1	Overview.....	81
4.2	Cavity Pressure Control in Filling Stage.....	82
4.2.1	Simulation Result.....	82
4.2.2	Controller Development.....	85
4.2.3	Controller Performance.....	86
4.3	Cavity Surface Temperature Control	91
4.3.1	Simulation Result.....	91
4.3.2	Controller Development.....	96
4.3.3	Set-Point Control	97
4.4	Multiple-input Multiple-output Controller.....	100
4.4.1	Controller Development.....	100
4.4.2	Set-Point Control	104
5	Conclusions & Recommendations	109
5.1	Conclusions.....	109
5.2	Future Work	111

References	113
Appendix	126
A Cavity Pressure and Temperature Sensors	126
B Calculation of Convection Coefficient.....	128
C Code for SISO MPC Controller	129
D Closed-loop Response Code for MPC Controller	130
E MIMO MPC Controller Code.....	138

List of Figures

Figure 2.1 Multi-level control architecture	13
Figure 2.2 An example of PVT-Diagram.....	20
Figure 2.3 Topology of a three-layered feedforward neural network.....	35
Figure 2.4 Block diagram of a self-tuning regulator.....	36
Figure 2.5: Block diagram for model predictive control	39
Figure 2.6: Schematic of temperature variation of mold	45
Figure 3.1 Basic concept for model predictive control.....	55
Figure 3.2 displays the cell telephone front cover.	61
Figure 3.3 Model for simulation in Moldflow®	67
Figure 3.4 Pressure and temperature measurement position.....	69
Figure 3.5 the Two-dimensional model using for simulation in ANSYS®	71
Figure 3.6 Element geometry using in model meshing (ANSYS, 2007).....	72
Figure 3.7 Meshing model for FEM thermal simulation	73
Figure 3.8 ANSYS analysis flow chart.....	75
Figure 3.9 Illustration for MPI injection flow rate step profile.	80
Figure 4.1 Typical cavity pressure –time curve.....	84
Figure 4.2 Cavity pressure profile - time curve (from Moldflow® MPI simulation).	84
Figure 4.3 Closed-loop responses with a slope set-point (0.3 MPa/msec).	87

Figure 4.4 Cavity pressure set-point and simulation pressure output from MPI with MPC closed-loop control.....	87
Figure 4.5 Closed-loop responses with melt temperature change.	89
Figure 4.6 Cavity pressure set-point and simulation pressure output from MPI with MPC closed-loop control (with melt temperature change).	89
Figure 4.7 Closed-loop responses with mold temperature change.	90
Figure 4.8 Cavity pressure set-point and simulation pressure output from MPI with MPC closed-loop control (with mold temperature change).	90
Figure 4.9 Temperature contours in ANSYS simulation.	93
Figure 4.10 Typical simulated temperature profiles for cavity surface within a cycle.....	94
Figure 4.11 Simulated results of cyclic, transient mold surface average temperature variations in selected locations.....	94
Figure 4.12 Response of cavity surface average cycle temperature to a step change in coolant flow rate.....	95
Figure 4.13 Response of cavity surface average cycle temperature to a step change in coolant temperature.	95
Figure 4.14 Response with set point change to 98 °C, using coolant temperature as manipulated variable.	98
Figure 4.15 Response with set point change to 88 °C, using coolant temperature as manipulated variable.	98
Figure 4.16 Response with set point change to 98 °C, using coolant flow rate as manipulated variable.	99

Figure 4.17 Response with set point change to 90 °C, using coolant flow rate as manipulated variable.	99
Figure 4.18 Cavity pressure in packing stage with different packing pressure settings.	102
Figure 4.19 Cavity pressure in packing stage with different mold temperature settings.	103
Figure 4.20 Part surface temperature with different mold surface temperature settings.	103
Figure 4.21 Closed-loop response with $\lambda_1\lambda_2 = [0 \ 0]$	106
Figure 4.22. Closed-loop response with $\lambda_1\lambda_2 = [2 \ 5]$	107
Figure 4.23 Closed-loop response to an unmeasured disturbance step change.	108

List of Tables

Table 2.1 Three-Level Variables in Injection Molding	10
Table 3.1 Recent research using model predictive controller for molding process.....	47
Table 3.2 Model-based predictive controller structure	60
Table 3.3 Cell Phone Front Cover Properties	62
Table 3.4 Main properties of Cypoloy C1000 HF.	63
Table 3.5 H-13 and water thermal properties.	64
Table 3.6 Key processing parameters using in simulation.	68
Table 3.7 Processing conditions using for simulation.	75
Table 4.1 Tuning parameter of SISO MPC controller.....	86
Table 4.2 Tuning parameters of surface temperature MPC controllers.....	97
Table 4.3 Main tuning parameter of the MIMO MPC controller.	105

Chapter 1

Introduction

Injection molding is one of the most important polymer processing methods and commonly used for mass-production of thermoplastic parts with complex shapes and high precision requirements. Injection molded parts are used in various product industries such as automobiles, appliances and electronics. The market continuously asks for lower priced and higher quality products. However, injection molding is a complicated process in which the material undergoes significant changes in properties due to large pressure and temperature variations, complicated by factors such as: part and mold design and material selection. All these factors make it very challenging to predict part quality and maintain stable operation. Recently, advanced control systems are developed to improve the process capability and reduce the production cost.

1.1 Injection Molding Process

Injection molding involves many variables and disturbances during molding. The main variables of injection molding have been categorized into three levels in order to understand and control the process better: machine variables (e.g. injection speed, mold temperature, pack pressure), in-mold process variables (such as cavity pressure, cavity

melt temperature and melt front advancement), and quality variables (e.g. part weight, dimensions, visual defects such as sink marks).

Machine variables control have been developed for many years and some advanced technologies such as the programmable logic controller (PLC), proportional-integral-derivative (PID) controller, computer-aided engineering (CAE) software and design of experiments (DOE) have been applied and proven to be very useful for machine setup and machine variable control. Many advanced process control technologies have been investigated and reported, such as Artificial Neural Networks (ANNs), Model-Based Predictive Controller (MPC) and self-tuning regulators (STR), which target to improve production efficiency, stability, and part quality. However, the relatively high investment cost and complexity in model tuning and training slow down their further application in industry. Quality variable control has been investigated since the last decade, but it still faces many technical problems such as, complexity of quality dynamic models and lack of online feedback. Comparing the three levels of control (machine variable control, in-mold process variable control and quality variable control), in-mold process variable control is a practical and attractive control strategy.

Currently, the variables used most commonly for in-mold process control are: cavity pressure, nozzle pressure, cavity surface temperature, and mold separation (the gap between the part and mold cavity). Cavity pressure control is the most active research direction and has been proved to be closely related to part quality such as part weight, part dimensions, and flash or sink mark. Cavity surface temperature is another variable, which can reflect the future part quality like weld line and shrinkage. Mold separation is

not a direct process variable related to melt polymer, but mold separation is a very accurate indicator for part weight and thickness. Linear variable differential transformers (LVDT) and ultrasonic shear transducers are used to detect and monitor the gap development.

Model-based Predictive Controller (MPC) is one of the most popular control systems in chemical process control. Researchers have been working for a decade to apply MPC into injection molding process control. Single-Input/Single-Output (SISO) and Multiple-Input/Multiple-Output (MIMO) control systems are investigated in controlling the machines variables (e.g. filling velocity and barrel temperature) and process variables (e.g. cavity pressure and cavity surface temperature).

In order to develop a model-based controller, the most important step is to collect useful data. Usually there are two methods to acquire data, one is to get the data from production lines, the other one is using CAE software (e.g. Moldflow[®], Moldex3D[®], and ANSYS[®]) to simulate the process and collect the data from the result of the analysis. The latter is very attractive, since you can conveniently obtain the data and develop the controller at very low cost, even before the mold is manufactured. The simulation result is very valuable for both mold manufacturers and product processors, since the controller can greatly reduce the test trial period in the startup stage, and predict the process capability, controllability, and stability. Controllability is defined as the ability to move the system from one condition to another (i.e. target set point) over finite time, by manipulating inputs properly.

Accompanying the developments in process control, new measurement instrumentation has been applied to measure and monitor the key process variables in injection molding: cavity pressure, cavity surface temperature, and melt front advancement. Different types of cavity pressure sensors have been widely used in the injection molding industry, while the melt temperature sensors mounted inside the mold cavity are getting more attention. Optical sensors, ultrasonic transducers and other advanced instrumentation are used to monitor the melt front advancement, mold separation, degree of solidification, and used to decide fill-to-pack switchover time. Through the use of in-mold instrumentation, the filling, packing and cooling phases can be understood much more clearly, which is very meaningful for part quality control. However, the related high investment and maintenance costs block them from routine industrial application.

1.2 Research Objective

The objective of this project is to develop a multivariable MPC set point controller that controls the key process variables in order to improve the part quality controllability. There are many different variables and disturbances in the injection molding process, which make the process very complex. A set point controller is very meaningful for process setup and in-line control.

Control model development is done by using simulation software, so that we can actually develop the controller even before mold manufacturing. In this study, a thin-wall cellular telephone cover is selected as the component to be studied, due to its high

precision requirements. The key to develop the MIMO Controller is to select the controller structure. Since both cavity pressure and cavity surface temperature have been proved to be closely related to the final product quality and easily measured, so a two-input by two-output system to control the packing and holding stage is studied. Injection molding is very nonlinear and time-variant, so DMC (Dynamic Matrix Controller) is a good choice for approximate control. In order to develop the controller, the first task is to develop the in-mold process model. Moldflow Plastics Insight (MPI) is used to conduct the process analysis and the dynamic model is identified by Matlab Identification Toolbox. The controller coding, simulation and tuning are done based on Matlab MPC Toolbox.

1.3 Thesis Outline

Chapter 2 presents a literature review of injection molding process control development in the past two decades. Three levels of process control variables and advanced control strategies and algorithms are discussed. Modern measurement instrumentation used in injection molding process control is presented. The theory of MPC controller and its application in injection molding process are explained.

Chapter 3 shows the development procedure of the MIMO MPC controller: selection of the controller structure, model development, model meshing and analysis, and process dynamic model identification. Single-input/single-output (SISO) controllers are discussed as comparison cases since they are currently used in industry. Finally,

multiple-input/multiple-output (MIMO) controllers using the Matlab MPC Toolbox is presented.

Chapter 4 presents the simulation results from CAE software and the MPC controller development procedure. Controller performance, tuning, and validation with CAE simulation software were presented.

Chapter 5 emphasizes the value of applying MIMO MPC controller, for mold manufacturers and product processors. Future work of the project and possible research directions are described.

Chapter 2

Literature Survey

Nowadays, in most injection molding plants, solutions to quality problems as well as new mold setup and validation are mostly based on technicians' knowledge and experience. That definitely can't meet the requirements of modern production, which requires high quality, high efficiency, and lower energy consumption. Even after proper machine and mold set-up, part quality may still shift outside of the specified limits during production due to a variety of causes such as material property changes in different batches of resin, or changes in the ambient environment and machine characteristics. Process conditions must be readjusted which may cause a lot of downtime and scrap. An advanced control system is an effective method to help improving injection molding process control and capability.

An extensive literature review is presented below to describe the knowledge of injection molding process control and discuss the current development of advanced control systems, which are applied to polymer injection molding.

2.1 Controlled Variables and Manipulated Variables

Injection molding process variables were classified into three levels. The first level includes *machine variables*, which can be independently set up and used for control; the second level includes *in-mold process variables*, such as cavity pressure and melt temperature, which can not be independently controlled; and the third level includes *part quality variables*, for example, part weight and dimension (Wang et al., 1999). **Table 2.1** shows the three levels variables (Wang et al., 2000b) and their common applications in modern process control systems.

The variables in level 1, machine variables, are basically the machine setup conditions, which can be controlled and manipulated directly by some common controllers and sensors installed in the machine, such as programmable logic controllers (PLC) and proportional-integral-derivative controllers (PID). Recently, more advanced controllers, such as artificial neural networks (ANNs), Fuzzy controllers and self-tuning regulators (STR) have also been studied and applied to modern molding machines in order to improve the controllability of those variables. Machine variables are used as manipulated variables and adjusted in-cycle or cycle-to-cycle for process control in order to get high quality product.

In-mold process variables, the level 2 dependent variables, are affected not only by machine settings, but also by material properties, mold structure, and environmental conditions. In-mold process variables have been proven to be closely related to the final part quality (Agrawal et al., 1987; Bernard, 1983; Michaeli and Gruber, 2004). On the other hand, process variables can be controlled by manipulating machine variables in-

cycle and cycle-to-cycle. Therefore, through monitoring the in-mold process variables and adjusting the machine variables when needed, final product quality properties can be maintained in a stable and required level, which is the target of process control.

Quality variables are mostly part quality standard items, such as part weight, shrinkage, burn mark and so on, which can be either quantitative or qualitative. They are the final responses of the molding process and can only be controlled by adjusting process conditions, such as machine settings, material and mold configurations. In recent research, part weight is the most commonly used as quality controlled variable for modern process controllers due to its easy acquirement (Kamal et al., 1999; Yang and Gao, 2006). The relationship among quality variables, machine variables and process variables is very complex and not well understood. Moreover, sometimes it is very difficult or even impossible to quantify or qualify the variables on-line. Quality variables are used as controlled variables in modern control systems, which may involve a long time delay.

Table 2.1 Three-Level Variables in Injection Molding (Wang et al., 2000b)

Variable Category	Typical variables	Application in Control Systems
<i>Level 1 – Machine variables (independently controllable)</i>		
Temperature	Barrel temperature (in several zones)	Continuous control, as controlled variable (MIMO)
	Nozzle temperature	Cycle-to-cycle control, as controlled variable
	Coolant temperature	Cycle-to-cycle control, as manipulated variable
Pressure	Pack/hold pressure	In cycle or Cycle-to-cycle control, manipulated variables (through hydraulic servo-valve)
	Backpressure	
	Maximum injection pressure	Cycle-to-cycle control, manipulated variable
Sequence and Motion	Clamp/fill/pack/hold/recovery/eject	Standard sequence
	switchover point	Cycle-to-cycle control, as manipulated variable
	Injection (ram) speed (constant or profiled)	In cycle control, as controlled or manipulated variable
	Screw rotation speed	In cycle or Cycle-to-cycle control, manipulated variable
	Shot size and cushion (via screw displacement)	Cycle-to-cycle control, as manipulated variables

(Continued)

(Continue)

<i>Level 2 – Process (dependent) variables</i>		
Temperature	Melt temperature (nozzle, runner, cavity)	Cycle-to-cycle control, as controlled variables
	Rate of heat dissipation and cooling	Cycle-to-cycle control, as controlled variables
Pressure	Melt pressure (in the nozzle, cavity)	In cycle control, as controlled variable
Flow	Melt front advancement	In cycle control, as controlled variable
	Maximum shear stress	Cycle-to-cycle control, as controlled variables
Clamping	Maximum clamping force	Cycle-to-cycle control, as controlled variable
	Mold separation (maximum value, profile)	In cycle or Cycle-to-cycle control, controlled variable
<i>Level 3 – Part Quality Variables (final response)</i>		
Mechanical and Dimensions	Part weight and part thickness	Cycle-to-cycle control, as controlled variables, involve lots of time delay and very complex dynamic model
	Shrinkages and warpage	
	Strength at the weld lines	
Appearance	Sink marks	
	Aesthetic defects; burn marks, flash, etc.	
	Appearance at the weld lines	

2.2 Levels of Process Control

Based on the idea of variables categorizing, Wang et al. (1999) proposed multiple-level control system architecture, as shown in **Figure 2.1**. These three feedback loops correspond to the three levels of measured/controlled variables. The first level in the feedback control system (machine control) has been successfully investigated and researched for decades and the technologies have been applied in the industry, including the programmable logic controller (PLC), PID (proportional-integral-derivative) controller, and design of experiments (DOE). In-mold process control, the second level control loop, is currently the most active research field. Many research reports have been published recently and some applications are in the market. Unlike the first two levels feedback control, research and implementation of automated closed loop third level control (quality control), started in the last decade, and is still facing many difficulties and challenges.

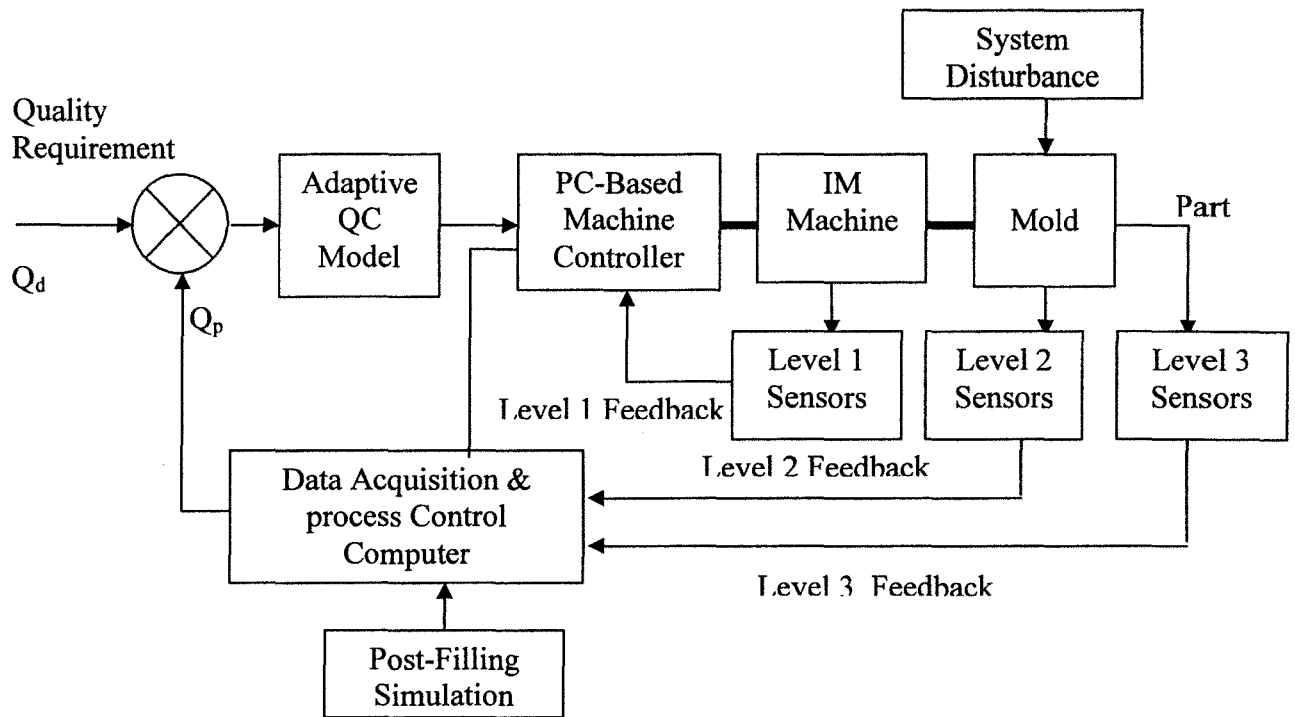


Figure 2.1 Multi-level control architecture (Wang et al., 1999)

2.2.1 Level 1: Machine Variable Control

Machine control (level 1 feedback in Fig 2.1) aims to use advanced controllers and technologies to measure, monitor and control the machine variables in order to keep the machine running smoothly and efficiently, so that higher quality parts and greater profit can be achieved. Injection speed, barrel temperature, mold temperature, filling-to-packing switchover mode, and packing pressure are the variables that have been proven to be closely related to part quality (Havlicsek and Alleyne, 1999; Chang, 2002).

Most modern injection molding machines are equipped with controllers like PLC, PID, and some advanced machines are directly controlled by PCs (Personal computers). Commercial CAE software has been used to assist the process set up. Various empirical models such as design of experiment (DOE) (Johnson, 2002), and expert systems also have been employed to optimize the process parameters. Adaptive control schemes i.e. artificial neural networks (ANNs) (Huang et al., 2004; Wang et al., 2000a; Ouyang et al., 2004), self-tuning regulator (STR) (Yang and Gao, 1999), generalized predictive control (GPC), fuzzy logic control (FLC) (Tsoi and Gao, 1999), and iterative learning control (ILC) (Gao et al., 2001), Multi-model Predictive Controller (Dubay et al., 2007) have been reported for injection velocity control recently. Melt temperature is one of the most important variables of injection molding process, a lot of research works have been done in order to better control the barrel or nozzle temperature. Zhao and Gao (1999) and Wang et al. (2000a) used neural networks to predict the nozzle melt temperature and got a very good result. With those new technologies and algorithms, machine setup and control

is much easier and more efficient, which already bring great benefit to processors. More discussion about advanced control algorithms follows.

2.2.2 Level 2: In-mold Process Variable Control

Machine variable control is not sufficient to keep the production unit maintaining good product quality for a long period depending on the types of disturbances, which affect the polymer melt. Changes in the molding conditions, such as percentage of regrinding material, machine property fluctuation, or environmental temperature changes, may affect the product quality outcomes. On the other hand, to control in-mold process variables, which are regarded as the most accurate indicators of final part quality, will help to minimize those effects and keep the part quality properties more stable. Moreover, in-mold process variables are relatively easy to monitor and control on-line, compared to quality variables. Cavity pressure, nozzle injection pressure, melt temperature, cavity surface temperature, and mold separation, are the interesting process variables usually monitored and controlled in modern process control research. The idea of modern process control is to measure the process variables and determine the relationship with part quality, and then set up the control model to manipulate the related machine variables in order to meet the quality requirement. For example, cavity pressure is regarded as a good predictor for part weight and dimensions, while packing pressure, melt temperature, injection speed and mold temperature affect cavity pressure. Therefore, by adjusting those machines variables, stable and accurate part quality can be met.

The injection molding process must be instrumented and designed such that the key process variables are observable and controllable (Kazmer et al., 2005a). As we know, the polymer state within the mold is crucial to part quality. The relationship between machine variables, measured state variables inside the cavity and part quality variables is very important for process control.

The main challenges for process control are to:

1. Determine the process dynamics relationship between machine parameters and process variables, which are the basis of designing and developing the process controller.
2. Monitor the filling status in the cavity since process variables can never be observed directly. In this case, measuring systems include cavity pressure sensors, cavity temperature sensors and filling sensors. (More discussion in section 2.3)

2.2.2.1. Cavity Pressure Control

Cavity pressure is a useful indicator during injection molding for achieving high part quality and maintaining consistent part weight and overall dimension (Dubay, 2001). The cavity pressure profile is an important parameter to monitor thermoplastic melt filling, packing and cooling stages, which affect not only the melt flow status, but also the molecular orientation. Cavity pressure as an in-mold process variable measured at a specific point in the mold cannot be directly controlled and adjusted; however, some machine variables have a significant effect on cavity pressure. Schenker (1993) concluded that mold temperature and the hold pressure have the most important effects on cavity

pressure and part weight. Kamal et al. (1999) proposed to control part weight by controlling the peak melt pressure and the estimated bulk temperature of the polymer in the mold cavity at the time the gate freezes.

Many researchers have studied using cavity pressure as a process parameter to determine the filling-to-packing switchover time in order to improve part quality. In the injection molding process, filling-to-packing switchover time refers to the time to switch from the mold filling phase to the packing phase, which has higher pressure and much lower filling speed compared to the filling stage. An inappropriate switchover time may cause many quality problems, such as mechanical and dimensional defects (i.e. warpage and weld line strength) and appearance defects (i.e. shrinkage, burn marks, and weld line position). Filling time, ram position, hydraulic pressure, cavity pressure, and melt filling advancement are commonly used to initiate the switchover. Sheth et al. (2001) compared these common transfer methods, and found that the part weight and thickness can be kept more stable and accurate by using cavity pressure transfer. Edwards et al (2003) compared cavity pressure to ram position and ultrasonic sensors (monitor melt filling position) for switchover. They found that ultrasonic sensors for switchover resulted in the highest repeatability in part weights, followed closely by that of cavity pressure. They concluded that switchover by cavity pressure was more easily adjusted, while the ultrasonic sensor installation was non-invasive and non-intrusive, allowing installation without machining the cavity surface.

Other researchers applied some advanced adaptive control technologies for process variable modeling and control. Woll and Cooper (1997) developed a

multivariable nonlinear process model that related process behavior to machine controllable parameters such as filling speed, packing pressure, and cooling time and mold temperature. The control strategy uses an artificial neural network (ANN), embedded within a cascade controller design, to analyze sensor patterns, identify process characteristics, and control part quality. In the cascade design, the machine controller set points of an inner loop are updated based on ANN analysis of mold cavity pressure patterns. Gao et al. (1996a) analyzed cavity pressure dynamics in the filling and packing stages and designed a self-tuning regulator (STR) to control the cavity pressure. They found that the dynamics of cavity pressure were both nonlinear and time varying in relation to the servo-valve opening. Dubay (2001) developed a model-based predictive controller (MPC) and implemented it to control cavity pressure during filling. In the experiments, a Dynisco FT444DH force transducer was mounted behind an ejector, to measure the cavity pressure, and monitor the opening of the hydraulic servo-valve, which was used to control the injection speed. Based on their experiment results, a predictive controller was developed and implemented for controlling cavity pressure for filling phase. The MPC method provides good control for variable cavity pressure ramp set points. It can be developed for any plastic material used and for different injection machines.

Besides the filling and packing stages, research has been done to control the cavity pressure time profile, as opposed to pressure values at specific time points in the cycle, during the cooling stage. Gao et al. (1996b) developed a coolant temperature control system (based on a Dahlin Control algorithm) to control the cavity pressure

profile during the cooling stage. They proposed several alternative variables for pressure control, and *Controlled Pressure Cooling Time (CPTC)* was defined and investigated. Using CPTC as controlled variables and coolant temperature as the manipulated variable, the control system proved to be an effective approach to control cavity pressure during cooling. Pramujati et al. (2006a) designed a predictive controller, using Dynamic Matrix Control (DMC), to control the cavity pressure during the cooling phase by manipulating the coolant flow rate. They used a process time constant, defined as the time from the end of the packing to the time at which the pressure decreases to 63.2% of the pressure at the end of packing as the controlled variable. The results show the shape of cavity pressure profiles during cooling can be controlled effectively and efficiently.

Michaeli et al. (2004) discussed cavity pressure control in both the filling and holding phases and developed an online cavity pressure controller. During the filling phase, a constant gradient of the cavity pressure is realized. While in the holding phase, a constant specific volume was given and a PVT-optimization calculated the required cavity pressure so that the 1-bar-line was always reached, which is based on the PVT (Pressure, Specific Volume, and Temperature) behavior of the polymer. PVT data provides information about the specific volume (volume per unit mass) of a material under different pressure and temperature conditions. It can be used to calculate the material shrinkage between any two pressure-temperature states. Polymeric materials can be characterized through PVT data, where the specific volume can be expressed as a function of the pressure and temperature: $v = v(p, T)$ (Shay et al., 1998). PVT data is typically plotted as isobars (lines of constant pressure); **Figure 2.2** is an example of a

PVT-diagram. It can be used to estimate part volumetric shrinkage by tracing the injection molding process on the diagram as indicated. The PVT data for common commercial polymer materials are available in the database of the major CAE program providers, such as Moldflow®.

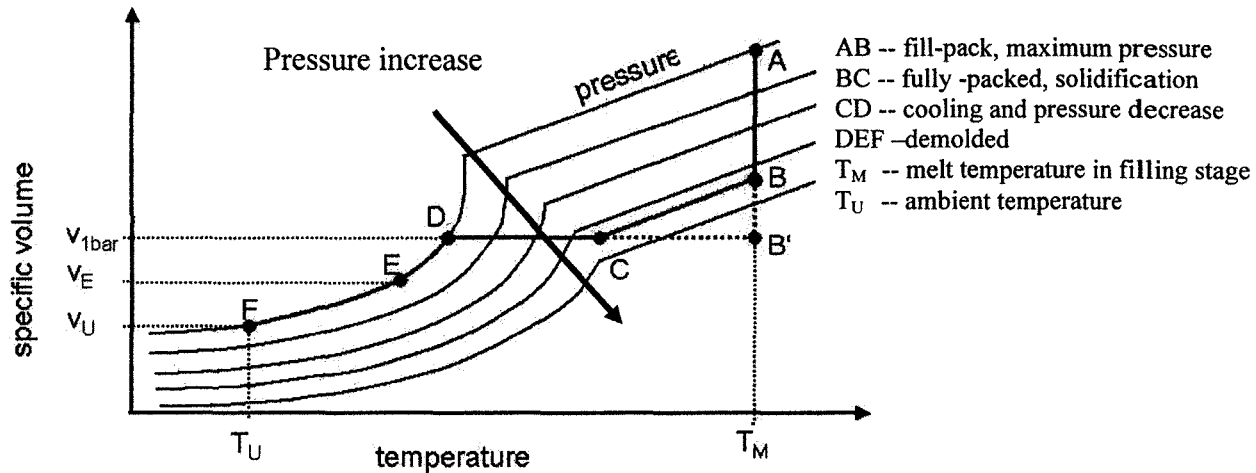


Figure 2.2 an example of PVT-Diagram (Shay et al., 1998).

Previous research works about cavity pressure control have been reported. For instance, Rafizadeh et al. (1996, 1997, 1999a, 1999b) developed a physically-based adaptive controller to control cavity pressure in filling and packing phases. Kazmer and Barkan (1997a, 1997b) investigated the process capability of multi-cavity pressure control in the filling and packing stages.

2.2.2.2. Nozzle Pressure Control

The drawback of using cavity pressure control is the machining necessary for pressure sensor installation, which incurs extra cost and sometimes is not possible due to part design. In that case, nozzle pressure is usually taken as an alternative, to provide information close to the mold cavity pressure. The most attractive is that nozzle pressure control can be employed to different molds in the same machine, without any mold machining problems.

As advanced adaptive control technologies are applied in injection molding, some empirical models have been developed for various specific applications. Kazmer and Petrova (1999) developed a hybrid neural network that combined the training of neural networks with analytical knowledge of the molding process. They compared three different network configurations: conventional, simulation, and hybrid neural networks, with melt temperature, shot size and ram velocity as inputs and nozzle injection pressure as an output. The three networks were trained with the backpropagation-learning rule on randomly chosen process points. The hybrid network showed the ability to predict the pressure more accurately than the other two networks, especially when the number of training data points was limited. Yang and Gao (1999b) applied an ARX model to describe the nonlinear and time-varying dynamics between the servo – valve opening and melt pressure in the nozzle. An adaptive self-tuning controller was then designed and implemented, incorporated with some improvements, like an anti-windup estimate, an adaptive feedforward, and cycle-to-cycle adaptation. The controller works well for different conditions, such as set point profiles, molds, and materials.

Sometimes nozzle pressure and melt temperature transducers are mounted on the machine nozzle for process monitoring. Orzechowski et al. (1998) used a low-cost process monitoring and control system with nozzle-based pressure and temperature transducers for molding process control. They found that temperature and pressure measurements in nozzle could provide effective control of melt quality and shot size uniformity. For example, from the nozzle pressure verse time plotting, it could detect abnormal situations like throat starvation, cold slug in the nozzle and material properties change.

Because of its relative lower cost and easy installation, nozzle pressure control is a good alternative method for pressure control, when cavity pressure control is not appropriate. However, it is still not accurate enough for adaptive control due to the lack of physical dynamic models and the variation of molding machine configuration (e.g. nozzle configuration difference, temperature variation in nozzle tip).

2.2.2.3. Melt Temperature Control

Melt temperature has an important effect on the part quality and has a strong influence on other important process variables such as melt flow rate, melt pressure at the nozzle and in the cavity, and cooling time. Intelligent computation- based methods, such as Dynamic Matrix Control (DMC), and advanced control strategies have been used to model the temperature process dynamics since the 1980s. Zhao and Gao (1999) developed a set of Artificial Neural Networks (ANNs) to predict the effect of operating conditions (screw rotation speed, backpressure, barrel heater temperatures, nozzle heater

temperature, dwell time and injection velocity profile) on the melt temperature during plasticization. A physically-model based simulation was used to generate the temperature profile during the filling phase. In their experiment, the overall prediction of nozzle melt temperature is in good agreement with the measurement. Their modeling provided a basis for process optimization and advanced control. Speight et al. (1995) discussed a unique process parameter referred to as the Pressure Temperature Interactive ‘Viscosity’ Index (PTI), which is derived from both infrared nozzle melt temperature and hydraulic injection pressure. The PTI Interactive ‘Viscosity’ is measured in-line and very sensitive to polymeric material variation independent of polymer rheology. PTI and nozzle melt temperature measurements are shown to be a potential aid to thermoplastic injection molding closed-loop process and statistical process control.

Dubay et al. (2003) presented a strategy for combining computational fluid dynamics (CFD) with model predictive control to characterize the melt process in the extruder. The CFD model includes the screw and barrel using commercial CFD software. CFX-TASCflow was used for the CFD simulation, which generated the melt temperature response. The MPC algorithm is embedded into the CFD source code and able to sample the control move response directly from the CFD transient response. They found that the performance of the MPC controller could be improved by using CFD simulations of the process in order to re-evaluate the controller dynamic matrix. However, the process combining with CFD simulation and MPC modeling is complicated. It is not practical to perform the CFD simulation of the extruder for on-line applications. However, more complete CFD models will be able to capture the rheological effects and non-linear

response of the polymer melt, which are very useful for future MPC development. (Dubay et al., 2003)

However, to obtain actual melt temperature measurements inside the cavity online is not easy; infrared temperature measurement system is one of the common methods. On the other hand, to measure and control the cavity surface temperature is much easier, through temperature sensors installed in the cavity surface. Cavity surface cycle average temperature is the most popular index to reflect the cavity temperature status, which is defined as the average of the part (cavity) surface temperature at a specific location from the beginning to the end of a single cycle. The advantages of using cycle average temperature as the controlled variable are that it indicates the average heat flow during a complete cycle and is relatively immune to process noise and secondly, it is sensitive to the cycle period and the minimal control time unit is one cycle. (Patterson et al., 1990)

Hernandez et al. (2005) and Dubay et al. (2006) developed a PI (Proportional-Integral) controller to control the cooling stage in injection molding, using average part surface temperature as controlled variable and coolant flow rate as manipulated variable. Both the simulation and experimental results presented good closed loop responses. The work provides a method to measure and control the mold cavity temperature in order to achieve effective and efficient cooling system in injection molding process. Hernandez et al. (2007a, 2007b) used an adaptive predictive controller, ASPC (adaptive simplified predictive controller) to control the average cavity temperature. They pointed out that ASPC shows improved closed loop performance when the system was subjected to a set

point change in T_{avg} , compared to non-adaptive predictive controller, SPC (simplified predictive controller) and DMC (Dynamic Matrix Controller).

2.2.2.4. Mold Separation Control

Mold separation (an increase in gap between the mold cavity sections) is not a direct process variable solely related to polymer melt, but it still plays a very important role in connecting the machine variables (i.e. packing pressure, melt temperature and mold temperature) with part quality variables (i.e. part weight, thickness). Mold separation depends on melt viscosity and the integration of melt pressure across the projected area in the sprue, runner, and cavity (Chen and Turng, 2006). Mold separation control shows good potential for industrial applications, due to relatively easy installation, maintenance, flexibility, and relatively low investment cost.

Research work on mold separation control is active recently, because mold separation can be measured outside of the cavity, without the problem of machining and sensor installation. Early research recognized that mold separation was a reliable indicator of part weight and thickness when measured on the order of microns during volumetric filling (Agrawal et al., 1987; Buja, 1990; Wenskus and Miller, 1991). Wenskus and Miller (1991) studied mold separation control for part weight consistency. In their control system, a mold separation measurement signal threshold was used as the fill-to-pack switchover time, instead of the other traditional methods (e.g. ram position, hydraulic pressure, or filling time). Moreover, the threshold value for switchover was adaptively adjusted cycle to cycle. They found that the part weight and thickness consistency

improved, compared to the other traditional switchover methods. Wang et al. (1999) used a linear variable differential transformer (LVDT) mounted along the parting line outside the mold to measure the mold separation. An integrated adaptive injection molding control system was developed to automatically determine the fill-to-pack switchover point and continue to control the mold separation profile by manipulating the hydraulic pressure in the post filling stage. In that case, the disturbances (e.g. uncontrollable material variation or machine-to-machine characteristics) could be adaptively controlled so that part weight consistency could be improved

Chen et al (2002) studied the characteristics of mold separation and its effect on the quality of thin wall injection molded parts compared with conventional injection molding. They installed four linear variable differential transformers (LVDT) across the parting faces of the mold and used a computer based monitoring system to detect the mold separation signals. They investigated the effect of some key machine variables to mold separation and further to part weight and thickness. The machine variables are: melt temperature, packing pressure, clamping force and injection speed.

Chen and Turng (2006) developed an adaptive control system to predict and control part weight by controlling the mold separation profile. The system controls the maximum value of mold separation by manipulating the switchover point from filling to packing in terms of hydraulic pressure from cycle-to-cycle. The hydraulic pressure within a cycle is manipulated to control the mold separation profiles. Their experimental results show that the variation of part weight is reduced significantly as compared to the conventional hydraulic pressure control.

2.2.3 Level 3: Quality Variable Control

Quality control is also a very active research field recently, because of two implementation challenges: complex quality dynamic models and lack of on-line feedback. Quality dynamics, which relate quality variables to process variables or machine variables, are still not understood very well. Most of the quality variables (except part weight and dimensions) are very difficult to monitor either quantitatively or qualitatively on-line with current technology at a reasonable cost. However, both quality dynamic models and on-line feedback are the key points for adaptive quality controller design. In recent research, part weight and part dimensions are commonly used in quality control. Sheth and Nunn (2001) and Kim et al. (2000) investigated the relationship between part weight and a process variable (e.g. cavity pressure), then based on that information, adjusted the machine variables by an adaptive controller. Yang and Gao (2006) developed an online weight prediction model, with three main process variables (packing pressure, barrel temperature, and mold temperature) trajectories as the inputs, using a principal component regression model (PCR). With the enhancement of a nonlinear PCR model, the closed-loop weight control system was designed and tested, and showed good performance over a wide range of operating conditions. They pointed out that this methodology could be extended to the prediction and control of other quality variables. Chen and Turng (2007) developed a direct quality (part weight) feedback control system. The system combines both feedback and feedforward controls and has a cascade structure. The mold separation is controlled and adjusted via both a cycle-to-

cycle mass-based switchover point and a within-cycle holding pressure control. Compared to other control systems currently used in industry, the control system has been proved to be a significant improvement of both long-term and short-term consistencies in part weight.

In general, the level 3 feedback loop uses a quality dynamic function, related to lower level (level 1 or 2) variables, and then uses advanced control systems to manipulate the machine variables.

2.3 Measurement Instrumentation

The development of instrumentation technologies is the basis of process control, because most process variables have to be measured by some type of device. Through the installation of sensors (e.g. pressure, temperature and filling), on-line data can be collected and sent to an advanced adaptive controller. After analysis, the information will be fed back to the machine actuators to adjust the machine settings in order to get good quality parts, which meet customers' standards in both visual and functional requirements. The most commonly used sensors (**Appendix A**) in plastic injection molding process control are discussed in this section.

2.3.1 Cavity Pressure Sensors

Cavity pressure is an important variable that correlates directly with part quality variables such as part weight, dimensions, flash, shrinkage and so on. Continuous

monitoring and control of the cavity pressure is viewed as the most accurate method for producing consistent quality and low cost parts.

Today, there are two common technologies used in cavity pressure sensors. One uses strain gauge technology, which uses strain gauges mounted to a diaphragm or column to convert force exerted by the plastic melt into voltage output. (Groleau and Groleay, 2002) The second type of sensors, piezoelectric sensors, measure the charge generated on piezoelectric crystals for an applied stress or load. Even though these two types of sensors work quite differently from one another, they can all give robust and accurate readings that are able to meet customer requirements. Both technologies are widely accepted and applied in the industry.

Currently, there are two main styles of mounting methods used in pressure sensors, *direct* (also known as flush mount), and *indirect* (as slide or button configurations). Indirect sensors sense the force exerted by plastic on an ejector or stationary pin, which transmits the force to the sensor. The slide style sensors are commonly used as they are easily removed and relocated, but it can also slip out either fully or partially and cause erroneous readings (Vaughan, 2003). Button style sensors are more reliable, compared to slide sensors. Flush mount sensors, in a direct measurement style, are directly installed close to the cavity by machining a hole behind the cavity. The advantage of this sensor is that it comes in contact with the polymer melt material and measures pressure directly without a transmission aid, for example an ejector, which transmits melt pressure to a pressure sensor installed underneath. In some cases, indirect measurement is recommended, especially in high temperature application, or in which

there are installation problems for direct mounting, such as not enough room for a direct sensor installation, or in a special cavity surface requirement (e.g. a mirror surface). On the other hand, indirect measurement also suffers from many adverse conditions such as friction, bending of the ejector pin, platen distortion and so on, which will have a detrimental effect on the measurement. In general, direct measurement is more desirable for industrial processes.

Currently, as the technologies developed, many measurement instrumentation suppliers like RJG, Kistler, Priamus and Dynisco provide many advanced pressure sensors for different applications. Meanwhile, researchers are working with applying other pressure sensors into injection molding process monitoring, such as wireless pressure sensors. (Kazmer et al., 2003b; 2005b)

2.3.2 Temperature Sensors

Besides cavity pressure sensors, temperature sensors also play an important part in the qualitative assessment of the injection molding process. The cavity temperature change is related to final part quality such as weight, shrinkage and residual stress, quite similar to cavity pressure sensor control, but with lower investment. In some situations, we can use temperature sensors to monitor and control the process. On the other hand, temperature sensors have their own limitations: temperature sensors are generally not as accurate recording melt temperatures as pressure sensors are reading melt pressure (Kazmer et al., 2005a). Since the temperature sensors are embedded in mold steel, heat transfer from the sensor head to the surrounding metal is unavoidable and considered as

source of measurement noise. In addition, temperature sensors may have a significant phase lag and steady state error in the measurement of melt temperature.

The most commonly used temperature thermocouples in injection molding are surface mounted sensors. The sensor head can be as small as 1.0 mm and can be embedded in the cavity. Thermocouple types J, K and N are most commonly used. Thermocouple type N is more accurate and used compared to the others, because of its high stability and resistance to high temperature oxidation, and enhanced calibration ability and longevity (Burley, 2007).

2.3.3 Fill Sensors

Compared to the wide research and application with cavity pressure and temperature sensors, the measuring transducers used to monitor the melt flow state in cavity are not as well developed. Cavity pressure and temperature sensors cannot directly capture the flow front status, which requires knowledge of the melt-front velocity and melt-front position during the filling stage (Chen et al., 2004b). There are two measuring systems used in melt filling monitoring: optical sensors and on-line ultrasonic technology.

An optical sensor using fiber optics to access the mold cavity can be used to monitor the mold filling and cooling phases. Fluorescent dye was mixed with resin at low levels of concentration (~10 parts per million by weight) prior to processing (Bur and Thomas, 1997). The measured optical signal transmits through the resin, reflects off the opposite wall of the mold, and retraces its path through the resin to the optical sensor.

Using that sensor, the melt status during packing and cooling phases of the process can be monitored and process parameters can be optimized to achieve a good quality part.

Ultrasonic transducers used in injection molding have been widely studied for such variables as flow front advancement during the filling stage; gap development caused by the shrinkage of the part during the cooling (mold separation); and material solidification, which significantly affect part quality (Wang et al., 1997). The ultrasonic transducer is installed on the external surface of the mold; a sound pulse will strike a surface of the mold cavity and reflect back to the transducer. By measuring the change in intensity of the reflected echoes, the signal indicates the presence of polymer at that point. An ultrasonic transducer was used to control the injection phase and packing phase switchover by detecting the arrival of the melt at a specific position of the mold cavity (Edwards et al., 2001b). The main advantages of ultrasonic transducers are: 1. the ultrasonic sensors are attached to the external surface of the mold, which save a lot in engineering design and machining; and 2. the ultrasonic pulse-echo technique can directly sense the position of the polymer in the mold cavity. Thus one can detect the flow front arrival, end of filling, degree of solidification, detachment of the part and so on (Wen et al., 1999).

Another type of instrumentation used to measure material filling and cooling status in the mold cavity is a Linear Variable Differential Transformer (LVDT). Wang et al. (1999) used this transformer to measure the mold separation. LVDT is a common type of electromechanical transducer that can convert the rectilinear motion of an object to which is coupled mechanically into a corresponding electrical signal. LVDT can measure

movements as small as millionths of an inch, which is accurate enough for injection molding process control. It provides a measuring range from 2.5mm to 100mm, and linearity error as small as $\pm 0.1\%$ of full scale output. The applications of LVDT in injection molding are mostly focused on high-speed injection molding (Chen and Turng, 2005a, 2006, 2007).

2.4 Advanced Control Algorithms

Many advanced control algorithms have been employed in the closed-loop injection molding process control such as fuzzy control (He et al., 1998; Lau et al., 1999; Vagelatos et al., 2001), neural networks (Ribeiro, 1999; Kening et al., 2002; Zheng and Chen, 2005), genetic algorithm (Kim et al., 1996; Gao et al., 1999) and iterative learning control algorithm (Gao et al., 2001; Liang et al., 2002; Wang et al., 2002; Tan et al., 2003). Their applications cover not only in machine variables control, but also in in-mold process variables control. The basic idea of advanced control algorithms is to control process variables or quality variables, through studying process status and available past information. Among them, artificial neural networks, self-tuning regulator and model predictive control are commonly used and proved to be most attractive for injection molding process control.

2.4.1 Artificial Neural Networks

Artificial Neural Network is an information-processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. It is

composed of a large number of highly interconnected processing elements (nodes) working in unison to solve specific problems. ANN is configured for a specific application, such as pattern recognition or data classification, through a training process. The key advantage of ANNs is their remarkable ability to derive meaning from complicated or imprecise data, which can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. Other advantages such as adaptive learning, self-organization and real time operation are also recognized.

The process elements, or nodes, are arranged in a layered structure. Each node consists of a summing junction, which adds together the weighted inputs from other nodes, and an activation function, which generates an output, and fans out to serve as an input to other nodes within the network structure. Nodes transmit signals to each other via weighted links, which attenuate or amplify the signal depending on the weight value (Stergiou and Siganos, 1996).

A feed forward neural network topology, **Figure 2.3**, is one of the most popular ANN structures (Rewal et al., 1998). The network configuration is termed 6-3-2, which means six inputs, three layers (input, hidden layer and output) and two outputs. Its function mapping is achieved by adjusting the weights of internal network connections such that, a given input, the difference between the network response and the desired output is an acceptable value. In order to determine the parameters of the function, 'training' the network is the most important step. If the training set is proper and a good representation of the problem space, then the resultant network is capable of capturing

inherent relationships in the data (Rewal et al., 1998). The back propagation network (BPN) is a very popular training algorithm first proposed by Rumelhart et al (1986). The error term between the network output and desired response is propagated backwards to adjust the internal weights of each node so that the new error term calculated on the forward pass is reduced. Normally, besides the training set, a validation set and a test set are necessary in order to achieve the best generation.

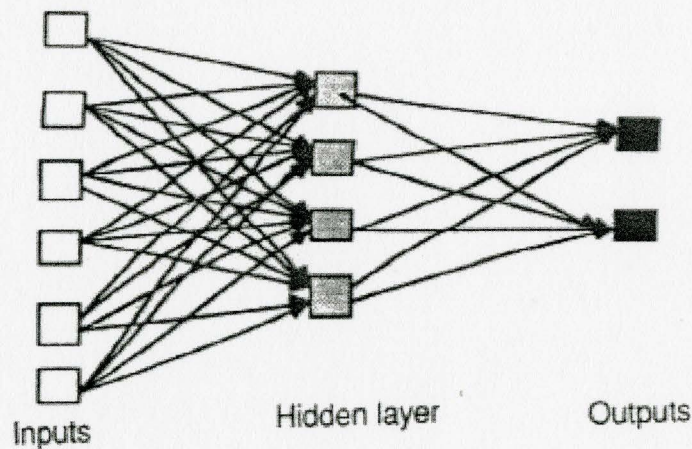


Figure 2.3 Topology of a three-layered feedforward neural network. (Rewal et al., 1998)

Researchers have developed many ANN systems for cavity pressure control and temperature control in injection molding process control (Woll et al., 1996; Rewal et al., 1998; Kazmer and Petrova, 1999; Fara et al., 2001). Part weight and cavity (nozzle) pressure are the most commonly used variables set for ANN systems development. Back propagation is the most popular training technique for ANN systems.

2.4.2 Self-Tuning Regulator

A self-tuning regulator is a controller that adaptively finds its control parameters in the control law. It is a class of adaptive controller, which is used when the process to be controlled has a constant but unknown parameter. The controller design procedure is:

1. Find specifications for the closed-loop system.
2. Determine a model for the process to be controlled.
3. Choose a design method.
4. Calculate the parameters in the controller.

In many cases it is desirable to automate these steps and that is the idea behind adaptive and self-tuning regulators. (STRs) (Wittenmark, 1998) The main advantage of using STRs as adaptive controller lies in the large number of combinations of schemes that can be implemented based on the needs of the plant. STRs are commonly used in the processes with long delay time, time-varying disturbances.

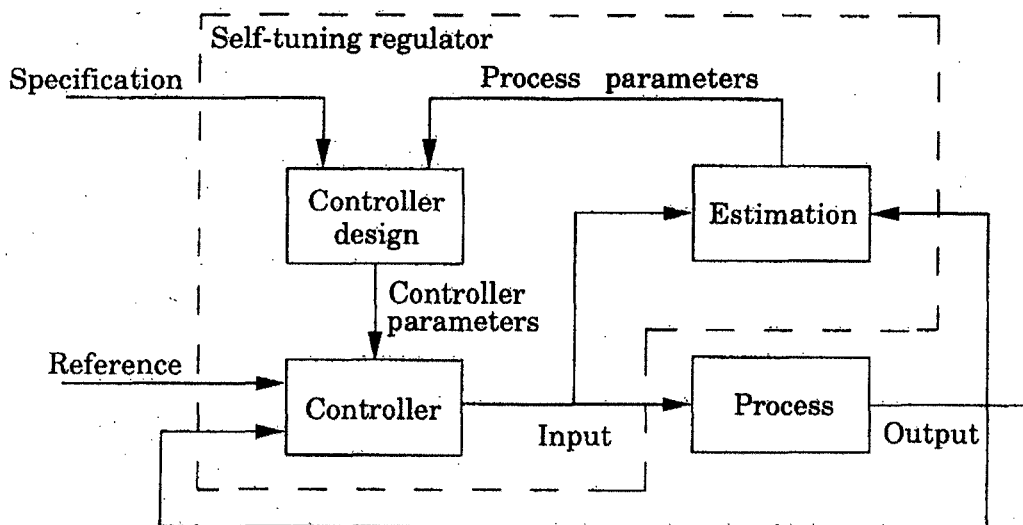


Figure 2.4 Block diagram of a self-tuning regulator. (Wittenmark, 1998)

The structure of the self-tuning controller is best described from the block diagram in **Figure 2.4**. It consists of two closed loops. The first loop is a conventional controller feedback-loop consisting of the process and the controller where the output of the process is measured and compared with the desired output (reference signal) of the closed-loop system. The second loop, the updating loop, has two main blocks, the estimator and the controller design. Estimator can make an estimate of the parameters in process model, which will be used to determine the parameters of the controller design. The process model is normal assumed as a single input and single output system. There are many different methods to calculate the parameters, such as the Least Squares algorithm. Controller design uses the process model and the specification to determine the controller parameters that are then sent to the controller.

Gao et al (1994, 1996a) developed a self-tuning regulator to control the cavity pressure so that it follows a preset profile through manipulation of the servo-valve opening. They found a cavity pressure model in relation to the servo-valve opening. Their experiment showed that the methods worked very well under a wide range of process conditions.

2.4.3 Model Predictive Control (MPC)

Model predictive control (MPC) is a family of control techniques that optimize a given criterion by using a process model to predict system evolution and compute a sequence of future control actions (Guzmer et al., 2005). In order to develop a MPC controller, the most important step is to extract or identify a reasonably accurate dynamic

model of the process, which captures the dynamic and static interactions between inputs, outputs, and disturbance variables (Seborg et al. 2003). Process models can be developed directly from process test data by using system identification methods to create an empirical model, the accuracy of which depends on the design and collection of the process test data. The other method to achieve the dynamic model is to develop the model from first-principles or deterministic type models. First-principles models are typically more expensive to develop, but are able to predict process behavior over a much wider range of operating conditions after validation. While the empirical models take less time to develop, sometimes they may not be accurate enough for process control. In reality, process models used in MPC technology are based on an effective combination of process data and theory. For example, the key parameters in first-principles models are typically estimated by process test data (Qin and Badgwell, 2002). In addition, if the process is subject to disturbances, a disturbance or noise model can be added to the process model, thus allowing the effect of disturbances on the predicted process output to be taken into account (Astom and Wittenmark, 1990).

In MPC applications, the output variables are referred as controlled variables (CVs), while the input variables are called manipulated variables (MVs), and measured disturbance variables are called DVs. Figure 2.5 is the general structure of MPC algorithm (Seborg et al. 2003). A process model is used to predict the current plant outputs. The difference between the process outputs and model outputs, called residuals, serve as the feedback signal to a prediction block. The Prediction block generates the predicted outputs, which are then used by set-point calculations and control calculations

at each sampling instant to calculate manipulated inputs (MVs). The set-points for control calculations, so called targets, are calculated from an economic optimization based on a steady-state model of the process. The control calculations are based on current measurements and predicted outputs. Moreover, the objective of the MPC control calculations is to determine a sequence of control moves (manipulated inputs changes) so that the predicted response moves to the set point in an optimal manner (Seborg et al., 2003). MPC accepts a variety of models (physical models and empirical models, linear and nonlinear), objective functions, and constraints (the upper and lower limits, or the maximum moving step size of CVs and MVs). Multivariable dynamic matrix control (DMC) algorithm is a commonly used model predictive control algorithms (Marlin, 2000).

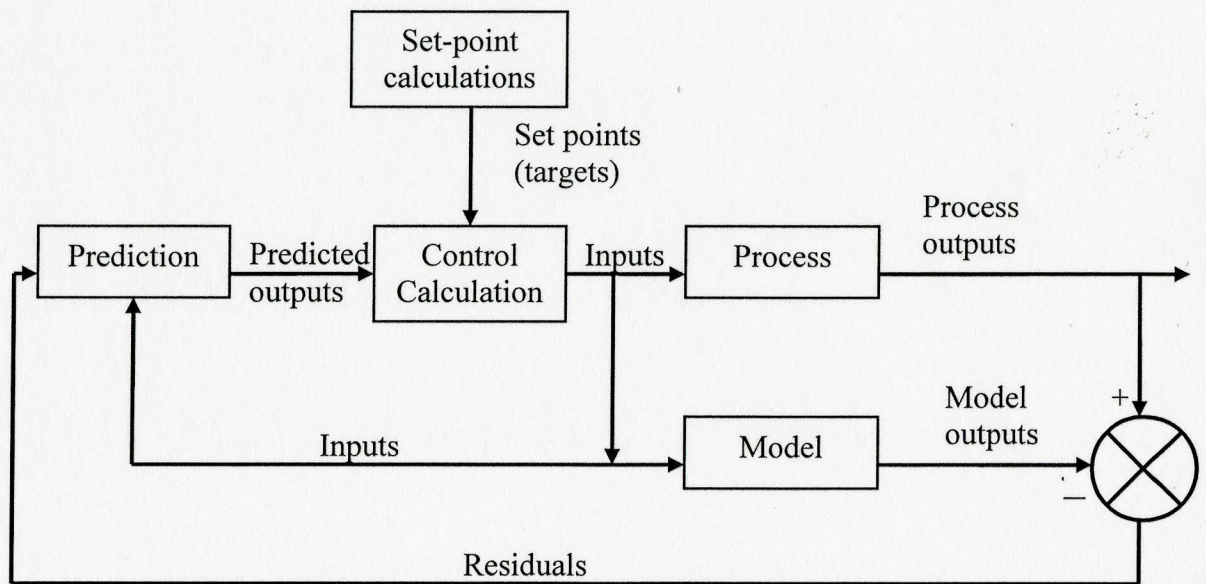


Figure 2.5: Block diagram for model predictive control. (Seborg et al. 2003)

Moreover, MPC can be used in a process with multivariable and long delay times (dead time), which is very meaningful and attractive for injection molding process control. The main advantage of MPC is ease of tuning and implementation. It can also be used for non-linear processes. MPC inherently provides a feed-forward action for compensation of the measurable disturbances and for tracking a reference trajectory (Ambady and Kazmer, 2001).

DMC was developed by Cutler and Ramaker (1979), and since then has been successfully applied to different industrial processes. The Dynamic Matrix Control algorithm can be used to develop single-variable (Single-input and Single-output control systems, SISO) and multivariable (Multi-input and Multi-output, MIMO) control systems. Basically, the task of the control algorithm is to determine future adjustments to the manipulated variable that will result in the predicted controlled variable returning quickly to the set point (Marlin, 2000). Data of past inputs and outputs must be collected for model identification, which is the estimation of parameters for a model that can be used to predict outputs, given inputs. Then the output response to a unit input, named as the step weight, can be figured out by using a transfer function. It is very important for controller design to choose the key parameters for DMC algorithm such as the sampling time, sample size required for process model to reach steady state, the controlled-variable horizon, and the manipulated-variable horizon. Moreover, the relative values of the two tuning parameters: the weighting for each controlled variable and the weighting for each manipulated variable, determine how much importance is placed on the controlled

variable and on the variability of the manipulated variables. Adding feedback and tuning helps improve the performance of the DMC controller.

It is reported that there were more than 4600 total MPC applications (two times of the number 5 years earlier) in different industries such as chemical, polymers, refining, petrochemicals and so on (Qin and Badgwell, 2002). MPC has been employed for injection molding process control, specifically, in machine variables control (level one) and process variables control (level two). Ambady and Kazmer (2001) used MPC for dynamically tracking and controlling the mold wall temperature profile in injection molding by regulating the preheat time, preheat temperature, and coolant temperature. Dubay (2001) used MPC to control the cavity pressure during the filling phase. Pramujati et al. (2006a, b) designed a DMC to control the cavity pressure during cooling phase by manipulating the coolant flow rate. Generalized predictive control (GPC) (Lee and Sullivan, 1988) is a popular MPC technique. Dubay et al. (1997) and Yao and Gao (2005) employed a MIMO GPC for melt temperature control. The experiments results showed GPC is effective in controlling barrel temperature, with significantly reduced variations and overshoots compared to a well-tuned PID controller.

2.5 CAE Simulation Programs Applications

Due to the more complex product design and the demands for higher part quality, traditional trial-and-error methods can not satisfy modern industrial requirements. Using computer aided engineering (CAE) programs for process simulation and analysis is a

necessity for the injection molding industry. Many CAE programs are applied in injection molding industry such as Moldflow[®], Modex3D[®] and ANSYS[®] (Xu and Kazmer, 1999).

2.5.1 Molding Process Analysis Programs

CAE programs now can simulate the melt behavior: flow and heat transfer, inside the mold cavity, and then predict the part quality like weld lines, shrinkage and so on. Possible defects such as inability to fill a cavity, dimensions over limit and warpage can be addressed by simulation before mold making, which will be used to modify the mold design (Wang et al., 2005). CAE programs also help process engineers to optimize the processing windows and solve quality problems (Kalnin and Zluhan, 1999; Wang and Yoon, 1999). Thanks to its improved accuracy, researchers have used CAE programs to collect process test data instead of in-line production data, in order to identify the process model and develop the modern process controllers. Based on the CAE simulation results, Wang et al. (2005) used ANN technique to build a relationship model between quality index (sink-mark index) and selected process conditions, and then developed a controller to control the part weight. The controllers were trained with CAE results and the simulation results showed reasonably agreement with experiment result. Chen and Turng (2005a) extracted neural network models from CAE predictions and developed an online quality controller to control part weight.

However, the simulation results are not always satisfied for real-time application, since sometimes there is large difference from measured results. Ainoya and Amano (2001) and Koelling and Xu (2003) pointed out the large discrepancies between

simulation and observed results that resulted from neglecting some important factors, such as the pressure –dependent viscosity, variable heat transfer coefficient, heat capacity, and juncture pressure loss. Ainoya et al. (2001) concluded that the CAE simulation program is sufficiently reliable to predict the static state of molding with accuracy, if high quality material databases can be employed.

2.5.2 Mold Surface Temperature Transient Simulation

Even though current CAE programs of polymer processing contributed a lot in mold design and process optimization, there are still some limitations in applications. For instance, it is not routine to predict the transient mold temperature cycle-to-cycle using current injection molding software. Cooling simulation is usually conducted with a cycle-averaged approach and one spatial dimensional temperature transient variation (e.g. Chen et al., 2004).

Patterson et al. (1990) studied mold temperature control and discussed the tendency of temperature variation of mold at start-up period and the response of mold surface temperature variation for a step change in coolant temperature through experiments. Chen and Chung (1992) used simulation programs to study mold surface temperature transients. They applied a modified boundary element technique (finite difference method) to calculate the cyclic transient mold cavity surface temperature. **Figure 2.6** shows the schematic of temperature variation of the mold (Chen et al., 1999).

Since the end of the 1990s, researchers have applied commercial FEM (Finite Element Method) programs to simulate the mold temperature distribution, such as

ANSYS® (Xu and Kazmer, 1999; Chen et al., 1999, 2000, 2004, 2005, 2007.) and LUSAS Analyst (Tang et al., 2005). In order to simulate the continuous production, the mold temperature distribution was used as the initial conditions for the next simulation cycle until the temperature distribution reached steady-state. They compared the cooling time estimates of 22 cycles and found the differences in the first few cycles were very prominent, but after that, no large differences were observed. However, due to the large computing work, their simulation was based on 2D and an axis-symmetric simple part. As computer software and hardware developed recently, three-dimensional simulation of transient temperature distribution is reported. Chen et al (2004) simulate the transient temperature distribution for a lens mold with embedded heaters. They measured the mold surface temperature distribution online with an infrared thermal image system. The simulation results of mold cavity side shows well coincidence with measured results, while that of the core side shows reasonable coincidence but slightly larger difference. They analyzed that the complicated structure and ejector system in the core may lead to this problem. It shows that ANSYS® can be used to simulate the thermal process in injection molding process and the simulation results are valuable for mold design and process control.

Finite element software is capable of simulating the transient temperature distribution and the result is in reasonable agreement with experimental result. The simulation software will be a very helpful tool to investigate the dynamic response of mold surface temperature when cooling conditions or melt temperature is changed.

Through the simulation studies, proper mold temperature control systems and cooling time can be designed.

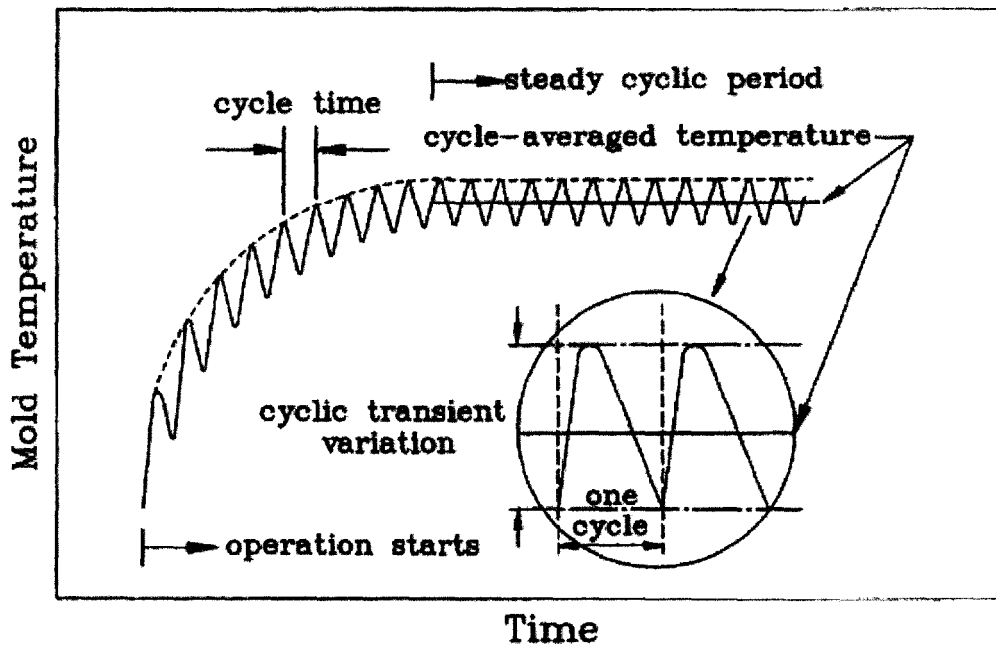


Figure 2.6: Schematic of temperature variation of mold. (Chen et al., 1999)

Chapter 3

Model-Based Predictive Controller Development

3.1 Overview

Model-Based Predictive Control (MPC) has been extensively used in different industries, thanks to its superior advantages in process control. Researchers have implemented MPC control in thermoplastics injection molding process. **Table 3.1** lists the recent research using advanced controllers for injection molding process control. The investigations include the injection molding machine variables (e.g. injection velocity, barrel temperature, nozzle pressure) and in-mold process variables (e.g. cavity temperature, cavity pressure, and mold separation). The manipulated variables (such as hydraulic valve opening, heating zone output, coolant temperature and coolant flow rate) are adjusted and controlled cycle-to-cycle or in-cycle, based on the control objective. Single-Input/Single-Output (SISO) and Multiple-Input/Multiple-Output (MIMO) MPC controllers have been developed. The MPC controllers demonstrate significant improvement for injection molding process control, compared to other traditional control algorithms (e.g. PID, PLC).

Table 3.1 Recent research using model predictive controller for molding process

In-cycle or cycle-to-cycle	Reference	Controlled Variables	Manipulated Variables	Control Strategy
In-cycle	(Dubay, 2001)	Cavity pressure in filling	injection speed (hydraulic servo valve)	SISO MPC
In-cycle	(Pramujati et al., 2005)	Injection velocity	Hydraulic servo valve opening	Simplified predictive controller
In-cycle	(Gao et al., 1996a)	Cavity pressure in filling and packing	Hydraulic servo valve opening	Self-tuning control
In-cycle	(Dubay et al., 2007)	Injection velocity	Hydraulic servo valve opening	SPC and MMPC (Multimodel Predictive Control)
In-cycle	(Lakhram and Dubay 2002)	Injection velocity	Hydraulic servo valve opening	Self-optimizing MPC
In-cycle	(Yang and Gao., 2001)	Injection filling velocity	Hydraulic servo valve opening	PI; STR; GPC; Fuzzy control
In-cycle	Rafizadeh et al., 1997	Cavity pressure in filling	Hydraulic servo valve opening	Physically based adaptive control
In-cycle	Yang et al., 1998	Nozzle melt packing pressure	Hydraulic servo valve opening	SISO & MIMO MPC

(Continued)

(Continue)

In-cycle or cycle-to-cycle	Researchers	Controlled Variables	Manipulated Variables	Control Strategy
Cycle-to-cycle	(Pramujati et al., 2006a)	Cavity pressure after packing	Coolant flow rate	SPC and DMC
Cycle-to-cycle	(Ambady et al., 2001)	Mold wall temperature	Heat gas preheat time & temperature, coolant temperature	SISO MPC
Cycle-to-cycle	(Hernandez et al., 2005)	Average cavity surface temperature	Coolant flow rate	PI controller
Cycle-to-cycle	(Gao et al., 1996)	Cavity pressure in cooling	Coolant temperature	Dahlin Controller
Cycle-to-cycle	(Pramujati et al., 2007)	Cavity pressure in cooling	Coolant flow rate	Adaptive simplified predictive controller
Cycle-to-cycle	(Hernandez et al., 2007)	Average part temperature	Coolant flow rate	Adaptive simplified predictive controller
Cycle-to-cycle	(Abu and Dubay, 2007)	Injection speed, barrel temperature	Hydraulic servo valve opening Heating band output	Extended predictive controller
Cycle-to-cycle	(Haber and Kamal, 1987)	Peak cavity pressure	Hydraulic pressure	Not specific
Cycle-to-cycle	(Patterson et al., 1990)	Cycle average surface temperature	coolant temperature	PI, PID, Dahlin controller

(Continue)

In-cycle or cycle-to-cycle	Researchers	Controlled Variables	Manipulated Variables	Control Strategy
In-cycle and Cycle-to-cycle	(Yang and Gao, 1999b)	Nozzle pressure	Hydraulic servo valve opening	Adaptive control
In-cycle and Cycle-to-cycle	(Chen and Turng, 2006)	Mold separation	Switch point Hydraulic pressure	PID-based adaptive control
Continuous	(Dubay , 2002)	Barrel temperature	Heating zone output	MIMO MPC
Continuous	(Dubay et al., 1997)	Barrel temperature	Heating zone output	MIMO MPC
Continuous	(Gerber et al., 2006)	Barrel temperature	Heating zone output	MIMO MPC
Continuous	(Pramujati and Dubay, 2003)	Barrel temperature	Heating zones 1,2,3 output	MIMO MPC
Continuous	(Tsai et al., 1995)	Barrel temperature	Heating zone output	Multivariable Self-tuning Controller
Continuous	(Yao and Gao, 2005)	Barrel temperature	heater output	Adaptive GPC

Recently, Matlab[®] has been used a lot for process dynamic model identification and model predictive controller development. Hernandez et al. (2005) used Matlab[®] to implement the cavity temperature PI controller. Chen and Turng (2005) used Simulink[®] to construct and simulate their multilevel process and quality control system. Dubay (2001) used System Identification Toolbox to identify an auto-regressive with exogenous input (ARX) process model for cavity pressure MPC control. MPC Toolbox provides windows-based graphical user interface and predefined functional blocks, which can be used to develop the control system conveniently. A MPC program will be developed based on the MPC Toolbox in this study.

In order to develop a MPC controller, to find a proper plant dynamic model is the first and most important step. In this study, off-line models are identified from CAE simulation results, instead of real machine production data. Nevertheless, the accuracy of the plant model depends on how accurate the CAE program can simulate the process. Two different CAE programs are used to simulate the process. Moldflow Plastics Insight, injection molding process simulation software is used to simulate cavity pressure and temperature of the polymer melt. FEM (finite element method) simulation software, ANSYS[®], is used to simulate the mold temperature distribution and transient in cooling stage.

3.2 Controller Structure Selection

3.2.1 Multivariable Process Control

Among the main process variables, cavity pressure has been often used as a controlled variable. Cavity pressure profiles can be divided into three stages (filling, packing/holding and cooling). Research has been done to control each stage of injection molding either cycle-to-cycle or in-cycle using SISO MPC controllers. Cavity surface temperature has also been reported to be controlled cycle-to-cycle by a SISO MPC Controller (Hernandez et al., 2007). Recent studies with MIMO MPC controllers are mainly focused on controlling zone temperatures in the barrel as a continuous control variable. Here, we investigate the controllability of a two-by-two MPC set-point controller to control the cavity pressure and part surface temperature in a cycle. SISO controllers are developed to control cavity pressure in the filling stage and mold surface temperature after filling in order to show controllability.

Among MPC controllers, Dynamic Matrix Control is the most popular strategy in industrial process control, due to easy tuning and good performance. The DMC algorithm can be used to control process with dead time, input and output constraints, and unusual dynamic behavior. Normally, the dynamic matrix is developed from experimental sampled data by step input excitations. The DMC algorithm can be used as SISO or MIMO process control. Since it can be designed within the model predictive control structure, and the algorithm can be designed without determining the analytical inverse of the process model, the extension to multivariable systems is straightforward (Marlin, 2000).

The DMC controller uses a dynamic model of the process to calculate the future behavior of the controller variable that would occur without future control adjustments. Common used models in MPC are the finite step and impulse response models. The following discussion on multivariable model predictive control is based on summaries in Swartz (1991, 2004). A detailed discussion of MPC may be focused in Maciejowski (2002).

The formulation for single-input and single-output (SISO) systems is described first; therefore its extension to MIMO systems is discussed.

The general form of finite step response model is given by

$$y(k) = \sum_{i=1}^N [s(i)\Delta u(k-i)] + s(N)u(k-N) + d(k) \quad (3.1)$$

where $y(k)$ represents the output at time step k ; u and Δu represent the input and the input change respectively; $s(i)$ are the unit step response coefficients; and $d(k)$ represents unmodelled disturbances. N represents the number of time steps for the system to reach steady state. The general form can be used to predict the response for a set of future inputs:

$$\hat{y}(k+l|k) = \underbrace{\hat{y}_f(k+l|k)}_{\text{effect of past inputs}} + \underbrace{\sum_{i=1}^l s(i)\Delta \hat{u}(k+l-i|k)}_{\text{effect of future inputs}} + \hat{d}(k+l|k) \quad (3.2)$$

where $\hat{y}(k+l|k)$ represents the model output at time step $k+l$, based on the information available at step k . $\hat{y}_f(k+l|k)$ may be computed from the step response model as:

$$\hat{y}_f(k+l|k) = \sum_{i=l+1}^{N-1} [s(i)\Delta u(k+l-i)] + s(N)u(k+l-N) \quad (3.3)$$

Using the step response model to predict the output to P time steps into the future gives an expression of the following form: $y = A \Delta u + b$. Here y is a vector of future output predictions; Δu is a vector of future input moves; b represents the effect of past inputs and predicted disturbances on the outputs; A is named as dynamic matrix. The equation can be written in matrix form as below:

$$\begin{aligned}
& \underbrace{\begin{bmatrix} \hat{y}(k+1|k) \\ \hat{y}(k+2|k) \\ \vdots \\ \hat{y}(k+M|k) \\ \vdots \\ \hat{y}(k+P|k) \end{bmatrix}}_y = \underbrace{\begin{bmatrix} s(1) & 0 & \dots & 0 \\ s(2) & s(1) & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ s(M) & \vdots & \dots & s(1) \\ \vdots & \vdots & \dots & \vdots \\ s(P) & s(P-1) & \dots & s(P-M+1) \end{bmatrix}}_A \underbrace{\begin{bmatrix} \Delta \hat{u}(k|k) \\ \Delta \hat{u}(k+1|k) \\ \vdots \\ \vdots \\ \Delta \hat{u}(k+M-1|k) \end{bmatrix}}_{\Delta u} \\
& + \underbrace{\begin{bmatrix} \hat{y}_f(k+1|k) \\ \hat{y}_f(k+2|k) \\ \vdots \\ \vdots \\ \hat{y}_f(k+P|k) \end{bmatrix}}_b + \underbrace{\begin{bmatrix} \hat{d}(k+1|k) \\ \hat{d}(k+2|k) \\ \vdots \\ \vdots \\ \hat{d}(k+P|k) \end{bmatrix}}_b
\end{aligned}$$

(3.4)

The objective of the MPC algorithm is to calculate a sequence of control moves (manipulated input changes, Δu) so that the predicted response output (y) moves to the set point (y_s) in an optimal manner. **Figure 3.1** shows the basic concept for MPC control (Seborg et al. 2003). To calculate the manipulated moves, the common method is to minimize a performance-based objective function. For a prediction horizon of P and a control horizon of M , the on-line solution of open-loop optimization can be expressed as:

$$\begin{aligned}
& \text{Minimize } \Phi = (y_s - y)^T Q (y_s - y) + \Delta u^T \Lambda \Delta u \quad (3.5) \\
& \Delta u \\
& \text{Subject to } y = A \Delta u + b
\end{aligned}$$

where $Q = \text{diag}(q_1, \dots, q_p)$ is weighting matrix for outputs and $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_M)$ is weighting matrix for inputs, usually chosen to be diagonal.

The solution to (3.5) is given by:

$$\Delta u = (A^T Q A + \Lambda)^{-1} A^T Q (y_s - b) \quad (3.6)$$

In practice, only the first input move is implemented. The next output measurement is used to estimate the disturbance. MPC controller tuning may be accomplished through the weighting matrices Q and Λ , and through the prediction and control move horizons, P and M . Discussion about the effects of the tuning parameters on closed-loop stability can be found on Swartz (1991) and Seborg et al. (2003). The disturbance estimate is commonly assumed to be constant over the prediction horizon.

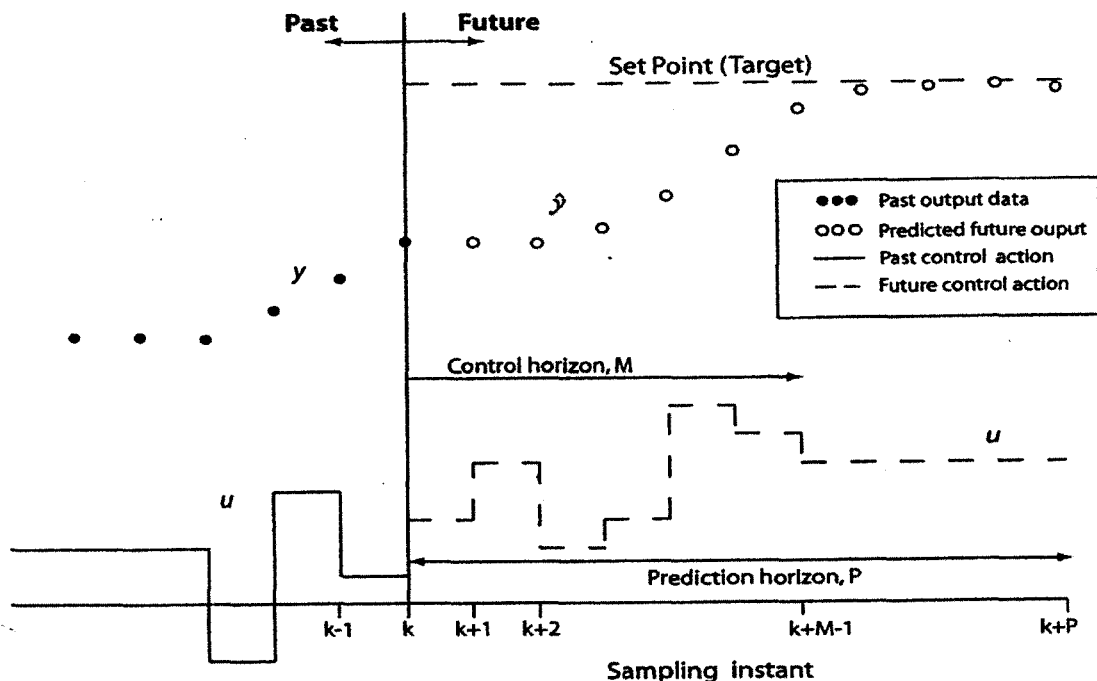
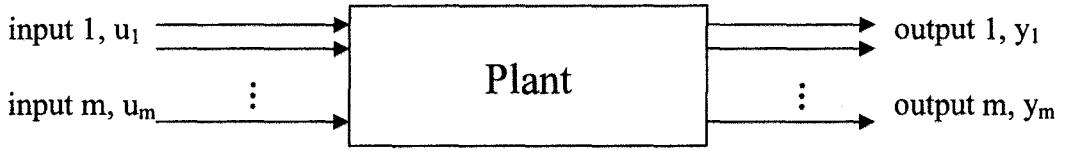


Figure 3.1 Basic concept for model predictive control (Seborg et al. 2003)

For MIMO systems:



Each output is in general influence by each input. The step response model for output y_i can be expressed as:

$$y_i(k) = \sum_{j=1}^m \left[\sum_{l=1}^{N-1} s_{ij}(l) \Delta u_j(k-l) \right] + \sum_{j=1}^m [s_{ij}(N) u_j(k-N)] \quad (3.7)$$

where $s_{ij}(l)$ is the response at time step l of the output i to a unit step in the j^{th} input.

Similar as SISO system, the prediction equation is $y = A \Delta u + b$, where

$$y = \begin{bmatrix} \hat{y}_1(k+1|k) \\ \vdots \\ \hat{y}_1(k+M-1|k) \\ \dots \\ \vdots \\ \dots \\ \hat{y}_m(k+1|k) \\ \vdots \\ \hat{y}_m(k+M-1|k) \end{bmatrix} \quad (3.8)$$

output 1
output m

$$\Delta u = \begin{bmatrix} \Delta \hat{u}_1(k|k) \\ \vdots \\ \Delta \hat{u}_1(k+M-1|k) \\ \dots \\ \vdots \\ \dots \\ \Delta \hat{u}_m(k|k) \\ \vdots \\ \Delta \hat{u}_m(k+M-1|k) \end{bmatrix} \quad (3.9)$$

input 1
input m

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1m} \\ A_{21} & A_{22} & \dots & A_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ A_{m1} & A_{m2} & \dots & A_{mm} \end{bmatrix} \quad (3.10)$$

A_{ij} is the dynamic matrix corresponding to the i^{th} controlled variable and the j^{th} manipulated variable.

The open-loop optimization has the similar equation as SISO system (3.5). However, the weighting matrices are different. Q is an $mP \times mP$ matrix; Λ is an $mM \times mM$ matrix (also termed the move suppression factor).

$$Q = \text{diag}[\underbrace{q_1 \cdots q_1, q_2 \cdots q_2, \cdots, q_m \cdots, q_m}_P]$$

P elements corresponding
to output 1

$$\Lambda = \text{diag}[\underbrace{\lambda_1 \cdots \lambda_1, \lambda_2 \cdots \lambda_2, \cdots, \lambda_m \cdots \lambda_m}_M]$$

M elements corresponding
to input 1

The solution is given as before by:

$$\Delta u = \underbrace{(A^T Q A + \Lambda)^{-1}}_R A^T Q (y_{set} - b) \quad (3.11)$$

where the set point trajectory defined as:

$$y_s = \left[\begin{array}{c} y_{s1}(k+1) \\ \vdots \\ y_{s1}(k+P) \\ \cdots \\ \vdots \\ \cdots \\ y_{sm}(k+1) \\ \vdots \\ y_{sm}(k+P) \end{array} \right] \left. \vphantom{\begin{array}{c} y_{s1}(k+1) \\ \vdots \\ y_{s1}(k+P) \\ \cdots \\ \vdots \\ \cdots \\ y_{sm}(k+1) \\ \vdots \\ y_{sm}(k+P) \end{array}} \right\} \text{output 1 set point trajectory}$$

The rows 1, (M+1), (2M+1), ... , (m-1)M +1 of R are extracted to calculate the first move of each of the m inputs for implementation.

DMC control algorithms have been developed to handle different challenging applications, for example, quadratic dynamic matrix control (QDMC) (Garcia and Morshedi, 1986) which is used to control the system with constraints in controlled variables or manipulated variables. The Matlab MPC Toolbox makes it very easy to develop constrained SISO or MIMO DMC controller and carry out open-loop or closed-loop simulation.

3.2.2 Controlled and Manipulated Variables Selection

The first step in MPC controller design is to choose control variables and manipulated variables. The controlled variable is the target of the control system, and depends on manipulated variables and disturbances. Controlled variables are measured on-line and the measurements are used for feedback control, while manipulated variables are inputs that can be adjusted when needed, such that their change will affect the controlled variables directly or indirectly. For injection molding process control, manipulated variables are level-one variables (machine variables) while controlled variables are level-two (process variables) or level-three variables (quality variables).

As we know, in-mold process variables are closely related to product quality, special cavity pressure and cavity temperature. The cavity pressure profiles in filling, packing and cooling stage are quite different. Model-based predictive controllers have been employed to control the cavity pressure profile (Pramujati et al., 2006a, 2006b;

Dubay, 2001). Cavity pressure profile can be controlled cycle-to-cycle, using the peak value, the slope, time constant τ and *Controlled Pressure Cooling Time (CPCT)* as controlled variables. More attention has been paid to cavity pressure trajectory control in-cycle, which can greatly improve the process performance and directly determine the final part quality. Using the pressure trajectory as the control objective, the entire profile of the pressure development will be under control, which can ensure that the melt will follow the same course in every cycle so that the product quality will be more stable. In our study, MPC controller will be developed to control the cavity pressure trajectory in filling and packing stages.

Besides cavity pressure, part surface temperature profile also plays a great influence on part quality. So part surface temperature is selected as the other control variable in our two-by-two MPC controller. Advanced controllers have been developed to control the average temperature cycle-to-cycle (Hernandez et al., 2007; Dubay et al., 2006). However, control of the part surface temperature trajectory within a cycle is still rare.

Manipulated variables that affect cavity pressure and part surface temperature have been studied. A common method to control cavity pressure is to adjust the opening of a hydraulic servo-valve installed in machine hydraulic system. The opening of the valve directly affects the filling speed, filling pressure and packing pressure (Gao et al., 1996a). Since CAE software does not usually model physical units (such as the servo-hydraulic valve) directly, using filling flow rate or pressure settings as manipulated variables to control cavity pressure is reasonable. On the other hand, mold temperature

has been proven to strongly affect the cavity pressure, which can be controlled by adjusting coolant temperature and flow rate directly. So in this study SISO and one two-by-two MPC controllers are developed and their controller structure is shown in **Table 3.2**.

Table 3.2 Model-based predictive controller structure

MPC Controller	Controlled Variables	Manipulated Variables
SISO	Cavity pressure in filling stage	Filling flow rate
SISO	Mold surface temperature	Coolant temperature
SISO	Mold surface temperature	Coolant flow rate
MIMO	1. Cavity pressure trajectory in Packing stage 2. Part surface temperature trajectory in packing Stage	1. Packing pressure 2. Mold temperature

3.3 Simulation Process Preparation:

To collect simulation result data for controller development, a cell telephone cover was used as an example for process analysis in Moldflow[®] and thermal simulation software, ANSYS[®]. Product material, mold material and coolant selection are also very important for simulation work, which will be discussed in the follow section.

3.3.1 Part Selection

Thin-wall molding has been widely applied in different industries (e.g. portable electronics and computer), due to its attractive advantages such as material saving, cycle

time, part weight and overall part size reduction (Chen et al., 2000; Selden, 2000). Thin-wall molding is quite different from the conventional injection molding process, especially in the pressure control during filling and packing stages. A thin-wall cell telephone cover is selected as the studied component (see **Figure 3.2**). This part is the front cover of a cell phone, which includes many keyboard holes and a display window with a thickness of 1.2mm. The complex structure makes it very difficult to fill the cavity and control a mold temperature. Moreover, without proper control, the part is subject to warpage and weak mechanical properties. **Table 3.3** shows the dimensions and other data required for simulation.

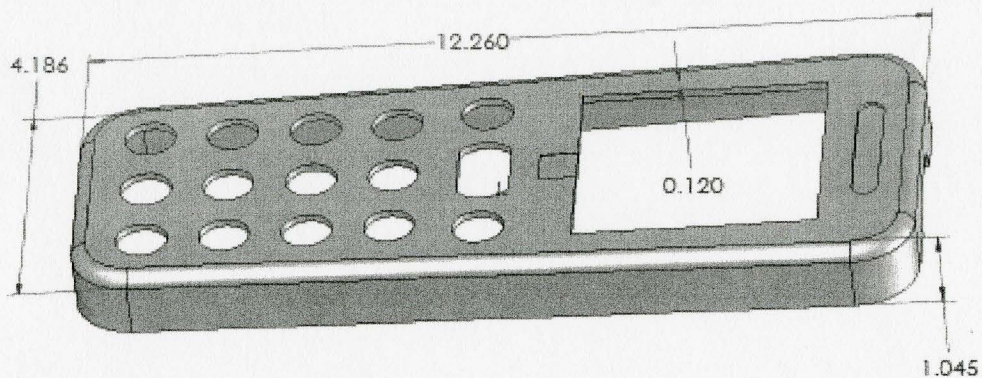


Figure 3.2 displays the cell telephone front cover.

Table 3.3 Cell Phone Front Cover Properties

Surface Area	36.36 cm ²
Volume	7.09 cm ³
Weight	8.17 g
Length	12.26 cm
Width	4.186 cm
Height	1.045 cm
Thickness	0.12 cm

3.3.2 Material Selection

Proper material selection is a critical step in product development process, which affects not only product design, but also future process control and final part quality. Material used for thin-wall molding must have high-flow capability and good impact strength. For cell phone cover, the material must also have good heat resistance, UV stability and a wide processing range. Engineering thermoplastic materials typically used for cell phone covers are polycarbonate (PC), acrylonitrile-butadiene-styrene teropolymer (ABS), and PC/ABS resins. GE Plastics provides a wide range of material for thin-wall molding application. Since the thickness of the cover is only 1.2mm, high-flow capability material must be used. Cycloy PC/ABS C1000 HF is selected because of its good high-flow capability, UV stability and excellent impact capability. **Table 3.4** shows material properties as provided in Moldflow®.

Table 3.4 Main properties of Cycloy C1000 HF.

Mold Surface Temperature	86 °C
Mold Temperature Range	79-93 °C
Melt Temperature	274 °C
Ejection Temperature	110 °C
Specific Heat	2270 J/kg-K
Thermal Conductivity	0.26 W/m-K
Melt Density	0.97681 g/cm ³
Maximum Shear Rate	40000 s ⁻¹
Maximum Shear Stress	0.4 MPa

3.3.3 Mold and Coolant Selection

Due to its high injection pressure requirement during processing, the mold steel used in thin-wall molding must have high stiffness and good stability, especially for telecommunication products. Thin-wall molding also requires thicker and larger support plates in case of mold deformation during production, which may cause warpage and dimensional problems in the final part. Hardened tool steels such as P20, H-13 and D-2 are commonly used in thin-wall molding; H-13 is selected here and the key parameters for Moldflow[®] analyses are shown in **Table 3.5**.

Coolants are usually mixtures of different chemicals that inhibit corrosion, algae growth or improve the thermal properties (Reynolds, 2000). In this case, water is selected as the coolant for the simulation work. The main properties used for thermal analysis is displayed in **Table 3.5**.

Table 3.5 H-13 and water thermal properties.

Property	H-13	Water
Specific Heat	462 J/kg-K	4180 J/kg-K
Thermal Conductivity	29.5 W/m-K	0.643 W/m-K
Density	7.76 g/cm ³	0.988 g/cm ³

3.4 Injection Molding Process Simulation:

Moldflow Plastics Insight (MPI), which is widely used for injection molding process simulation in both academia and industry, is used to analyze the molding of the cell telephone cover. MPI provides useful information about melt flow status inside the cavity (e.g. pressure, temperature, filling time, shear rate), and final product quality predictions (e.g. weld lines, air traps, shrinkage and warpage). This information aids the designer to modify the part or mold design and the process engineer to optimize the process settings. MPI only simulates the polymer melt inside the mold, not the temperature profile through the mold itself. FEM software such as ANSYS® can be used to analyze both the part and the mold thermal activities in 2D or 3D.

3.4.1 Model Preparation

Setup of a proper model is the first and most important step to start the analysis and obtain accurate and valuable results. There are usually five steps to prepare a model for analysis:

- 1. Import Product Geometry:** A 3D CAD drawing with the format in IGES (Initial Graphics Exchange Specification, a vendor neutral standard format used to

exchange geometric models between various CAD and CAE systems) or STL (Standard Triangulation Language, where the shape of the object is defined by a mesh of tiny triangles laid over the surfaces) is commonly used for Moldflow, which includes most details of the part design, such as dimensions and structure of the component. An IGES file of the cover was developed and imported into Moldflow.

2. Meshing Geometry: Correct meshing not only covers the details of the part structure but also it is the basis of simulation work. The more accurate the part meshing, the more accurate results can be attained. There are mainly four mesh types for analysis (Jaworski and Yuan, 2003): a. Beam (1D), simple 1D line element connecting two nodes with an assigned cross sectional area shape usually used to model melt delivery systems (cold and hot runner); b. mid-plane (2.5D), represents a 3D part with a 2D planar surface at the center of the thickness. A thickness property is assigned to this planar surface, so called 2.5D. To create a mid-plane model sometimes is very difficult and time consuming. Given that, it is still widely used because it has more analysis options and a relatively fast solution time. c. Dual Domain[®] (modified 2.5D), this patented technology represents a 3D part with a boundary or skin mesh on the outside surface of the part. It is similar to mid-plane mesh but this boundary shell mesh has aligned and matched mesh on the both corresponding outside surfaces and the distance between the mesh defines the part thickness. d. Three dimensional (3D), a 3D mesh typically consists of four noded tetrahedral elements and it is a true 3D representation of the part.

For injection molding flow analysis, most commercial CAE simulation software utilizes the generalized Hele-Shaw (GHS) flow model along with some other

assumptions. Through a comparison of the simulation results to experimental results, Jaworski and Yuan (2003) concluded that a 3D mesh can provide the most accurate result but involves significantly long calculation times and a large amount of computer resources. Mid-plane or Dual Domain meshes are accurate for thin wall components and allow more analysis options and shorter solution times. Fusion mesh is used for the cell phone cover meshing in MPI, which consists of a mixture of different types, including regions with traditional Midplane elements and surface (double-skin) shell elements. The meshing component is shown in **Figure 3. 3**.

3. Analysis Task Selection: select analysis task needed such as: Fill, Cool+Flow, Cool+Flow+Warpage. The selection is base on what kind of results you want.

4. Injection Locations and Runner System: Valve gate hot runner feeding system is used in the cell phone cover mold and the gate size is 1.5mm. Injection gate point locates in the middle of the part (See **Figure 3.3**). In this system, a valve pin is driven by an air cylinder to open and close the gate during cavity filling and packing stage. The valve gate system is very popular in telecommunication and computer industrial components manufacturing, due to its advantages in short cycle time, material saving and good visual quality of the part.

5. Cooling System Design: Moldflow provides a very convenient tool for cooling channel design and meshes it with beam elements. A coolant channel for the cell phone cover is designed for the simulation work, shown in **Figure 3.3**.

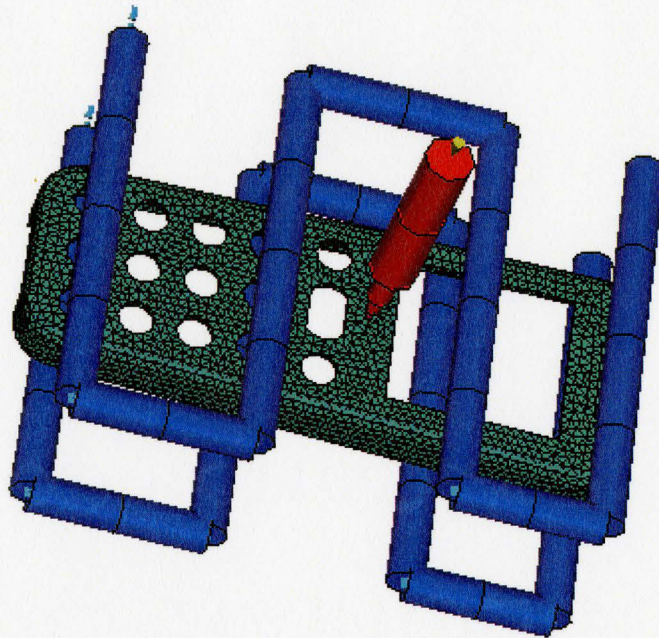


Figure 3.3 Model for simulation in Moldflow[®]

3.4.2 Process Conditions

As discussed before, there are three level variables (machine variables, in-mold process variables and quality variables) in the injection molding process. Among them, only machine variables can be manipulated in order to meet the final part quality requirement. Proper process conditions settings will lead to stable production and good quality parts. Compared to conventional injection molding process, the main difference in thin-wall molding is much higher injection speed and pressure, and short cycle time. With the valve gate control system and automatic component handling system (e.g. industrial robot), production can be run in a cycle time lower than 10s for cell phone cover. **Table 3.6** lists the main processing parameter used in my simulation work.

Table 3.6 Key processing parameters using in simulation.

Mold Surface Temperature	85 °C
Melt Temperature	275 °C
Injection Speed	70 cm ³ /s
Filling-Packing Switchover	By Volume: 95%
Packing Pressure Profile	70 MPa --2 s
Valve Gate Operation Time	0-2.5 s
Cycle Time	12 s
Coolant Temperature	75 °C
Coolant Flow Rate	10 liter/minute

3.4.3 Simulation Result

Depending on the selection of analysis task, Moldflow provides lots of analysis results for further process investigation. Cooling and Flow analysis was carried out in this study, which can provide the information about melt flow status (e.g. temperature, pressure, shear rate, shear stress and so on) in filling, packing and cooling stage.

The temperature and pressure distribution on the part is also very important for process control. This distribution will affect molecular orientation and density, which may cause shrinkage, warpage and other quality problems. In order to control the entire cavity pressure profile in a cycle, the sampling position is chosen close to the gate, since this is the first part to be filled and the last part to get frozen. On the other hand, a point near the end of the part is selected as a sampling position for part surface temperature control. Because the part close to the end of the cavity is the first part to get frozen, this

can reflect the melt status change easily. Moreover, the temperature profile in those points is not affected by the runner system and it can reflect the real heat transfer between mold and part. The pressure and temperature measure positions are shown in **Figure 3.4**. Node 1838 is for pressure measurement and Triangle 5092 is for temperature measurement. The pressure and temperature measurement results may have a little bit difference if different points nearby are selected, but it can still reflect the main trend.

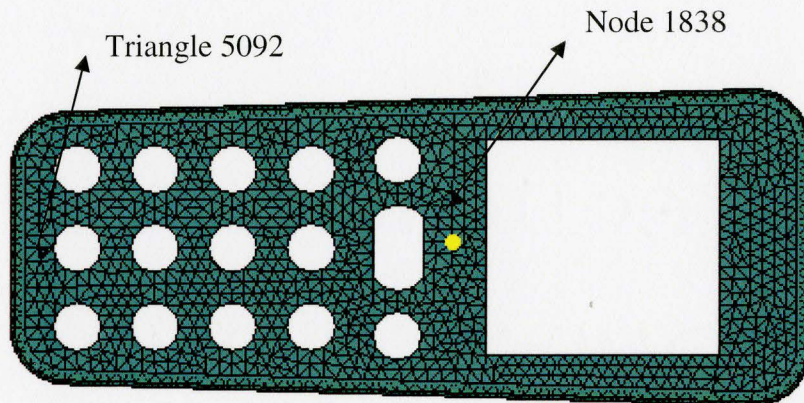


Figure 3.4 Pressure and temperature measurement position

3.5 FEM Thermal Simulation

Moldflow and other injection molding process simulation software usually use boundary element method as the major numerical scheme to solve for the heat transfer with mold. One of the main assumptions is that the initial temperature of the mold is constant. The cooling simulation is usually conducted using a cycle-averaged approach with one-dimensional heat transient variation (Chen et al., 2004). To investigate the mold

temperature transient variation in different cycles in the startup stage or after process conditions change, experiments on the actual mold, or FEM programs are applied to the injection molding process analysis. A 2D model of the cell telephone cover mold is prepared in ANSYS®. Investigations about mold surface temperature distribution and transient variation between cycles after coolant conditions change (e.g. coolant temperature and flow rate) is done.

3.5.1 Model Preparation

A significant amount of research has been done to investigate the mold surface temperature distributions and transient variation in 2D or 3D models by ANSYS® (Xu and Kazmer, 1999; Chen et al., 2004, 2005, 2007). **Figure 3.5** shows the two-dimensional model, developed in ANSYS® for simulation work.

Model meshing is very important in FEM simulation and directly affects the simulation accuracy. There are about 40 elements for ANSYS thermal analysis. The commonly used 2-D solid elements are a triangle with 6 nodes, and a quadrilateral with 4 or 8 nodes. Different condition or property (such as temperature) can be set in each node. In our model, the coolant holes are meshed as triangular elements, the part is meshed as quadrilateral elements with 8 nodes, the cavity and core are triangular elements with 4 nodes. **Figure 3.6** shows the different element geometries used in model meshing. The element size and density must be defined properly in order to get more accurate results. Higher density and smaller size are necessary for some special areas, like cavity surface, the part, and the area around the coolant holes. **Figure 3.7** shows the meshing part.

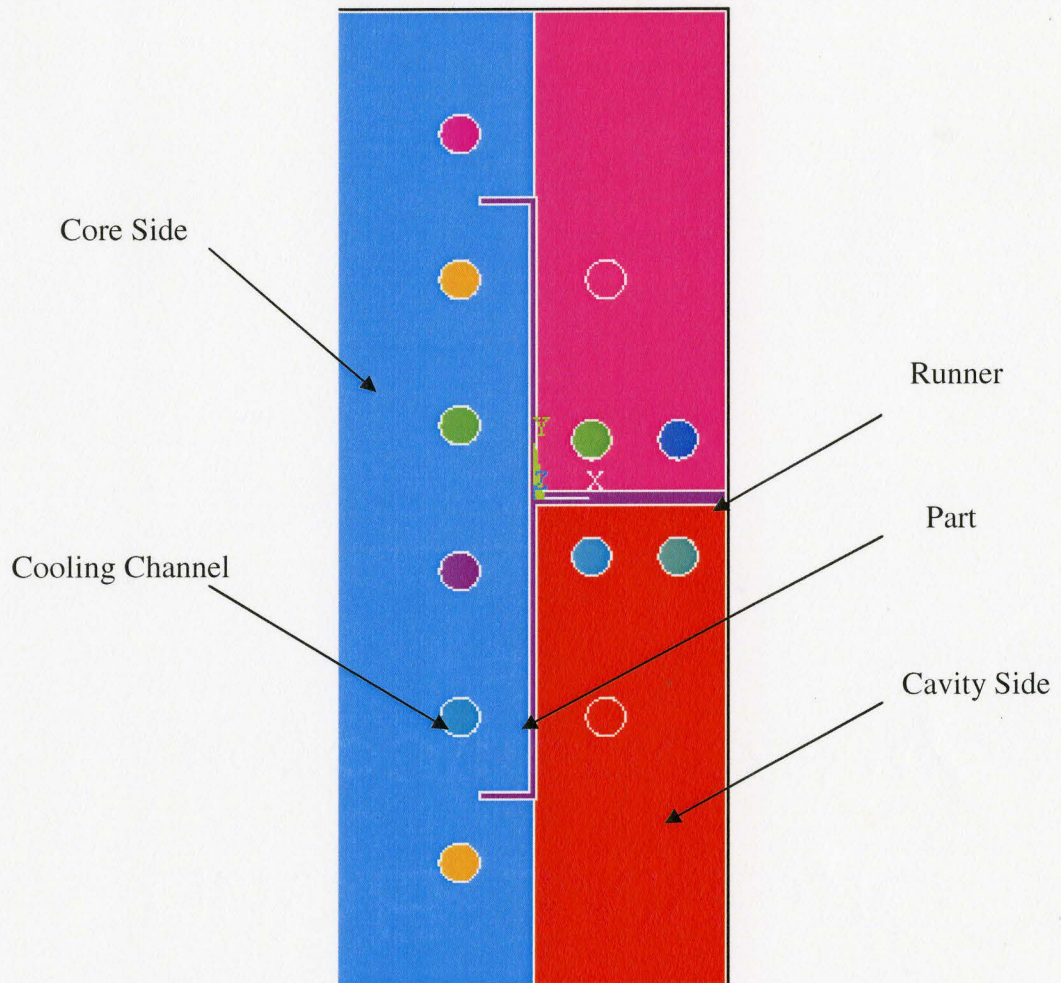
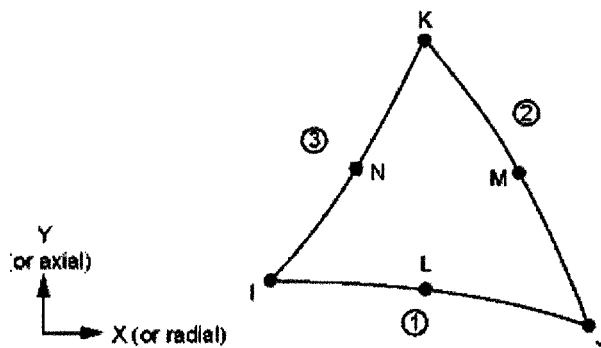
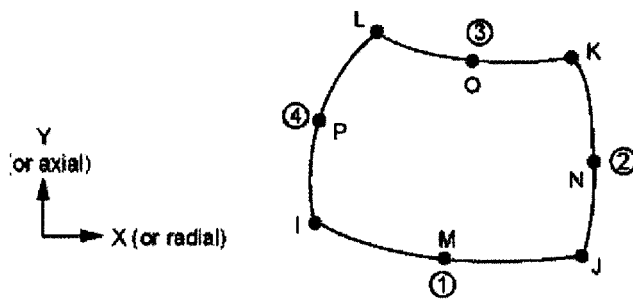


Figure 3.5 the Two-dimensional model using for simulation in ANSYS®



Isoparametric quadratic triangular elements (PLANE 35 Geometry in ANSYS®)



Isoparametric quadratic quadrilateral elements (PLANE 77 Geometry in ANSYS®)

Figure 3.6 Element geometry using in model meshing (ANSYS, 2007)

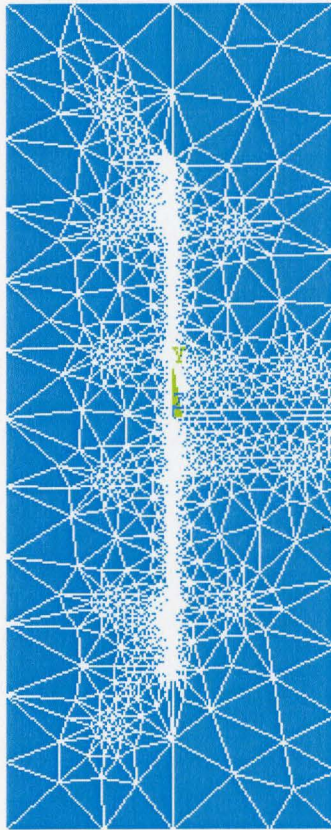


Figure 3.7 Meshing model for FEM thermal simulation

3.5.2 Analysis

After setting up the mesh, the next step is to decide the initial conditions such as temperature and material properties, for the mold, part and the coolant channels. The main material thermal properties (specific heat, thermal conductivity and density) for each material must be selected before model meshing. **Table 3.7** lists the initial processing temperature conditions, taken from Moldflow analysis. The heat transfer coefficient between the mold and the coolant must be calculated, based on the material properties and initial conditions. (**Appendix B**). In order to simplify the simulation and reduce computing time, radiative heat transfer is assumed negligible (Xu and Kazmer, 1999).

In the first cycle, the initial conditions are applied to each area (component), the system will transfer the conditions to each element and node when analysis is started. The analysis result is written to a specific file, which records all the temperature values (and other information of each node) at each sampling time. This information will be used as initial conditions to for the next run. Part and coolant temperatures have to be set to their initial temperatures manually in order to simulate the cyclical molding process. **Figure 3.8** shows the procedure to run the analysis (Xu and Kazmer, 1999). Following this procedure, we can run the simulation continuously until the temperature distribution in the mold gets stable and there is no variation between cycles. A step change is made to the coolant flow rate or the coolant temperature to investigate the dynamic response of the system. Result data will be collected to develop the MPC controller.

Table 3.7 Processing conditions using for simulation.

Initial Melt Temperature	548 K
Initial Mold Temperature	338 K
Coolant Temperature	348 K
Coolant Flow Rate	10 liter/minute
Cycle Time	12 s
Sampling Time	0.05 s

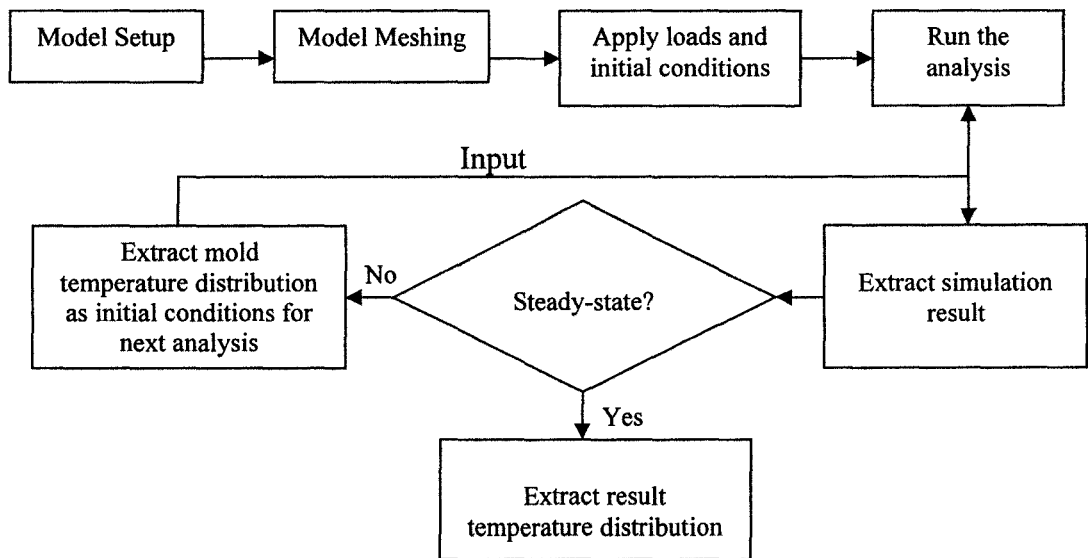


Figure 3.8 ANSYS analysis flow chart

3.6 MIMO Controller Development

3.6.1 Model Identification

To develop first-principles dynamic models is difficult and time-consuming; moreover, the model may be too complicated to be applied for on-line process control. On the other hand, empirical models provide the dynamic relationship between selected input and output variables, which is very meaningful for modern process control. In this study, empirical models are developed based on CAE simulation results and applied for MPC control.

MPC controller accepts many different kinds of dynamic model such as autoregressive with exogenous input (ARX), autoregressive moving average with exogenous excitation (ARMAX), and Finite Impulse response (FIR) model. This property makes it easier to develop the controller compared to other advanced algorithms (e.g. ANNs and STR), and suitable for a range of process complexity (e.g. linear, non-linear, long delay time). The ARX model, a linear polynomial model, is a very popular model form in industry and has been applied by researchers to develop modern controllers for injection molding processes. Pramujati and Dubay (2006a) used this model to develop SPC and DMC controller for cavity pressure control during cooling stage. They pointed out that this model is suitable within the range of inputs and outputs of the open loop test from which it was created, especially if the controlled variable is nonlinear or time variant.

The ARX model is a linear difference equation that uses least squares estimation to determine the model parameters. The model structure is: (Ljung, 2002)

$$A(q)y(t) = B(q)u(t-nk) + e(t) \quad (3.12)$$

Where the value $A(q)$ and $B(q)$ are polynomials of the delay operator q^{-1} and are related to the plant $G(q)$ in the form

$$G(q) = q^{-nk} \frac{A(q)}{B(q)} \quad (3.13)$$

The coefficients of $A(q)$ and $B(q)$ are evaluated by least-squares minimization and can be expressed as:

$$A(q) = 1 + a_1 q^{-1} + \dots + a_{na} q^{-na} \quad (3.14)$$

$$B(q) = b_1 + b_2 q^{-1} + \dots + b_{nb} q^{-nb+1} \quad (3.15)$$

Where na and nb represent the order of the system, while nk represents the process delay as an integer multiple of the sampling interval. The parameter $e(t)$ is the error that exists in the formulation of the ARX model. The process to identify those parameters is called System Identification (SI). System identification is an iterative process to estimate different models for the data and compare the model performance, and then choose the simplest model that adequately describes the dynamics of the system. ARX models will be used in our study and developed by Matlab System Identification Toolbox.

3.6.2 MPC Controller Program

Using the functional code in Matlab and MPC Toolbox is very convenient to write the program for MPC controller. The program structures for both SISO and MIMO controller are same and can be developed following the standard procedure. The first part

of the program is to setup plant model, where the ARX dynamic model will be used. Then the controller model is defined and there is usually some measured process and model mismatch. Here, we assume there is no model and process mismatch, since the “process” is a computer simulation. The controller parameters will be determined for the MPC controller. The key parameters are prediction horizon (P) and control horizon (M), the input weights (uwt) and output weights (ywt). Those parameters can be adjusted and used to tune the MPC controller in order to improve controller performance. Constraints can be applied to inputs or outputs, which make sure that the process does not move outside of the process limit and cause safety problems. Applying constraints to the controlled or manipulated variables can also keep the process running smoothly and not too aggressively.

The MPC controller can be implemented to control the process by giving a set-point to outputs. Based on the model response, proper adjustment to the tuning parameter (e.g. P, M and uwt) can be done to get better performance. An example of the MPC closed-loop simulation program is shown in **Appendix C**. In this simulation, the plant model is assumed to predict the real plant status perfectly.

3.6.3 Closed-loop Controller Simulation

In order to simulate the closed-loop control, work has been done to modify the MPC program. The program is rewritten from a program in MPC Toolbox (**Appendix D**). The idea is to couple the MPC controller and CAE simulation software together to run the closed-loop simulation manually. Simulation work has been done to investigate the SISO MPC controller for cavity pressure in the filling stage. First, the cavity pressure trajectory

in the filling stage is given to the MPC controller as a set point and run for one sampling interval. The first manipulated variable (injection flow rate, u_1) will be sent back to CAE program (MPI) as processing condition. In this case, there is only one step in the injection flow rate setting profile, and the filling time is one sampling time. MPI will run a sampling time and stop filling the cavity further. The cavity pressure value is sampled and sent as feedback to the MPC controller as output. The controller uses this value to update its process outputs and run another sampling time to calculate the next manipulated variable value (u_2), which will lead the system to meet the fixed pressure trajectory. Then the injection flow rate setting profiles will have two steps (u_1 - u_2). MPI programs will run two sampling times and stop to extract the cavity pressure for the next simulation. Figure 3.9 illustrates the injection flow rate profiles in k interval. The simulation result of MPI at k interval will be sent to MPC to calculate the next manipulated moves. Then $u(k+1)$ will be added to the injection flow rate profile for the next analysis. By doing this repeatedly, we can simulate and check the closed-loop controller performance, as if the controller is working with the “perfect” injection molding machine. That is very meaningful for the future application of the MPC controller used in actual production.

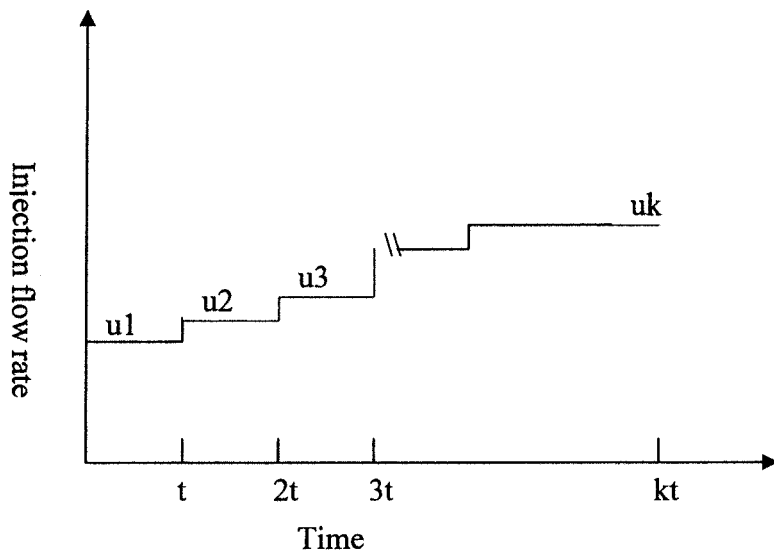


Figure 3.9 Illustration for MPI injection flow rate step profile.

Chapter 4

Simulation Results

4.1 Overview

The melt flow history inside the cavity during filling and packing stages is decisive for the final part quality. The process control objective is to keep the process running in the same and proper way in every cycle, which will lead to stable product quality. In this study, MPC controllers were developed to control the filling and packing phases, based on the simulation results from Moldflow[®] and ANSYS[®].

1. A SISO MPC controller was developed to control cavity pressure in filling stage by manipulating injection flow rate.
2. Two SISO MPC controllers were developed to control mold surface temperature, using coolant temperature and coolant flow rate as manipulated variables.
3. A two-by-two MPC controller was proposed to control cavity pressure profile and part surface temperature during packing stage. The manipulated variables are packing pressure and mold temperature.

4.2 Cavity Pressure Control in Filling Stage

A single-input/single-output DMC controller was developed to control the cavity pressure in the filling stage based on the simulation result from Moldflow[®]. A prescribed pressure trajectory during the filling stage was set as the control objective. In order to check the controller performance, some process conditions (melt temperature and mold temperature) are changed and closed-loop simulation was done, by creating a DMC controller in Matlab with results from Moldflow[®] MPI. The result shows that the DMC controller is able to keep the cavity pressure profile very well.

4.2.1 Simulation Result

Unlike conventional thermoplastic injection molding, one of the main characteristics of thin-wall molding is that the cavity pressure during the filling stage increase very quickly. Moreover, the cavity pressure is usually much higher than conventional molding. **Figure 4.1** shows the typical cavity pressure-time curve for conventional injection molding. A typical thin-wall molding cavity pressure profile is shown in **Figure 4.2**. The dynamic response of the cavity pressure as a function of filling flow rate was determined through a step change applied to the flow rate. The step profile was: for the first 0.05 seconds, a flow speed with $30\text{cm}^3/\text{sec}$ will be used to fill the mold. Then, a flow-rate of $70\text{cm}^3/\text{sec}$ is used to fill the cavity until the filling-to-packing switch-over time, which was controlled by the percentage of volume filled and 95% was selected in this study. From the MPI simulation results, the filling-to-packing switch-over time is 0.145 seconds and the cavity pressure is 47.5 MPa, which will be used to define the end

of filling stage. However, the filling time and cavity pressure may be changed without proper control, if process conditions are changed, or unexpected disturbances affect the process. In this study, the target is to keep the process meeting that point (switch-over at time 0.145s with pressure at 47.5 MPa) in every cycle by manipulating the filling flow rate. The sampling time was 0.005 seconds, and the cavity pressure data at point N1838 was used to develop the dynamic controller model. Since this point is close to injection gate and able to record the whole cavity pressure profile during a cycle. If picking other sampling locations, the cavity pressure profile may be a little different from each other. In that case, the process dynamic model parameters are also different, which will finally affect the MPC controller parameters. However, even with different sampling position, so long as it can reflect the whole process dynamic, the process and the part quality can be under controlling with proper controller design. **Figure 4.2** shows the cavity pressure profile at the selected location (Node 1838).

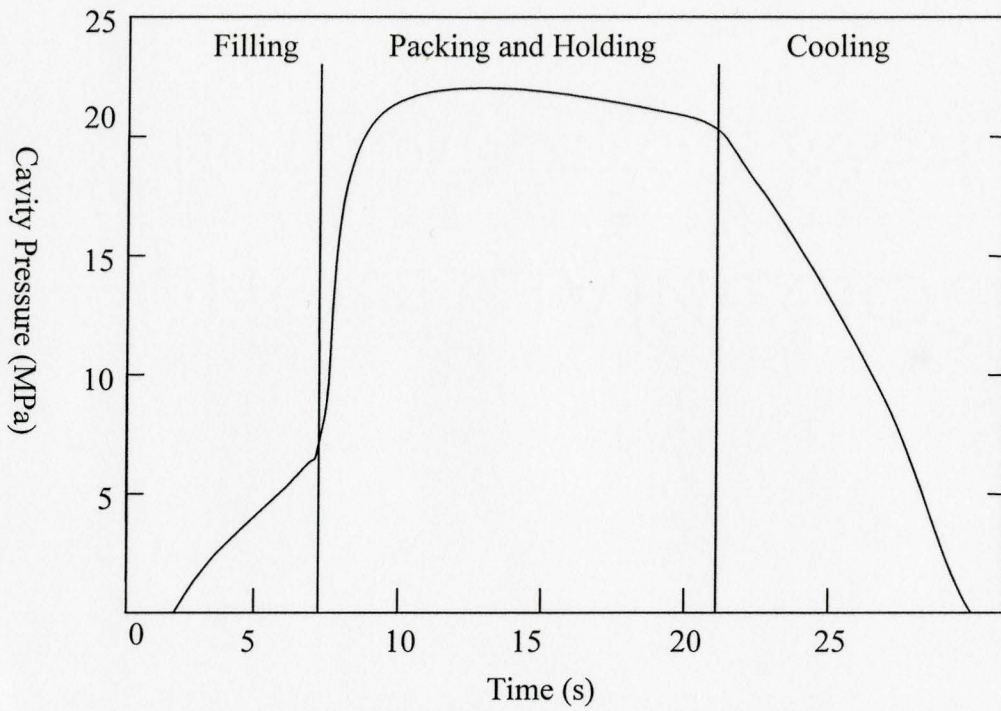


Figure 4.1 Typical cavity pressure –time curve.

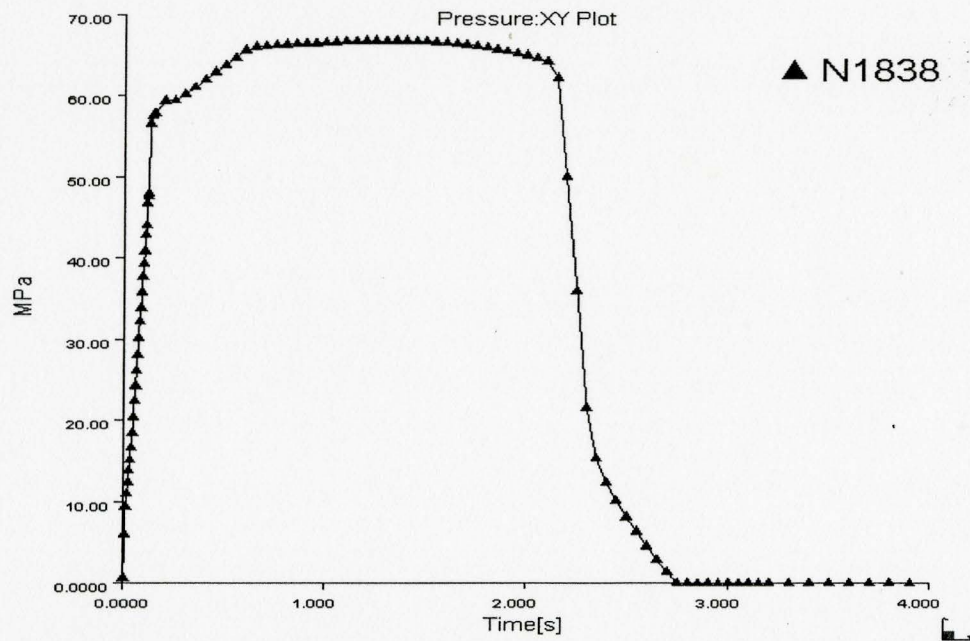


Figure 4.2 Cavity pressure profile -time curve (from Moldflow[®] MPI simulation).

4.2.2 Controller Development

In the development of a DMC controller, an important step is to develop the dynamic matrix model. The conventional procedure is to run a unit step change to a steady-state continuous system. From the output curve, the dynamic matrix can be easily defined. In this case, another method must be used to define the dynamic matrix, since there is no steady-state in an injection molding system as it is inherently a batch production system. The Matlab[®] System Identification Toolbox was used to develop an ARX model from the raw data. The identified model was transferred into a step response model by using the Matlab[®] Model-based Predictive Control Toolbox.

Based on the pressure data from MPI, the first pressure measurement at Node 1838 was sensed at 0.02 seconds after injection start. In other words, there are four sampling time delays in the system. A second-order ARX model with $n_a = 2$ and $n_b = 2$ is given as:

$$\text{Discrete-time model: } A(q)y(t) = B(q)u(t) + e(t)$$

$$A(q) = 1 - 1.607 (+0.1572) q^{-1} + 0.6267 (+0.1537) q^{-2}$$

$$B(q) = 0.02209 (+0.01347) q^{-4} - 0.002708 (+0.01401) q^{-5}$$

The model was then used to develop the MPC controller based on the Matlab[®] MPC Toolbox. The key tuning parameters are listed in **Table 4.1**. The control horizon M is selected to be 2, which can generate more stable manipulated variable moves, when the cavity pressure trajectory was used as control objective not a fixed set point. The prediction horizon P is often selected to be $P=N+M$ so that effect of the last input move

is taken into account. The input weight is selected to be 0.85 by testing different values in the closed-loop controller. Basically, increasing the value of the input weight tends to make the controller more conservative by reducing the magnitudes of the input moves.

Table 4.1 Tuning parameter of SISO MPC controller

Controller Tuning Parameter	Value
Prediction Horizon (P)	32
Control Horizon (M)	2
Output Weight	1
Input Weight	0.85

4.2.3 Controller Performance

Three experiments, based on CAE process simulations, were done to check the DMC controller performance. Moldflow[®] MPI was used to check the closed-loop response of the system.

Experiment 1: A target cavity pressure profile with a slope as 0.3 MPa/msec was used as the controller set-point. First, a MPC closed-loop simulation was done in Matlab[®], without MPI participation. **Figure 4.3** shows the response of cavity pressure and the filling flow rate profile. Then Moldflow[®] MPI was used to check the closed-loop response. The pressure setting profile and the simulation output from MPI are plotted in **Figure 4.4**. The controller works very well to keep process output repeating the set-point, and minimize the manipulated variable moves. A small deviation in the start of the filling stage was found, which might be due to the effect of dead time, but this difference is acceptable since the effect on part quality is limited.

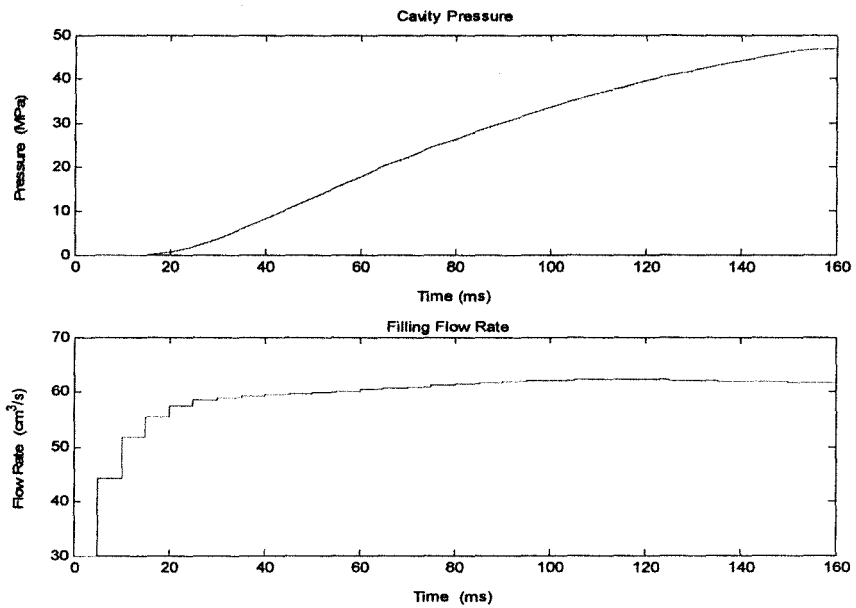


Figure 4.3 Closed-loop responses with a slope set-point (0.3 MPa/msec).

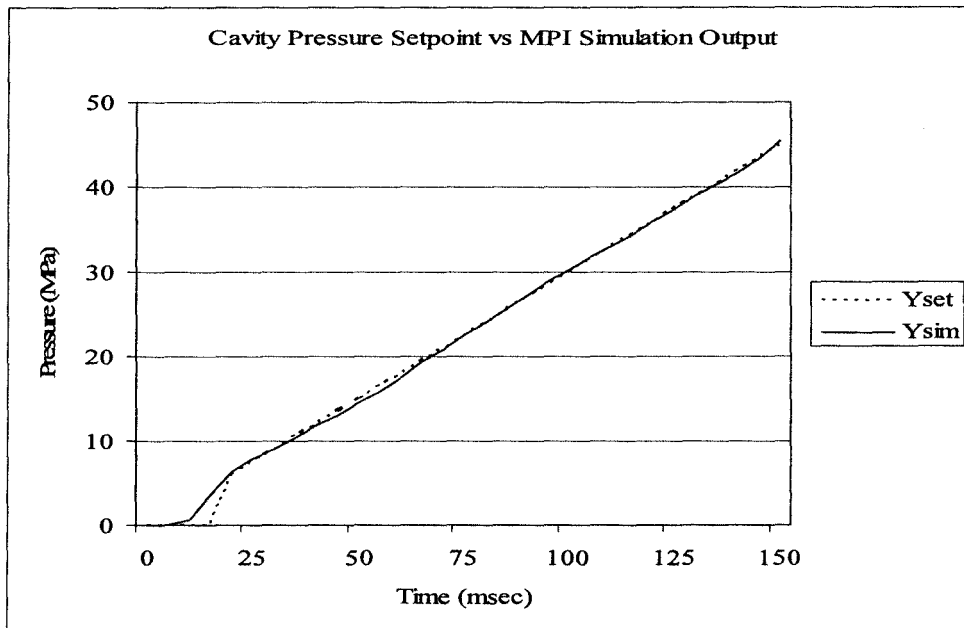


Figure 4.4 Cavity pressure set-point and simulation pressure output from MPI with MPC closed-loop control.

Experiment 2: The cavity pressure trajectory was set as the control objective of the MPC controller. As an example, consider that there is an unexpected change in the process, which causes the melt temperature to increase from 275 °C to 280 °C. Without control, the cavity pressure profile will be changed. The MPC controller is used to adjust the filling flow rate in order to reduce the effect of the melt temperature change and maintain the target cavity pressure profile. The melt temperature setting in MPI was changed to 280 °C in the beginning of the simulation. **Figure 4.5** shows the closed-loop response of the system and **Figure 4.6** illustrates the set-points and actual cavity pressure profiles taken from the MPI simulation result. In general, the control system maintains the cavity profile very close to the set trajectory. However, the difference in the early stage is larger. The possible reasons may be due to the dead time effect; the process control model accuracy, compared to the simulation; or the increasing melt temperature leads to better melt flow capability.

Experiment 3: The MPC controller setting is the same as experiment 2, but the process condition is changed to the mold average surface temperature (rather than the melt temperature). **Figure 4.7** shows the closed-loop response of the system and **Figure 4.8** illustrates the set-point target and actual cavity pressure profiles taken from the MPI simulation result. The cavity pressure profile reasonably matched the setting trajectory. The system had larger deviations from the set point trajectory, compared to experiment 2. The possible reasons include dead time, the process control model accuracy.

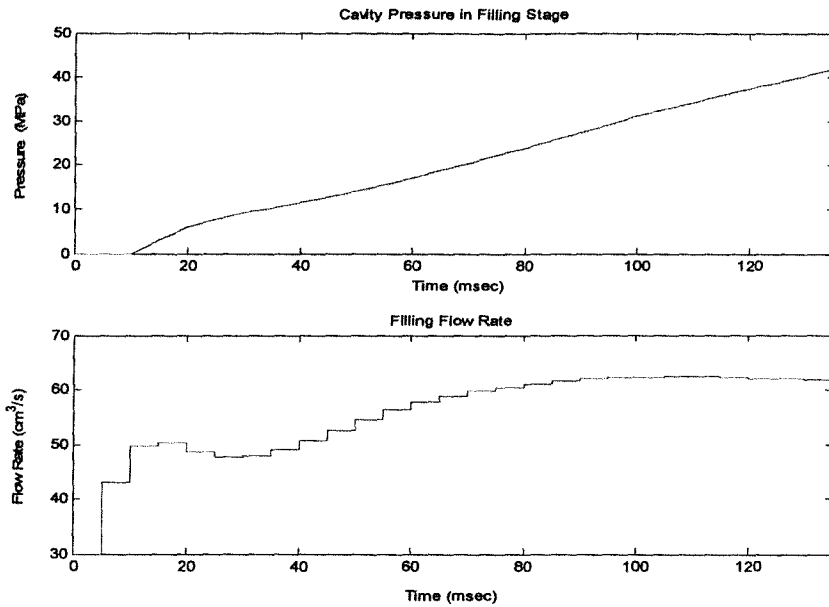


Figure 4.5 Closed-loop responses with melt temperature change.

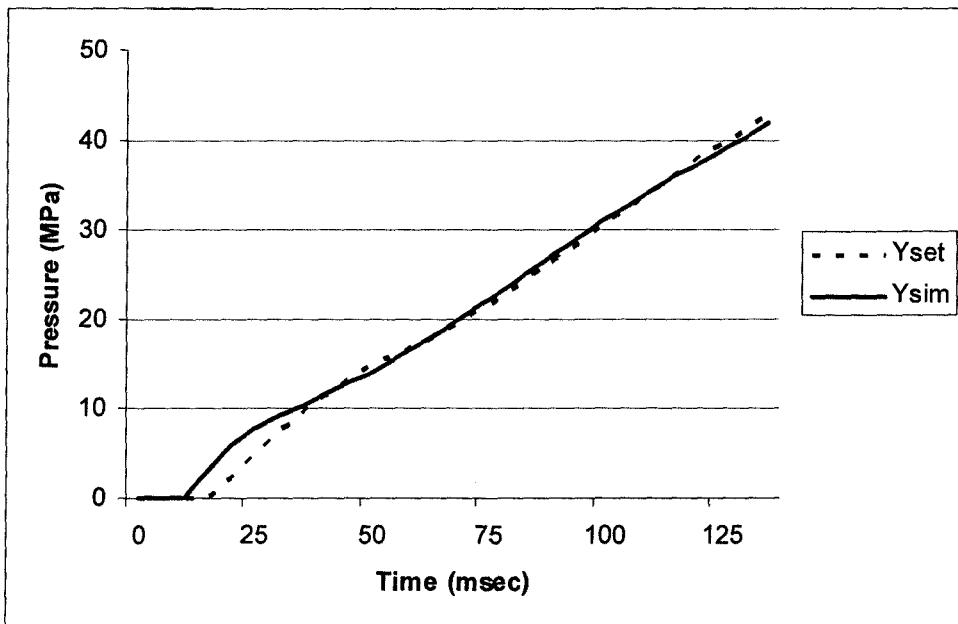


Figure 4.6 Cavity pressure set-point and simulation pressure output from MPI with MPC closed-loop control (with melt temperature change).

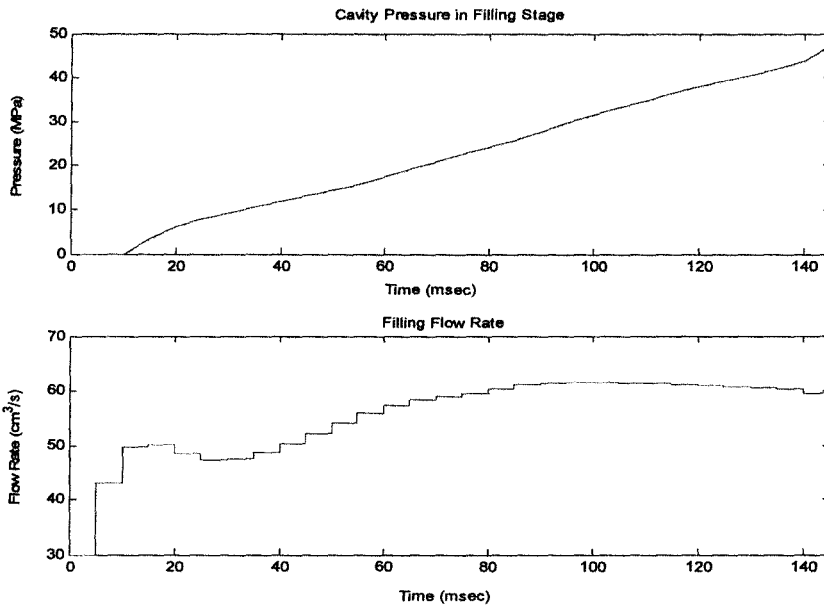


Figure 4.7 Closed-loop responses with mold temperature change.

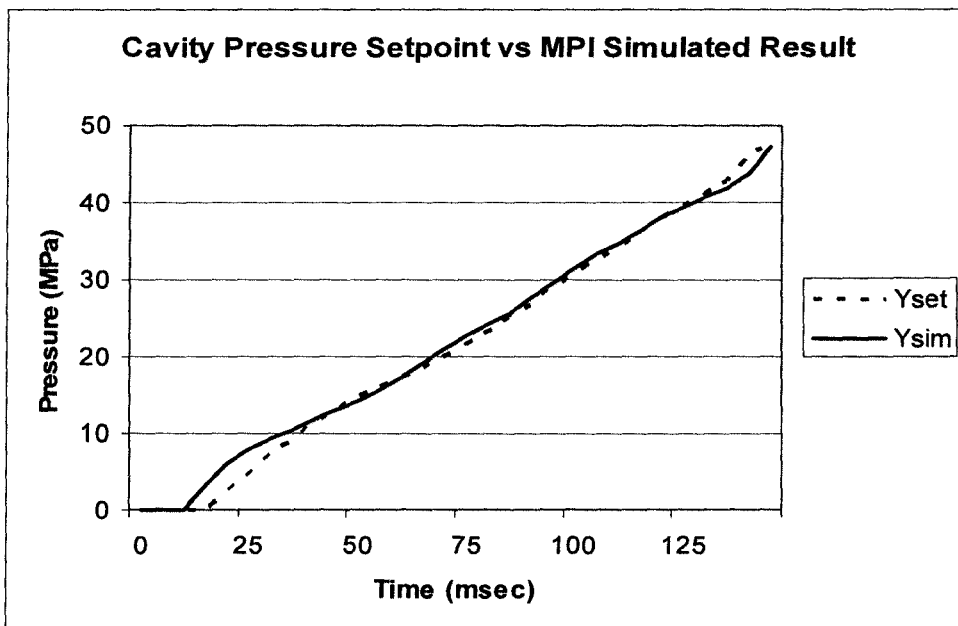


Figure 4.8 Cavity pressure set-point and simulation pressure output from MPI with MPC closed-loop control (With mold temperature change).

4.3 Cavity Surface Temperature Control

4.3.1 Simulation Result

A two-dimensional transient analysis of the temperature field in the mold using ANSYS was carried out. After setting the initial conditions and running the first simulation cycle, the mold temperature result was used as the initial conditions for the next simulation. The simulation continues until the mold temperature reaches the cyclic steady-state. **Figure 4.9** shows a temperature contour plot of the system. We see that the cavity temperature is quite uniform except in the runner position. This is reasonable, since the thickness of the part here is larger than that of the other locations. Three locations on the mold surface were selected to measure the cavity surface temperature separately in order to check the cavity temperature status. **Figure 4.10** illustrates a typical cavity surface temperature within a cycle. Obviously, the cavity surface temperature drops very fast in the first few seconds. Therefore, the sampling time is very important in order to capture the dynamic transient of the cavity surface temperature. However, smaller sampling times need much more computational time and faster hardware requirements. Since the total cycle time is 12 seconds in this study, the sampling time is selected to be 0.1 second, which is suitable to reflect the dynamic temperature profile. **Figure 4.11** shows average cycle temperature transients at three different measured points on the cavity surface. The average cycle temperature at selected locations on cavity surface can be taken from ANSYS®. After running about 20 cycles, at all measured positions, the average cycle temperature difference between two continuous cycles is less than 0.05%, which means that the system is very close to the steady-state. Further tests can be

implemented to investigate the dynamic response of the process in order to develop the process model. A step change was then given to either the coolant flow rate or coolant temperature and then the simulation was run. For a coolant flow rate change, the step change is -5 LPM (liter per minute), from 10 LPM to 5 LPM, and the step change for the coolant temperature is +10 °C, from 75 °C to 85 °C. **Figure 4.12** and **4.13** show the response of cavity surface average cycle temperature to the step change in coolant flow rate and coolant temperature. After 20 cycles, the system started to move back to a steady-state. Even though the system is not totally at its steady-state, it is still able to represent the main dynamics of the system to the step change. The simulation results for the temperature at Node 805 were selected to develop the dynamic model, since the temperature profiles sampled at Node 805 and Node 915 are very similar.

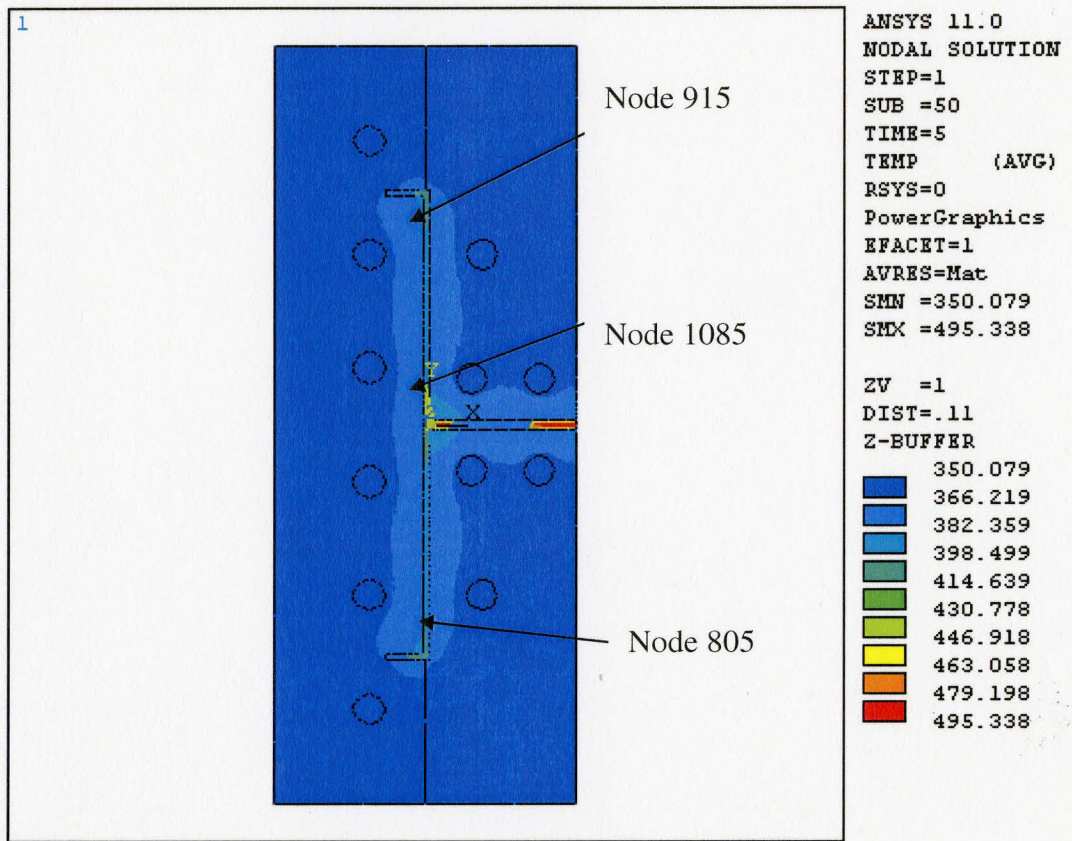


Figure 4.9 Temperature contours in ANSYS simulation.

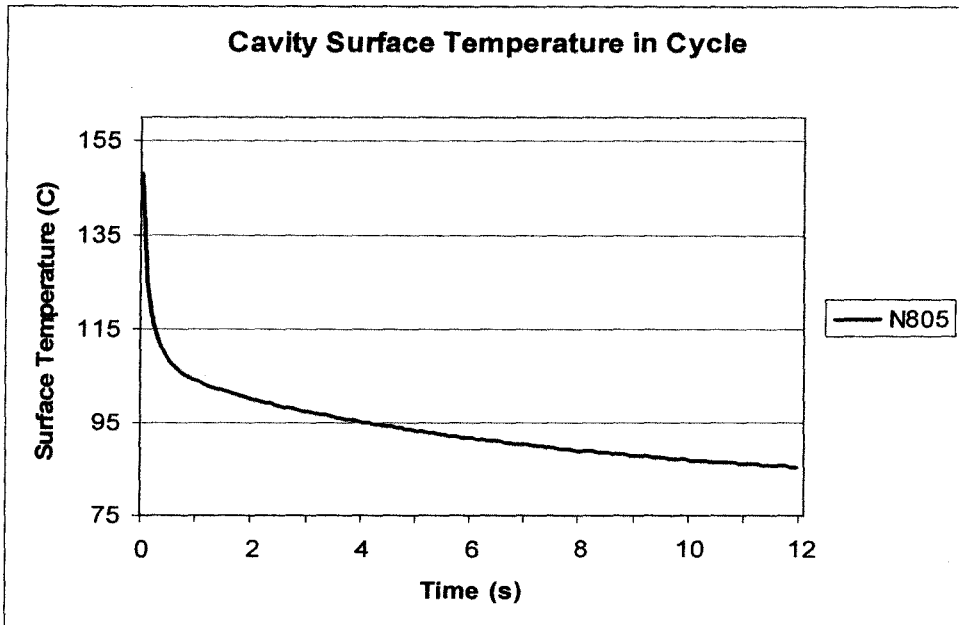


Figure 4.10 Typical simulated temperature profiles for cavity surface within a cycle.

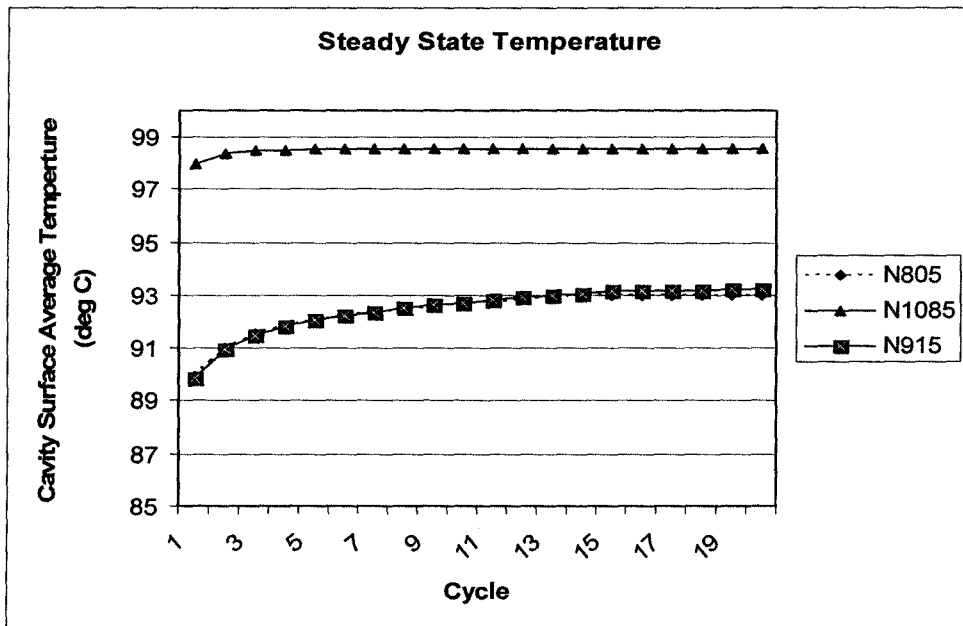


Figure 4.11 Simulated results of cyclic, transient mold surface average temperature variations in selected locations.

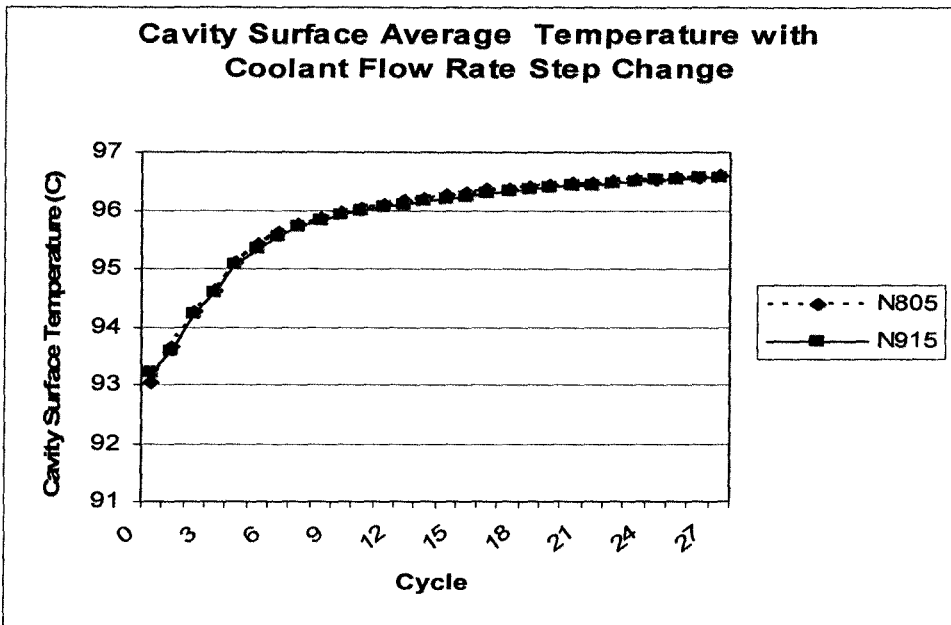


Figure 4.12 Response of cavity surface average cycle temperature to a step change in coolant flow rate

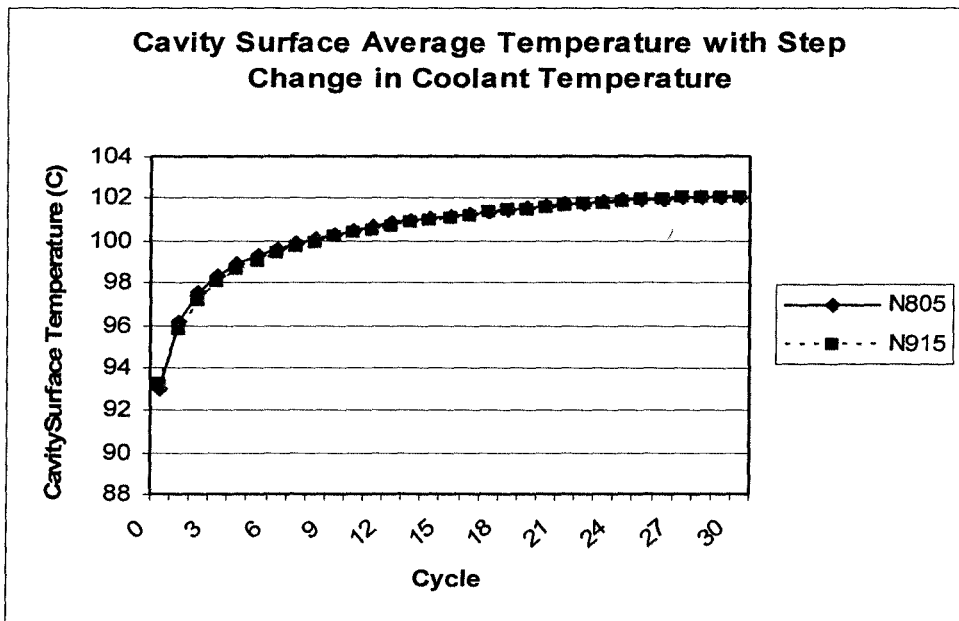


Figure 4.13 Response of cavity surface average cycle temperature to a step change in coolant temperature.

4.3.2 Controller Development

The Matlab[®] System Identification Toolbox was used to develop the ARX model. A second-order ARX model $n_a = 2$ and $n_b = 2$ with no delay was performed to identify the dynamic response to a step change in coolant temperature. The ARX model is given by,

$$(1 - 1.245q^{-1} + 0.3158q^{-2}) y(t) = (0.03265 + 0.03265q^{-1}) u(t)$$

where $y(t)$ is cycle average temperature and $u(t)$ represents coolant temperature. Similarly, a first-order ARX model with no delay was identified to describe the dynamic response to a step change in coolant flow rate.

$$(1 - 0.8085q^{-1})y(t) = -0.1339u(t)$$

Based on these models, the MPC controller code can be developed by using the Matlab[®] MPC Toolbox. Here we assume that the model perfectly reflects the process, so there is no model-process mismatch. **Table 4.2** lists the key tuning parameters for the MPC controller. A *blocking* function in Control Horizon (M) is used to minimize the frequent change in manipulated variable and make it more stable. For example, $M = [2 \ 3]$ defines two blocks and indicates that $u(k+1) = u(k)$ and $u(k+3) = u(k+2) = u(k+1)$.

From the above simulations, we found the steady-state temperature in the system (Node 805) was 93.08 °C. The coolant temperature setting is 75 °C and coolant flow rate 10 liters per minute. Set point change was then applied to the system in order to check the controller performance.

Table 4.2 Tuning parameters of surface temperature MPC controllers

Surface Temperature MPC Controller	Manipulated Variable	
	Coolant Temperature	Coolant Flow Rate
Controller Tuning Parameter		
Prediction Horizon (P)	30	30
Control Horizon (M)	[2 3]	[2 3]
Output Weight	1	1
Input Weight	0	0

4.3.3 Set-Point Control

For the MPC controller with coolant temperature as the manipulated variable, the surface cycle average temperature was set to 98 °C; **Figure 4.14** shows the controller response. **Figure 4.15** is the response with a set point in 88 °C. Similarly set point in 98 °C and 90 °C were given to the MPC controller with coolant flow rate as the manipulated variable. (**Figure 4.16-17**). Both MPC controllers are able to meet the target settings within 10 cycles, based on a closed-loop test and without considering any disturbance. However, more tests and experiments should be done to verify the controller performance, which is not included in this study.

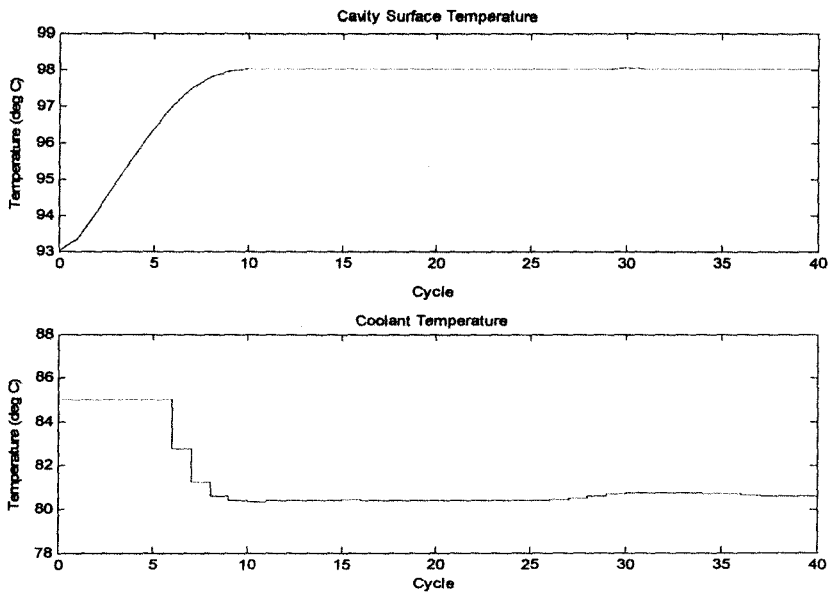


Figure 4.14 Response with set point change to 98 °C, using coolant temperature as manipulated variable.

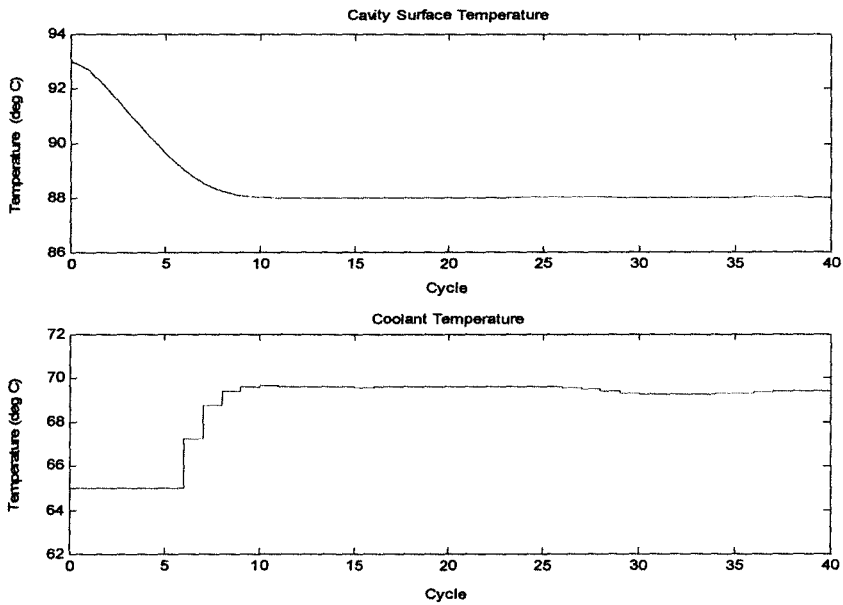


Figure 4.15 Response with set point change to 88 °C, using coolant temperature as manipulated variable.

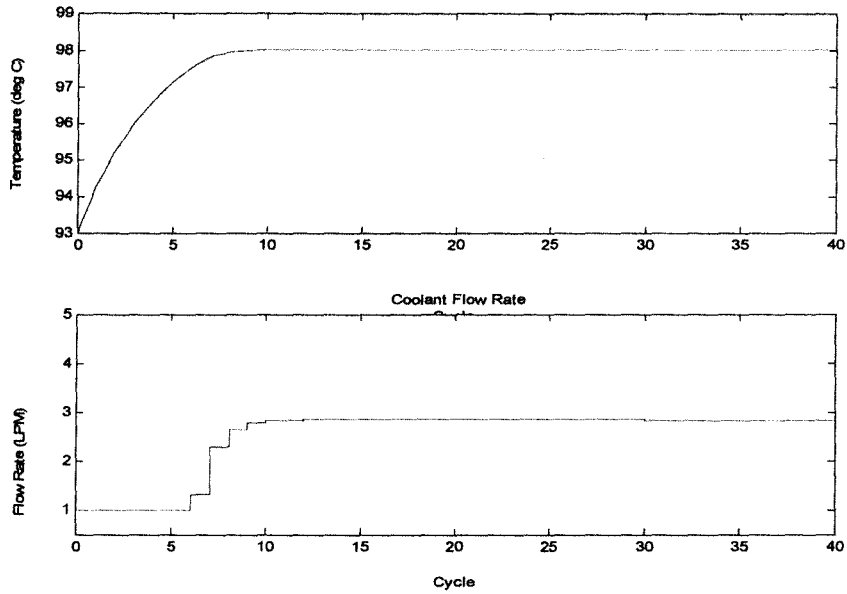


Figure 4.16 Response with set point change to 98 °C, using coolant flow rate as manipulated variable.

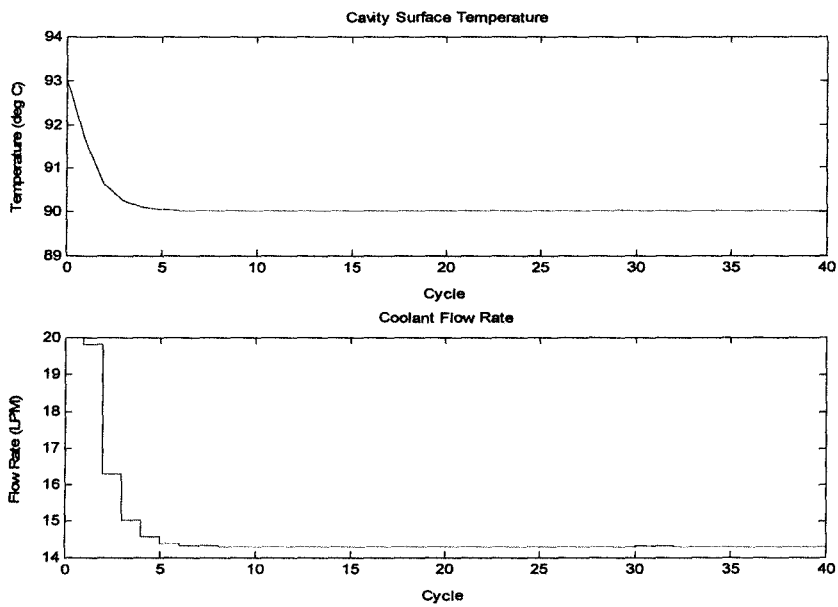


Figure 4.17 Response with set point change to 90 °C, using coolant flow rate as manipulated variable.

4.4 Multiple-input Multiple-output Controller

4.4.1 Controller Development

In this two-by-two MPC controller, the in-cycle cavity pressure and part surface temperature in packing stage are the controlled variables and the packing pressure and mold surface temperature are the manipulated variables. Step changes in the packing pressure and surface temperature are used to determine the coefficients of the dynamic matrix in the open-loop response of the controlled variables. Simulation results from Moldflow determine that the packing time should be set to 2 seconds, at which over 70% of the material has solidified. The packing pressure was set to 60 MPa and then stepped up to 70 MPa. The mold surface temperature was set to 85 °C, then stepped down to 75 °C. A total of four simulation tests were run with Moldflow[®] MPI. Cavity pressure and part surface temperature results were collected in a sampling time of 0.02 seconds. **Figure 4.18** shows the cavity pressure at Node:1838 in the packing stage, with different packing pressure settings and constant mold surface temperature at 85 °C. The cavity pressure profiles demonstrate a strong response to packing pressure change. However, when mold temperature was changed and packing pressure was kept constant, no difference was found in cavity pressure profiles in the packing stage when using the fusion meshing model for analysis in MPI. The fusion mesh model is a 2.5 dimensional domain meshing model, which may not accurate enough to simulate the pressure changes due to changes in mold temperature. A 3-dimensional model analysis usually gives more accurate results, so a 3D mesh model was used with the same process condition settings. **Figure 4.19** shows the pressure profiles with two different mold temperature settings and

the deviations of the two profiles. The difference is small and very nonlinear, which means the response of cavity pressure in packing stage to mold surface temperature change is very weak. As expected, a strong response was found in part surface temperature profiles to a change in mold temperature. **Figure 4.20** illustrates the part surface temperature profile in Node: 5092. However, there is no difference in part surface temperature profiles when packing pressure settings were changed. The reason may due to no heat transfer property are changed in the system. Based on the simulation results from MPI, the control loop pairing was selected as:

Cavity pressure in Packing (y1) ----- Packing pressure setting (u1)

Part surface temperature (y2) ----- Mold temperature (u2)

The interactions, mold temperature to cavity pressure in packing and packing pressure to part surface temperature, are very weak. It is theoretically reasonable not to consider the interactions in the controller design. Therefore, the dynamic matrix structure of the

MIMO controller is $A = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix}$ The mold temperature can not be changed within an

analysis case run in Moldflow[®] MPI v5.0. This limitation makes it difficult to get the dynamic model of the part surface temperature to a mold temperature change. Approximate data calculated from two simulation results ($T_{\text{mold}} = 75$ and 85 °C) were used to develop the model. Further validation and a model update are required before on-line application. The cavity pressure profiles, with a step change in packing pressure from 60 MPa to 70 MPa, were selected to develop the model. The cavity pressure and part surface temperature at the filling-to-packing switch-point were viewed as the initial status

of the system ($T_{\text{mold}} = 85 \text{ }^{\circ}\text{C}$). At this point, cavity pressure is 48 MPa and the part surface temperature is $102.5 \text{ }^{\circ}\text{C}$. The data were then imported into Matlab[®] System Identification Toolbox to develop ARX models.

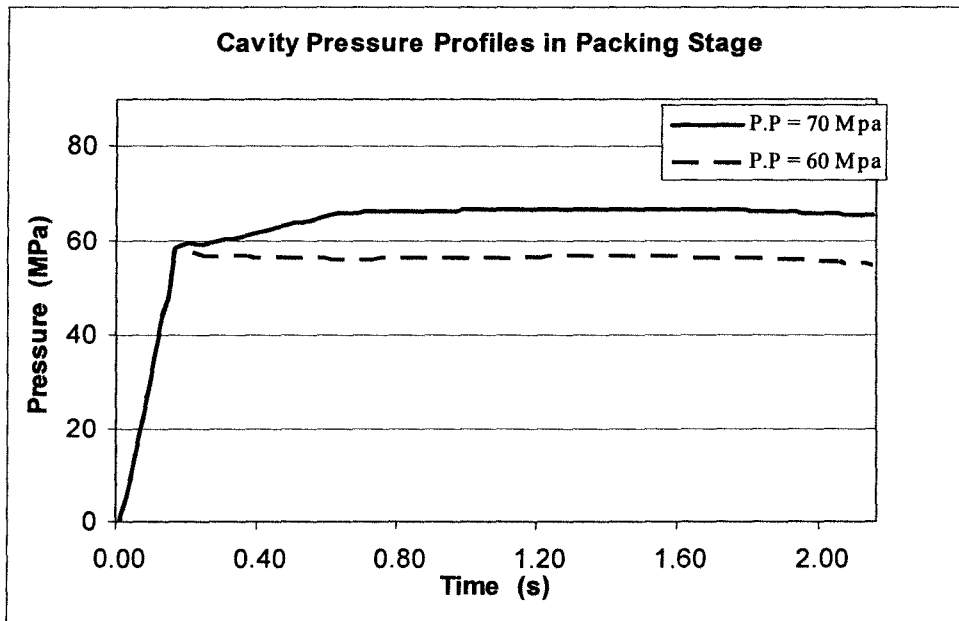


Figure 4.18 Cavity pressure in packing stage with different packing pressure settings.

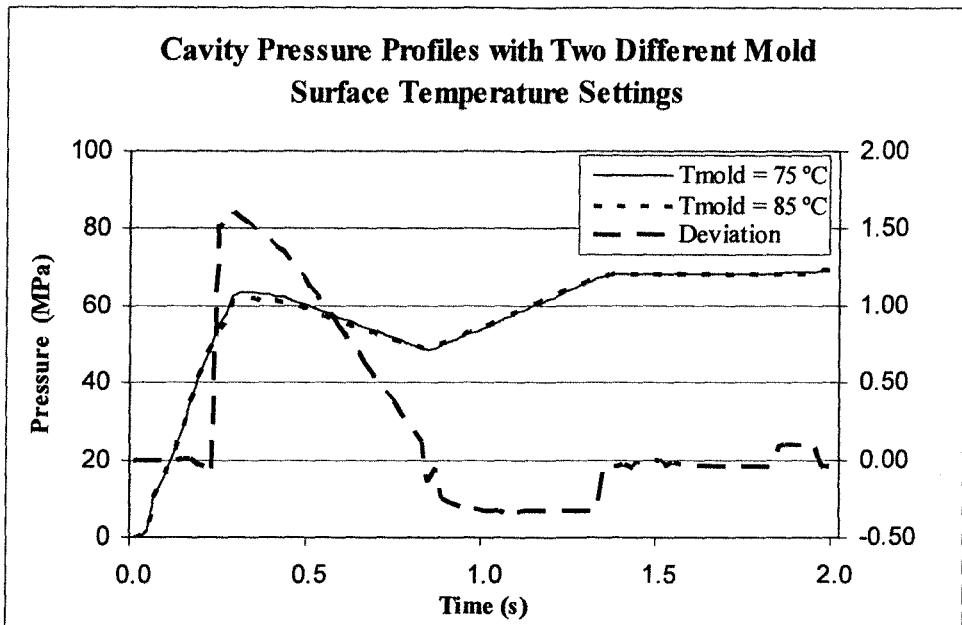


Figure 4.19 Cavity pressure in packing stage with two different mold temperature settings.

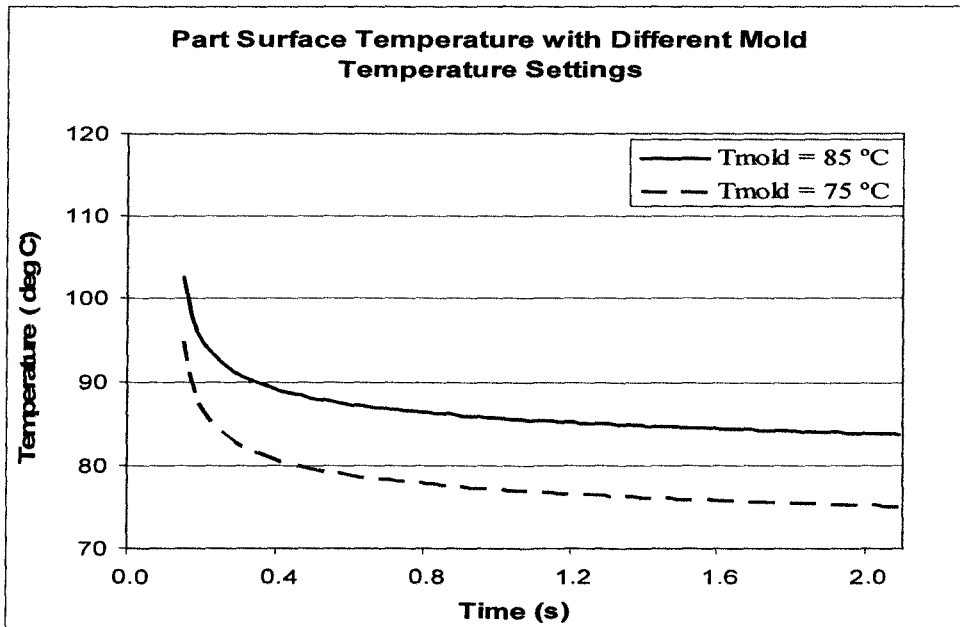


Figure 4.20 Part surface temperature with different mold surface temperature settings.

Two ARX models with $n_a = 3$, $n_b = 3$ and without delay were used to identify the models for y_1-u_1 and y_2-u_2 , which are given as:

y1-u1:

$$A(q) = 1 - 2.044 q^{-1} + 1.224 q^{-2} - 0.1722 q^{-3}$$

$$B(q) = 0.002826 + 0.002826 q^{-1} + 0.002826 q^{-2}$$

y2-u2:

$$A(q) = 1 - 1.068 q^{-1} + 0.1287 q^{-2} + 0.002569 q^{-3}$$

$$B(q) = 0.01822 + 0.01822 q^{-1} + 0.01822 q^{-2}$$

A two-by-two MPC controller was then developed by using Matlab[®] MPC Toolbox.

4.4.2 Set-Point Control

The controller was then implemented to control the process closed-loop without model/process mismatch. The control objective is the trajectory of the cavity pressure and part surface temperature in packing stage. In order to optimize the controller, the main tuning parameters were adjusted and the controller performance was compared. The biggest effect comes from manipulated variables weights (Λ). **Figure 4.21** shows the closed-loop response with manipulated variable weights $\lambda_1\lambda_2 = [0 \ 0]$. **Figure 4.22** displays the response with $\lambda_1\lambda_2 = [2 \ 5]$. In comparing the two results, the zero-weighted controller shows more aggressive (ringing) in manipulated variables adjustment, which is difficult to be fulfilled in reality, while the weighted system shows much better results in input control. On the other hand, the former system can meet the target set-point much faster than the second system, which may be due to the aggressive manipulation of the

inputs. In both control systems, the pressure control loop performs much better than the part surface temperature control loop. The main reason may be own to the much larger and faster temperature drop in the early stage than the rest of the time domain. The model is not able capture the entire dynamics of the process. The main tuning parameters are listed in **Table 4.3** and the code is in **Appendix E**.

Table 4.3 Main tuning parameter of the MIMO MPC controller.

MIMO MPC Controller Tuning Parameter	
Prediction Horizon (P)	40
Control Horizon (M)	4
Output Weight	[1 1]
Input Weight	[2 5]

In order to check the controller performance, an unmeasured disturbance step change is given to the system. An unmeasured disturbance may come from any aspect of the injection molding process, such as material change, machine property fluctuation, mold status and ambient conditions change. For example, different batch material may have a different melt viscosity, which may affect the cavity pressure development in the packing stage. In this case, the MPC controller will calculate and determine the adjustment to manipulated variable in order to move the cavity pressure back to its setting point. **Figure 4.23** shows an example for the closed-loop response of the system to a step change in unmeasured disturbance.

In general, the MPC controller performs very well to keep the process running in its set point. However, this is still simulation result and based on an assumption that the model perfectly reflects the process. Moreover, measured and unmeasured disturbances, which always exist in injection molding processes, were not involved during controller development. Due to the functional limitation and simulation accuracy of CAE software, the simulation results may differ from real production data. Therefore, model parameter and controller tuning parameter should be validated and updated, on-line, if needed before being applied for production control.

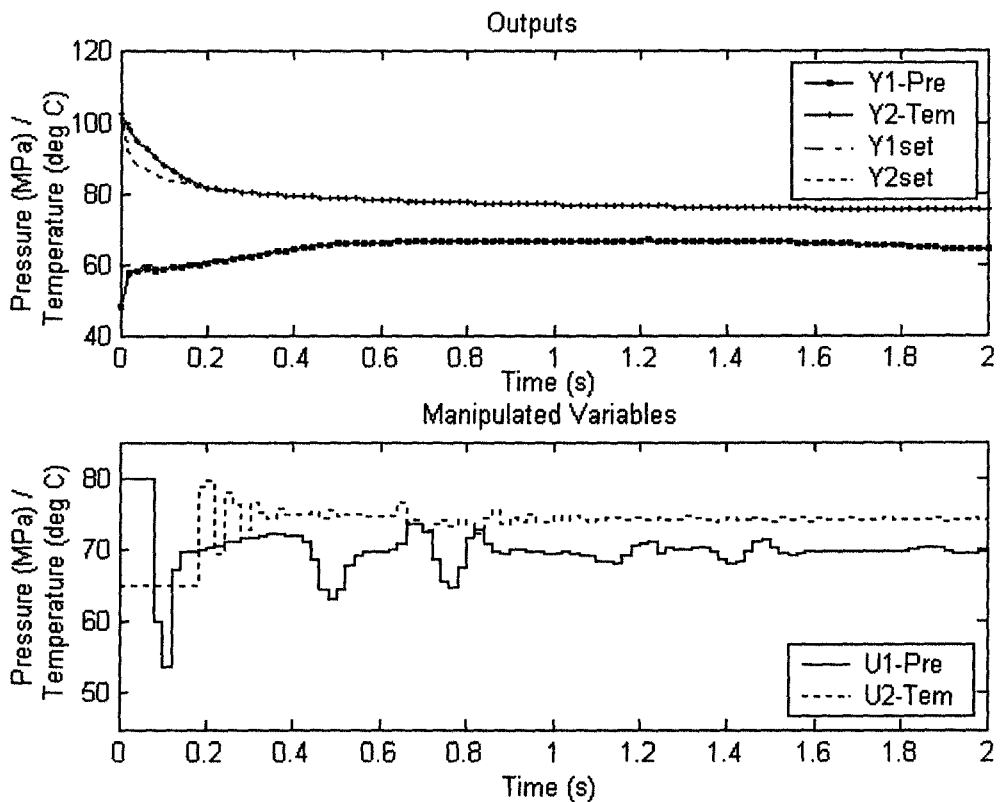


Figure 4.21 Closed-loop response with $\lambda_1\lambda_2 = [0 \ 0]$.

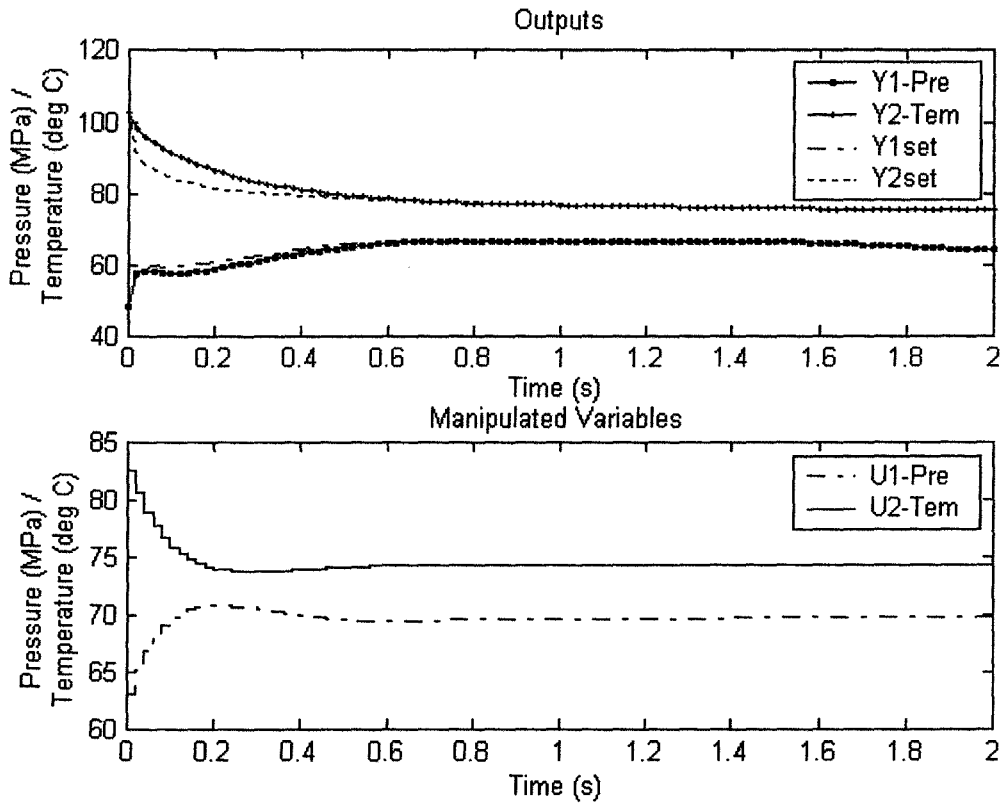


Figure 4.22. Closed-loop response with $\lambda_1\lambda_2 = [2 \ 5]$.

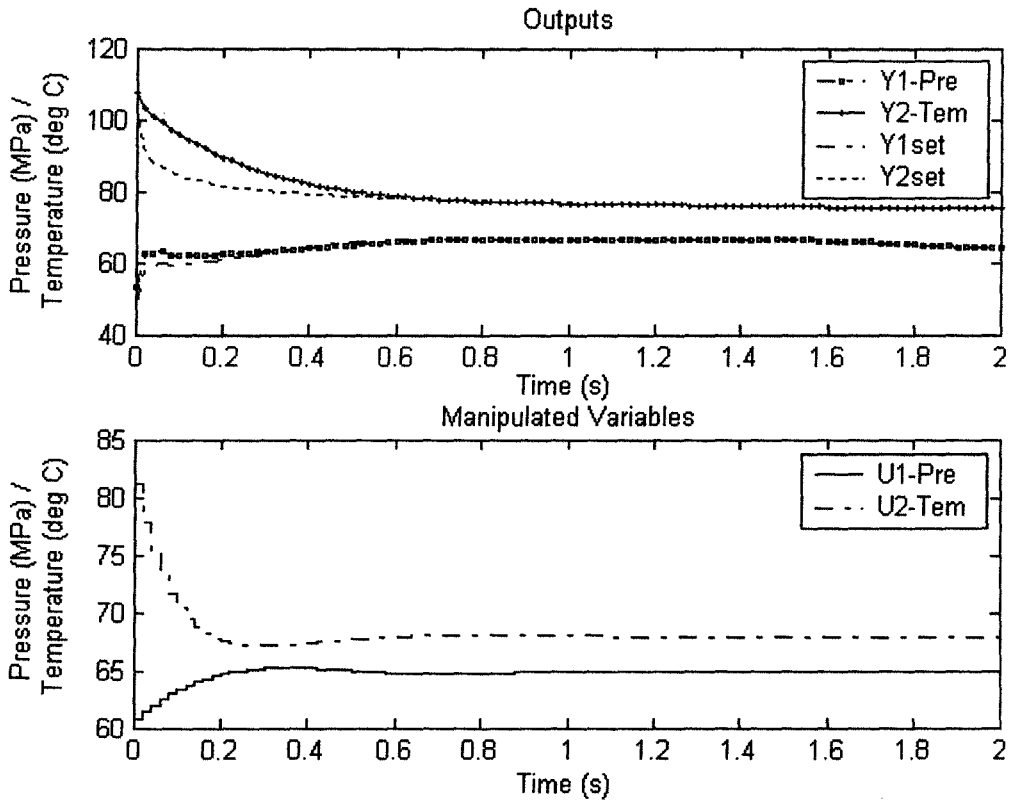


Figure 4.23 Closed-loop response to an unmeasured disturbance step change.

Chapter 5

Conclusions & Recommendations

5.1 Conclusions

This thesis work focused on using a DMC controller to control cavity pressure and cavity surface temperature, which have been proven to be the most important process variables. The target was to demonstrate the procedure to develop the MPC controller for process variables control with off-line data from CAE simulation software. A thin-wall cell phone front cover was designed and used as the studied parts. Moldflow Plastics Insight was used to simulate the injection molding process and ANSYS® FEM software was used to simulate the mold cooling process. The MPC programs are developed based on Matlab MPC Toolbox.

A SISO MPC controller is developed to control the cavity pressure in filling stage, using the injection flow rate as manipulated variable. This is in-cycle process control; the pressure trajectory is used as the control objective. In order to check the controller performance, Moldflow MPI was incorporated with a MPC controller to implement closed-loop control. The controller was proved to be able to control the cavity pressure very well and meet the target trajectory even with melt temperature or mold temperature change.

MPC controller was also used to control cavity surface temperature by manipulating the coolant flow rate and coolant temperature. A 2-dimensional model was developed to simulate the cooling process with ANSYS. A step change for coolant temperature or coolant flow rate was made to the steady state, which the mold temperature distributions are repeated in continuous cycles. Two SISO DMC controllers were then developed to control cavity surface cycle average temperature. Different set-points were given to the controller to check the closed-loop control response. Results show that the controller performs very well to trace the cycle average temperature set-point and the manipulation to the input is very smooth.

A two-by-two DMC controller was developed to control the cavity pressure and part surface temperature in packing stage, which are believed to greatly affect final part quality (e.g. part dimension, part weight, shrinkage and warpage). Packing pressure and mold temperature were manipulated variables. The control loop-pairing are selected as: packing pressure/cavity pressure and mold temperature/part surface temperature. Very strong interactions were found in every control loop-pairing. However, the interactions of mold temperature to packing pressure and packing pressure to part surface temperature are very weak. Due to the functional limitations of the CAE software, approximate data were calculated from simulation results to develop the model, which must be updated with production data before application. The controller performs well in meeting the cavity pressure set-point, but for part surface temperature control, large deviations are observed. The possible reason may be due to the strong nonlinearity in the early part of the packing stage. Closed-loop implementation of the simulated case was not carried out

due to CAE software limitations, but an on-line controller could be developed using production data on the machine with the model as a starting point.

With the application of CAE simulation software, MPC controllers can be developed efficiently by just using the part and mold design. This will save time in the new mold trial and error period and provide useful guides to help operators to set up the process. The simulation results may also help to predict design problems. However, even though current CAE software has improved in simulation accuracy, the dynamic model parameters and controller tuning parameters should be validated with production data before on-line application. On-line model parameter identification is highly recommended to improve controllability for processes with strong nonlinearity and time-variance.

5.2 Future Work

The objective for the future work is to further develop the multivariable model-based predictive controller for injection molding process control, based on simulation results. A few research directions can be investigated in order to improve the controllability for the process.

- 1. Use quality variable as control objective for MPC controller.** The two-by-two MPC controller can be further developed to control quality variables (e.g. part weight, density, shrinkage, warpage and part dimensions), which are related to cavity pressure and part temperature. Among quality variables, density (the inverse of specific volume), is directly affected by packing pressure and mold temperature. Density can be predicted when the cavity pressure and bulk melt temperature are known by using the material PVT

data. Bulk melt temperature has been studied and proven to be estimated very well by measuring part surface temperature (Verela et al., 1996; Kamal et al., 1999). Given a target density profile, it is possible to control the cavity pressure and surface temperature profiles to follow specific trajectories. Therefore, to use a predefined density as a control objective, the multivariable MPC controller will be able to control the cavity pressure and bulk temperature profile in order to meet this in-cycle target. Then the final part quality will be more stable. The challenge is to investigate the relationship between cavity pressure and bulk temperature profile based on PVT data, considering the effect on cycle time, clamping force requirement and production cost.

2. MPC controller implementation The proposed MPC controller controls in-mold process variables by manipulating processing parameters (e.g. injection flow rate, packing pressure) not a direct signal, such as the voltage to control the opening of hydraulic servo-valve. The idea is to develop a standard MPC controller module, which can communicate with molding machine and carry out online process control without modifying the machine hydraulic or electrical system. However, there are still many problems that have to be solved first, such as the communication between MPC controller and injection molding machine, the response time of the system and so on.

3. Develop 3-dimensional model for thermal analysis. In this study, 2-dimensional model was used to simulate the cooling stage with FEM software. However, using 3-dimensional model will be able to provide more information about the temperature transient and distribution in the mold, and the simulation results are more accurate than that of the 2-dimensional model.

References

- Abu-Ayyad, Ma'moun, Dubay, R. "Development of an Extended Predictive Controller for Injection Speed". *SPE ANTEC Technical Papers 2007*: 680-684
- Agrawal, A. R.; Pandelidis, I.O. and Pecht, M. "Injection Molding Process Control: A Review" *Polym. Eng. Sci.* 27 (18), 1987: 1345-1357.
- Ainoya, K., and Amano, O. "Accuracy of Filling Analysis Program." *SPE ANTEC Technical Papers 2001*: 726-735
- Ambady, Prasanth, and Kazmer, David. "Model predictive Control of Injection Molding." *SPE ANTEC Technical Papers 1* (2001): 1-5
- ANSYS Inc. "ANSYS Documentation Release 11.0", 2007.
- Astom, K. J., and Wittenmark, B. *Computer controlled systems: theory and design*, 3rd ed. Prentice Hall, c1990
- Bernard, Sanschagrin. "Process Control of Injection Molding." *Polymer Engineering and Science* 23 (8), 1983: 431-438.
- Buja, F. J. US Patent 4904172, 1990.
- Bur, A. J. and Thomas, C. L. "Fluorescence Monitoring of Polymer Injection Molding: Model development." *Polym. Eng. Sci.* September 37 (9), 1997:1430-1441
- Burley, A Noel, "Nicrosil/Nisil Type N Thermocouples" *Omega website* Internet. Available from <http://www.omega.com/temperature/Z/pdf/z041-044.pdf> (Accessed 05 July, 2007)
- Chang, T. C. "Robust Process Control in Injection Molding—Process Capability Comparison for Five Switchover Modes" *Journal of Injection Molding Technology* 6 (4), 2002: 239-246.
- Chen, Shia-Chung and Chung, Yung-Chien. "Simulations of cyclic transient mold cavity surface temperatures in injection mold-cooling process." *International Communications in Heat and Mass Transfer*, 19 (4), July-August 1992: 559-568
- Chen, Shia-Chung; Hu, Sheng-Yan and Jong, Wen-Ren "Integrated Simulations of Structural Performance, Molding Process, and Warpage for Gas-Assisted Injection-

- Molded Parts III: Simulation of Cyclic, Transient Variations in Mold Wall Temperatures." *Journal of applied Polymer Science*, 71 (2), 1999: 339-351
- Chen, S.C.; Peng, H.S.; Huang, L.T.; Yuan, C.; Chang, C.T.; Wei, D.Y.; Ruo, R.C. and Cheng, C. "Thin-Wall Injection Molding" *SPE ANTEC Technical Papers 2000a*:543-547
- Chen, Shia-Chung; Hu, Sheng-Yan; Chao, Sher-Meng and Chien, Rean-Der. "Simulation of a mold-cooling process for gas-assisted injection molded parts designed with a top rib on the gas channel." *Polym Eng. & Sci.* 40 (3), March 2000b: 595-606
- Chen, S.C.; Liao, W.L.; Peng, H. S.; Huang, L.T. and Chang, Y. F. "Characteristics of Mould Separation and Its Effects on Qualities of Thin Wall Injection Moulded Parts." *Plastics, Rubber and Composites*, 31 (1), 2002: 12- 19
- Chen, Shia-Chung; Chau, Shiu-Wu; Lin, Ming-Cheng and Lin, Yu-Wan. "Three-Dimensional Simulation of Transient Temperature Distribution for Lens Mold Embedded with Heaters." *SPE ANTEC Technical Papers*, 2004a: 631-635.
- Chen, Shia-Chung; Li Hai-Mei; Shen Chang-Yu; Jiang Zhi-Liang, and Huang Shun-Chih. "3D Simulation and Verification for Mold Temperature Control Technologies." *SPE ANTEC Technical Papers 2005*: 491-495.
- Chen, Shia-Chung; Wang, I-Chao; Chiou, Yan-Chen; Chou, Ya-Yuen and Yang, Wen-Hsien. "An Investigation on the Temperature Behavior in Mold Embedded with Heater." *SPE ANTEC Technical Papers 2007*: 718-722.
- Chen, X.; Gao, F and Chen, G. "Capacitive transducer for in-mold monitoring of injection molding." *Polym. Eng. Sci.* 44(8), August 2004b:1571-1578
- Chen, Zhongbao and Turng, Lih-Sheng. "Online Adaptive Injection Molding Process and Quality Control." *SPE ANTEC Technical Papers 3 (2005a)*: 30-34.
- Chen, Zhongbao and Turng, Lih-Sheng. "A Review of Current Developments in Process and Quality Control for Injection Molding." *Advances in Polymer Technology* 24 (3), 2005b: 165-182

- Chen, Zhongbao and Turng, Lih-Sheng. "Adaptive Online Quality Control for Injection – Molding by Monitoring and Controlling Mold Separation." *Polym. Eng. Sci.* 46 (9), 2006:569-580.
- Chen, Zhongbao and Turng, Lih-Sheng. "Injection Molding Quality Control by Integrating Weight Feedback into Cascade Closed-loop Control System." *Polym. Eng. Sci.* 47 (6), 2007: 852-862
- Cutler, C., and Ramaker, B. "Dynamic Matrix Control- A computer control algorithm." *AIChE Nat. Meet.*, April, 1979.
- Dontula, N.; Sukanek, P.C.; Devanathan, H. and Campbell, G. A." An experimental and theoretical investigation of transient melt temperature during injection molding" *Polym. Eng. Sci.* 31 (23), 1991: 1674-1683
- Dubay, R. "Predictive Control of Cavity Pressure During injection Filling". *Journal of Injection Molding Technology* 5(2), June 2001:72-79
- Dubay, R. "Self-optimizing MPC of Melt Temperature in Injection Moulding". *The Instrumentation, System, and Automation Society Transactions* 41 (2002): 81-94
- Dubay, R.; Bell, A. C. and Gupta, Y. "Control of plastic melt temperature: A multiple input multiple output model predictive approach." *Polym. Eng. Sci.* 37(9), 1997: 1550-1563
- Dubay, R.; Healy, A. and Gerber, A. "Controlling Melt Temperature In Injection Molding Using An Adaptive CFD Predictive Controller." *SPE ANTEC Technical Papers* 2003: 3300-3304
- Dubay, Rickey; Hernandez, J. M. and B. Pramujati. "A Generic Controller for Part Surface Temperature on a plastic Injection Molding Machine." *Advances in Polymer Technology* 25 (1), 2006: 1-12.
- Dubay, Rickey; Pramujati, Bambang; Han, Jianguo and Strhmaier, Franz. "An Investigation on the Application of predictive Control for Controlling Screw Position and Velocity on An Injection Molding Machine." *Polym. Eng. Sci.* 47(4), 2007: 390-399

- Edwards, R, and Thomas, C. L. "On-line Measurement of Polymer Orientation Using Ultrasonic Technology." *Poly. Eng. Sci.* 41 (9), Sept. 2001a: 1644-1653
- Edwards, R.; Thomas, C. L. and Peter, R. "Controlling Injection Phase/Packing Phase Switchover Using an Ultrasonic Sensor." *SPE ANTEC Technical Papers* 2001b: 949-952
- Edwards, R.; Thomas, C. L. and Diao, L.Y. "A Comparison of Position, Cavity Pressure, and Ultrasonic Sensors for Switch/Over Control in Injection Molding." *SPE ANTEC Technical Papers* 2003: 586-590
- Fara, D. A.; Rawabdeh, I. and DIA ABU-AL-NADI. "Neural Network Control for Cavity Pressure During Filling and Packing Stages of the Thermoplastics Injection Molding Process" *Journal of Injection Molding Technology*, 5(2), June 2001: 105-119
- Gao, F.; Patterson, W. I. and Kamal, M. R. "Cavity Pressure Dynamics and Self-Tuning Control for Filling and Packing Phases of Thermoplastics Injection Molding." *Polym. Eng. Sci.* 36(9), 1996a: 1272-1285.
- Gao, F.; Patterson, W. I. and Kamal, M. R. "Cavity Pressure Control During the Cooling Stage in Thermoplastic Injection Molding." *Polym. Eng. Sci.* 36(19), 1996b: 2467-2476.
- Gao, F., Patterson, W. I. and Kamal, M. R. "Self-tuning Cavity Pressure Control of Injection Molding Filling" *Advances in Polymer Technology* 13(2), 1994: 111-120.
- Gao, F.; Yang, Y. and Shao, C. "Robust iterative learning control with applications to injection molding process" *Chemical Engineering Science* 566 (24) 2001: 7025-7034
- Gao, F.; Li, Mingzhong; Wang, Fuli; Wang, Baoguo and Yue, Pollock. "Genetic Algorithms and Evolutionary Programming Hybrid Strategy for Structure and Weight Learning for Multilayer Feedforward Neural networks." *Industrial and Engineering Chemistry research* 38 (11), November 1999: 4330-4336.
- Gao, F.; Yang, Yi and Shao, Cheng. "Robust Iterative Learning Control with Applications to Injection Molding process." *Chemical Engineering Science* 56 (24), December 2001: 7025-7034.

- Garcia C. E. and Morshedi A. M. "Quadratic Programming Solution of Dynamic Matrix Control (QDMC)". *Chem. Eng. Community*, 46 ,1986: 73-87
- Gerber, A.; Healy, A. and Dubay, R. "CFD-based Predictive Control of Melt Temperature in Plastics Injection Molding." *Applied Mathematical Modeling* 30, 2006: 884-903
- Groleau, M. R., and Groleay, R. J. "Comparing Cavity Pressure Sensor Technologies Using In-Mold Data" *SPE ANTEC Technical Papers* 2002
- Guzmer, J. L.; Berenguel, M. and Dormindo, S. V. "Interactive teaching of constrained generalized predictive control - a tool for teaching basic concepts " *IEEE Control Systems Magazine*, Apr. 2005: 52-66
- Haber, A., and Kamal. M.R. "The Dynamic of Peak Cavity Pressure in Injection Molding" *Polym. Eng. Sic.* 27 (18), Mid-Oct.1987: 1411-1418
- Havlicsek, H. and Alleyne, A. "Nonlinear control of an electrohydraulic injection molding machine via iterative adaptive learning" *IEEE/ASME Transaction Mechatron* 4(3), 1999: 312-323.
- He, W.; Zhang, Y.F.; Lee, K.S.; Fuh, J.Y.H. and Nee, A.Y.C. "Automated process Parameter Resetting for Injection Moulding: A Fuzzy –neuro Approach." *Journal of Intelligent manufacturing* 9 (1), February 1998
- Hernandez, J. M.; Dubay, Rickey and Pramujati, B. "Cavity Temperature Control in Injection Molding Machine." *SPE ANTEC Technical Papers* 2005: 714-718.
- Hernandez, J. M.; Dubay, Rickey and Pramujati, B. "Cavity Temperature Control During the Cooling Cycle in An Injection Molding Machine Using A predictive Controller." *SPE ANTEC Technical Papers* 2007 (a) 2514-2518.
- Hernandez, J. M.; Dubay, Rickey and Pramujati, B. "Cavity Temperature Control in Injection Molding Machine." *International Polymer Processing XXII* (2007b) 3: 276-283.
- Huang S.N., Tan K.K., and Lee T.H. "Neural –network-based Predictive Learning Control of Ram Velocity in Injection Molding." *IEEE Transaction on Systems, Man and Cybernetics Part C: Applications and Reviews* 34 (3), 2004: 363-368.

- Johnson, B.P. "The Use of a Design of Experiments (Doe) to Optimize Processing Conditions of an Injection Molded Gear". *SPE ANTEC Technical Papers* 2002, No: 771
- Jaworski, M. J., and Yuan, Z. "Theoretical and experimental comparison of the Four Major Types of mesh Currently Used in CAE Injection Molding Simulation Software". *SPE ANTEC Technical Papers*, 2003: 642-646.
- Kalnin, Flavio A., and Zluhan, Gilberto P. "Injection Mold's Problems Solved by CAE analysis." *SPE ANTEC Technical Papers* 1999: 675-679.
- Kamal, M. R., Varela, A. E., and Patterson, W. I. "Control of part weight in injection molding of amorphous thermoplastics". *Polym. Eng. Sci.*39 (1999): 940-952
- Kazmer, David and Philip, Barkan. "Multi-Cavity Pressure Control in the Filling and Packing Stages of the Injection Molding Process." *Polym. Eng. Sci.*37 (11), 1997a:1865-1879.
- Kazmer, David and Philip, Barkan. "The Process Capability of Multi-Cavity Pressure Control for the Injection Molding Process." *Polym. Eng. Sci.*37 (11), 1997b:1880-1895.
- Kazmer, D., and Petrova, T. "Hybrid neural models for pressure control in injection molding" *Advances in Polymer Technology* 18(1), 1999: 19-31
- Kazmer, D.; Fan, B. F. and Mukhari, R. "Real Time Flow Rate Estimation in Injection Molding," *Website <http://kazmer.uml.edu/Staff/papers>. Conference papers*, June 21, 2003a
- Kazmer, D.; GAO, R. X.; Theurer, C. B. and Zhang, L. "Wireless Pressure Sensor for Injection Molding." *SPE ANTEC Technical Papers* 2003b: 3290-3294
- Kazmer, D.; Gao, R.; Cui, Y.; Johnson, S. and Knepper, P. "Concept Design of A Wireless Pressure, Temperature, and Flow Rate Sensor for Injection Molding." *SPE ANTEC Technical Papers* 2005b: 3640-3644.
- Kazmer, D.; Knepper, P., and Johnston, S. A Review of In-mold Pressure and Temperature Instrumentation." *SPE ANTEC Technical Papers* 2005a: 3300-3304

- Kening, S.; Ben-David, A.; Omer, M. and Sadeh, A. "Control of Properties in Injection Molding by Neural Networks." *Engineering Applications of Artificial Intelligence* 14 (6), December 2001: 819-823.
- Kim, Seung J.; Lee, K. and Kim, Y. I. "Optimization of Injection Molding Conditions using Genetic Algorithm." *Proceedings of SPIE- The International Society for Optical Engineering* 2644, (1996): 173-180.
- Kim, J. G.; Shin, I. H.; Han, W. H. and Lee, J. W. "Injection Molded Part Quality Prediction Using a New Method of Analysis" *Journal of Injection Molding Technology*. 4 (4), Dec 2000: 201-210
- Kim, Sang-Woo and Turng, Lih-Sheng. "Three-Dimensional Numerical Simulation of Injection Molding Filling of Optical Lens and Multiscale Geometry Using Finite Element Method." *Polym. Eng. & Sci.* 46 (9), 2006: 1263-1274
- Koelling, W, and Xu, G.J. "Study of Cavity Pressure and its Prediction during Injection Molding." *SPE ANTEC Technical Papers* 1 (2003): 566-570
- Lakhrum, R, and Dubay, R. "Self-optimizing MPC of Injection Velocity during mold filling." *SPE ANTEC Technical Papers* 2002
- Lee, P.L. and Sullivan, G.R. "Generic model control (GMC)", *Computer Chemical Engineering*, 12(6), 1988:573-580.
- Lau, H. C. W.; Wong T. T. and Pun K.F. "Neural-Fuzzy Modeling of Plastic Injection Molding Machine for Intelligent Control." *Expert Systems with Applications* 17(1), July 1999: 33-43.
- Liang, Jui-Ming, and Wang, Pei-jen. "Self-learning Control for Injection Molding Based on Neural Networks Optimization." *Journal of Injection Molding Technology* 6(1), March 2002: 58-71
- Ljung, Lennart. "System Identification Toolbox User's Guide." Release 13. The Math Work Inc. 2002
- Maciejowski, J. M. "Predictive Control with Constraints". Prentice Hall, 2002
- Marlin, Thomas E. "*Process Control: Designing process and Control Systems for Dynamic Performance*". McGraw-Hill Higher Education, 2000.

- Menges, George; Michaeli, Walter, and Mohren, Paul. *"How to Make Injection Molds"*. Hanser Publishers, Munich, 2001.
- Michaeli, Walter and Gruber Juan. "Increasing quality by online Control of the Cavity Pressure." *SPE ANTEC Technical Papers* 1 (2004):688-692.
- Michaeli, Walter and Gruber, Juan. "Increasing quality by Online Control of the Cavity Pressure." *SPE ANTEC Technical Papers* 2004: 688-692
- Morari, Manfred and Ricker, N. Lawrence. "Model Predictive Control Toolbox User's Guide." Release 13. The MathWork Inc. 2002
- Orzechowski, S; Paris, A and Dobbin, C. "A Process Monitoring and Control System for Injection Molding Using Nozzle-Based Pressure and Temperature Sensors" *Journal of Injection Molding Technology* 2(3), 1998: 141-148
- Ouyang G.; Li, X.; Guan, X.; Zhang, Z.; Zhang, X., and Du, R. "Ram Velocity Control in Plastic Injection Molding Machines with Neural Network Learning Control." *Lecture Notes in Computer Science* 3174 (2004): 169-174.
- Patterson, W. I.; Kamal, M.R. and Gao, F. "Mold Temperature Measurement and Control." *SPE ANTEC Technical papers* 1 (1990):227-232.
- Pierick, D. and Noller, R. "The Effect of Processing Condition on Shrinkage" *SPE ANTEC Technical Papers* 1991: 252-258
- Pramujati, B. and Dubay R. "Processing Identification of Barrel Temperature in Injection Molding: A MIMO Approach." *SPE ANTEC Technical Papers* 2003: 3560-3563
- Pramujati, B., Dubay, R. and Han, J.G. "Injection Velocity Control in Plastic Injection Molding." *SPE ANTEC Technical Papers* 2005: 734-738
- Pramujati, B.; Dubay R. and Samaan, C. "Cavity Pressure Control During Cooling in Plastic Injection Molding." *Advances in Polymer Technology* 25(3), 2006a: 170-181
- Pramujati, B.; Dubay, R. and Samaan, C. "Coolant Flow Rate Manipulation for Cavity pressure Control" *SPE ANTEC Technical Papers* 2006b:1265-1269
- Qin, S. Joe and Badgwell, Thomas A. "A survey of Industrial Model Predictive Control Technology." *Control Engineering Practice* 11 (2003): 733-764.

- Rafizadeh, M., Patterson, W. I. and Kamal, W. R. "Physically-Based Model of Thermoplastics Injection Molding for Control Applications". *Archive International Polymer Processing* 1996/04: 352-362.
- Rafizadeh, M; Patterson, W. I. and Kamal, W. R. "Physically-Based Adaptive Control of Cavity Pressure in Injection Molding: Filling Phase." *SPE ANTEC Technical Papers* 1 (1997):582-587.
- Rafizadeh, M; Patterson, W. I. and Kamal, W. R. "Physically-Based Adaptive Control of Cavity Pressure in Injection Molding: Packing Phase." *Iranian Polymer Journal (English Edition)* 8 (2), 1999a:99-113
- Rafizadeh, M; Patterson, W. I. and Kamal, W. R. "Physically Based Dynamic Model for Control of Cavity pressure in Injection Moulding of Amorphous Plastics." *Plastics, Rubber and Composites* 28 (2), 1999b: 47-61
- Rewal, N.; Toncich, A. and Friedl, C. "Predicting Part Quality in Injection Molding Using Artificial Neural Networks" *Journal of Injection Molding Technology* 12 (3), 1998: 109-119.
- Reynolds, D.W. "Optimization of Cooling Circuits in Injection Molding." Thesis. McMaster University Library: TP 1150.R49 2000 c.2 (July 2000)
- Ribeiro, B. "Fault detection in a Thermoplastic Injection Molding Processing Using Neural Networks." *Proceedings of the International Joint Conference on Neural Networks* 5 (1999):3352-3355.
- Rumelhart, R. C. and McClelland, J. L. *Parallel Distributed process: Exploration in the Microstructure of Cognition: 1. Foundations*, Mit Press. Cambridge. Mass. (1986)
- Schenker, J. R. "Using Adductive Induction to Create A Math Model of An Injection Molding Process". *SPE ANTEC Technical Papers*. 39 (1993): 2137-,
- Seborg, Dale E., Edgar, Thomas F. and Mellichamp, Duncan A. *Process Dynamics and Control*. John Wiley & Sons, Inc. 2003.
- Selden, R. "Thin Wall Molding of Engineering Plastics- A literature Review." *Journal of Injection Molding Technology*, 4(4), December 2000: 159-166.

- Shay, R.M.; Poslinski, A.J and Fakhreddine, Y. “Estimating Linear Shrinkage of Semicrystalline Resins from Pressure-Volume-Temperature (PVT) Data” *SPE ANTEC Technical Papers*, 1998
- Sheth, B.; Higdon, C. M. F.; Schott, N. R.; Barry, R. D. and Davison, B. “Improved Part Quality Using Cavity Pressure Switchover” *SPE ANTEC Technical Papers* 2001
- Sheth, H. R. and Nunn, R. E. “An Adaptive Control Methodology for the Injection Molding Process. Part 2: Experimental Application” *Journal of Injection Molding Technology*, 5 (3), 2001: 141-151
- Speight, R.G.; Yazbak, E. P. and Coates, P. D. “In-line Pressure and Infrared Temperature measurements for Injection Moulding process Control.” *SPE ANTEC Technical Papers* 1 1995: 647-651.
- Stergiou, C. and Siganos, D. Internet. Available from:
http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html (Accessed on 05 July, 2007)
- Swartz, C. L. E. “Model predictive Control – Formulation , properties and Applications.” Material presented in Sastech Process Control and Instrumentation Symposium, Secunda, Oct 14-15, 1991.
- Swartz, C. L. E. “Introduction to Model Predictive Control”, Industrial Short Course, Burlington, Onatario, 2004
- Tan, K.K.; Huang, S.N. and Zhao, S. “A Novel Predictive and Iterative Learning Control Algorithm.” *Control and Intelligent Systems* 31(1), 2003: 1-9.
- Tang, S.H.; Kong, Y.M.; Sapuan, S.M.; Samin, R. and Sulaiman, S. “Design and Thermal Analysis of Plastic Injection Mould.” *Journal of Materials Processing Technology* 171 (2006): 259-267.
- Tsai, Ching-Chih and Lu Chi-Huang. “Multivariable Adaptive Predictive PI Temperature Control of a Plastic Injection Molding Process.” *Journal of Control System and Technology*, 5 (March 1997), 1: 11-24.

- Tsoi, Hoi-Pang and Gao, F. "Control of Injection Velocity Using a Fuzzy logic Rule-based Controller for Thermoplastics Injection Molding." *Polym. Eng. Sci.* 39 (1), 1999: 3-17.
- Turng, L. S. and Chen, Z. B. "A review of current developments in process and quality control for injection molding" *Advances in Polymer Technology* 24 (3), 2005: 165-182
- Vagelatos, G. A.; Rigatos, G. G. and Tzafestas S.G. "Incremental Fuzzy supervisory Controller Design for Optimizing the Injection Molding Process." *Expert Systems with Applications* 20 (2), February 2001: 207-216.
- Varela, A.E.; Kamal, M.R.; and Patterson, W.I. "A method for Estimating Bulk Melt Temperature and Part Weight in Injection Molding of Amorphous Thermoplastics" *Advances in Polymer Technology* 15 (1), 1996: 17-28
- Vaughan, Robert. "Pressure Sensors: The nerve System of molding Process." *RJG Website* 2003. Internet. Available from (Accessed, 05 July 2007): http://www.rjg-ptm.com.cn/Document/Articles/mold_maker_paper.pdf.
- Wang, Baoguo; Gao, F. and Yue, Polock. "Neural Network Approach to Predict Melt Temperature in Injection Molding Processes." *Chinese Journal of Chemical Engineering* 8 (4), November 2000a: 326-331.
- Wang, Fan; Dong, Shaoqiang; Danai, Kourosh and Kazmer, David O. "Input Profiling for Injection Molding by Reinforcement Learning." *American Society of Mechanical Engineers, Dynamic Systems and Control Division (Publication) DSC* 70, (2002): 701-708.
- Wang, H.; Cao, B.; Jen, C. K.; Nguyen K. T. and Viens, M. "On-line Ultrasonic Monitoring of the Injection Molding Process." *Polym. Eng. Sci.*, 37(2), Feb.1997: 363-376
- Wang, K. K. ; Hiebbber, C. A. ; Wang, N. ; Zhou, J. ; Lee, M. C. and Sakurai, Y. *In Cornell Injection Molding Program, Progress Report* 21, 2000b;
- Wang, K. K.; Zhou, J. and Sakurai, Y. "An Integrated Adaptive Control for Injection Molding." *SPE ANTEC Technical Papers* 1999: 611-615

- Wang, Lixia; Shen, Changyu; Li, Qian and Chen Jinbo. "Application of CAE Technique in Process Optimization and Part Quality Control of Injection Molding." *SPE ANTEC Technical Papers* 2005: 652-656.
- Wang, T. James and Yoon, C.K. "Effects of Process Conditions on Shrinkage and Warpage in the Injection Molding Process." *SPE ANTEC Technical Papers* 1999:584-588
- Wen S.-S.L.; Jen, C.-K.; Nguyen, K.T.; Derdouri, A. and Y. Simard. "Recent Advances in Ultrasonic Monitoring of the Injection Molding Process". *SPE ANTEC Technical Papers* 1999: 636-640
- Wenskus, J. and Miller, A. B. US Patent 5063008, 1991.
- Wittenmark, Bjorn, "Self-Tuning Regulators" (1998) Lund Institute of Technology (Lund University). Internet. Available from: <http://www.control.lth.se/~bjorn/wit98.html> (Accessed on 05 July, 2007)
- Woll, S. L. B. and Cooper, D. J. "Pattern-based Closed-loop Quality Control for the Injection Molding Process." *Polym. Eng. Sci.* 37 (5), 1997: 801-812
- Woll, Suzanne L. B.; Cooper, Douglas J and Souder, B. V. "Online pattern-based part quality monitoring of the injection molding process". *Polym. Eng. Sci.* 36 (11), June 1996
- Xu, Haoyu and Kazmer, David. "A Study of Cooling for Steady-State Injection Molding." *SPE ANTEC Technical Papers*, 1999: 422-426
- Yang, Y and Gao, F. "Adaptive Control of Nozzle Melt Packing Pressure." *SPE ANTEC Technical Papers* 1998: 472-476
- Yang, Y and Gao, F. "Injection velocity Control Using A Self-tuning Adaptive Controller." *Archive International Polymer Processing* 1999 (a)/02: 196-204
- Yang, Y and Gao, F. "Cycle-to-Cycle and Within-Cycle Adaptive control of Nozzle Pressure During Packing-Holding for Thermoplastic Injection Molding." *Polym. Eng. Sci.* 39(10), 1999 (b): 2042-2063.
- Yang, Y and Gao, F. "Strategies for Injection Filling Velocity Control." *J. Reinforced Plastics and Composites*, 20(13), 200:1160-1170.

- Yang, Yi and Gao, F. "Injection Molding Product Weight: Online Prediction and Control Based on a Nonlinear Principal Component Regression Model." *Polym. Eng. Sci.*46 (4), 2006: 540-548.
- Yao, K. and Gao, F. "Adaptive GPC Control of barrel Temperatures in Injection Molding." *SPE ANTEC Technical Papers* 2005:813-817.
- Zhao, C. and Gao, F. "Melt Temperature Profile Prediction for Thermoplastic Injection Molding." *Polym. Eng. Sci.*39 (9), 1999: 1787-1801.
- Zheng, S. R. and Chen M. "Development of Cognitive Function of Cerebral Neural Network: An Optimization System for Injection Molding Parameters Based on Hybrid Neural network and Genetic Algorithm." *Chinese Journal of Clinical Rehabilitation* 9 (1), 2005: 209-213.

Appendix A

Cavity Pressure and Temperature

Sensors

Common Melt Temperature Sensors (Kazmer et al., 2005a)

Brand	Model#	Head Diameter (mm)	Operating Temperature (° C)	Thermocouple Type
Primus	Type4001A/4002A	1.0	0...200	N
Primus	Type4003A/4004A	1.0	0...200	N
Primus	Type4005A/4006A (Machinable)	1.0	0...200	N
Kistler	Type 6193A03	1.0	0...200	K
Kistler	Type 6194A04	2.5	0...200	K
Kistler	Type 6195A04	2.5	0...200	K

Common Cavity Pressure Sensors (Kazmer et al., 2005a)

Brand	Model#	Head Diameter (mm)	Max Sensor Temperature (°C)	Sensitivity	Sensing Means	Mounting Style
Kistler	Type6152A	6.0	200	-9.4 pC/Bar	Piezoelectric	Direct
Kistler	Type6157BB	4.0	300	-9.4 pC/Bar	Piezoelectric	Direct
Kistler	Type6158A/6159A	2.5	200	-2.5 pC/Bar	Piezoelectric	Direct
Kistler	Type6183AE	1.0	200	-2.5 pC/Bar	Piezoelectric	Direct
RJG	LS-F-40-20K	4.0	218.3	+/-1 %	Strain Gauge	Direct
RJG	LS-F-25-20K	2.5	218.3	+/-1 %	Strain Gauge	Direct
RJG	LP-F-40-30K	4.0	200	+/- 0.5 %	Piezoelectric	Direct
RJG	LP-F-25-30K	2.5	200	+/- 0.5 %	Piezoelectric	Direct
RJG	T-6157	4.0	218.3	1.7 mV/V	Strain Gauge	Direct
RJG	T-6159	2.5	218.3	0.5 mV/V	Strain Gauge	Direct
RJG	6157	4.0	200	9.4 pC/Bar	Piezoelectric	Direct
RJG	6159	2.5	200	2.0 pC/Bar	Piezoelectric	Direct
Priamus	Type 6001A	4.0	200	ca -10 pC/Bar	Piezoelectric	Direct
Priamus	Type 6001B	4.0	200	ca -10 pC/Bar	Piezoelectric	Direct
Priamus	Type 6002B	4.0	200	-5 pC/Bar	Piezoelectric	Direct
Priamus	Type 6003A	2.5	200	~ -5 pC/Bar	Piezoelectric	Direct
Priamus	Type 6003B	2.5	200	~ -5 pC/Bar	Piezoelectric	Direct
Priamus	Type 6004A	6.0	200	ca -10 pC/Bar	Piezoelectric	Direct
Priamus	Type 6004B	6.0	200	ca -10 pC/Bar	Piezoelectric	Direct
Priamus	Type 6005A	1.5	200	-1.5 pC/Bar	Piezoelectric	Direct
RJG	9204	12, 6.0	200	2.2 pC/N	Piezoelectric	Indirect
RJG	9211	6.0	200	4.4 pC/N	Piezoelectric	Indirect
RJG	T-412,413, 414	12, 7.0	121.1/Hi:218.3	2.0 mV/V	Strain Gauge	Indirect
RJG	LS-B-127	12,7.0	121.1/Hi:218.3	+/- 0.5%	Strain Gauge	Indirect
RJG	LP-B-127-2000	12,6.0	200	+/- 0.5%	Piezoelectric	Indirect
RJG	LP-B-60-562	6.0	200	+/- 0.75%	Piezoelectric	Indirect
Kistler	Type9211A	6.0	-40~150	-4.4 pC/Bar	Piezoelectric	Indirect
Kistler	Type9204B	12,6.0	-50~200	-1.6 pC/Bar	Piezoelectric	Indirect
Kistler	Type6189	2.5	200	-6.5 pC/Bar	Pressure/Temp Combination	Direct
Kistler	Type6190A	4.0	200	-2.5 pC/Bar	Pressure/Temp Combination	Direct

Appendix B

Calculation of Convection Coefficient

Taken from Menges et al. (2001)

The convective heat transfer coefficient can be calculated as:

$$\alpha = \left(0.037 \left(\frac{\dot{V}_c \cdot 4 \cdot 1000}{d_c \cdot \pi \cdot v_c \cdot 60} \right)^{0.75} - 180 \right) \cdot \text{Pr}^{0.42} \cdot \frac{k_c}{a_c}$$

v_c Kinematics viscosity of coolant [cm²/s]

d_c Cooling channel diameter [cm].

Pr Prandtl number

k_c Thermal Conductivity of the coolant

a_c Diffusivity of the coolant

Diffusivity:

$$a = \frac{k}{\rho \cdot Cp}$$

k Thermal Conductivity

ρ Density

Cp Specific heat capacity

Appendix C:

Code for SISO MPC Controller

```
num1=[0.02209 -0.002708];
den1=[1 -1.607 0.6267];
delt1=5;
delay1=4;
g1=poly2tfd(num1,den1,delt1,delay1);
tfinal= 150;
delt2=5;
nout1=1;
model=tfd2step(tfinal,delt2,nout1,g1);

%plant model, no plant/model mismatch.
num1=[0.02209 -0.002708];
den1=[1 -1.607 0.6267];
delay1=4;
g2=poly2tfd(num1,den1,delt1,delay1);
tfinal= 150;
delt2=5;
nout1=1;
plant=tfd2step(tfinal,delt2,nout1,g2);

%calculate the MPC controller gain matrix for no plant/model mismatch,
%output Weight=1, Input Weight=0
%Input Horizon=2, output Horizon=32
ywt=1; uwt=0.85;
M=2; P=32;
Kmpc=mpccon(model,ywt,uwt,M,P);
usat=[20 80 40]; tfilter=[]; dmodel=[];
dstep=[];

% Simulate and plot response
tend=160;
[yp,u,ym]=mpcsim(plant,model,Kmpc,tend,r,usat);
plotall(yp,u,delt2)
```

Appendix D

Closed-loop Response Code for MPC Controller

A portion of the program was taken from Matlab Model-based Predictive Control Toolbox functional code. The program is running to calculate and provide the manipulate variable move based on an input value from plant output (here it is the simulation outputs from Moldflow MPI).

```
% plant(model): the step response coefficient matrix of the plant (model)
%     generated by the function tfd2step.
% Controller: the constant control law matrix computed by the function mpcecon
%     (closed-loop simulations).For open-loop simulation, controller=[].
% tend: final time of simulation.
% Setpoint: for the closed-loop simulation, it is a constant or time-varying
%     reference trajectory. For the open-loop simulation, it is the
%     trajectory of the manipulated variable u.
% OPTIONAL INPUTS:
% uconstaint: the matrix of manipulated variable constraints.It is a constant
%     or time-varying trajectory of the lower limits (Ulow), upper limits
%     (Uhigh) and rate of change limits (DelU) on the manipulated
%     variables. Default=[].
% tfilter: time constants for noise filter and unmeasured disturbance lags.
%     Default is no filtering and step disturbance.
% dplant: step response coefficient matrix for the disturbance effect on the
%     plant output generated by the function tfd2step. If distplant is
```

```

% provided, dstep is also required. Default = [].
% dmodel: step response coefficient matrix for the measured disturbance
% effect on the model output generated by the function tfd2step.
% If distmodel is provided, dstep is also required. Default=[].
% dstep: matrix of disturbances to the plant. For output step disturbances
% it is a constant or time-varying trajectory of disturbance values
% For disturbances through step response models,it is a constant or
% time-varying trajectory of disturbance model inputs.Default=[].
% OUTPUT ARGUMENTS: y (system response), u (manipulated variable) and
% ym (model response)

% model construction

% plant inaccurate transfer function:  $g=0.2\exp(-0.12)/s$ 
% build a step response models for a sampling period of 0.02.
num1=[0.02209 -0.002708];
den1=[1 -1.607 0.6267];
delt1=5;
delay1=3;
g1=poly2tfd(num1,den1,delt1,delay1);
tfinal= 150;
delt2=5;
nout1=1;
model=tfd2step(tfinal,delt2,nout1,g1);

%plant model
num1=[0.02209 -0.002708];
den1=[1 -1.607 0.6267];
delay1=3;
g2=poly2tfd(num1,den1,delt1,delay1);
tfinal= 150;
delt2=5;
nout1=1;
plant=tfd2step(tfinal,delt2,nout1,g2);
%calculate the MPC controller gain matrix for no plant/model mismatch,
%output Weight=1, Input Weight=0
%Input Horizon=5, output Horizon=30

ywt=1; uwt=0.85;

```

```

M=2; P=40;
controller=mpccon(model,ywt,uwt,M,P);
% Simulate and plot response for unmeasured step disturbance through plant with
different input weight.
uconstraint=[30 80 40]; tfilter=[];dmodel=[];
dstep=[];
tend=150;

%Simulation starts

%Basic settings

[npny,nu] = size(plant);
[nmny,num] = size(model);

ny = plant(npny-1,1);    ny1 = model(nmny-1,1);
delt = plant(npny,1);    deltm = model(nmny,1);

[ls,ws] = size(setpoint);
ntend = fix(tend/delt);

npny = npny-2-ny;        nmny = nmny-2-ny;
np = npny/ny;           nm = nmny/ny;
noutp = plant(npny+1:npny+ny,1); noutm = model(nmny+1:nmny+ny,1);
intp = ones(ny,1)-noutp;    intm = ones(ny,1)-noutm;

noutdp = ones(ny,1);    noutdm = ones(ny,1);

if (any(intp)==1 | any(intm)==1)
    casenum = 2;
end;

% check the setpoint argument
% for closed-loop simulation
[nuc,pony] = size(controller);
p = (pony/ny);
if (nuc~=nu)
    error('Check the dimension of CONTROLLER.');
```

```

        setpoint(is,:) = setpoint(ls,:);
    end;
end;

% if [p(horizon)+1] > nm (the truncation order of the model),
% make nm = p+1, and add the last step response coefficients
% (p+1-nm) times to the MODEL

if ((p+1)>nm)
    k = nmny; kx = nmny-ny+1;
    int = intm*ones(1,nu);
    for ib = nm+1:p+1
        model(k+1:k+ny,:) = model(kx:k,:)+int.*(model(kx:k,...
            -model(kx-ny:kx-1,:));
        k = k+ny; kx = kx+ny;
    end;
    nmny = k; nm = p+1;
end;

% Loop controller status

% interpret uconstraints

if (isempty(uconstraint)==0)
    [nuconstraint,ncol]=size(uconstraint);
    for ir = 1:nuconstraint
        for iu = 1:nu
            if (uconstraint(ir,iu)==-inf)
                uconstraint(ir,iu) = -1.0E10;
            end;
        end;
        for iu = nu+1:3*nu
            if (uconstraint(ir,iu)==inf)
                uconstraint(ir,iu) = 1.0E10;
            end;
        end;
    end;
end;
end;

% interpret filter and disturbance time constants

[nrtfil,nctfil]=size(tfilter);

if (isempty(tfilter))

```

```

    tdist=[];
end;
fa=ones(1,ny);
KF = ones(nmny,ny);

for i=1:ny

if (isempty(tdist))
    if (noutm(i)==0|noutdm(i)==0)
        alpha(i)=1;
    else
        alpha(i)=0;
    end;
else
    if tdist(i)==0
        alpha(i)=0;
    elseif tdist(i)==inf
        alpha(i)=1;
    elseif ((noutm(i)==0|noutdm(i)==0) & tdist~=inf)
        disp('WARNING: tdist(i) must equal inf');
        disp('    for integrating outputs. ');
        disp('    tdist(i) will be set to inf. ');
        alpha(i)=1;
    else
        alpha(i)= exp(-delt/tdist(i));
        if alpha(i) < 0 | alpha(i) > 1
            error('Disturbance time constants must be positive');
        end;
    end;
end;
    fb(i) = alpha(i)*fa(i)^2/(1+alpha(i)-alpha(i)*fa(i));
end;
KFA = zeros(nmny,ny);
Ad=diag(alpha);
for i=2:nm
    KFA((i-1)*ny+1:i*ny,1:ny)=KFA((i-2)*ny+1:(i-1)*ny,1:ny) + Ad^(i-2);
end;

KFA(nmny+1:nmny+ny,1:ny)=Ad^(nm-1);
KF(nmny+1:nmny+ny,1:ny)=zeros(ny,ny);

KFB=KFA*diag(fb);
KF = KF + KFB;

```



```

hp = noutp.*noutdp-ones(ny,1);
HP = diag(hp);
cm = noutm.*noutdm;
CM = diag(cm);
hm = noutm.*noutdm-ones(ny,1);
HM = diag(hm);
gp = 2*ones(ny,1)-noutp.*noutdp;
GP = diag(gp);
gm = 2*ones(ny,1)-noutm.*noutdm;
GM = diag(gm);
t1 = clock;

% Integrating Closed-loop simulation

% Build a utility matrix used in the closed-loop simulation

E = zeros(pny,1);
    YP = zeros(npny,1);
    YM = zeros(nmny,1);
Xdist=zeros(ny,1);

fprintf(' Time remaining %g/%g\n',tend,tend);

setpointblock = setpoint(1:p,:);
E = setpointblock(:) - E;
Du = controller*E;
u(1,:) = Du';
Du = u(1,:);

if (isempty(uconstraint)==0)
    Dusign = sign(Du);
    Du = min(abs(Du),uconstraint(1,2*nu+1:3*nu));
    Du = Dusign.*Du;
    u(1,:) = Du';
    u(1,:) = max(u(1,:),uconstraint(1,1:nu));
    u(1,:) = min(u(1,:),uconstraint(1,nu+1:2*nu));
    Du = u(1,:);
end;

YPA = plant(1:npny,1:nu)*Du;
YMA = model(1:nmny,1:nu)*Du;

for i = 1:(npny-ny)
    YPA(i) = YP(i+ny) + YPA(i);

```

```

end;
YPA(npny-ny+1:npny) = HP*YP(npny-2*ny+1:npny-ny)+GP*YP(npny-ny+1:npny)...
    +YPA(npny-ny+1:npny);

for i = 1:(nmny-ny)
    YMA(i) = YM(i+ny)+YMA(i);
end;
YMA(nmny-ny+1:nmny) = HM*YM(nmny-2*ny+1:nmny-ny)+GM*YM(nmny-
ny+1:nmny)...
    +YMA(nmny-ny+1:nmny)+CM*Xdist;
XdistA=Ad*Xdist;
u_last=u(1:nu);
fprintf(1,'Manipulate input: u1 = %6.3f\n',u_last);
YPA(1:ny)=input('plant output is :');
DY = YPA(1:ny)-YMA(1:ny);

YMA = YMA + KF(1:nmny,1:ny)*DY;
XdistA = XdistA + KF(nmny+1:nmny+ny,1:ny)*DY;

YP = YPA; YM = YMA;
Xdist = XdistA;

y(2,:) = YP(1:ny)';
ym(2,:) = YM(1:ny)';

for k=2:ntend+1          % Loop k closed-loop

    if ((fix(k/10)-k/10)==0)
        fprintf(' Time remaining %g/%g\n',tend-k*delt,tend);
    end;

    E = YM(ny+1:ny+pny);

    setpointblock(:,1:p) = [setpointblock(:,2:p), ...
        setpoint(min(1s,p+k-1),:)]';
    E = setpointblock(:) - E;
    Du = controller*E;
    u(k,:) = u(k-1,:)+Du';

    Du = u(k,:)' - u(k-1,:);

    if (isempty(uconstraint)==0)
        rusat = min(k,nuconstraint);
        Dusign = sign(Du);

```

```

    Du = min(abs(Du),uconstraint(rusat,2*nu+1:3*nu)');
    Du = Dusign.*Du;
    u(k,:) = u(k-1,:)+Du';
    u(k,:) = max(u(k,:),uconstraint(rusat,1:nu));
    u(k,:) = min(u(k,:),uconstraint(rusat,nu+1:2*nu));
    Du = (u(k,:)-u(k-1,:))';
end;
YPA = plant(1:npny,1:nu)*Du;
YMA = model(1:nmny,1:nu)*Du;
for i = 1:(npny-ny)
    YPA(i) = YP(i+ny) + YPA(i);
end;
YPA(npny-ny+1:npny) = HP*YP(npny-2*ny+1:npny-ny)+GP*YP(npny-
ny+1:npny)...
    +YPA(npny-ny+1:npny);

for i = 1:(nmny-ny)
    YMA(i) = YM(i+ny)+YMA(i);
end;
YMA(nmny-ny+1:nmny) = HM*YM(nmny-2*ny+1:nmny-ny)+GM*YM(nmny-
ny+1:nmny)...
    +YMA(nmny-ny+1:nmny)+CM*Xdist;

XdistA=Ad*Xdist;
u_last=u(k,1:nu);
fprintf(1,'Manipulate input: u = %6.3f\n',u_last);
YPA(1:ny)=input('plant output is :');
DY = YPA(1:ny)-YMA(1:ny);
YMA = YMA + KF(1:nmny,1:ny)*DY;
XdistA = XdistA + KF(nmny+1:nmny+ny,1:ny)*DY;
y(k+1,1:ny) = YPA(1:ny)';
ym(k+1,1:ny) = YMA(1:ny)';
YP = YPA; YM = YMA;
Xdist = XdistA;
end;          % Loop k closed-loop
y = y(1:ntend+1,:);
ym= ym(1:ntend+1,:);
u = u(1:ntend+1,:);
end;          % Loop simulation - integrating
end;
if tend ~= 0,
    fprintf('Simulation time is %g seconds.\n', etime(clock,t1));
end

```

Appendix E

MIMO MPC Controller Code

```
%y1-u1
```

```
num11=[0.002826 0.002826 0.002826];  
den11=[1 -2.044 1.224 -0.1722];  
delt1=0.02;  
delay11=0;  
g11=poly2tfd(num11,den11,delt1,delay11);
```

```
%u1-u2
```

```
%num12=[-1.068 1.022 0.006085];  
%den12=[1 2.518 1.808 -0.3381];  
num12=[0];  
den12=[0 1];  
delay12=0;  
g12=poly2tfd(num12,den12,delt1,delay12);
```

```
% y2-u1
```

```
%num21=[0.005902 -0.007659 0.01015];  
%den21=[1 -0.8102 -0.3305 0.1483];  
num21=[0];  
den21=[0 1];  
delay21=0;  
g21=poly2tfd(num21,den21,delt1,delay21);
```

```
% y2-u2
```

```
num22=[0.01822 0.01822 0.01822];  
den22=[1 -1.068 0.1287 0.002569];  
delay22=0;  
g22=poly2tfd(num22,den22,delt1,delay22);  
%plant model  
tfinal=0.7;  
delt2=0.02;
```



```

nout=2;
plant=tf2step(tfinal,delt2,nout,g11,g12,g21,g22);
%calculate the MPC controller gain matrix for no plant/model mismatch,
%output Weight=1, Input Weight=0
%Input Horizon=5, output Horizon=30
model=plant;
ywt=[1 1]; uwt=[2 5];
M=4; P=40;
Kmpc=mpccon(model,ywt,uwt,M,P);
% Simulate and plot response for set point control through plant with different input
weight and no disturbance.
usat=[-20 -20 20 20 20 20];

tfilter=[];dmodel=[];
dstep=[]; dplant=[];
tend=2;
r=[18 -12];
[yp,u,ym]=mpcsim(plant,model,Kmpc,tend,r,usat,tfilter,dmodel,dplant,dstep);
yp(:,1)=yp(:,1)+sspre(:,1);
yp(:,2)=yp(:,2)+tem(:,1);
u(:,1)=u(:,1)+60;
u(:,2)=u(:,2)+85;
YT=[yp,r(:,1)+sspre(:,1),r(:,2)+tem(:,1)];
plotall(YT,u,delt2)

```