

Thermal Management in Laminated Die System

Jaho Seo*, Amir Khajepour, and Jan P. Huissoon

Abstract: The thermal control of a die is crucial for the development of high efficiency injection moulds. For an effective thermal management, this research provides a strategy to identify a thermal dynamic model and to design a controller. The neural network techniques and finite element analysis enable modeling to deal with various cycle-times for moulding process and uncertain dynamics of a die. Based on the system identification which is experimentally validated using a real system, controllers are designed using fuzzy-logic and self-tuning PID methods with backpropagation and radial basis function neural networks to tune control parameters. Through a comparative study, each controller's performance is verified in terms of response time and tracking accuracy under different moulding processes with multiple cycle-times.

Keywords: Fuzzy logic, laminated die, plastic injection moulding, self-tuning PID control, system identification, various cycle-times.

1. INTRODUCTION

Die casting is a primary manufacturing process to produce metal parts by injecting molten metal under high pressure into dies. Since this process enables parts with complex shapes requiring high precision and good repeatability, it is widely used in the metalworking industry as a fabrication method. Thermal control is a key issue in this process since uniform temperature in the dies contributes to production quality by reducing problems such as shrink porosity, poor fill [1] and prolonged cycle-times for part solidification [2].

Many approaches have been proposed to deal with the thermal control in die systems. A PI [3] and PID algorithms [1] were applied to manage the cavity temperature on a plastic injection moulding and high-pressure die-casting, respectively. To improve limitation of a PID control in presence of uncertain or nonlinear dynamics, several studies have been undertaken. A Dahlin controller [4] for mold temperature control was evaluated in an injection molding process with uncertainties. Using a simplified mathematical process model and cost function to determine control commands, model predictive control (MPC) has been utilized in diverse range of die systems for thermal control [5–8]. Despite of improved performance compared to the PID

control, the suggested controllers are not robust in some circumstances. Specifically, because these controllers are based on a linear “best-fit” approximation (e.g., ARX and ARMAX), the performance of the controllers is affected largely by modeling errors arisen from uncertain dynamics.

Although accurate modeling of the thermal dynamics of dies is a prerequisite to a successful thermal control, it is a difficult task in practice due to die's several characteristics. For example, a die is a complex continuous system with cooling and heating channels causing the modeling and control to be quite complicated. Unmodeled thermal dynamics of dies (such as convection and radiation) also provide further analytical model challenges.

To deal with the limitations of aforementioned simplified models and inherent challenges of modeling in a die system, this paper considers a neural network (NN) approach. By applying NN techniques, the thermal dynamics with uncertainties in a laminated die is modeled. In addition, our modeling covers various cycle-times for plastic moulding process that has not been studied (i.e., most documented approaches for thermal management of die systems have only considered a fixed cycle-time). In this study, system identification methods are conducted using the temperature distribution obtained through a finite element analysis (FEA). After verifying the identified model through experimental validation, controllers are designed using the fuzzy-logic and NN based self-tuning PID control methods.

Section 2 describes a laminated die system. Section 3 presents a methodology for modeling thermal dynamics using FE simulation and NN techniques, and validating the derived model with experimental data from a real die. In Section 4, controller design approaches and performance of designed controllers are discussed. Finally, Section 5 provides concluding remarks.

This work has been supported by the Natural Sciences and Engineering Research Council of Canada and Ontario Research Fund.

Jaho Seo is with the Department of System Reliability, Korea Institute of Machinery & Materials, 156 Gajungbukno, Yuseong-gu, Daejeon, 305-343, Korea (e-mail: seojaho@kimm.re.kr).

Amir Khajepour and Jan P. Huissoon are with the Department of Mechanical and Mechatronics Engineering, University of Waterloo, 200 University Avenue West, Waterloo, Ontario, N2L 3G1, Canada (e-mails: {akhajepour, jph}@uwaterloo.ca).

* Corresponding author.

2. SYSTEM DESCRIPTION

2.1 Laminated die system with conformal cooling channels

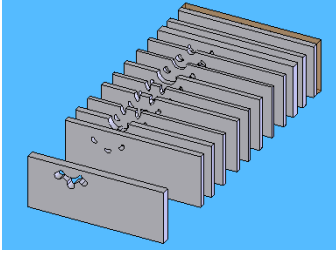


Fig. 1. Laminated tooling.

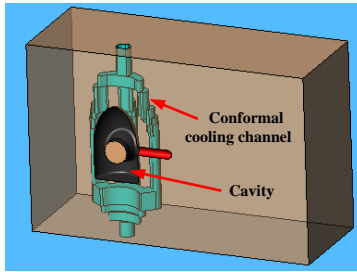


Fig. 2. Laminated die with conformal cooling channel.

Laminated tooling (Fig. 1) is a manufacturing technique which fabricates a die by joining sliced sheets of a suitable material rather than the traditional approaches using machining or forming [9]. Since the optimization of slicing process is available in laminated tooling, the direction and thickness of each slice in the die can be optimized to have the least volume deviation between desired die design and assembled slices [10]. This feature of laminated tooling enables to achieve conformal cooling channels encompassing the shape of the cavity. The conformal cooling channels provide more efficient cooling through fast and uniform heat removal from the die [11]

Figure 2 shows a laminated die (plastic injection mould) with conformal cooling channels which is used to analyze the thermal dynamics in this study. Hot polymer injected into the mould cavity is cooled to the demoulding temperature by heat transfer to coolant (water) through the conformal cooling channel in close proximity to the cavity. Since the dominant heat transfer mechanism for the injection moulding process is the convection by the coolant [3, 12], the flow rate and temperature of the coolant are chosen as the controllable process parameter to control the temperature distribution of the die. FE simulations for thermal dynamic analysis of the die are carried out based on this model.

2.2 Various cycle times

The cycle-time for a plastic injection moulding process is allocated for main phases of injection, packing, holding, cooling and ejection.

Table 1. Material properties of Santoprene 8211-45 (polymer) and ASTM-A36 (mould).

Material	Density (kg/m^3)	Thermal conductivity ($W/m^{\circ}C$)	Specific heat capacity ($J/kg^{\circ}C$)
Santoprene 8211-45	790	0.1	2380
ASTM-A36	7830	54	490

Since the cooling phase among these phases needs a large portion of the cycle-time which is used to cool the polymer down to solidification temperature [13], the cooling time plays a key role in the cycle. The previous studies dealing with the system identification and thermal control in the injection moulding process have used a predetermined cooling time (thus cycle-time). However, the system identification using a fixed cycle-time cannot cope with the wide range of thermal dynamics variation according to the cycle-times. The control strategy based on this limited identification cannot effectively control the die temperature. In this study, various cycle-times of the moulding process are considered in the thermal dynamics modeling.

3. MODELING METHODOLOGY

3.1. Finite element analysis

The objective of the FEA is to obtain the input-output data set to train the NN for thermal dynamic modeling of injection moulding process. The injection moulding process of the die in Fig. 2 can be briefly described as follows:

Polymer is injected into the cavity at a temperature of $165^{\circ}C$ before the injection moulding cycle starts. During the cycle, the heat removal from the melt and mould is achieved through heat convection with the coolant and air. After the cooling phase, the polymer part is ejected from the cavity of opened mould. For the FEA, ANSYS CFX 11 software package was used with 3D CAD files from SolidWorks.

The material properties of the polymer and mould are provided in Table 1. The properties of water were used for the coolant. Flow of the coolant in the cooling channel was assumed to be turbulent. The initial temperature of the mould was assumed as the room temperature ($25^{\circ}C$). A transient state analysis in the FEA was carried out to observe the temperature distribution over the time. For the FEA meshing, the convergence study for mesh refinement was conducted until a convergence in the temperature distribution was achieved. As a result, mesh models were generated with a total of 1129514 elements and 217719 nodes for the mould with conformal channel using the tetrahedral mesh generation (default mesh setting for CFX).

3.2. Selection of sensor locations

Four sensor locations at which temperature distribution was monitored during the moulding process are presented in Fig. 3. Numbering shown in the figure

is for assigning target nodes from node 1 (i.e., Mo1) to node 1152 (i.e., Mo1152) using total 1152 target nodes. From the study by Seo et al. [14], the sensor locations were identified by using a constant ratio based clustering method and sensitivity analysis.

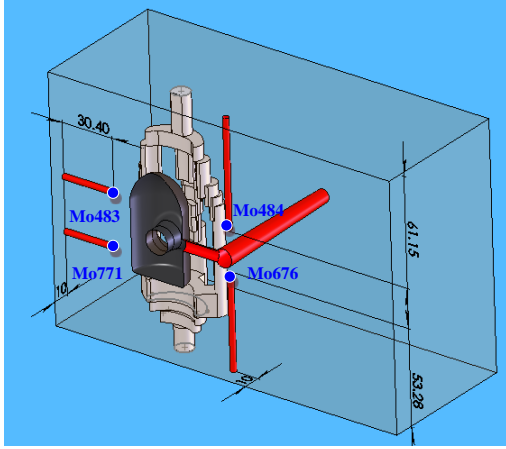


Fig. 3. Four optimal sensor locations after clustering and sensitivity analysis.

Clustering on target sensor locations enables to create a regular pattern through which grouping target locations according to similarity becomes available. Thus, the sensor locations belonging to different groups can provide the unique information representing each group and also the location information (e.g., dense region of the sensor locations) can be obtained. Through sensitivity analysis, it is available to find the most sensitive node to input changes as a representative sensor location in each cluster. The temperature distributions at four sensor locations in Fig. 3 were used for identifying a thermal model using the FEA. For an experimental validation of the identified thermal model, thermocouples were installed at the same locations.

3.3. Modeling of thermal dynamics

As a solution to model die thermal dynamics considering uncertainties, a NN approach is presented in the study. System identification using a NN and FEA data is carried out and the proposed model is experimentally validated.

- Introduction of NARX

To capture a dynamical model to describe the temperature distribution in the die, NARX (nonlinear autoregressive with exogenous inputs) model is provided. NARX is a powerful class of nonlinear dynamical model in which embedded memory is included as temporal changes to reflect the dynamic response of the system. Due to this feature, the inputs of the NARX are composed of past values of the system inputs and outputs as follows:

$$y(k) = y_N(k) + e(k) \\ = f(y(k-1), y(k-2), \dots, y(k-n_y), u(k), u(k-1), u(k-2), \dots, u(k-n_u)) + e(k), \quad (1)$$

where $y(k)$, $y_N(k)$, $u(k)$ and $e(k)$ stand for the system output, neural network output, system input and error between $y(k)$ and $y_N(k)$ at time k , respectively. n_y and n_u are the number of past outputs and past inputs, respectively and are referred to the model orders.

- System identification using NARX model and FEA

When applying the NARX model to our modeling, the outputs in the NARX model are the temperatures at 4 nodes in Fig. 3. However, the rapid change of the temperature during the cycle makes it difficult to use this temperature profile as a set-point and thus alternative variable is required as an output. Instead, cycle average temperature [3, 4, 15] defined in Eq. (2) serves as an output.

$$T_a = \frac{\int_0^P T dt}{P} = \frac{\sum_0^M T_n}{M}, \quad (2)$$

where t is time, T_a is the cycle average temperature, T is the instantaneous temperature at a specific location, P is the cycle-time, M is the total number of samples during one cycle, and T_n is the temperature at the n^{th} sample time. It is convenient to use T_a for relatively short injection cycles and its use has an advantage of its insensitiveness to process noises.

The past states of temperatures at each node (i.e., past outputs) and the flow rate (input) of the coolant are used as inputs to the NARX model. A real plastic moulding process uses the coolant with constant temperature (15.5 °C) and thus, the coolant temperature was utilized as a boundary condition instead of an input in the NARX model. As mentioned before, our modeling is extended to deal with various cycle-times rather than a predetermined cycle-time that previous studies have considered. Therefore, the cycle-time for the moulding process is additionally included as the input variable in the model. The NARX model to cover all above inputs (flow rate, cycle-time, past values of outputs at 4 nodes) and outputs (T_a at 4 nodes) is a multi-input multi-output (MIMO) system.

Data for training and testing the NARX model was obtained from the FEA. The following table shows inputs conditions for flow rate and cycle-time to generate the steady state outputs used for training and testing of the NARX modeling. A part of the training data sets is demonstrated in Fig. 4, which is generated for node Mo484 using all flow rates in Table 2 and one cycle-time (91sec) among 4 types of cycle-times.

Table 2. Input conditions of FEA for NARX modeling.

	Input variables	Range
Training	Flow rate (gpm)	0, 1, 3, 4, 5, 6, 8
	Cycle-time (sec)	61, 71, 81, 91
Testing	Flow rate (gpm)	0, 2, 7.13
	Cycle-time (sec)	66, 86

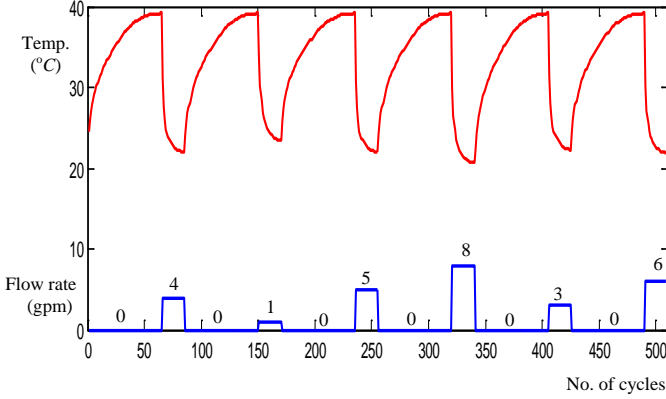
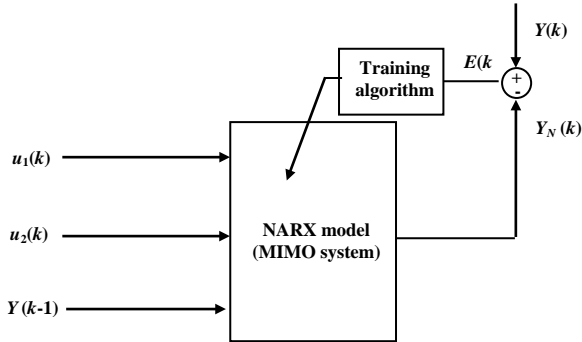


Fig. 4. Training data using all flow rate ranges (0, 1, 3, 4, 5, 6, 8gpm) and one cycle-time (91sec) at node Mo484.



- Temperature at 4 nodes which are Mo483, Mo484, Mo676, Mo771:
 $Y(k)=[y_{Mo483}(k); y_{Mo484}(k); y_{Mo676}(k); y_{Mo771}(k)]$
- Neural network output: $Y_N(k)$
- System inputs: $u_1(k)$: cycle-time., $u_2(k)$: water flow rate,
 $Y(k-1)$: past values of $Y(k)$ with one cycle-time delay.

Fig. 5. Schematic of NARX model.

From the figure, it can be seen that for data gathering, each flow rate (input) with a given cycle-time keeps constant until a corresponding steady state of temperature at the node (output) is reached over many cycles [3]. Additional data sets for modeling at the same node (Mo 484) were generated by using the same flow rate ranges (0, 1, 3, 4, 5, 6, 8gpm) with each different cycle-time (i.e., 61, 71, 81sec).

This method was also applied for data generation at

the remaining three nodes (Mo483, Mo676, Mo771). Next step is to determine the model order in the NARX model of Eq. (1) by applying the Lipschitz criterion [16] and using the training data sets. The obtained model orders are presented in the schematic of NARX model (see Fig. 5).

For better results and reduction of the calculation time [17], data was normalized by Eq. (3) and Eq. (4) before they were used for training and testing process.

$$y_{Nor,i}(k) = \frac{y_i^{\max} - y_i(k)}{y_i^{\max} - y_i^{\min}}, \quad (3)$$

$$u_{Nor,m}(k) = \frac{u_m(k)}{u_m^{\max}}, \quad (4)$$

where $y_{Nor,i}$, y_i^{\max} , y_i^{\min} are the normalized value, maximum, minimum of the output y (temperature) at i^{th} node, respectively. $u_{Nor,m}$ and u_m^{\max} are the normalized value and maximum of the m^{th} input u_m (cycle-time or flow rate), respectively.

The Levenberg-Marquardt (LM) algorithm was adopted as a training algorithm. 20% of data sets randomly picked from the training data sets was used as a validation set to determine the number of hidden layer nodes. Specifically, after training different neural networks with different hidden nodes between 4 and 35, the neural network having the least generalization error (i.e., smallest mean square error) computed using a validation set was selected for the testing. By this process, the optimal number of hidden layer nodes (6 nodes) was determined.

An accuracy of the developed NARX model is compared with that of ARX (auto-regressive exogenous) and ARMAX (autoregressive moving average exogenous) models which are linearly approximated models using a least-squares formulation.

For this comparison, training data in Fig. 4 for node Mo484 was used. The structures (orders) of estimated ARX and ARMAX models are presented in Eq. (5) and Eq. (6), respectively.

$$\text{ARX model: } A(q)y_m(t) = B(q)u_m(t) + e_m(t), \quad (5)$$

where

$$A(q) = 1 - 1.389q^{-1} + 0.4094q^{-2}, B(q) = 0.4524 - 0.4096q^{-1}$$

$$y_m(t) = y_{Nor,i}(k), u_m(t) = u_{Nor,m}(k), e_m(t): \text{estimation error.}$$

$$\text{ARMAX model: } A(q)y_m(t) = B(q)u_m(t) + C(q)e_m(t), \quad (6)$$

where

$$A(q) = 1 - 1.39q^{-1} + 0.4096q^{-2}, B(q) = 0.4518 - 0.4089q^{-1}$$

$$C(q) = 1 - 0.01944q^{-1} + 0.03512q^{-2}$$

$$y_m(t) = y_{Nor,i}(k), u_m(t) = u_{Nor,m}(k), e_m(t): \text{estimation error.}$$

The ARX and ARMAX models having the above structures (i.e., 2nd order of $y_m(t)$ and $u_m(t)$) show the

smallest estimation error for the training data in Fig 4.

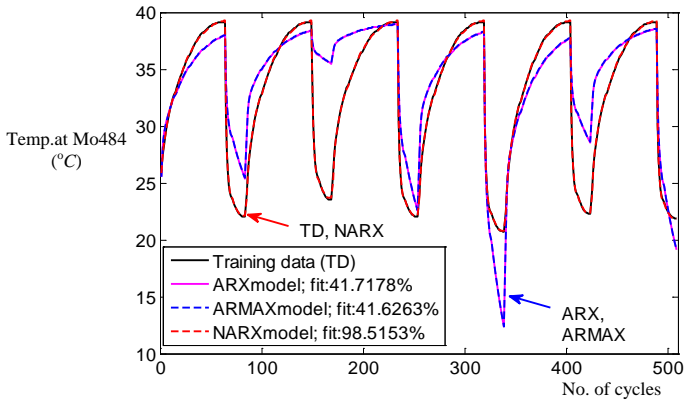


Fig. 6. Training results with ARX, ARMAX and NARXmodels at node Mo484 using training data in Fig. 4.

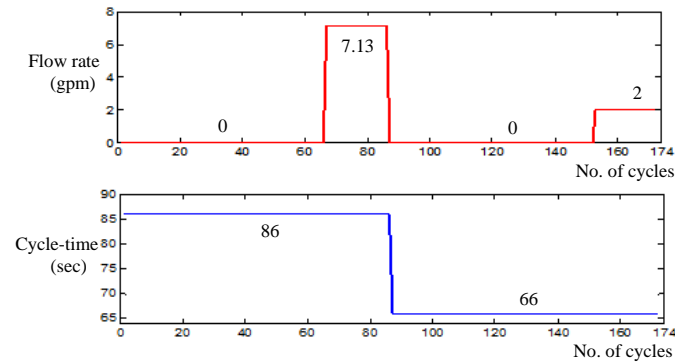


Fig. 7. Input conditions for testing data.

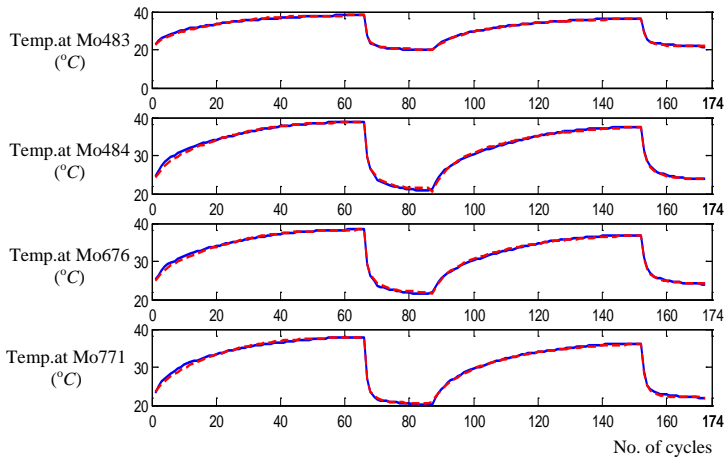
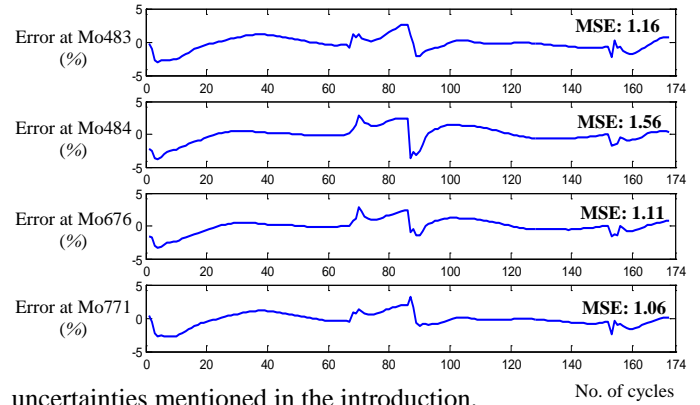


Fig. 8. Testing results (comparison between model (Y_N) and actual outputs (Y)).

The comparative results of modeling accuracy are shown in Fig. 6. In the figure, while the NARX model shows high accuracy (a goodness of fit: 98.5%), the ARX and ARMAX models have a lower level of accuracy (a goodness of fit: 41.7% for ARX, 41.6% for ARMAX). Therefore, conventional methods based on a

linear “best-fit” approximation may not be appropriate in modeling the thermal dynamics of dies with complex



uncertainties mentioned in the introduction.

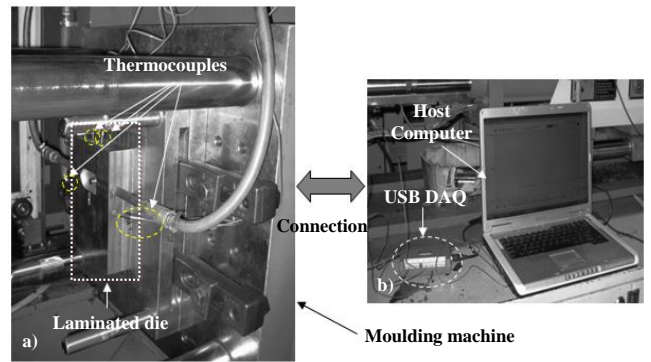


Fig. 9. Testing results (error between Y_N and Y).

Fig. 10. Layout of experimental set up.

Then, the performance (i.e., accuracy) of the NARX model trained using the full training data (2040 ea. in Table 2) with 200 epochs was tested with the testing data sets (174 ea.). The input conditions (from Table 2) and the simulation results for the testing are shown in Fig. 7 and Fig. 8–9, respectively.

By referring to the MSE values in Fig. 9, it can be noted that our NARX model has a good performance of temperature estimation at all nodes for new input conditions that are not utilized for training data.

- Experimental set up for validation of NARX model

Experiments were carried out to validate the dynamic model identified from the FEA. For an experimental validation, a set up was designed as follows:

As represented in Fig. 10-a), the laminated die was installed on the moulding machine at the WebPlas Inc [18]. Four thermocouples (type K) inserted in the die were used to measure the temperature at 4 nodes. The thermocouples were connected to a USB data acquisition module (DAQ) (see Fig. 10-b)). The temperature distribution acquired by DAQ was sent to the host pc where data was monitored and saved.

The temperature of the die was cooled down through the coolant circulating by a chiller with operating pressure of 586,054 Pa (85 psi). For the experimental validation, the cycle-time of covering all phases (i.e., injection, packing, holding, cooling and ejection) was varied from 91 to 61sec. The flow rate of the coolant was controlled by the supply valves.

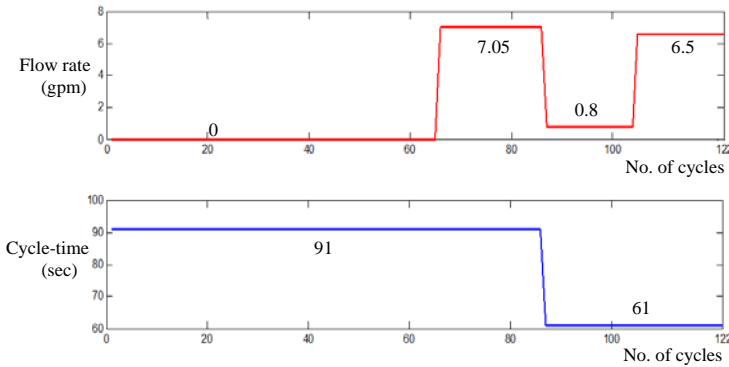


Fig. 11. Input conditions for experimental validation data.

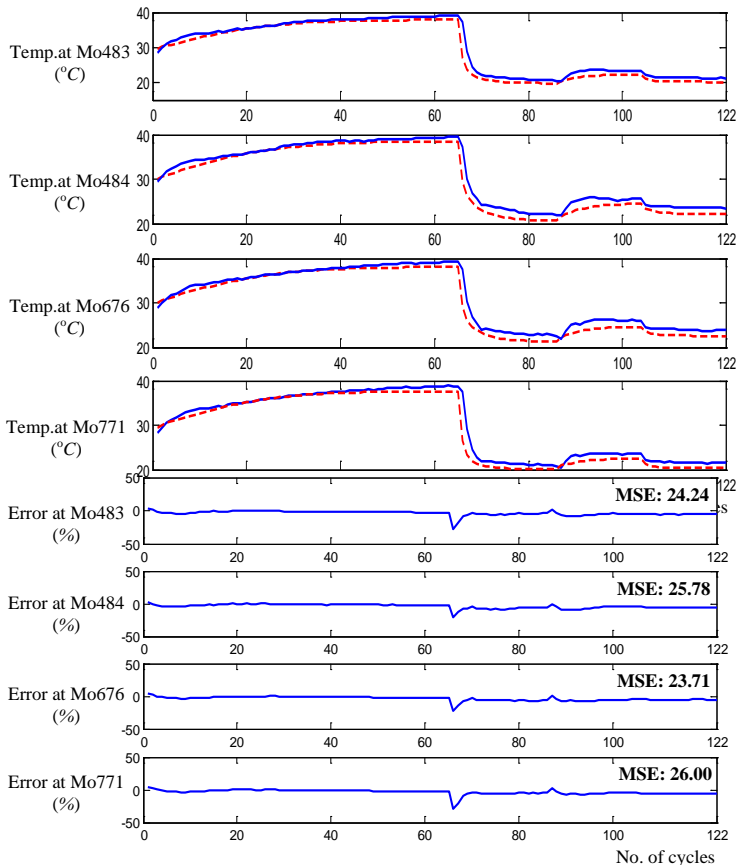


Fig. 12. Experimental validation results (comparison between model (Y_N) and actual outputs (Y)).

Fig. 13. Experimental validation results (error between Y_N and Y).

Using the input conditions (flow rate: 0–7.05–0.8–6.5gpm which are not used for modeling in Table 2; cycle-time: 91–61sec during 122 cycles) for experimental validation in Fig. 11, the NARX model was evaluated. By comparing the outputs of the NARX model and the experimental system in Fig. 12–13, it can be seen that the NARX model's outputs almost match the experimental outputs. Therefore, the NARX model obtained using data sets obtained from the FEA model has the ability to predict actual temperature responses at selected sensor locations.

After the model validation, the FE model was used to collect data sets for the NARX model. The NARX model was then used for controller development. In reality, it is less repeatable and in fact expensive to obtain experimental data sets for identifying the NARX model. From these points of view, the controllers were developed based on the NARX model after identification.

4. CONTROLLER DESIGN

4.1. Control of cavity-wall temperature

Cooling time is the time that the part cools from melt temperature to demoulding temperature at which it is ejected from the cavities. Therefore, the cavity-wall temperature after part ejection can indicate the quality of the final product. Specifically, since the cooling efficiency of a mould depends on the amount of heat extracted from the plastic material in the cavity at a certain cavity-wall temperature during the cycle, the desired temperature of the cavity wall should be ensured for the better quality (i.e., degree of defects in the part such as shrink porosity and poor fill) of final product after ejection. However, because the cavity wall is an area where locating sensors is challenging, the temperature measured at 4 nodes can be used instead of the cavity-wall temperature by utilizing the relation between them. This relation could be obtained from the training data sets used for NARX modeling, and is presented in Fig. 14.

The figure shows a linear relation between an average of four T_a at 4 nodes and cavity-wall temperature right after ejection. Since an average of four T_a is relatively insensitive to noise, this value is more suitable to use for analyzing the cavity-wall temperature rather than each T_a . R -squared value of 0.624 ($R=0.787$) represents a strong relationship between an average of four T_a (range: 20.45–29.21°C) and cavity-wall temperature after ejection (range: 23.26–31.77°C).

When the part reaches the desired demoulding

temperature for this material (Santoprene 8211-45: 85 °C [19]), the cavity-wall temperature is around 25°C, which corresponds to 22°C for an average of four T_a . Therefore, 22°C will be used as an objective value to control the quality of moulding process (or product quality). Although the Fig. 14 shows some fluctuations in cavity-wall temperature (maximum value of 5°C), the fluctuation at the objective value for cooling (22°C for an average of four T_a) is within 1.14°C (=25.77°C-24.63°C). This value represents the control accuracy of 5.7% (=1.14°C/22°C), which is acceptable for this product quality.

4.2. Fuzzy logic control

Fuzzy logic control is an effective approach in uncertain systems for which it is difficult to obtain a mathematical model. By capturing the knowledge of humans into the fuzzy rules, fuzzy logic enables to implement a controller without an in-depth knowledge on the system dynamics.

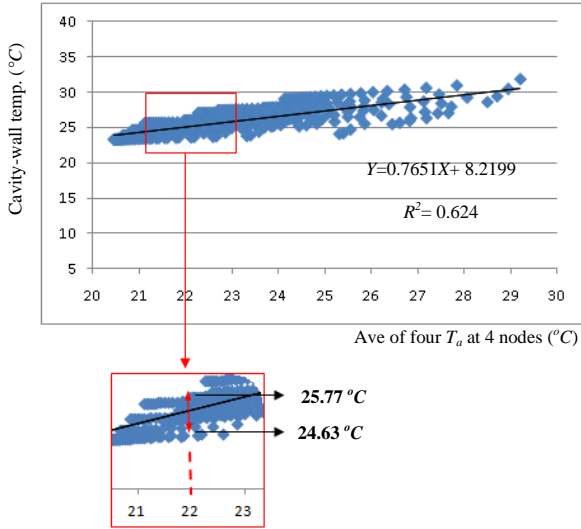


Fig. 14. Linear relationship between an average of four T_a at 4 nodes and cavity-wall temperature right after ejection.

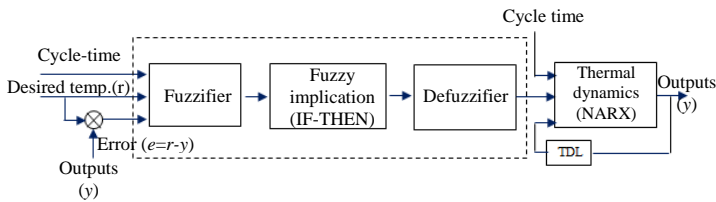


Fig. 15. Fuzzy control structure.

Therefore, a fuzzy logic control will be more appropriate to deal with the temperature control of our die systems due to uncertain dynamics. Figure 15 shows the structure of a fuzzy control system used for this study. Note that TDL refers to a unit time (cycle-time) delay.

Fuzzified input variables are the desired temperature of an average of four T_a , error and cycle-time. Besides

the error variable commonly used in fuzzy control, the desired temperature and cycle-time variables are additionally considered.

As one of input variables in fuzzy control, the cycle-time affects T_a at each node (and thus average of four T_a). More specifically, short cycle-time increases T_a when the water is provided (i.e., non zero gpm of the flow rate) since the cooling time as a dominant phase of the cycle time is decreased in shorter cycle-time whereas the time for the other phases (injection, packing, holding and ejection) is fixed. On the other hand, T_a in short cycle-time is decreased compared to one with longer cycle time in the case of non-supply of the water (i.e., no time for cooling). Therefore, the flow rate as an output variable in fuzzy control should be controlled differently along with the cycle-time and whether the water is supplied or not.

The desired temperature is also introduced as another input variable. This feature distinguishes our control design from the previously developed ones in fuzzy thermal control area for die systems [20-22] which use only error and error-change as input variables. As seen in Fig. 4, each flow rate cannot make the temperature at nodes drop below a certain value (e.g., 23.5°C and 20.7°C for 1gpm and 8gpm, respectively with 91sec of cycle-time at Mo484) once it reaches a steady state value. This means each desired temperature value to be controlled needs a different flow rate suitable for the corresponding value. Therefore, only error and error-change terms are not enough to control different levels of T_a . For example, if there is the same magnitude of error (e.g., 1°C) between actual temperature and desired value at different levels of desired T_a (e.g., 23.5°C and 20.7°C at Mo484), then control strategy requires a different operating flow rate to track a different desired temperature (e.g., operating flow rates should vary around 1gpm and 8gpm for maintaining the desired values of 23.5°C and 20.7°C, respectively despite the same error of 1°C).

Using the input and output variables, the fuzzy control rules can be briefly expressed as:

$$\text{Rule: If } DT \text{ is } DT_i \text{ and } ET \text{ is } ET_j \text{ and } Ct \text{ is } Ct_k, \text{ THEN } Q \text{ is } Q_r, \quad (7)$$

where DT is the desired average of four T_a , ET is the error between actual temperature and desired value, C_t is the cycle-time, Q is the flow rate of the coolant. $i = LB, \dots, HVB$; $j = NE, ZE, PE$; $k = 61, \dots, 91$ seconds; $r = Off, \dots, H_b$ (refer to the membership functions in Fig. 16-(d)). All fuzzy rules applied to the thermal control can be drawn out as Table 3. For the 9th fuzzy rule in Table 3 as an example, if DT is L_S (Low_Small) and ET is PE (Positive Error) and C_t is 61sec, then Q is H_b (big) level of H (High) flow rate).

The membership functions and control rules in this fuzzy logic control are obtained by trial and error design procedures. The Mamdani method [23] is used for fuzzy

implication (i.e., if-then statement in Eq. (7)) and the centroid method is used for defuzzification of a fuzzy control inference.

4.3. Self-tuning PID with BP neural network

As an alternative approach, a PID controller with backpropagation (BP) neural network automatic tuning is applied. This PID controller overcomes a limitation of conventional PID controller that the determination of control parameters is not easy in nonlinear or uncertain systems, by the self-learning ability of tuning PID parameters online [24].

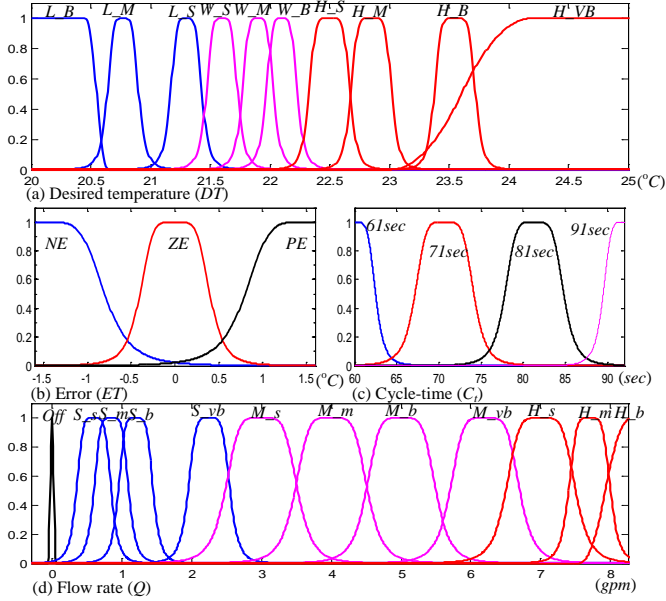


Fig. 16. Membership functions of fuzzy inputs (a), (b), (c) and output (d).

Table 3. Fuzzy control rules.

Rule No.	DT	ET	Ct	61 sec	71 sec	81 sec	91 sec
1	L _B (Low_Big)	NE (Negative Error)	H _b ^L	H _b	H _m ^K	H _m	
2		ZE (Zero Error)	H _b	H _b	H _m	H _m	
3		PE (Positive Error)	H _b	H _b	H _m	H _s ^J	
4	L _M (Low_Medium)	NE	H _b	H _b	H _m	H _m	
5		ZE	H _b	H _b	H _m	H _s	
6		PE	H _b	H _b	H _s	M _{vb} ^I	
7	L _S (Low_Small)	NE	H _b	H _m	H _m	H _s	
8		ZE	H _b	H _m	H _s	M _{vb}	
9		PE	H _b	H _m	M _{vb}	M _b ^H	
10	W _S (Warm_Small)	NE	H _m	H _m	H _s	M _{vb}	
11		ZE	H _s	H _s	M _{vb}	M _b	
12		PE	M _{vb}	M _b	M _b	M _m ^G	
13	W _M (Warm_Medium)	NE	H _s	H _s	M _{vb}	M _b	
14		ZE	M _{vb}	M _{vb}	M _b	M _m	
15		PE	M _b	M _b	M _m	M _s ^F	
16	W _B (Warm_Big)	NE	M _{vb}	M _b	M _m	M _m	
17		ZE	M _b	M _m	M _m	M _s	
18		PE	M _m	M _s	M _s	S _{vb} ^E	
19	H _S (High_Small)	NE	M _b	M _b	M _m	M _s	
20		ZE	M _m	M _m	M _s	S _{vb}	
21		PE	M _s	S _{vb}	S _{vb}	S _b ^D	

22	H _M (High_Medium)	NE	M _m	M _m	S _{vb}	S _{vb}
23		ZE	M _s	S _{vb}	S _b	S _b
24		PE	S _{vb}	S _b	S _b	S _b
25	H _B (High_Big)	NE	M _s	S _{vb}	S _{vb}	S _{vb}
26		ZE	S _{vb}	S _b	S _b	S _b
27		PE	Off	Off	Off	Off
28	H _{VB} (High_Very Big)	NE	S _m ^C	S _m	S _m	S _s ^B
29		ZE	Off ^A	Off	Off	Off
30		PE	Off	Off	Off	Off

^A means zero flow rate; ^{B,C,D,E} mean s (small), m (medium), b (big), vb (very big) levels of S (Small) flow rate; ^{F,G,H,I} mean s (small), m (medium), b (big), vb (very big) levels of M (Medium) flow rate; ^{J,K,L} mean s (small), m (medium), b (big) levels of H (High) flow rate.

The schematic diagram of self-tuning PID control based on BP neural network is given in Fig. 17. The PID control algorithm is given as follows:

$$\begin{aligned} u(k) &= u(k-1) + \Delta u(k) \\ &= u(k-1) + K_p \Delta e_c(k) + K_i e_c(k) + K_D \Delta^2 e_c(k), \end{aligned} \quad (8)$$

where $u(k)$ is the control variable at the time instant k . $\Delta u(k)$ is the velocity form of the PID algorithm. K_p , K_i , K_D are proportional, integral and derivative gains, respectively. $e_c(k) = r(k) - y(k)$ is the control error; $\Delta e_c(k) = e_c(k) - e_c(k-1)$; $\Delta e_c^2(k) = e_c(k) - 2e_c(k-1) + e_c(k-2)$.

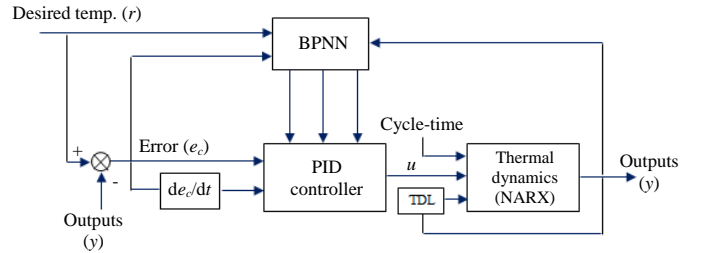


Fig. 17. Self-tuning PID control with BP neural network.

Three-layer BP neural network was used with the following structure:

The inputs and outputs in the input layer are given by

$$\begin{aligned} x_j(k) &= [x_1(k), x_2(k), x_3(k)] \\ &= [r_{Nor}(k), y_{Nor}(k), e_{Nor}(k)] \\ o_j^{(i)}(k) &= \begin{cases} x_j(k), & j = 1, 2, 3 \\ 1, & j = 4, \end{cases} \end{aligned} \quad (9)$$

where $x_j(k)$ and $o_j^{(i)}$ are inputs and outputs in the input layer. $r_{Nor}(k)$, $y_{Nor}(k)$ and $e_{Nor}(k)$ are the dimensionless desired value of average of four T_a , dimensionless output of NARX model and dimensionless error, respectively which are normalized by Eq. (3). i denotes the input layer.

The inputs and outputs in the hidden layer and ones in the output layer are defined in Eq. (10) and Eq. (11), respectively.

$$net_l^{(h)}(k) = \sum_{j=1}^n w_{lj}^{(h)} o_j^{(i)}(k) \quad (10)$$

$$o_l^{(h)}(k) = \begin{cases} f[net_l^{(h)}(k)], & l=1, 2, \dots, p-1 \\ 1, & l=p, \end{cases}$$

$$net_r^{(o)}(k) = \sum_{l=1}^p w_{rl}^{(o)} o_l^{(h)}(k) \quad (11)$$

$$\begin{aligned} o_r^{(o)}(k) &= g[net_r^{(o)}(k)], \quad r=1, 2, 3 \\ &= [o_1^{(o)}(k), o_2^{(o)}(k), o_3^{(o)}(k)] \\ &= [K_p(k), K_I(k), K_D(k)], \end{aligned}$$

where $net_l^{(h)}$, $o_l^{(h)}$, $w_{lj}^{(h)}$ are inputs, outputs, weights in the hidden layer. $net_r^{(o)}$, $o_r^{(o)}$, $w_{rl}^{(o)}$ are inputs, outputs, weights in the output layer. f and g are $\tanh(x)$ and $(1+\tanh(x))/2$, respectively. h and o denote the hidden layer and the output layer, respectively.

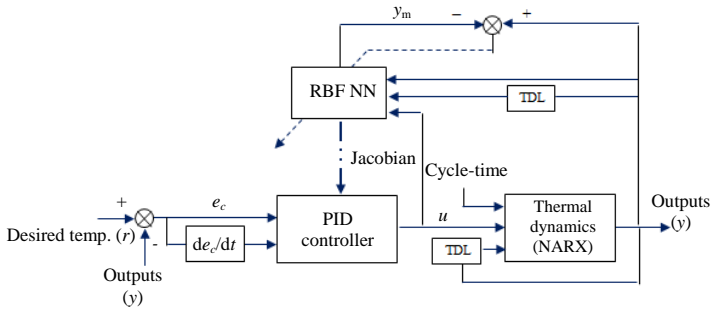


Fig. 18. Self-tuning PID control with RBF neural network.

Using the gradient-descent (or steepest descents) algorithm [25], the weights in the output and hidden layer are updated by the following equations [26].

$$\begin{aligned} \Delta w_{rl}^{(o)}(k) &= \eta \delta_r^{(o)} o_l^{(h)}(k) + \alpha \Delta w_{rl}^{(o)}(k-1) \\ \delta_r^{(o)} &= e(k) \frac{\partial y(k)}{\partial \Delta u(k)} \frac{\partial \Delta u(k)}{\partial o_r^{(o)}(k)} g'[net_r^{(o)}(k)], \quad r=1, 2, 3, \end{aligned} \quad (12)$$

$$\begin{aligned} \Delta w_{ij}^{(h)}(k) &= \eta \delta_l^{(h)} o_j^{(i)}(k) + \alpha \Delta w_{ij}^{(h)}(k-1) \\ \delta_l^{(h)} &= f'[net_l^{(h)}(k)] \sum_{r=1}^3 \delta_r^{(o)} w_{rl}^{(o)}(k), \quad l=1, 2, \dots, p-1, \end{aligned} \quad (13)$$

where η is the leaning speed rate and α is the momentum factor for fast convergence.

4.4. Self-tuning PID with RBF neural network

Another type of neural network using the radial basis function (RBF) is considered for design of self-tuning PID controller, due to the advantages of RBF neural network such as simple structure, faster learning algorithms and effective mapping between a controlling system's input and output [27].

Figure 18 describes the structure of self-tuning PID controller based on RBF neural network. Main function of this control technique is to first identify the plant

dynamics and secondly adjust the PID parameters adaptively based on RBF neural network identification (specifically, Jacobian information, $\partial y/\partial \Delta u$ (see Eq. (19)).

The input vector, output vector in hidden layer and weight vector between hidden layer and output layers in RBF network are respectively,

$$\begin{aligned} X &= [x_1, x_2, \dots, x_n]^T \\ &= [\Delta u_{Nor}(k), y_{Nor}(k), y_{Nor}(k-1)]^T \\ H &= [h_1, h_2, \dots, h_m]^T \\ W &= [w_1, w_2, \dots, w_m]^T, \end{aligned} \quad (14)$$

where X , H and W are the input vector, output vector in hidden layer and weight vector between hidden and output layers. $\Delta u_{Nor}(k)$ and $y_{Nor}(k)$ are normalized values of $\Delta u(k)$ and $y(k)$ by Eq. (4) and (3), respectively. Each element (h_j) of vector H has a format of the Gaussian kernel function given by

$$\begin{aligned} h_j &= \exp\left[-\left(\frac{\|X - C_j\|^2}{2b_j^2}\right)\right], \quad j=1, \dots, m \\ C_j &= [c_{1j}, c_{2j}, \dots, c_{ij}, \dots, c_{nj}]^T, \end{aligned} \quad (15)$$

where h_j is the Gaussian kernel function, C_j and b_j are the center and width of h_j .

The output of RBF network, y_m is expressed as:

$$y_m = w_1 h_1 + w_2 h_2 + \dots + w_m h_m, \quad (16)$$

To update w and b_j and C_j , the cost function is required as:

$$J(k) = \frac{1}{2} [y(k) - y_m(k)]^2, \quad (17)$$

By using the gradient-descent technique and Eq. (17), and w and b_j and C_j are updated according to the following algorithms ([27]).

$$\begin{aligned} w_j(k) &= w_j(k-1) - \eta^* \frac{\partial J}{\partial w_j(k)} + \alpha^* (w_j(k-1) - w_j(k-2)) + \beta^* (w_j(k-2) - w_j(k-3)) \\ b_j(k) &= b_j(k-1) - \eta^* \frac{\partial J}{\partial b_j(k)} + \alpha^* (b_j(k-1) - b_j(k-2)) + \beta^* (b_j(k-2) - b_j(k-3)) \\ c_{ij}(k) &= c_{ij}(k-1) - \eta^* \frac{\partial J}{\partial c_{ij}(k)} + \alpha^* (c_{ij}(k-1) - c_{ij}(k-2)) + \beta^* (c_{ij}(k-2) - c_{ij}(k-3)), \end{aligned} \quad (18)$$

where η^* is the leaning speed rate. α^* and β^* are the momentum factors.

By the gradient descent method, the adjustment of PID parameters [27] is given by

$$\begin{aligned}
 K_p(k) &= K_p(k-1) + \eta^{**} e_c(k) \frac{\partial y}{\partial \Delta u} \Delta e_c \\
 K_I(k) &= K_I(k-1) + \eta^{**} e_c(k) \frac{\partial y}{\partial \Delta u} e_c \\
 K_D(k) &= K_D(k-1) + \eta^{**} e_c(k) \frac{\partial y}{\partial \Delta u} \Delta^2 e_c(k),
 \end{aligned} \tag{19}$$

where η^{**} is the leaning speed rate for PID parameter update and $\partial y / \partial \Delta u$ is the Jacobian information of controlled plant.

4.5. Performance of designed controllers

By adopting aforementioned three control techniques, the performance of the controllers was evaluated through simulations with MATLAB/Simulink.

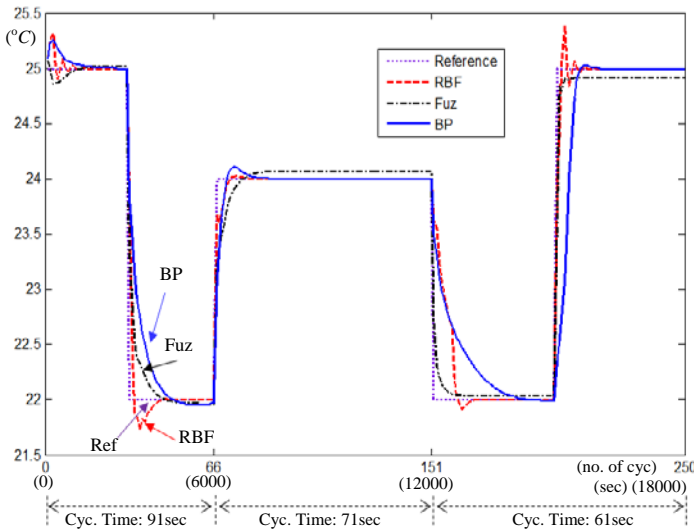


Fig. 19. Each controller's performance for multi-setpoints with various cycle-time.

Figure 19 shows the variation of average of four T_a (at 4 nodes) in NARX model with each controller. For the first range (0–6000sec), the cycle-time of 91sec (thus 66 no. of cycles) is considered with multi-setpoints of 25°C and 22°C (which corresponds to the demoulding temperature). Then, set-point is increased to 24°C and kept constant during 6000–12000sec with a different cycle-time of 71sec (total 85 cycles). Then multi-setpoints of 22°C and 25°C are used during 12000–18000sec with the cycle-time of 61sec (total 99 cycles).

Although the fuzzy logic controller shows an acceptable control performance for all cycle-times, its tracking error is slightly greater than that of the other controllers during 6000–12000sec (error: -0.07°C) and 12000–18000sec (errors: -0.03°C for 22°C and $+0.08^\circ\text{C}$ for 25°C).

Self-tuning PID controllers with BP and RBF show better tracking performance for all ranges. However, the PID controller with BP neural network reacts relatively slowly. For example, it takes 39 cycles (=190-151

cycles) for the PID controller with BP to reach 22°C right after changing the cycle-time from 71 to 61sec, whereas those with RBF and fuzzy logic require 16 and 14 cycles, respectively that are considered acceptable as a response time.

To cover various cycle-times (various thermal dynamics) of moulding process, a fuzzy logic controller needs the refined membership functions for output variable (flow rate) as seen in Fig. 16. This makes the controller design more complicated. In spite of having refinement of membership functions, it is difficult to provide an accurate performance for all cycle-times. By considering the limitation of the fuzzy controller and slow response of the self-tuning PID controller with BP, the performance of RBF based self-tuning PID is superior to the others under thermal dynamics with various cycle-times.

5. CONCLUSION

This study addresses the thermal control strategies for plastic injection moulding in a laminated die. In this study, a thermal model was developed using FEA and NN techniques that enabled to tackle the problem of various cycle-times and uncertain dynamics of a laminated die system. The identified model was validated using experimental tests and showed a good result in predicting the temperature distributions at selected sensor locations.

Based on this model, fuzzy-logic and self adaptive PID controllers by applying BP and RBF neural networks were introduced and each controller's performance was evaluated. The fuzzy controller showed less accurate tracking despite refined membership functions for output variable to cope with various cycle-times. By utilizing learning algorithms to tune control parameters on-line, both self-tuning PID controllers exhibited accurate control performance. However, the self-tuning PID controller with BP exhibited slower response than one with RBF. Therefore, the RBF based self-tuning PID controller was effective in terms of tracking accuracy and fast response for the quality of plastic injection moulding process with diverse cycle-times.

By combining several existing advanced techniques including FEA, nonlinear modeling and control associated with neural networks, this study can fundamentally contribute to formulating a unique framework for mould thermal control. Also, since the overall methods in the study can be applicable to the thermal management problem of general systems, these methods can be extended to other relevant industrial problems including plastic processing, microelectronic fabrication, etc.

REFERENCES

- [1] W. Bishenden and R. Bhola, "Die temperature control", *Transaction of the 20th International Die Casting Congress and Exposition, North American Die Casting Association*, pp. 161-164, 1999.

- [2] L. Kong, F. She, W. Gao, S. Nahavandi and P. Hodgson, "Integrated optimization system for high pressure die casting processes", *Journal of Materials Processing Technology*, vol. 201, pp. 629-634, 2008.
- [3] R. Dubay, B. Pramujati and J. Hernandez, "Cavity temperature control in injection molding", *2005 IEEE International Conference on Mechatronics and Automation*, pp. 911-916, 2005.
- [4] F. Gao, W. Patterson and M. Kamal, "Dynamics and Control of Surface and Mold Temperature in Injection Molding", *International Polymer Processing*, vol. 8, pp. 147-157, 1993.
- [5] R. Dubay, "Self-optimizing MPC of melt temperature in injection moulding", *ISA Transactions*, vol. 41, no. 1, pp. 81-94, 2002.
- [6] C. Lu and C. Tsai, "Adaptive decoupling predictive temperature control for an extrusion barrel in a plastic injection molding process", *IEEE Transactions on Industrial Electronics*, vol. 48, no. 5, pp. 968-975, 2001.
- [7] C. Tsai and C. Lu, "Multivariable self-tuning temperature control for plastic injection molding process", *IEEE Transactions on Industry Applications*, vol. 34, no. 2, pp. 310-18, 1998.
- [8] R., Vetter, D. Maijer, M. Huzmezan and D. Meade, "Control of a die casting simulation using an industrial adaptive model-based predictive controller", *Proceedings of the Symposium Sponsored by the Extraction and Processing Division of the Minerals, Metals and Materials Society*, pp. 235-246, 2004.
- [9] P. Dickens, "Research developments in rapid prototyping", *Proc. Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 209, pp. 261-266, 1995.
- [10] H. Ahari, A. Khajepour and S. Bedi, "Manufacturing optimization of laminated tooling with conformal cooling channels", *Rapid Prototyping Journal*, vol. 17, no. 6, pp. 429-440, 2011.
- [11] X. Xu, E. Sachs and S. Allen, "The design of conformal cooling channels in injection molding tooling", *Polymer Engineering and Science*, vol. 41, no. 7, pp. 1265-1279, 2001.
- [12] H. Hu, F. Chen, X. Chen, Y. Chu and P. Cheng, "Effect of cooling water flow rates on local temperatures and heat transfer of casting dies", *Journal of Materials Processing Technology*, vol. 148, pp. 439-451, 2004.
- [13] G. Potsch and W. Michaeli, *Injection molding: An introduction*, Hanser/Gardner, 1997.
- [14] J. Seo, A. Khajepour and J. Huissoon, "Optimal sensor location in laminated die system", *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 134, no. 2, 2011.
- [15] J. Hernandez, R. Dubay and B. Pramujati, "A new scheme for controlling part cooling in plastic injection molding", *International Journal of Polymer Processing*, vol. 22, no. 3, pp. 276-283, 2007.
- [16] X. He and H. Asada, "A new method for identifying orders of input-output models for nonlinear dynamic systems", *Proc. American Control Conference*, pp. 2520-2523, 1993.
- [17] J. Sola and J. Sevilla, "Importance of input data normalization for the application of neural networks to complex industrial problems", *Transactions on Nuclear Science*, vol. 44, pp. 1464-1468, 1997.
- [18] <http://www.webplas.ca>
- [19] <http://www.santoprene.com>
- [20] C. Gong, Y. Feng, J. Wang and Y. Song, "A fuzzy-PID control system of PTFE sintering furnace based on Lonworks", *26th China Control Conference*, pp. 266-269, 2007.
- [21] T. Yang, H. Hu, X. Chen, Y. Chu and P. Cheng, "An intelligent control system for die thermal management", *Proc. Automation & Control/Advanced Metallurgical Models/Sensors*, pp. 11-20, 2005.
- [22] T. Yang, X. Chen and H. Hu, "A fuzzy PID thermal control system for die casting processes", *IEEE Multi-conference on Systems and Control*, pp. 389-394, 2008.
- [23] E. Mamdani, "Advances in the linguistic synthesis of fuzzy controllers", *International Journal of Man-Machine Studies*, vol. 8, pp. 669-678, 1976.
- [24] B. Guo, H. Liu, Z. Luo and F. Wang, "Adaptive PID Controller Based on BP Neural Network", *International Joint Conference on Artificial Intelligence*, pp. 148-150, 2009.
- [25] G. Arfken, *Mathematical Methods for Physicists*, Academic Press, 1985.
- [26] J. Wang, C. Zhang and Y. Jiang, "Adaptive PID control with BP neural network self-tuning in exhaust temperature of micro gas turbine", *3rd IEEE Conference on Industrial Electronics and Applications*, pp. 532-537, 2008.
- [27] M. Zhang, X. Wang and M. Liu, "Adaptive PID Control Based on RBF Neural Network Identification", *17th IEEE International Conference on Tools with Artificial Intelligence*, pp. 681-683, 2005.
- [28] J. Jiang, S. Wen, Z. Zhou and H. He, "Fuzzy barrel temperature PID controller based on neural network", *CISP '08 Proceedings of the 2008 Congress on Image and Signal Processing*, pp. 90-94, 2008.
- [29] L. Yu, S. Fei, and X. Li, "RBF neural networks-based robust adaptive tracking control for switched uncertain nonlinear systems", *International Journal of Control, Automation, and Systems*, vol. 10, no. 2, pp. 437-443, 2012.



Jaho Seo received the B.S degree in agricultural machinery and process engineering from Seoul National University, Seoul, Korea, in 1999, the M.E degree in mechanical engineering

from University of Quebec (Ecole de Technologie Superieure), Montreal, Canada, in 2006 and the Ph.D. degree in mechanical engineering from University of Waterloo, Waterloo, Canada, in 2011.

He was a chief engineering officer (2000-2002) and metal casting officer (2002-2003) with Republic of Korea Navy, Korea. He was with the Department of Mechanical and Mechatronics Engineering, University of Waterloo as a postdoctoral fellow in 2011. Since 2012, he has been a senior researcher in the Department of System Reliability at Korea Institute of Machinery & Materials (KIMM), Daejeon, Korea where he is involved in research on control systems for construction equipments and wind turbines.



Amir Khajepour received the Ph.D. degree in 1996 from the University of Waterloo, ON, Canada, where he is currently a professor in the Mechanical and Mechatronics Engineering and the Canada Research Chair in “Mechatronic Vehicle Systems”. He is an expert in modeling and control of dynamic systems and developed extensive

research programs that been utilized in several key multidisciplinary areas. His research resulted in several patents, technology transfers, and over 270 publications including 3 books and 6 book chapters. He is a fellow of American Society of Mechanical Engineering and Canadian Society for Mechanical Engineering. He is an associate editor of the International Journal of Vehicle Autonomous Systems and International Journal of Powertrain.



Jan Paul Huissoon received his PhD in mechanical engineering from Trinity College Dublin, Ireland, in 1983. In 1986 he joined the University of Waterloo, Ontario, Canada, where he is currently professor and deputy chair in the Department of Mechanical and Mechatronics Engineering. His research interests include automatic calibration

and intelligent control of robotic GMA welding, autonomous robotic and intelligent vehicle systems, and mechatronics design. He is a member of the Institute of Engineers of Ireland and of The American Welding Society, and is a registered professional engineer in Ontario.