University of Southern Queensland

Faculty of Engineering and Surveying

Internal Model Control (IMC) of a Fruit Drying System

A dissertation submitted by

Scott A. Geddes

in fulfilment of the requirement of

Courses ENG4111 and 4112 Research Project

towards the degree of

Bachelor of Engineering (Computer Systems)

Submitted: November, 2006

Abstract

The use of automation within the food processing industry, has allowed for the continual improvement in production and quality control, including that for dried fruit products.

By monitoring and controlling process variables within a commercial fruit dryer (dehydrator), quality control features such as moisture-content, texture, and colour etc, can be attained. The three minimum process variables within most commercial fruit dryers to achieve successful drying are air-temperature, air-velocity, and relative-humidity.

This project investigates the control of air-temperature in a fruit dehydrator by firstly implementing a PID controller. Then as a separate exercise, an IMC controller is implemented, and a performance comparison between the PID and IMC controllers was conducted.

The PID controller was initially designed using the Ziegler-Nichols (step response method), but was determined to be too inaccurate for this purpose. So optimization using Steepest Descent Minimization was also used to determine the PID controller gains.

IMC being considered a robust adaptive controller [6], is especially suited to this plant, having both a large time-constant and transport-delay.

Theoretically IMC can always provide perfect system stability if the open-loop plant is first determined to be stable and an exact model of the plant is maintained, further allowing the designer to adjust other filter parameters without effecting stability.

Determining an IMC controller for this process involved, first obtaining the process inverse (non-linear transport-lag term ignored), then designing a low-pass filter with the filter parameter being the only parameter requiring tuning in the entire IMC control system.

Using performance criteria of time-delay compensation and disturbance rejection for the plant, it was determined that the IMC controller outperformed the PID controller.

University of Southern Queensland
Faculty of Engineering and Surveying

ENG4111 & ENG4112 *Research Project*

Limitations of Use

The council of the University of southern Queensland, its Faculty of Engineering and Surveying, and the staff of the University of Southern Queensland, do not accept any responsibility for the truth, accuracy or completeness of material contained within or associated with this dissertation.

Persons using all or any part of this material do so at their own risk, and not at the risk of the Council of the University of Southern Queensland, its Faculty of Engineering and Surveying or the staff of the University of Southern Queensland.

This dissertation reports an educational exercise and has no purpose or validity beyond this exercise.   The sole purpose of the course pair entitled "Research Project" is to contribute to the overall education within the student's chosen degree program.   This document, the associated hardware, software, drawings, and other material set out in the associated appendices should not be used for any other purpose: if they are so used, it is entirely at the risk of the user.

Prof R Smith
Dean
Faculty of Engineering and Surveying

Certification

I certify that the ideas, designs, experimental work, results, analyses and conclusions set out in this dissertation are entirely my own effort, except where otherwise indicated and acknowledged.

I further certify that the work is original and has not been previously submitted for assessment in any other course or institution, except where specifically stated.

Scott Geddes

Student Number: Q82405374

                  Signature

                  Date

Acknowledgments

I am grateful to Dr. Paul Wen for his valuable comments and suggestions throughout this project and during the preparation of this dissertation.

Scott Geddes

University of Southern Queensland

September 2006.

Contents

List of Figures

List of Tables

Notation

D	distance between heat source and thermostat	[m]
$G_P(s)$	process transfer function	
K_P	proportional gain	
K_I	integral gain	
K_D	derivative gain	
L	transport delay approximation using ZN (step response method)	
R	steepest gradient using ZN (step response method)	
T	general transport delay	[s]
T_m	model transport delay	[s]
T_p	plant transport delay	[s]
τ	plant time-constant	[s]
τ_f	filter time-constant	[s]
V_{air}	air velocity between heat source and thermostat	[m/s]
s	derivative transfer function	
$1/s$	integrator transfer function	

List of Terms

<u>continuous dryer:</u>   Dryer which dries fruit continuously on a conveyor, or rack rail system (as opposed to a batch dryer which needs to be load/unload fruit directly to/from the dryer).

<u>decay ratio:</u>   The ratio of maximum amplitudes of successive oscillations.

<u>dead time:</u>   (see transport delay).

<u>finish dehydration:</u>   The drying of fruit is conducted in a mechanical dehydrator that has first been partly dried in the sun (sun-dried).   The sun-drying component improves the colour of many products (whilst not forget, moisture is also being removed by less expensive sun-drying).

<u>full dehydration:</u>   The drying of fruit is conducted in a mechanical dehydrator from the fresh (wet) state to completion, ready for packaging (often results in poor colour).

<u>overshoot:</u>   The magnitude by which the controlled variable swings past the setpoint.

<u>plant:</u>   The air in the drying chamber of the fruit dryer (dehydrator).

<u>rise time:</u>   The time the response takes to comes up to the new setpoint.

<u>robustness:</u>   The system should not be overly sensitive to changes in process parameters.

<u>setpoint:</u>   Desired output.

settling time: The time it takes the amplitude of the oscillations to decay to some fraction of the change in setpoint.

tunnel dryer: see continuous dryer.

transport delay: Time delay between two processes in a system model.

tray dryer: see batch dryer.

ZN: Ziegler-Nichols

chapter 1

Introduction

1.1 Introduction

Dried tomatoes, apples, figs, sultanas, and also rice, beans (including coffee) and many spices are only a handful of the dried fruit products available to the consumer today. Fruit by definition is the reproductive part of any edible seed plant; yet, consumers of dried fruit usually refer to only any sweet tasting dried plant product as being fruit including for example banana and ginger (which by strict definition is not a fruit). For this reason the term dried fruit in this dissertation refers to any plant, or part of that plant in its dried form.

Historically, many cultures throughout the world have developed methods to dry fruit for; food preservation, culinary pleasure, and medicinal use. But by far sun drying has been the most popular method, although simple kiln arrangements were used for certain seasonal fruits, especially for those harvested in colder climates, and during times of bad weather. In many parts of the world these methods are still practised today.

A popular method of drying is finish dehydration. The fruit is first sun dried until a certain moisture content and colour is obtained, then the fruit is then placed in the dehydrator for the remainder of the time required. As fruit dries, the internal moisture becomes increasingly difficult to remove; and it is at this later stage of drying that the dehydrator can be of most value.

Sun drying certainly has the advantage of using free energy from the sun; however a mechanical dehydrator has the benefit of greater control over the

temperature, air-velocity, relative humidity, thus the total drying regime. Some dehydrators have the ability to monitor parameters in real time such as, specimen size/depth, specimen moisture content, colour etc within the drying environment, to provide real-time feedback to the control parameters (providing a drying regime of quality consistency for different types and sizes of fruit).

One fundamental problem using dehydrators is that, for many fruits the dehydrator cannot give the fruit that authentically sundried appearance demanded by the market. Future research into the application of artificial light (UV) in the drying process may fill this gap in dehydrator technology.

1.2   Project Aim

The aim of this project was to design, simulate, and test a PID controlled system to control temperature for a given plant, then, using this exact same transfer function of the plant, an Internal Model Control (IMC) controlled system model was also investigated, and compared to the PID model.

The fruit dryer process was modelled as a transfer function consisting of a temperature gain term, first order lag term, with a transport delay term. The design of a controller for this process shall consider the following criteria:-

- the system must always be stable and bounded.
- a fast response for this system is not required.
- it must reach setpoint within 60 seconds with less than 5% overshoot at any time.
- the system must be robust enough to control the process model errors and disturbances.
- for the system to be robust enough to control the non-linear transport lag term (in the process model).

1.3 Dissertation Structure

This dissertation is divided into the following chapters.

Chapter 2:   Fruit Drying Modelling

An introduction is given to the types of fruits which are dried, and the parameters that are used to dry them.   This chapter also discusses how these parameters interact with each other, and how they must be modelled in a control system to achieve successful drying.

Chapter 3:   The Fruit Dryer Plant Model

The temperature model of the fruit dryer plant is obtained to allow for simulations of the system in subsequent chapters, and achieve accurate control of this process.

Chapter 4:   PID Temperature Control of the Fruit Drying Plant

The theory of the PID controller is briefly outlined, and then control of the fruit drying plant using the PID is implemented.   Plant/plant model mismatches and disturbances are simulated.

Chapter 5:   Internal Model Control (IMC) Theory

This chapter provides the theory of IMC.   The theory warrants an entire chapter since much of chapter 6 where the IMC controller is implemented numerically, assumes complete knowledge of the reader to the material of chapter 5.

Chapter 6:   IMC Temperature Control of the Fruit Drying Plant

The IMC controller is numerically implemented for the fruit drying plant transfer function.   Plant/plant model mismatches and disturbances are simulated.

Chapter 7: PID/IMC Evaluation and Comparison

An evaluation of the significance of the responses of the PID and IMC systems is conducted then a comparison is made between the performances of both systems.

Chapter 8: Conclusions and Future Work

A final discussion is given on the overall outcomes, and provides some relevant material for future work.

chapter 2

Fruit Drying Modelling

2.1 Introduction

Recent advances in fruit drying technology have led to the development of new methods and techniques to dry fruit, including the use of microwave, infra-red, and U.V. radiation to provide the drying energy [7]. Yet today, drying is still mostly achieved, by placing the fruit in a controlled environment of increased air-temperature and air-flow, and low humidity.

Commercial fruit dryers (mechanical dehydrators) are designed to provide a consistent, quality assured finished product, and a steady production cycle of dried fruit during all weather and seasons. A disadvantage of many dehydrators is that they fail to provide the colour of their 'sundried' counterparts (that the market demands). One solution being developed is a combination of initial UV light application (simulating sundrying) followed then by the insertion into the dehydrator, this can have considerable advantages without the loss of fruit quality.

2.2 The parameters modelled

There are two distinct phases of drying, an initial fast rate of moisture loss followed by a slower second phase. Initially, when the fruit surface is wet, water evaporates from the fruit forming a thin boundary layer of high-humidity air. The thickness of this layer determines the rate of drying in the first phase of drying. Positive forced air movement from a fan over the fruit surface reduces the thickness of the high-humidity layer, thus increasing the evaporation rate.

During the second phase of drying, the rate of moisture loss decreases. The second phase begins when the rate of moisture movement to the surface of the fruit is less than the rate of evaporation from the surface. That is, the speed of drying is limited by the rate at which moisture can move through the fruit tissue. These principles also apply to drying using more traditional means, such as sun-drying.

Under mechanical dehydration the overall speed of drying depends on the relative humidity, and the speed and temperature of air passing over the fruit. These parameters need to be monitored and controlled throughout, and inattention to any of them can jeopardise the success of the entire process.

2.2.1   Relative Humidity

Relative humidity control is the most important factor for efficient dehydration. Air in a dehydrator continually circulated without replacement would rapidly become saturated with water vapour. Evaporation would stop and the fruit would begin to 'cook'.

To avoid this it is necessary to 'bleed off' some of the moist air and replace it with dry air from outside. The aim is to keep the relative humidity below approximately 40%. It must be done carefully because if too much air is bled off with little or no recirculation, heating costs can be extremely high. A compromise must be made between full discharge of partly saturated air (maximum drying rate) and full recirculation of saturated air (minimum heat use).

The effect of relative humidity on drying rate is complex and varies with the moisture level of the fruit. In the early stages of full dehydration, the drying rate is proportional to the relative humidity because of the ready availability of the

moisture to be removed.   As the fruit dries, further moisture is more difficult to remove and maintenance of a low relative humidity level becomes less important. However, in this second stage of the drying process, the relative humidity of air should still be kept below 40%.   Relative humidity of the air is measured before it has passed through the heating unit and fan and after the fruit stack.

2.2.2   Air Speed

The movement of air has two essential functions in the drying of fruit.   It transfers heat from the heating device to the fruit (to provide the energy required to vaporise the water).   Secondly, it serves as a vector for the moisture to be transferred from inside the dehydrator to the outside atmosphere.

Air speeds of 3 to 5 m/s are recommended for fruit dehydration.   Speeds above 6 m/s are used for certain heat sensitive fruits, but these speeds are usually uneconomical because of the much greater power needed to drive the fan.

Air speed is most important in the initial stages of full dehydration when free water is present on the surface of the fruit.   Under these conditions the drying rate is doubled when the airspeed is increased.   Less attention need be paid to airspeed in the later stages of drying or when using finish dehydration, but airspeed of less than 3.0 m/s can slow the drying rate.   Dehydrators designed with a fixed airspeed of 3.4 m/s over the fruit circulate a sufficient volume of air to provide heat for both evaporation and the removal of moisture from the unit for most fruits.

2.2.3   Air Temperature

Air temperature is increased to supply the heat required to evaporate fruit moisture and to increase the moisture-carrying capacity of the air.

Air at 60°C can carry five times more moisture than air at 32°C.   Compared with

cold air, a relatively small volume of hot air is needed to carry moisture out of the dehydrator.   Additional heat is necessary to heat trays, compensate for heat lost through insulation, and heat the fresh air required to maintain a low humidity.

The maximum operating temperature is determined by the temperature at which discolouration and off flavours is produced in the fruit (see below).

Maximum counter flow operating temperatures.	
Fruit	Max. temp. (o C)
Coffee	95
Nuts & Grains	85
Prunes	70
Apples	65
Grapes	65
Nectarines	65
Peaches	65
Apricots	55
Pears	55
Spices	35

Table 2-1 Maximum dehydrator temperatures using full dehydration.

2.3   Batch or Continuous Flow

The dehydrator can be worked either as a batch system, or a continuous flow system.

In the batch system the dehydrator is filled with fruit and run until the entire load is dried to the desired moisture content.   Another complete batch is then loaded, and the cycle repeated.

The continuous flow system is started by filling the dehydrator to one-third to half of its capacity.   Racks on wheels (or trucks) of fruit are routinely added to the

start of the dehydrator until it is full (or almost full). Then, when the first truck has dried sufficiently it is removed from the finish, and another put in at the start, and so on. This method is recommended for finish-drying stone fruits and pears.

A problem with the batch system is uneven rates of moisture removal from the fruit at different locations throughout the dehydrator. That is to say, fruit closest to the fan and heat source shall become dry quickest whilst microclimates near corners and badly sealed doors may be slower. For these reasons, at the completion of drying in a batch system there is a gradient in fruit moisture content from low at one end of the dehydrator to high at the other end.

Another problem associated with the batch system is the continual adjustment of shutters required to control relative-humidity during the drying cycle. This problem can be overcome with automatic relative-humidity control. An automated system allows the dehydrator to be operated overnight without supervision.

In contrast, in a continuous flow system, some fruit must be loaded and unloaded periodically while the dehydrator is operating, requiring labour around the clock. This regular removal and replacement of fruit to and from the dehydrator results in some extra heat loss when the doors are opened. A special truck and rail system are essential for efficient operation and to minimise energy losses. As the trucks of fruit are removed and reloaded periodically, the labour required for placing and removing fruit from the trays is spread out, as is the requirement for fresh fruit when using a continuous flow system.

2.4   Parallel, Counter or Cross Flow

Airflow through a continuous flow system can be parallel flow, which is the same direction as the fruit, counter flow, which is in the reverse direction to the fruit, or finally cross flow, which is perpendicular to direction of fruit.

The counter flow system is recommended for finish dehydration, because the high moisture air from each new truck entering does not pass over the drier fruit. Drying is slow at first because of the high humidity and low temperature, but as the truck proceeds through the dehydrator it is exposed to air of higher temperature and lower relative humidity.   Care should be taken with temperature control because the driest fruit, which is most sensitive to high temperature, is exposed to the maximum operating temperature within the unit.   Parallel and cross flow is more effective when using full dehydration.

chapter 3

The Fruit Dryer Plant Model

3.1   Introduction

Monitored parameters such as air-temperature, air-speed and humidity would typically be only the minimum parameters monitored.   The drying regime for a fruit, that is, for different species, different size/thicknesses etc, are all controlled by adjusting air-temperature, air-speed and humidity during a prescribed time regime.   Although some dehydrators have the ability to monitor these and other variables in real time such as, moisture-content, color etc, allowing for the specimen characteristics to dictate the drying regime, and not just simply a drying time.

Although a modern commercial fruit dryer may provide the means to control and monitor all these variables within a drying chamber, this project shall be limited to the control and monitoring of the dryer's air-temperature only.

3.2   The general fruit dryer model

A modern commercial mechanical dehydrator consists of a drying chamber (or tunnel), shutters/louvres, fan, and heat source (see figure below).
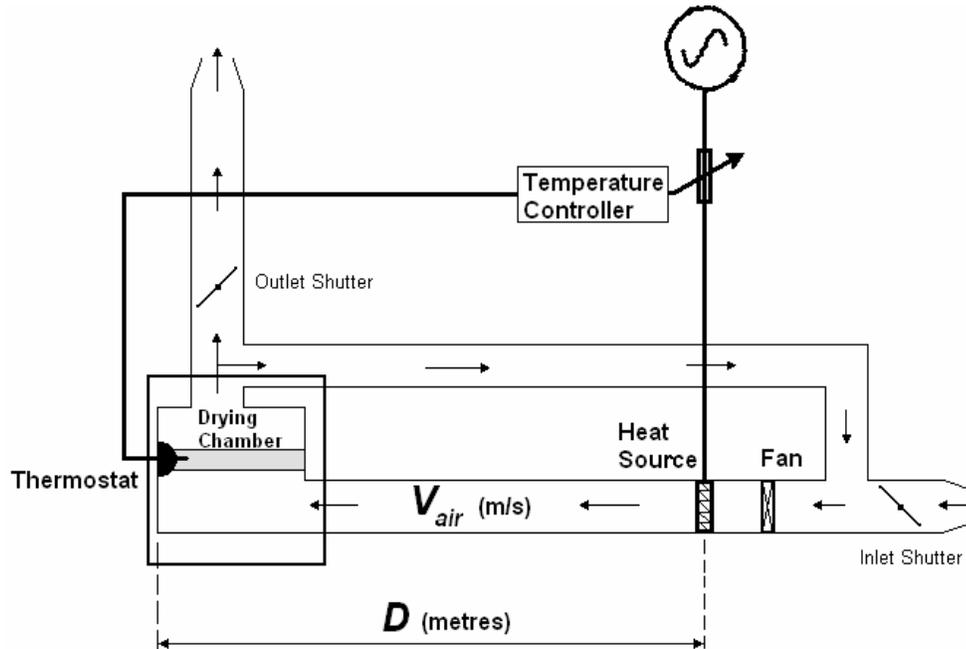
Figure 3-1 Typical fruit dryer showing thermal schematic of the plant.

The fruit drying process above was modelled by:

- a first-order lag term for the heat source (heater banks),

- a delay term acting between the heater banks and the drying chamber,

- an inverse gain term for the air temperature in the drying chamber (the air in the drying chamber was simply assumed to be a linear multiple of the heater-bank elements temperature),

- and an inverse gain term for the temperature to voltage conversion of the feedback thermostat.
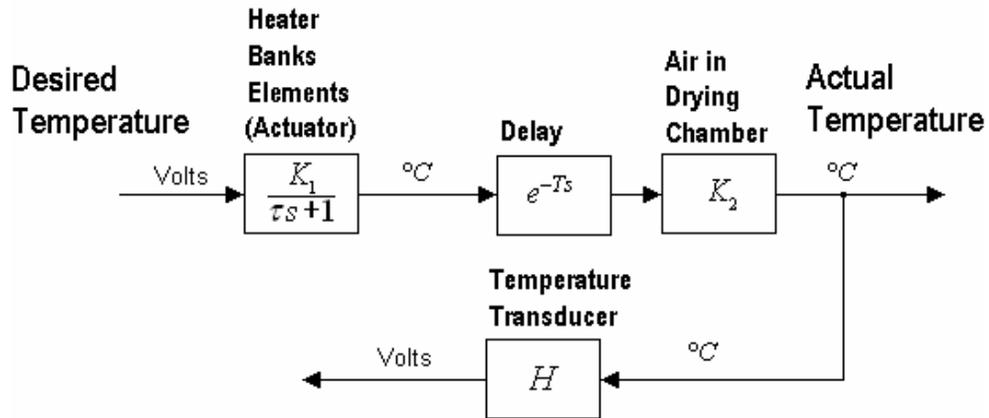
Figure 3-2 System model of the open-loop thermal transfer function of the plant.

The open-loop transfer function is then be represented by:

$$G_P(s)H(s) = \frac{K_1 K_2 H}{\tau s + 1} e^{-Ts}$$

$$\therefore G_P(s)H(s) = \frac{K_1 K_2 H}{\tau s + 1} e^{-\left[\frac{D}{V_{air}}\right]s}$$

3.3 The numerical fruit dryer model

This system was a 24V system, and was designed with $H = 1/5$. So if 1V was applied to the input of the controlled closed-loop system, then there was a temperature of 5 degrees Celsius at the output. Thus our dryer's upper temperature limit was 120 degrees Celsius. The feedback term not only provided feedback, but also provided a temperature to voltage conversion.

The first order lag term (heater bank elements) was modelled with a time-constant τ of 10 seconds, and $K_1 = 30$. It was assumed that air in the drying chamber would take just less than one minute (ie 5τ approximately) to

reach a steady state value when the open-loop system was provided with a step input.

It was assumed in our model that the air temperature in the drying chamber was 20% of the heater bank element's temperature. So a value of $K_2 = 0.2$ was simply chosen to model the air temperature in the drying chamber.

The value for D is small, in small modular fruit dryers that are available, but a larger food manufacturer providing 'hot-air' to many areas of the factory floor could typically have the heat source positioned larger distances away from the drying chamber. So an arbitrary value for the distance between the heat source and the thermostat was made to be $D = 36\ metres$.

Recall from the previous chapter where the minimum and maximum air speeds used for drying fruit was 3 m/s and 6 m/s respectively. Therefore using the table below as an example we can determine the transport-delay times for when the fan speed is set for a particular type of fruit.

		D=36 metres
	Vair	T=D/Vair
	(m/s)	(s)
Spices	3	12
Apricots	6	6

Table 3-1 Transport delay time for a particular fan setting.

In this dissertation we mainly focused on the setting for maximum transport delay (T=12 seconds) when designing our control systems.

3.4 The open-loop plant transfer function of the fruit dryer

So our open-loop plant transfer function for minimum and maximum transport-delay is respectively:

$$G_P(s)H(s) = \frac{30 \times 0.2 \times 0.2}{10s+1} e^{-\left[\frac{36}{V_{air}}\right]s}$$

$$\therefore G_P(s)H(s) = \frac{1.2e^{-6s}}{10s+1} \quad and \quad G_P(s)H(s) = \frac{1.2e^{-12s}}{10s+1}$$

chapter 4

PID Temperature Control for the Fruit Drying Plant

4.1 Introduction

The aim of this chapter is to detail the design, then simulate a PID controller for the temperature system model of the fruit dryer.

The tuning parameters shall be obtained, then system responses shall be recorded observing certain design objectives (for later comparison with the IMC input/output responses).

4.2 PID Controller Review

Controllers based on the PID design algorithm remain the most popular due to their inherent ability to converge to a solution for most linear (and many non-linear) applications, for a large given domain of inputs, and/or initial values.   The ability for a PID system to perform with stability, with low (or zero) transient and steady state errors, depends on the accurate selection of the tuning parameters K_P, K_I, and K_D.

$$u(t) = K_P\left[e(t) + \frac{1}{\tau_i}\int_0^t e(t)\,d\tau + \tau_d\,\frac{d}{dt}e(t)\right]$$

$$u(t) = K_P e(t) + K_I\int_0^t e(t)\,d\tau + K_D\,\frac{d}{dt}e(t)$$

$$U(s) = E(s)\left[K_P + K_I\,\frac{1}{s} + K_D s\right]$$

where: $u(t)$ = controller output (and the total error)

$e(t)$ = desired value – measured value

K_P, K_I, and K_D are the respective error term gains.

To allow the controller to be designed within a stable system criterion, these three error gain terms will determine the response of the closed-loop system to inputs and initial conditions by the following action:

- provide control action via proportionality to the error, implemented using an all-pass gain factor.

- reducing steady-state errors via low-frequency compensation, implemented using an integrator.

- improving transient response via high-frequency compensation, implemented using a differentiator.

Controller parameters are tuned such that the closed-loop control system is always stable and should meet given objectives associated with the following:

- stability, robustness.

- setpoint tracking performance at transient, including rise-time, overshoot, and settling time.

- steady-state error performance.

- disturbance rejection from load surges, plant/model uncertainties, environmental noise.

	Rise time	Overshoot	Settling time	Steady state error	Stability
K_P Increasing	decrease	increase	small increase	decrease	degrade
K_I Increasing	small decrease	increase	increase	large decrease	degrade
K_D Increasing	small decrease	decrease	decrease	small change	improves

Table 4-1 Effects of independently tuning gain values in closed-loop system.

For given objectives, tuning techniques for PID controllers can be categorized into two general methods according to their application as follows:

- Analytical methods   Tuning parameters are calculated from an analytical or algebraic relation between the plant model and an objective (eg. IMC) needs to be in an analytical form and the model must remain extremely accurate.   Real-time automated tuning would be included in this method.

- Heuristic methods   Manual tuning/programming (such as the Ziegler-Nichols tuning rules) or from an artificial intelligence base in the form of a neural network rule/formulae).

Both methods may utilize numerical optimization algorithms, providing controller

parameters to produce many different improved responses to the system.

4.3   PID control of temperature

The open-loop plant was now transformed into a closed loop system, with the addition of feedback and a PID controller. Two methods were investigated to determine whether they could provide accurate temperature setpoint tracking, within a reasonably fast response time for the dryer, without compromising stability of the system.   They were, the Ziegler-Nichols (step-response method), and optimization of the PID coefficients using a steepest descent minimization algorithm.
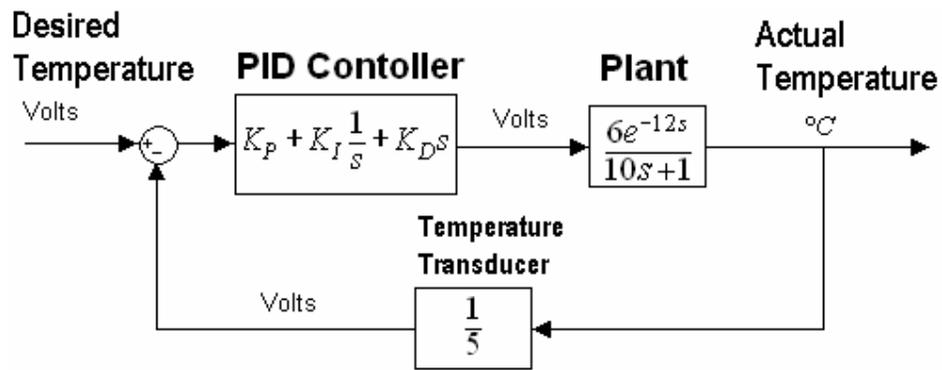
Figure 4-1 PID closed loop system implementation.

4.4   Ziegler-Nichols (step-response method)

The Ziegler-Nichols step-response method was initially used to design the controller using [1].   It is a method whereby process information (tuning parameters) is obtained from the open-loop step response of the process.   Then these tuning parameters are used in the design of the controller for the closed-loop system.   This method can only be used when a very simple response is being used to model the process, and is primarily used for processes with the following general open-loop transfer function model (which is the same

as ours):

$$G(s) = \frac{Ke^{-Ts}}{\tau s + 1}$$

This process has a general open-loop step response below:
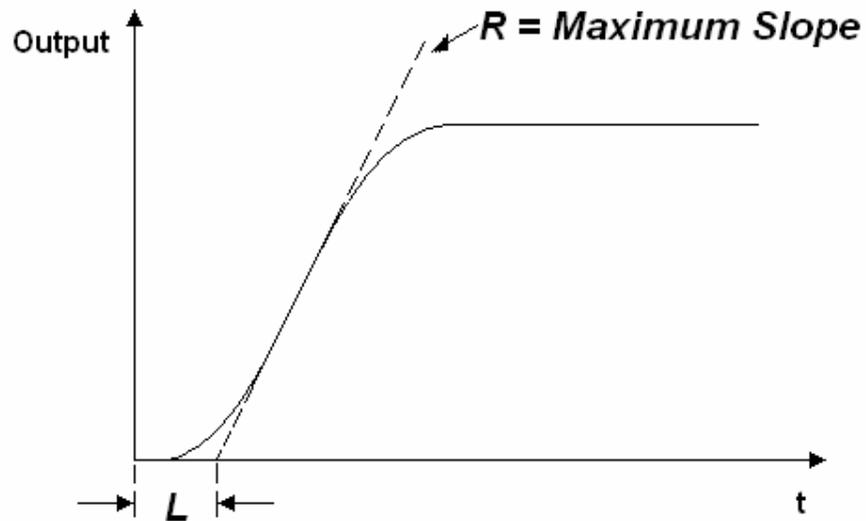
Figure 4-2 Open loop step response of the process.

To design the controller from this output the following steps are taken:

- determine the steepest gradient R.

- determine the delay term L.

- R and L are then substituted into the following table to obtain the gain terms of the controller.

Parameters	Denotation
Proportional gain	Kp=1.2/(R*L)
Integral time-constant	Ti=2*L
" gain term	Ki=Kp/Ti
Derivative time-constant	Td=0.5*L
" gain term	Kd=Kp*Td

Table 4-2 PID parameters for Ziegler-Nichols step response method.

- then the gain terms are substituted into in the controller equation,

$$U(s) = \left[K_P + K_I \frac{1}{s} + K_D s \right] E(s)$$

- $U(s)$ is the controlled input signal to the plant, it is this signal which determines the output $Y(s)$ in the closed-loop system as follows:
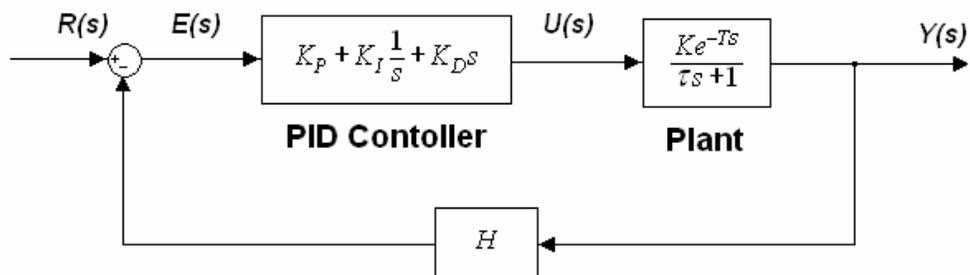
Figure 4-3 PID controller implementation.

4.5   PID Temperature Controller Simulation

The components of the model of the fruit drying process was exactly in the form of the one just described, comprising a gain term, a delay term, and a first-order lag term, and with a feedback term to consider.   The open-loop step response of our

fruit drying process was considered to be one of the most popular open-loop responses encountered in industry.

To design the PID controller for the fruit drying process we shall follow the steps outlined in the previous section as follows:

- obtain the open-loop step response below:
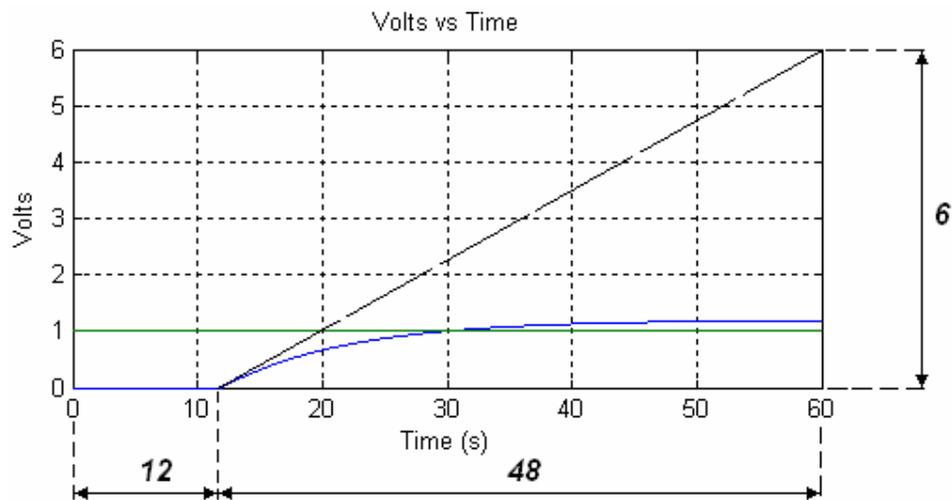
Figure 4-4 Response of open-loop system to a 1V step input.

- the steepest gradient term was determined to be $R = \dfrac{6}{48}$ (although choosing the numerator term was difficult since there was a high degree of discretion due to the curvature of the response).

- the delay term $L = 12$.

- R and L are then substituted into the following table to obtain the gain terms of the controller.

| Parameters	Denotation	Value
Proportional gain	Kp=1.2/(R*L)	0.8000
Integral time-constant	Ti=2*L	24.0000
" gain term	Ki=Kp/Ti	0.0333
Derivative time-constant	Td=0.5*L	6.0000
" gain term	Kd=Kp*Td	4.8000

Figure 4-5 PID parameters for Ziegler-Nichols steepest descent method.

A simulation of the closed-loop system was conducted with the above determined controller gain values:
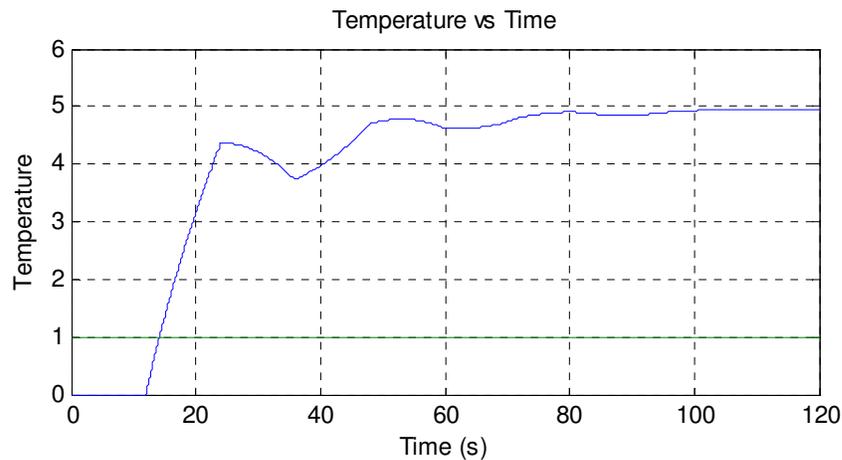
Figure 4-6 Step input response using ZN PID controller parameters.

From the above plot it is immediately seen to have oscillatory behaviour and does not reach steady state until past 120 seconds. It also illustrates one of the drawbacks of using the step response method, as opposed to using an optimization technique. And that being the step response method uses larger integral and derivative time-constants than optimization techniques, disallowing

the system to reach steady-state sooner. Also the slope R near the time-axis can be too large for visual judgement (or other graphical means of estimation), which when implemented allows for inaccurate gain control action to occur, thus a slower time to reach set-point.

4.6   Optimization Theory

Optimization allowed us to tune the PID parameters by a more accurate criterion required by the system. In this case the specifications of the controller were to disallow overshoot above 5%, but still reach steady state within the required 60 seconds.

The optimization technique used here shall be the steepest decent minimization method from [1] (p. 4.12). The system details and numerical calculations are contained in the source code in appendix B, but a general discussion of this method shall be given here.

Below is the model being simulated to determine PID controller gains:
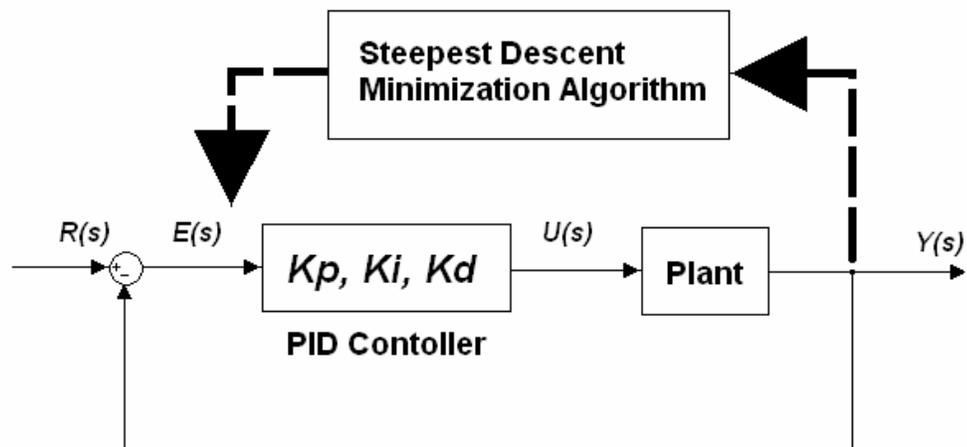
Figure 4-7 PID controller implementation.

An initialization of PID parameters (see code Appendix B) was chosen, and then the response to a step input was simulated. The simulated error value E(s), is tested against the IAE (Integral Absolute Error) criteria, which forces a magnitude increase/decrease of the respective PID controller gains.

The IAE performance criteria used to obtain these minimum values shall be:

$$IAE = \int_0^T |e(t)| \, dt$$

And the general algorithm to implement steepest descent minimization shall be:

$$Optimized \ System = \min[IAE] = \min\left[\int_0^T |e(t)| \, dt\right]$$

Steepest descent minimization searches for a minimum slope. An initial value is chosen, simulated, substituted into an objective function, and then the magnitude of this objective function is compared with the previously simulated value. The minimum value is chosen for the next iteration, whilst the maximum of the two is rejected.

In more detail, steepest descent minimization uses the surface variable S in say 3 dimensions x, y, z (but could use higher dimensionality), where the gradient of S is:

$$\nabla S = \frac{\partial S}{\partial x}\hat{i} + \frac{\partial S}{\partial y}\hat{j} + \frac{\partial S}{\partial z}\hat{k}$$

This method searches using its objective function in the direction of negative gradient $-\nabla S$. Initial values $x(0), y(0), z(0)$ are chosen then iteratively run through following formulae:

$$x(k+1) = x(k) - \eta \nabla S_x(k)$$
$$y(k+1) = y(k) - \eta \nabla S_y(k)$$
$$z(k+1) = z(k) - \eta \nabla S_z(k)$$

(where η is the normalized step size)

If there is proportionality to the negative gradient $-\nabla S$, then the (k+1) iteration replaces the (k) iteration. If there is not proportionality to $-\nabla S$, then the (k) iteration is simply retained (see source code appendix B).

4.7   Optimization of PID gains

The following system model was optimized.
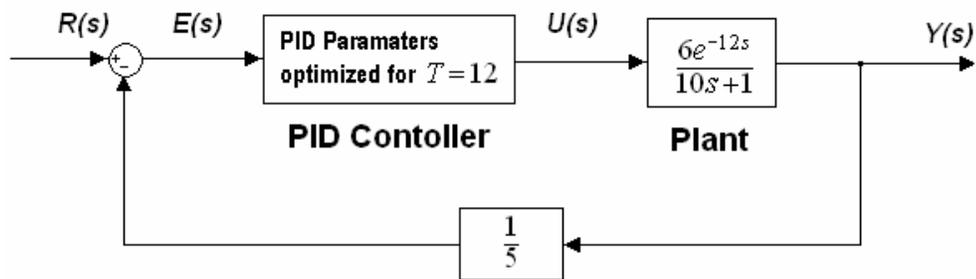
Figure 4-8 System model for plant transport delay $T = 12$**.**

After optimization of the system (see appendix B for source code used) the following PID controller parameters were obtained and a simulation using these values was conducted.

Parameters	Denotation	Value
Proportional gain	Kp	0.6170
Integral step time	Ti	0.0500
" gain term	Ki	0.0649
Derivative step time	Td	0.0500
" gain term	Kd	0.1796

Figure 4-9 PID controller parameters obtained from optimization.

The response of the system using the above optimized PID controller parameters is shown below.
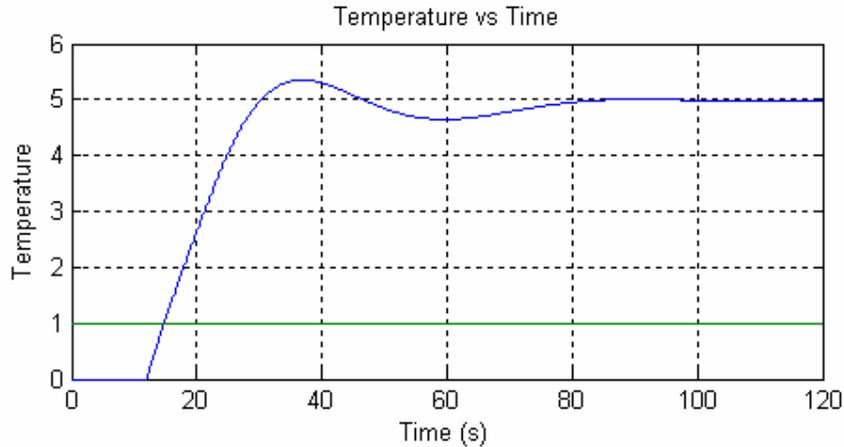
Figure 4-10 Response using optimized PID controller parameters.

The system did not reach our specified setpoint in under 60 seconds. It displayed approximately 10% overshoot initially, but had a high enough damping to eliminate oscillatory behaviour within approximately 80 seconds.

Two other choices were then considered at this stage, to use a system with less damping, or use an optimized PI controller instead. The higher damped system overcame the overshoot problem but did not solve the problem of reaching set-point within specifications when disturbances were considered later. The optimized PI controller had a similar problem meeting overshoot specifications (specified as 5%). So a decision was made to implement the PID controlled system with this overshoot compromised.

4.7.1 Frequency response of the optimized PID system

As part of determining the PID system's stability, noting that as the transport-delay T increased the phase margin decreased, resulting in the system becoming less stable, a check on the system stability was conducted.

As the transport-delay increases, its phase margin usually decreases, indicating that the system is becoming less stable (or unstable). If there is a decrease in the transport-delay this usually increases the phase margin and can indicate that the system is becoming more stable.
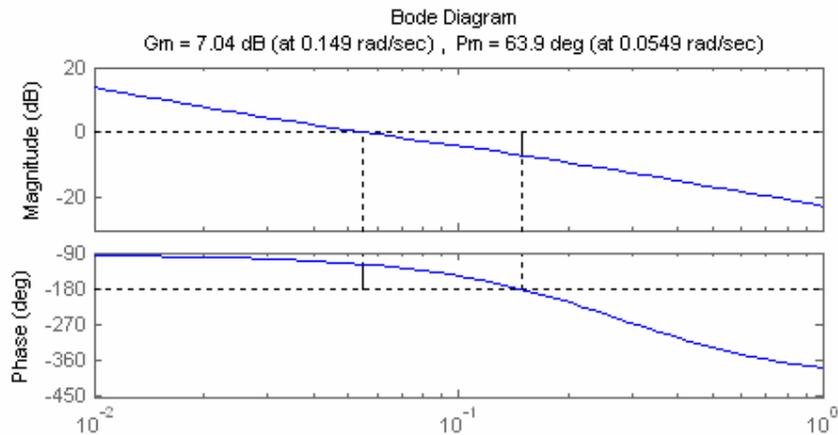
Figure 4-11 Frequency response using optimized PID controller parameters.

The system was stable with a comfortable Pm=63.9 degrees, and the use of a PID compensator was not required. This robustness needed to be established before accurate responses to disturbances could be evaluated, and also a stable robust PID system can only be properly compared with a stable robust IMC controller later.

4.8 Simulation of the PID controlled fruit dryer system

To fully test the PID controlled system, the following change in plant parameter values and other external disturbance simulations were independently conducted:

1. A change in the plant, due to a change in plant transport delay values.

2. A change in the plant, due to a change plant time-constant values.

3. Unit step disturbance at the plant output.

4. Pulse disturbance in at the plant output.

5. White noise disturbance at the plant output.

4.8.1 PID – Simulation of varying plant transport delays.

For systems requiring robustness to changes in transport delay capabilities, this is not usually where the application the PID controllers have their best reputation. Although our fruit dryer model was designed using the value of $T_d = 12$ seconds to fully consider the effects of simulated drifts in transport delay values, smaller transport delay times are considered, using the following model.
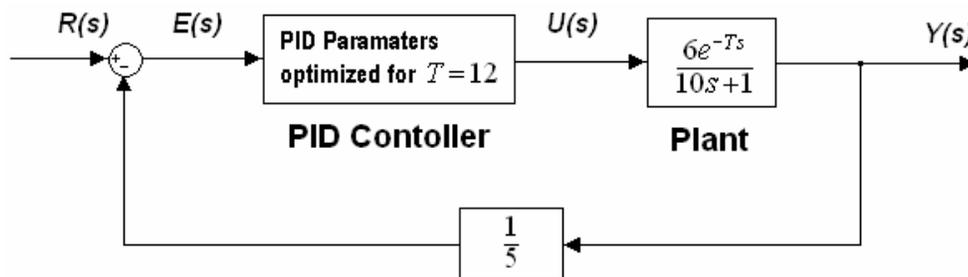
Figure 4-12 Model considered with varying plant transport delay values

$$T_d.$$

Using now the model of the original transfer function with $T_d = 12$, then observe below the response due to incremental changes in the transport delay T_d, stepping down from $T_d = 12$ to $T_d = 6$ in steps of 3 seconds (this is simulating V_{air} increasing). While keeping all other parameters the same for a moment) provides us with a system to simulate below.
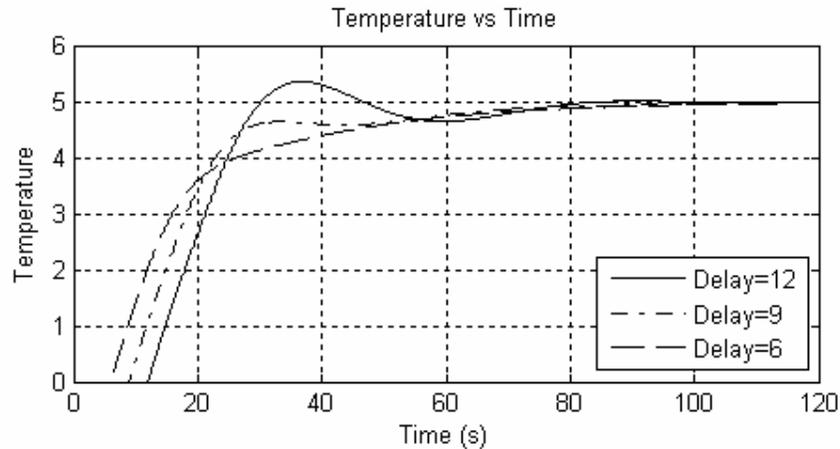
Figure 4-13 PID step response to varying plant transport delay from

$T_d = 12$ **to** $T_d = 6$ **(in steps of 3 seconds).**

As the transport delay was decreased, the response experienced more damping with a much slower rise time, but fell short of reaching set point within 60 seconds but did not display any instability. Although when $T_d > 12$ the system quickly experiences large values of overshoot.

For every fan setting (ie V_{air}), that is, for every transport delay value, it would be required to complete the optimization of their respective PID parameters, and use those PID parameters for that particular V_{air} setting. This would be a formidable task if the sensitivity of the system required the optimization of PID parameters at many time steps to achieve the desired accuracy.

4.8.2   PID – Simulation of varying plant time-constants.

During the lifetime of a plant (ie over the longer term) its time-constant may drift in value. Or at some stage there may be a physical alteration made to the plant that may cause a sudden change in the plant's time-constant that needs to be compensated for. A model of such a system is shown below:
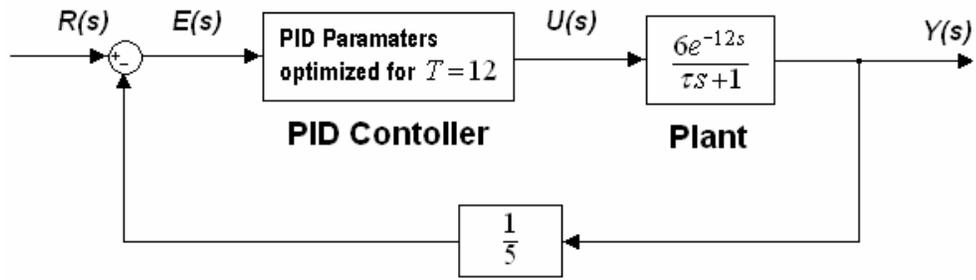
Figure 4-14 Model considered with varying plant time-constant values τ.

To illustrate this, a simulation was made whereby our modelled plant time-constant value of $\tau = 10$ seconds (original designed system), was varied from $\tau = 8$ to $\tau = 14$ (values chosen were $\tau = 8, 10, \& 14$). These values for τ were chosen to not cause more than 10% overshoot in the response (see response below).
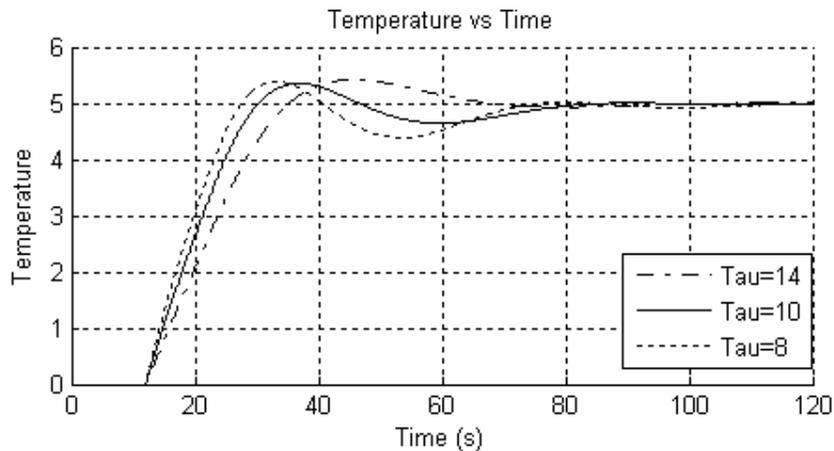
Figure 4-15 PID model step response to varying plant time-constant, values τ seconds.

As expected a reduction in the time-constant to $\tau_f = 8$ allowed higher frequency components to enter the system which was not intended for in the controller design for $\tau_f = 10$, and overshoot and a more oscillatory system resulted.

Values higher than $\tau_f = 14$ increase overshoot and damping but neither of these disturbances reaches setpoint within 60 seconds.

4.8.3   PID - Unit step disturbance at the plant output.

Simulating disturbances at summing junctions is not a multiplicative process, thus a better understanding of the magnitudes in the response of the system shall be gained by using the full system (non-normalized).   This practise shall be adopted for all simulated disturbances at summing junctions in this dissertation.
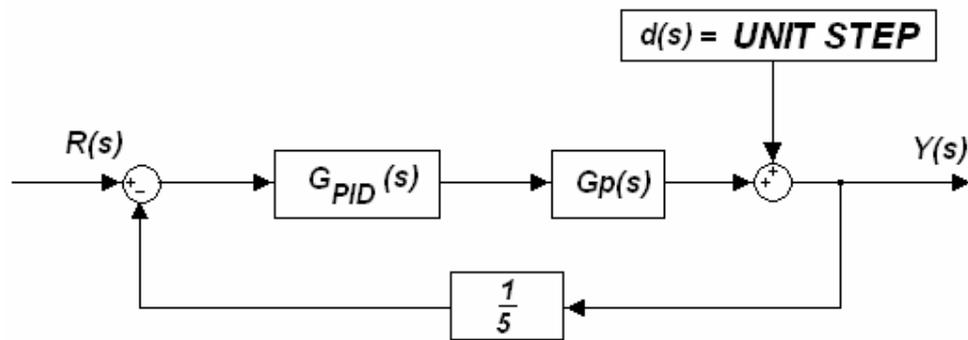
Figure 4-16 PID model for a unit step disturbance at the plant output.

To simulate a sustained external disturbance at the output of the plant a unit step disturbance of (1 degrees Celsius) was applied at t=80 seconds (chosen since system was closest to steady state).   It was seen that the step was sustained at the output until t=92 seconds due to the 12 second delay, at which time the controller forces the system to reach the steady state at approximately t=120 seconds.   This system could be regarded as robust since it displays no instability, and displays set-point tracking to a unit step disturbance.
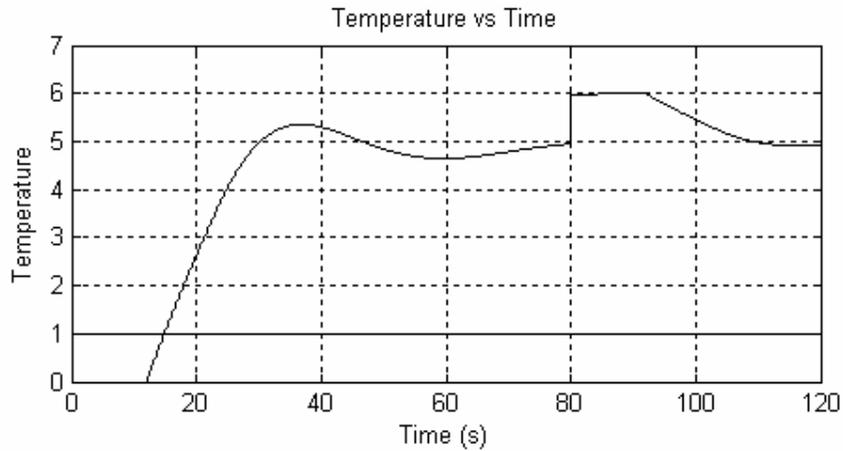
Figure 4-17 PID model response to a unit step disturbance of 1 degrees Celsius.

4.8.4　PID - Unit pulse disturbance at the plant output.
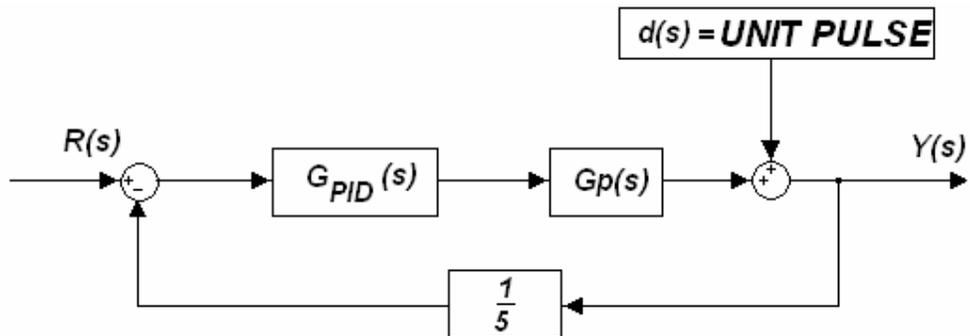
Figure 4-18 PID model for a unit pulse disturbance at the plant output.

A unit pulse disturbance (of 1 degrees Celsius, for 1 second duration) was applied at t=80 seconds (arbitrarily chosen) with the output response shown below.　Again the effects of the delay term disallow the control action until t=92 seconds, at which time the difference between the plant and model was fed back to the set point summing junction to further allow for the controller's action.　By t=110 seconds the system has returned to the set point, and again displays all the

attributes of a stable, robust system:
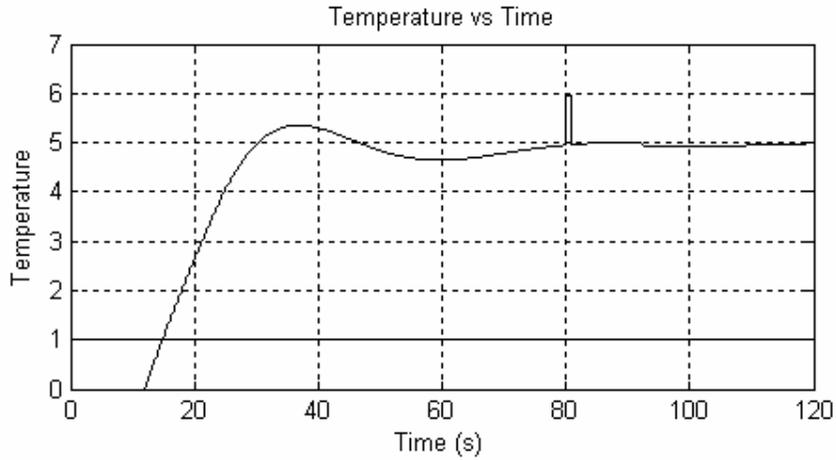
Figure 4-19 PID model response to a unit pulse disturbance of 1 degrees Celcius.

4.8.5   PID - Band limited white noise disturbance at the plant output.
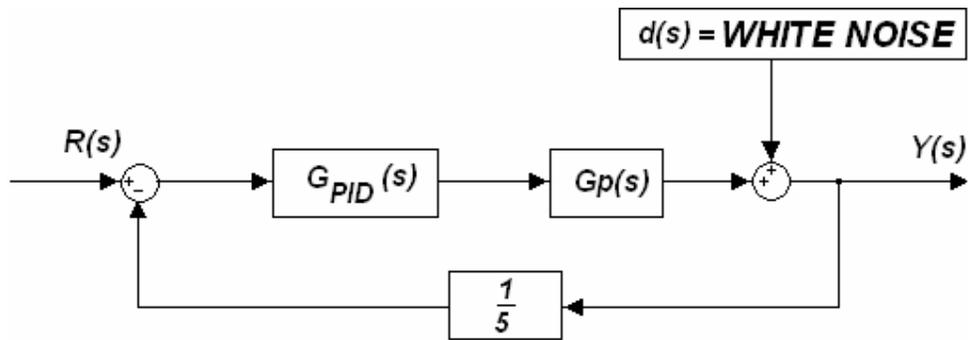
Figure 4-20 PID model of a band limited white noise disturbance at the plant output.

Very often the plant may suffer from random disturbances which may not be easily identifiable, which means that a simple transfer function to model the

disturbance (as was the case with the step and pulse disturbances) may not be adequate.

A stochastic disturbance model in the form of a band limited white noise source shall be used to simulate unknown disturbances of this kind. The output of the random disturbance subsystem is shown below. It comprises a sinusoid with variable amplitude and frequency (the low-pass filter cut off frequency was 2 rad/s (0.32 Hz)).
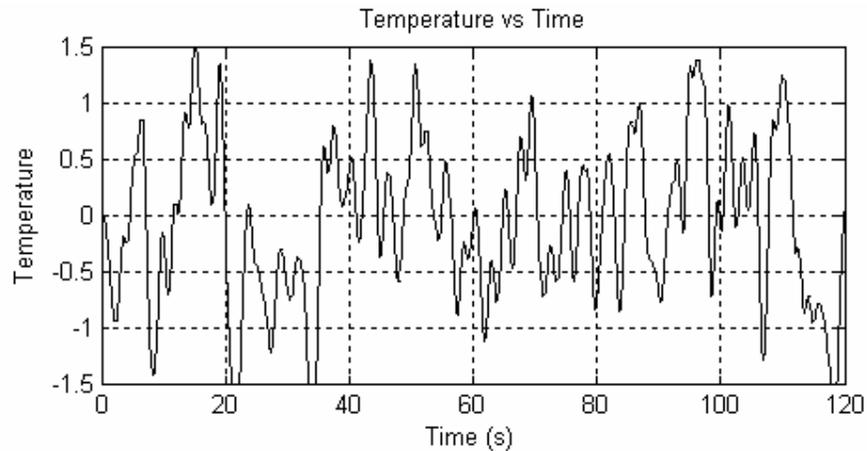
Figure 4-21 Simulation of band limited white noise disturbance source.

The response of the system is shown below, it closely follows the noise (since the disturbance was applied to the output of the plant), there was no instability, and oscillates about the correct set point of 5 degrees Celsius.
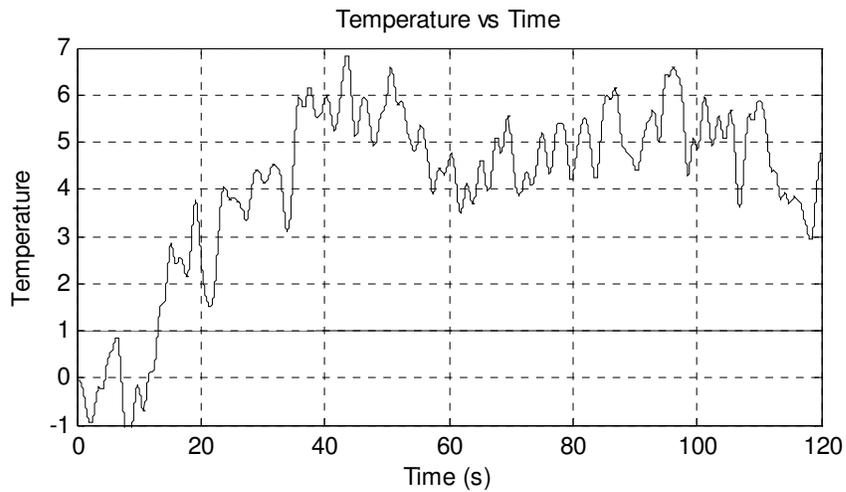
Figure 4-22 PID model response to a band limited white noise disturbance source.

4.9   PID Conclusion

This chapter discussed the theory of the PID controller, the implemented PID controller design of the temperature controller of our fruit drying plant. To determine PID parameters The ZN (steepest decent method) was first used but proved to be too inaccurate, and the IAE Optimization method was used.

The controller was tested by changing plant parameters, introducing disturbances, then observing the response.   No attempt was made to compensate for undesirable responses caused by simulated changes in plant parameters, as this would require the re-calculation of PID parameters for every plant parameter change, and disturbance simulated in this chapter.   A possible remedy would be to have a range of PID parameters that could be dynamically loaded into the PID controller for a range of different responses.   But this still leaves the formidable task of dynamically detecting the response, to decide upon which PID parameters to load into the PID controller within a reasonable timeframe.

chapter 5

Internal Model Control (IMC) Theory

5.1   IMC Introduction

The IMC controller is a model based controller, and is considered to be robust. Mathematically, robust means that the controller must perform to specification, not just for one model but for a set of models [2].   The IMC controller design philosophy adheres to this robustness by considering all process model errors as bounded and stable (including transport lag differences between the model and the physical system).   IMC is implemented by firstly obtaining the inverse of the process (or invertible components of it) to be controlled, then multiplying this calculated inverse with a low-pass filter.   The output response to the reference inputs, the sensitivity function and the complementary sensitivity function can be adjusted directly by the low-pass filter [8].

5.2   IMC System Theory

The theory of IMC states that "*control can be achieved only if the control system encapsulates, either implicitly or explicitly, some representation of the process to be controlled*" [6].   A further mathematical definition using the controller $G_c(s)$, and the process $G_p(s)$ is given below:
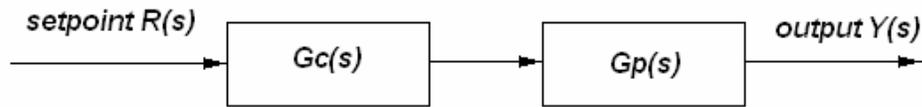
Figure 5-1 Open-loop control system.

Let $\tilde{G}_p(s)$ be a model of $G_p(s)$, and

let $G_c(s) = \tilde{G}_p(s)^{-1}$ (ie the controller is the inverse of the process model),

then $G_p(s) = \tilde{G}_p(s)$ (ie an exact representation of the process).

This theoretical control performance whereby the output remains equal to the input (or setpoint), without the use of feedback, informs us of two things:

- assuming we have complete knowledge of the process (encapsulated in the process model) being controlled, then perfect control can be achieved.
- feedback is only necessary when knowledge about the process is inaccurate or incomplete.

Both these above conditions in practise are unachievable in an open-loop system, for the following reasons:

- a mismatch between the actual process, and the process model.
- the process model may not be invertible.
- unknown disturbances within the system.

IMC is a closed-loop system design that, takes into consideration the above three conditions, to achieve controllability of the process.

5.3   IMC General System Design

Below is the general IMC system design [6]:
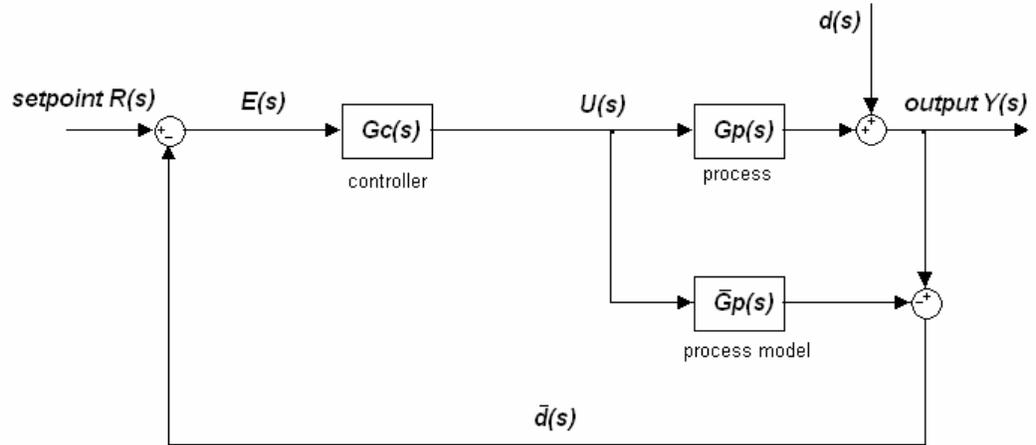
Figure 5-2 IMC control system.

The output, $Y(s)$, is compared with the output of the process model, resulting in

signal $\tilde{d}(s)$ below,

$$\tilde{d}(s) = \left[G_p(s) - \tilde{G}_p(s) \right] U(s) + d(s)$$

If $\tilde{d}(s) = 0$, then, $\tilde{d}(s)$ is a measure of the difference in behaviour between the

process and its model.

If $G_p(s) = \tilde{G}_p(s)$, then $\tilde{d}(s) = d(s)$,

thus $\tilde{d}(s)$ is considered the missing information in the process model $\tilde{G}_p(s)$,

and therefore can be used to improve control.

Then $\tilde{d}(s)$ is used to subtract from the setpoint $R(s)$ here below,

$$U(s) = \left[R(s) - \tilde{d}(s) \right] G_c(s)$$

$$U(s) = \left[R(s) - \left[G_p(s) - \tilde{G}_p(s) \right] U(s) + d(s) \right] G_c(s)$$

$$U(s) = \frac{\left[R(s) - d(s) \right] G_c(s)}{1 + \left[G_p(s) - \tilde{G}_p(s) \right] G_c(s)}$$

Substitute $U(s)$, into output $Y(s)$ below,

$$Y(s) = G_p(s)U(s) + d(s)$$

to obtain the closed-loop expression below,

$$Y(s) = \frac{\left[R(s) - d(s) \right] G_c(s) G_p(s)}{1 + \left[G_p(s) - \tilde{G}_p(s) \right] G_c(s)} + d(s)$$

$$Y(s) = \frac{G_c(s)G_p(s)R(s) + \left[1 - G_c(s)\tilde{G}_p(s) \right] d(s)}{1 + \left[G_p(s) - \tilde{G}_p(s) \right] G_c(s)} \qquad \textbf{Eqn 5.3-1}$$

If $G_c(s) = \tilde{G}_p(s)^{-1}$ and,

If $G_p(s) = \tilde{G}_p(s)$, then theoretically zero error setpoint tracking and disturbance rejection can both be achieved.

If $G_p(s) \neq \tilde{G}_p(s)$, zero error disturbance rejection can be achieved, provided that

$$G_c(s) = \tilde{G}_p(s)^{-1}.$$

To improve robustness, the effects of mismatch between the process, and process model should be minimised. Since the differences between process and the process model usually occur at the systems high frequency response end, a low-pass filter $G_f(s)$ is usually added to attenuate this effect. Thus IMC is designed using the inverse of the process model in series with a low-pass filter, ie:

$$G_{IMC}(s) = G_c(s)G_f(s)$$

$$G_{IMC}(s) = \tilde{G}_p(s)^{-1}G_f(s) \qquad \textbf{Eqn 5.3-2}$$

(where $\tilde{G}_p(s)^{-1} = G_c(s)$, and where the order of the filter being usually

chosen such that $\tilde{G}_p(s)^{-1}G_f(s)$ is proper (ie the highest numerator power is

always less than the denominator's), to disallow excessive differential control action)

Then substituting **Eqn 5.3-2** into **Eqn 5.3-1** to obtain an expression which includes the $G_{IMC}(s)$ term below:

$$Y(s) = \frac{G_{IMC}(s)G_p(s)R(s) + \left[1 - G_{IMC}(s)\tilde{G}_p(s)\right]d(s)}{1 + \left[G_p(s) - \tilde{G}_p(s)\right]G_{IMC}(s)} \qquad \textbf{Eqn 5.3-3}$$

5.3.1 IMC Controller Design

The process model $\tilde{G}_p(s)$, must first be factored into invertible and non-invertible components, that is:

$$\tilde{G}_p(s) = (invertable) \times (non\_invertable)$$

$$\tilde{G}_p(s) = \tilde{G}_p^{+}(s) \times \tilde{G}_p^{-}(s) \qquad \textbf{Eqn 5.3-4}$$

The non-invertible component $\tilde{G}_p^{-}(s)$, contains terms which if inverted, will lead to instability and realisability problems, ie terms containing positive zeros and time-delays (that were previously not there).

Then using ONLY the invertible component of the process model, let

$$G_c(s) = \tilde{G}_p^{+}(s)^{-1} \qquad \textbf{Eqn 5.3-5}$$

and then substituting into

$$G_{IMC}(s) = G_c(s) \times G_f(s) \text{, then}$$

$$G_{IMC}(s) = \tilde{G}_p^{+}(s)^{-1} \times G_f(s) \qquad \textbf{Eqn 5.3-6}$$

5.3.2 IMC Filter Design

In **Eqn 5.3-2**, $G_f(s)$ is the low-pass filter (of the appropriate order) of the form shown below,

$$G_f(s) = \frac{1}{\left(\tau_f s + 1\right)^n} \qquad \textbf{Eqn 5.3-7}$$

(where τ_f is the filter time constant, and *n* is the order of the filter (and the relative difference between the numerator and denominator, $G_{IMC}(s)$ is said to be proper if *n=1*). As a rule of thumb τ_f, can be chosen to at be at least twice as fast (up to 20 times as fast) as the open-loop response [8], but another method of choosing τ_f is using the following formulae:

$$\tau_f \geq \left[\lim_{s \to \infty} \frac{D(s)N(0)}{20s^n D(0)N(s)} \right]^{\frac{1}{n}}$$

where $D(s)$ and $N(s)$ are taken from the following equation:

$$\tilde{G}_p(s) = \tilde{G}_p^{+}(s) \times \tilde{G}_p^{-}(s) \quad \text{where} \quad \tilde{G}_p^{+}(s) = \frac{N(s)}{D(s)}$$

5.3.3 IMC Time Delay Compensation

Examining the output of the close-loop system by letting $G_p(s) = \tilde{G}_p(s)$, then substituting **Eqn 5.3-6** into **Eqn 5.3-3** to get,

$$Y(s) = \left[\tilde{G}_p^{+}(s)^{-1} G_p(s) G_f(s) \right] R(s) + \left[1 - \tilde{G}_p^{+}(s)^{-1} \tilde{G}_p(s) G_f(s) \right] d(s)$$

$$\therefore Y(s) = \left[\tilde{G}_p^{-}(s) G_f(s) \right] R(s) + \left[1 - \tilde{G}_p^{-}(s) G_f(s) \right] d(s)$$

$$\therefore Y(s) = \left[\frac{e^{-Ts}}{\tau_f s + 1} \right] R(s) + \left[1 - \frac{e^{-Ts}}{\tau_f s + 1} \right] d(s)$$

Thus, we can see that the IMC scheme has the following properties:

- it provides time-delay compensation
- the filter can be used to shape both the setpoint tracking and disturbance rejection responses.
- at the steady-state, the controller will give offset free responses.

5.3.4 IMC Sensitivity and Complementary Sensitivity

The Sensitivity function will be used as in [9], to specifically see the consequences of the controller design in IMC, and then briefly compared to the controller in a classical control system.

Since,

$$Sensitivity = \frac{Y(s)}{d(s)} = \frac{E(s)}{R(s) - d(s)}$$

and from Eqn 5.3-3, let

$$\varepsilon(s) = \frac{Y(s)}{d(s)}$$

$$\varepsilon(s) = \frac{Y(s)}{d(s)} = \frac{1 - G_{IMC}(s)\tilde{G}_p(s)}{1 + G_{IMC}(s)\left[G_p(s) - \tilde{G}_p(s)\right]} \qquad \textbf{Eqn 5.3-8}$$

again substituting $G_p(s) = \tilde{G}_p(s)$ into **Eqn 5.3-9** above to get,

$$\varepsilon(s) = 1 - G_{IMC}(s)\tilde{G}_p(s) \qquad \textbf{Eqn 5.3-9}$$

$$\eta(s) = G_{IMC}(s)\tilde{G}_p(s) \qquad \textbf{Eqn 5.3-10}$$

(where $\varepsilon(s)$ is said to be the complementary of $\eta(s)$ (and visa versa))

Eqn 5.3-10 and **Eqn 5.3-11** display an extremely important theoretical point for the IMC strategy, and that being the controller $G_{IMC}(s)$, appears as a linear controller in both equations. Contrast this for a moment with the corresponding controller $G_c(s)$, in both the sensitivity, and complementary sensitivity functions, for the classical closed-loop control system below,

$$\hat{\varepsilon}(s) = \frac{1}{1 + G_C(s)G_P(s)}$$

$$\hat{\eta}(s) = \frac{G_C(s)G_P(s)}{1 + G_C(s)G_P(s)}$$

where the controller $G_c(s)$, can have differential action on the system. Whilst the sensitivity function determines performance, and the complementary sensitivity function determines robustness (used by IMC), this implies that (compared to the classical scheme) IMC provides a much simpler design to obtain a robust control system.

chapter 6

IMC Temperature Control for the Fruit Drying Plant

6.1   Introduction

The theory of IMC control introduced in the previous chapter shall now be implemented to control the temperature of the fruit drying plant.   The fruit drying plant was especially suited for IMC control since:

- The system was open-loop stable (open loop response plot shown in chapter 3).

- IMC provided time delay compensation since there existed a delay in the process transfer function (between the time the heat from the heater banks is applied, to the time the thermostat in the drying chamber sensed this applied change).

- IMC provided 'robustness' for the controller to provide an offset free response at the steady state temperature of the plant (the drying of fruit is measured in minutes and hours not seconds, so we were after accurate temperature set-point tracking and disturbance rejection, and did not require a system with 'high' sensitivity, or with a 'fast' response time).

6.2   Numerical IMC controller design

The design of the IMC controller implemented the theory from the previous chapter by determining the following:

1. a process model that was an exact representation of the process.

2. the inverse of the process model.

3. the filter parameter.

An exact process model representation was simply that of the open-loop process:

$$\tilde{G}_p(s) = G_p(s)$$

$$\therefore \tilde{G}_p(s) = \frac{6e^{-Ts}}{5(10s+1)}$$

Then using only the invertible term of the process model (ie ignore the time delay term) below:

$$G_c(s) = \left[\tilde{G}_p^{\,+}(s) \right]^{-1}$$

$$\therefore G_c(s) = \frac{5(10s+1)}{6}$$

This was then substituted into the IMC controller below:

$$G_{IMC}(s) = G_c(s)G_f(s)$$

$$\therefore G_{IMC}(s) = \left[\tilde{G}_p^{\,+}(s) \right]^{-1} G_f(s) \qquad\qquad \textbf{Eqn 6.2-1}$$

$$\therefore G_{IMC}(s) = \left[\frac{5(10s+1)}{6} \right] G_f(s)$$

6.2.1   Numerical IMC filter design.

Before we can determine $G_f(s)$ we first must determine τ_f :

$$\tau_f \geq \left[\lim_{s \to \infty} \frac{D(s)N(0)}{20s^n D(0)N(s)} \right]^{\frac{1}{n}}$$

$$\geq \left[\lim_{s \to \infty} \frac{5(10s+1) \times 6}{20s^1 \times 5 \times 6} \right]^{\frac{1}{1}}$$

$$\geq \lim_{s \to \infty} \frac{10s+1}{20s}$$

$$\geq \lim_{s \to \infty} 0.5 + \frac{1}{20s}$$

$$\tau_f \geq 0.5$$

This value of $\tau_f = 0.5$ was then substituted into $G_f(s)$ below

$$G_f(s) = \frac{1}{\left(\tau_f s + 1\right)^n}$$

$$\therefore G_f(s) = \frac{1}{\left(0.5s + 1\right)}$$

Eqn 6.2-2

Therefore substituting Eqn 6.2-1 into Eqn 6.1-1 we obtained the IMC controller to be:

$$G_{IMC}(s) = \left[\frac{5(10s+1)}{6} \right] \left[\frac{1}{0.5s+1} \right]$$

and since $G_{IMC}(s)$ has been determined independent of transport delay, this can now be implemented in the model below:
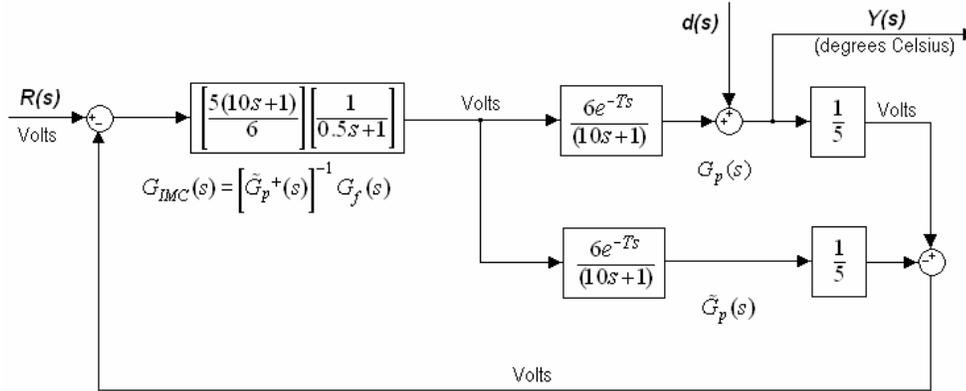
Figure 6-1 IMC system implementation ($\tau_f = 0.5$).

It is one of the most important attributes of IMC design, that once the process model inverse, and the filter has been determined, that the controller is complete. Since the controller has been determined independent of delay terms, the IMC controlled system can be designed around any delay value.   For example, the system was designed for a transport delay of 12 seconds but the step response to any delay value displays the same response except translated by the delay value (see figure below).
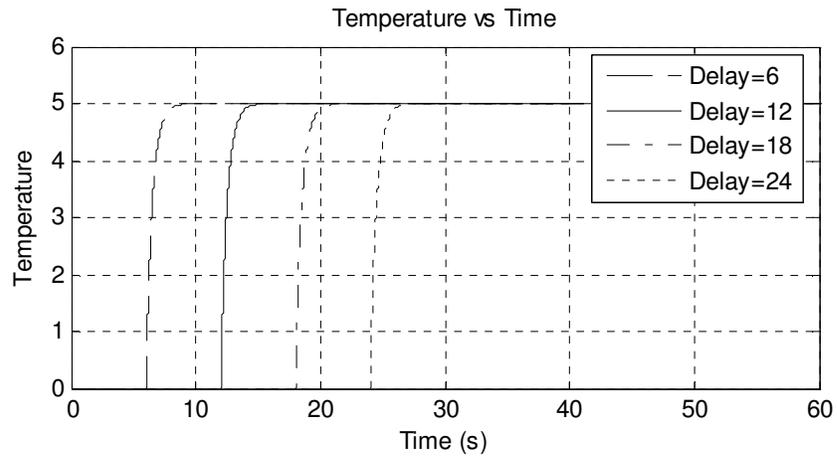
Figure 6-2 Response of system to a unit step input ($\tau_f = 0.5$).
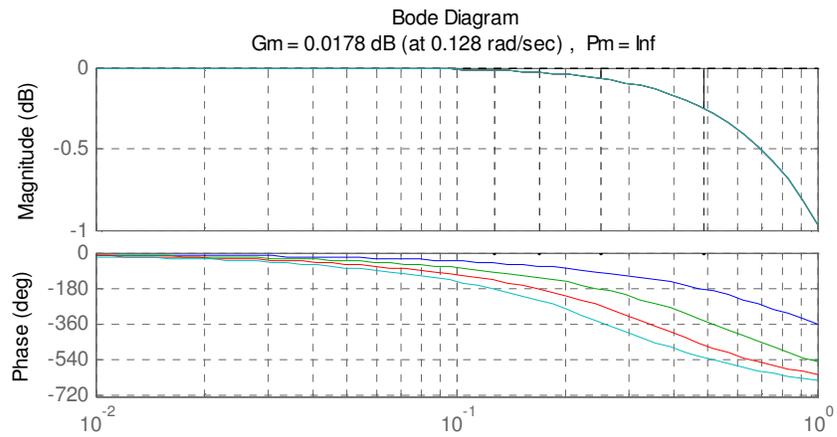
Figure 6-3 Bode response to the above range of delays ($\tau_f = 0.5$).

Within the accuracy of our pade approximation for delay in the transport delay transfer function the phase margin shall remain infinite (thus stable) for any delay

value.

6.2.2 Numerical IMC considering different delays.

Since the delay term e^{-12s} was omitted from the derivation of the controller $G_{IMC}(s)$ (when plant transfer functions of the form of our $G_p(s)$ are designed for), the controller for our system with transport-delay of 6 seconds e^{-6s} (for drying apricots) would have exactly the same $G_{IMC}(s)$. In fact we can set the delay to whatever value we choose.

This allows us to simply say 'set' the delay for 6 seconds (for drying apricots) or 'reset' the delay for 12 seconds (for drying spices), or any delay value in between, depending on the fruit we are drying. (In practise this set/reset process would be automatically conducted by an instrument to monitor V_{air} to provide us these delay values). Recall our linear model for transport delay is simply $T = \dfrac{D}{V_{air}}$.

If we make changes to the supply ducting, or our distance D metres changes between heater-banks and drying chamber, that cause changes in transport delay, our controller remains essentially the same (assuming the plant remains the same of course).

6.3 Implementing the IMC controller for the fruit dryer.

Although we could have used this calculated minimum value of $\tau_f = 0.5$ (which would provide the system with the fastest response time without instability), a value of $\tau_f = 1.5$ was chosen to in fact increase the time to reach setpoint temperature in the un-delayed system from approximately 2.5 seconds to approximately 7.5 seconds.

The reason for this is to prevent more than 10% overshoot to occur when a plant model mismatch of transport delay occurs. When this mismatch did occur, it was

assumed that the time-constant of the instrument monitoring V_{air} was less than 0.05 second, (allowing for the mismatch to be compensated for well within 0.5 second).

This selection of τ_f was justified, based on the rule of thumb where "τ_f can be chosen to be *at least* twice as fast as the open-loop response" [8]. According to this rule the time constant of the open-loop system is 10 seconds therefore $\tau_f < (10/2)$.

$G_f(s)$ (acting as a low pass filter) with its amended value of τ_f, is shown below:

$$G_f(s) = \frac{1}{(1.5s+1)}$$ **Eqn 6.3-1**

Therefore substituting Eqn 6.2-1 into Eqn 6.1-1 we obtained the amended IMC controller which was implemented in our fruit dryer system:

$$G_{IMC}(s) = \left[\frac{5(10s+1)}{6}\right]\left[\frac{1}{1.5s+1}\right]$$

This can now be implemented in the fruit dryer model below:
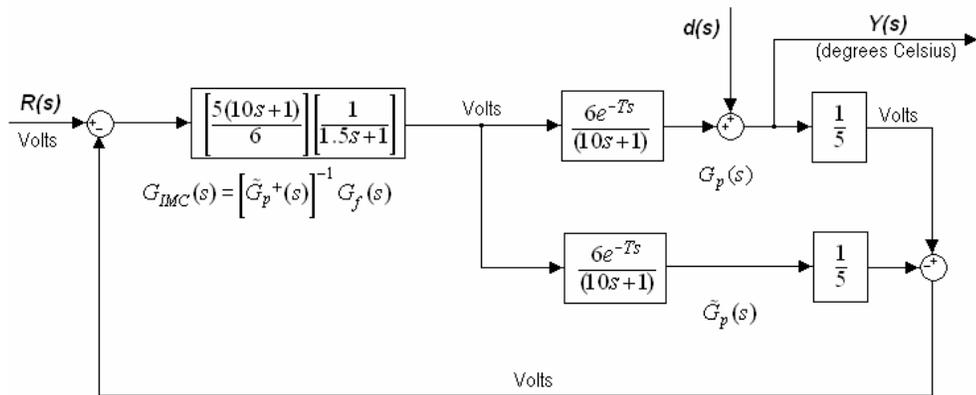
Figure 6-4 IMC system implementation for the fruit dryer ($\tau_f = 1.5$).

6.4 Simulation of the IMC fruit dryer system

A simulation of this system (with T=12 seconds) using this controller was firstly conducted with no plant model matches, and no disturbances, that is with $d(s) = 0$, and as per IMC theory, it was determined to achieve a near ideal response. The response of the above system to a unit step input is shown below:
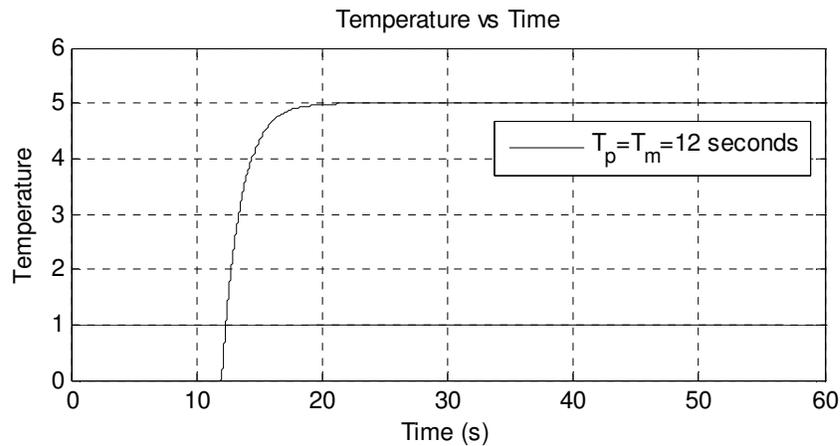
Figure 6-5 Response of IMC system for the fruit dryer with no plant/model mismatches or disturbances ($\tau_f = 1.5$).

But further investigation was conducted about the IMC design to see whether it can be robust enough to resist plant model mismatch and other external disturbances. And, still reach a steady state, offset free response within 60 seconds. To fully address each disturbance individually, the following plant model mismatch/disturbance simulations were independently conducted for the T=12 second system (note: a different transport-delay value apart from T=12 seconds could have been chosen instead for this analysis).

1. A plant/model mismatch of transport delays.

2. A plant/model mismatch of time-constants.

3. Unit step disturbance at the plant output.

4. Pulse disturbance in at the plant output.

5. White noise disturbance at the plant output.

6.5 IMC Plant model mismatch of transport delays

In our fruit dryer, once our fan speed has been set for spices (3 m/s) the transport-delay is then 12 seconds. This fan speed would be kept as constant as possible by a separate controller to the temperature controller, but forms an integral part or the transport delay value used in the plant model.

To realistically consider changes in the value of V_{air} it was assumed that the maximum transport-delay time tolerance which could occur in this system without compensatory action occurring was 0.4 seconds. So when we simulated a plant model mismatch of transport-delays, the maximum difference between the plant transport-delay and the plant model transport delay allowed was 0.4 seconds.

To simulate a change in value of the plant transport delay, T_p was decreased from $T_p = 12$ to $T_p = 11.6$ whilst T_m remained at $T_m = 12$, the system model and its response is shown below.
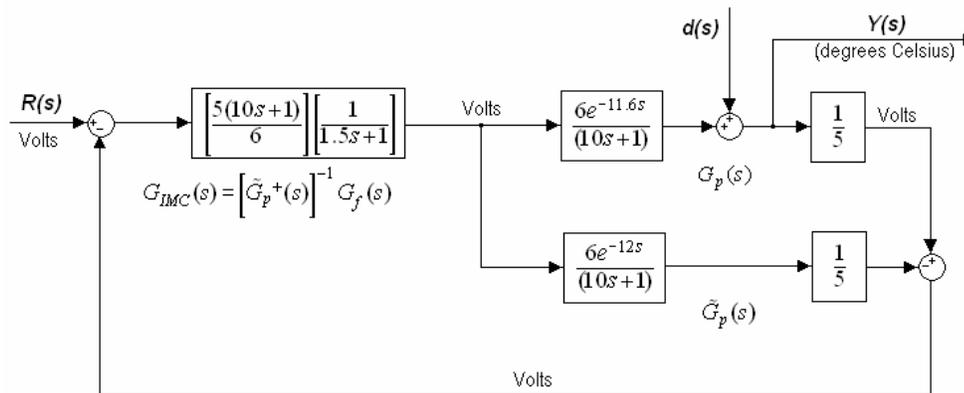
Figure 6-6 Mismatch of transport-delays ($T_p = 11.6$, $T_m = 12$).
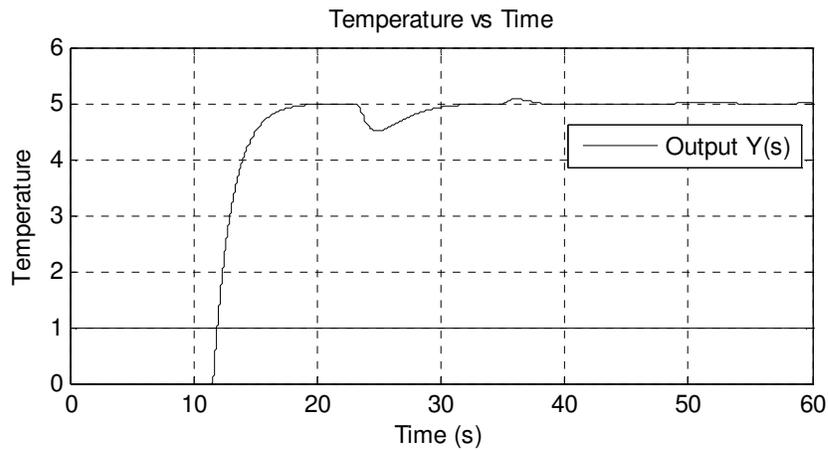
Figure 6-7 Response before compensating ($T_p = 11.6$, $T_m = 12$).

To compensate for this plant model mismatch, T_m must be made to equal T_p again to adhere to the IMC philosophy of requiring that the plant model must be an exact representation of the plant. Below is shown the system model and response after this compensatory action has occurred.
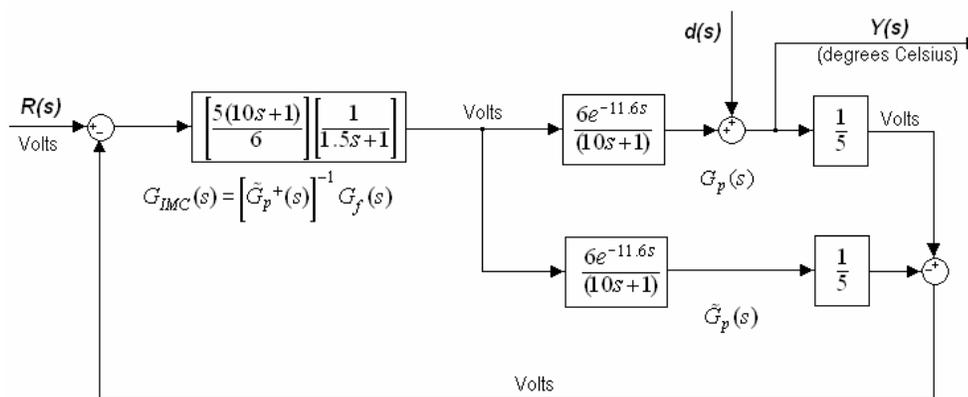
Figure 6-8 Compensated system for a mismatch of transport-delays
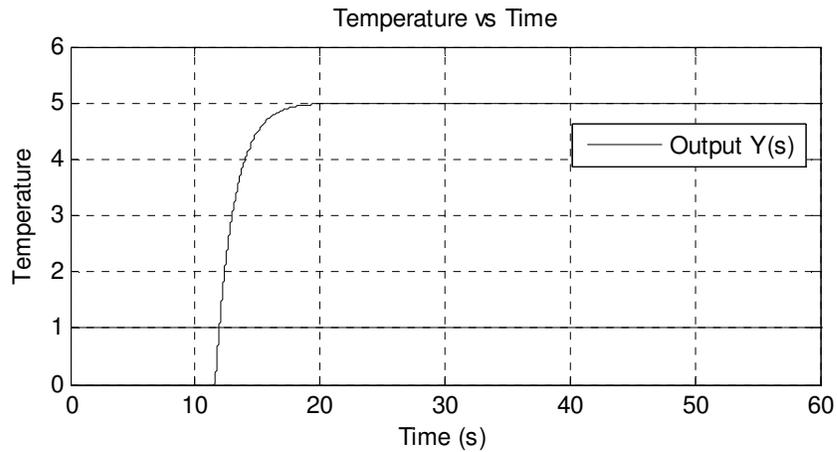
($T_p = 11.6$, $T_m = 11.6$).

Figure 6-9 Compensated response for a mismatch of transport delays

$(T_p = 11.6, \ T_m = 11.6)$**.**

A system whereby T_p drifted in value above T_m could have also been examined, except the uncompensated system would display overshoot instead of undershoot (as in the figure above), but the compensation strategy would be the same.

We could have also simulated a similar mismatch of plant and plant model transport-delays for any plant transport-delay value, and the controller remains the same, thus output responses remain the same except translated along the time axis.

It is for these reasons that systems requiring robustness to changes in transport delay capabilities are usually where IMC controllers find their most suitable applications.

6.6 IMC - Plant model mismatch of time-constants

During the lifetime of a plant (ie over the longer term) its time-constant may drift in

value. Or at some stage there may be a physical alteration made to the plant that may cause a sudden change in plant's time-constant that needs to be identified and compensated for. To illustrate this compensation a plant model mismatch of the time-constants where the plant time-constant increased from 10 seconds (original system) to 12 seconds is shown below:

Figure 6-10 Plant, plant model mismatch of time-constants

$$(\tau_p = 12, \ \tau_m = 10).$$

Figure 6-11 Response before compensating, ($\tau_f = 1.5$, $\tau_m = 10$).

As before to determine $G_f(s)$ we first must determine the minimum allowable τ_f,

again using:

$$\tau_f \geq \left[\lim_{s \to \infty} \frac{D(s)N(0)}{20s^n D(0)N(s)} \right]^{\frac{1}{n}}$$

$$\geq \left[\lim_{s \to \infty} \frac{5(12s+1) \times 6}{20s^1 \times 5 \times 6} \right]^{\frac{1}{1}}$$

$$\geq \lim_{s \to \infty} \frac{(12s+1)}{20s}$$

$$\geq \lim_{s \to \infty} \frac{6}{10} + \frac{1}{20s}$$

$$\tau_f \geq 0.6$$

Since we are already using $\tau_f = 1.5$ our controller requires no change. After the model was adjusted to the plant's time-constant, perfect control was achieved below.
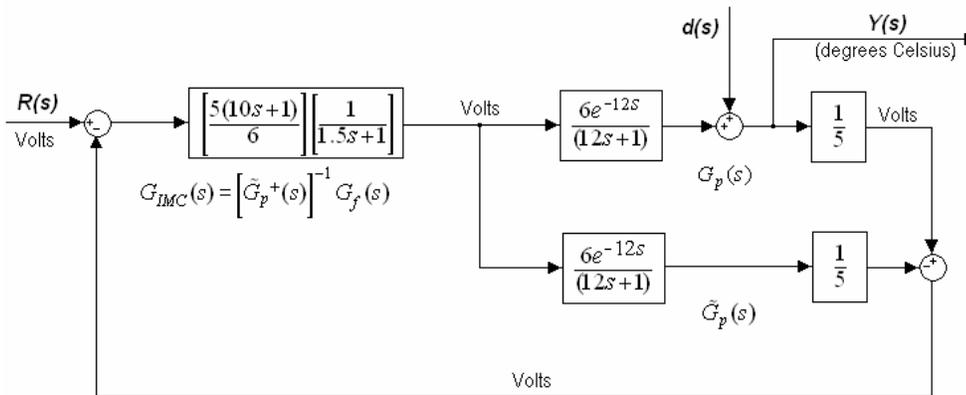
Figure 6-12 Compensated plant/model mismatch of time-constants

$(\tau_p = 12, \ \tau_m = 12)$.
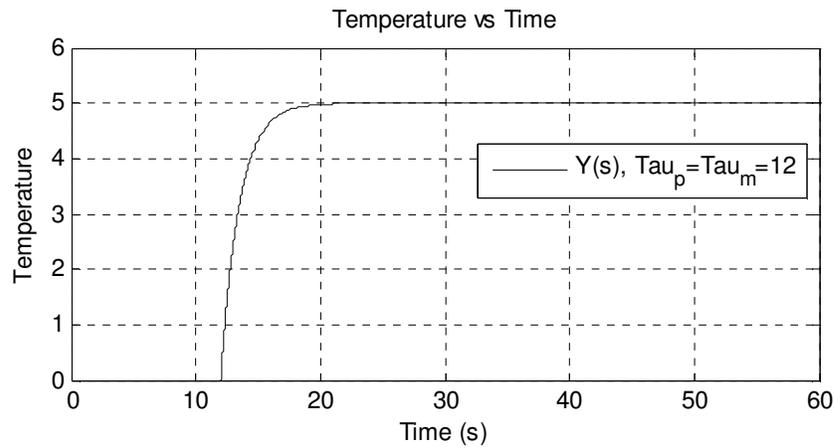
Figure 6-13 Response of compensated system ($\tau_p = 12$, $\tau_m = 12$).

6.7   IMC – Unit step disturbance at the plant output
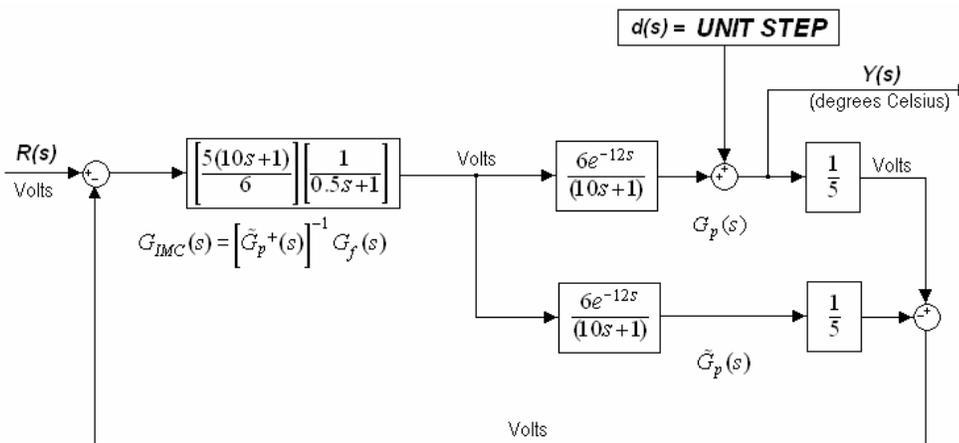
Figure 6-14 IMC model for a unit step disturbance.

To simulate a sustained external disturbance at the output of the plant a unit step input was applied at t=30 seconds (arbitrarily chosen).   It was seen that the step was sustained at the output until t=42 seconds due to the delay, at which time the

controller forces the system to reach the steady state at approximately t=45 seconds.   This system could be regarded as robust since it displays no instability, and excellent set-point tracking to a step disturbance.

Figure 6-15 IMC response to a unit step disturbance.

6.8   IMC – Unit pulse disturbance at the plant output
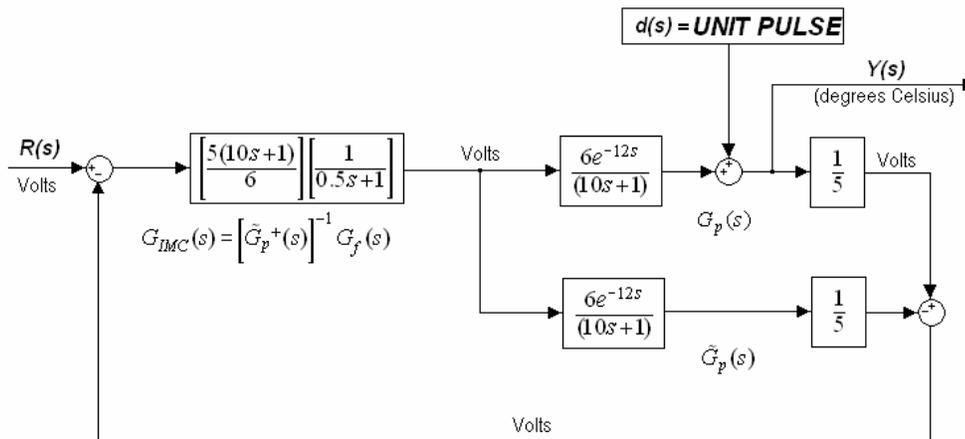
Figure 6-16 IMC model to a unit pulse disturbance.

A unit pulse disturbance (of 1 second duration) was applied at t=30 seconds with

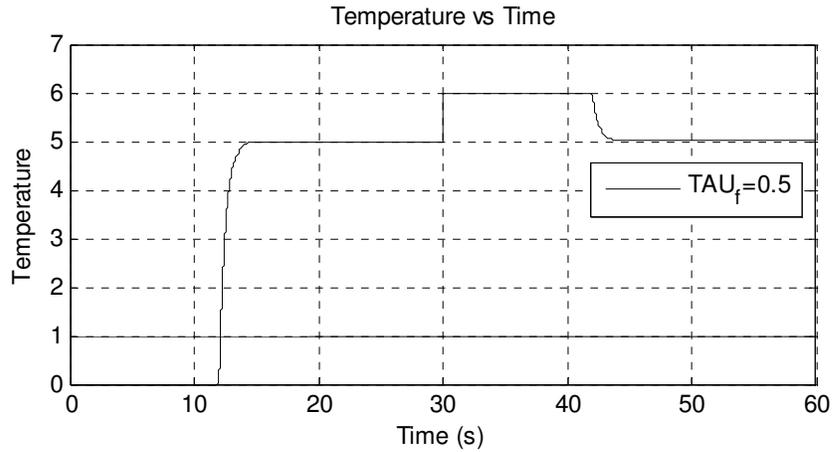the output response shown below. Again the effects of the delay disallow the control action until t=42 seconds at which time the difference between the plant and model is fed back to the set point summing junction to further allow for the controllers action. By t=45 seconds the system has returned to the set point, and again displays all the attributes of a robust system:
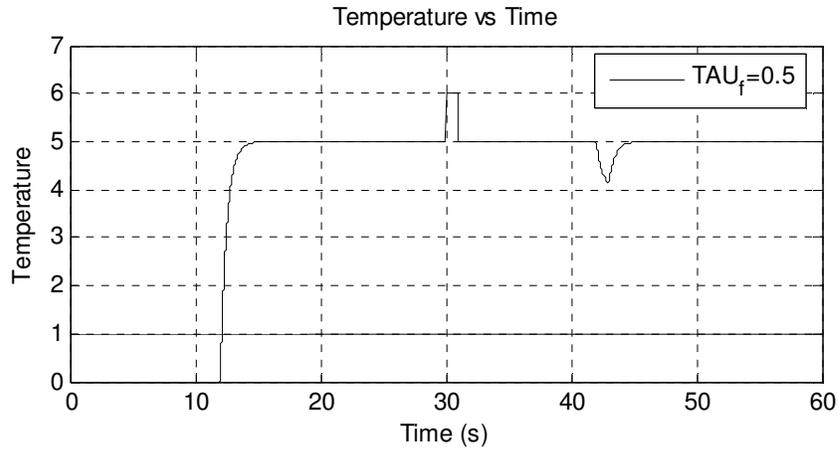
Figure 6-17 IMC response to a pulse disturbance.

6.9   IMC – Band limited white noise disturbance at the plant output.

The same white noise source as was used for the PID white noise disturbance source shall be used here. It comprises a sinusoid with variable amplitude and frequency (the low-pass filter cut off frequency was 2 rad/s (0.32 Hz)).
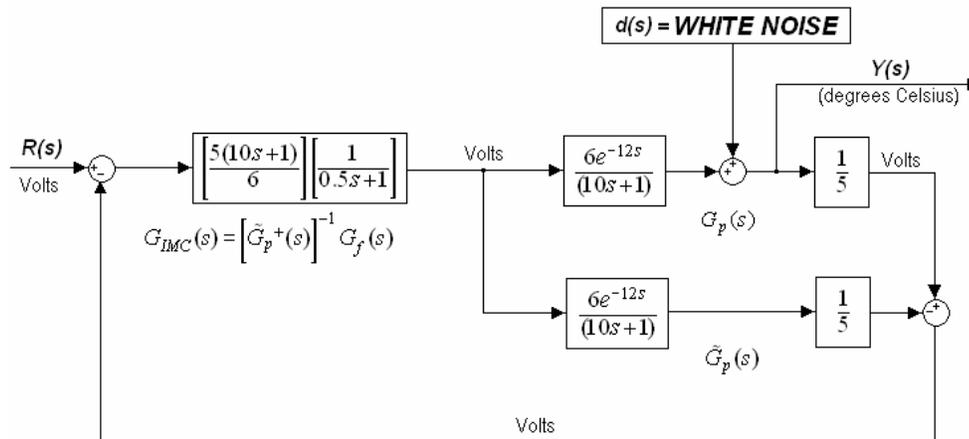
Figure 6-18 IMC model for a band limited white noise disturbance.

The response of the system is shown below, it closely follows the noise (since disturbance applied to the output of the plant), there is no instability, and oscillates about the set point.
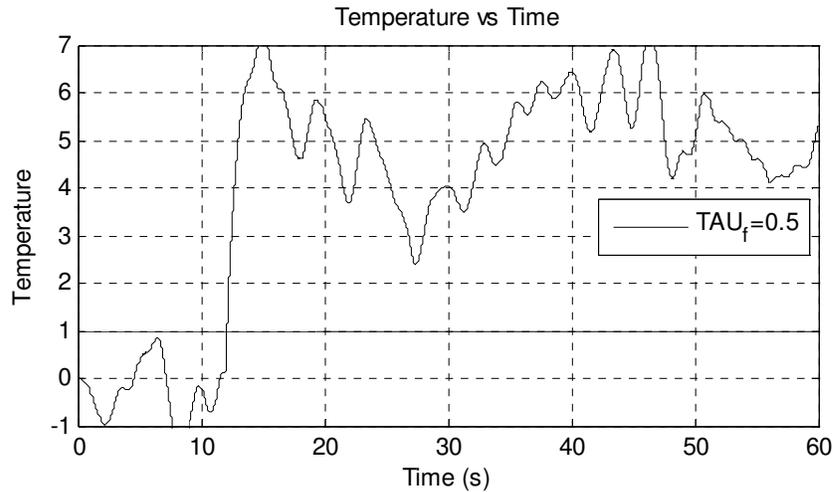
Figure 6-19 IMC response to a band limited white noise disturbance source.

6.10 Conclusion

This chapter implemented IMC theory to the temperature controller of our fruit drying plant. The controller was tested to accurately follow the setpoint firstly (basic requirement), and provide time delay compensation (as the transfer function for this process has a time delay term). It was shown also to fulfil the requirement of providing offset free responses at steady state (even during all tested disturbances at the output of the plant branch).

chapter 7

PID/IMC Evaluation and Comparison

7.1   Introduction

The theory, design, implementation, and simulation of both the PID and IMC control systems were detailed in previous chapters. There performance was *individually* compared against specifications required by the system.   But a performance comparison *between* the two systems is now performed.

7.2   Performance comparison criteria

The methods of design, implementation, and then the responses of the PID and IMC models are compared.

7.3   PID and IMC comparison

7.3.1   PID controlled system.

The PID controller was designed with the intention of being stable and robust. For systems with large delays PID controllers have not traditionally been the best choice, and this was evident in the initial stages of design.   It is not simply for systems with *large* delays, but systems with *varying* delays (ie a range of transport delay values) that the PID controller fell short of meeting specifications.

To design the PID controller for this plant it was required to obtain the optimized PID parameters for one transport delay value.   These PID parameters were unique to the PID controller at this particular transport-delay value.   To develop a PID controlled system that was a function of PID parameters and transport delay

values, to varying transport-delay values would be a formidable task.

The PID controller by virtue of its error integrator limited the higher frequencies in the system. This control action kept the system stable but also limited the ability of the error differentiator to rapidly bring the system to set-point. For systems with even larger transport delay values this would pose an even bigger problem than our simulated system.

In this instance also we were fortunate, the PID controller parameters provided us with a system that was stable and robust for T=12 seconds, since we had a comfortable phase margin of approximately 60 degrees. Had this not been the case, a PID compensator would have needed to be designed similar to [4] (p.708), but again, this would be effective for only one particular transport delay value.

The PID system's performance could be considered more robust when considering its response to large variations in the plant time-constant values, and when applying disturbances. But these attributes are well documented for PID systems modelled with 'small' delays in the plant, and their responses (relative to the original system of T=12 seconds) were relatively unchanged at the output. Although the low pass integrator again in the system dictated a slower time for disturbances to reach setpoint (ie after the delay time of 12 seconds had elapsed).

7.3.2   IMC controlled system.

The IMC controller provided us with the time delay compensation that the PID could not. Not only for a fixed transport-delay but for any delay value we chose. It was a system that provided stability and robustness for not only *large* delay values in the plant, but for *varying* delays (ie a range of transport delay values).

Since the IMC controller was determined independently of the transport-delay, it

provided a system whose sensitivity was a linear function of the transport-delay value and the filter time constant.

When we adjusted the filter time constant, we did so in the knowledge that its ability to adjust our set point tracking would work for any value of transport-delay. When we intentionally increased the filter time-constant from 0.5 to 1.5 to allow the attenuation of higher frequency components during disturbance rejection, this did not cause instability, or oscillatory behaviour. Although there was of course a longer time to reach set-point again during these disturbance simulations. Unlike the PID controller where this low-pass frequency oscillation in the system affected the system's ability to reach set point within 60 seconds (This oscillatory behaviour in the PID controlled system becomes even more pronounced if we chose larger delay values).

During all simulations of the IMC controlled system an offset free response was achieved at the steady state. The system was extremely sensitive to plant/plant model mismatches. The theory states that the plant model must be an exact representation of the plant, but when conducting a simulation of this mismatch, a certain tolerance was allowed for and the system still provided an offset free response at the steady state. This would be a condition of the IMC controller's robustness that its tolerance bounds be especially considered in the design process. We needed to establish that the sensitivity of the air-speed sensor providing the transport delay information, provided this information for any value of transport delay, and be well below the systems sensitivity to the transport delay value.

Disturbances simulated displayed no instability. But unlike the PID system, the IMC had a faster response time to reach set-point because of our choice of filter time-constant. The filter time-constant also acted as a low-pass filter during the white noise disturbance simulation, displaying a lower frequency in the response.

But this filtering did not cause oscillatory behaviour in the system or instability.

chapter 8

Conclusions and Future Work

8.1 Conclusions

The IMC design philosophy allowed us to implement an internal model of the plant, which ran in 'parallel' with the plant. Our choice of filter parameter could be chosen independently of the transport-delay value, and it was this filter parameter that provided the robustness of the system.

For a plant such as our fruit drying system, the filter parameter in the IMC controller was the only parameter requiring tuning.

8.2 Future Work

For our model it was assumed that a drying regime of controlling temperature alone would successfully dry the fruit payload, and for fruits requiring a high drying temperature this is an accurate model. But for many fruits (including spices) requiring drying temperatures below 55 degrees Celsius, it is the accurate monitoring and control of relative humidity that needs to be included in the model (especially in high humidity environments/countries). It is at these lower specified drying temperatures that applying temperature (and air-flow) only, cannot remove saturated air in the drying chamber.

References

[1]   Aigner, P., & Phythian, M., & Wen, P., & Black, J., Computer Controlled Systems, University Southern Queensland, Toowoomba, 2004.

[2]   Bahram, S. & Hassul, M., Control System Design, Prentice-Hall, NJ, 1993.

[3]   Black, J., & Pax, R., & Phythian, M., Linear Systems & Control , University Southern Queensland, Toowoomba, 2000.

[4]   Dorf, Richard C., & Bishop, Robert H., Modern Control Systems (Eighth Edition).

[5]   Fliess, M. & Join, C. & Sira-Ramirez, H., Complex Continuous Nonlinear Systems, http://www.stix.polytechnique.fr/publications/fliess/SYSID06_3.pdf, 2006.

[6]   Garcia, C. E. and Morari, M., 1982, 'Internal Model Control 1. Unifying Review and Some New Results', Industrial Engineering Chemical Process Design and Development, vol. 21 [Online].   Available: http://pubs.acs.org.ezproxy.usq.edu.au/cgi-bin/archive.cgi/iepdaw/1982/21/i02/pdf/i200017a016.pdf [Accessed 18th April 2006].

[7]   Hall, C. W., Drying and Storage of Agricultural Crops, AVI Publishing Company, Westport, Connecticut, 1980.

[8]   Morari, M., and Zafiriou, F., Robust Process Control. Prentice-Hall, NJ. 1989.

[9]   Muing, T Tham, Internal Model Control (Introduction to Robust Control) http://lorien.ncl.ac.uk/ming/robust/imc.pdf, 2002.

[10]   Ricardo S. Sanchez-Pena, & Mario Sznaier, Robust Systems Theory and Applications, John Wiley & Sons, Inc., Canada, 1998.

[11] Ziegler, J.G, Nichols, N.B., "Optimum settings for automatic controllers", Trans.ASME, vol. 64, pp. 19, 1942.

Appendix A

FOR: Scott A. Geddes

TOPIC: Internal Model Control (IMC) of a Fruit Drying System.

SUPERVISORS: Dr. Paul Wen & Dr. Tony Ahfock

ENROLMENT: ENG4112, Semester 2, 2006

PROJECT AIM: The aim of this project is to design an Internal Model Controller to control the air temperature of a fruit drying system (dehydrator). A PID controller is also to be designed for the system, and a performance comparison based on time-delay compensation, robustness and noise immunity criteria be conducted between both systems.

SPONSORSHIP: USQ Faculty Engineering & Surveying

PROGRAMME:

1. Research the theory of Internal Model Control.
2. Research fruit dryers, and drying (dehydration).
3. Design and simulate the PID control system.
4. Design and simulate the IMC control system.
5. Performance comparison between the PID and IMC systems.
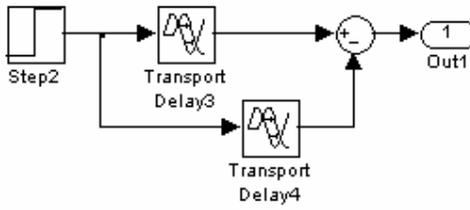
AGREED: _____ (student)   _____ (Supervisor/s)
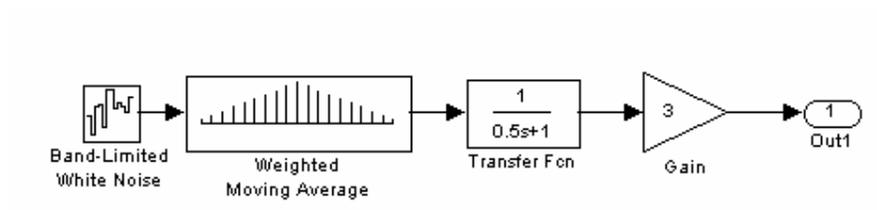
Appendix B

Simulink® Models and Matlab® source code.
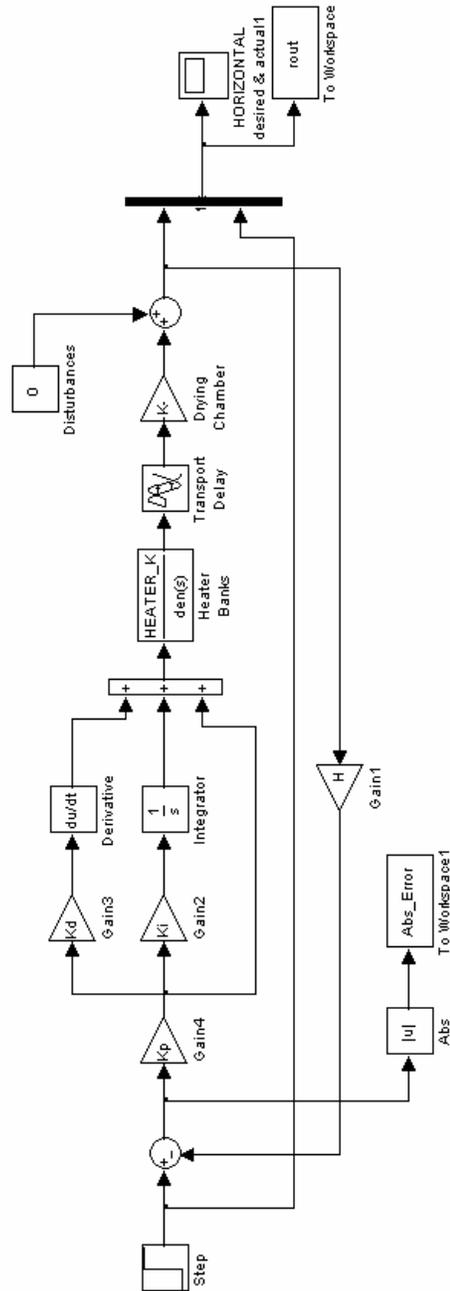
Model-1 Simulink® model of PID control system simulation.

Model-2 Simulink® model of step and pulse disturbance sources.

Model-3 Simulink® model of white-noise disturbance source.

```
clear; home;
% Kp= 1.0; Ki= 0.1;  Kd= 0.0;   %start with these values

%dK=1e-5, IAE_initial =  1.0e+004 * 2.2246    2.2246
Kp =    0.6170
Ki =    0.0649
Kd =    0.1796

HEATER_K=30;
HEATER_TAU=10; % Plant time constant
DELAY=12; %Transport Delay
PLANT_K=0.2; %Plant Transfer Function Gain
H=1./5;

dK=1e-5
step=dK.*10;
IAE_initial=[0];
flag=1;
n=1;
IAE_floor=inf;
while(flag==1);
    [t,x,y]=sim('PID_Disturbance_None_.mdl',[],[],[]);
    IAE=sum(Abs_Error,1);
    IAE_initial(n)=IAE;

    %dS/Kp
    Kp=Kp+dK;
    [t,x,y]=sim('PID_Disturbance_None_.mdl',[],[],[]);
    IAE=sum(Abs_Error,1);
    dSdKp=(IAE-IAE_initial(n))./dK;
    Kp=Kp-dK;

    %dS/Ki
    Ki=Ki+dK;
    [t,x,y]=sim('PID_Disturbance_None_.mdl',[],[],[]);
    IAE=sum(Abs_Error,1);
    dSdKi=(IAE-IAE_initial(n))./dK;
    Ki=Ki-dK;

    %dS/Kd
    Kd=Kd+dK;
    [t,x,y]=sim('PID_Disturbance_None_.mdl',[],[],[]);
    IAE=sum(Abs_Error,1);
    dSdKd=(IAE-IAE_initial(n))./dK;
    Kd=Kd-dK;

    norm=sqrt(dSdKp.^2+dSdKi.^2+dSdKd.^2); %calculate norm
```

```matlab
    Kp=Kp-(step./norm).*dSdKp;  %prepare variables for next iteration
    Ki=Ki-(step./norm).*dSdKi;
    Kd=Kd-(step./norm).*dSdKd;

    if(n>=3);
        if(IAE_initial(n)<= IAE_floor );
            IAE_floor=IAE_initial(n);
        else
            flag=0;
        end

        if( ( (IAE_initial(n)+IAE_initial(n-1))...
                >= (IAE_initial(n-1)+IAE_initial(n-2)) )...
                ||(n==1000) );
            flag=0;
        end
    end
    IAE_initial(n)
    n=n+1
end

IAE_initial
Kp=Kp+(step./norm).*dSdKp
Ki=Ki+(step./norm).*dSdKi
Kd=Kd+(step./norm).*dSdKd
[t,x,y]=sim('PID_Disturbance_None_.mdl',[],[],[]);
plot(t, rout)
grid on;
axis([0 120 0 6]);
title('Temperature vs Time');
xlabel('Time (s)');
ylabel('Temperature');
```

Code-1 Matlab® source code for PID steepest decent method.

```
clear; home;
T=12; %Transport Delay
Kp =    0.6170
Ki = Kp*0.0649
Kd = Kp*0.1796

HEATER_K=30;
HEATER_TAU=10; % Plant time constant
%DELAY=3; %Transport Delay
PLANT_K=0.2; %Plant Transfer Function Gain
H=1./5;

Gc=tf([Kd Kp Ki], [1 0]);
Gp=tf(HEATER_K.*PLANT_K, [HEATER_TAU 1]);

[num_delay, den_delay]=pade(T, 2) % order >= 1st!!!
Gd=tf(num_delay, den_delay);

GcGp=series(Gc, Gp);
GcGpGd=series(GcGp, Gd);
GcGpH=series(GcGp, H); % mag(Gd)=0, so not used here

freq=logspace(-2, 0); %rad/sec
[mag, phase]=bode(GcGpH, freq); % output NOT db!!!
[mag_delay, phase_delay]=bode(Gd, freq); % output NOT db!!!
mag=mag(1,:); %NOT db!!!
%mag_db = 20*log10(mag);
phase=phase(1,:);
phase_delay=phase_delay(1,:);
%phase_total=phase-((180/pi)*T*freq); % phase is in degrees
phase_total=phase+phase_delay; % phase is in degrees

margin(mag,phase_total,freq)

G_closed=feedback(GcGpGd, H, -1);
[num, den]=tfdata(G_closed, 'v');
figure(2);
t=(0:0.001:120);
[y, x ,t]=step(num, den, t);
plot(t, y);
grid on;
axis([-10 120 -1 6]);
title('Temperature vs Time');
xlabel('Time (s)');
ylabel('Temperature');
```
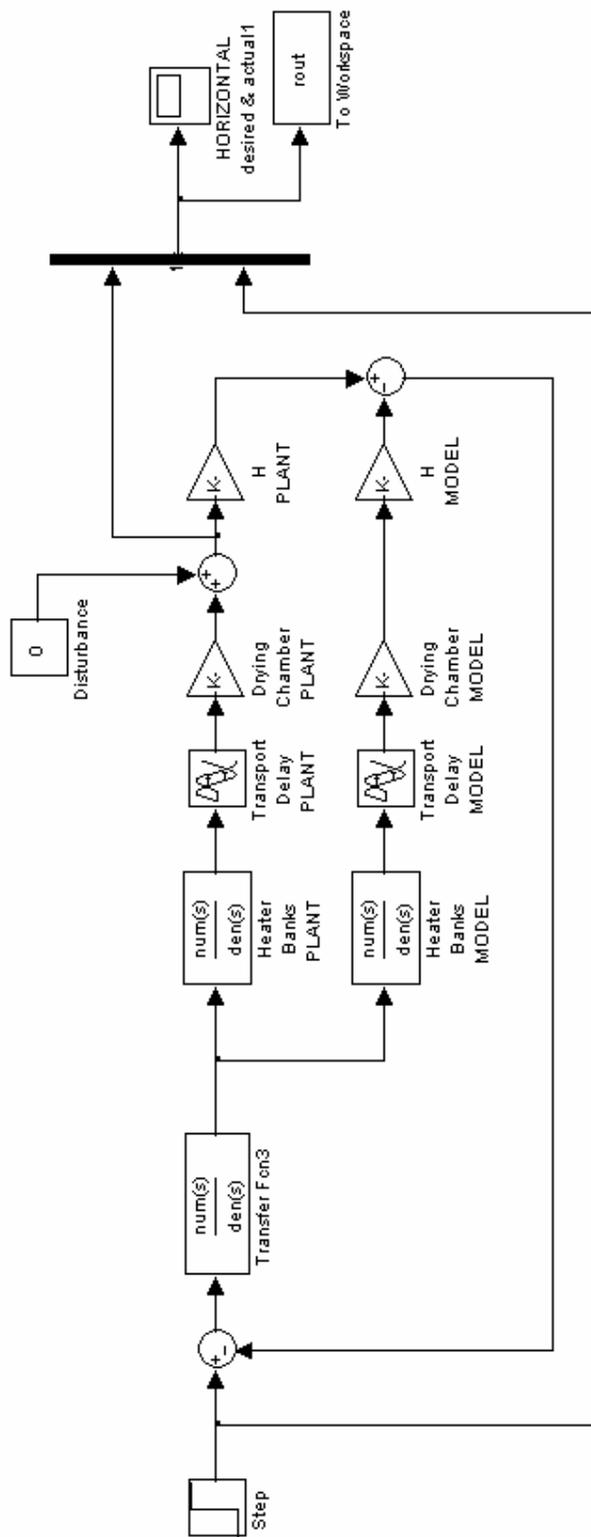
Code-2 Matlab® source code for PID bode diagram (Figure 4-11)

Model-4 Simulink® model of IMC control system.

```
clear; home;

PLANT_HEATER_K=30; % Heater time constant
MODEL_HEATER_K=30; % Model time constant

PLANT_HEATER_TAU=10; % Heater time constant
MODEL_HEATER_TAU=10; % Model time constant

FILTER_TAU=1.5;
%FILTER_TAU=0.5;

PLANT_DELAY=12; %Plant Transport Delay
MODEL_DELAY=12; %Model Transport Delay

PLANT_K=0.2; %Plant Gain
MODEL_K=0.2; %Model Gain

TOTAL_PLANT_K=PLANT_HEATER_K.*PLANT_K; %Plant Gain
TOTAL_MODEL_K=MODEL_HEATER_K.*MODEL_K; %Model Gain

PLANT_H=1./5; %Plant Thermostat
MODEL_H=1./5; %Model Thermostat

[t,x,y]=sim('no_disturbance_.mdl',[],[],[]);

plot(t, rout, '-k');
legend('Output Y(s)');

grid on;
axis([0 60 0 6]);
title('Temperature vs Time');
xlabel('Time (s)');
ylabel('Temperature');
```

Code-3 Matlab® source code for IMC model.

```
clear; home;
PLANT_HEATER_K=30; % Heater time constant
MODEL_HEATER_K=30; % Model time constant
PLANT_HEATER_TAU=10; % Heater time constant
MODEL_HEATER_TAU=10; % Model time constant

FILTER_TAU=0.5;

PLANT_DELAY=12; %Plant Transport Delay
MODEL_DELAY=12; %Model Transport Delay
PLANT_K=0.2; %Plant Gain
MODEL_K=0.2; %Model Gain
TOTAL_PLANT_K=PLANT_HEATER_K.*PLANT_K; %Plant Gain
TOTAL_MODEL_K=MODEL_HEATER_K.*MODEL_K; %Model Gain
PLANT_H=1./5; %Plant Thermostat
MODEL_H=1./5; %Model Thermostat

Gf=tf(1, [FILTER_TAU 1]);
%
for(PLANT_DELAY=6:6:24);
    [num_delay, den_delay]=pade(PLANT_DELAY, 2); % order >= 1st!!!
    Gd=tf(num_delay, den_delay);

    GfGd=series(Gf, Gd);

    freq=logspace(-2, 0); %rad/sec
    [mag, phase]=bode(Gf, freq); % output NOT db!!!
    [mag_delay, phase_delay]=bode(Gd, freq); % output NOT db!!!
    mag=mag(1,:); %NOT db!!!
    phase=phase(1,:);
    phase_delay=phase_delay(1,:);
    phase_total=phase+phase_delay; % phase is in degrees
    %phase_total=phase;

    margin(mag,phase_total,freq);
    hold on
end
grid on;
```

Code-4 Matlab® source code for IMC bode diagram (Figure 6-3)