

Instrumentation and Control Assignment 1

by Ralph Whittaker

FILE	24559_RALPH_WHITTAKER_INSTRUMENTATION_AND_CONTROL_ASSI GNMENT_1_101701_1642943876.DOCX (3.22M)		
TIME SUBMITTED	06-JUN-2017 01:16AM (UTC+0100)	WORD COUNT	3654
SUBMISSION ID	73634490	CHARACTER COUNT	21096

Instrumentation and Control Assignment 1

by

RALPH WHITTAKER

UWTSD

STUDENT ID: P148322

TEST RANGE ENGINEERING

MODULE: Instrumentation and Control

M5X00940

LECTURER: DR STEPHEN MOSEY

UNIVERSITY OF WALES

JUNE 2017

TABLE OF CONTENTS

SECTION	PAGE
TITLE PAGE	1
TABLE OF CONTENTS	2
ABSTRACT	3-4
INTRODUCTION	4-7
METHOD	7-12
RESULTS	13
DISCUSSION	13-14
CONCLUSION	14
REFERENCES	15
BIBLIOGRAPHY	16

Abstract

The capability to autonomously control a process plays a vital role in all modern day computerised industrial processing. Using the transfer function of a process, a control system can be designed and built to incorporate many different specifications and variations of control for many different scenarios across all the sectors of industry. With the use of a proportional-integral-derivative (PID) controller, a common control loop feedback system that is often used in industrial processes, it can be proven that an automated computerised controller can provide an efficient technique for process control.

The following report firstly proves how a controller can be designed, built and tuned to control a given industrial process within specified tolerances (**TASK 1**). It then progresses to prove how a control system can be used to find the combined closed loop transfer function of an entire system (the industrial process and the control system). To achieve this, the control system is simplified by combining a PID controller and a transfer function into a single process block within the feedback loop (**TASK 2**). Thus, the newly found closed loop transfer function of the whole system can be used instead of individual process components, so long as the mathematical operations and relationships between the controller, plant and feedback are accurately defined within. Lastly, evidence will be presented on how the control system can be finalised and concentrated even further into a single block controller, incorporating the feedback, using the working methodology of a single transfer function block for the entire process (**TASK 3**). The control system is proven and validated by comparing the results provided from simulations of the single transfer function and the simulation results from Task 1 and Task 2.

The document begins with the initial presented problem of being tasked to design a proportional control system for a stepped input industrial process, using the given transfer function $G(s)$ of the system:

$$G(s) = \frac{1}{as^2 + bs + c}$$

The transfer function of a system is a mathematical representation of the behaviour of the control system, describing the relationship between the inputs and outputs of a system. Using the transfer function of a process, control systems can be tailored to the users' needs, making a process even more efficient and cost effective, especially when no human interaction is necessary. Once a controller has been designed by a user and applied to a process, an entire processing system can be left to work alone. This is widely exploited within the manufacturing industry, where machines can build and package an entire final product along an automated assembly line.

The task required the process to be under control within the given specification requirements of a **2 second time limit** and with **less than a 5% overshoot** tolerance. Using the computer software package MATLAB and Simulink, a control system was designed to demonstrate the capabilities of controlling the entire process autonomously.

A Pole-zero diagram and a Nyquist diagram will be used to critically analyse the designed control methods and compare the results of a final individual transfer function block to that of the results provided by the initial controller comprised of the combination of the individual control components within a feedback loop mechanism. The results displayed on a Simulink scope will act as evidence and will prove that when accurately designed, an applied single block (combined) transfer function for a process can be equally as efficient as that of a multiple block component controller.



Introduction

In industry, it is very important to be able to control the behaviour of certain automatic systems for which human interaction is neither present or able to carry out the process adjustments within the required time frame or to desired accuracy. It proves beneficial to use computerised controllers to control processes that need controlling over long and short durations, especially processes that require immediate response, complex adjustments or those that are within dangerous working environments.

Figure 1 below represents the typical components of a process with a *PID controller and a feedback control loop system*:

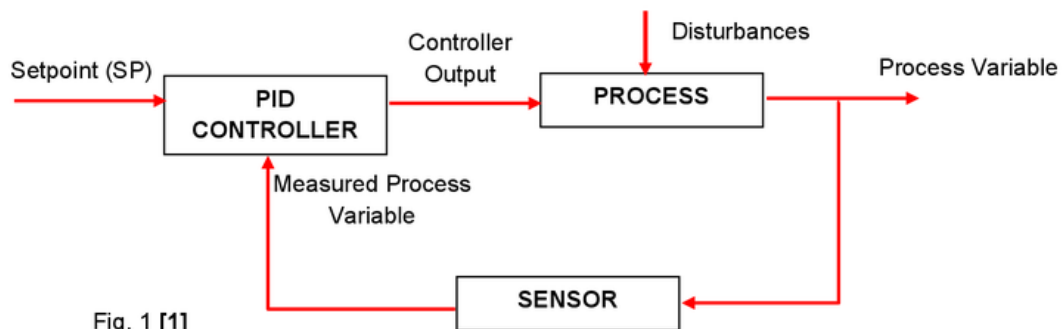


Fig. 1 [1]

Sensors are used to measure the output performance of a system (Process Variable) and use these measurements to provide feedback to the controller. The controller then makes the necessary corrective decisions and produces an output based on its comparisons (algorithms) with a predetermined setpoint. The setpoint is the specific value setting within the controller that is aimed for by that of the process (aka plant). The plant then performs these corrections based on the instructions from the controller. If the controller calculates that the PV is within tolerance of the SP then the controller outputs a zero and the process plant is left switched off. An unacceptable difference in the SP and PV, results in the controller sending a process signal to the actuator inside the plant which in turn performs the necessary corrective action. When controlling the temperature inside a house, this will either be a heating or cooling process, depending on whether the PV is lower or higher than the SP and the boiler will be turned on or off. Disturbances are eventualities that affect the process plant, for example a window being opened in a house [2] with a central heating system will affect how hard the boiler must work (harder) to heat the house on a cold day.

Another example of a controlled system is that of a **conditioning** unit (a refrigerator) that automatically and continuously tries to maintain a set temperature point throughout the day during various eventualities such as surrounding air temperature variations or where the door is frequently opened and closed for access to food inside. Human input or correction is not needed since a thermostat controller has been designed to correct the temperature of the unit according to the set point. If the temperature inside rises above the manual set point for which food needs to be maintained safely outside the dangerous zone and prevent it from perishing (usually between 0-4 degrees Celsius), the compressor (plant) is switched on and refrigerant is pumped through the coils. This cools the air inside until the thermostat registers an acceptable steady internal air temperature, as close to the set point as possible. The thermostat at this stage then signals the compressor to switch off by cutting its power. A refrigerator would be deemed inefficient if the user had to stand next to it all day and perform manual temperature control.

Other examples of what control systems are used for include the autopilot systems that are used to automatically control the trajectory of aeroplanes on long haul flights. Another example is the adaptive cruise control [3] used in the latest car technology, that provides additional control to cruise control by also automatically varying the speed of the car based on the distance and speed of traffic in front. These systems automate processes, allowing users to then multitask more easily and concentrate on other manual and visual processes.

The following are examples of different types of commonly used controllers, each formed by selectively turning on the individual components of a PID controller. Each involves calculating the **Error** by subtracting the Process Variable (PV) from the Setpoint (SP):

Proportional Control (P Controller)

Figure 2 below shows a **P only** controller that works by first calculating the **Error** by subtracting the **Process Variable (PV)** from the **Setpoint (SP)** and then multiplying this **Error** by the **Proportional Gain (K_P)** to produce the controller output. The Proportional Gain is tuned to achieve the desired performance of the controller, outputting a response that is proportional to the error signal.

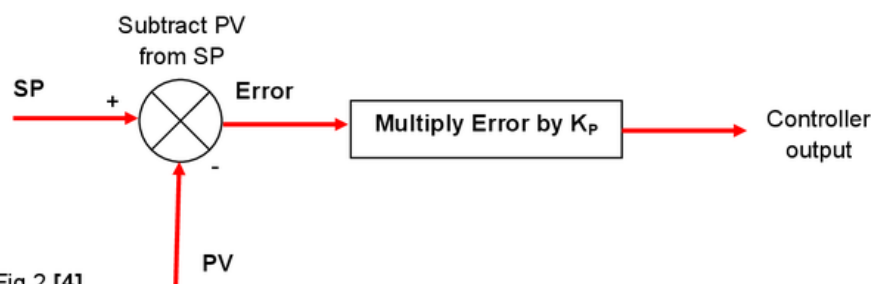


Fig.2 [4]

Proportional Integral Control (PI Controller)

The **Proportional and Integral** controller (Figure 3) involves the addition of two actions. The **Proportional Gain** is individually multiplied by the **Error** and the **Integral (I)** is individually multiplied by the **Error**. The integral involves integrating the error signal throughout the duration of the system run time. The two actions are then **added together** to form the controller output, increasing the affect to accurately match the output to the desired input value. The affect or impact of *I* depends on how long it takes for the Integral action to match that of the Proportional action (minutes per repeat). The lesser amount of time it takes results in larger Integral action.

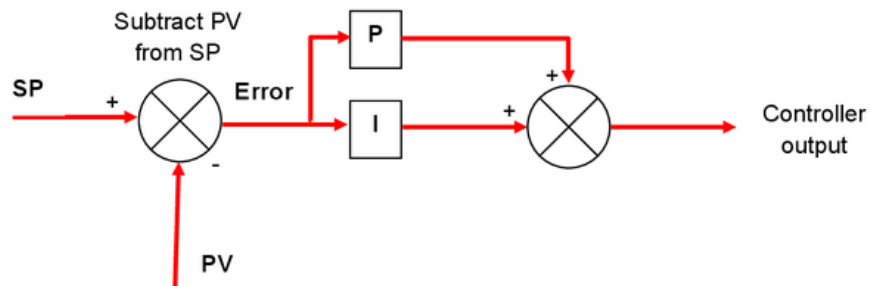


Fig. 3 [5]

Proportional Derivative Control (PD Controller)

A **PD** controller (Figure 4) is the combination of **Proportional Gain** and **Derivative action**. Derivative action can provide a faster response and better loop response by predicting what is going to happen in the future by projecting the current rate of change [6]. It makes use of the rate of change of the output to control the transient response. It therefore makes use of a future predicted value. The **Proportional Gain** is individually multiplied by the **Error** and the **Derivative (D)** is individually multiplied by the **Error**. The two actions are then **added together** to form the controller output. System control can become compromised however if there is a lot of noise (spikes) on the feedback signal channel, as this can falsely represent the sensed output. As a result, derivation action is inappropriately increased and the controller mismanages (overshoots) control of the process by trying to over control or under control the process using the anomalous PV. The control becomes disturbed (ineffective) and therefore a filter is necessary to remove any spikes.

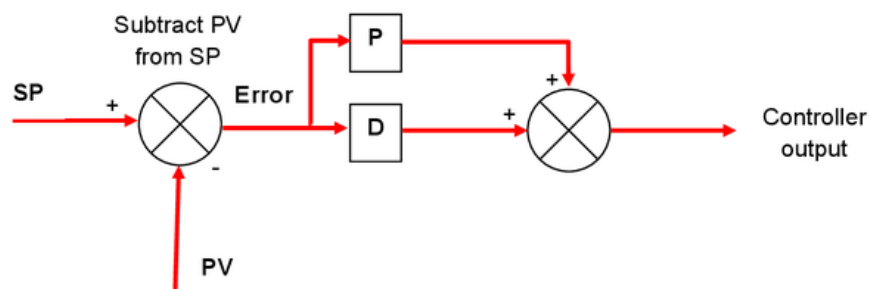


Fig. 4

Proportional control by itself only provides a system with a control state of steadying the error and the output therefore never quite reaches the set point value. The introduction of integral control aids the proportional controller but overshooting is still problematic if the set point is a challenging target. Derivative control can be used to further hone the control by involving the output rate of change into the solution, this has the effect of damping the integral control and improves the control systems response.

Proportional Integral Derivative Control (PID Controller)

Figure 4 below represents a full *PID* control system [1]. This controller represents all 3 components, *P*, *I*, and *D* being turned on and used. Depending on the suitability and performance needed of the controller, a PID controller can be tuned to provide control in many complex processes. With a PID controller, the *Proportional Gain (P)*, *Integral (I)* and *Derivative (D)* are individually multiplied by the *Error*. All three actions are then *added together* to form the controller output signal to the processing plant.

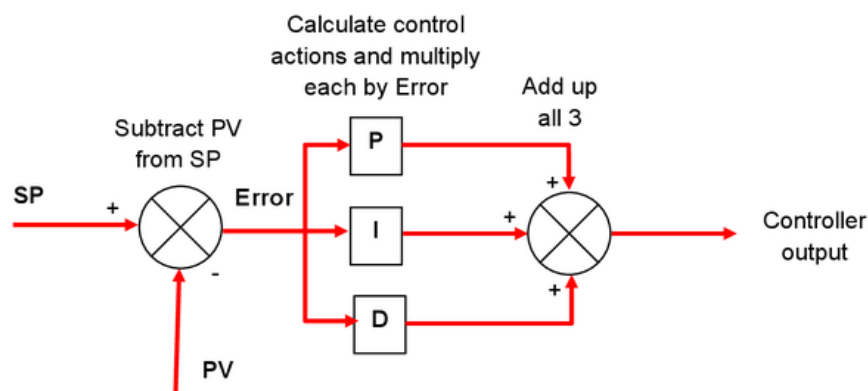


Fig. 4 [7]

The following is the method used for simulating and tuning a PID controller on MATLAB and Simulink for the task set.

Method

-TASK 1

The task began by first creating a variable in MATLAB to store the process's plant transfer function $G(s)$, the unique transfer function a , b , c coefficients were provided as follows: $a = 2$, $b = 1$, $c = 5$

$$G(s) = \frac{1}{2s^2 + 1s + 5}$$

MATALAB command window input: `G=tf([1], [2 1 5])`

```

Command Window

>> G=tf([1], [2 1 5])

G =

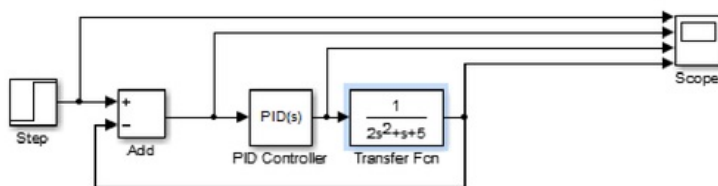
      1
-----
 2 s^2 + s + 5

Continuous-time transfer function.

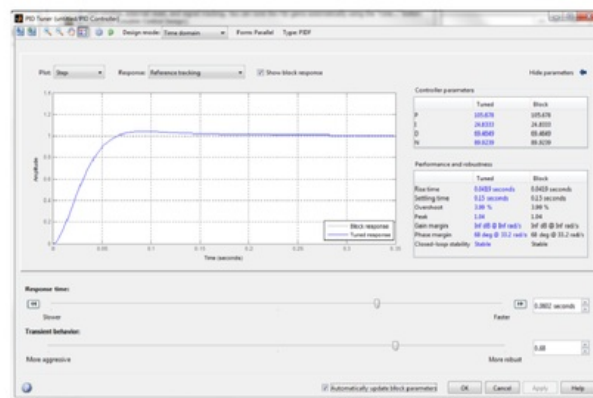
fx >> |

```

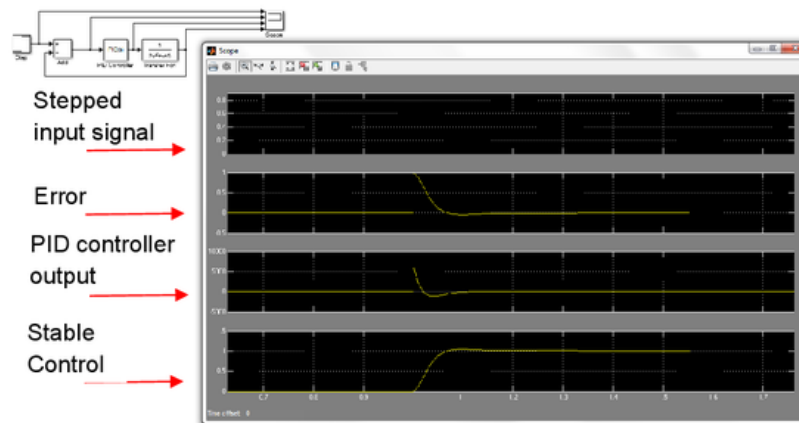
Simulink was then used to design and simulate a PID Controller, a control system that suits the given transfer function of the process.



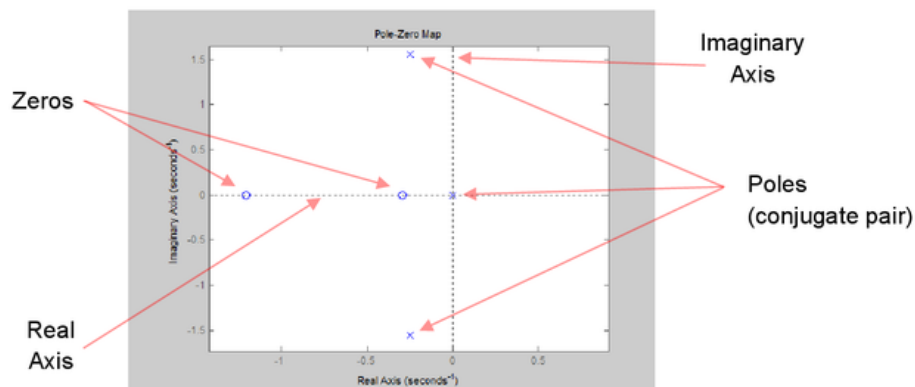
The PID controller block parameters were then tuned according to the design specifications required by increasing and decreasing the response time and transient behaviour sliders in the PID controller Design window. Below shows the block tuned, using the built in PID tuner, to bring the process under control within a settling time of 0.15 seconds and within an overshoot of 3.99%. This fine tuning of the controller, creates a more efficient process, allowing the system to run as close as possible to the planned specifications and fits the requirement of the task. (Max 2 second settling time and Max 5% overshoot). As a result, the closed loop stability status was **Stable** and a **linear plant model** was obtained. The designed controller settings were then exported and used as the initial controller within the process.



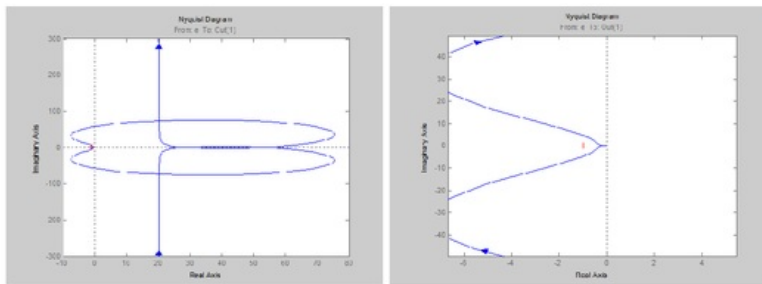
Running the controller and viewing the results on a scope provided evidence that the controller design successfully controlled the process and a linear stability was achieved, as shown below in the bottom window. The systems' input is represented by stepped input over time with a total gain of 1.



To critically analyse the control system, a pole-zero diagram and a Nyquist diagram were created using MATLAB to ensure plot is stable (outside the curve). The following graph shows the results of the pole-zero plot of the transfer function generated by the original (TASK 1) controller set up.



The following graph shows the results of Nyquist diagram of the transfer function generated by the (TASK 1) controller set up, it is useful for evaluating the signals returned and observing the error and phase gain margins.

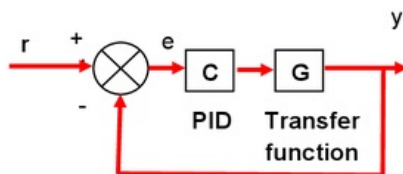


(Magnified)

-TASK 2

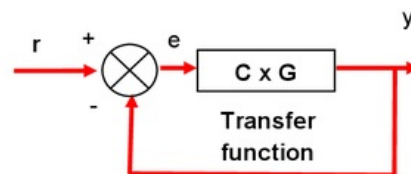
Simulink was used to create a second version of the controller setup, this time excluding the PID controller from the feedback loop system. Here the system uses an operation block that combines the PID controller and transfer function together as a single operation of multiplication.

MATLAB was used to calculate a new transfer function (H) to replace the plant controller by simply combining the PID controller and transfer function together as one operation block as below: (r = stepped input, e = error, y = controlled output)



TASK 1

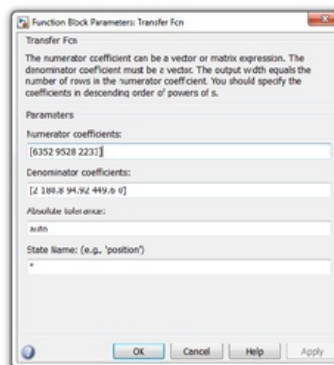
Becomes



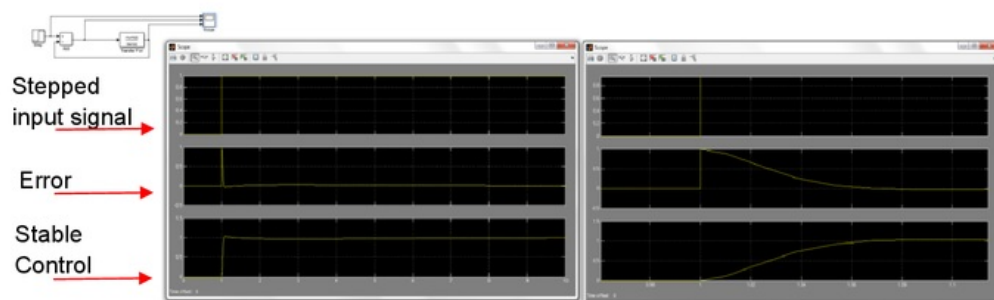
TASK 2 ($H = C \cdot G$)

```
>> H=G*C2
H =
From input "e" to output:
      4352 s^2 + 9528 s + 2233
-----
      2 s^4 + 180.8 s^3 + 94.92 s^2 + 449.6 s
Continuous-time transfer function.
Warning! Using a default value of 0.2 for maximum step size. The simulation step size will be equal to or less than this value.
You can disable this diagnostic by setting 'Automatic solver parameter selection' diagnostic to 'none' in the Diagnostic page of
the configuration parameters dialog
>>
>> pzplot(H)
>> nyquist
```

MATLAB performed the mathematical sums and displayed the new transfer function. The function coefficients calculated by MATLAB were used during setup for the numerator and denominator in the design stages of the single transfer function block. A zero replaces the non-existent coefficients in the numerator of the transfer function block parameters.



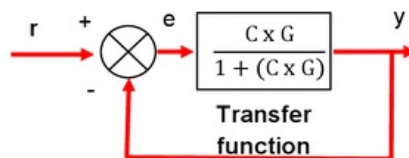
Once defined the system simulation was then ran with the single transfer function block, ready for comparison with the previous scope results.



(Magnified)

-TASK 3

The final task was to incorporate the feedback into the single transfer function block, using Simulink to simulate the model and MATLAB to calculate the new transfer function (H2).



(r = stepped input, e = error, y = controlled output)

Below shows the mathematical results calculated by MATLAB for the numerator and denominators for the design of the third version of the control system consisting of the final concentrated transfer function block (H2).

To remove the feedback H was multiplied by $1/(1+H) = H/(1+H)$

```
>> H2=H/(1+H)

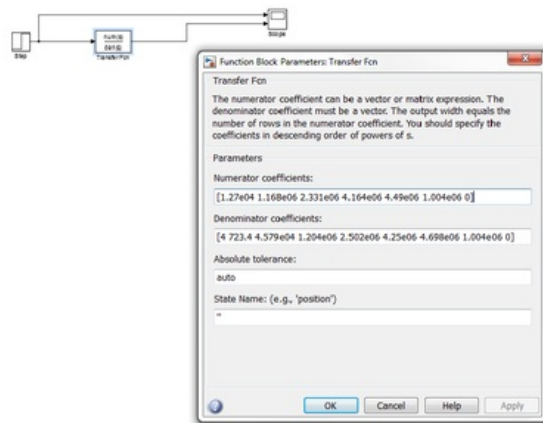
H2 =

      1.27e04 s^6 + 1.168e06 s^5 + 2.331e06 s^4 + 4.164e06 s^3 + 4.496e06 s^2 + 1.004e06 s
-----
      4 s^8 + 723.4 s^7 + 4.579e04 s^6 + 1.204e06 s^5 + 2.502e06 s^4 + 4.25e06 s^3 + 4.698e06 s^2 + 1.004e06 s

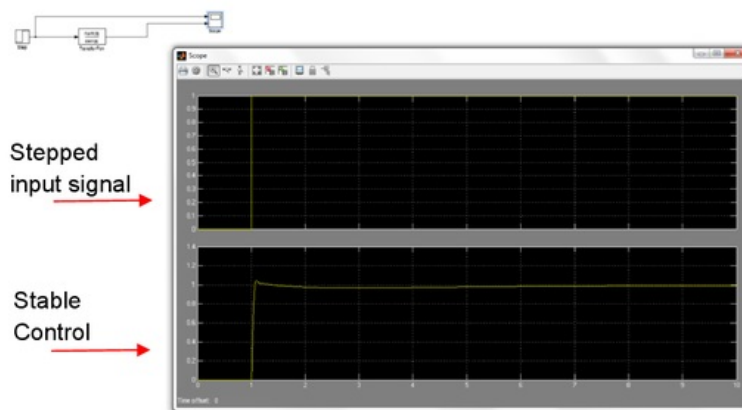
Continuous-time transfer function.

fx >> |
```

Simulink was used to draw the final model of the control system, combining the PID controller and feedback with the transfer function of the industrial process, forming one mathematically combined transfer function block. The new numerator and denominator coefficient values for H2 were entered as the new transfer function parameters.

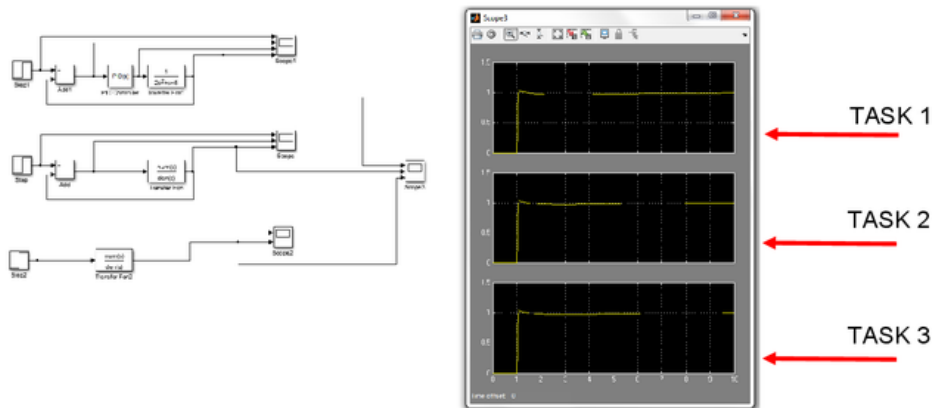


Lastly process control output results of the final combined control system were displayed in Simulink on a scope.



Results

Finally, all results from Task 1, Task 2 and Task 3 were compared in Simulink by connecting the process control outputs into one scope ready for making a conclusion as to whether elements of a control system can be successfully combined in this way.



Discussion

The task has proven the powerful computational ability of using computer software (MATLAB and Simulink) as the method to mathematically design and develop the controller of an industrial process. More importantly the results have proven that the responses of the open loop (no feedback loop) controller system can be created to give the same results as a closed loop control system and can bring the process under control within a settling time of 0.15 seconds and within an overshoot of 3.99%, well within tolerance of the set points of the process's (tasks') specification. There is evidence of slight overshoot at the start and then slight undershoot in all three set ups but all identically settle and become stable with approximately a gain of 1, within the 2 second and 5% tolerance.

The pole-zero plot created in TASK 1 reflected that a stable system had been achieved, where the poles and zeros were on the left-hand side of the map due to the negative complex frequency variable value 's' (approaching zero). The 's' signifies the locations of the poles in relation to the real and imaginary axis, which were for this task correctly plotted on the real x axis as a conjugate pair. This meant enough damping was occurring to stabilise the system. The calculated and plotted poles and zeros of the system provided information on the characteristic terms that compose the response of the controller, showing the effect of control on the system. The plots enabled the control system to be designed with an understanding of how the parameters can be manipulated to obtain a response within the set points [8]. Zeros occur when the value of 's' makes the transfer function zero, when the system is excited in such a way that it does nothing, making the system stop. The diagram reflects how the controller was damping the error signal down to an acceptable stability for the industrial process. Alternatively, *if* the poles and zeros were shown to be on the *right* of the diagram this would mean the system was *unstable*, with positive feedback left over but this was not that case in all eventualities.

The Nyquist's diagram enabled evaluation of the return signal and provided the ability to observe the error (maximum instability). Again, it proved the systems successful stability and allowed interpretation of the phase gain margins, with the red cross positioned outside the curve. Stable systems do not include point '-1' under the curve.

Once the control system was satisfactorily tuned (TASK 1 completed) the function block diagram was then collapsed down into subsequently fewer process blocks. By retaining the correct mathematics within the processes the system was successfully condensed into one single process block.

From looking at the scope displays it is evident that the individual transfer function block performs the same as the individual component system (as in task 1). When comparing all three system set ups, the open loop system (entire PID controller transfer function, plant and feedback) performs the same as the closed loop system with individual components as per designed in the original controller layout. Even though the model equation has been simplified, all the scopes display the same results of control and system performance.

The results are in fact as expected, with the reason being that so long as the mathematical transforms within the blocks are correctly written and processed, the control elements can be combined with no noticeable effect on the control performance of the system. With the aim of providing the industrial process with a gain of 1, the results of all the versions of the control system both open and closed, provided the same accuracy and speed of control.

Conclusion

The findings are evidence that MATLAB and Simulink were a good choice for designing and simulating the behaviours of a computerised automated control system for a stepped input industrial process. Such computer software provides an easier and quicker method for a control system designer to utilise the built in PID tuner and quadratic equation to calculate and plot the complex frequency variable 's' (or system behaviour) than by hand, especially for even more complex controller systems.

Having analysed the results, it has been proven that all the elements of a complex controller can be simplified into a simple block, so long as the mathematical operations and relationships between the controller transfer function, plant and feedback are correctly defined. To conclude this task, it has been proven that a closed loop control system can be transformed into an open loop control system without loss of control performance. The task was successful in controlling and stabilising the industrial process within the specified tolerances.

References

- [1] PID for Dummies, Anatomy Of A Feedback Control System, Jim Hogenson, Control Solutions, Minnesota, 2016, originally by Finn Peacock, Idiots Guide to PID Algorithms, [Online] Available: https://www.csimn.com/CSI_pages/PIDforDummies.html [Accessed: May 4th 2017]
- [2] Denizen Robo, Elements of Process Control, Automation Forum, Automation, Control System, Instrumentation, Process Control, December 12th 2015, [Online] Available: <http://www.automationforum.co/2015/12/elements-of-process-control.html>, [Accessed: May 14th 2017]
- [3] Dean Gibson, Cruise control and adaptive cruise control: the complete guide, Auto Express, 9th January 2017. [Online] Available: <http://www.autoexpress.co.uk/car-tech/98225/cruise-control-and-adaptive-cruise-control-the-complete-guide> [Accessed: May 20th 2017]
- [4] PID for Dummies, Proportional Control, Jim Hogenson, Control Solutions, Minnesota, 2016, originally by Finn Peacock, Idiots Guide to PID Algorithms, [Online] Available: https://www.csimn.com/CSI_pages/PIDforDummies.html [Accessed: May 4th 2017]
- [5] PID for Dummies, A match made in heaven: The P + I Controller, Jim Hogenson, Control Solutions, Minnesota, 2016, originally by Finn Peacock, Idiots Guide to PID Algorithms, [Online] Available: https://www.csimn.com/CSI_pages/PIDforDummies.html [Accessed: May 4th 2017]
- [6] How Do PID Controllers Work: Application and Theory, Data Track Process Instruments, Derivative (rate), published 2014 [Online] Available: <http://www.datatrackpi.com/technical-papers/how-do-pid-controllers-work.html> [Accessed: May 4th 2017]
- [7] PID for Dummies, Under the Hood of The PID Controller, Jim Hogenson, Control Solutions, Minnesota, 2016, originally by Finn Peacock, Idiots Guide to PID Algorithms, [Online] Available: https://www.csimn.com/CSI_pages/PIDforDummies.html [Accessed: May 4th 2017]
- [8] Transfer Functions, Pole and Zeros, G.F. Franklin et al. "Feedback Control of Dynamic Systems", 5th Edition. Upper Saddle River, NJ, 2006, Pearson Education, Inc, D. J. Inman. "Engineering Vibration", 3rd Edition. Upper Saddle River, NJ, 2008, Pearson Education, Inc. [Online] Available: https://www.maplesoft.com/content/EngineeringFundamentals/10/MapleDocument_10/Transfer%20Functions,%20Poles%20and%20Zeros.pdf [Accessed June 1st 2017]

Bibliography

1. 2.14 Analysis and Design of Feedback Control Systems, Presentation by Prof. David Trumper, Publication date Spring 2010, Massachusetts, Institute of Technology, Department of Mechanical Engineering, 77 Massachusetts Avenue, Room 3-173 Cambridge, Massachusetts 02139. Online PDF module paper, (pg. 1-13)
2. Transfer Functions, Pole and Zeros, G.F. Franklin et al. "Feedback Control of Dynamic Systems", 5th Edition. Upper Saddle River, NJ, 2006, Pearson Education, Inc, D. J. Inman. "Engineering Vibration", 3rd Edition. Upper Saddle River, NJ, 2008, Pearson Education, Inc. Online PDF module paper (pg.1, 10-20)
3. PID Theory Explained, Definition of Terminologies, National Instruments, March 29th 2011, [Online website] Available: <http://www.ni.com/white-paper/3782/en/>, [Accessed June 4th 2017]

Instrumentation and Control Assignment 1

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

58/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4



Comment 1

Your abstract started to turn into an introduction about half way through. Good attempt, but you should have included a brief statement of your results.

PAGE 5



Comment 2

Air conditioning?

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13



Comment 3

Should this read: "...correctly plotted on the real x axis, and as a conjugate pair."?

GRADING FORM: SMU GENERIC_GJ

RALPH WHITTAKER**58.30****RESEARCH-INFORMED LITERATURE**

Inclusion of a wide range of research-informed literature, including sources retrieved independently. Selection of relevant and credible sources. Very good use of referencing conventions, consistently applied.

3.25**KNOWLEDGE AND UNDERSTANDING**

Knowledge is accurate with a good understanding of the field of study.

16.5**ANALYSIS**

Sound, logical, analytical thinking; synthesis and evaluation. Ability to devise and sustain persuasive arguments, and to review the reliability, validity & significance of evidence. Ability to communicate ideas and evidence accurately and convincingly. Sound, convincing conclusions / recommendations.

19.8**PRACTICAL APPLICATION AND DEP.**



A good and appropriate application of standard methods, materials, tools and/or techniques. Good appreciation of the context of the application, with some use of examples, where relevant.

13.75

SKILLS FOR PROFESSIONAL PRAC.



Can communicate effectively in a suitable format, but may have minor errors. Mostly coherent work and is in a suitable structure. You started captioning your figures well, but as the report went on you appear to have given up inserting captions.

5