Smith Predictor-based Control of Systems with Time Delay

by

Ivan Manuel Iampita
17866

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Engineering (Hons)
(Electrical and Electronics)

January 2017

Universiti Teknologi PETRONAS

Bandar Seri Iskandar

32610 Tronoh

Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Smith Predictor-based Control of Systems with Time Delay

by

Ivan Manuel Iampita

17866

A project dissertation submitted to the

Electrical and Electronics Engineering Programme

Universiti Teknologi PETRONAS

in partial fulfillment of the requirement for the

BACHELOR OF ENGINEERING (Hons)

(ELECTRICAL & ELECTRONICS)

Approved by,

____

Ms. Suhaila Badarol Hisham

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

January 2017

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not undertaken or done by unspecified sources or persons.

____

IVAN MANUEL IAMPITA

ABSTRACT

Smith Predictor is an effective method for control of processes in which the process time delay is long as it improves the performance of a PI controller. However, processes are susceptible to delay variation due to uncertainties and this predictive model does not guarantee plant robustness and stability due to the increase of negative phase lag. A temperature control experiment is conducted in a shell-tube Heat Exchanger Process Pilot Plant due to its time delay presence characteristic of interest. The process model in the form of transfer function is identified using Statistical Modelling with accuracy of 93.86%. A Smith Predictor-PI controller is developed using Matlab/Simulink to simulate the control of the temperature loop and results using performance indicators such as Integral Absolute Error (IAE) and Integral Square Error (ISE) show that it performs better than a PI controller. Due to robust tuning the Classical Smith Predictor (SP) shows good robust control and presents good performance indicators when process delay is varied in the estimated bounded interval from 0.21 minutes to 0.33 minutes. However, results from the Classical SP show that an improved control performance can be attained with a Filtered Smith Predictor (FSP). The FSP developed shows a fast rise time, settling time and overall lower ISE and IAE values than the Classical SP-PI and PI controller. Analysis on the process time delay variation shows that the FSP control structure can withstand the changes in process time delay while providing good performance indicators in the estimated range determined by the upper and lower bounded internal from 0.21 minutes to 0.31 minutes.

ACKNOWLEDGEMENT

It has been an immense honor to have worked on this project *Smith Predictor-based Control of Systems with Time Delay*, a Final Year Project as in partial fulfillment of the requirement for the Bachelor of Engineering (Hons) Electrical and Electronics.

Much gratitude goes to Universiti Teknologi Petronas and the Department of Electrical and Electronics Engineering for facilitating and providing resources for the execution of this project. I would like as well to extend my utmost gratitude for Ms. Suhaila Badarol Hisham, my Supervisor for all the support, guidance, wisdom provided throughout the project. I am truly thankful and honored to have worked with her as she has challenged me to deliver good work and that has helped me to mold into a better student as well as a better professional.

Furthermore, I would like to thank Mr. Azhar Zainal for facilitating the execution of the experiment needed to carry out this project as well as all the examiners who offered thoughtful suggestions for the improvement of this work.

Immense appreciation goes to my family and friends for the support, availability and motivation extended to me throughout the whole project and superseding all above I am thankful to God for the countless blessings, strength and perseverance He provided me throughout the execution of this project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS AND NOMENCLATURES

FOPDT	First-Order-Plus-Dead-Time
CE	Characteristic Equation
SP	Smith Predictor
FSP	Filtered Smith Predictor
MSP	Modified Smith Predictor
CDM	Coefficient Diagram Method
2DOF	Two Degree of Freedom
ZOH	Zero-Order Hold
IAE	Integral Absolute Error
ISE	Integral Squared Error
S-IMC	Simple Internal Method Control
PID controller	Proportional-Integral-Derivative controller

CHAPTER 1
INTRODUCTION

1.1 Background

The control of processes with time delay are often challenging and it affects the system performance greatly. Although PID controllers are the main approach for control of processes, they may exhibit poor performance in the presence of dead time. Dead time introduces less control to the aggressive changes caused by the system and it is usually part of process dynamics originated from the association of different parts of the process. Dead time is determined as the amount of time it takes for a control signal effect to be felt on the plant as to control a desired condition, for instance temperature [1-4].

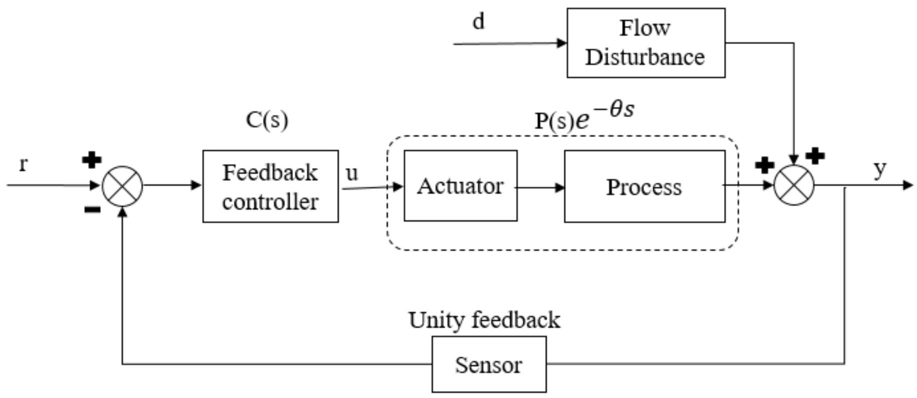

Figure 1: Temperature control loop block diagram

Consider the block diagram in Figure 1 above for a temperature control loop, in which the actual process is denominated by $P(s)e^{-\theta s}$, a FOPDT system, therefore the closed-loop transfer function becomes [5]:

$$T_{yr} = \frac{C(s)P(s)e^{-\theta s}}{1 + C(s)P(s)e^{-\theta s}} \tag{1}$$

The characteristic equation (CE) of the closed loop system is denoted by,

$$1 + C(s)P(s)e^{-\theta s} = 0$$

The presence of the delay element $e^{-\theta s}$, in the closed-loop CE makes it difficult to design a controller that can guarantee the process stability and system performance.

PID controllers are ineffective to processes where time delay is long as they may produce unsatisfactory results and it is difficult to tune parameters to achieve a desired control performance [5, 6]. A Smith Predictor provides an adjustment around an existing PI controller so that aggressive changes in output variable helps to overcome the unwanted effects caused by the time delay. Hence, the application of the Smith Predictor in this project to improve the plant performance.

1.2 Problem Statement

Smith Predictor can be employed together with a PI controller to reduce the unwanted effects caused by process delay. However, the above controller does not account for robust stability and performance of the plant and does little to the disturbance rejection. Moreover, it has limits reducing the effect of process time delay variation which may lead the plant to instability as well as it is dependent on the accurate representation of the process. Hence, a technique that accounts for robust stability and improved performance of the SP control structure is required as to study the limitations of the controller for a closed-loop temperature control when the process delay varies within a norm-bounded interval.

1.3 Objectives

1. Develop a controller based on a Smith Predictor for temperature control loop with inherent time delay and accuracy above 80 percent.
2. Perform the robustness analysis of the plant
3. Investigate the limits of the above controller when the process time delay varies within specified range

1.4 Scope of Study

The focus of this project is on the study of a Smith Predictor for the compensation of dead-time for a temperature control process loop in the case that a PI controller alone is ineffective. The study of this project involves different stages.

Firstly, different techniques of the Smith Predictor are discussed in the literature review to improve the robust stability and performance of this control structure. A temperature control loop experiment is conducted on a shell-tube heat exchanger and the dynamic behavior of the process is extracted.

Secondly, a PI and a Classical Smith Predictor-PI based controllers are developed to prove the benefits of the latter. The Classical SP is analyzed when process delay varies and subsequently to improve the performance of the Classical Smith Predictor-PI based controller a Filtered Smith Predictor (FSP) controller is developed. For all controllers, a robust tuning is considered to provide best control.

Lastly, a study on the limitations of the FSP to reduce the effects of time delay when the temperature control loop process delay varies is done. All simulations for the temperature control loop performance are done on Matlab/Simulink. Then the result analysis and discussions are carried.

CHAPTER 2
LITERATURE REVIEW

This section overviews the Classical Smith Predictor and reviews different versions of the Smith Predictor control structure designed to improve the robustness of this technique and provide better control.

2.1 Overview of the Classical Smith Predictor

Smith Predictor is a dead time control structure designed to improve the performance of closed-loop stable processes with long time delays. This technique has proven to be an effective predictive control algorithm for time delay compensation [7, 8].

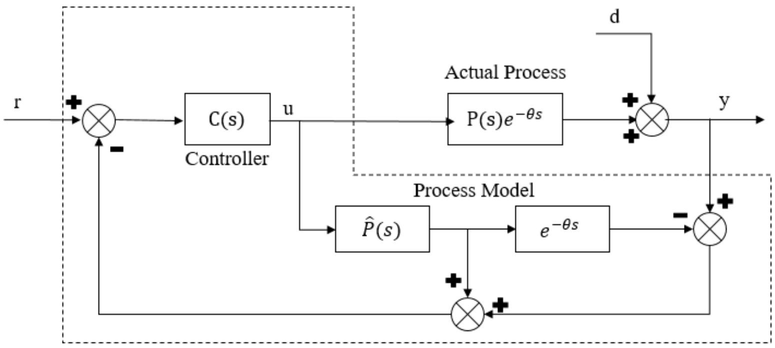

Figure 2: Classical Smith Predictor Control Structure

The block diagram in Figure 2 above represents the conventional Smith Predictor control structure. It consists of a controller $C(s)$, the actual process with the inherent process time delay $P(s)e^{-\theta s}$, an inner loop with the model representation of the plant, $\hat{P}(s)$, and the prediction of the actual dead time, $e^{-\hat{\theta}s}$, in $\hat{\theta}$ minutes into the future[7]. Wade further suggests that modelling of the Smith Predictor is applied to two parts, the linear $\hat{P}(s)$ and nonlinear $e^{-\hat{\theta}s}$ portions. $\hat{P}(s)$ in practice is an open loop stable first order lag, therefore, the transfer function for the closed-loop diagram, assuming perfect model representation and disturbance ($d = 0$), is as follow:

$$T_{yr} = \frac{C(s)P(s)e^{-\theta s}}{1 + C(s)\hat{P}(s) + C(s)\left[P(s)e^{-\theta s} - \hat{P}(s)e^{-\hat{\theta} s}\right]} \tag{2}$$

$$T_{yr} = \frac{C(s)P(s)e^{-\theta s}}{1 + C(s)\hat{P}(s)} \tag{3}$$

The characteristic equation (CE) of the closed-loop system, $1 + C(s)P(s) = 0$, appears without the dead time element which allows easier adjustment of the manipulated variable to be implemented in the actual process, hence allowing much more aggressive control of the model than the plant which could yield a good control as long as the model and the plant are reasonably accurate [5, 7, 8]. Therefore, the Smith Predictor control structure offers a method that removes the time delay term to outside the feedback control loop thus allowing to design the controller according to the delay free path of the plant, hence offering more control related to the process performance [7-11]. Figure 3 below shows the Smith Predictor apparent control loop.

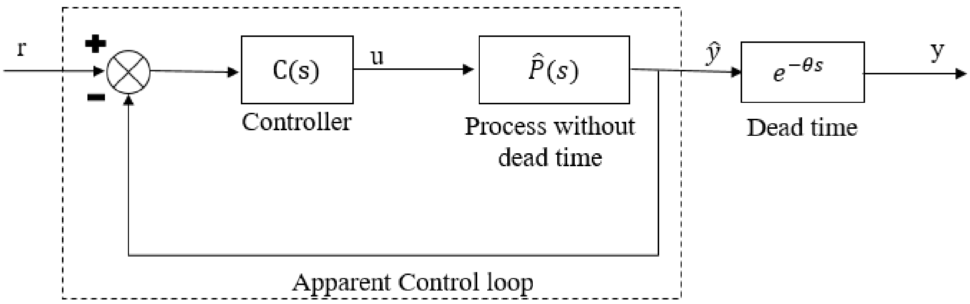

Figure 3: Smith Predictor apparent control loop

In the real world, it is impossible to find the exact model representation to the plant and the actual behavior and stability depends on all terms without cancellation, reports expressed both in [8, 11]. The classical Smith Predictor analysis is limited to stable processes with fixed time delays. This predictive model is sensitive to time variation and uncertainties as it introduces extreme instability to the system, hence the system performance deteriorates [10, 11]. Therefore, a need for a method that can compensate for the model errors such that it can make up for the limitations of the Smith Predictor is desirable.

2.2 Review of *Modified Smith Predictor using Coefficient Diagram Method (CDM)*

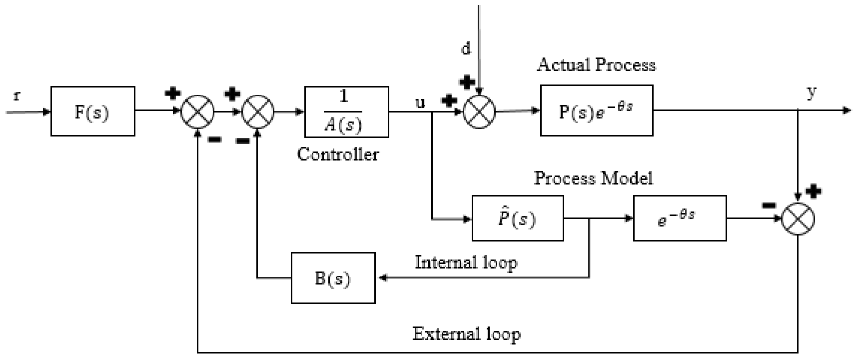

Figure 4: Modified SP using CDM

The block diagram above represents the Modified SP using CDM. The CDM uses a characteristic polynomial to represent a control system as shown in Equation 4. From Figure 4, $A(s)$ is the denominator of the tranfer function of the controller, $C(s)$, the reference and feedback transfer functions are the numerator $F(s)$ and $B(s)$ respectively [10].

$$P(s) = A(s)D(s) + B(s)N(s) \text{ where:}$$
$$A(s) = \sum_{i=0}^{n} l_i s^i \text{ and } B(s) = \sum_{i=0}^{n} k_i s^i \tag{4}$$

Based on the desired settling time t_s, a controller is tuned with the equivalent time constant τ parameter. τ specifies the time response speed. Another parameter to tune is the stability index γ_i, which is chosen to determine the stability and robustness when this parameter is varied. Both compute the target characteristic polynomial, Equation 5. Equations 4 and 5 are compared and the controller is designed, Equation 6 [10].

$$P_{target}(s) = a_0 \left\{ \left[\sum_{i=2}^{n} \left(\prod_{j=1}^{i-1} \frac{1}{\gamma_{i-j}^{j}} \right) (\tau s)^i \right] + \tau s + 1 \right\} \text{ where:} \tag{5}$$
$$\tau = t_s/(2.5 \sim 3) \text{ and } \gamma_i = [2.5,2,2,..],\ i = 1,...,n-1. \gamma_0 = \gamma_\infty = 0$$

$$C(s) = \frac{B}{A} = \frac{k_1 s + k_0}{s} = K_c\left(1 + \frac{1}{T_i s}\right) \tag{6}$$

Puawade et.al., claims that the Modified Smith Predictor using CDM can be implemented to design a controller to have the narrowest bandwidth, minimum degree, and a close loop time response with a non-existent or very small overshoot, and accounts also for the robust stability of the control system.

2.3 Review of *A two degree of freedom (2DOF) Modified Smith Predictor (MSP)*

The MSP consists of the classical SP with the addition of two filters in the control loop, $F_1(s)$ improves the reference signal and $F_2(s)$ improves the disturbance rejection. The proposed MSP is meant for process that can be approximated by FOPDT models, the block diagram is shown below in Figure 5.

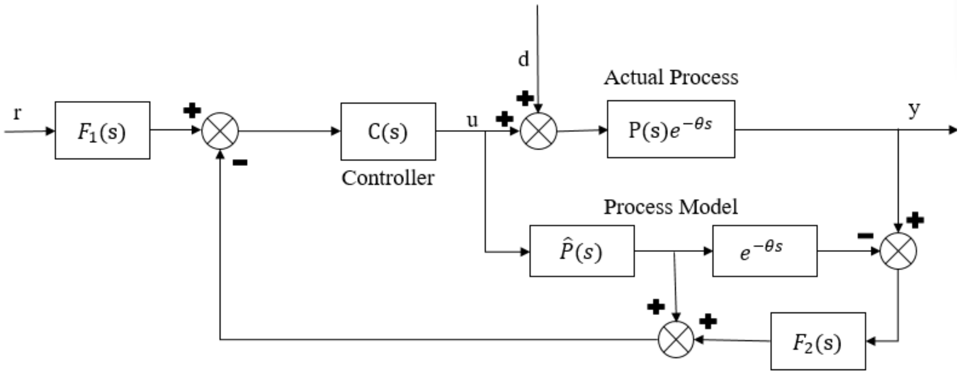

Figure 5: Block diagram of the 2DOF Modified SP

The process and disturbance closed-loop transfer function is written as follows:

$$T_{yr} = \frac{F_1(s)C(s)P(s)e^{-\theta s}}{1 + C(s)\hat{P}(s)} \quad \text{where: } F_1 = \frac{T_0 s + 1}{K_0 s + 1} \tag{7}$$

$$\frac{Y(s)}{D(s)} = P(s)\left[1 - \frac{F_2(s)C(s)P(s)e^{-\theta s}}{1 + C(s)\hat{P}(s)}\right] = P(s)\lambda \text{ where: } F_2 = \frac{T_0 s + 1}{K_1 s + 1} \tag{8}$$

The Filter $F_1(s)$ is tuned for set-point tracking in which T_0 is set equal to T as to cancel the long time constant and achieve faster response, $T_0 = T$. As shown in Equation 9, the gain of the plant is set to 1 as to allow proper set-point tracking. Hence, the reference filter depends monotonically on K_0, where $K_0 = T_0/7$. This value was derived as a basis of the desired set point. On the other hand, Filter $F_2(s)$, needs to set λ in Equation (8) and (10) close to zero for improved response and disturbance rejection which in turn is dependent on K_1 . $K_1 = T_0/14$ is good value to start with, often chosen on how fast the disturbance rejection is required [9].

$$T_{yr} = F_1\left(\frac{e^{-\theta s}}{Ts+1}\right) = \frac{T_0 s+1}{K_0 s+1} \times \frac{e^{-\theta s}}{Ts+1} = \frac{e^{-\theta s}}{K_0 s+1} \tag{9}$$

$$\lambda = 1 - \frac{F_2(s)C(s)P(s)e^{-\theta s}}{1+C(s)\hat{P}(s)} = 1 - \left(\frac{e^{-\theta s}}{Ts+1}\right)F_2(s) = 1 - \left(\frac{e^{-\theta s}}{K_1 s+1}\right) \tag{10}$$

The authors argue that the proposed Modified SP performs better as compared to the Filtered SP proposed by Normey-Rico (2009) [1]. The latter was considered the best solution for stable plants with long dead time among the latest modifications on the conventional SP. Moreover, authors claim to present a MSP that is robust for any amount of dead time [9].

2.4 Review of *Neuro-fuzzy Smith Predictor Compensator*

The Classical Smith Predictor combines the approach of Backpropagation Neural Network Algorithm to design a Neuro-fuzzy Smith Predictor compensator that provides good robustness, reduces the sensitivity of the SP to process modelling errors and increases system performance [12, 13]. The modelling error between the actual plant and the plant model is fed to the fuzzy logic compensator. Each parameter namely K, τ, and θ are compensated individually by three different Neuro-fuzzy compensators, for instance, if the time delay θ, process exceeds the $\hat{\theta}$ model, the compensator increases the time delay of the model [12]. Figure 6 shows the Neuro-Fuzzy Compensator combined with Smith Predictor:

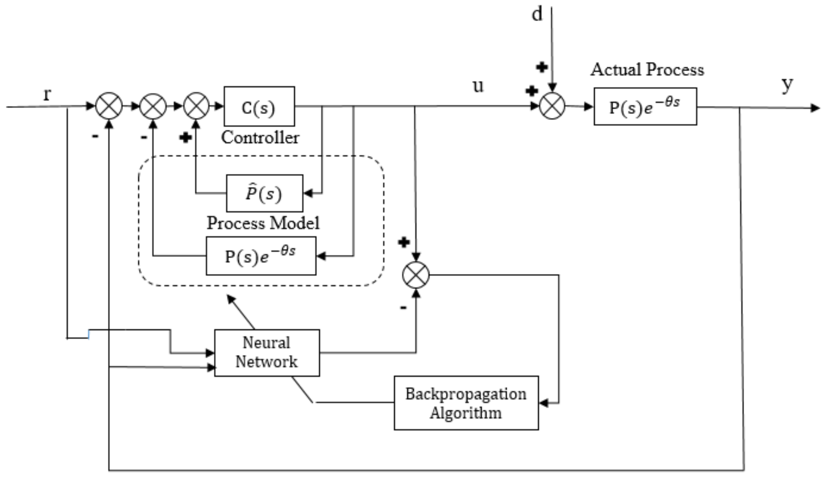

Figure 6: Neuro-Fuzzy Smith Predictor Compensator

Under simultaneous variations of the process parameters such as dead time, process gain and time constant this technique performs reliably and provides good robust stability. However, the downside is that when the training algorithm is online, the tuning approach is very complicated and it takes much time [12, 13].

2.5 Review of *Filtered Smith Predictor(FSP)*

The concept of FSP is based on two filters. $F(z)$ is used to improve the reference signal of any undesirable overshoot at reference whereas $F_r(z)$ is used to increase robustness or improve disturbance rejection response respectively [2].

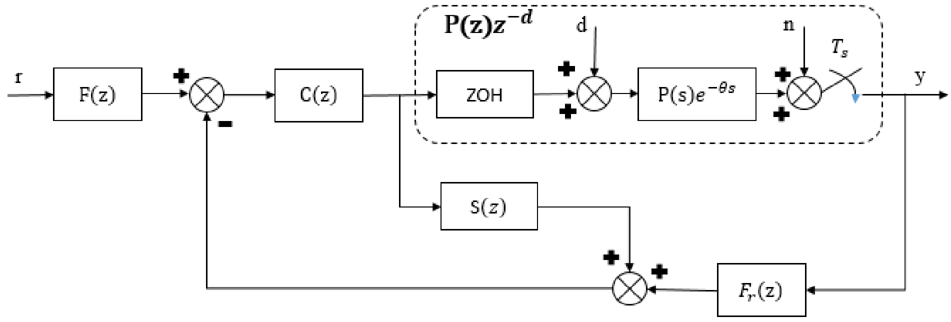

Figure 7: Discrete FSP control structure

As shown in Figure 7 above of the FSP control structure, the ZOH discretizes the process model in the internal loop. The Sampling period is chosen based on a rule of thumb proposed by [14] which is highly dependent on dead-time estimation error and its choice may affect performance and robust stability of a control system. Table 1 below summarizes the relationship between sampling time and dead-time estimation error:

Table 1: Sampling time choice guide [14]

$\delta\theta_{max}$	Maximum sampling time
0-17%	$T_s = \delta\theta_{max}$
17-45%	$T_s = \delta\theta_{max} / 2$
45-82%	$T_s = \delta\theta_{max} / 3$
82%-100%	$T_s = \delta\theta_{max} / 4$

Equations (11) to (13) below show the discrete model, the controller and predictor filter:

$$P_n(z) = K_p \frac{(1-a)}{z-a} z^{-d} \text{, where: } d = \theta / T_s \text{ and } a = e^{T_s/\tau} \tag{11}$$

$$C(z) = \left(\frac{1-\lambda}{Kp(1-a)} \right) \frac{z-a}{z-1} \tag{12}$$

$$F_r(z) = \frac{(1-\beta)z}{z-\beta} \tag{13}$$

The predictor filter $F(z)$ is tuned to stabilize $S(z)$ which consist of the linear and nonlinear parts of the model and at the same time to satisfy the robustness condition. A measure of robustness is evaluated when the transfer function satisfies [2, 15]:

$$\left| \frac{C(e^{jwT_s})P(e^{jwT_s})F_r(e^{jwT_s})}{1 + C(e^{jwT_s})P(e^{jwT_s})} \right| < \frac{1}{N\left|e^{jwT_s} - 1\right|} \quad 0 < \omega < \frac{\pi}{T_s} \text{, where: } N = d \tag{14}$$

The discrete FSP is easy to tune, can be used in a unified manner to control stable, integrating and unstable dead-time processes which takes into account both performance and robust stability. In addition, it has vast implementation of control systems applications in digital computer [2].

CHAPTER 3
METHODOLOGY

3.1 Introduction

To achieve the objectives of this project a temperature process control loop model is obtained from a heat exchanger pilot plant experiment. Firstly, a Classical Smith Predictor-PI based controller and a PI controller are developed using Matlab/Simulink to simulate the control of the closed loop temperature process. An analysis on performance and robust stability is done.

Subsequently, a study is done to improve the control performance of the temperature process by employing the Filtered Smith Predictor technique due to its practicability to develop a controller that is easy to tune and accounts for robust stability. Figure 8 below shows the various steps taken to study this project.

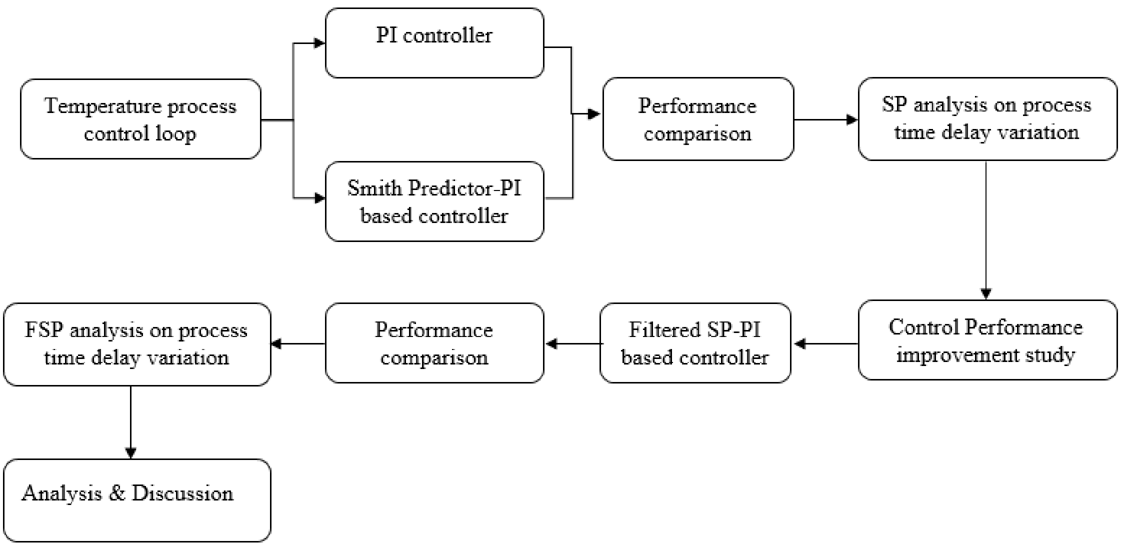

Figure 8: Methodology

3.2 Experimental Modelling

Figure 9 below shows a Drum-Heat Exchanger Process Pilot Plant experiment used to model the parameters of a simple heat exchanger temperature control loop. The unit simulates real shell-tube heat exchanger processes similar to Industrial Plants [16].

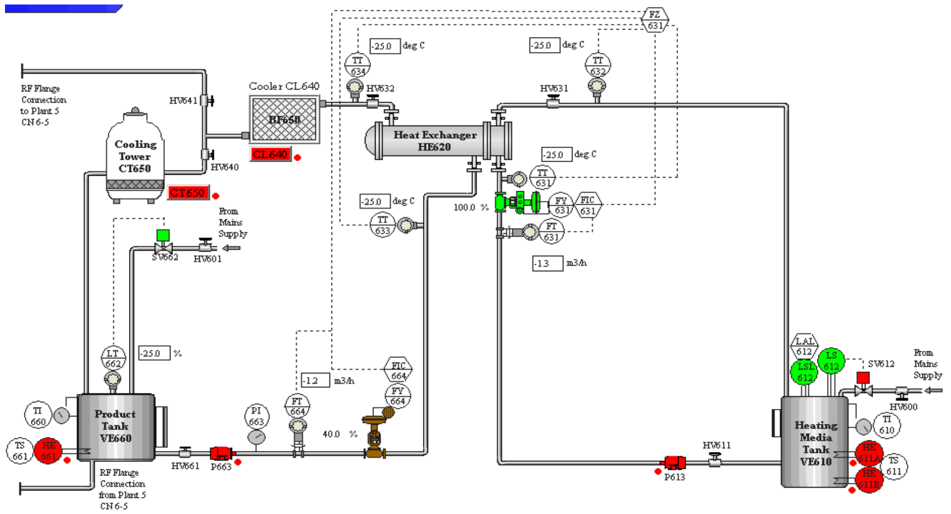

Figure 9: Drum-Heat Exchanger Process Pilot Plant

3.2.1 Description of the Process Pilot Plant

The heating media tank (VE610) hot water heats cold water, product stream from the product tank (VE660) through a heat exchanger (HE-620). Then the cooler (CL-640) and the cooling tower (CT-650) cool the product stream and it returns to the product tank (VE660). Temperature control is important in industrial processes because it transfers heat energy from a stream of higher energy to a stream of lower energy and it dictates the product quality, hence the need of control and monitoring [16].

3.2.2 Temperature control experiment

The experiment was carried out in 3 different runs which consisted in introducing a step change of 10% (40-50%) to the water temperature at an initial value of 42^{o} C as to obtain three different process models with varying model parameters.

The data was gathered from the Pilot Plant server. Statistical model identification with linear regression is used to minimize the sum of square errors as to get the best estimation of the model parameters [8]. Then each model is evaluated with its corresponding data and a cross-validation is done to determine the best model for the temperature control loop.

Finally, the model is used to develop PI, CSP and FSP controllers. Robust stability and performance of the controllers are analyzed based on performance measures for both fixed time delay and when the process time delay varies.

3.3 Performance Measures

The Integral Absolute Error (IAE), Integral Squared Error (ISE), and settling time measure the performance from the beginning of the process until it reaches the band of 2% of its steady-state value. It is considered as such because noise, measurement errors are usually in this range [17]. The other performance measures are the overshoot and rise time. The rise time is considered as the time the response takes to rise from 10% to 90% of its steady-state value. Table 2 below shows ISE and IAE formulas for measure of performance in the closed loop temperature control process.

Table 2: IAE and ISE Performance Measures

Performance Indexes
$\mathrm{IAE} = \int_{t_0}^{t_f} \lvert sp(t) - y(t) \rvert \, dt$, and $\mathrm{ISE} = \int_{t_0}^{t_f} [sp(t) - y(t)]^2 \, dt$ where:

3.4 Gantt Chart and Key Milestones

A Gantt chart of activities is provided below in Figure 10 and 11. The project is divided into two stages, Final Year Project 1 and 2 to be completed into two semesters respectively. Both Figures show the project progress and the expected deliverables of the project i.e., Key Milestones.

FINAL YEAR PROJECT 1															
No.	Activity	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	Selection of Topic														
2	Literature Review of SP and MSP techniques														
3	Experiment on a Heat Exchanger Pilot Plant														
4	Temperature Process Model Identification										★				
5	Model Verification and Validation														
6	Simulation of CSP on Matlab/Simulink														
7	SP-PI controller design													★	
8	Submission of Interim Report														★

Progress Deliverables

Figure 10: FYP1 Gantt Chart

FINAL YEAR PROJECT 2															
No.	Activity	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	SP-PI controller robustness analysis														
2	FSP-PI controller design														
3	FSP-PI controller robust analysis								★						
4	Analysis due to process time delay varition											★			
5	Results and Discussion														
6	Submission of Draft Final Report													★	
7	Submission of Final Report														★

Progress Deliverables

Figure 11: FYP2 Gantt Chart

CHAPTER 4
RESULTS AND DISCUSSION

4.1 Process Modelling

Based on the Statistical Model Identification, 3 models have been obtained as follows,

$$P_1(s) = \frac{0.183e^{-0.27s}}{2.76s+1} \qquad P_2(s) = \frac{0.192e^{-0.33s}}{2.65s+1} \qquad P_3(s) = \frac{0.183e^{-0.30s}}{2.76s+1}$$

The models are verified for each set of data used to estimate its parameters as shown in Figure 12.

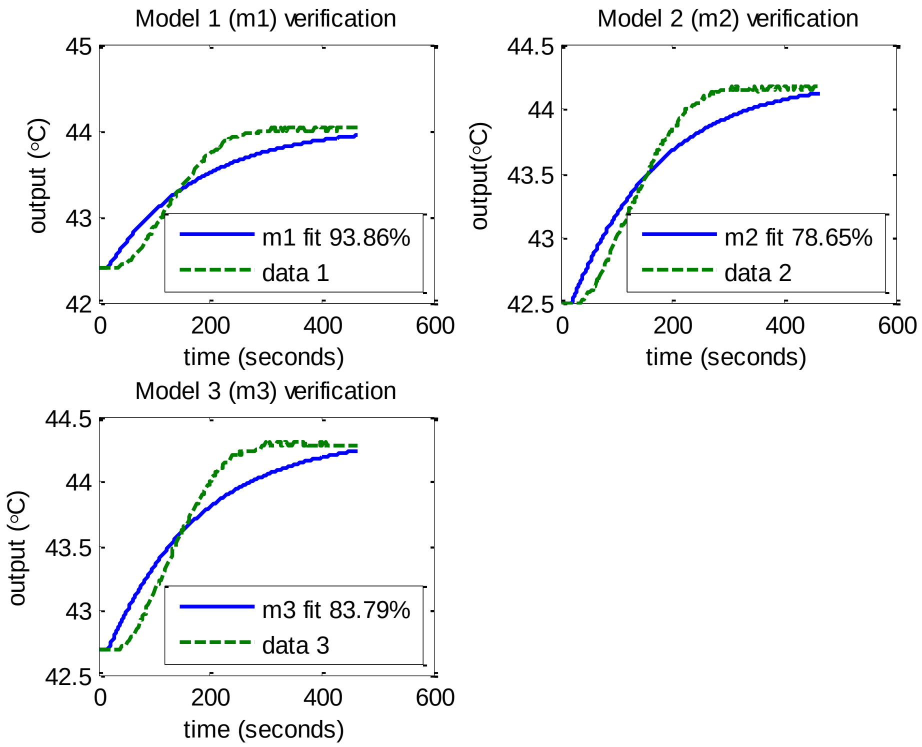

Figure 12: Model verification for each set of data

The models in Figure 12 show how accurately they represent each process data from the temperature control loop experiment. Model 1 is chosen to represent the process data since it fits the data with a 93.86% accuracy as compare to the others. Next in Figure 13, Model 1 is validated with other set of data (from process data 2 and 3). The result shows that Model 1 fits as the model for the temperature control loop. Based on the transfer functions, a maximum dead-time error estimated is $\delta\theta = 22\%$.

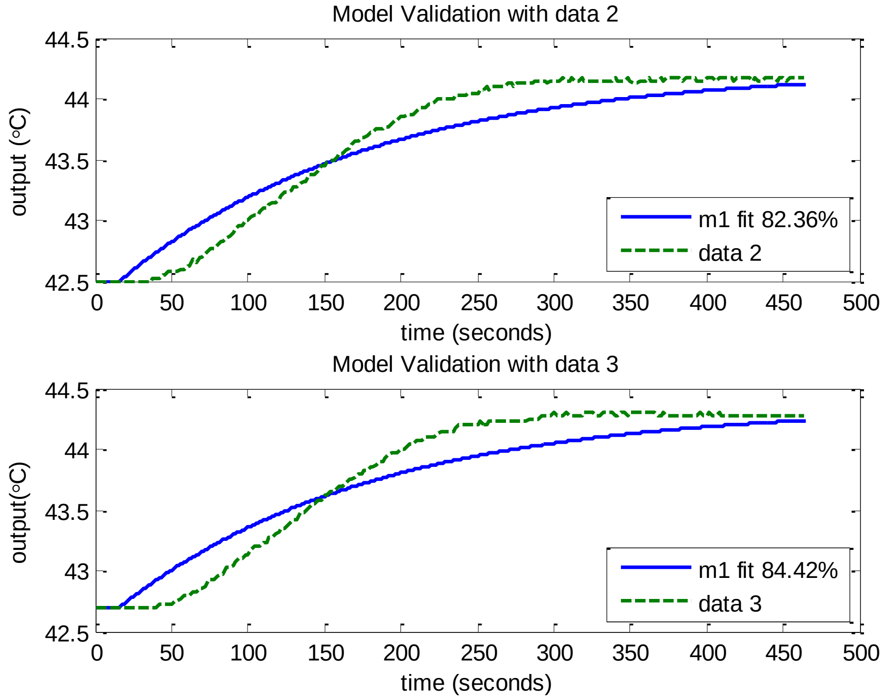

Figure 13: Cross validation of model 1 with data 2 and 3

4.2 Smith Predictor and PI controller tuning

The Smith Predictor PI controller is tuned with 2DOF without filter to attain good robust characteristics. On the other hand, shown in Table 3 the PI controller is tuned using the Simple Internal Method Control (S-IMC). This is because for processes in which $\tau > 8 \times \theta$, it offers good robustness and smooth response as in this case [14].

Table 3: Tuning Method for Classical SP and PI controller

Tuning Method	Controller	Kc	T_i	Filter
S-IMC for PI	$Kc\left(\frac{Tis+1}{Tis}\right)$	$\frac{1}{2\theta}\frac{1}{Kp/\tau}$	$\min[Ti, 8\theta]$	———
2DOF for CSP	$Kc\left(\frac{Tis+1}{Tis}\right)$	$\frac{\tau}{KpT_0}$	τ	$\frac{(1+T_0 s)}{(1+T_1 s)}$, $T_1 \in [\tau, T_0]$

The coefficient T_0 is chosen based on a measure of robustness, dP(jω). Table 4 and Figure 14 show when the Classical SP-PI controller is tuned for different values of T_0 and dP(jω) intersects twice the value modelling error δP(jω), a good trade-off between performance vs robustness is achieved. Therefore, T_0 = 0.374 is chosen to tune Classical SP.

Table 4: Trade-off between robust and performance stability [14]

Measure of Robustness versus performance
$dP(j\omega) > 2 \times \delta P$, where: $dP(j\omega) = \mid 1 + j\omega_n (To / \delta\theta_{max}) \mid$ and $\delta P = \mid 1 - e^{-j\omega_n} \mid$

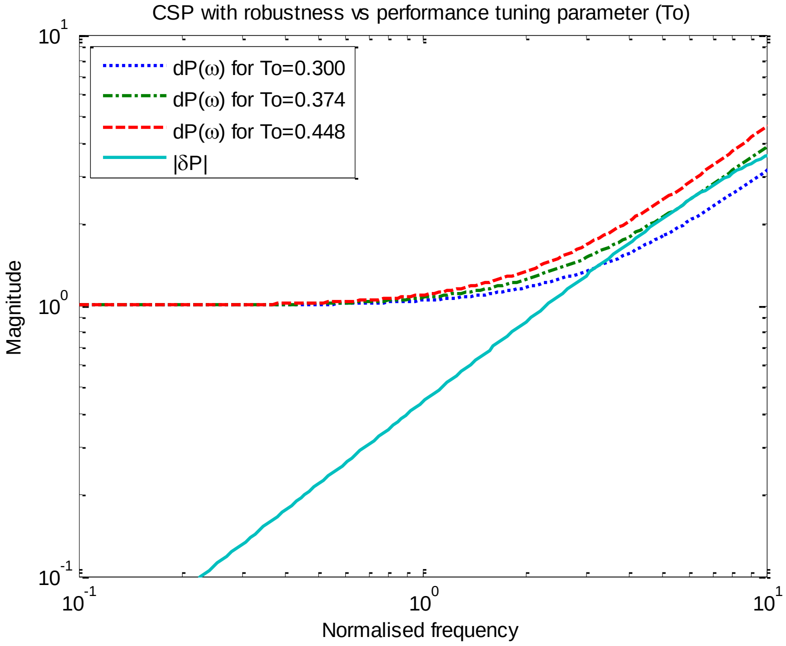

Figure 14: Tuning T_0 for Robustness and Performance stability trade-off

4.2.1 Performance vs Robustness Analysis

Figure 15 shows the control performance test of a PI controller and another with the SP-PI at nominal delay $\theta = 0.27$ minutes. The SP-PI shows a fast response with no overshoot whereas with the PI controller, the process overshoots before it achieves zero steady-state. Figure 16 further shows the control performance test when process time delay is at $\theta = 0.33$minutes i.e., at upper bound process time delay limit due to uncertainty (dead-time estimation error, 22%). The PI control performance shows increase in overshoot whereas the SP-PI shows a robust control that maintains a fast response and minimal overshoot. Hence the Smith Predictor is an effective technique in reducing the undesired effects of process time delay in the closed loop temperature control process.

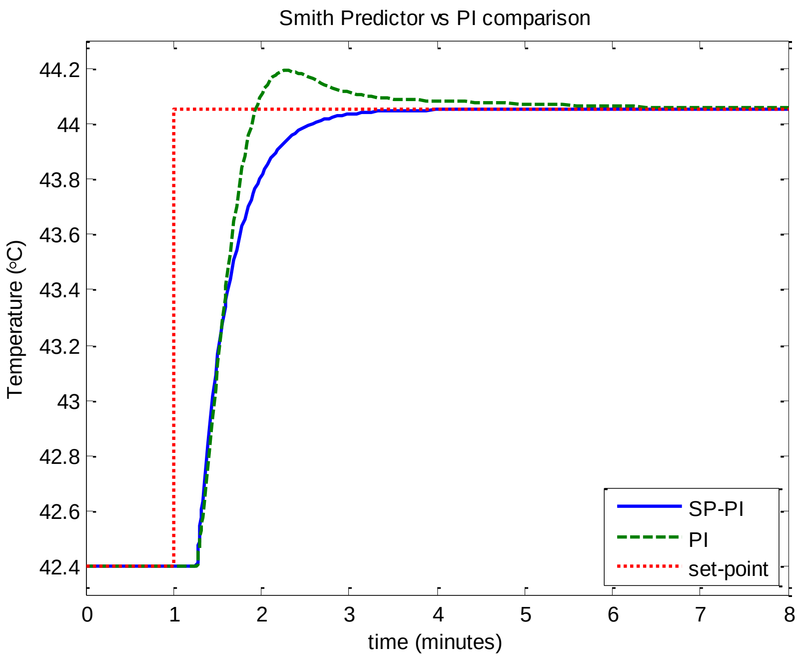

Figure 15: Comparison between Classical SP and PI controller, $\theta = 0.27$ minutes

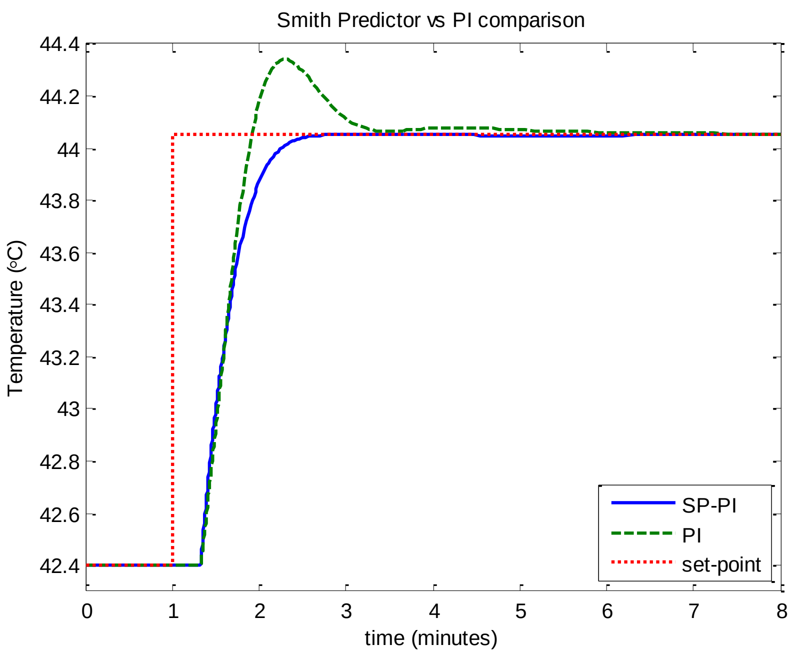

Figure 16: Comparison between Classical SP and PI controller $\theta = 0.33$ minutes

The comparison between the Classical SP and PI control performance is further discussed in Table 5, where IAE and ISE performance measures prove that the Smith

Predictor outperforms the PI controller since lower error values between the output and reference signals are obtained.

Table 5: IAE and ISE Performance indexes of SP-PI vs PI controller

	$\theta = 0.27$ (nominal delay)		$\theta = 0.33$	
Controller	IAE	ISE	IAE	ISE
SP-PI	1.0692	1.2500	1.0714	1.3818
PI	1.1180	1.2440	1.2801	1.4331

The Classical SP-PI control performance of variations of the temperature control process models P_1, P_2, P_3 obtained in Section 4.1 are shown in Figure 17. The process models represent how the actual process P varies in the set of family of transfer functions centered around $P = P(1 + \delta P_n)$ [14]. P_3 has a faster rise time and settling time as compared to P_1 and P_2. However, IAE and ISE performance indexes shown in Table 6 similar error values between the models.

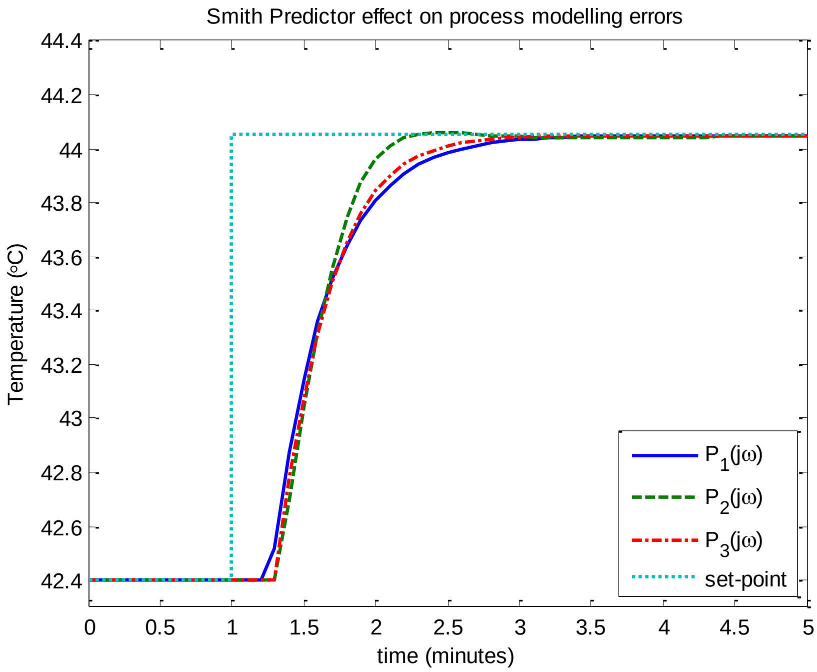

Figure 17: CSP performance on variation of process models P_1, P_2, P_3

Table 6: SP-PI performance as actual process varies between P1, P2, P3.

Performance Index	$\delta P_1(j\omega)$	$\delta P_2(j\omega)$	$\delta P_3(j\omega)$
IAE	1.0724	1.0165	1.0725
ISE	1.2538	1.3300	1.3253

Analysis on the effect Classical SP control performance when the process time delay varies is discussed to study the limits of this controller in reducing the effects of dead time. The process delay variation is considered for 3 cases namely the lower bound value $\theta = 0.21$ minutes, the nominal delay $\theta_n = 0.27$ minutes and the upper bound value $\theta = 0.33$ minutes. This assumption is based on the nominal delay θ_n and the dead time estimation error $\partial\theta$ i.e., $\theta = \theta_n(1 \pm \partial\theta)$.

Figure 18 shows that as the process time delay increases, rise time and settling time reduce and overshoot increases. Table 7 further shows IAE and ISE values increase as process delay increases. Nonetheless, the Classical Smith Predictor shows robust control with the variation of process time delay.

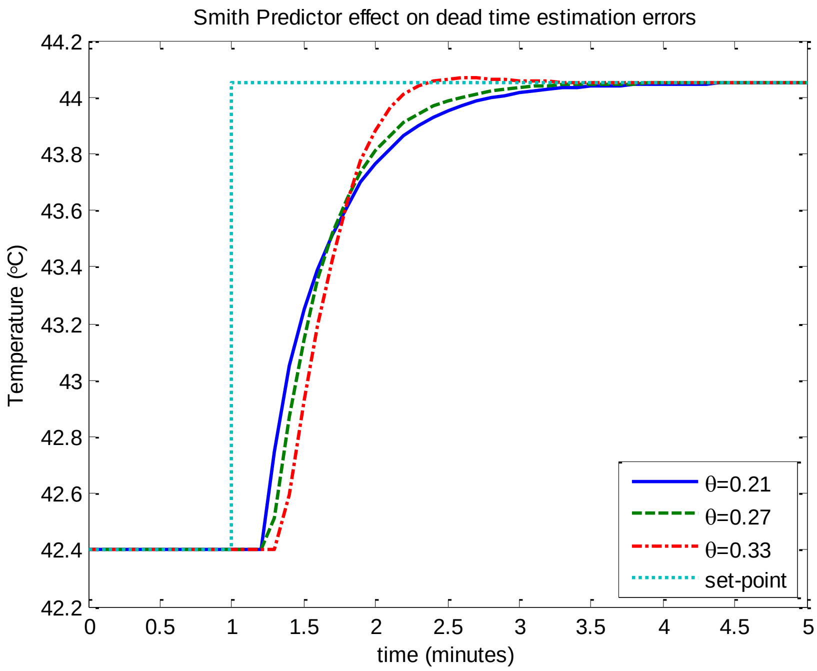

Figure 18: CSP closed loop response effect on dead time estimation errors

Table 7: Smith Predictor Performance as process time delay varies

Performance Index	Θ = 0.21	Θ = 0.27	Θ = 0. 33
IAE	1.0722	1.0724	1.0747
ISE	1.1424	1.2538	1.3854
Rise time (min.)	1.0155	0.8316	0.6155
Settling time (min)	3.0244	2.7529	2.2227
Overshoot (%)	0	0	0.0406

The control performance of this technique can be further improved with the use of a filter for the output signal to be fed back to the system thus reducing the effects of dead time even more. Therefore, an analysis on Filtered Smith Predictor to improve the performance of the Classical SP and account for robust stability when process delay varies will be discussed in the next section.

4.3 Filtered Smith Predictor (FSP)

The Filtered Smith Predictor as reviewed in Section 2.5 uses two filters that can be used to further improve the performance of the temperature control. In this study, the reference filter is kept at unity as for most stable plants [14] as no considerable amount of undesirable overshoot is observed. The predictor filter ensures that the error fed back to the controller provides the system with robust control.

The discrete model, the controller and the predictor filter of the FSP are shown below,

$$P_n(z) = \frac{0.0072}{z - 0.9605} z^{-2} \qquad C(z) = \left(\frac{1-\lambda}{0.0072}\right)\frac{z - 0.9605}{z - 1} \qquad F(z) = \frac{(1-\beta)z}{z-\beta}$$

Based on the Sampling time choice guide in Table 1, Section 2.5, the sampling time for the temperature control loop is chosen as $T_s = 0.1$ minutes. Tuning of the FSP is done for λ and β to adjust the controller gain and to satisfy the robustness condition respectively. This subsequently provide tight control for the process.

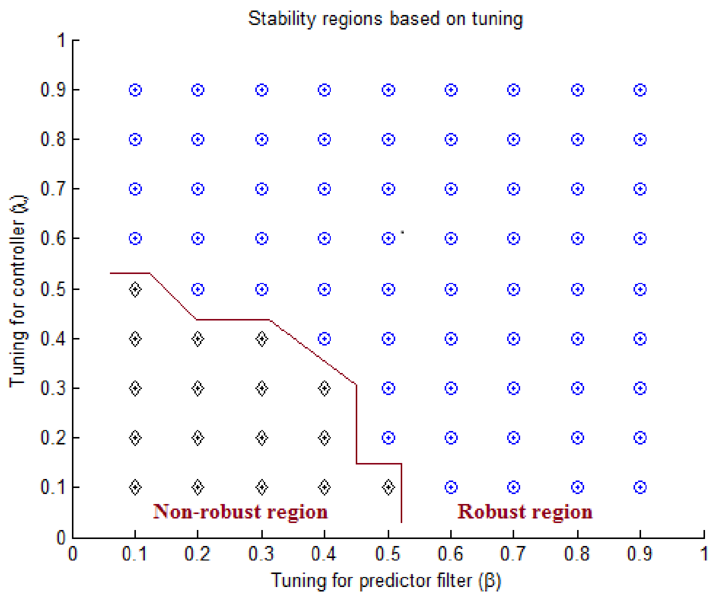

Figure 19: Filtered Smith Predictor tuning using λ and β parameters.

In Figure 19 above, as the controller tuning parameter λ increases, performance indicator such as overshoot decrease and rise time and settling time reduce. Meanwhile as β increases, overshoot decreases but rise time and settling time remain unchanged.

The robust stable region is determined for the case when the magnitude of the small gain theorem is less than unity at all frequencies whereas the non-robust region when its magnitude is more than unity for one or more frequencies. As λ and β increase from (0.1, 0.1) to (0.3, 0.4) the size of frequencies crossing the unity magnitude tend to shrink before it becomes robust stable.

Based on robust tuning in Figure 19 above tuning parameters λ =0.7 and β =0.1 are chosen from the robust stable region to study the FSP as to improve the performance of the CSP. Figure 20 below shows the comparison of the Filtered Smith Predictor to the Classical Smith Predictor. Performance indicators shown in Table 8 below present evidence that the FSP outperforms the SP-PI based controller and improves the performance of the closed loop temperature control process.

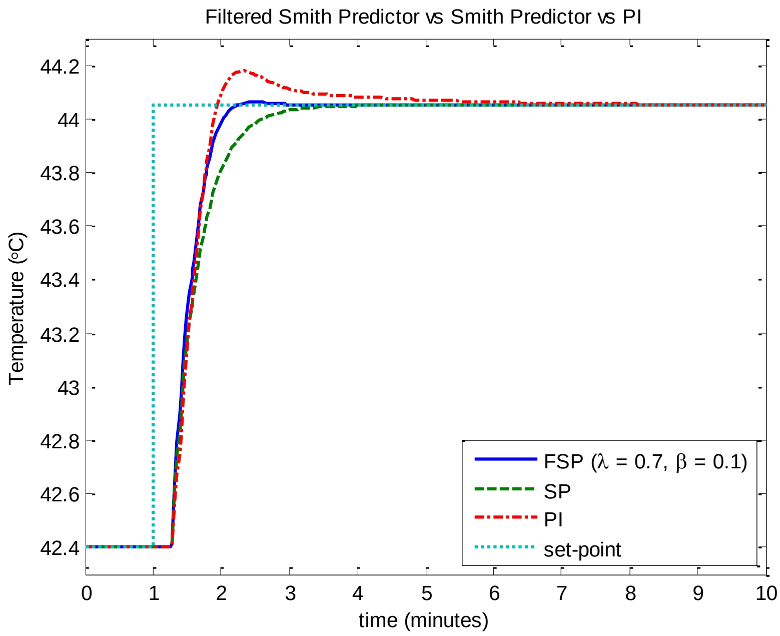

Figure 20: Performance comparison between FSP, SP and PI at $\theta = 0.27$ minutes

Table 8: Performance measures between Filtered and Classical Smith Predictor

Predictor	IAE	ISE	OV (%)	Rise time (min)	Settling time(min)
FSP	0.9023	1.1518	0.0289	0.5547	2.0822
CSP	1.0762	1.2538	0	0.8399	2.7801
PI	1.1208	1.2440	0.2870	0.4947	3.8521

The performance results in Table 8 above indicate that the added predictor filter on the Smith Predictor attenuates possible oscillations in the plant output where uncertainty errors occur [14]. Therefore, the control structure can compensate dead-time and improve the control performance of the temperature control loop.

It is also important noting the effect the controller tuning parameter λ plays on the performance of the closed loop system. The smaller the value, a poor control loop performance is achieved and a bigger value, for instance like $\lambda = 0.9$ a rather sluggish performance is attained. A good value would be in the range of 50-80 %. Figure 21 above shows a case where λ is varied and β is kept constant.

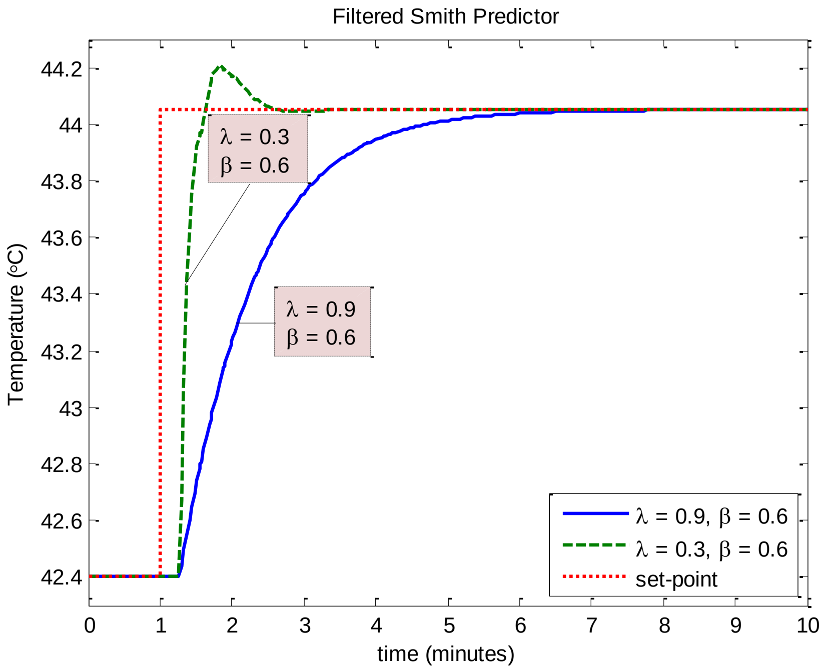

Figure 21 Effect of λ on the performance of the temperature control loop

4.3.1 Performance analysis on varying process delay

This section will analyze the limits of the FSP controller in reducing the effects caused by process delay variation within the temperature control loop. The analysis is defined through a bounded interval based on the process nominal delay and the dead-time estimation error.

The nominal delay is $\theta_{\text{n}} = 0.27$ minutes and the dead time estimation error $\delta\theta = 22\%$. Therefore, the estimated range considers the upper and lower bound interval resulting in a range [0.21, 0.33] minutes.

To appreciate this control structure, two (2) FSP control structures will be compared and analyzed using the previous robust FSP tuning parameters ($\lambda = 0.7$ and $\beta = 0.1$)and with non-robust FSP tuning parameters ($\lambda = 0.1$ and $\beta = 0.1$). The analysis will be based on IAE and overshoot performance indicators

Table 9 below shows the effect of varying the process time delay on the performance of the temperature control loop process. Small increments in steps of 0.02 minutes were chosen in the estimated range from 0.21 to 0.33 minutes (shown shaded in Table 9) to study the response of the system. The interval is further extended to 0.13 minutes past the lower limit value to observe where the non-robust FSP goes into instability.

The ISE and overshoot increase as the delay is incremented in the given range for both cases. The ISE is lower in most instants for the non-robust FSP case than the robust due to the smaller β value chosen. Nonetheless the non-robust FSP control loop exhibits poor performance characterized by oscillations which is undesirable. Meanwhile, the robust FSP shows robust control with good performance indicator.

Table 9: Varying delay performance comparison for robust and non -robust regions

Time (minutes)	Robust FSP			Non-robust FSP		
	System	ISE	OV (%)	System	ISE	OV (%)
0.13	stable	0.8592	0.00025	**unstable**	2.0336	--
....	stable	**....**	**....**	oscillatory	**....**	**....**
0.21	stable	1.0127	0.00063	oscillatory	0.6842	0.0059
0.23	stable	1.0559	0.00076	oscillatory	0.7391	0.1728
0.25	stable	1.1029	0.0017	oscillatory	0.8012	0.6203
0.27	stable	1.1513	0.0285	oscillatory	0.8748	1.0712
0.29	stable	1.2027	0.0898	oscillatory	0.9717	1.5172
0.31	stable	1.2548	0.1721	**unstable**	1.4907	1.0960
0.33	stable	1.3115	0.2764	**unstable**	338.0309	--

To illustrate the analysis done in Table 9, Figure 22 and 23 show the effect of varying the process time delay at nominal delay $\theta = 0.27$ minutes and when the process changes to $\theta = 0.32$ minutes respectively. The effects are summarized in the Table 9.

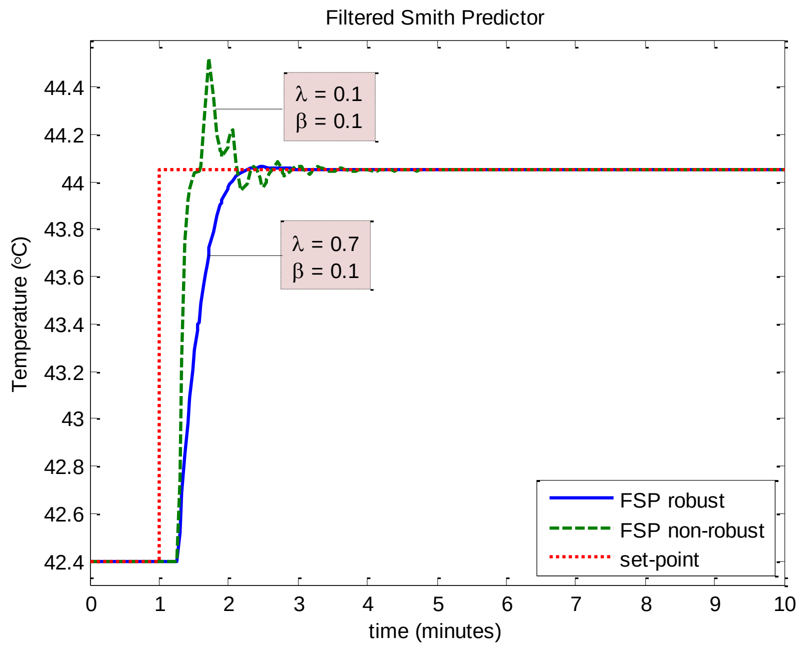

Figure 22: Robust vs non-robust FSP at nominal delay $\theta = 0.27$ minutes

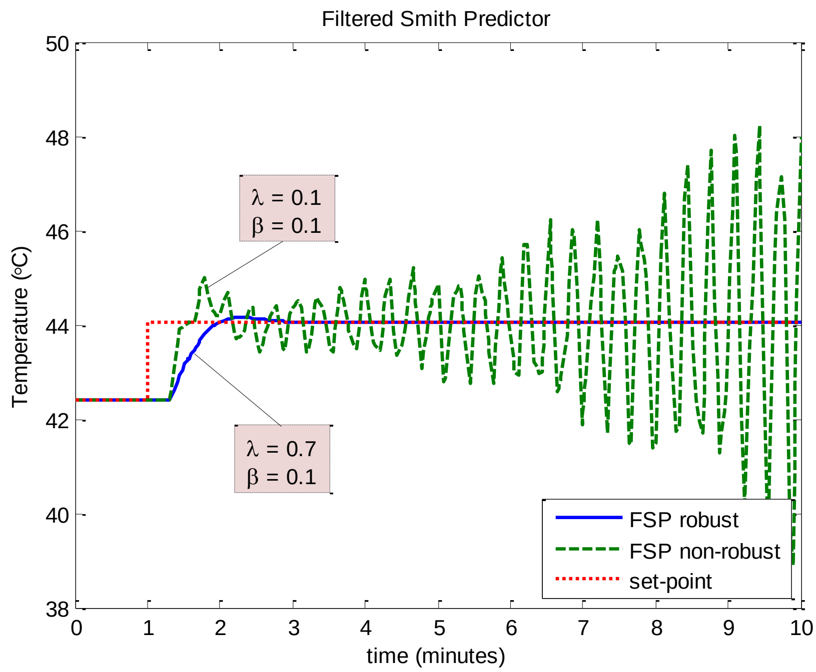

Figure 23: Robust vs non-robust FSP at process delay $\theta = 0.32$ minutes

CHAPTER 5

CONCLUSION AND RECOMENDATIONS

This project studies the limitations of the Smith Predictor structure to control a closed loop temperature process when process time delay varies. To model the process, a simple heat exchanger temperature control experiment was conducted on a process pilot plant. Three runs were carried out and the model was identified using Statistical Modelling with linear regression. Model verification and validation was done and process model P_1 was chosen because it fits above 80% of all data.

The model is used to develop a PI and Classical Smith Predictor-PI based controllers to prove the effectiveness of the Classical SP-PI over the PI controller and study the limits of the Classical SP when process time delay varies. Results show that the Classical SP provides robust control and presents good performance indicators (ISE and IAE) in the estimated interval from 0.21 minutes to 0.33 minutes.

Further improvement on the performance of the closed loop temperature process is achieved with the use of a predictor filter to attenuate possible oscillations at the output signal where uncertainty error occurs. Hence, a Filtered Smith Predictor-PI controller is developed to improve the performance of the temperature control loop. Tuning is done for the controller parameter λ, and predictor filter parameter β, and a robust and a non-robust stability regions are obtained.

Tuning parameters $\lambda = 0.7$ and $\beta = 0.1$ for a robust FSP are chosen based on their expected improved characteristics such as fast rise time, settling time and minimal overshoot. Thus, improved performance of the temperature process is achieved with the FSP. Analysis on the process time delay variation in the estimated bounded interval from 0.21 minutes to 0.33 minutes, shows that the robust FSP withstands the variation of time delay and good performance indicators such as lower ISE and minimal overshoot are attained. Meanwhile a non-robust FSP with $\lambda = 0.1$ and $\beta = 0.1$ as the process delay increases, ISE values and overshoot increase until the process goes into instability.

Future work for the improvement of this study can be done to investigate the ability of the Filtered Smith Predictor to control disturbance rejection speed subjected to the temperature process. The often types of disturbances for the temperature process are the temperature variation of the input fluid and the flow variation of input fluid, the latter being more prominent in practice [18]. In the FSP control structure, robust stability or disturbance rejection can be improved by selecting the appropriate predictor filter parameter value β from the interval $0 < \beta < 1$.

REFERENCES

[1] J. E. Normey-Rico and E. F. Camacho, "Unified approach for robust dead-time compensator design," *Journal of Process Control,* vol. 19, pp. 38-47, 1// 2009.

[2] J. E. Normey-Rico, P. Garcia, and A. Gonzalez, "Robust stability analysis of filtered Smith predictor for time-varying delay processes," *Journal of Process Control,* vol. 22, pp. 1975-1984, 12// 2012.

[3] C. Rodríguez, J. E. Normey-Rico, J. L. Guzmán, and M. Berenguel, "On the filtered Smith predictor with feedforward compensation," *Journal of Process Control,* vol. 41, pp. 35-46, 5// 2016.

[4] P. R. C. Mendes, J. E. Normey-Rico, V. João Jr, and D. M. Cruz, "A filtered Smith predictor based subspace predictive controller," *IFAC Proceedings Volumes,* vol. 47, pp. 1011-1016, // 2014.

[5] V. J. Poorani and L. D. V. Anand, "Comparison of PID controller and Smith predictor controller for heat exchanger," in *Emerging Trends in Computing, Communication and Nanotechnology (ICE-CCN), 2013 International Conference on*, 2013, pp. 217-221.

[6] A. Mehrafrooz, M. R. Sorkhkolaei, and A. Yazdizadeh, "Simultaneous application of adaptive PID controller and smith dead-time predictor rule in nonlinear water level control in Neka power plant," in *2011 6th IEEE Conference on Industrial Electronics and Applications*, 2011, pp. 1342-1347.

[7] H. L. Wade, "Basic and Advanced Regulatory Control - System Design and Application (2nd Edition)," ed: ISA, 2007.

[8] T. E. Marlin, *Process control : designing processes and control systems for dynamic performance*. Boston [etc.]: McGraw-Hill, 2000.

[9] K. Kirtania and M. A. A. S. Choudhury, "A two-degree-of-freedom dead time compensator for stable processes with dead time," in *Advanced Control of Industrial Processes (ADCONIP), 2011 International Symposium on*, 2011, pp. 385-390.

[10] P. Puawade, V. Boonpranchoo, V. Tipsuwannaporn, and A. Numsomran, "Smith predictor design by CDM for temperature control system," in *Control*

Automation and Systems (ICCAS), 2010 International Conference on, 2010, pp. 1522-1526.

[11] Q.-C. Zhong, *Robust Control of Time-delay Systems*, 1 ed.: Springer-Verlag London, 2006.

[12] H. Chen, Z. Zouaoui, and Z. Chen, "A neuro-fuzzy compensator based Smith predictive control for FOPLDT process," in *2011 IEEE International Conference on Mechatronics and Automation*, 2011, pp. 1833-1838.

[13] H. Chen, Z. Zouaoui, and Z. Chen, "Neuro-Fuzzy Modified Smith predictor for IPDT and FOPDT Processes Control," *IFAC Proceedings Volumes,* vol. 46, pp. 839-844, 2013/01/01 2013.

[14] J. E. Normey-Rico, "Control of dead-time processes," ed: Springer Science & Business Media, 2007.

[15] C.-Y. Kao and B. Lincoln, "Simple stability criteria for systems with time-varying delays," *Automatica,* vol. 40, pp. 1429-1434, 8// 2004.

[16] U. T. Petronas, "Exp. 4: Simple Heat Exchanger Temperature (TIC-634)," E. Department, Ed., ed.

[17] V. Feliu-Batlle, R. Rivas-Perez, and F. J. Castillo-Garcia, "Robust fractional-order temperature control of a steel slab reheating furnace with large time delay uncertainty," in *Fractional Differentiation and Its Applications (ICFDA), 2014 International Conference on*, 2014, pp. 1-6.

[18] S. Padhee, Y. B. Khare, and Y. Singh, "Internal model based PID control of shell and tube heat exchanger system," in *Students' Technology Symposium (TechSym), 2011 IEEE,* 2011, pp. 297-302.

APPENDICES

TemperatureVol_2

Signal_Inde	Earliest_TDa	Sample_Val	Sample_Val	Sample_Val	Sample_Val	Sample_Val	Sample_Val	Sample_Val	Sample_Val	Sample_Val	Sample_Val
16	16 11:54:59 AM	42.400001526	42.400001526	42.400001526	42.400001526	42.400001526	42.400001526	42.400001526	42.400001526	42.400001526	42.400001
16	16 11:56:11 AM	42.674999237	42.725002289	42.725002289	42.75	42.774997711	42.799995422	42.799995422	42.849998474	42.849998474	42.874996
16	16 11:57:23 AM	43.374996185	43.400001526	43.400001526	43.424999237	43.449996948	43.475002289	43.475002289	43.5	43.5	43.550003
16	16 11:58:35 AM	43.875003815	43.849998474	43.875003815	43.875003815	43.900001526	43.900001526	43.900001526	43.924999237	43.924999237	43.924999
16	16 11:59:47 AM	44	44	44	44	44	44	44.024997711	44	44.024997711	44.024997
16	16 12:00:59 PM	44	44	44	44	44	44	44	44	44.024997711	44.024997
16	16 12:02:11 PM	44.024997711	44.024997711	44.024997711	44.024997711	44.024997711	44.024997711	44.024997711	44.024997711	44.024997711	44.024997
16	16 12:03:23 PM	44.024997711	44.024997711	44.024997711	44.050003052	44.024997711	44.050003052	44.024997711	44.050003052	44.050003052	44.024997

Figure 24: Original data of PV for Run 1 as collected in experiment

Least Square Method for determining model parameters a and b

1. Determine model parameters a and b:

$$U = \begin{bmatrix} Y_i' & X_{(i+1)-\Gamma}' \\ \cdots & \cdots \\ Y_{n-1}' & X_{n-\Gamma-1}' \end{bmatrix} \qquad z = \begin{bmatrix} Y_{i+1}' \\ \cdots \\ Y_n' \end{bmatrix} \qquad \begin{bmatrix} a \\ b \end{bmatrix} = (U^T U)^{-1} U^T z$$

2. Minimization of sum of square errors:

$$\sum_{i=\Gamma+1}^{n} E_1^2 = \sum_{i=\Gamma+1}^{n} [(Y_{i+1}')_m - (a(Y_i')_m + b(Y_{i-\Gamma}')_m)]^2$$

Where: i is the start of the applied input change and m is measured value.

Figure 25: Determining model parameters a and b [8].

Run 1 (gamma = 8)						
#Sample, i	Input, X (MV)	Output, Y (PV)	Ouput, Y'(i+1)	Ouput, Y'(i)	Input, X'(i)	SQ ERROR
1	40	42.40000153				a
2	40	42.40000153				0.988
3	40	42.40000153				b
4	40	42.40000153				0.0022
5	40	42.40000153				
6	40	42.40000153				
7	40	42.40000153				
8	40	42.40000153				
9	40	42.40000153	0	0	0	0
Continue samples 10 to 98						
99	50	43.69999695	1.325000763	1.299995422	10	0.000346157
100	50	43.72500229	1.325000763	1.325000763	10	3.72099E-05
101	50	43.72500229	1.349998474	1.325000763	10	0.000357124
102	50	43.75	1.349998474	1.349998474	10	3.36402E-05
Continue samples 103 to 234 (last)						

Figure 26: Statistical Model Identification for Run 1 with dead time, Γ = 8. Run 2 and 3 follow similar procedure

Dead time, Γ	a	b	$\sum E^2$
7	0.9884	0.0022	0.0503
8	**0.9880**	**0.0022**	**0.0498 (minimum)**
9	0.9880	0.0022	0.0504
10	0.9881	0.0022	0.0511
11	0.9876	0.0023	0.0506

Calculations
$\theta = \Gamma \times \Delta t = 8 \times 0.033\,\text{min} = 0.27\,\text{min}$
$\tau = \dfrac{-\Delta t}{\ln(a)} = \dfrac{-0.033\,\text{min}}{\ln(0.988)} = 2.76\,\text{min}$
$K_p = \dfrac{b}{(1-a)} = \dfrac{0.0022}{1-0.988} = 0.1833°C/\%$
$P_1(s) = \dfrac{0.183\,e^{-0.27s}}{2.76s+1}$

Figure 27: Dead time Γ estimation for Statistical modelling (left) and plant parameters calculation (right) for Run 1.

```
clear all
close all

%load original data from excel
d1 = xlsread('C:\Users\user pc\Documents\Temperature.xlsx', 'Take1','13..1235');
d2 = xlsread('C:\Users\user pc\Documents\Temperature.xlsx', 'Take2','aj3..aj235');
d3 = xlsread('C:\Users\user pc\Documents\Temperature.xlsx', 'Take3','13..1235');

%adjust time scale
time = 0:2:464;


%Simulation
sim('Model_Verification')

for i=1:233
    model1(i,:) = m1(2*i,:);
    model2(i,:) = m2(2*i,:);
    model3(i,:) = m3(2*i,:);
end


%Model fitting using Comparative fit index

%Verification for first run
fit1 = 100*sum(abs((model1-d1)./d1));
fit1 = sprintf('%.2f',fit1);

%Verification for second run
fit2 = 100*sum(abs((model2-d2)./d2));
fit2 = sprintf('%.2f',fit2);

%Verification for third run
fit3 = 100*sum(abs((model3-d3)./d3));
fit3 = sprintf('%.2f',fit3);

h=subplot(2,2,1);
plot(tout,m1,time,d1,'--','LineWidth',2)
set(h);
ylabel('output (\circC)');
xlabel('time (seconds)');
title ('Model 1 (m1) verification')
legend(['m1 fit ', num2str(fit1),'%'],'data 1','Location','SouthEast')
```

```
h=subplot(2,2,2);
plot(tout,m2,time,d2,'--','LineWidth',2)
set(h);
ylabel('output(\circC)');
xlabel('time (seconds)');
title ('Model 2 (m2) verification')
legend(['m2 fit ', num2str(fit2),'%'],'data 2','Location','SouthEast')

h=subplot(2,2,3);
plot(tout,m3,time,d3,'--','LineWidth',2)
set(h);
ylabel('output (\circC)');
xlabel('time (seconds)');
title ('Model 3 (m3) verification')
legend(['m3 fit ', num2str(fit3),'%'],'data 3','Location','SouthEast')
```

Figure 28: Model Verification

```
%%%%%%%%%%%%% CSP Robustness vs Performance %%%%%%%%%%%%%%%%%%%%%%

clear all
close all

% dead-time
dl=0.22;  %dead-estimation error


for i=1:3
  T0=0.3+0.074*(i-1); %tuning robust stability vs robust performance trade-off

ww=logspace(-2,2,300);

for ii=1:300
    w=ww(ii);


    dp(i,ii)=abs(1+j*dl*w*(T0/dl)); %dp should intercept error

     e(ii)=(abs(1-exp(-j*dl*w)))*2; %error should intercept dp


end


end

% parameters
h=subplot(1,1,1);
loglog(ww,dp(1,:),':',ww,dp(2,:),'-.',ww,dp(3,:),'--',ww,e,'-','LineWidth',2)
title ('CSP with robustness vs performance tuning parameter (To)')
legend('dP(\omega) for T_0=0.300','dP(\omega) for T_0=0.374','dP(\omega) for T_0=0.448','|\deltaP|','Location','NorthWest')
axis([0.1 10 .1 10])

ylabel('Magnitude');
xlabel('Normalised frequency');
set(h);
```

Figure 29: Matlab code for Classical Smith Predictor T_0 tuning

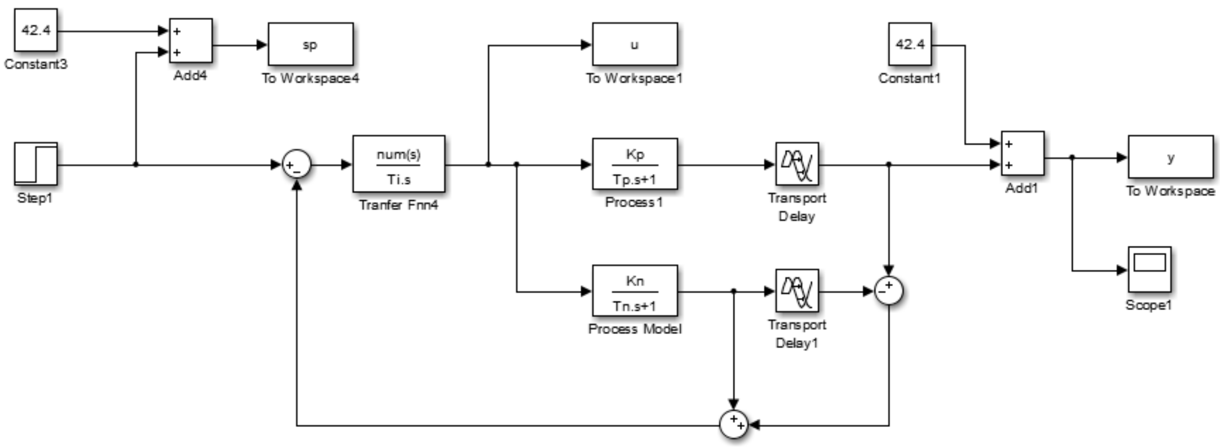

Figure 30: Matlab/Simulink model for temperature closed-loop control process based on Smith Predictor.

```
%Script file for Simulink model CSP_PI

clc
clear all
close all

% simulation
t_sim=8;                %Simulation time
tstep=1;                 %Step time

% process from take1
Kp=0.183;
Tp=2.76;  %165.57sec
Td=0.27;  %Td =0.27min =16sec
%Change the values of Td to see effect of delay (error=22%)


Td_max=0.33; %20sec (largest delay among three)

% model taken from take1
Kn=0.183;
Tn=2.76;  %165.67 sec
Tdn=0.27;  %16sec

%  PI S-IMC Tuning Method
Kcx = (1/(2*Tdn))*(1/(Kn/Tn));
Tx = [Tn 8*Tdn];
Tix = min(Tx);

% Smith predictor
% Robust tuning
dl = (Td_max-Tdn)/Tdn;  %dead-estimation error
To = 1.7*dl;
Kc=Tn/(Kn*To) ;
Ti=2.76;    %Ti = Tn

%Simulation
sim('CSP_PI_Model',t_sim)
```

```
%Calculate IAE
IAE_1 = trapz(time,(abs(sp-y)))  % between set point and Smith Predictor
IAE_2 = trapz(time,(abs(sp-y1)))  % between set point and PI controller

%Calculate ISE
ISE_1 = trapz(time,((sp-y).^2))  % between set point and Smith Predictor
ISE_2 = trapz(time,((sp-y1).^2))  % between set point and PI controller


%%%%%%%%%%%%%%%%%%%%
h = plot(time,y,time,y1,'--',time,sp,':');
set(h,'LineWidth',2);
ylabel('Temperature (\circC)');
xlabel('time (minutes)');
title ('Smith Predictor vs PI comparison')
legend('SP-PI','PI','set-point','Location','SouthEast')
axis([0 8 42.3 44.4])
```

Figure 31: Matlab code for Classical SP-PI vs PI control structures

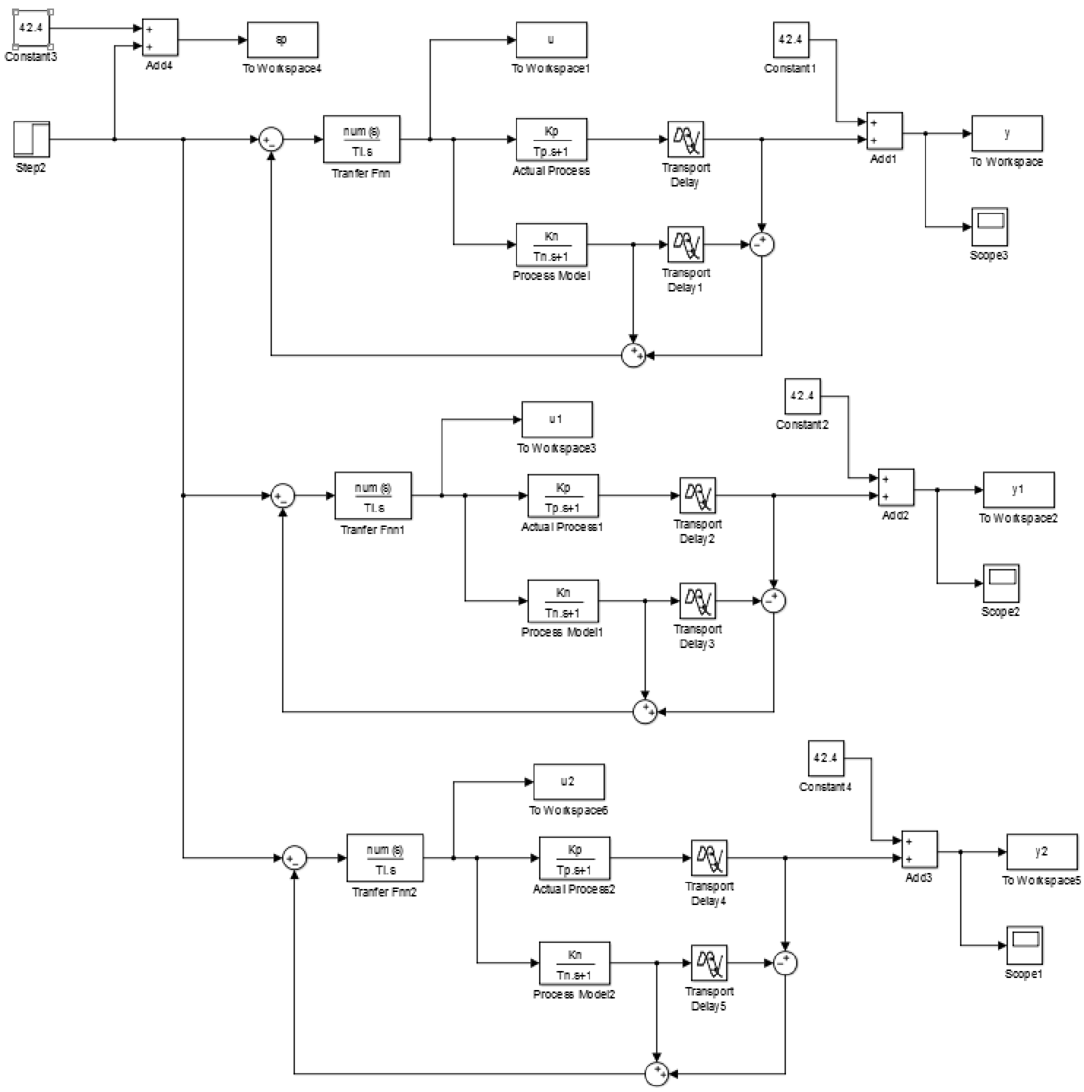

Figure 32: Process time delay variation with CSP

```
%%%%%%%%%%%%% FSP robust tuning %%%%%%%%%%%%%%%%%%%%%%

clc
clear all
close all

%%%%%%%%%%%%%%%%%%%%%%%%%%
% simulation
t_sim=50;                %Simulation time
tstep=1;                 %Step time
%%%%%%%%%%%%%%%%%%%%5%%%%

% process
Kp=0.183;
Tp=2.76;
Td=0.33  ;

%largest delay among the 3 runs
Td_max=0.33;

% model in continous time

Kn=0.183;
Tn=2.76;
Tdn=0.27;

%dead-estimation error
dl = abs((Td_max-Tdn)/Tdn);

%Sampling time
Ts = 0.5*dl;      %Chosen based on Sampling time choice Table 1

%%%%%%%%%%%%%%%%%%%%%%%%%%
% model in discrete time
a = exp(-Ts/Tn);
b =Kp*(1-a);
num = b;
%%%%%%%%%%%%%%%%%%%%5%%%%

%dead-time in discete time (Samples of Td)
d = round(Tdn/Ts) ;

%create a row vector of 300 logarithcally spaced values 0 < w < pi/Ts
ww=logspace(0,pi/Ts,300);
for i=1:9

  lambda = 0.1*i;

for ii=1:9

   beta = 0.1*ii;

for iii=1:300
    w=ww(iii);
```

```
    % common model parameters
    a = exp(-Ts/Tn);
    b = Kp*(1-a);
    z = exp(j*w*Ts);

    %Process
    numP = Kp*exp(-j*w*Td);
    denP = 1+j*w*Tp;
    P(iii) = (numP/denP);

    %discrete delay free model
    numPn = b;
    denPn = (z-a);
    Pn(iii) = (numPn/denPn);

    %controller
    numC = (1-lambda)*(z-a);
    denC = b*(z-1);
    C(i,iii) = (numC/denC);

    %predictor filter
    numFr = (1-beta)*z;
    denFr = (z-beta);
    Fr(ii,iii) = (numFr/denFr);

    %Transfer function
   Tf(i,ii,iii)=  abs(C(i,iii)*P(iii)*Fr(ii,iii))/abs(1+C(i,iii)*Pn(iii));

   %multiplicative error
   e(iii)=(abs(1-exp(-j*w*Ts)))*d ;

   %Condition based on small gain theorem
   limit(i,ii,iii) = Tf(i,ii,iii)*e(iii);

end
     %%####### Check condition for robust stability#####%%

     if (limit(i,ii,:)<1)   %check for robust stability
     tune(i,ii) = 1;
     tune_lambda(i,ii)=lambda;
     tune_beta(i,ii)=beta;
     end
end

end
```

```
        %####### Plot values #####%%

for i=1:9
    for ii =1:9

        if tune(i,ii)==1
           figure(1);
           hold on
           plot(0.1*ii,tune_lambda(i,ii),'o--b')
        else
           figure(1);
           hold on
           plot(0.1*ii,0.1*i,'d--k')
        end

    end
end

 axis([0 1 0 1])
 title ('Stability regions based on tuning')
 ylabel('Tuning for controller (\lambda)');
 xlabel('Tuning for predictor filter (\beta)');
```

Figure 33: Matlab code for FSP tuning

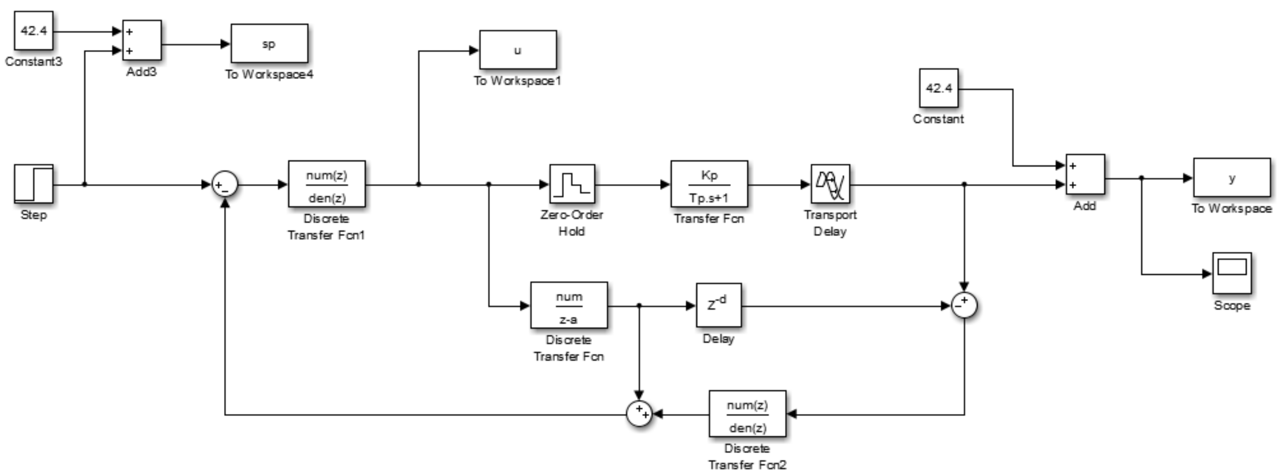

Figure 34: Simulink model for closed-loop temperature control process based on FSP.

```
 %Script file for Simulink model FSP_discrete_model
clc
clear all
close all

% simulation
t_sim=10;                %Simulation time
tstep=1;                 %Step time

% process
Kp=0.183;
Tp=2.76;  %165.57sec
Td=0.27;  %Td =0.27min =16sec
%Change the values of Td to see effect of delay (error=22%)

Td_max=0.33; %20sec (largest delay among three)

% model in continous time
Kn=0.183;
Tn=2.76;  %165.67 sec
Tdn=0.27;  %16sec

%%%%%%%%  PI S-IMC Tuning Method %%%%%%%%%%
Kcx = (1/(2*Tdn))*(1/(Kn/Tn));
Tx = [Tn 8*Tdn];
Tix = min(Tx);


%%%%%% Smith predictor %%%%%%%%
% Robust tuning
dl = (Td_max-Tdn)/Tdn;  %maximum modelling error
To = 1.7*dl;
Kc=Tn/(Kn*To) ;
Ti=2.76;    %Ti = Tn

%Sampling time
Ts = 0.5*dl ;    %Chosen based on Sampling time choice Table 1

% model in discrete time
a = exp(-Ts/Tn);
b =Kp*(1-a);
num = b;
%den = [1 a];
d = round(Tdn/Ts);   %dead-time in discete time (Samples of Td)

%%%%%dicrete controller%%%%
lambda = 0.7;
c = 1-lambda;
numC = (c/b)*[1 -a];
denC = [1 -1];
```

```
%Predictor Filter Fr(z)
beta = 0.1;
numFr = (1-beta)*[1 0];
denFr = [1 -beta];

%Simulation
sim('FSP_discrete_Model',t_sim)

%Calculate Settling time, Rise time and Overshoot for FSP
        S1 = stepinfo(y,time);
        st_fsp = S1.SettlingTime;
        rt_fsp = S1.RiseTime;
        Ov_fsp = S1.Overshoot;

%Calculate Settling time, Rise time and Overshoot for SP
        S2 = stepinfo(y1,time);
        st_sp = S2.SettlingTime;
        rt_sp = S2.RiseTime;
        Ov_sp = S2.Overshoot;

%Calculate Settling time, Rise time and Overshoot for PI
        S3 = stepinfo(y2,time)
        st_pi = S3.SettlingTime;
        rt_pi = S3.RiseTime;
        Ov_pi = S3.Overshoot;

%Calculate IAE
IAE_1 = trapz(time,(abs(sp-y)));  % between set point and FSP
IAE_2 = trapz(time,(abs(sp-y1)));  % between set point and SP
IAE_3 = trapz(time,(abs(sp-y2)))  % between set point and PI

%Calculate ISE
ISE_1 = trapz(time,((sp-y).^2));  % between set point and FSP
ISE_2 = trapz(time,((sp-y1).^2));  % between set point and SP
ISE_3 = trapz(time,((sp-y2).^2))  % between set point and PI

%%%%%%%%%%%%%%%%%%%
h = plot(time,y,time,y1,'--',time,y2,'-.',time,sp,':');   %,,
set(h,'LineWidth',2);
ylabel('Temperature (\circC)');
xlabel('time (minutes)');
title ('Filtered Smith Predictor vs Smith Predictor vs PI')
legend('FSP (\lambda = 0.7, \beta = 0.1)','SP','PI','set-point','Location','SouthEast')
axis([0 10 42.3 44.3])
```

Figure 35: Matlab code for FSP vs CSP and PI

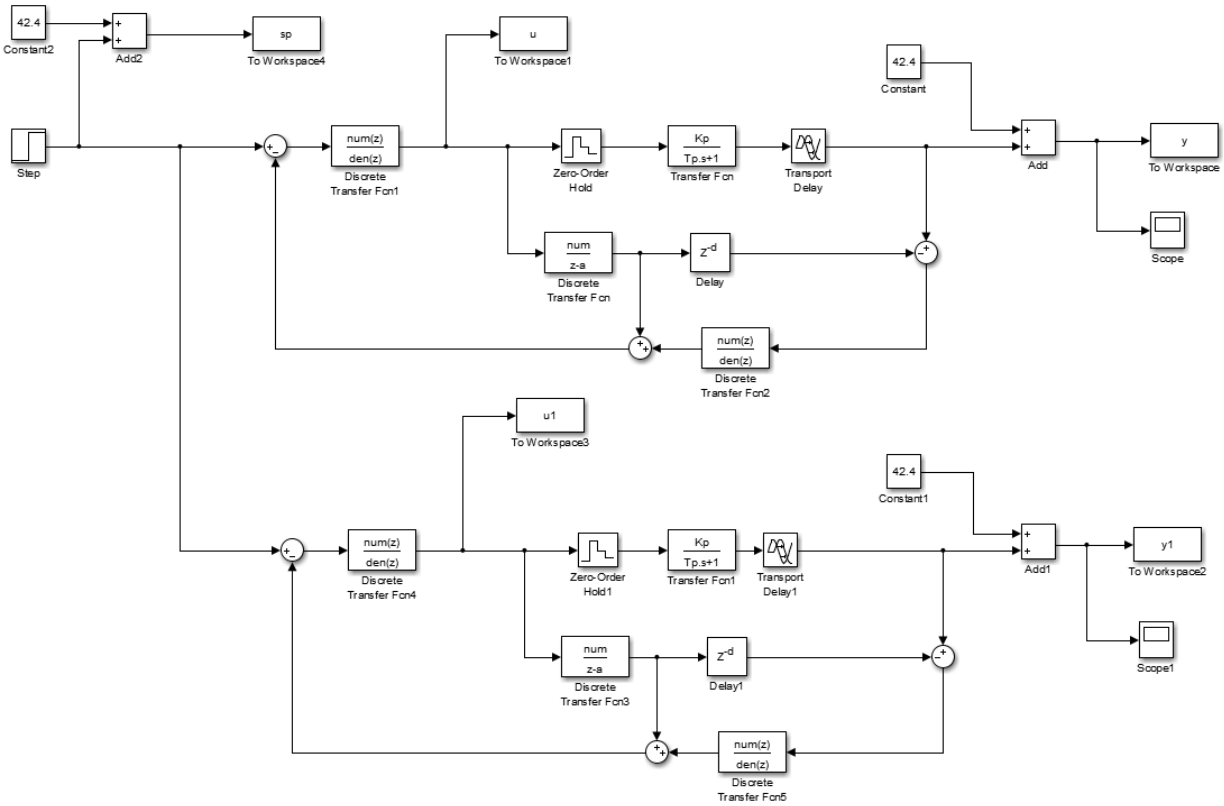

Figure 36: Process time delay variation with FSP

```
%Script file for Simulink model FSP_varying_model
clc
clear all
close all

% simulation
t_sim=10;                 %Simulation time
tstep=1;                  %Step time

% process
Kp=0.183;
Tp=2.76;  %165.57sec
Td=0.27;  %Td =0.27min =16sec

Td_max=0.33; %20sec (largest delay among three)

% model in continous time
Kn=0.183;
Tn=2.76;  %165.67 sec
Tdn=0.27;  %16sec

%maximum modelling error
dl = (Td_max-Tdn)/Tdn;

%Sampling time
Ts = 0.5*dl ;    %Chosen based on guide
```

```
% model in discrete time
a = exp(-Ts/Tn);
b =Kp*(1-a);
num = b;
d = round(Tdn/Ts);   %dead-time in discete time (Samples of Td)

%discrete controller for loop 1
lambda1 = 0.7;
c1 = 1-lambda1;
numC1 = (c1/b)*[1 -a];

%discrete controller for loop 2
lambda2 = 0.1;
c2 = 1-lambda2;
numC2 = (c2/b)*[1 -a];

denC = [1 -1];  %denominator for both loops

%Predictor Filter for loop 1
beta1 = 0.1;
numFr1 = (1-beta1)*[1 0];
denFr1 = [1 -beta1];

%Predictor Filter for loop 2
beta2 = 0.1;
numFr2 = (1-beta2)*[1 0];
denFr2 = [1 -beta2];

%Simulation
sim('FSP_Varying_Model',t_sim)

%Calculate Settling time, Rise time and Overshoot
        S1 = stepinfo(y,time);
        st_robust = S1.SettlingTime;
        rt_robust = S1.RiseTime;
        Ov_robust = S1.Overshoot

%Calculate Settling time, Rise time and Overshoot
        S2 = stepinfo(y1,time);
        st_nrbst = S2.SettlingTime;
        rt_nrbst = S2.RiseTime;
        Ov_nrbst = S2.Overshoot

%Calculate IAE
IAE_1 = trapz(time,(abs(sp-y)));  % for robust FSP
IAE_2 = trapz(time,(abs(sp-y1))) ; % for non-robust FSP

%Calculate ISE
ISE_1 = trapz(time,((sp-y).^2))  % for robust FSP
ISE_2 = trapz(time,((sp-y1).^2))  % for non-robust FSP
```

```
%%%%%%%%%%%%%%%%%%%
h = plot(time,y,time,y1,'--',time,sp,':');
set(h,'LineWidth',2);
ylabel('Temperature (\circC)');
xlabel('time (minutes)');
title ('Filtered Smith Predictor')
legend('FSP robust','FSP non-robust','set-point','Location','SouthEast')
axis([0 10 42.3 44.6])
```

Figure 37: Matlab code for process time delay variation with FSP