

**APPLICATION AND EVALUATION OF
STATISTICALLY-BASED SIGNAL
FILTERS ON A PILOT SCALE
FLOW LOOP**

By

NAGAPPAN MUTHIAH

BE in Instrumentation and Control

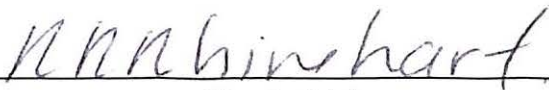
University of Madras, India

2001

Submitted to the Faculty of the Graduate College of the
Oklahoma State University in partial
fulfillment of the requirements for
the Degree of
MASTER OF SCIENCE
July, 2004

**APPLICATION AND EVALUATION OF
STATISTICALLY-BASED SIGNAL
FILTERS ON A PILOT SCALE
FLOW LOOP**

Thesis Approved:



Thesis Advisor







Dean of Graduate College

ACKNOWLEDGEMENTS

I express my sincere gratitude to my advisor, Dr. R. Russell Rhinehart, for providing me with constant encouragement and guidance, during the entire period of this research. He was always willing to spend time with me and help me with any kind of difficulties that I faced.

I thank Dr. James Robert Whiteley for making me realize how interesting the world of Process Control can be! I really enjoyed his courses in Process Control, which laid out a strong foundation for my research work.

I would like to thank my friend Konda Mettu Reddy who helped me with the calibration of the flow transmitter. Special thanks to Mr. Shyam Sunder Uma Chander who clarified the various concepts of filters from an electrical engineer's perspective.

I am also thankful to Mrs. Genny Hasty and Mrs. Eileen Nelson who created a very friendly atmosphere every time I visited the chemical engineering department.

Finally, I would like to thank Edward E. and Helen Turner Bartlett Foundation for partial financial support.

TABLE OF CONTENT

Chapter	Page
1. Introduction	
1.1: Sources of Noise	1
1.2: Effect of Process and Measurement Noise	2
1.3: Ways to Reduce Process Noise	4
1.4: Novel Methods to Filter	5
2. Literature Survey	
2.1: Types of Filtering	7
2.2: Filters for Noise Removal	8
2.3: Analog First-Order Filter	10
2.4: Digital First-Order Filter	10
2.5: Open Loop Response of a First-Order Filter	13
2.6: Metric – Ways to Quantify Lag	14
2.7: Concept of the CUSUM Filter	14
2.8: Working of the CUSUM filter	15
2.9: Open Loop Response of a CUSUM Filter	19
2.10: Different Ways to Place a Filter - Mathematical Analysis	20
2.11: Metrics - Ways to Quantify the Goodness of a Filter in Closed Loop	26
2.12: Filtering in Regulatory Mode – Mathematical Analysis	27
3. Cooling Water Flowrate System - Experimental Setup	
3.1: Description of the Process	31
3.2: Description of Data Acquisition and Control System	34
3.3: Development of Process Model	35
3.4: Design of the PI Controller	36
3.5: Design of Digital PI Controller	37

Chapter	Page
4. Experimental Procedure, Results and Analysis	
4.1: Experimentation Description	39
4.2: Description of Test: Procedure	44
4.3: Servo Mode	48
4.4: Regulatory Mode	59
4.5: Comparison of Results with Paper – “A CUSUM type on-line filter”	61
5. Simulation Setup	
5.1: Flow Loop Process Model in Simulink	63
5.2: Simulink Model for Digital PI Controller	64
5.3: Simulink Model for Calculating ISE and Valve Travel	64
5.4: Simulink Model for CUSUM Filter	71
5.5: Simulink Model for First-Order Filter	74
5.6: Simulink Model for Closed Loop Configurations	77
5.7: Simulink Model for Open Loop Testing	83
5.8: Simulink Model for Disturbance Rejection Closed Loop Testing	84
6. Simulation Results and Analysis	
6.1: Closed Loop Testing – Change in Setpoint	86
6.2: Servo Mode Analysis	86
6.3: Regulatory Mode Analysis	89
6.4: Closed Loop Testing – Change in Disturbance	92
6.5: Disturbance Rejection Analysis	92
6.6: Comparison of Experimental and Simulation Results	93
6.7: Open Loop Testing	95
7. Conclusion and Recommendations	
7.1: Conclusions	99
7.2: Recommendations	100
References	101
Appendixes	
Appendix-A-- Calibration of Flow Transmitter	102
Appendix B -- Tabulation of Experimental Results	107
Appendix C -- Tabulation of Simulation Results	112

LIST OF TABLES

Tables	Page
Table A.1: Values of $\log_{10} \dot{M}$ and $\log_{10}(I - I_o)$ at various control valve positions	105
Table B.1: Experimental - Configuration A1 - First-order filter on PV	108
Table B.2: Experimental - Configuration A2 – CUSUM filter on PV	109
Table B.3: Experimental - Configuration B1 - First-order filter on MV	110
Table B.4: Experimental - Configuration B2 – CUSUM filter on MV	111
Table C.1: Simulation - Configuration A1 - First-order filter on PV	113
Table C.2: Simulation - Configuration A2 – CUSUM filter on PV	113
Table C.3: Simulation - Configuration B1 - First-order filter on MV	114
Table C.4: Simulation - Configuration B2 – CUSUM filter on MV	114

LIST OF FIGURES

Figure	Page
Figure 1.1: Effect of process noise on level control loop	3
Figure 1.2: Configuration A - normal practice of using a filter	5
Figure 1.3: Configuration B - novel method of using a filter	6
Figure 2.1: Flowchart of digital first-order filter	12
Figure 2.2: First-order filter open loop response block diagram	13
Figure 2.3: First-order filter open loop response	13
Figure 2.4: Flow Chart of the CUSUM filter	15
Figure 2.5: Working of the CUSUM filter	17
Figure 2.6: The Code of implementing the CUSUM filter	18
Figure 2.7: CUSUM filter open loop response block diagram	19
Figure 2.8: CUSUM filter open loop response	19
Figure 2.9: Block diagram for filter on PV – servo mode	20
Figure 2.10: Block diagram for filter on MV – servo mode	22
Figure 2.11: Block diagram for filter on PV – regulatory mode	24
Figure 2.12: Block diagram for filter on MV – regulatory mode	25
Figure 2.13: Trend for Case 1 - process noise and zero disturbance	27
Figure 2.14: Trend for Case 2 - disturbances and zero noise	29
Figure 2.15: Summed response due to disturbances and noise in process	30

Figure	Page
Figure 3.1: Distillation column setup	32
Figure 3.2: Cooling water flow control valve and orifice meter	33
Figure 3.3: Cooling water flow rate system	34
Figure 3.4: Snapshots of Control Station FOPTD fitting	35
Figure 4.1: Block diagram for Configuration A1	40
Figure 4.2: Camile GUI for Configuration A1	40
Figure 4.3: Block diagram for Configuration A2	41
Figure 4.4: Camile GUI for Configuration A2	41
Figure 4.5: Block diagram for Configuration B1	42
Figure 4.6: Camile GUI for Configuration B1	43
Figure 4.7: Experimental setup for Configuration B2	43
Figure 4.8: Camile GUI for Configuration B2	44
Figure 4.9: Transient period of controller with CUSUM filter on MV (trigger 1.5)	47
Figure 4.10: SS period of controller with CUSUM filter on MV (trigger 1.5)	47
Figure 4.11: Results of Configuration A and B for servo	49
Figure 4.12: Servo - Configuration A1 highlighted	50
Figure 4.13: Configuration A1 - servo, $\lambda=0.60$	51
Figure 4.14: Configuration A1 - servo, $\lambda=0.775$	52
Figure 4.15: Servo - Configuration A2 highlighted	52
Figure 4.16: Configuration A2 - servo, trigger=1.0	53
Figure 4.17: Configuration A2 - servo, trigger =3.0	54
Figure 4.18: Servo - Configuration B1 highlighted	54

Figure	Page
Figure 4.19: Configuration B1 - servo, $\lambda=0.60$	55
Figure 4.20: Configuration B1 - servo, $\lambda=0.825$	56
Figure 4.21: Servo - Configuration B2 highlighted	56
Figure 4.22: Configuration B2 - servo, trigger =1.0	58
Figure 4.23: Configuration B2 - servo, trigger =3.0	58
Figure 4.24: Regulatory response	59
Figure 5.1: Process model subsystem	63
Figure 5.2: Information window for “Process model subsystem”	63
Figure 5.3: Simulink block for the process model	64
Figure 5.4: Simulink block for a digital PI controller	64
Figure 5.5: ISE Simulink block inside the masked blocks	65
Figure 5.6: ISE subsystem to calculate transient and steady state ISE	66
Figure 5.7: Information window for the “steady state ISE subsystem”	67
Figure 5.8: Information window for the “transient ISE subsystem”	67
Figure 5.9: Valve travel subsystem in Simulink	68
Figure 5.10: Implementing valve travel subsystem in Simulink	69
Figure 5.11: Information window for “steady state valve travel subsystem”	70
Figure 5.12: Information window for “transient valve travel subsystem”	70
Figure 5.13: CUSUM filter subsystem	71
Figure 5.13b: Matlab code -“CUSUM_Code.m”	72
Figure 5.13c: Matlab code – “init_code.m”	73
Figure 5.14: CUSUM filter block	73
Figure 5.15: Information window for CUSUM filter block	74

Figure	Page
Figure 5.16: First-order filter subsystem	75
Figure 5.16a: Matlab code – “FOF_Code.m”	75
Figure 5.17: First-order filter block	76
Figure 5.18: Information window for first-order filter block	76
Figure 5.19: Configuration A1 – First-order filter on PV	77
Figure 5.20: Configuration A2 – CUSUM filter on PV	80
Figure 5.21: Configuration B1 – First-order filter on MV	81
Figure 5.22: Configuration B2 – CUSUM filter on MV	82
Figure 5.23: Simulink setup for open loop testing	83
Figure 5.24: Simulink setup - disturbance rejection – Configuration A1	84
Figure 6.1: Simulation results of Configuration A and B for servo	87
Figure 6.2: Simulation results of Configuration A and B for regulatory	90
Figure 6.3: Simulation results for Config. A and B for disturbance change	93
Figure 6.4: Open loop testing results: sampling time – 0.2 sec	96
Figure 6.5: Open loop testing results: sampling time – 0.4 sec	96
Figure 6.6: Open loop testing results: sampling time – 0.6 sec	97
Figure A.1:Regression Analysis	105

CHAPTER 1

INTRODUCTION

1.1: Sources of Noise

In process control, various kinds of noise are introduced in analog transmission signals at various stages. A brief idea of the different types of noise are discussed this section.

Process Noise: Noise can be intrinsic to the process itself. It can be caused due to turbulence, variations due to mixing or non-uniform multiphase flows [9]. Noise from such sources is usually successive, small and short lived but real transients in process outputs. For example, a noisy signal will be obtained if a float system is used to measure the level of the river. The source of noise will be the waves in the river [1].

Thermal Noise: Another source is the thermal noise which can be introduced in the input circuit of the measurement device. The main reason for thermal noise is the random movement of electrons in materials which cause small temperature dependent currents in the conductor. These noise levels are very low, in the order of micro-volts and are usually ignored [1].

Thermal EMF Noise: If a conductor is made of different materials, then a temperature gradient in that conductor can induce small thermoelectric voltages called Thermal EMF. This noise can be reduced by making sure that temperature gradients don't exist within a measuring device [1].

Other Sources: Audio-phonic noises are induced due to physical vibration of the measurement system. External noises are introduced into a measuring device via sensor or communications wiring. Noise can be due to electric field coupling as well. Noise can also be introduced in the cable connecting the sensor and the measuring device. Further details about sources of noise can be obtained from Reference [1]. In general, the noise can be viewed as an independent random addition to the true signal. Since there are many independent sources, the noise distribution is nearly Gaussian.

1.2: Effect of Process and Measurement Noise

Process and instrument noise degrades process control. Ideally, it is desired that the controller responds to true process changes only, but due to noise, even when the process is in steady state, the controller keeps taking action. When noisy signals are given to the controller, a noisy controller output is obtained. This is undesirable for several reasons. The noisy controller output causes more wear and tear in the final control element. This reduces the life of the final control element which is usually a control valve. Also, since a noisy signal is given to the control valve, this noise is further propagated in the process; and hence, the noise in the process variable is further increased. In order to explain this concept, let's take the example of a simple level controller. The

diagram of the level control loop is shown in Figure 1.1. Let's consider a tank which has an inlet flow and an outlet flow. The outlet flow can be throttled using the flow control valve. The level of the tank is measured by the level transmitter and this signal is given to the level controller. The level controller compares the setpoint and the measured signal and accordingly gives a signal to the control valve.

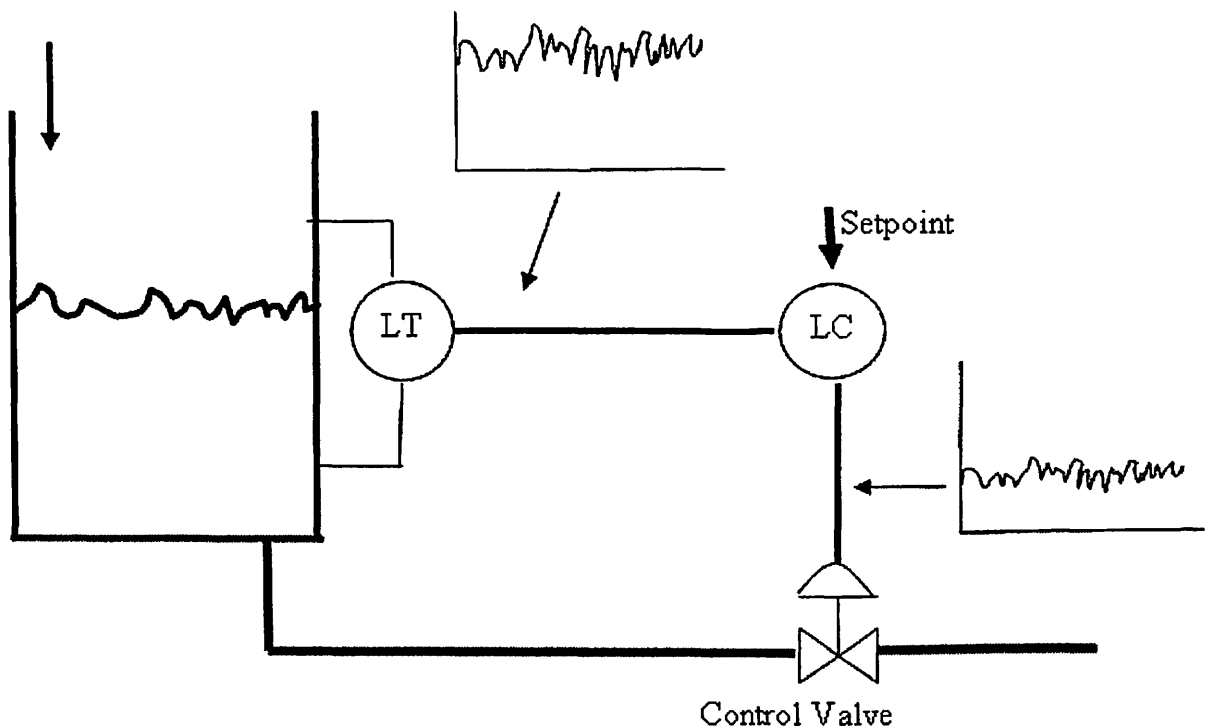


Figure 1.1: Effect of process noise on level control loop

For instance, if a setpoint of 5 ft is given to the level controller and if the level of the tank was 5 ft, ideally, output of the level controller should be constant. Unfortunately that is not the case. Since the level transmitter has process noise in it, noisy data is given to the level controller. As a result, the level controller output is also noisy. Hence, the control valve position keeps changing. This would result in unwanted use of the control

valve even when the process is static, which would lead to more wear and tear. Also, control valve position keeps changing; the level of tank will also keep changing. In this way the variability of the whole process will increase because of the process noise.

1.3: Ways to Reduce Process Noise

Filters are used to reduce the process measurement noise. They seek to identify the true process signal in the presence of noise. The normal practice is to filter the noisy process measurement obtained from the sensor and this filtered process variable is given as an input to the controller. This configuration is shown in Figure 1.2 and is named as Configuration A in this study. A first-order filter is widely used for the above application; however, a first-order filter has an undesirable characteristic that it introduces lag in the control loop. This lag limits the aggressiveness of the controller, which is not desirable. Addition of such time lags introduces time delays which results in reduction of stability margin [2]. When controllers respond to such lagged signal, degradation of controller performance takes place.

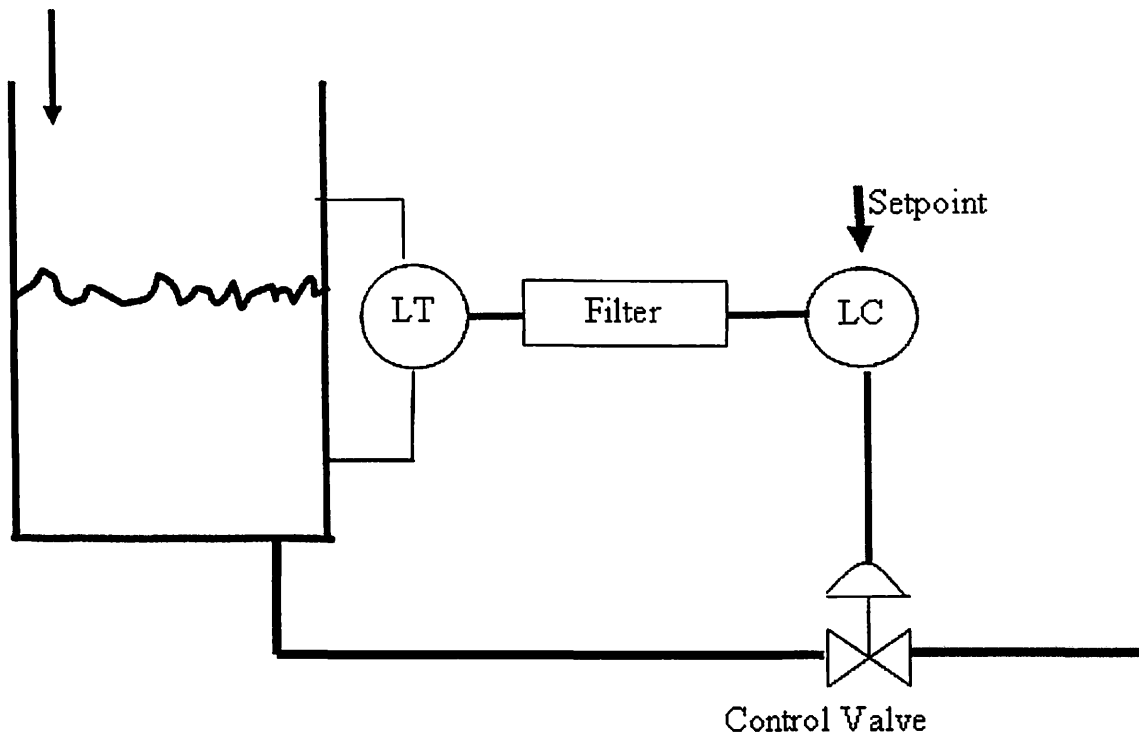


Figure 1.2: Configuration A - normal practice of using a filter

1.4: Novel Methods to Filter

A new filter based on a cumulative sum (CUSUM) of deviations [7,8], which has a working principle based on Statistical Process Control, can also be used in the same configuration to remove the process noise. Another configuration of the filter is also possible. The noisy process variable obtained from the sensor can be directly given to the controller. This will result in the controller output to be noisy as well. The noisy controller output can then be filtered and this filtered signal can be given to the control valve. This kind of setup is shown in Figure 1.3 and is named as Configuration B in this study. An analogy of this setup would be, “Don’t lie to the manager, but provide original information to him. Temper the management decisions and then take action.”

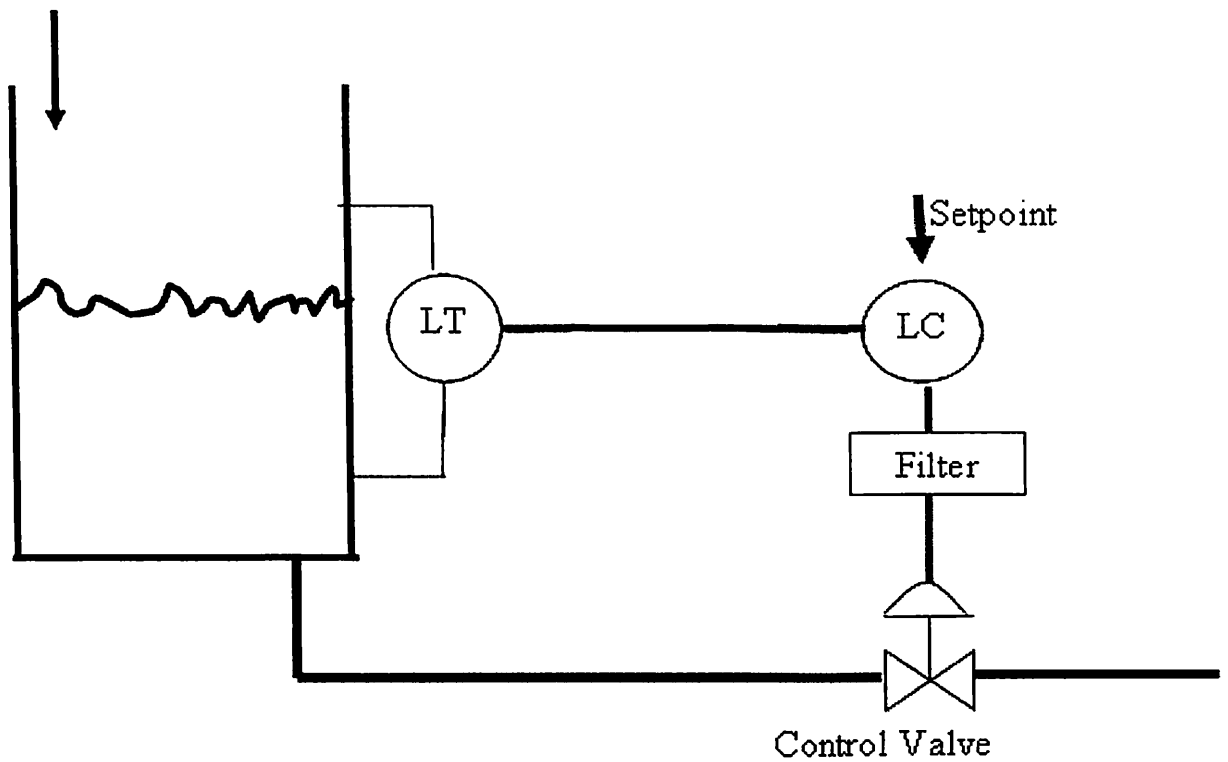


Figure 1.3: Configuration B - novel method of using a filter

This study compares the two filtering approaches for each configuration. Since credibility requires experimental demonstration, the combinations were tested on a cooling water flowrate loop of a distillation column in unit operations laboratory as well as with simulations performed using Simulink.

CHAPTER 2

LITERATURE SURVEY

2.1: Types of Filtering

Filters can be used for various purposes. A few applications are listed in this section.

Removing Noise: In process control, the measurement signals usually contain noise which cause unwanted jitter in the control valve. It is desired to remove this noise and extract the true process variable. In such cases, filters are used to “see” the true signal within the noisy signal. Predominantly, a first-order filter is used for this application.

Removing Outliers: In certain applications, it is required that the variability of the noisy data be preserved but removal of the extreme data points is required. Extreme data points can be caused due to a spark in the measurement system or a skip in the measurement sequence, resulting in a value of nearly infinity (or nearly zero) in between the valid data points. These extreme data points are called the outliers. In such cases, filters are used for removing these outliers. Examples of such filters are Hampel and weighted median filters [6].

Removing Faults: Neural Networks are used to identify patterns in the data obtained from a process. Once the patterns are identified, they are used to test the internal

consistency of the data. If it is found that the data is not consistent with the identified pattern, then it can be declared that the process contains faults. The faulty data is then discarded. This is also considered as one type of filtering.

Removing Aliasing: Special types of filters called Anti-Aliasing filters are used when an analog signal consisting of both, low frequency components and high frequency noise components have to be digitalized. If these analog signals were digitalized and ω_s were the sampling frequency, then, all signals with frequency greater than $\omega_s/2$ would appear as signals with low frequencies. This principle is called aliasing and would cause the misrepresentation of high frequency signals and corruption of low frequency signals. To avoid aliasing, these anti-aliasing filters are used ahead of the sampler to attenuate the higher frequency components [3].

2.2: Filters for Noise Removal

Different types of noise filters are available for removal of noise. A few of the filters are discussed in this section.

Averaging Filter: As the name suggests, these filters find the average of the numbers in the past, to find the output of the filter for the current sample. Though this filter is very simple to understand, it is computationally taxing and requires large storage space [12].

Moving Average Filters: These filters are similar to the averaging filters, except for the fact that they calculate the average more efficiently using recursive methods. The average is calculated using a moving window consisting of “n” data points. Compared to the averaging filter, the computational burden is less, but still, this method is not preferred

since the filtering cannot be initiated until “n” measurements have been made. Also this method gives equal weight to last “n” data points. In process control, where the measurements follow a trend, it is desired to give more weight to the most recent data since it is a better reflection of the state of the process [12].

Exponentially Weighted Moving Average filter: This filter, also known as the digital first-order filter, is the most commonly used noise reduction algorithm in process control. This filter places more weight on the most recent data and hence performs better than the moving average filter [12]. These filters attenuate the high frequency noise but they also introduce a lag in the system. The details of this filter and its implementation are discussed in later sections.

Novel Statistical Filter: The working principle of this filter is based on the statistical process control. It is believed that for process control applications, this statistical filter will perform better than the other filters discussed in this section [7,8]. The details of this filter and its implementation are discussed in later sections.

This study explores two filters, namely, the digital first-order filter and the statistical filter, for removing noise.

2.3: Analog First-Order Filter

As discussed in the previous section, the first-order filters are most commonly used for removing noise in process control applications since they are conceptually and computationally simple. While implemented digitally, today, the first-order filter has its roots in electronic Resistor-Capacitor (RC) circuitry.

The basic first-order differential equation represents the working of an analog first-order filter.

$$\tau_f \frac{dX_f(t)}{dt} + X_f(t) = X(t) \quad (2.1)$$

where $X(t)$ is the measurement variable which is given as in input to the filter and X_f is filter output. τ_f represents the filter time-constant. It can be seen that the filter has a steady state gain of 1. The differential equation of Equation 2.1 represents an RC circuit. The above filter is also called as the exponential filter [9].

The selection of the filter time-constant, τ_f , predominantly depends on the desired noise reduction. It also depends on the type of process. The filter time-constant should be very small as compared to the dominant time-constant of the process. For instance, it is desired to have $\tau_f < 0.1 * \tau_p$ where τ_p represents the dominant time-constant of the process [9].

2.4: Digital First-Order Filter

With the increasing use of computers for process control, digital filters are widely used since it is easy to program the digital filters in the computer. Digital filters have one big advantage that extra hardware is not required for filtering. A digital filter can be

thought as a computational block which takes in a sequence of numbers (filter input) and produces a new modified sequence of numbers (filter output) [10].

The most widely used digital filter is the digital first-order filter. The math involved in the first-order filter is described below [9]:

The Laplace transform of a first-order filter is

$$\frac{X_f(s)}{X(s)} = \frac{1}{\tau_f s + 1} \quad (2.2)$$

Where X_f is filter output and X is filter input.

The time domain equivalent of the above Laplace equation is

$$\tau_f \frac{dX_f(t)}{dt} + X_f(t) = X(t) \quad (2.3)$$

The time series of samples of the filter input (measured variables) are denoted as $X(i)$, $X(i-1)$, $X(i-2)$... and the corresponding filter outputs are denoted as $X_f(i)$, $X_f(i-1)$, $X_f(i-2)$...where i represents the current sampling instant.

Discretizing Equation 2.3 using the backward difference approximation

$$\tau_f \frac{X_f(i) - X_f(i-1)}{\Delta t} + X_f(i) = X(i) \quad (2.4)$$

where Δt represents the sampling time

Rearranging Equation 2.4 to solve for $X_f(i)$

$$X_f(i) = \frac{\Delta t}{\tau_f + \Delta t} X(i) + \frac{\tau_f}{\tau_f + \Delta t} X_f(i-1) \quad (2.5)$$

Let $\lambda = \frac{\tau_f}{\tau_f + \Delta t}$, then $(1 - \lambda) = \frac{\Delta t}{\tau_f + \Delta t}$

Substituting the values of λ in Equation 2.5

$$X_f(i) = (1 - \lambda)X(i) + (\lambda)X_f(i - 1) \quad (2.6)$$

Equation 2.6 represents the digital form of the first-order filter.

The above equation can be also written as follows:

$$X_{fof_i} = \lambda(X_{fof_{i-1}}) + (1 - \lambda)X_i \quad (2.7)$$

The level of filtering can be changed by changing the filter time-constant, τ_f . If it is decided to keep the sampling time, Δt , as constant, then the level of filtering can be changed by changing λ directly. Qualitatively, the digital first-order filter can be defined as a weighted sum of the previous filter output and the current unfiltered input. The flowchart of the digital filter is represented in Figure 2.1. The “:=” symbol in Figure 2.1 represents the assignment operator.

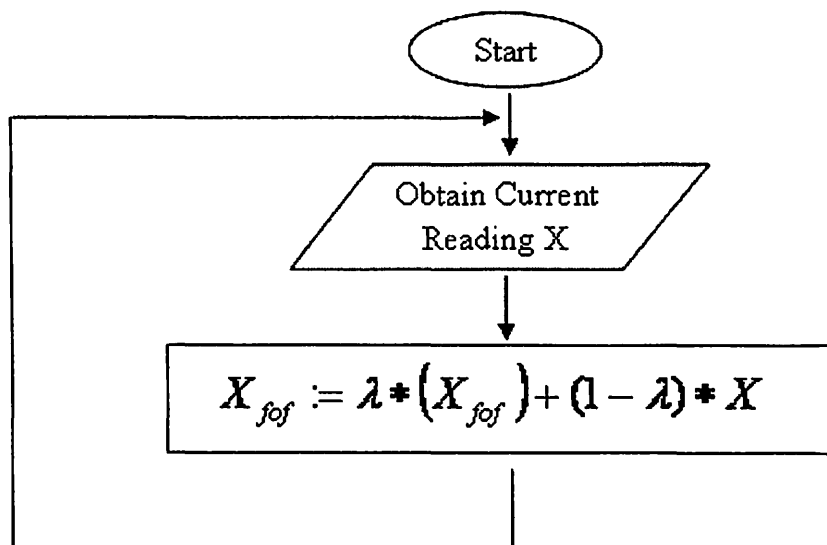


Figure 2.1: Flowchart of digital first-order filter

2.5: Open Loop Response of a First-Order Filter

Figure 2.2 shows a basic block diagram of finding the open loop response of a first-order filter.

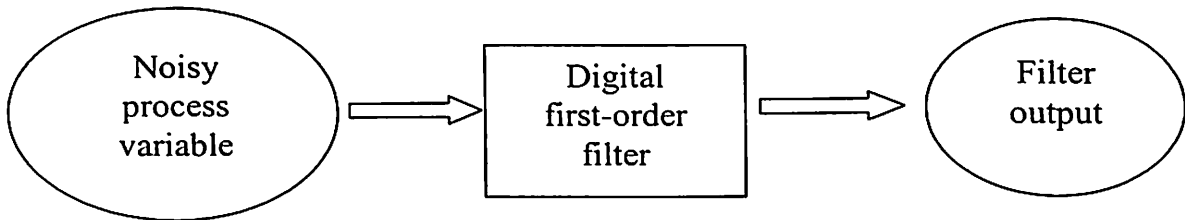


Figure 2.2: First-order filter open loop response block diagram

From this point, the digital first-order filter will be referred as just a first-order filter. Figure 2.3 shows a graph depicting the open loop response of the first-order filter. The horizontal axis represents time and the vertical axis represents two variables namely, the noisy process variable and the filtered process variable i.e. the output of the first-order filter.

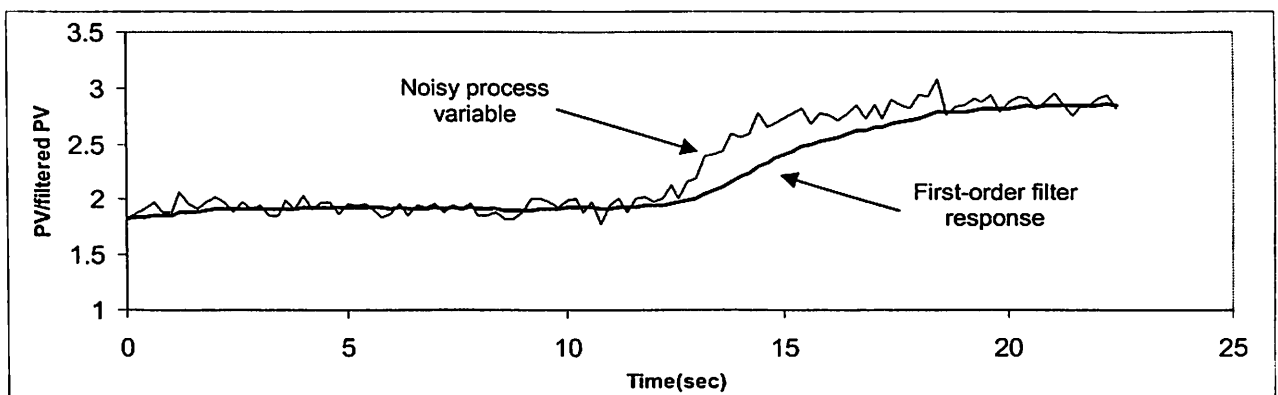


Figure 2.3: First-order filter open loop response

From Figure 2.3, it can be seen that the first-order filter reduces the noise present in the process variable. It can also be observed that during the steady state period (0 to 10 sec), the first-order output keeps changing at every sample. A disadvantage of the first-

order filter is that when there is a change in the steady state value, the first-order filter output lags behind; and hence, does not follow the process change immediately. This is evident from Figure 2.3, during the transient period (10 sec to 20 sec).

If more noise reduction is required, the value of λ has to be increased. However, if λ value is increased, it will result in more lag, which is undesirable.

2.6: Metric – Ways to Quantify Lag

From Section 2.5, it is evident that when the steady state value of the process variable changes (10 to 20 sec, Figure 2.3), the filtered output introduces a lag. One of the ways to quantify this lag is to measure the cumulative sum of the differences between the noisy process variable (PV) and the filter output during the transient phase. This summation can be used as a direct measure of the lag introduced by the filter. If the filter output introduced a larger lag, the cumulative sum of the differences between the PV and the filter output will be higher and vice versa.

2.7: Concept of the CUSUM Filter [7,8]

The working principle of the CUSUM filter is based on Statistical Process Control concepts. The CUSUM filter has a tuning parameter called “trigger” indicating “sigma” level of significance. If a trigger value of 2 is selected (meaning 2-sigma level of significance), the CUSUM filter keeps the filtered value constant until it is 95% sure that a change has occurred. If a trigger value of 3 is selected, then the CUSUM filter does not change the filtered value until it is 99.7% sure that a change has occurred. The idea being, the previous filtered output is retained until there is statistically sufficient evidence that a

true change in filter input has occurred. This idea of accepting the inherent process variability and reporting changes only when there is high statistical confidence that a change is justified is one of the fundamental perspectives of statistical process control.

2.8: Working of the CUSUM Filter

The flowchart for the working of CUSUM filter is shown in Figure 2.4.

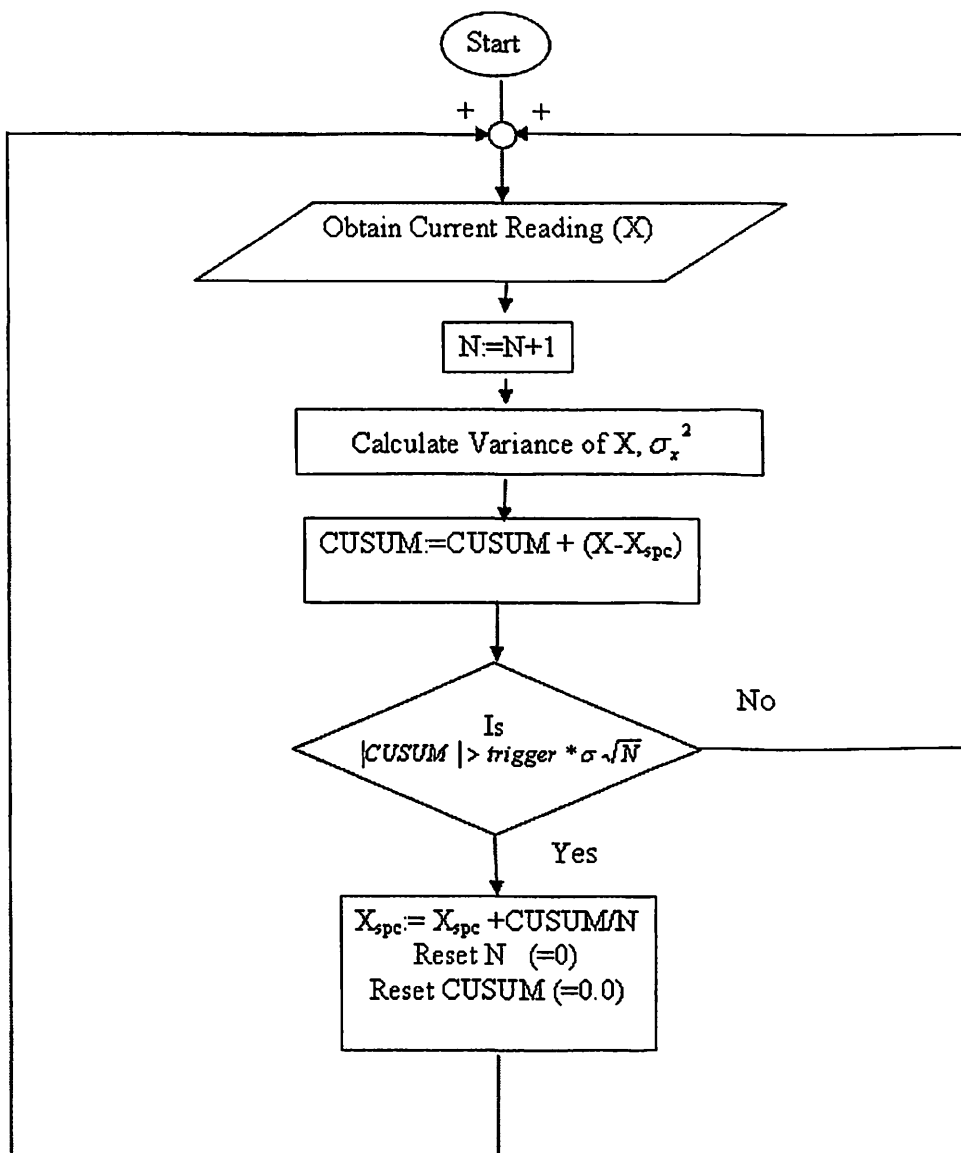


Figure 2.4: Flow chart of the CUSUM filter

The algorithm of the CUSUM filter is shown below

- Step 1:** The current reading of the unfiltered value is given as an input to the CUSUM filter.
- Step 2:** The value of N is incremented by 1 and the new variance is calculated. There are various methods to calculate variance. The method described in references [7,8] is used here. N represents the number of samples the CUSUM filter output value has been held constant.
- Step 3:** The value of CUSUM is updated. CUSUM refers to the cumulative algebraic sum of the difference between the filtered and the unfiltered values.
- Step 4a:** If the absolute value of CUSUM is greater than $\text{Trigger}^*(\sigma) * \sqrt{N}$ value, then the filter assumes that a true change in process has occurred; and hence, it updates the filtered value as $X_{\text{spc}} = X_{\text{spc}} + \text{CUSUM}/N$ and the variable N and CUSUM is reset. This rule assumes that the offset has been constant for the past N samples.
- Step 4b:** If the CUSUM is less than the $\text{Trigger}^*(\sigma) * \sqrt{N}$ value, then the filter assumes that the change in input is due to normal variability of the process and is not because of a true change in process. Hence, it does not change the filtered output.
- Step 5:** Repeat Step 1.

Figure 2.5 illustrates the working of the CUSUM filter algorithm. The graph in the top portion of Figure 2.5 represents a plot of the CUSUM filter input (actual variable) and CUSUM filter output (filtered variable) with respect to time. The noisy signal used as an input to the CUSUM filter is generated by a Gaussian random number generator. The graph in the bottom half of Figure 2.5 represents the values of $|CUSUM|$ and $2\sigma\sqrt{N}$ with respect to time.

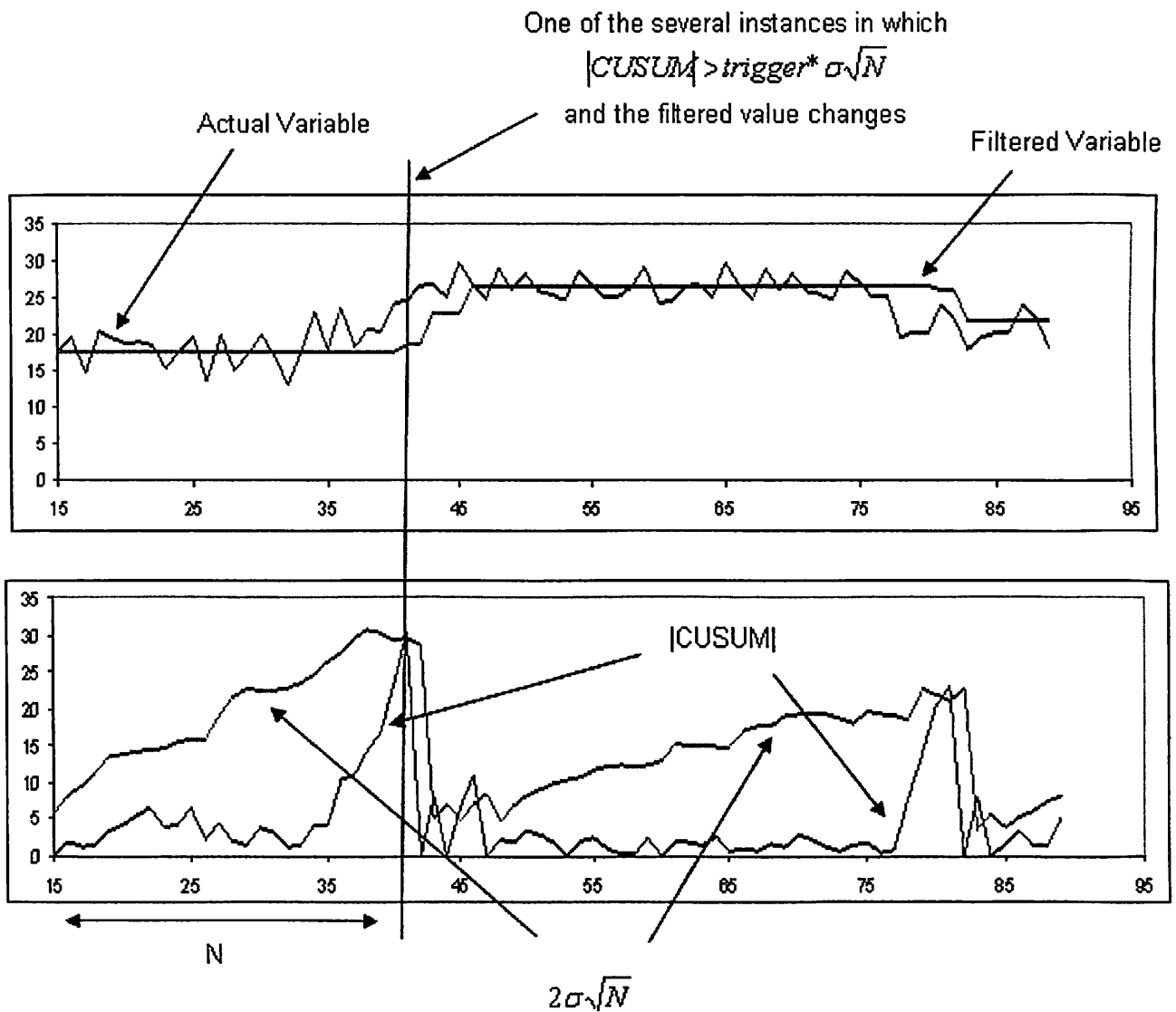


Figure 2.5: Working of the CUSUM filter

From Figure 2.5 it is obvious that as long as $|\text{CUSUM}| < (\text{trigger}) * \sigma\sqrt{N}$, the filter output is kept constant. Whenever $|\text{CUSUM}| > (\text{trigger}) * \sigma\sqrt{N}$, the filtered value changes and the value of CUSUM is reset to 0.0.

Figure 2.6 illustrates the Fortran embodiment of the executable instructions.

```
IF (first call) THEN
    N = 0
    XOLD = 0.0
    XSPC = 0.0
    V = 0.0
    CUSUM = 0.0
    M=11
    FF2 = 1.0/(M-1)/2.0
    FF1 = REAL((M-2)/(M-1))
END IF
Obtain X
N = N + 1
V = FF1* V + FF2*(X - XOLD)**2
XOLD = X
CUSUM = CUSUM + X - XSPC
IF (ABS (CUSUM) .GT. TRIGGER*SQR (V*N)) THEN
    XSPC = XSPC + CUSUM/N
    N = 0
    CUSUM = 0.0
END IF
```

Figure 2.6: The code of implementing the CUSUM filter [8]

2.9: Open Loop Response of a CUSUM Filter

Figure 2.7 shows a basic block diagram of the open loop response of a CUSUM filter.

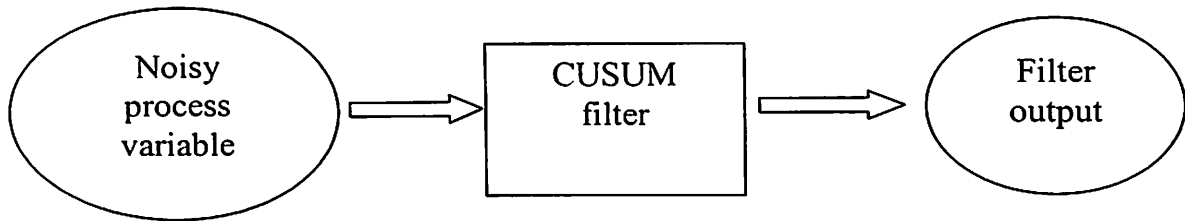


Figure 2.7: CUSUM filter open loop response block diagram

Figure 2.8 shows a graph depicting the open loop response of the CUSUM filter. The horizontal axis represents time and the vertical axis represents two variables namely, the noisy process variable and the filtered process variable i.e. the output of the CUSUM filter. The noisy process variable used as an input to the CUSUM filter is the same that was used as an input to the first-order filter in Section 2.5. Hence, Figure 2.3 and Figure 2.8 display the output of the filters for the same input.

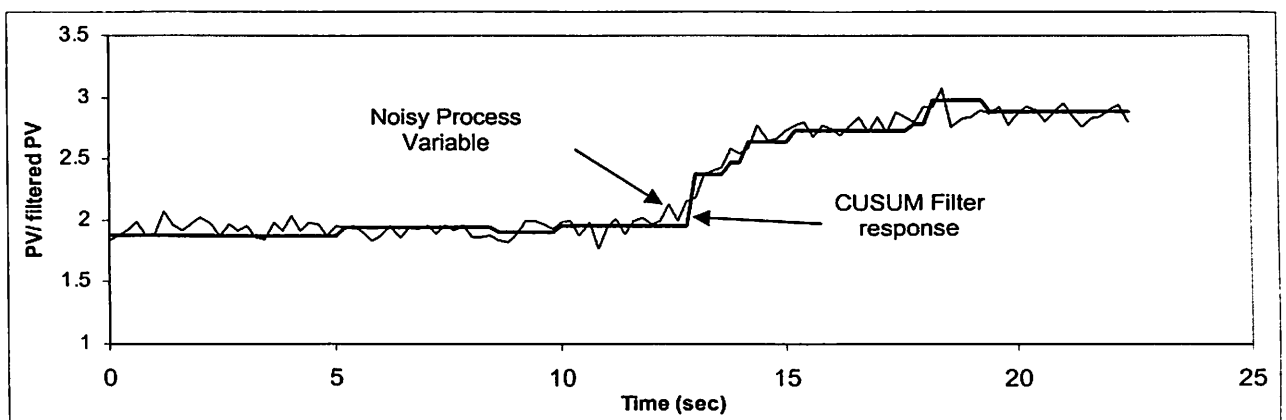


Figure 2.8: CUSUM filter open loop response

From Figure 2.8, it can be observed that during the steady state period (0 to 10 seconds), the CUSUM filter output is help constant for a major amount of time and hence

it successfully results in reduction of noise. Also during a transient (10 to 20 seconds), the CUSUM filter output does not lag behind but instead, it follows the change nearly immediately.

2.10: Different Ways to Place a Filter – Mathematical Analysis

In Section 1.3 and 1.4, different configurations of placing a filter were discussed. Mathematical analysis of these configurations is done in this section.

Configuration A – Servo Mode Analysis:

Figure 2.9 shows the block diagram of a control system having the filter on the input side of the controller. It is chosen to place the filter on the process variable in this configuration.

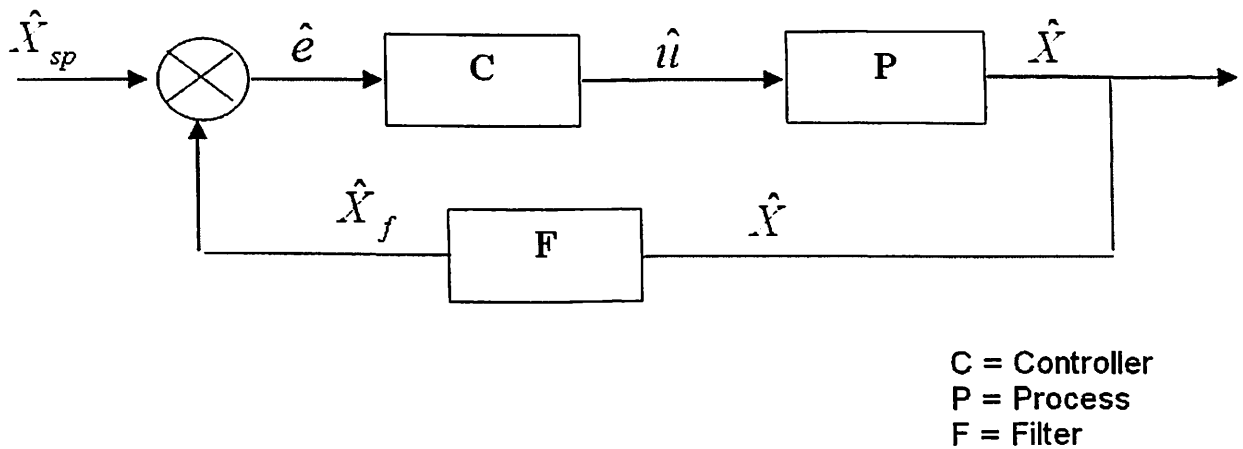


Figure 2.9: Block diagram for filter on PV – servo mode

Finding the relation between actual process variable and setpoint...

$$\hat{X} = PC\hat{e} \tag{2.8}$$

$$\hat{e} = \hat{X}_{sp} - \hat{X}_f = \hat{X}_{sp} - F\hat{X} \tag{2.9}$$

Substituting Equation 2.9 in Equation 2.8 gives

$$\hat{X} = PC(\hat{X}_{sp} - F\hat{X}) \quad (2.10)$$

Solving for \hat{X} from Equation 2.10 gives

$$\hat{X} = \left(\frac{PC}{1 + PCF} \right) \hat{X}_{sp} \quad (2.11)$$

Hence, the transfer function for the PV response to a setpoint change is given by

$$\frac{\hat{X}}{\hat{X}_{sp}} = \left(\frac{PC}{1 + PCF} \right) \quad (2.12)$$

The actuating error is given by the equation

$$error = \hat{X}_{sp} - \hat{X} \quad (2.13)$$

Substituting the value of \hat{X} from Equation 2.12 and simplifying yields

$$error = \left(\frac{1 + PC(F - 1)}{1 + PCF} \right) \hat{X}_{sp} \quad (2.14)$$

Finding the relation between manipulated variable and setpoint...

$$\hat{u} = C\hat{e} = C(\hat{X}_{sp} - \hat{X}_f) \quad (2.15)$$

$$\hat{u} = C(\hat{X}_{sp} - F\hat{X}) = C(\hat{X}_{sp} - FXP\hat{u}) \quad (2.16)$$

Simplifying Equation 2.16 and solving for \hat{u} gives

$$\hat{u} = \left(\frac{C}{1 + CFP} \right) \hat{X}_{sp} \quad (2.17)$$

Configuration B – Servo Mode Analysis:

Figure 2.10 shows the block diagram of a control system having the filter on the output side of the controller. It is chosen to place the filter on the manipulated variable in this configuration.

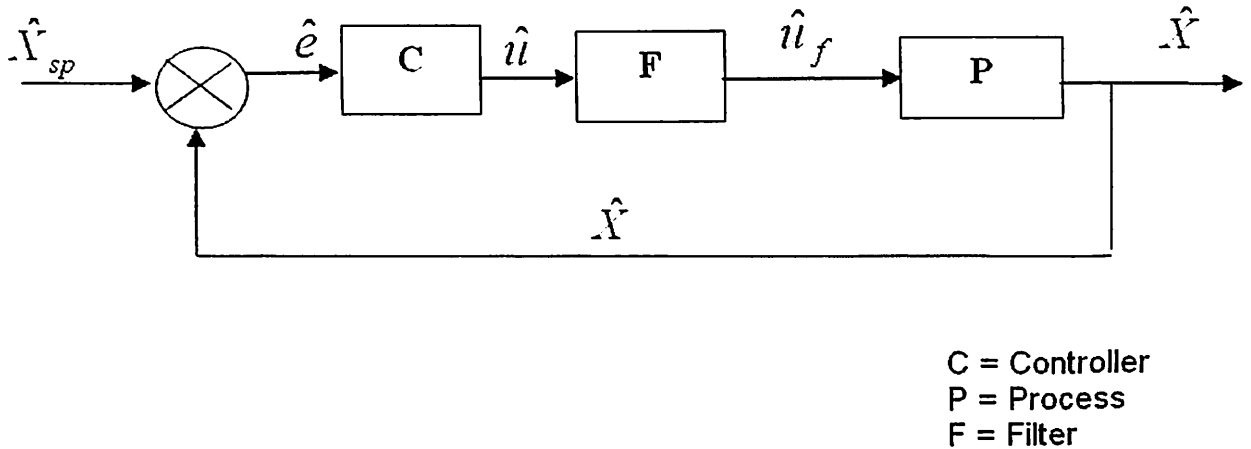


Figure 2.10: Block diagram for filter on MV – servo mode

$$\hat{X} = PFC\hat{e} \quad (2.18)$$

$$\hat{e} = \hat{X}_{sp} - \hat{X} \quad (2.19)$$

Substituting Equation 2.19 in Equation 2.18 gives

$$\hat{X} = PFC(\hat{X}_{sp} - \hat{X}) \quad (2.20)$$

Solving for \hat{X} from Equation 2.20 gives

$$\hat{X} = \left(\frac{PFC}{1 + PCF} \right) \hat{X}_{sp} \quad (2.21)$$

Hence, the transfer function for the PV response to a setpoint change is given by

$$\frac{\hat{X}}{\hat{X}_{sp}} = \left(\frac{PFC}{1 + PCF} \right) \quad (2.22)$$

The actuating error is given by the equation

$$error = \hat{X}_{sp} - \hat{X} \quad (2.23)$$

Substituting the value of \hat{X} from Equation 2.22 and simplifying yields

$$error = \left(\frac{1}{1 + PCF} \right) \hat{X}_{sp} \quad (2.24)$$

Since the filtered controller output is used as the manipulated variable, it is desired to find the relation between the filtered controller output, \hat{u}_f (not the actual controller output), and the setpoint, \hat{X}_{sp} .

$$\hat{u}_f = FC\hat{e} = FC(\hat{X}_{sp} - \hat{X}) \quad (2.25)$$

$$\hat{u}_f = FC(\hat{X}_{sp} - \hat{X}) = FC(\hat{X}_{sp} - P\hat{u}_f) \quad (2.26)$$

Simplifying Equation 2.26 and solving for \hat{u} gives

$$\hat{u} = \left(\frac{FC}{1 + FCP} \right) \hat{X}_{sp} \quad (2.27)$$

Conclusion:

From Equation 2.12 and Equation 2.22, it can be noted that the transfer function for Configuration A is different from that of Configuration B. Hence, placing the filter on the PV will give a difference response to a change in setpoint as compared to placing the filter on the MV.

Configuration A – Regulatory Mode Analysis:

Figure 2.11 shows the block diagram of a control system having the filter on the input side of the controller. It is chosen to place the filter on the process variable in this configuration.

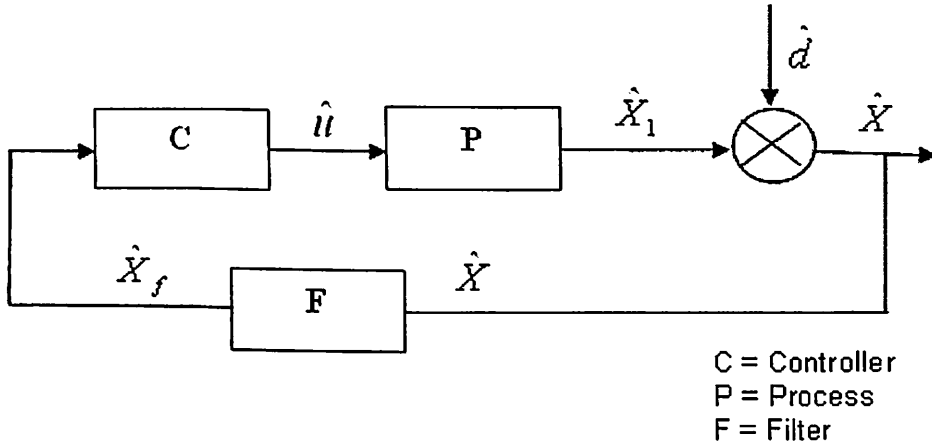


Figure 2.11: Block diagram for filter on PV – regulatory mode

$$\hat{X} = \hat{X}_1 + \hat{d} \quad (2.28)$$

$$\hat{X}_1 = PCF\hat{X} \quad (2.29)$$

Substituting Equation 2.29 in Equation 2.28 gives

$$\hat{X} = PCF\hat{X} + \hat{d} \quad (2.30)$$

Rearranging Equation 2.30 gives

$$\frac{\hat{X}}{\hat{d}} = \left(\frac{1}{1 - PCF} \right) \hat{d} \quad (2.31)$$

Configuration B – Regulatory Mode Analysis:

Figure 2.12 shows the block diagram of a control system having the filter on the output side of the controller. It is chosen to place the filter on the manipulated variable in this configuration.

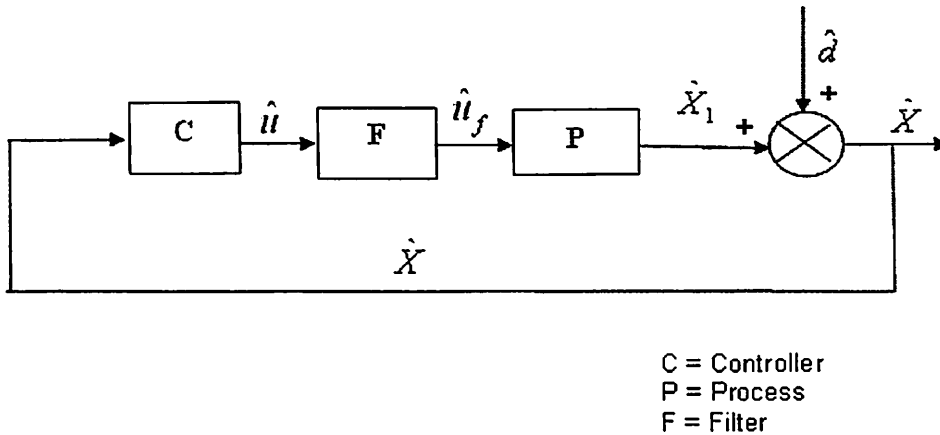


Figure 2.12: Block diagram for filter on MV – regulatory mode

$$\hat{X} = \hat{X}_1 + \hat{d} \quad (2.32)$$

$$\hat{X}_1 = PFC\hat{X} \quad (2.33)$$

Substituting Equation 2.33 in Equation 2.32 gives

$$\hat{X} = PFC\hat{X} + \hat{d} \quad (2.34)$$

Rearranging Equation 2.34

$$\frac{\hat{X}}{\hat{d}} = \left(\frac{1}{1 - PFC} \right) \hat{d} \quad (2.35)$$

Since all the blocks are linear, Equation 2.35 can be written as

$$\frac{\hat{X}}{\hat{d}} = \left(\frac{1}{1-PCF} \right) \hat{d} \quad (2.36)$$

Conclusion:

From Equation 2.31 and Equation 2.36, it can be noted that for regulatory mode, the transfer function for Configuration A is same as that of Configuration B. Hence, for regulatory mode, placing the filter on the PV will give the same response for disturbance rejection as compared to placing the filter on the MV. It should be noted that this conclusion is valid only for linear filters.

2.11: Metrics - Ways to Quantify the Goodness of a Filter in Closed Loop

There are two filters i.e. the first-order filter and the CUSUM filter, which can be used on the PV (Configuration A) or on the MV (Configuration B). Certain quantitative and qualitative criteria of comparison are required for understanding the relative advantages and disadvantages of these configurations.

The variability in both the process variable (PV) and manipulated variable (MV) is important. It is chosen to quantify PV variability by the Integral of the Square of Error (ISE), where error is the actuating error, the difference between the actual process variable (not the filtered PV) and the setpoint.

It is chosen to quantify MV variability by measuring valve travel. Valve travel is defined as the cumulative absolute value of the difference between consecutive controller

outputs. Low values of ISE portray better controller performance while low values of valve travel portray less wear and tear of the control valve. The general trend being, if the filtering is increased to obtain lower valve travel, there is a simultaneous increase in ISE value because disturbances are not quickly fixed. Hence, a balance between the ISE and valve travel is desired.

2.12: Filtering in Regulatory Mode – Mathematical Analysis

Case 1 - Process Noise and Zero Disturbance:

As discussed in Section 1.2, if noisy process data is given to an aggressive controller, the controller output will amplify the noise and hence will result in unnecessary tampering with the control valve. This will cause an increase in ISE value. Hence, at zero filtering, noise induces additional variability in both PV (ISE) and MV (travel). Qualitatively, the trend for Case 1 is shown in Figure 2.13. In Figure 2.13, the horizontal axis is represented by the valve travel and the vertical axis is represented by ISE.

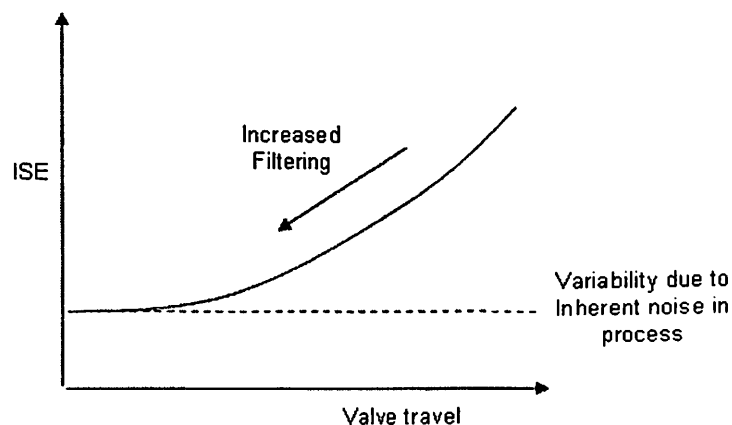


Figure 2.13: Trend for Case 1 – process noise and zero disturbance

If the process has no disturbance then, at the limit of zero valve travel i.e. infinite filtering, ISE will reflect process variance.

$$ISE = \sum_{i=1}^N (X_i - X_{sp})^2 \Delta t = \Delta t \sum_{i=1}^N (X_i - X_{sp})^2 \quad (2.37)$$

If the noisy process data is averaging at X_{sp} , then $X_{sp} = \bar{X}$

$$ISE = \Delta t \sum_{i=1}^N (X_i - \bar{X})^2 \quad (2.38)$$

If T is the entire window then Equation 2.38 can be written as

$$ISE = \frac{T}{N} \sum_{i=1}^N (X_i - \bar{X})^2 = T\sigma^2 \quad (2.39)$$

Where σ^2 represents process variance

$$\text{Hence } \frac{ISE}{T} = \sigma^2 \quad (2.40)$$

Case 2: Disturbance and Zero Noise

If the process has no noise but frequent disturbances, then zero filtering will result in lowest ISE. Increasing the filtering will result in slower valve response. Hence, the valve will take more time to counter the effect of the disturbances in the process, causing an increase in ISE value qualitatively. The trend for Case 2 is shown in Figure 2.14.

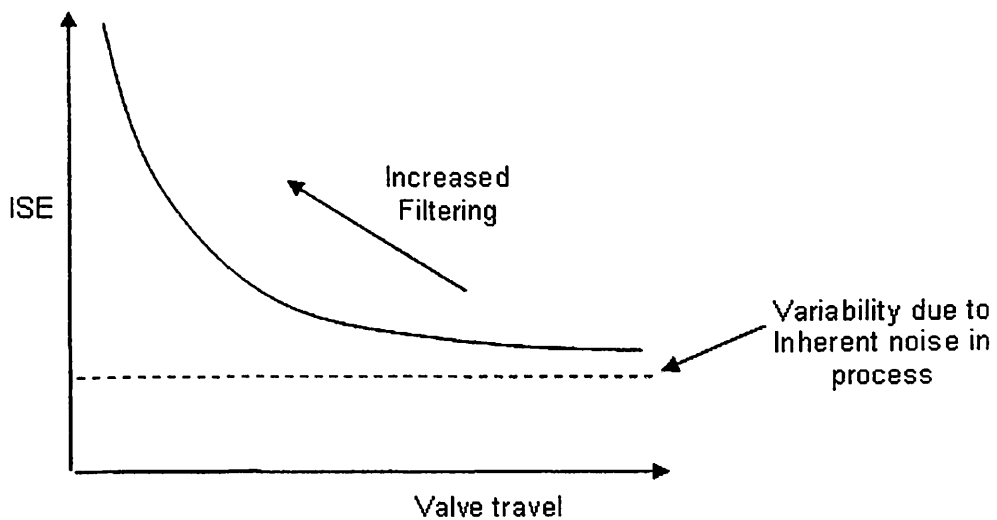


Figure 2.14: Trend for Case 2 –disturbance and zero noise

Conclusion:

When the process is influenced by noise and disturbance effects, a summed response should be obtained. The plot of the summed response is shown in Figure 2.15. As the filtering is increased to a certain extent, there is a reduction in ISE value and valve travel value as well. Further increase in filtering causes an increase in ISE value while reducing the valve travel further. If the noise contains autocorrelation, then the noisy data will include persistence and simulate the effect of a disturbance.

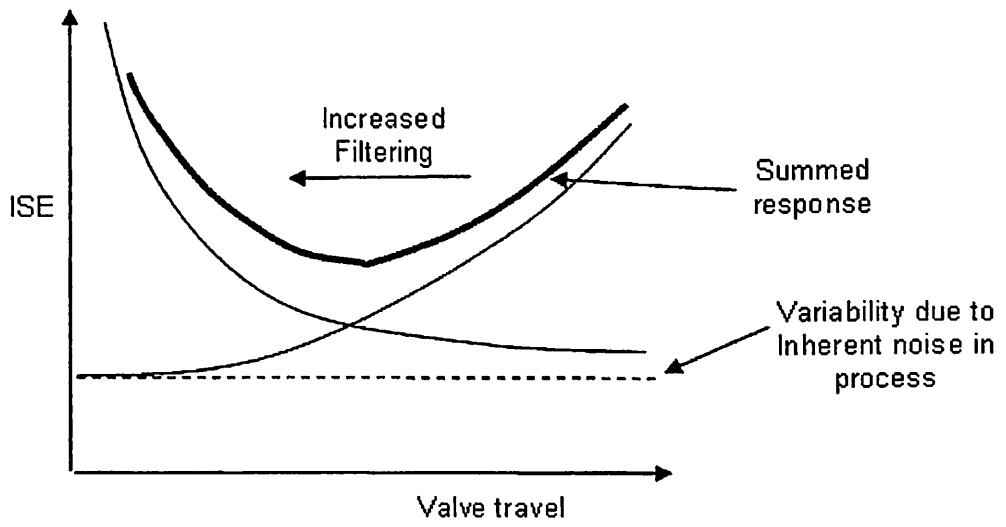


Figure 2.15: Summed response due to disturbances and noise in process

CHAPTER 3

COOLING WATER FLOWRATE SYSTEM – EXPERIMENTAL SETUP

The experimental analysis of the filters is carried out on the cooling water flow loop of the fractional distillation column installed in the Unit Operations Lab of school of chemical engineering at Oklahoma State University.

3.1: Description of the Process

Figure 3.1 shows the distillation column setup. The pilot scale distillation column is a Technovate Model 9079 fractional distillation column system. The cooling water flow rate process is a part of this distillation column setup. The vapors from the top tray escape out of the distillation column. These vapors are condensed in the condenser and then collected in a condensate tank. In order to condense the vapors, cooling water is used as the cooling medium. The fluid used for the cooling water flow loop is the water from the municipal water supply pipeline.

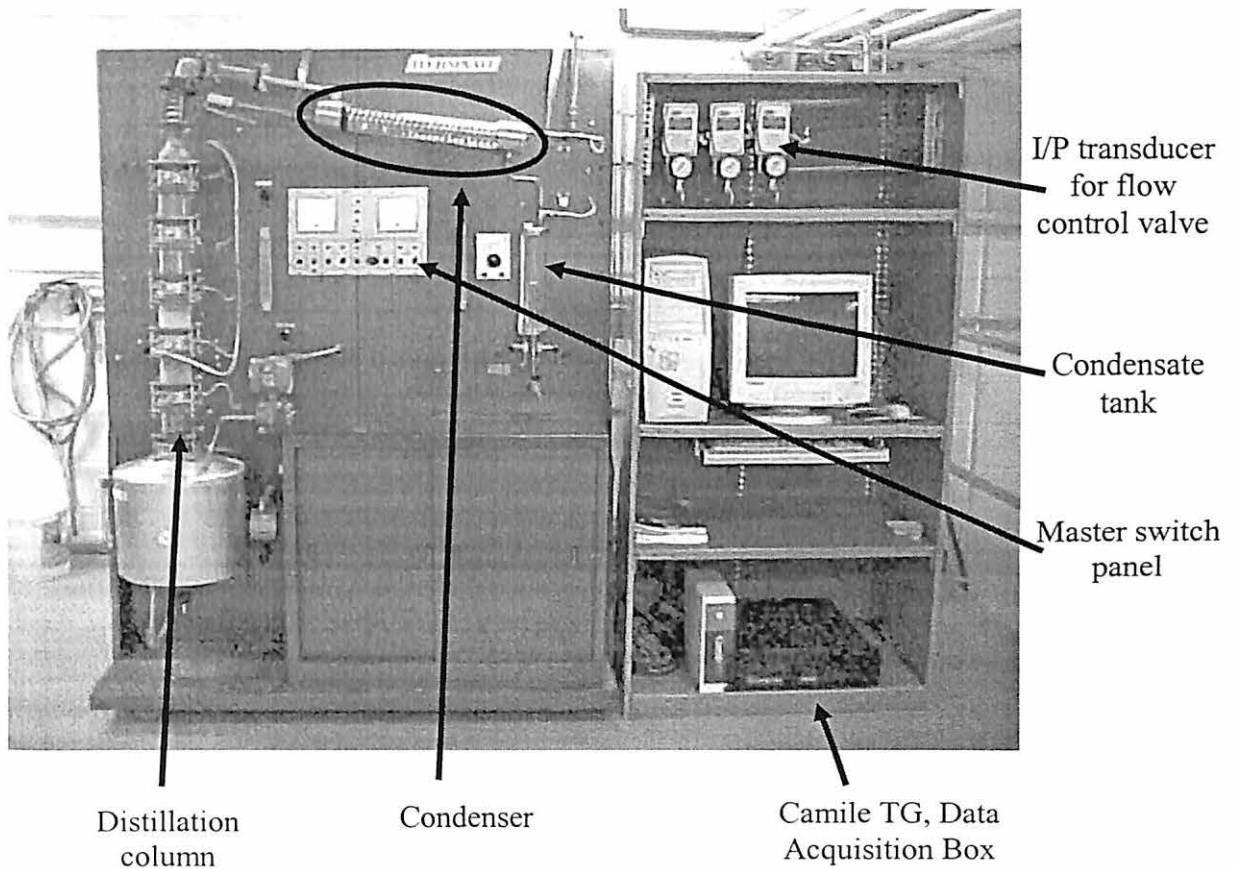


Figure 3.1: Distillation column setup

Figure 3.2 shows the cooling water flow control valve and the orifice meter. The calibration of the orifice meter is discussed in Appendix A. The controlled variable is the cooling water flowrate and the manipulated variable is the controller output signal given to the control valve. The water lines are $\frac{1}{4}$ inch tubing; and the control valve has a diaphragm air actuator and a C_v of 2.5.

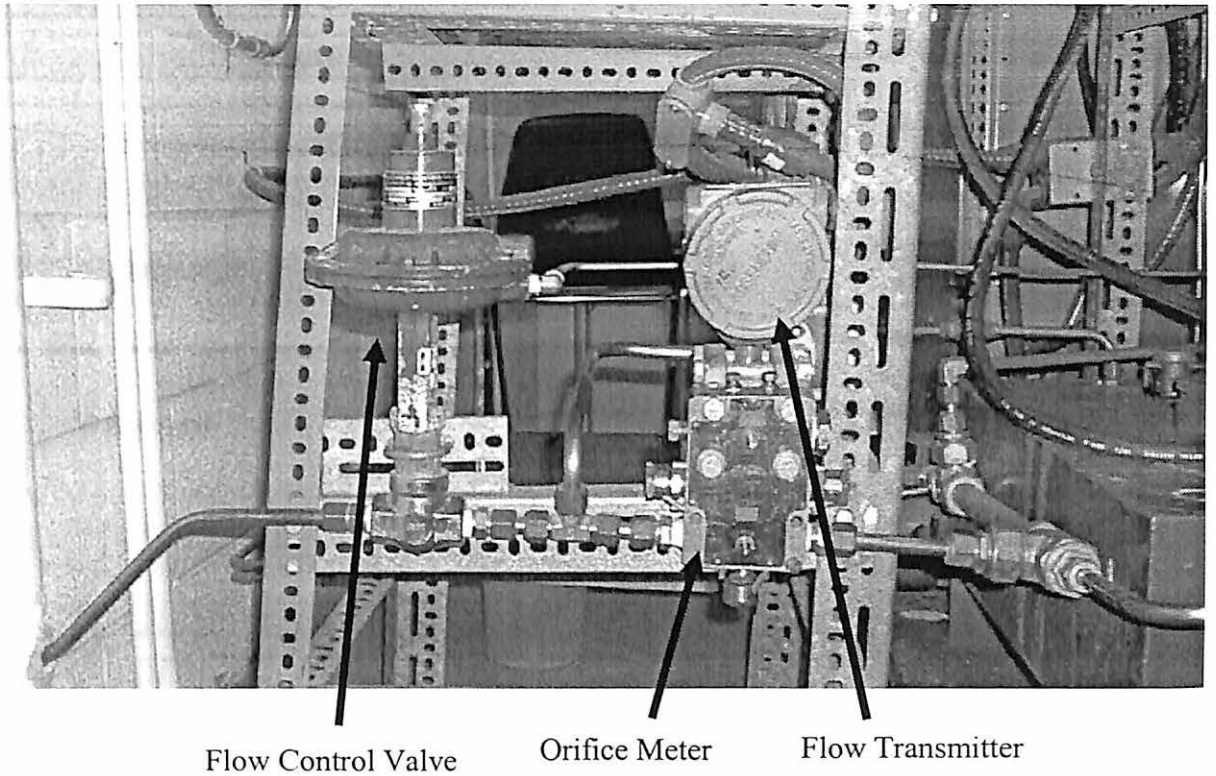


Figure 3.2: Cooling water flow control valve and orifice meter

The block diagram of the control system is shown in Figure 3.3. The cooling water flowrate is measured by the flow transmitter which gives a 4-20 mA signal. This signal is digitalized using an A/D converter. The digitalized signal is given to the computer. The digital value of current, i , is converted to flowrate value inside the computer and this value is given as an input to the controller. The controller accordingly gives an output in percentage. The percentage value is converted to 4-20 mA using a D/A converter. This 4-20 mA signal is further converted into 3-15 psi with the help of an I/P converter. The 3-15 psi signal is finally given to the control valve diaphragm, which changes the valve stem position, which changed the flowrate.

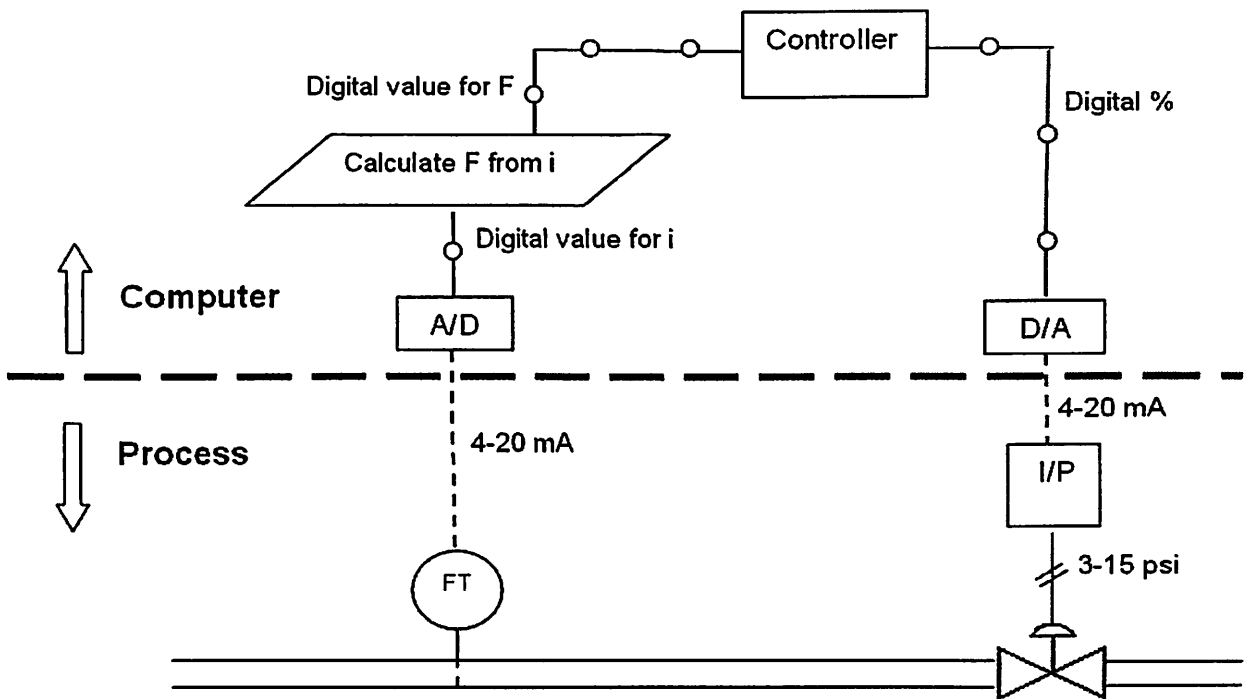


Figure 3.3: Cooling water flow rate system

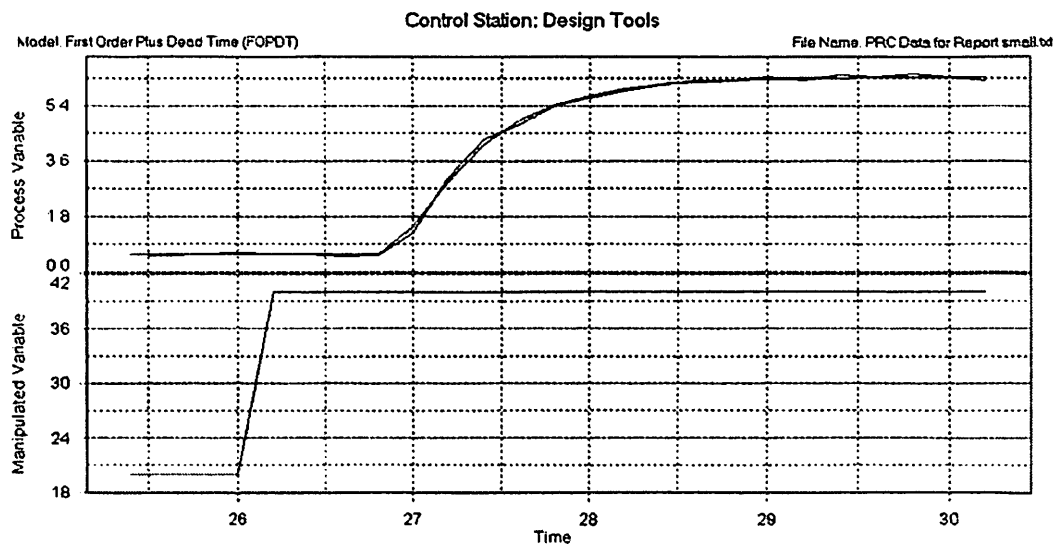
3.2: Description of Data Acquisition and Control System

Originally the Technovate distillation column unit was built for manual control. The manual control panel is marked as “master switch panel” in Figure 3.1. To automate the control systems, the manual controls are bypassed. Camile TG 2200 DAC (Data Acquisition and Control) installed on a Pentium II PC (333 MHz - CPU clock frequency) is used as the data acquisition and control system. Camile performs all data acquisition and control calculations. The data from the process is acquired by PC via RS-232 port. Graphical user interface screens can be built using Camile TG 2200 which can be used for monitoring and controller the cooling water flow loop. In this way, Camile helps the operator interact with the process.

3.3: Development of Process Model

For testing the filters in closed loop, the controller has to be tuned first. Controller tuning is done by the process reaction curve method [5] which requires a first-order plus time delay (FOPTD) model of the process. The development of the process model is described in this section.

A step change in controller output (manipulated variable) was performed and the change in cooling water flow rate was recorded. The sampling time was set to 0.2 seconds. This process reaction curve data was given to the Control Station software and the FOPTD model was calculated using the software. Snapshots of the FOPTD fit obtained from the Control Station software (<http://www.controlstation.com>) is shown in Figure 3.4.



Model Parameters	
Process Gain, K	0.2875
Overall Time Constant, τ	0.4228
Dead Time, θ	0.6748
Sum of Squared Error (SSE)	0.6051
Goodness of Fit (R^2)	0.9995

Figure 3.4: Snapshots of Control Station FOPTD fitting

The following process model was obtained:

$$K=0.29 \text{ (kg/min)/\%}$$

$$\tau =0.43 \text{ sec}$$

$$\theta =0.68 \text{ sec}$$

Process Transfer Function:

$$\frac{Y(s)}{U(s)} = \frac{0.29e^{-0.68s}}{(0.43s + 1)} \quad (3.1)$$

It is understood that the sampling time should be 1/5th of the smallest time constant of the process. In spite of that, the sampling time was set to 0.2 seconds because values lower than that were found to result in excessive computational load on the computer of this equipment and hence, resulted in slower response of the graphical user interface.

3.4: Design of the PI Controller

ITAE (Servo) Tuning Rules [5] were used to find controller tuning parameters for a PI controller.

Process Characteristics:

$$K=0.29 \text{ (kg/min)/\%, } \tau =0.43 \text{ sec, } \theta =0.68 \text{ sec, } \Delta t=0.2 \text{ sec}$$

PI controller design equations [5]:

$$K_c = \frac{0.586}{K} \left(\frac{\tau}{\theta} \right)^{0.916} \quad (3.2)$$

$$\tau_i = \frac{\tau}{1.03 - 0.165 \left(\frac{\theta}{\tau} \right)} \quad (3.3)$$

$$K_c = 1.33 \% / (\text{kg}/\text{min}) \quad \tau_i = 0.57 \text{ sec}$$

Since Camile uses the parallel form, not the standard form, of the PI control algorithm, K_c and τ_i must be converted to the independent multipliers for the proportional and integral functions

$$\begin{aligned} K_c &= 1.33 \% / (\text{kg}/\text{min}) \\ K_i &= (1.33/0.57) = 2.33 \% / (\text{kg}\cdot\text{sec}/\text{min}) \end{aligned}$$

3.5: Design of Digital PI Controller

Simulations of the closed loop configuration of filters were performed using Simulink. In order to replicate the PI controller of Camile in Simulink, the digital controller form is required. The design of digital PI controller is discussed in this section.

The method used to design the digital PI Controller is extracted from CHE 5853, Lecture 16, spring 2004 notes [13].

The velocity form of the control equation [13] is:

$$u(k) - u(k-1) = K_c \left[(e(k) - e(k-1)) + \frac{\Delta t}{\tau_i} (e(k)) \right] \quad (3.4)$$

Rearranging the above equations and taking the z-transform [13]

$$\frac{u(z)}{e(z)} = g_c(z) = \frac{\theta_0 + \theta_1 z^{-1}}{1 - z^{-1}} \quad (3.5)$$

Where

$$\begin{aligned} \theta_0 &= K_c \left(1 + \frac{\Delta t}{\tau_i} \right) \\ \theta_1 &= -K_c \end{aligned} \quad (3.6)$$

Substituting the values of K_c and τ_l into Equations 3.6:

$$\theta_0 = 1.33 \left(1 + \frac{0.2}{0.57} \right) = 1.80 \quad (3.7)$$

$$\theta_1 = -1.33 \quad (3.8)$$

$$\frac{u(z)}{e(z)} = g_c(z) = \frac{1.80 - 1.33z^{-1}}{1 - z^{-1}} \quad (3.9)$$

CHAPTER 4

EXPERIMENTAL PROCEDURE, RESULTS AND ANALYSIS

4.1: Experimentation Description

The first-order filter and CUSUM filter were each used in two configurations, as described below.

Configuration A: In Configuration A, the filter is used on the input side of the controller. This configuration is the conventional approach in the process industry. Here, the noisy process variable is filtered and this filtered variable is given as an input to the controller. The controller signal is given directly to the control valve. This configuration is explored with both filters.

Configuration A1 - First-Order Filter on PV: Figure 4.1, shows the general block diagram of Configuration A1.

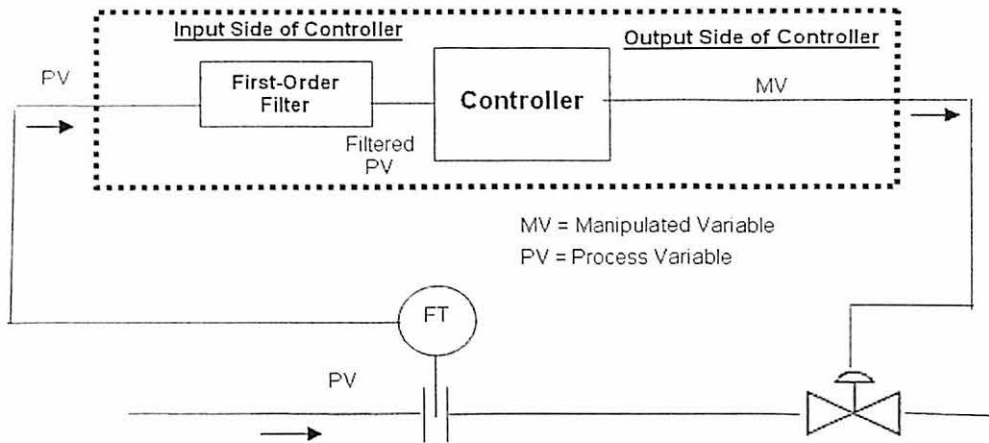


Figure 4.1: Block diagram for Configuration A1

A Graphical user interface (GUI) is developed using the Camile software. The GUI for running Configuration A1 is shown in Figure 4.2. The GUI is developed in such a manner that the configurations of the filters can be changed easily. The first-order filter can be used on the PV by pressing the “Select S2” and “Select F2” buttons on the GUI.

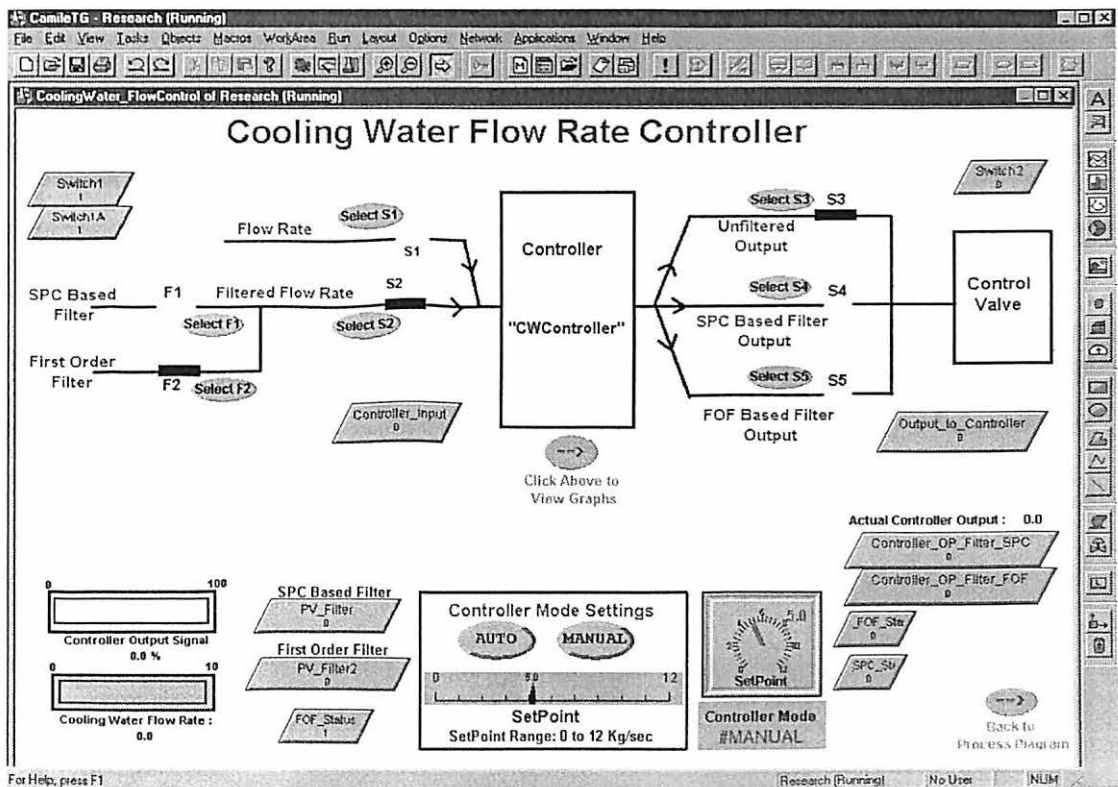


Figure 4.2: Camile GUI for Configuration A1

Configuration A2: CUSUM Filter on the PV

Figure 4.3, shows the general block diagram of Configuration A2.

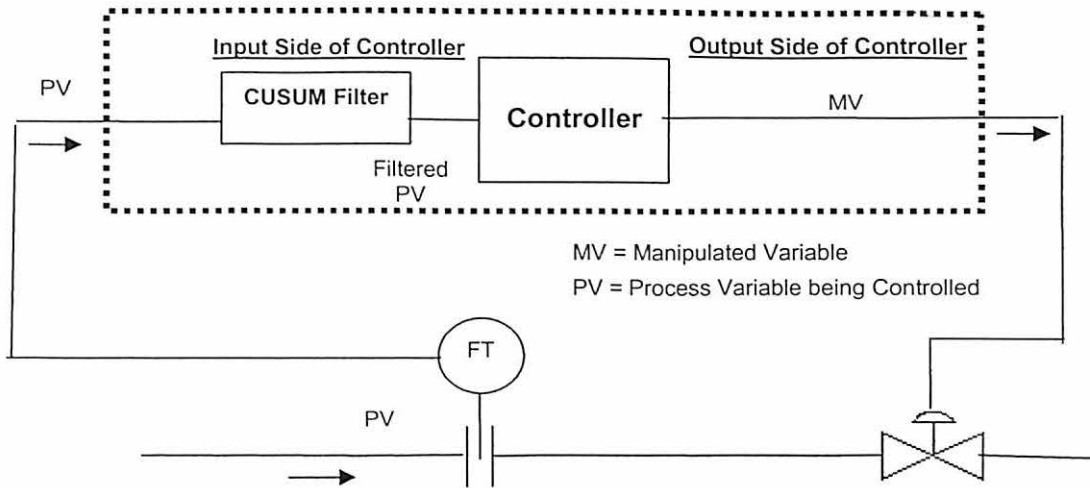


Figure 4.3: Block diagram for Configuration A2

The CUSUM filter can be used on the PV by pressing the “Select S2” and “Select F1” buttons on the GUI. The GUI for running Configuration A2 is shown in Figure 4.4.

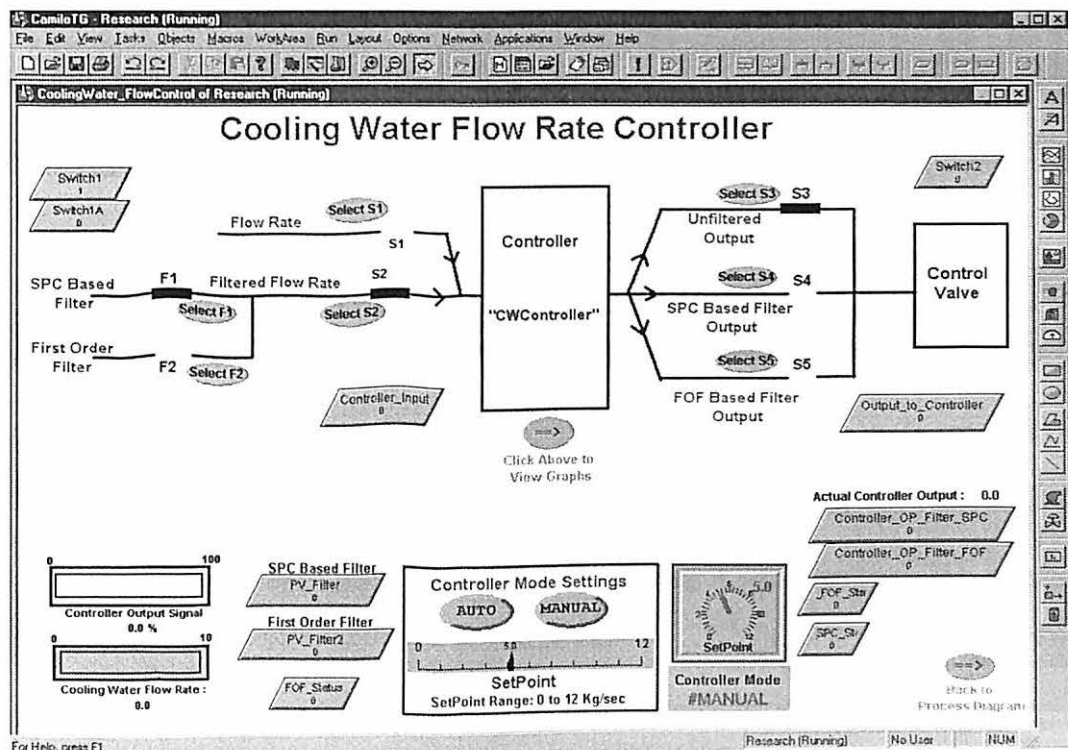


Figure 4.4: Camile GUI for Configuration A2

Configuration B: In Configuration B, the true noisy signal is given to the controller. Rather than filtering the noisy signal, the output of the controller is filtered. Both filters are also explored on the Manipulated Variable (MV).

Configuration B1 - First-Order Filter on MV: Figure 4.5, shows the general block diagram of Configuration B1.

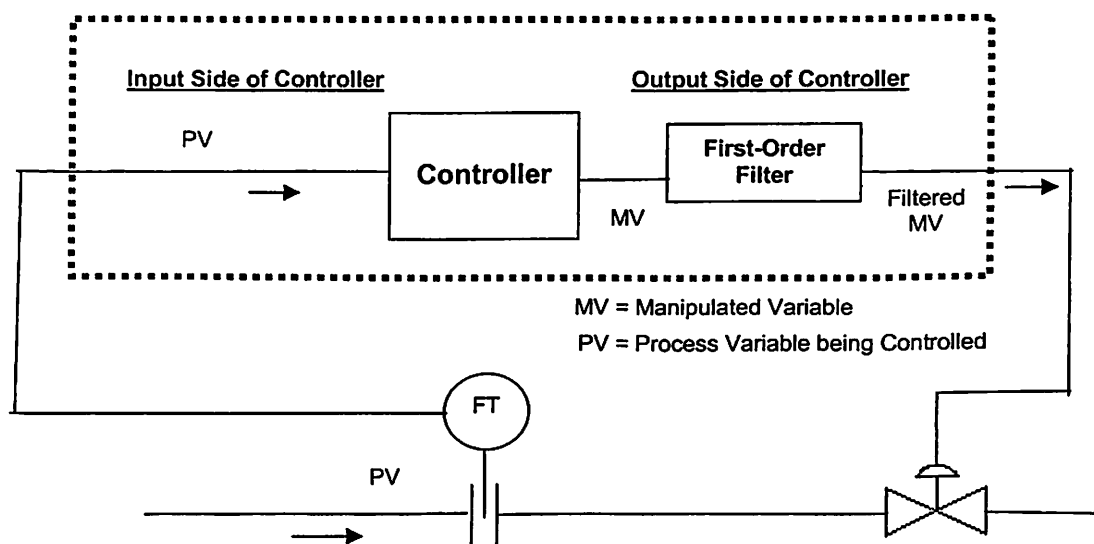


Figure 4.5: Block diagram for Configuration B1

The first-order filter can be used on the MV by pressing the “Select S1” and “Select S5” buttons on the GUI. The GUI for running Configuration B1 is shown in Figure 4.6.

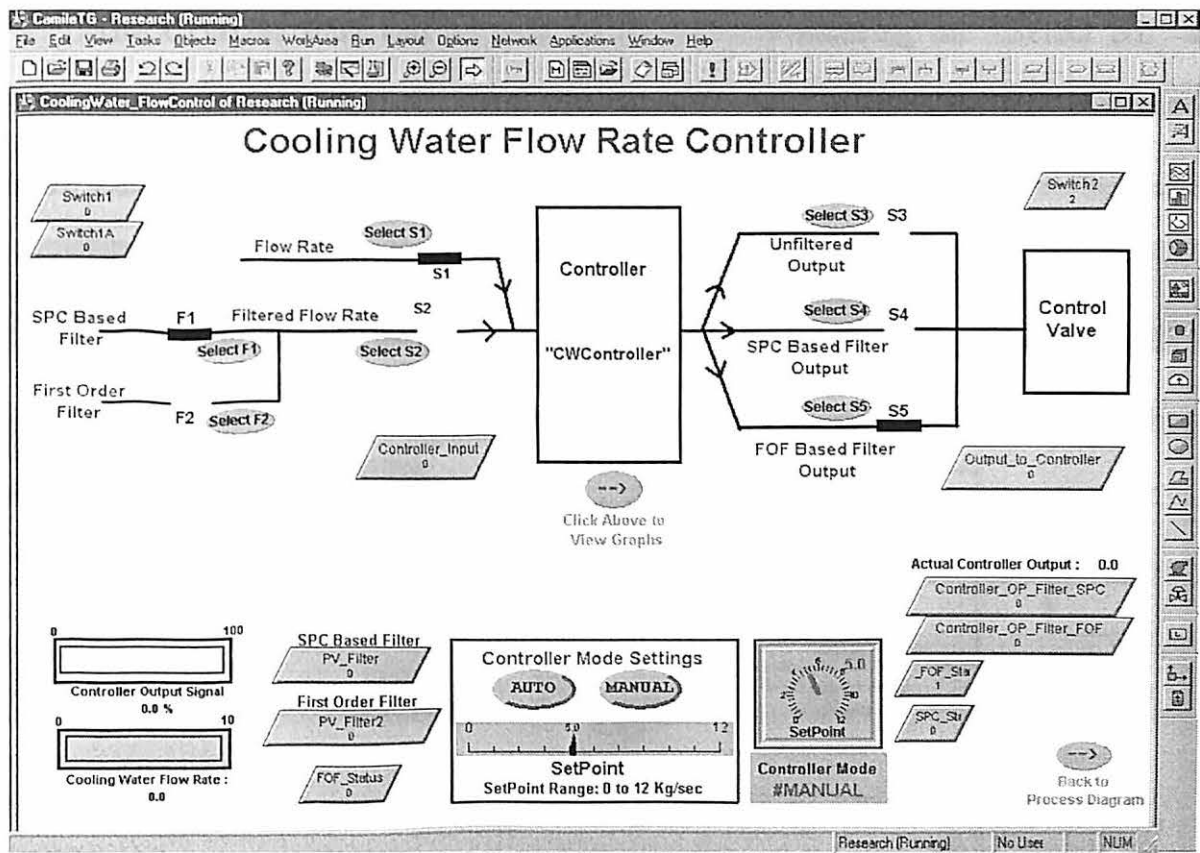


Figure 4.6: Camille GUI for Configuration B1

Configuration B2: CUSUM on MV: Figure 4.7 shows the general block diagram of Configuration B2.

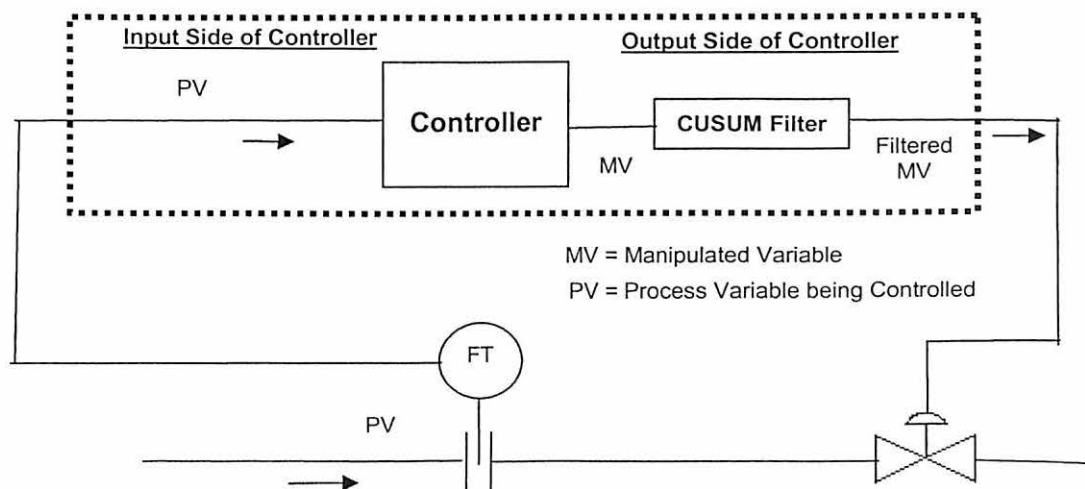


Figure 4.7: Experimental setup for Configuration B2

The CUSUM filter can be used on the MV by pressing the “Select S1” and “Select S4” buttons on the GUI. The GUI for running Configuration B2 is shown in Figure 4.8.

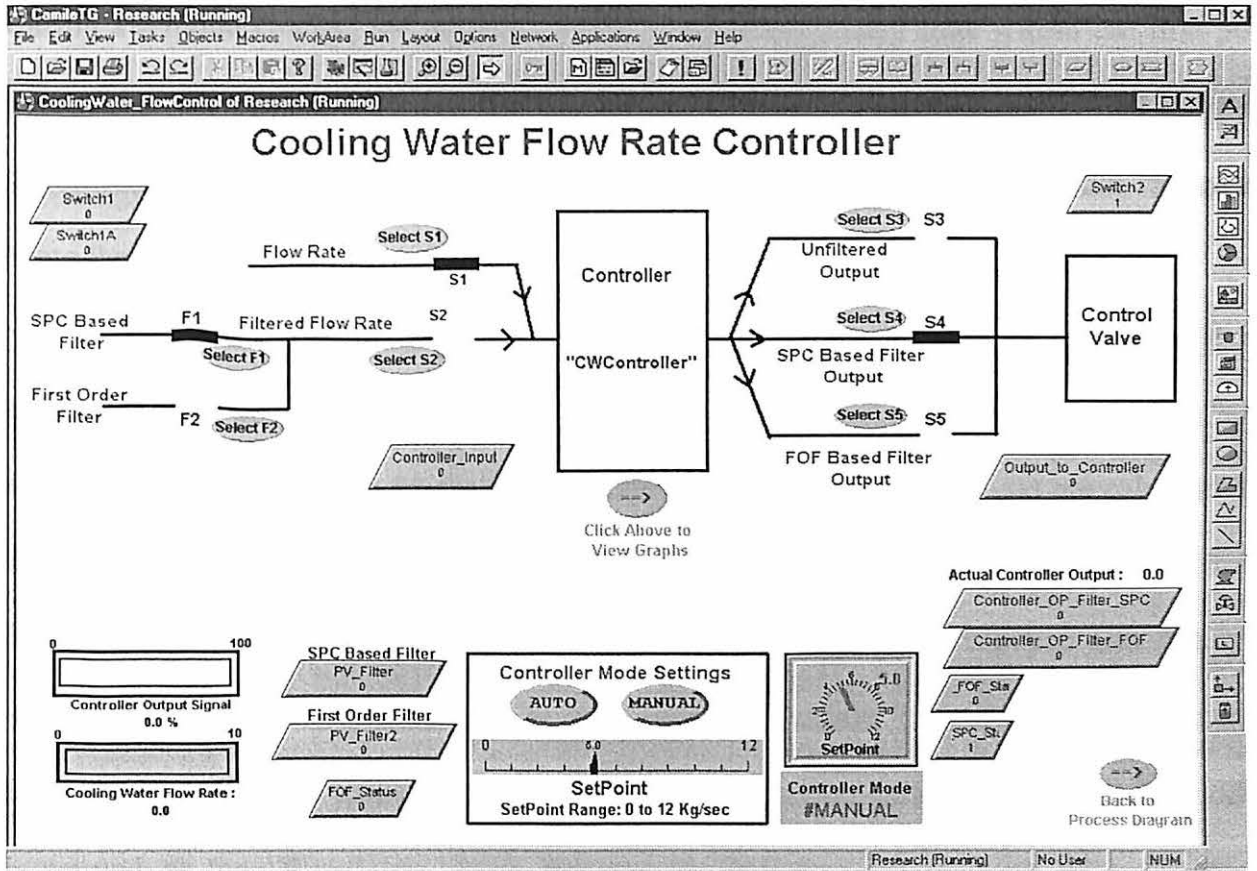


Figure 4.8: Camile GUI for Configuration B2

4.2: Description of Test: Procedure

The controller is tuned without using any filter by the LSU ITAE Tuning procedure [5]. The resultant controller obtained is an aggressive controller with minimum ISE values. The sampling time was set to 200 milliseconds.

One of the above two configurations and one of the two filters are selected. The controller is set to Automatic. The cooling water flowrate setpoint is changed from

0 kg/min to 5 kg/min. The transient response data is logged for 60 seconds after the setpoint change has occurred.

During the experimentation it was found that for almost all the configurations, during a setpoint change, the process variable reached the steady state within the first 30 seconds of the response. Hence, the first 30 seconds of the response is termed as the “Transient Period” and the second 30 seconds of the response is termed as the “Steady State Period”.

As discussed in Section 2.11, ISE and valve travel are used as a metrics for quantifying the goodness of the filters. The ISE value for the Transient Period (first 30 seconds) and the ISE value for the Steady State Period (second 30 seconds) are calculated from the experimental data, using the rectangular rule of integration. The ISE of the Transient Period contains the properties of the both the transient response and steady state response. In order to isolate the properties of transient response, the difference between these ISE’s is termed as “Servo ISE”. The ISE value for the Steady State Period is termed as the “Regulatory ISE”. ISE can be thought as a measure of controller performance. The lower the ISE value, the better the controller performance in both the Setpoint tracking (Servo ISE) and noise attenuation (Regulatory ISE).

The valve travel is calculated for the Transient Period and Steady State Period. Similar to the ISE, the Transient Period valve travel contains the properties of both the transient and steady state. Hence, in order to isolate the properties of transient phase only, the difference between the transient and steady state valve travel is found and termed as the “Servo Valve Travel”. The valve travel calculated for the steady state period is termed as the “Regulatory Valve Travel”. Valve travel can be thought as a measure of

wear and tear of a control valve. Therefore, it is desired to have lower valve travel since lower valve travel can be interpreted as lower wear and tear of the control valve.

The filtering parameters, i.e. the λ value for the first-order filter and the trigger value for CUSUM filter, are varied and the same tests are repeated. The logged data obtained from Camile is saved to a text file. This text file is then exported to Excel. Corresponding to each test, the following values are calculated using Excel.

- a) Servo Valve Travel
- b) Regulatory Valve Travel
- c) Servo ISE
- d) Regulatory ISE

Because of normal experimental variability, each test was replicated 3 times for each configuration and filter tuning value. For instance; Figures 4.9 and 4.10 show an experimental run with the CUSUM filter on the MV (Configuration B2), with a Trigger value of 1.5. The first graph in Figure 4.9 is a plot of the noisy process variable and setpoint with respect to time. The second graph in Figure 4.9 is a plot of the controller output with respect to time. Figure 4.9 captures the first 30 seconds of the response while Figure 4.10 captures the second 30 seconds of the response.

Transient Period (first 30 seconds)

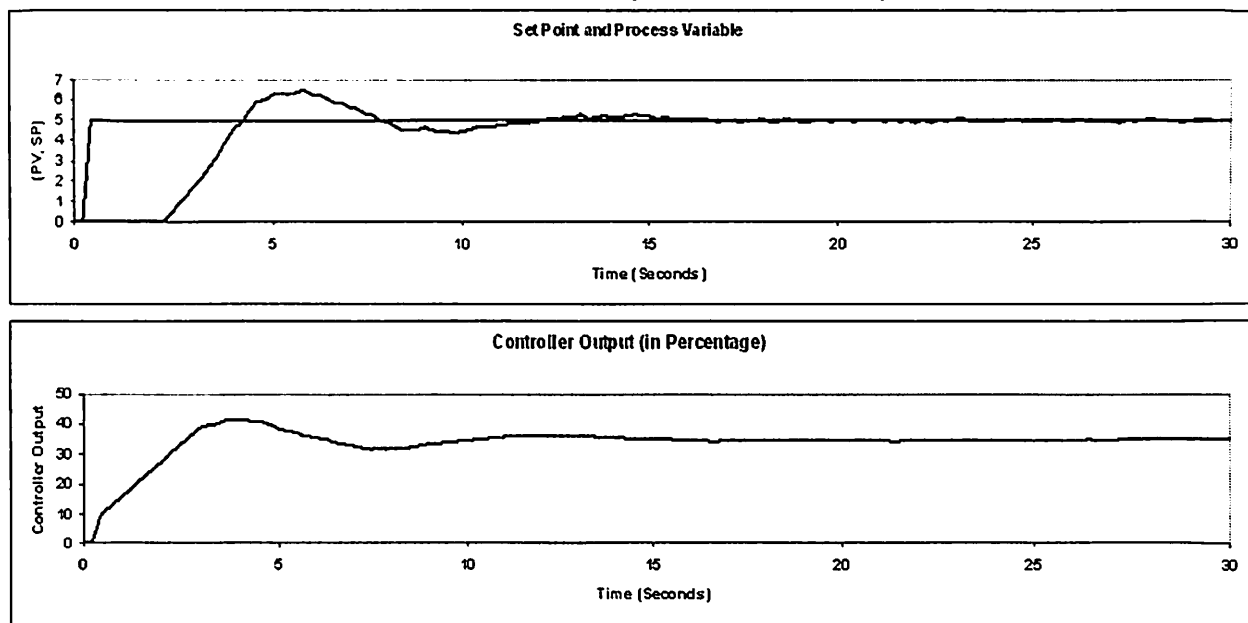


Figure 4.9: Transient period of controller with CUSUM filter on MV (trigger 1.5)

Steady State Period (30 seconds to 60 seconds)

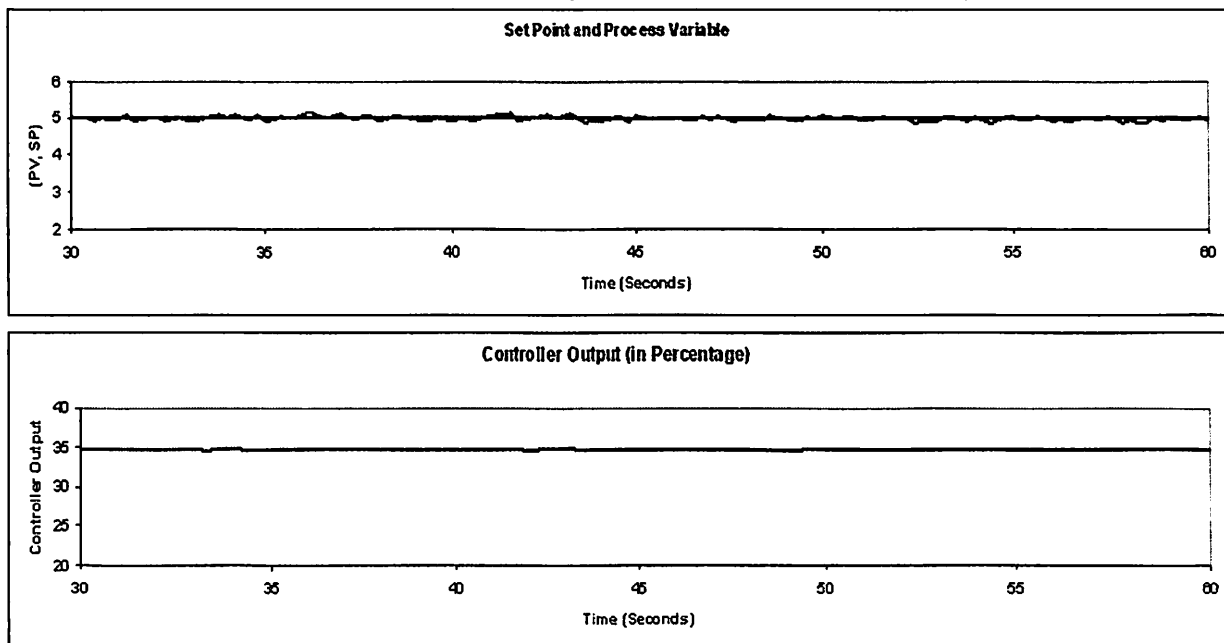


Figure 4.10: Steady state period of controller with CUSUM filter on MV (trigger 1.5)

4.3: Servo Mode

The experimental results of the tests conducted for servo mode for all configurations are shown as a graphical plot in Figure 4.11. The horizontal axis represents the servo valve travel value. Low valve travel reduces the wear and tear of the control valve. Hence, lower valve travel is labeled as “Better” and higher valve travel is marked as “Worse”. The vertical axis represents the Servo ISE value. Lower ISE can be interpreted as better controller performance. Hence, a lower ISE value is labeled as “Better”, and vice versa.

Various experiments were performed. For instance, a filter tuning parameter value and a filter configuration was selected, the setpoint change implemented, data obtained, and the corresponding ISE and valve travel were calculated. The result of one experimental run is shown as one single point in Figure 4.11. Appendix 2 contains the tabulation of all the data points.

- The squares represent the data corresponding to first-order filter on the PV (Configuration A1).
- The diamonds represent the data corresponding to CUSUM filter on the PV (Configuration A2).
- The “X” symbols represent the data corresponding the first-order filter on MV (Configuration B1).
- The triangular points represent the data corresponding to CUSUM filter on the MV (Configuration B2).

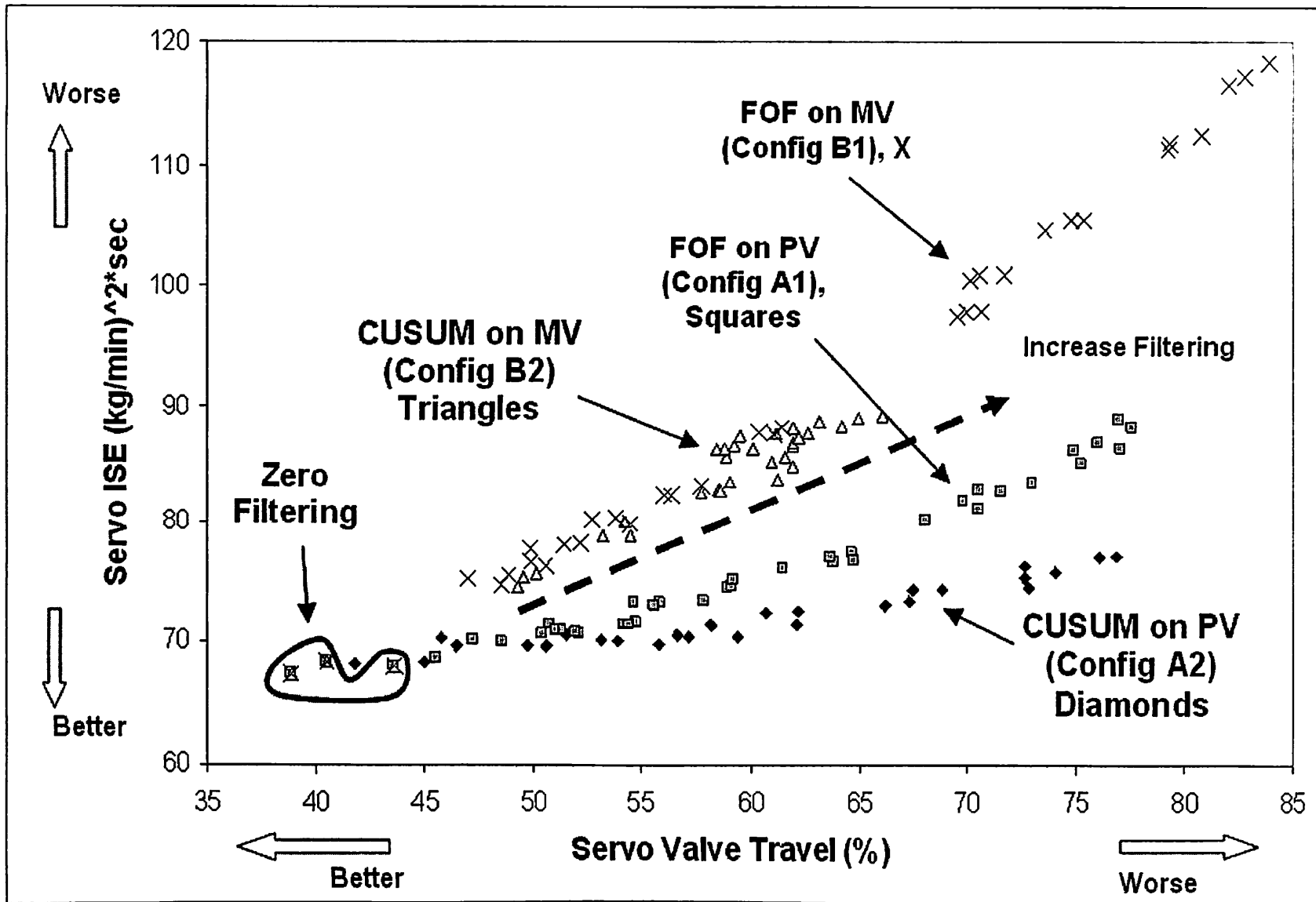


Figure 4.11 : Results of Configuration A and B for servo

Comments: When there was a setpoint change, the best ISE valve travel characteristics were obtained when no filtering was done (since no filtering has the smallest valve travel and ISE). Recall that the controller was tuned for aggressive behavior. Adding a filter either in the input side or in the output side of the controller resulted in degradation of controller performance. As a result the controlled system response became more oscillatory, resulting in a simultaneous increase in both the ISE and the valve travel. The scatter in the data indicates replicate variability. The normal replicate range on ISE is about $3 (Kg/min)^2 sec$ and that on valve travel is about 5%. However, the separate configurations are visually distinct within the variability.

Configuration A1:

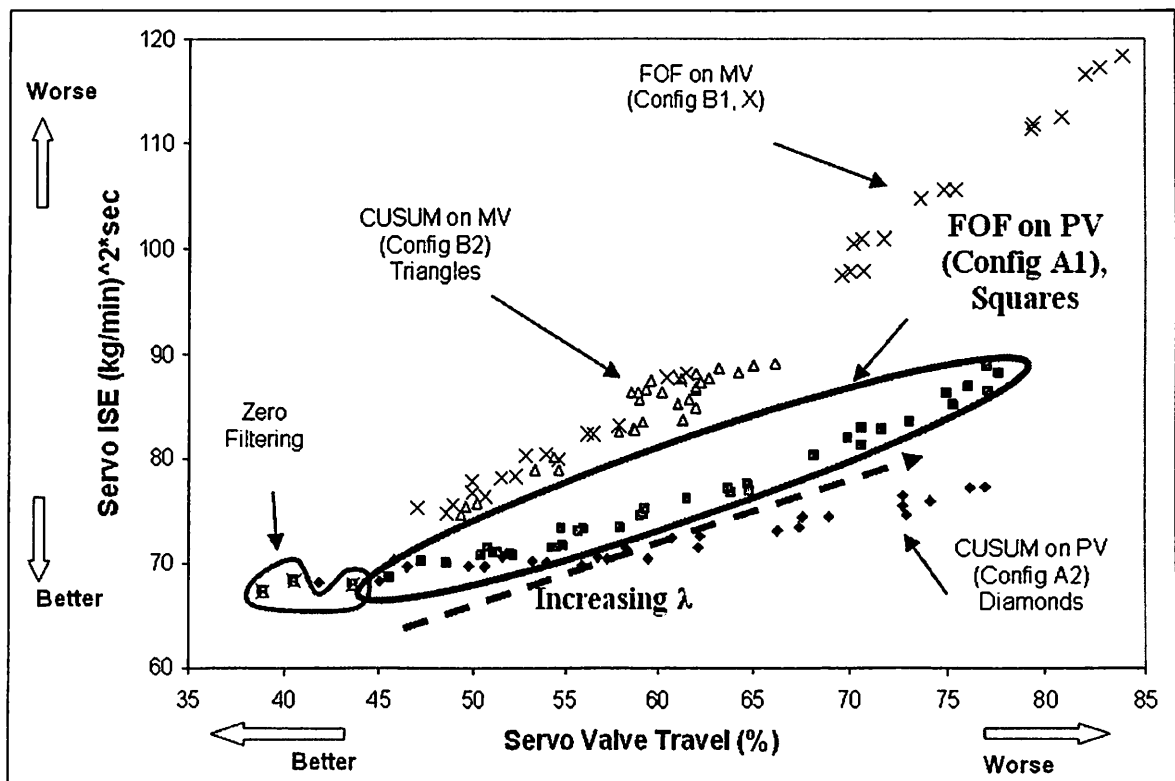


Figure 4.12: Servo - Configuration A1 highlighted

In Configuration A1 a first-order filter is used on the process variable (illustrated by solid squares). When the filtering was increased by increasing the λ value, both the ISE and valve travel worsened. Figure 4.13 and Figure 4.14 show that by increasing λ , the controller performance degraded as evidenced by more oscillations.

The upper graph in either figure is a plot of the noisy process variable and setpoint with respect to time. The lower graph is a plot of the controller output with respect to time.

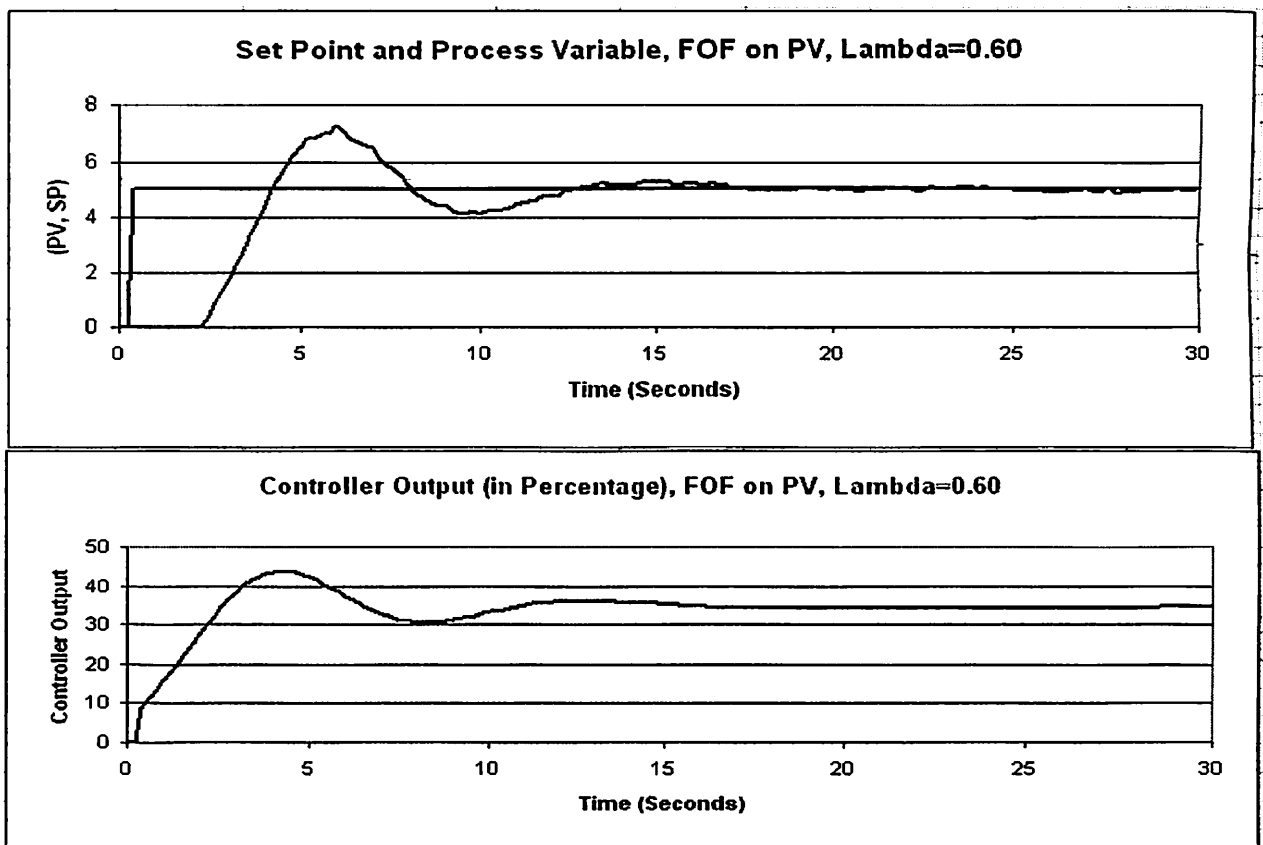


Figure 4.13: Configuration A1 - servo, $\lambda=0.60$

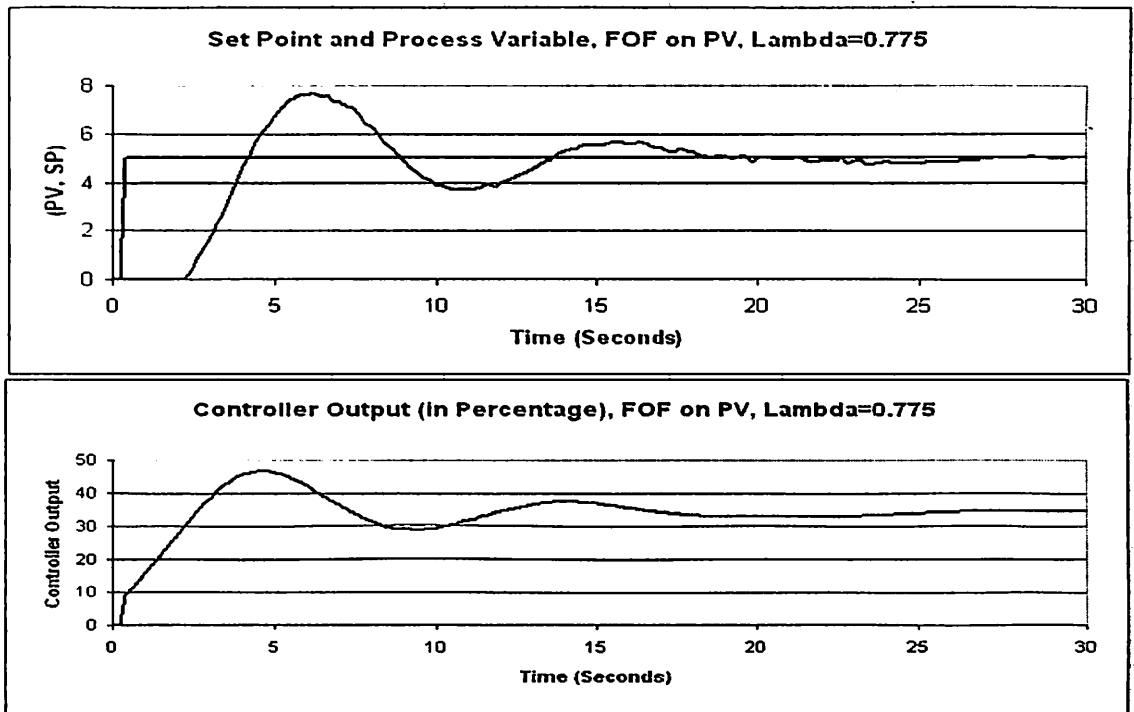


Figure 4.14: Configuration A1 - servo, $\lambda=0.775$

Configuration A2:

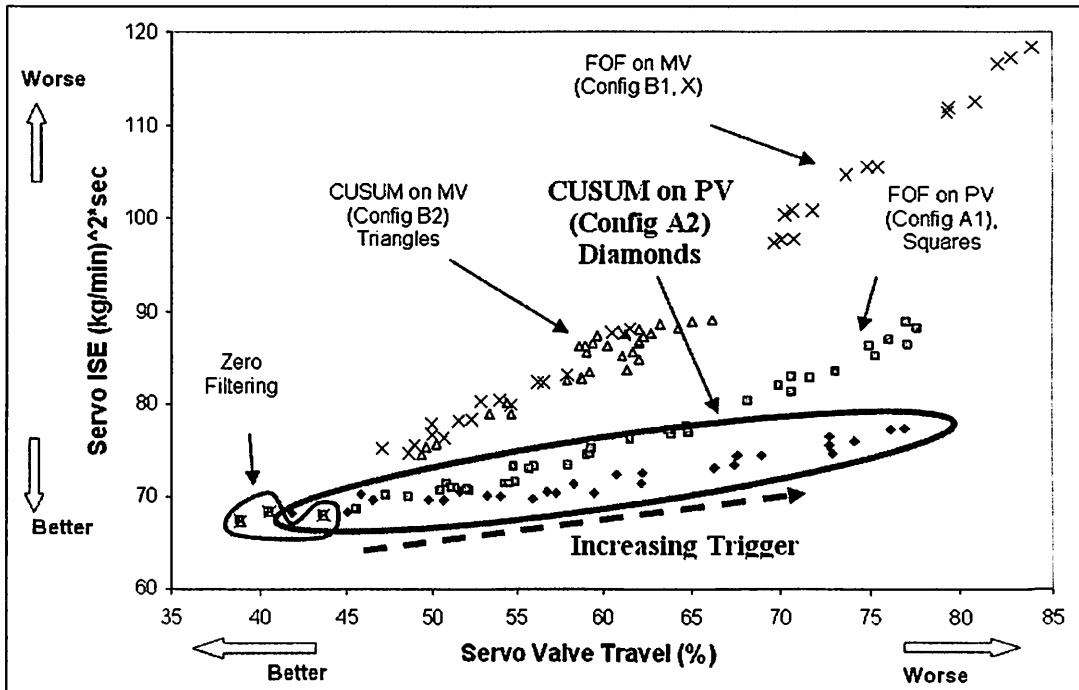


Figure 4.15: Servo - Configuration A2 highlighted

Points in Figure 4.15 represented by solid diamonds represent a CUSUM Filter on the input side of the controller (Configuration A2). It can be noticed from the graph that, when we increase the filtering by increasing the trigger value, the ISE/Valve travel characteristics worsen, but the average ISE values are better than the rest of the filters. Accordingly, this configuration showed the best ISE/Valve travel characteristics. If we wanted to use a filter during a servo response, then CUSUM filter on the input side will be a good option.

Figure 4.16 and Figure 4.17 show the effect of increasing trigger on controller performance. From the graphs it can be noticed that changing the trigger from 1.0 to 3.0 degrades the controller performance and results in more oscillations in the process variable. Therefore, it increases valve travel and ISE.

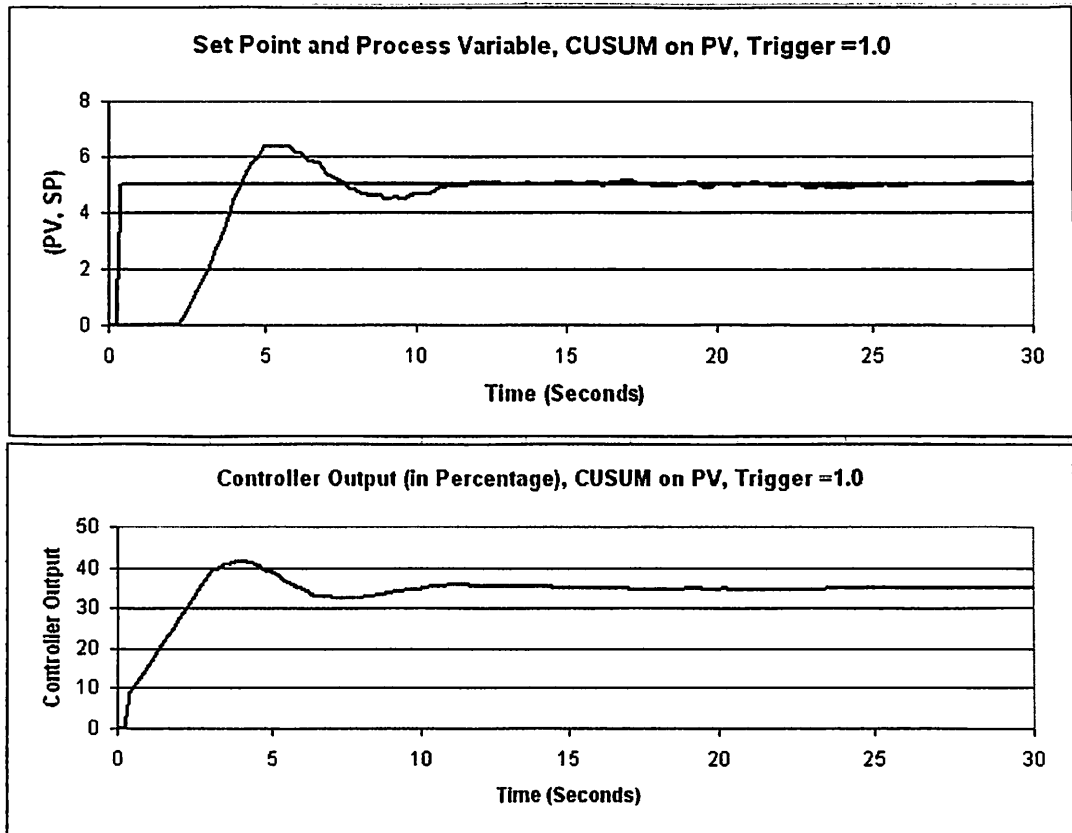


Figure 4.16: Configuration A2 - servo, trigger=1.0

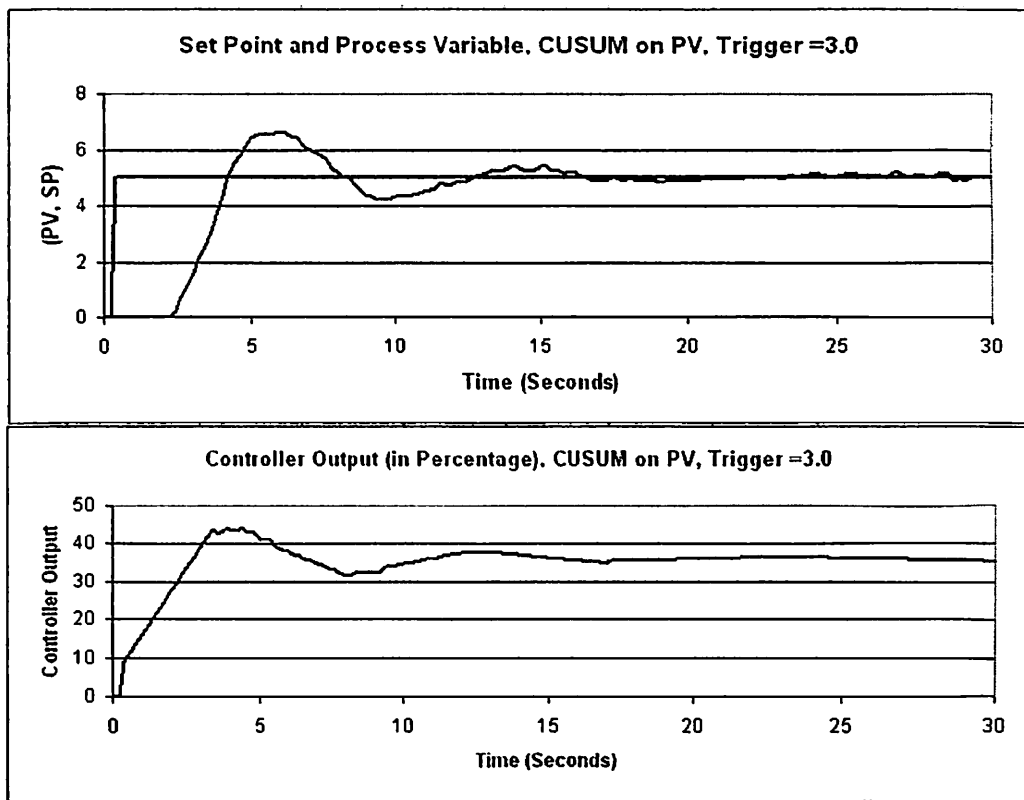


Figure 4.17: Configuration A2 - servo, trigger =3.0

Configuration B1:

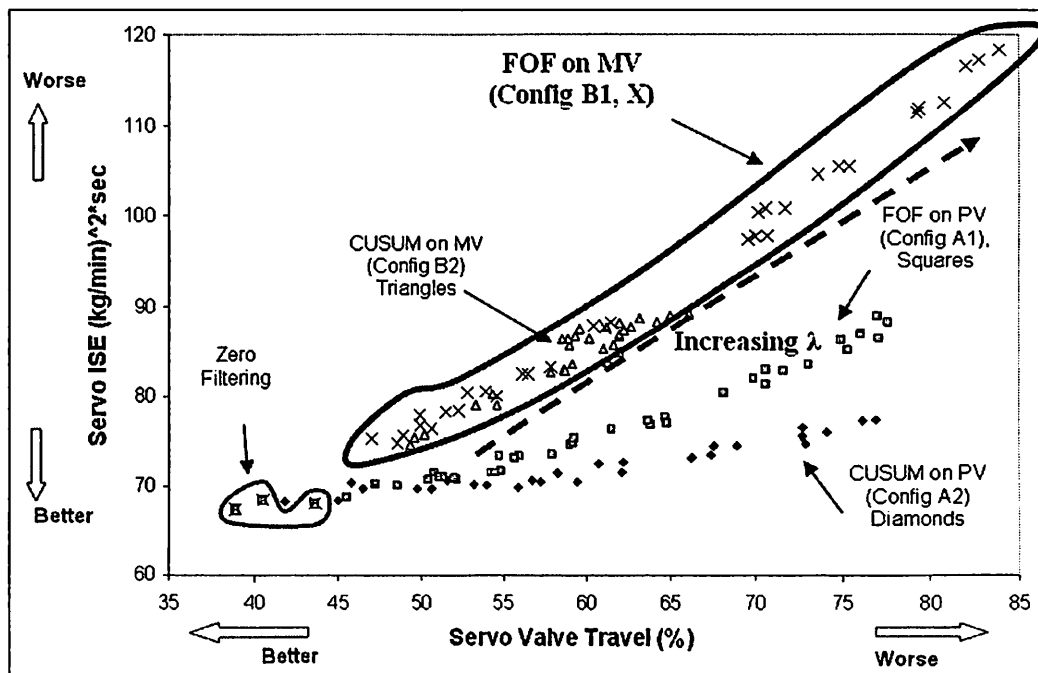


Figure 4.18: Servo - Configuration B1 highlighted

The data corresponding to first-order filter on the output side of the controller (Configuration B1), are represented by “X” points. It is evident from Figure 4.18 that the average value of the ISE for the different tuning parameters for this configuration is very high as compared to other configurations. For a small rise in the λ value (0.4 to 0.8), there is large rise in both ISE (80 to $117(Kg/min)^2 sec$) and valve travel (52 to 82%). The controller performance deteriorates, and hence, produces more oscillations resulting in increase in both ISE and valve travel.

Figure 4.19 and Figure 4.20 show the effect of increasing λ on controller performance. From the graphs it can be noticed that changing the λ from 0.60 to 0.825, makes the controller performance deteriorate; and hence, it increases valve travel and ISE. The process variable in Figure 4.20 is more oscillatory than in Figure 4.19.

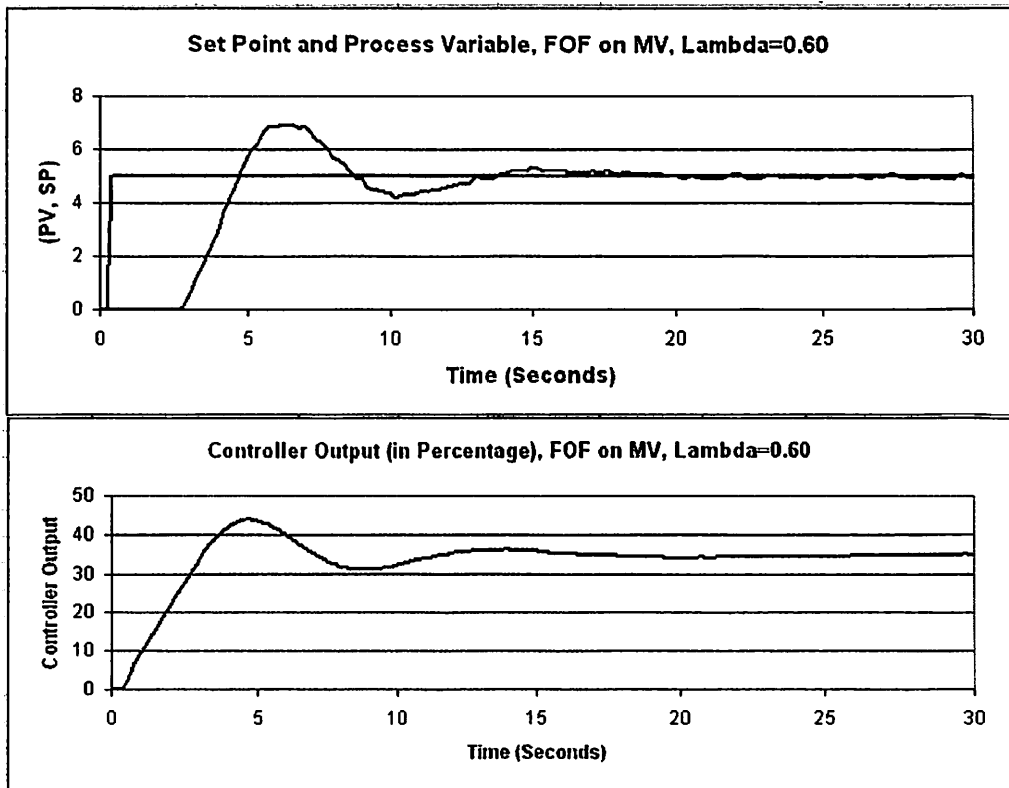


Figure 4.19: Configuration B1 - servo, $\lambda=0.60$

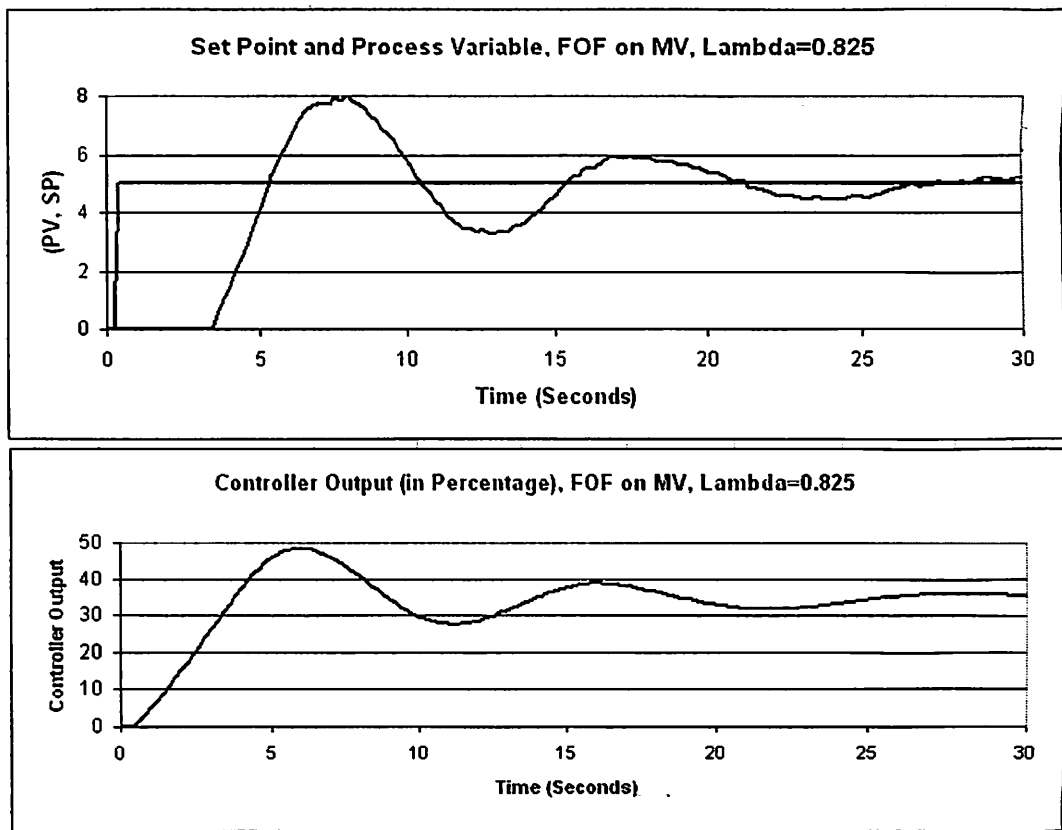


Figure 4.20: Configuration B1 - servo, $\lambda=0.825$

Configuration B2:

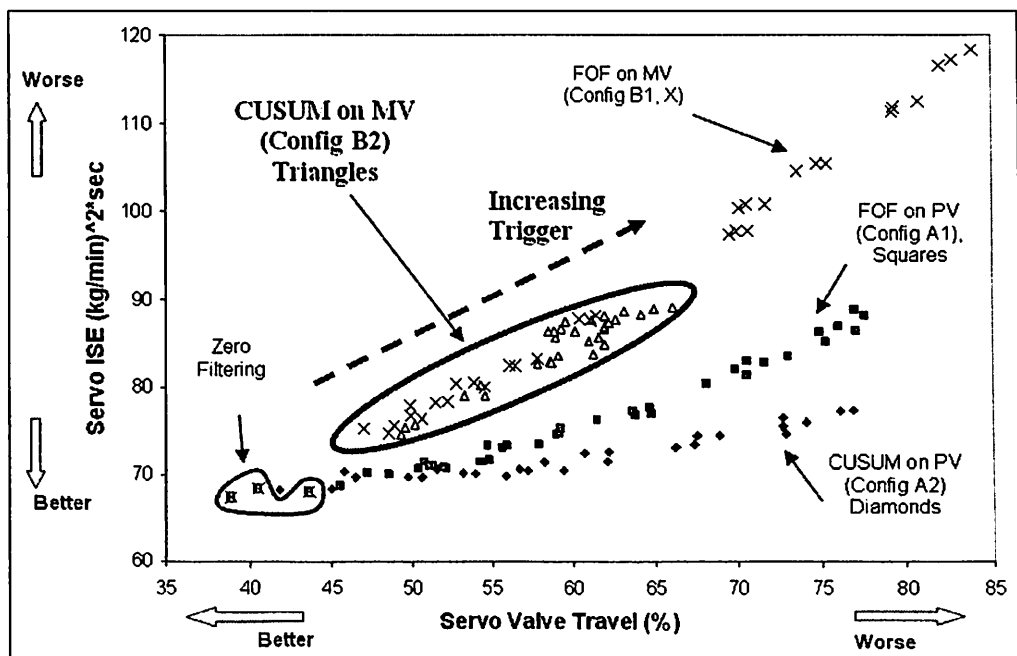


Figure 4.21: Servo - Configuration B2 highlighted

Adding a CUSUM Filter in the output side of the controller (Configuration B2, Figure 4.21) shows higher values of ISE compared to Configuration A1 and A2 for a given value of valve travel. The data points of this configuration are shown as solid triangles in Figure 4.21. As the filtering is increased by increasing the Trigger value (1 to 3), there is a not much of a rise in the valve travel (50 to 59%) but there is a big rise in the ISE valve (75 to 86 $(Kg/min)^2$ sec). This is due to the fact that, the higher the trigger value, the greater the deadtime in the filtered controller output. It can be generalized that having a filter on the MV (Configuration B1 and B2) results in higher average values of ISE during a Servo response for an aggressively tuned controller.

Figure 4.22 and Figure 4.23 show the effect of increasing trigger on controller performance. From the graphs it can be noticed that changing the trigger from 1.0 to 3.0, deteriorates the controller performance, and hence, it increases valve travel and ISE. Also there is an increased deadtime in filtered controller output (filtered MV).

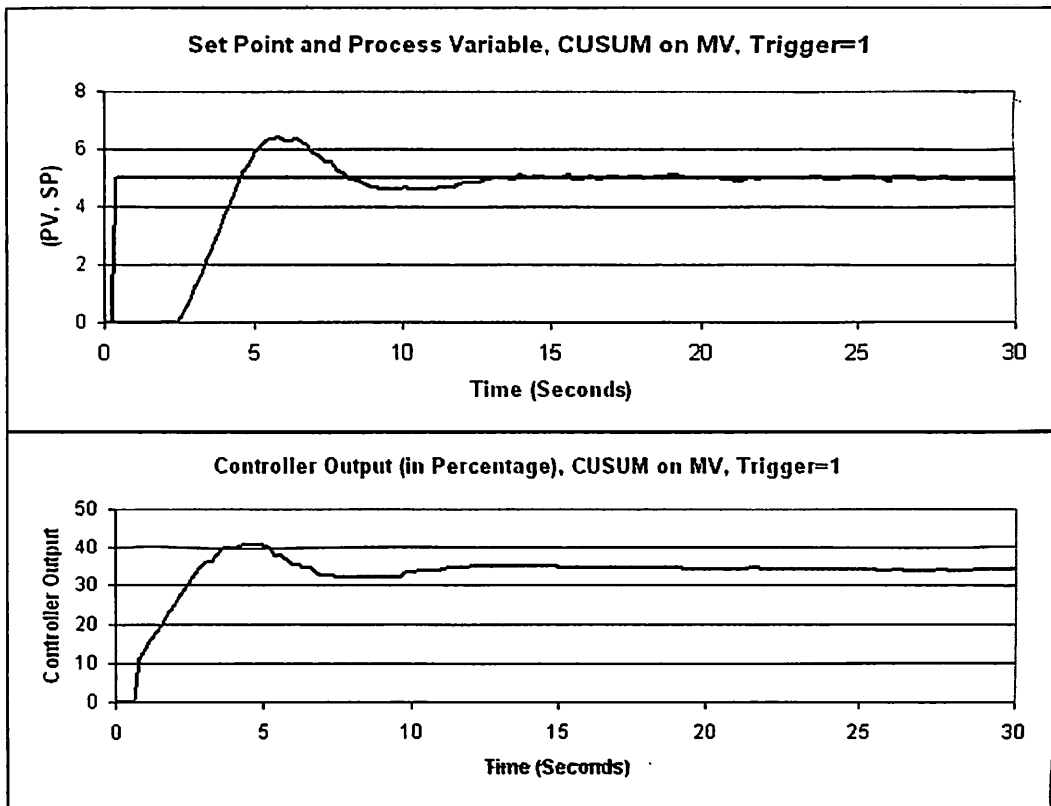


Figure 4.22: Configuration B2 - servo, trigger =1.0

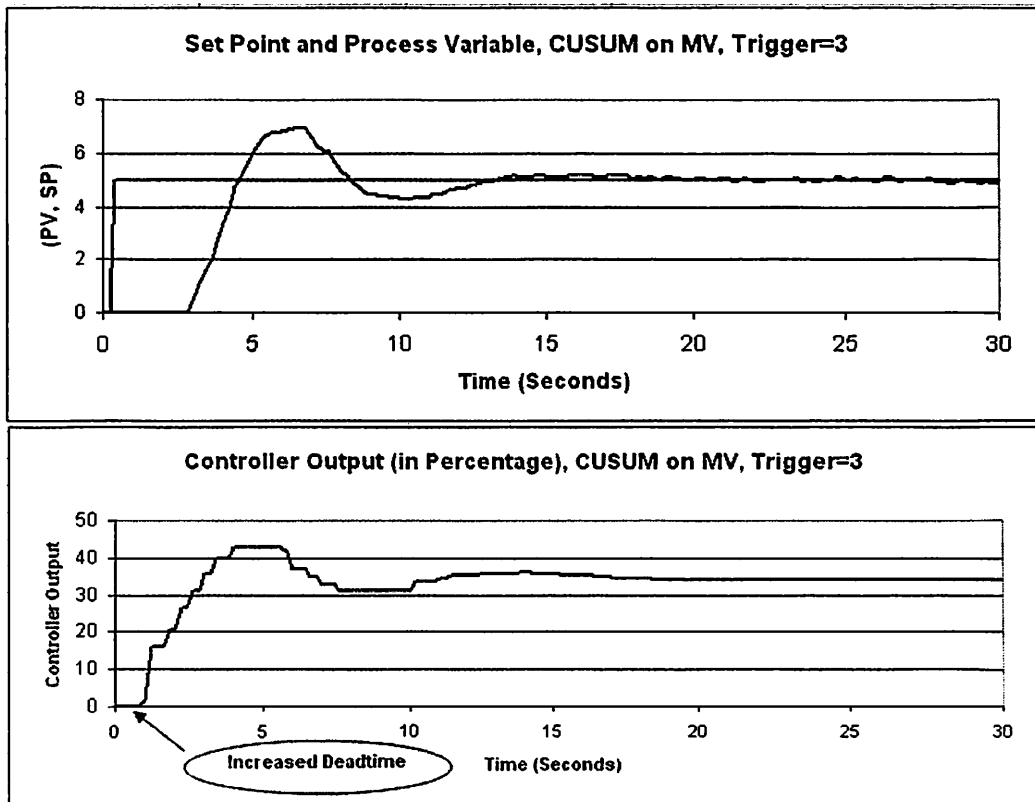


Figure 4.23: Configuration B2 - servo, trigger =3.0

Comments: It is suspected that the balance of ISE and valve travel which results from disturbances would be similar to this servo mode results. Unfortunately, the experimental setup was unable to provide reproducible disturbance events for testing.

4.4: Regulatory Mode

The experimental results of the tests conducted for regulatory mode for all configurations are shown as a graphical plot in Figure 4.24. The horizontal axis and the vertical axis and the data points represent the same parameters as for servo mode plot (Figure 4.11). The regulatory mode experiments did not represent an ideal noisy steady state since there were continual line pressure disturbances affecting the flow rate and autocorrelation in the noise was present.

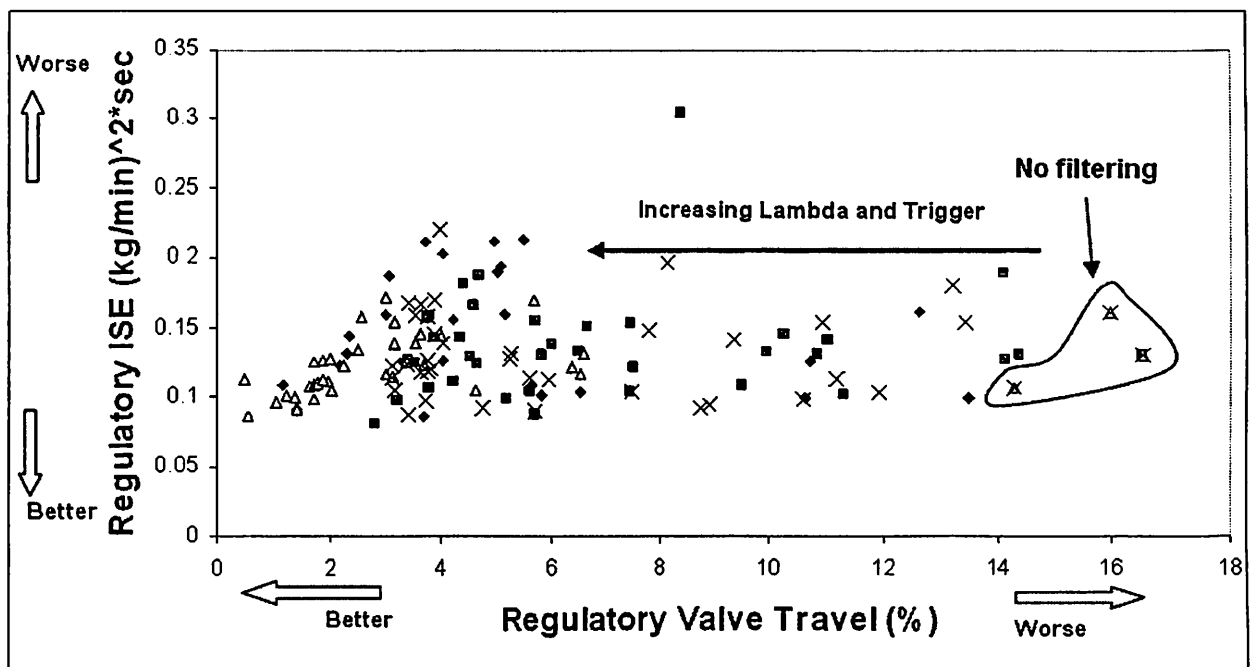


Figure 4.24: Regulatory response

Figure 4.24 shows that when no filters are used in the input side or the output side of the controller, higher valve travel (a value of 16%) is obtained than when filters were added.

Configuration A1: In Configuration A1, first-order filter on the PV, represented by solid squares, increasing the value of λ from 0 to 0.65 resulted in a reduction of valve travel from about 16% to 4% with no apparent change in ISE. If the λ value is further increased, there is an increase in ISE value from about 0.08 to 0.2 (Kg/min)² sec but the valve travel does not reduce any further. The valve travel is around 3% for high filtering values.

Configuration A2: In Configuration A2, i.e. CUSUM filter on the PV, represented by solid diamonds, as the trigger value is increased from 0 to 3, there is a reduction in valve travel (14% to 2%) and the ISE stays almost constant (about 0.14 (Kg/min)² sec). The valve travel reaches very low values of about 2%. If the trigger value is further increased, the ISE value increases and the valve travel also starts to increase.

Configuration B1: In Configuration B1, i.e. first-order filter on the MV, represented by “X” points, as the value of λ is increased from 0 to 0.75, there is reduction in valve travel and the ISE stays almost constant. The valve travel goes as low as 3.75% and stays there for even higher λ values.

Configuration B2: In Configuration B2, i.e. CUSUM filter on the MV, represented by solid triangles, as trigger value is increased from 0 to 3.5, there is a great reduction in valve travel, while ISE stays almost constant. The valve travel is about 1.5% for most of

the trigger parameters. Further increasing the trigger keeps the valve travel and ISE in the same range. This configuration has the least valve travel values.

Comments: It is noteworthy that experimental variability masks any differences between filters or configurations expect for those with lowest valve travel values.

4.5: Comparison of Results with Paper – “A CUSUM type on-line filter” [7]

Similar work has been done by Dr R. Russell Rhinehart in his paper titled “A CUSUM type on-line filter” [7]. In that paper, the author had simulated Configuration A1 and B2 and had compared the performances of the filters.

In the simulation used in the paper, the base process model was a first-order system which was slightly nonlinear. The process was subjected to a second-order autoregressive moving average input disturbance and the measured value included an additive Gaussian distributed noise. The manipulated variable was constrained and the measurement noise level was changed in between the simulation run. The way the controller was tuned (aggressive/conservative/tuned with filter/tuned without filter) was not mentioned in the paper. The simulation run consisted of a setpoint change followed by a change in level of noise and the manipulated variable hitting a constraint. The valve travel and ISE was measured for the sequence of events and the performance of Configuration A1 and B2 were compared. It was noticed that, in Configuration A1 (first-order on PV) used in the paper, as the filtering was increased there was continuous decrease in valve travel while, the ISE value decreased a bit and then started to rise. In

case of Configuration B2 (CUSUM on MV), as the filtering was increased, there was continuous reduction in valve travel and slight increase in ISE value.

The performance of Configuration A1 and B2 are clearly different from that obtained from the experimental tests conducted in this study. In the experimental setup of this work, the controller was tuned (with no filter on PV or MV) to be aggressive, and the valve travel and ISE values were characterized separately for servo and regulatory response (unlike the paper, where they were combined). For a step change in setpoint the valve travel and ISE always increased for increase in filtering which clearly contradicts the results shown in the paper. This is expected, as the paper had a different setup of testing the filters.

CHAPTER 5

SIMULATION SETUP

The cooling water flow loop was simulated using Simulink. This chapter contains the discussion of the creation of Simulink models representing the flow loop.

5.1: Flow Loop Process Model in Simulink

The process model is stored as a masked subsystem as shown in Figure 5.1.

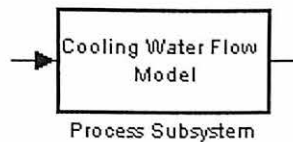


Figure 5.1: Process model subsystem

Figure 5.2 shows the information that is displayed, when the user clicks on the “Cooling Water Flow Model” subsystem.

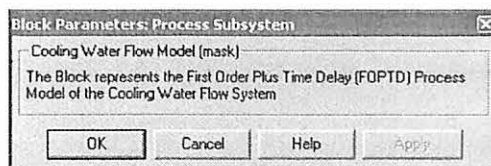


Figure 5.2: Information window for “process model subsystem”

The actual model stored inside the masked subsystem (Figure 5.1) is shown in Figure 5.3. This model is obtained from the Process Reaction Curve analysis discussed in Section 3.3.

Cooling Water Flow System Process Model

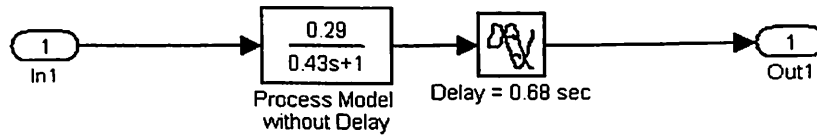


Figure 5.3: Simulink block for the process model

5.2: Simulink Model for a Digital PI Controller

From Section 3.4

$$\frac{u(z)}{e(z)} = g_c(z) = \frac{1.8 - 1.33z^{-1}}{1 - z^{-1}} \quad (5.1)$$

The Simulink block of the digital controller is shown in Figure 5.4.

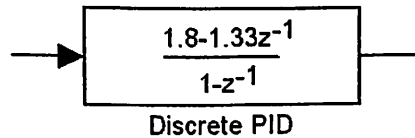


Figure 5.4: Simulink block for a digital PI controller

5.3: Simulink Model for Calculating ISE and Valve Travel

As discussed in Section 4.2, corresponding to every configuration and specific level of filtering (λ and trigger values), we have to measure the following

- a) Servo Valve Travel
- b) Regulatory Valve Travel
- c) Servo ISE
- d) Regulatory ISE

Where ISE is a measure of the controller performance and Valve Travel is a measure of the wear and tear of control valve. We have to develop Simulink models for quantifying ISE and Valve travel. The description of the models is given below.

ISE Value: The integral of the square of errors is one of the measures of performance of the controller. The error is calculated as the difference between the process variable and setpoint (not the filtered PV). The actual Simulink model stored inside the masked subsystems (explained later) is shown in Figure 5.5.

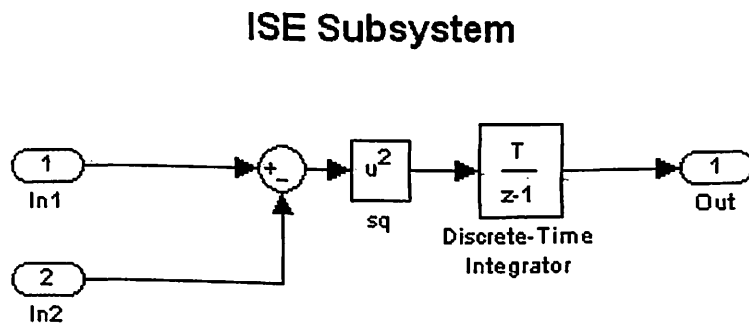


Figure 5.5: ISE Simulink block inside the masked blocks

In the cooling water system, it is observed that when a setpoint change occurs, the transient response is captured in the first 30 seconds and the steady state response is captured in the second 30 seconds.

Hence, two ISE subsystems have to be designed in such a way that one of the subsystems captures the transient ISE and the other one captures the steady state ISE.

The Simulink block diagram of the same is shown in Figure 5.6.

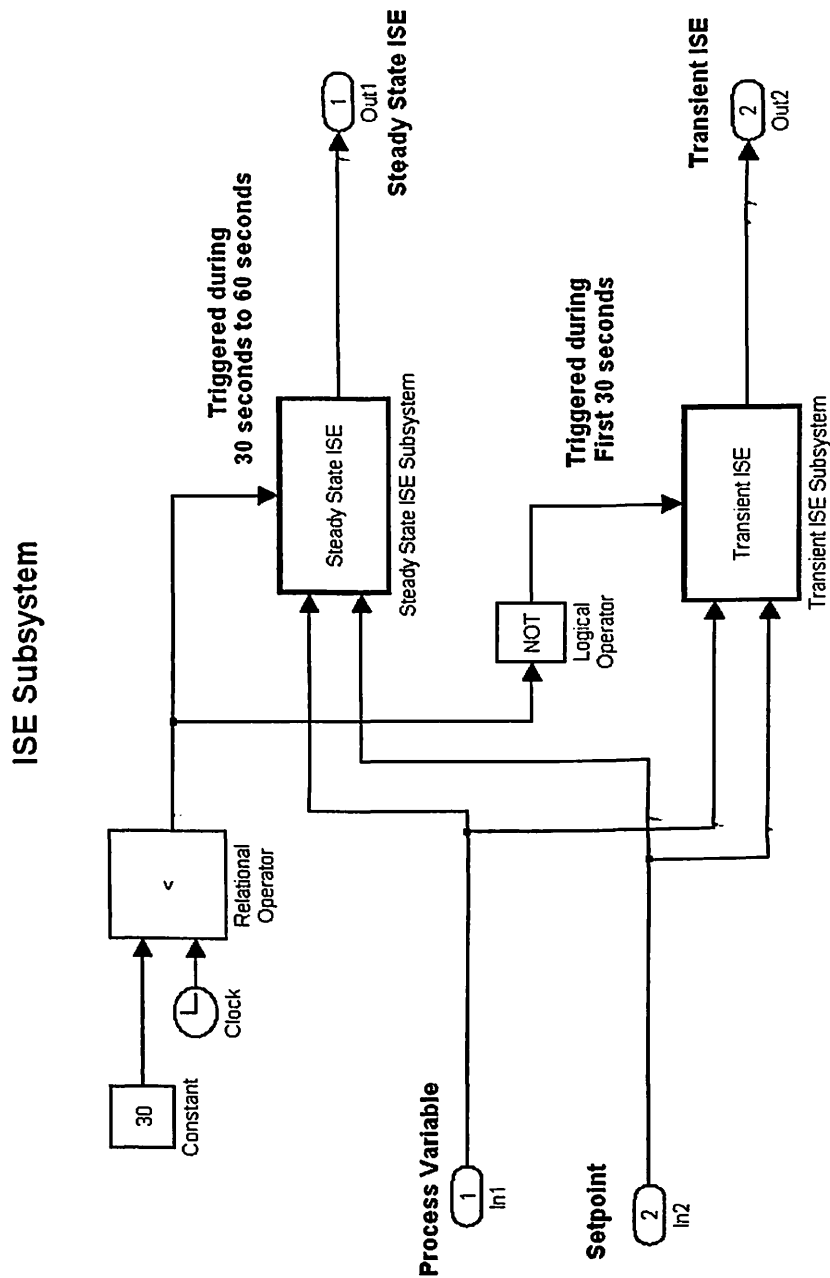


Figure 5.6: ISE subsystem to calculate transient and steady state ISE

Each ISE subsystem shown in Figure 5.6 is masked and contains the Simulink block shown in Figure 5.5. Figure 5.7 and Figure 5.8 shows the information that is displayed, when the user clicks on the masked subsystems (Steady State ISE subsystem, Transient ISE subsystem). These information windows are created to help the user understand the working of the subsystems.

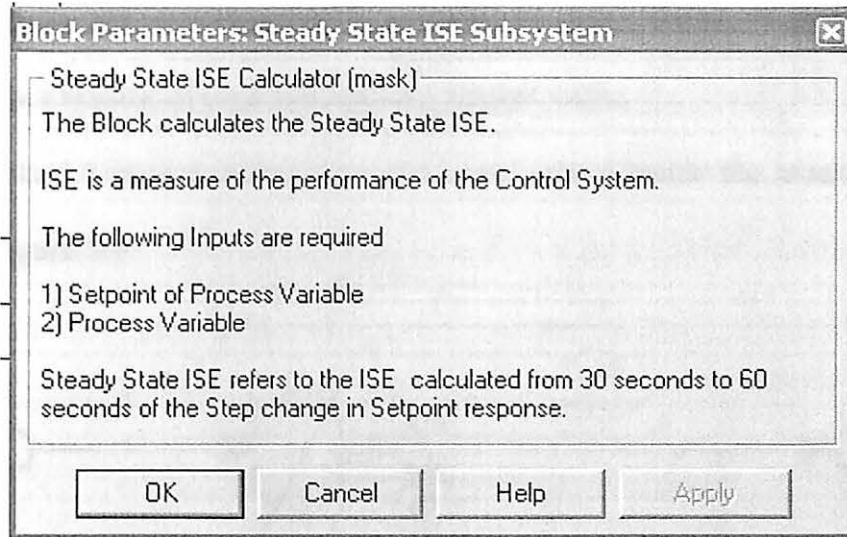


Figure 5.7: Information window for the “Steady state ISE subsystem”

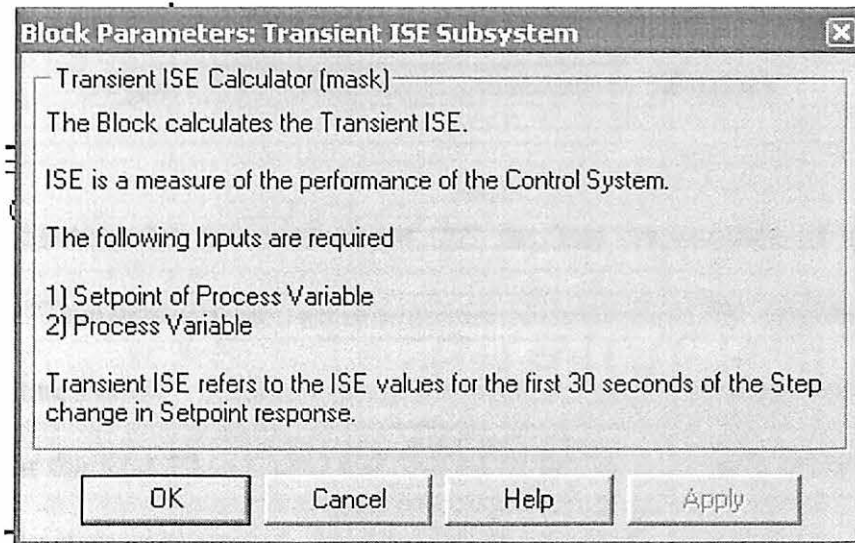


Figure 5.8: Information window for the “Transient ISE subsystem”

Similar to the discussion in Section 4.2, the difference in Transient ISE and Steady State ISE is termed as “Servo ISE”.

Valve Travel: The cumulative absolute values of difference between consecutive controller outputs are considered as a measure of valve travel. Valve travel can be considered as a measure of wear and tear of a control valve.

The actual Simulink model for valve travel stored inside the masked subsystems is shown in Figure 5.9.

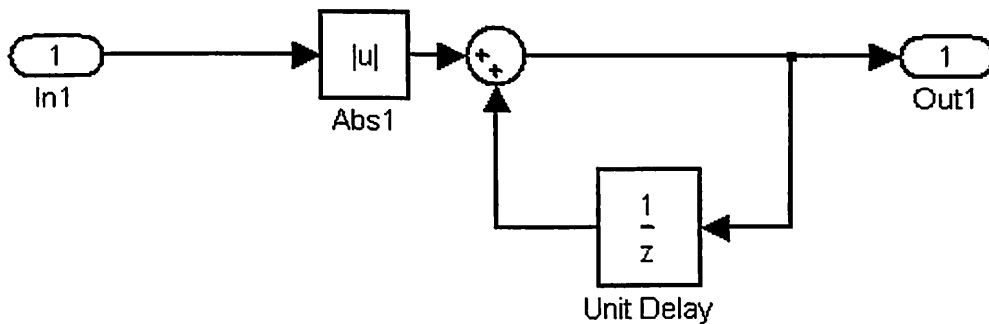


Figure 5.9: Valve travel subsystem in Simulink

From Section 4.2 it is understood that the first 30 seconds of the response is termed as the “Transient Period” and the second 30 seconds of the response is termed as the “Steady State Period”. Hence, a Simulink model is required which would capture the valve travel for the first 30 seconds (transient valve travel) and valve travel for second 30 seconds (steady state valve travel). Such a model is shown in Figure 5.10.

Valve Travel Subsystem

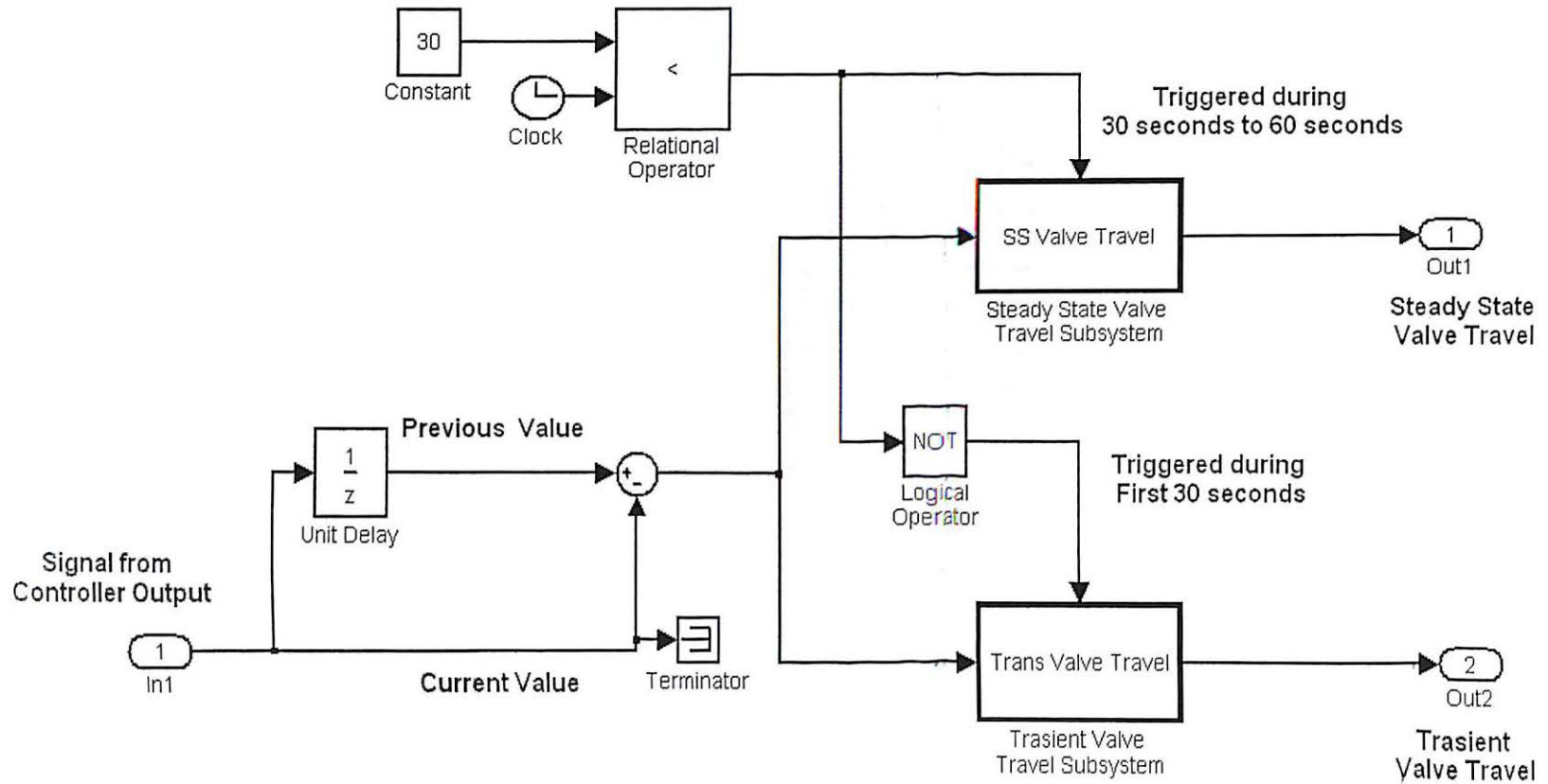


Figure 5.10: Implementing valve travel subsystem in Simulink

Each valve travel subsystem shown in Figure 5.10 is masked and contains the Simulink block shown in Figure 5.9. Figure 5.11 and Figure 5.12 shows the information that is displayed, when the user clicks on the subsystems (Steady State Valve Travel subsystem, Transient Valve Travel Subsystem). These information windows are created to help the user understand the working of the subsystems.

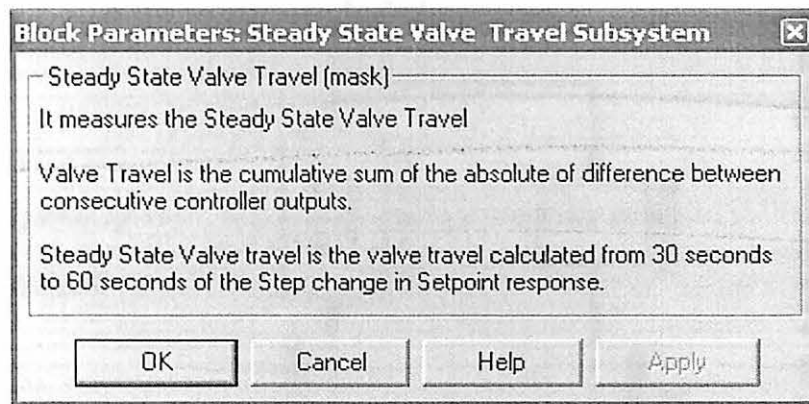


Figure 5.11: Information window for “steady state valve travel subsystem”

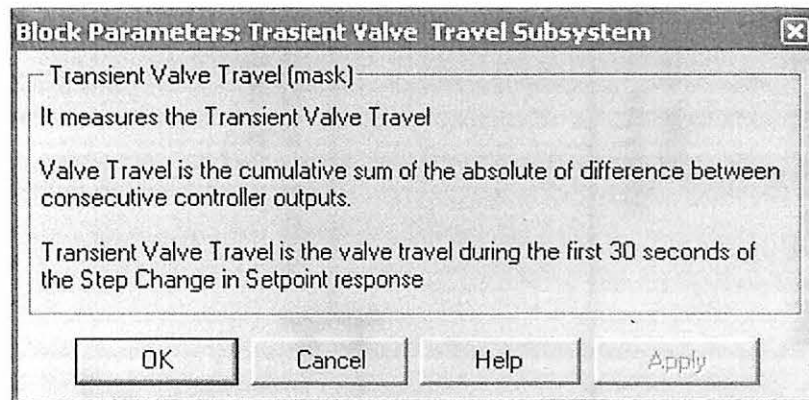
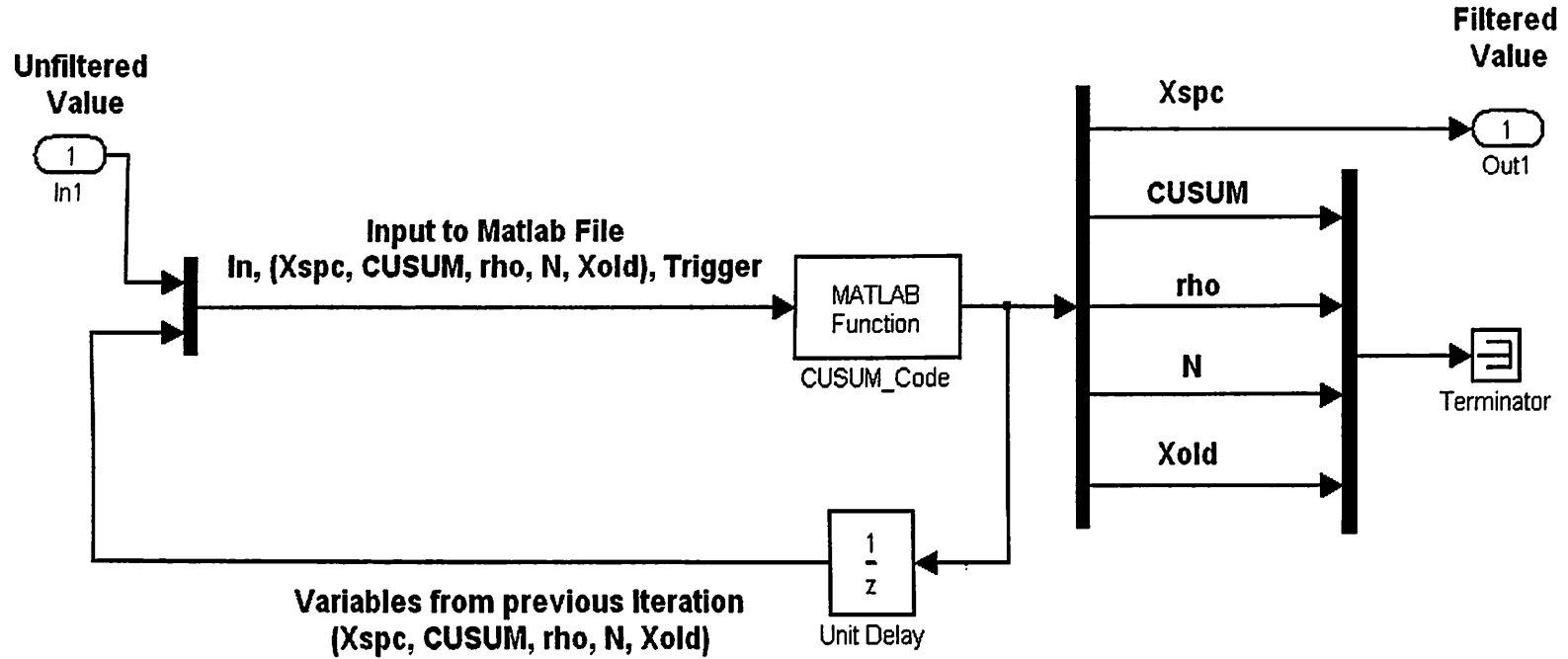


Figure 5.12: Information window for “transient valve travel subsystem”

CUSUM Filter Subsystem



Note: The variable "Trigger" is entered as a parameter in this subsystem

Figure 5.13: CUSUM filter subsystem

The CUSUM filter is implemented using the Simulink Model shown in Figure 5.13. The code for the Matlab function “CUSUM_Code.m” is shown in Figure 5.13b.

```

% CUSUM Filter Code

function out=CUSUM_Code(in,Xspc,CUSUM,rho,N,Xold,trigger)
% Function definition line

% The Function takes the following inputs
% in          Unfiltered Value given as input to the subsystem
% Xspc       previous iteration value
% CUSUM      previous iteration value
% rho        previous iteration value
% N          previous iteration value
% Xold       previous iteration value
% Trigger    parameter defined in the subsystem.

X=in;                % Input Current unfiltered value
rho=0.1*(X-Xold)^2+0.9*rho; % Calculate the Variance
Xold=X;              % Update the Xold value
CUSUM=CUSUM+(X-Xspc); % Update the CUSUM value
N=N+1;              % Increment the N counter

% **** The Filter Logic is defined in this if statement ****
if (abs(CUSUM)>trigger*sqrt(N*rho/2))
    Xspc=Xspc+CUSUM/N; % Update Filter Output
    CUSUM=0;          % Reset value of CUSUM
    N=0;              % Reset value of N
end;

% The Logic is: If ( |CUSUM|>Trigger*sqrt(N*rho/2) )
% Then
% it means that a true change in process has occurred.
% Hence change the filtered value
% Else
% it means that the change in variable is due to process noise
% and not due to any change in process.
% Hence keep the same filtered value.

out(1)=Xspc;
out(2)=CUSUM;
out(3)=rho;
out(4)=N;
out(5)=Xold;

```

Figure 5.13b: Matlab code -“CUSUM_Code.m”

Before running the code shown in Figure 5.13b, the file “init_code.m” (shown in Figure 5.13c) is executed first so that all the variables are initialized.

```
% Project Initialization Routine
clear;
clc;

T=0.2; % Setting the Sampling time as 200 milliseconds
N=0; % Reset value of N (used in CUSUM algorithm)
rho=0; % Reset value of variance rho (used in CUSUM algorithm)
Xspc=0; % Reset value of CUSUM filter O/P(used in CUSUM algorithm)
CUSUM=0; % Reset value of CUSUM (used in CUSUM algorithm)
Xold=0; % Reset variable Xold (used in CUSUM algorithm)

fprintf('All Variables have been Initialized.\n\n');
```

Figure 5.13c: Matlab code – “init_code.m”

The whole CUSUM Simulink subsystem (Figure 5.13) is stored as a single masked block shown in Figure 5.14.

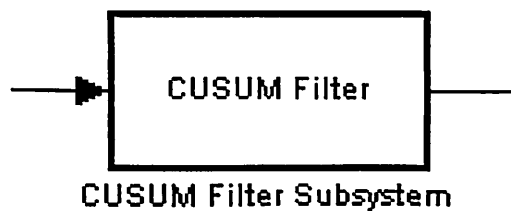


Figure 5.14: CUSUM filter block

Figure 5.15 shows the information that is displayed when the CUSUM filter masked subsystem is clicked (Figure 5.14).

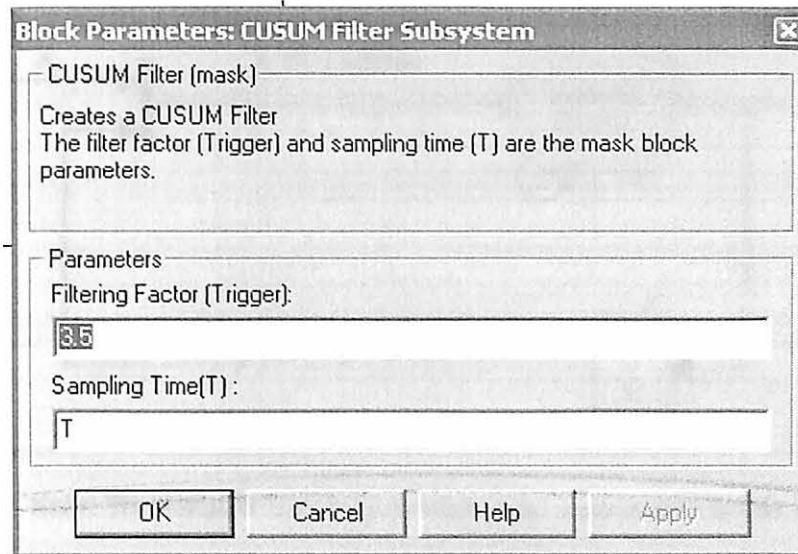


Figure 5.15: Information window for CUSUM filter block

The filtering factor and the sampling time can be changed by entering the appropriate values in the information window shown above in Figure 5.15.

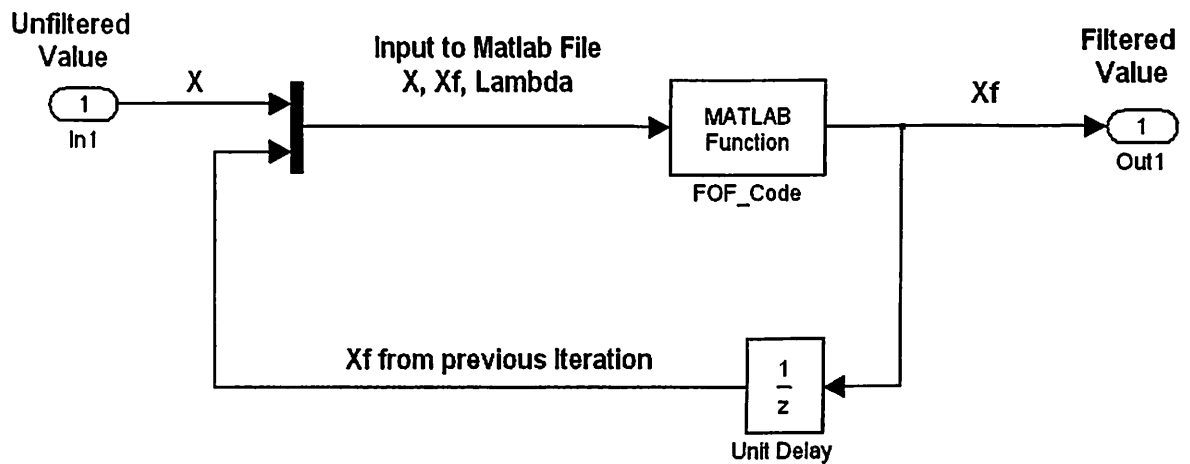
5.5: Simulink Model for the First-Order Filter

From Chapter 2, the equation governing the digital first-order filter is

$$X_{fof_i} = \lambda * (X_{fof_{i-1}}) + (1 - \lambda) * X_i \quad (5.2)$$

The implementation of the above equation in Simulink is very simple and is shown in Figure 5.16.

First-Order Filter subsystem



Note: The variable "Lambda" is entered as a parameter in this subsystem

Figure 5.16: First-order filter subsystem

The Matlab Code (“FOF_Code.m”) used for executing Equation 5.2 is shown in

Figure 5.16a

```

% First Order Filter Code

function out=FOF_Filter_Code(in,Xf,lambda) % Function
definition line

% The Function takes the following inputs
% in      Unfiltered Value
% Xf     Previous Filtered Value
% lambda  Filter Factor

X=in;

Xf=lambda*Xf+(1-lambda)*X; % Main Equation

out=Xf;
    
```

Figure 5.16a: Matlab code – “FOF_Code.m”

The whole first-order filter Simulink subsystem (Figure 5.16) is stored as a single masked block shown in Figure 5.17.

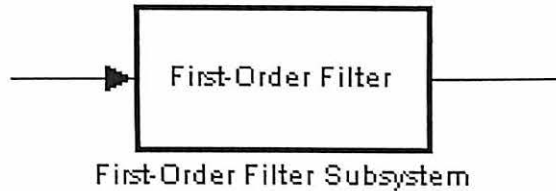


Figure 5.17: First-order filter block

Figure 5.18 shows the information that is displayed when the first-order filter masked subsystem is clicked (Figure 5.17).

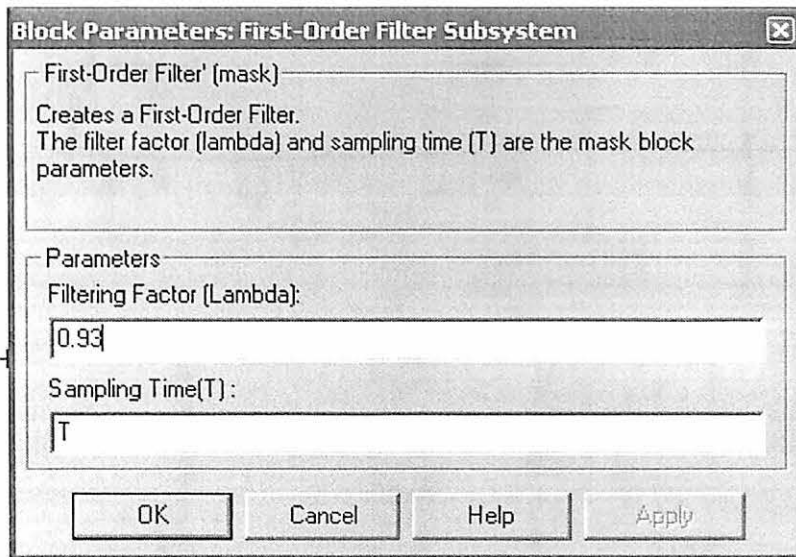


Figure 5.18: Information window for first-order filter block

The Filtering Factor and the Sampling Time can be changed by entering the appropriate values in the information window shown above in Figure 5.18.

5.6: Simulink Model for Closed Loop Configurations

The Simulink model used to simulate each configuration is discussed in the following section.

Configuration A1 - First-Order Filter on the PV: Figure 5.19 shows the Simulink setup for using the first-order filter on the PV.

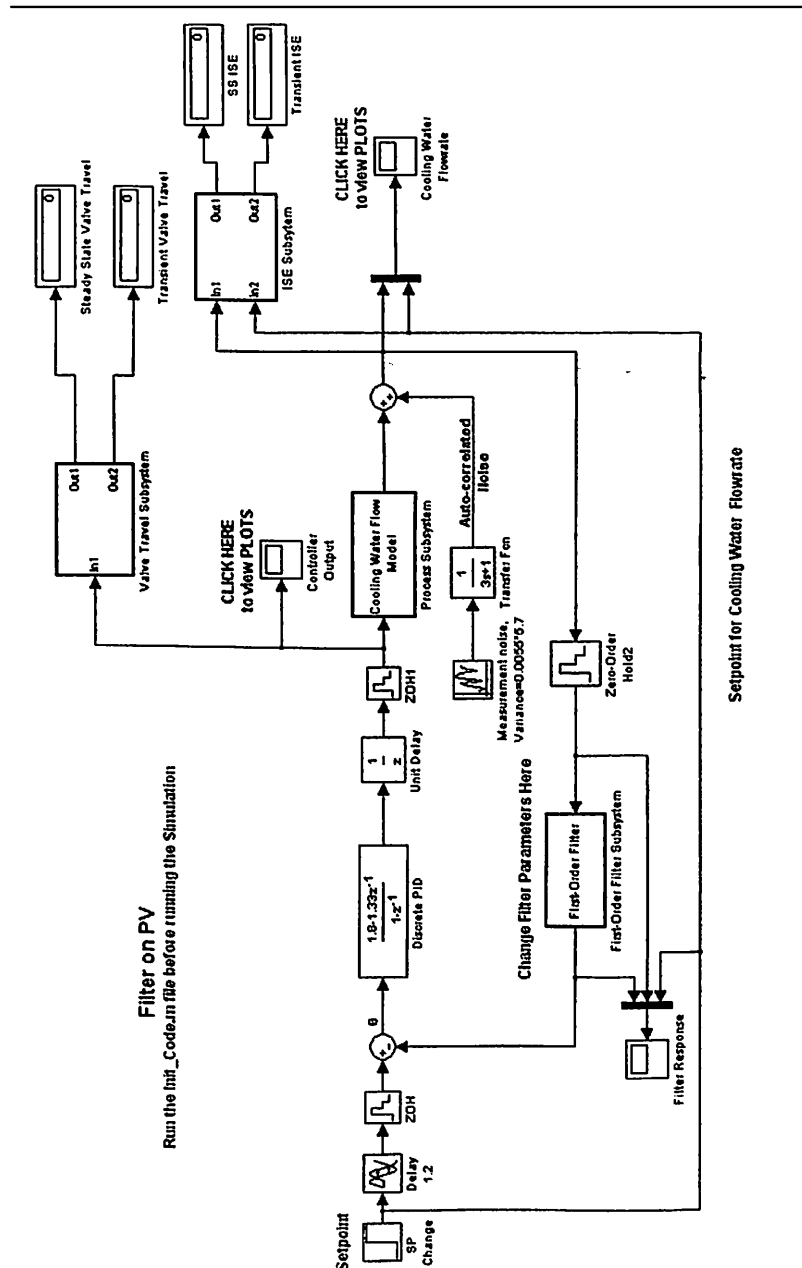


Figure 5.19: Configuration A1 – first-order filter on PV

General Description for All Closed Loop Simulink Blocks: In order to make results from the closed loop simulation duplicate those of the experimental closed loop setup, the following measures are taken.

- Camile uses the controller output of the previous interaction to **change** the manipulated variable of the current iteration. Hence, to make the simulation match the experimental action, a unit delay block is added next to the controller block.
- In actual experimentation, once a setpoint change was done from 0 kg/min to 5 kg/min, the controller output stayed zero for a certain amount of time which was 1.2 seconds more than the process model deadtime (obtained from the process reaction curve method, Section 3.3). In order to simulate ~~that~~ effect, a delay block of 1.2 seconds is added next to the setpoint change block.
- The valve travel subsystem and the ISE subsystem are setup in such a way that if we run the whole simulation for 60 seconds, the steady state and transient ISE/Valve travel are automatically calculated and displayed in the screen.
- Measurement noise is simulated by using the random number Simulink block. A variance of 0.0055 and a mean of 0 are selected in order to make the simulated noise nearly equal to actual process noise. This simulated noise has no autocorrelation. It was observed that the noise present in the experimental data had autocorrelation. Hence, in order to add auto-correlation in the simulation, a first-order block was added after the random number generator. This resulted in auto-correlated noise. By inspection of patterns in the experimental data, the time-constant for the first-order block is set to 3 seconds so that the simulation data

matches those patterns. The multiplication factor for the variance of noise is set to 5.7. Hence after the addition of the first-order block, the variance of the random number block is changed from 0.0055 to 0.03135 (=0.0055*5.7). The derivation for the propagation of variance relation which yielded the 5.7 factor is shown below.

If $y = f(x_1, x_2, x_3 \dots)$, then propagation of variance is given by the following equation

$$\sigma_y^2 = \left(\frac{\partial y}{\partial x_1}\right)^2 \sigma_{x_1}^2 + \left(\frac{\partial y}{\partial x_2}\right)^2 \sigma_{x_2}^2 + \left(\frac{\partial y}{\partial x_3}\right)^2 \sigma_{x_3}^2 \dots \quad (5.3)$$

For first-order block, the equation is

$$X_{f_{of\ i}} = \lambda * (X_{f_{of\ i-1}}) + (1 - \lambda) * X_i. \quad (5.4)$$

The propagation of variance for Equation (5.4) is given by

$$\sigma_{X_{f_{of\ i}}}^2 = (\lambda)^2 \sigma_{X_{f_{of\ i-1}}}^2 + (1 - \lambda)^2 \sigma_{X_i}^2 \quad (5.5)$$

Assuming that the variance of the filter output of consecutive iterations is same

$$\text{Let } \sigma_{X_{f_{of\ i}}}^2 = \sigma_{X_{f_{of\ i-1}}}^2 \quad (5.6)$$

Solving Equation 5.5 and 5.6 to obtain $\sigma_{X_{f_{of\ i}}}$ as a function of λ and σ_{X_i} yields

$$\sigma_{X_i} = \sqrt{\left(\frac{1 + \lambda}{1 - \lambda}\right)} \sigma_{X_{f_{of\ i}}} \quad (5.7)$$

Now $\lambda = \frac{\tau_f}{\tau_f + \Delta t}$, substituting the value of τ_f as 3; $\lambda = \frac{3}{3 + 0.2} = 0.94$

$$\text{Therefore } \sigma_{X_i} = \sqrt{\left(\frac{1.94}{0.06}\right)} \sigma_{X_{f_{of\ i}}} = 5.7 * \sigma_{X_{f_{of\ i}}} \quad (5.8)$$

Configuration A2 - CUSUM Filter on the PV:

Figure 5.20 shows the Simulink setup for using the CUSUM filter on the PV.

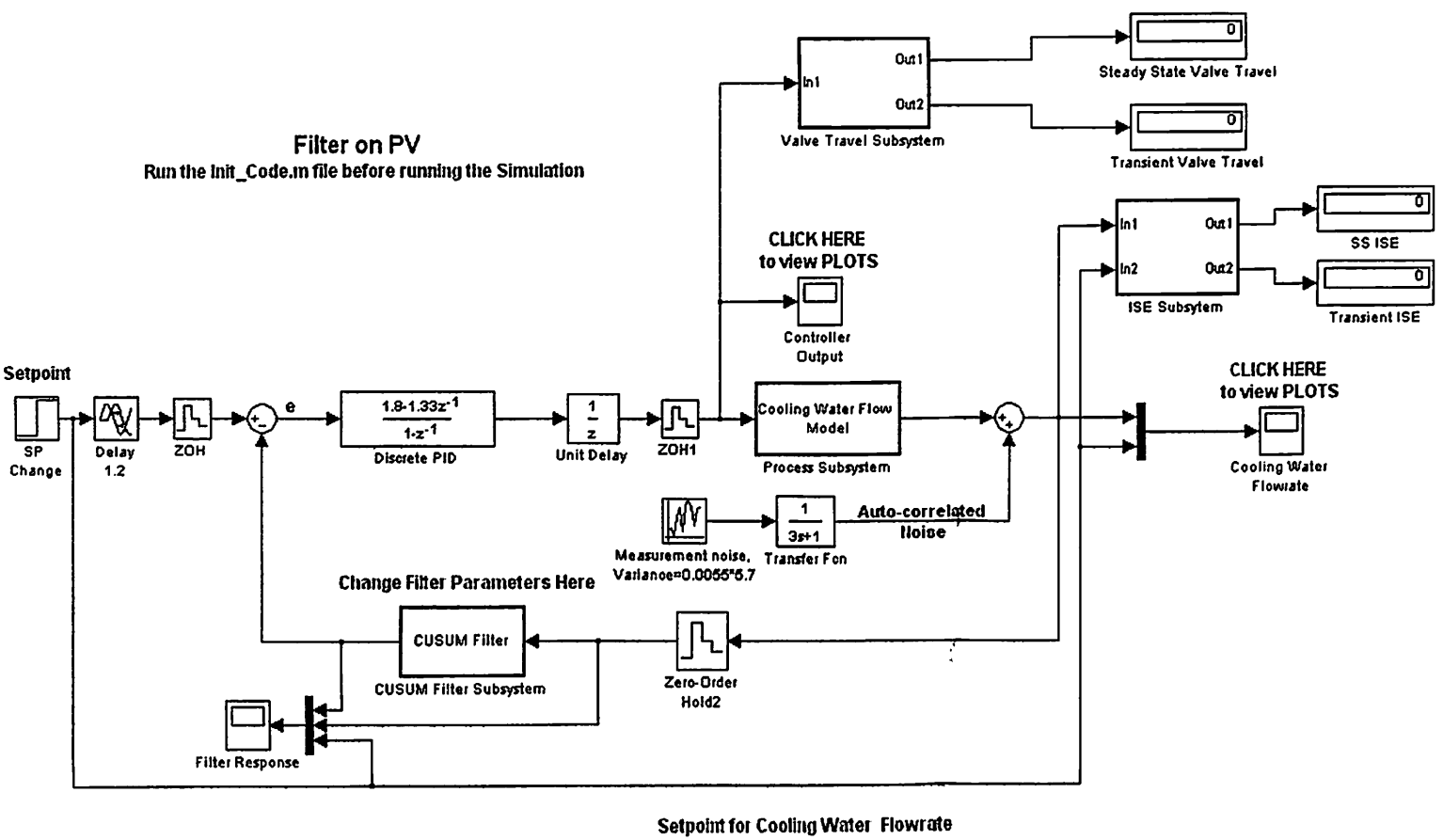


Figure 5.20: Configuration A2 - CUSUM filter on PV

Configuration B1 - First-Order Filter on the MV:

Figure 5.21 shows the Simulink setup for using the first-order filter on the MV.

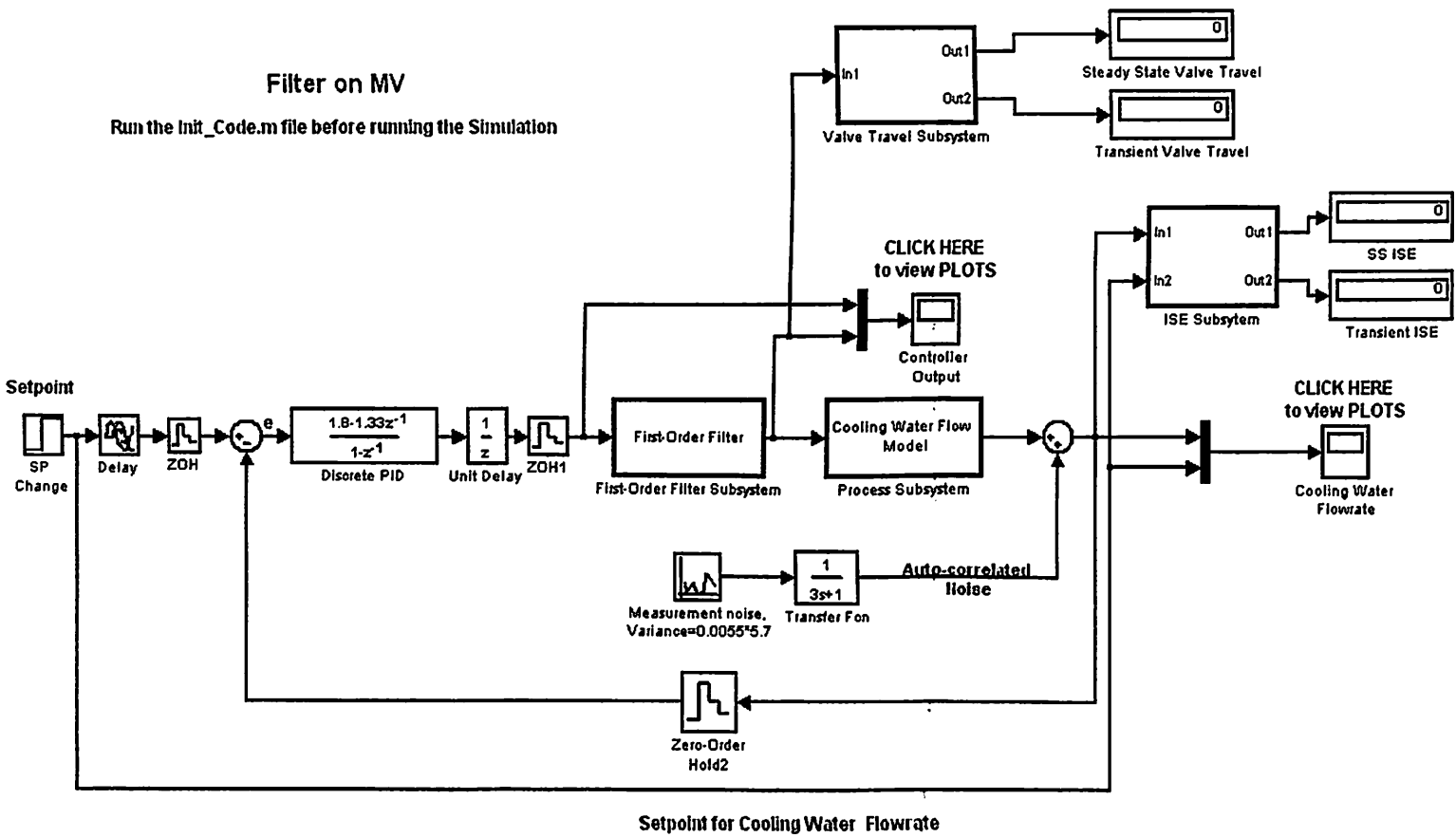


Figure 5.21: Configuration B1 – First-order filter on MV

Configuration B2 - CUSUM Filter on the MV:

Figure 5.22 shows the Simulink setup for using the CUSUM filter on the MV.

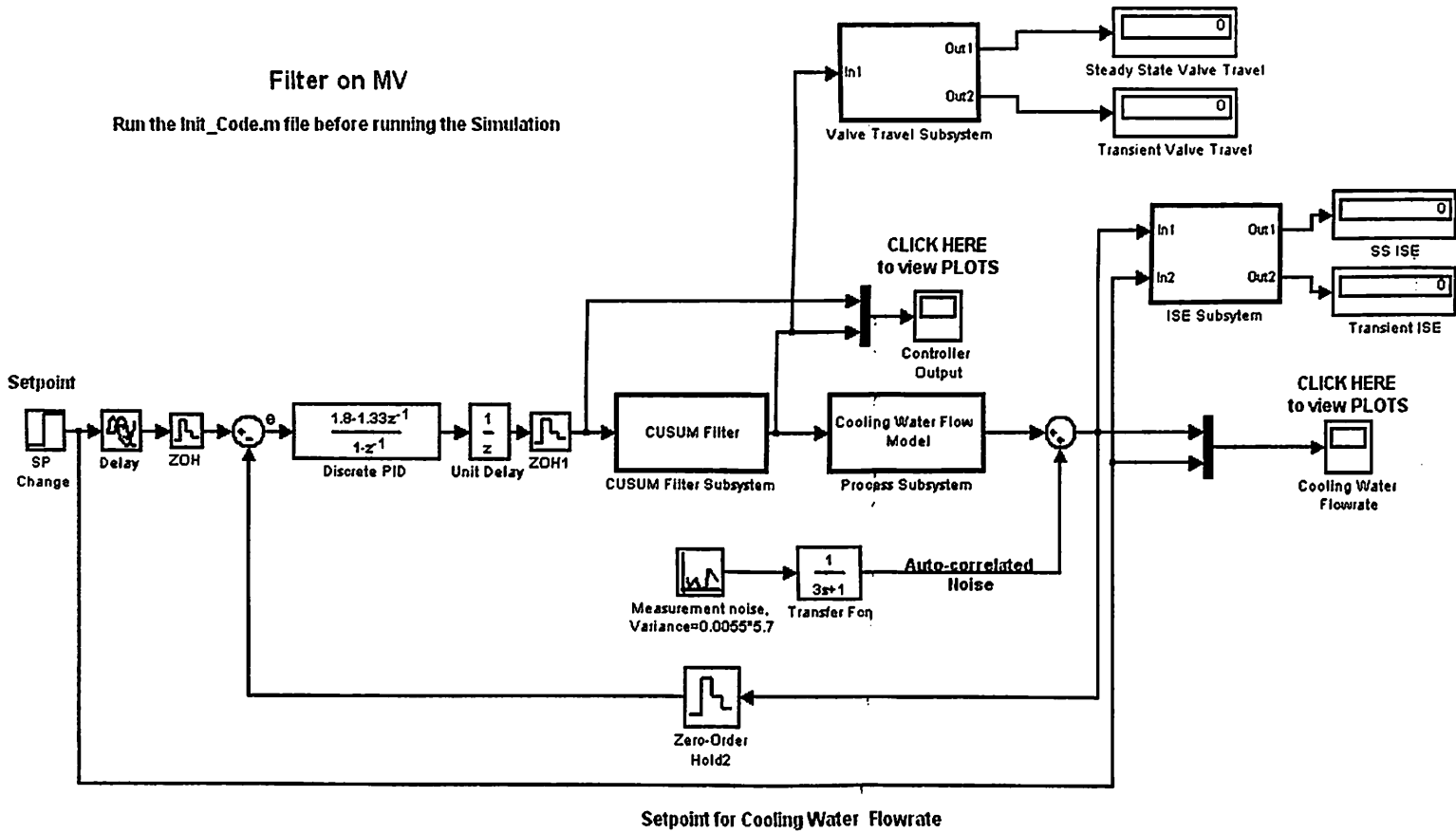


Figure 5.22: Configuration B2 – CUSUM filter on MV

5.8: Simulink Model for Disturbance Rejection Closed Loop Testing

Figure 5.24 shows the Simulink setup for conducting disturbance rejection closed loop testing of the filters for configuration A1.

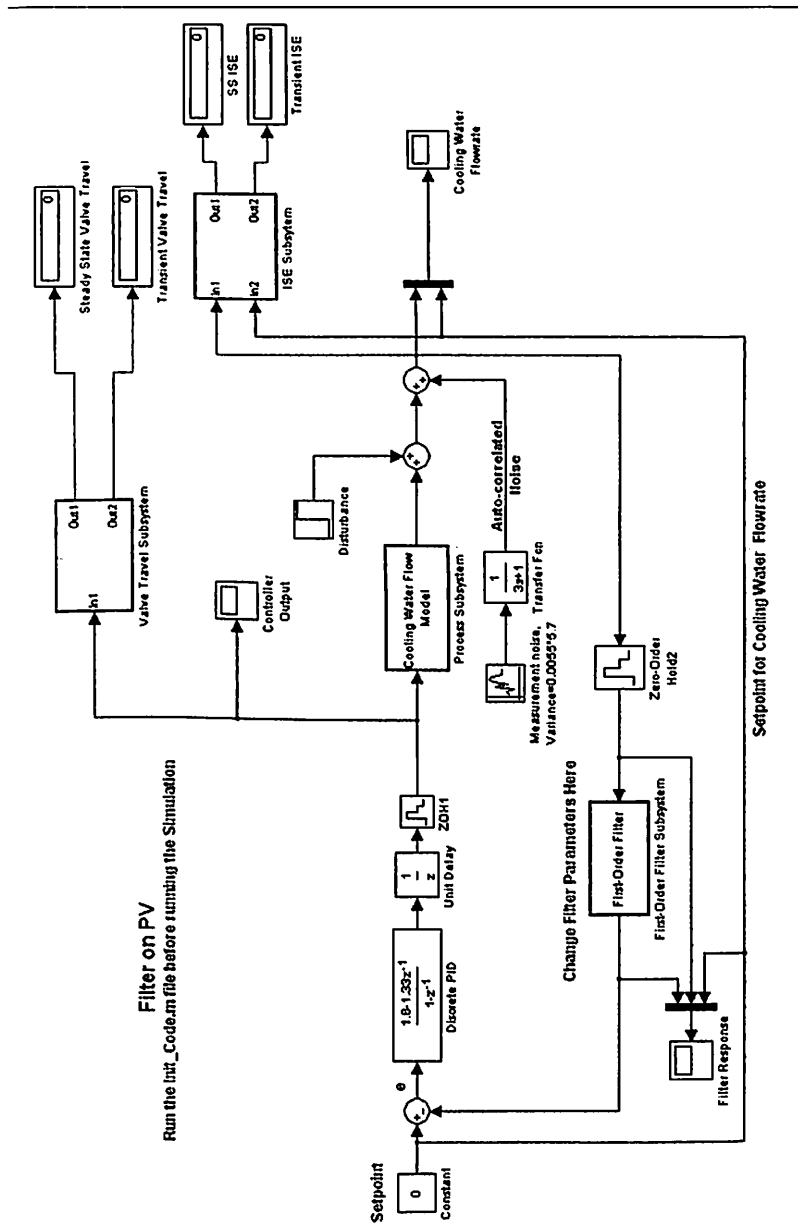


Figure 5.24: Simulink setup - disturbance rejection – Configuration A1

The differences between the closed loop setup for change in setpoint (Figure 5.19) and the closed loop setup for change in disturbance (Figure 5.24) for configuration A1 are described below.

The setpoint block is set to zero and an addition disturbance block is added after the process model block, which creates a step change in disturbance. Similar changes are made to the models shown in Figure 5.20, 5.21 and 5.22 to obtain the Simulink setup for disturbance rejection testing for Configuration A2, B1 and B2 respectively.

CHAPTER 6

SIMULATION RESULTS AND ANALYSIS

6.1: Closed Loop Testing – Change in Setpoint

The simulation models of configuration A1, A2, B1, B2 were executed (Figure 5.19, 5.20, 5.21 and 5.22). The response was captured and the values of the following were obtained from the simulation

- a) Transient ISE
- b) Steady state ISE
- c) Transient valve travel
- d) Steady state valve travel

The servo ISE is calculated as the difference between the transient ISE and steady state ISE. Similarly, the servo valve travel is calculated as the difference between the transient valve travel and the steady state valve travel.

6.2: Servo Mode Analysis

The simulation results of the tests conducted for servo mode for all configurations are shown as a graphical plot in Figure 6.1. The horizontal axis represents the servo valve

travel value. The vertical axis represents the servo ISE value. Similar to the discussion in Section 4.3, lower ISE and lower valve travel values are desired and hence are marked as “Better” and vice versa.

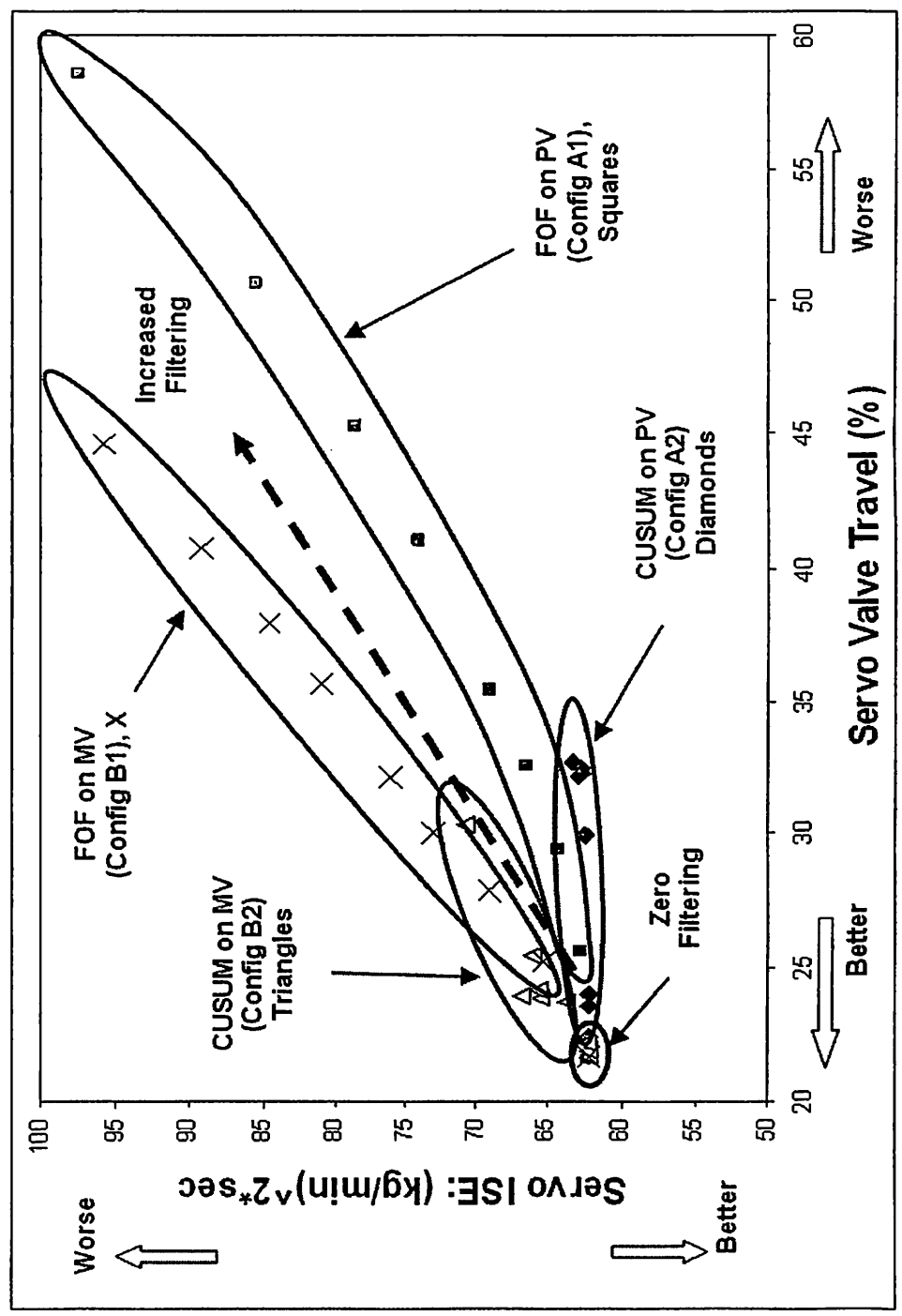


Figure 6.1: Simulation results of Configuration A and B for servo

Description: Various simulations were performed. For instance, a filter tuning parameter and a filter configuration was selected and the corresponding ISE and valve travel was calculated. The result of one simulation is shown as one single point in Figure 6.1. Appendix C contains the tabulation of all the data points.

- The squares represent the data corresponding to first-order filter on the PV (Configuration A1).
- The diamonds represent the data corresponding to CUSUM filter on the PV (Configuration A2).
- The “X” symbols represent the data corresponding the first-order filter on MV (Configuration B1).
- The triangular points represent the data corresponding to CUSUM filter on the MV (Configuration B2).

Comments: The results of the simulations for servo analysis are quite similar to that of the experimental results. When there is a setpoint change, the best ISE valve travel characteristics were obtained when no filtering was done (since no filtering has the smallest valve travel and ISE). Since the controller was tuned for aggressive behavior, adding a filter either in the input side or the output side of the controller resulted in deterioration of the controller performance. As a result the controlled system response became more oscillatory, resulting in a simultaneous increase in both ISE and valve travel.

The relative trends in the four approaches are the same. CUSUM on PV gives best performance, first-order filter on PV – second best, CUSUM on MV – third best and first-order on MV worst. The values of ISE in the simulation are close to the

experimental, providing further support that the model is a valid representation of the experimental system. The value of ISE for the experimental setup for no filtering is about $67 \text{ (Kg/min)}^2 \text{ sec}$ (see Appendix B) while the value of ISE for the simulation setup is about $62 \text{ (Kg/min)}^2 \text{ sec}$ (see Appendix C). In case of first-order filters (on MV and PV), the average ISE values for different tuning parameters are very high while in case of CUSUM filters, the ISE values never reach very high values for any filtering level. This particular trend can be observed in both simulation and experimental results.

6.3: Regulatory Mode Analysis

The simulation results of the tests conducted for regulatory mode for all configurations are shown as a graphical plot in Figure 6.2. The horizontal and the vertical axis and the data point labels represent the same parameters as for servo mode plot (Figure 6.1). Appendix C contains the tabulation of all the data points.

The regulatory mode analysis in simulation represents an ideal steady state condition with auto correlated noise, unlike the experimental setup which was affected by continual line pressure disturbances.

The random number generator seed was identical for all runs. This reduced variability in the simulated data and it made the small trends conspicuous. Referring back to Section 2.12, it can be noted that this general trend of increased filtering causing a decrease in valve travel and an increase in ISE value is expected as this simulation used auto-correlated data.

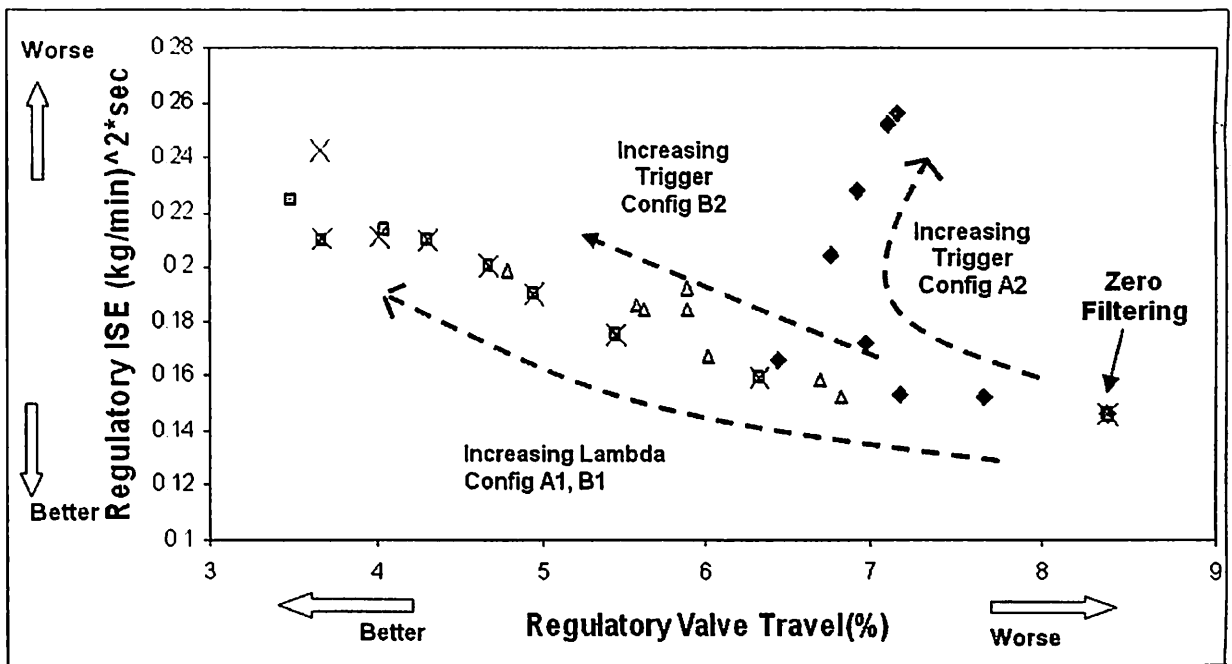


Figure 6.2: Simulation results of Configuration A and B for regulatory

Analysis:

Figure 6.2 shows that zero filtering resulted in higher valve travel values than when filters were added. Adding the first-order filter or the CUSUM filter in any configuration resulted in reduction of valve travel. This is identical to the observed experimental results.

In Configuration A1 i.e. first-order filter on the PV, represented by solid squares, increase in filtering results in a slight increase in ISE value and gradual reduction in valve travel. The values of ISE and valve travel are similar to that of the experimental results. This type of trend was expected. In the simulation, the continual line disturbances were not incorporated and the noisy process variable signals were averaging at the setpoint. Hence, the first-order filter was holding the process variable near the setpoint which resulted in less valve travel.

In Configuration A2, i.e. CUSUM filter on the PV, represented by solid diamonds, as trigger value is increased from 0 to 2.5; there is a reduction in valve travel (8.3% to 6.4%) and a gradual increase in ISE value (0.15 to 0.16). If the trigger is further increased, the ISE value increases and the valve travel value also starts to increase. This result was expected. When the trigger is increased from 0 to 2.5, the controller response to noise reduces and hence the control valve tampering is reduced causing reduction in valve travel. Further increase in trigger value (>2.5) makes the CUSUM filter hold the process variable at a particular value for a longer time (more sigma level of significance required). This result in an offset and the integral action starts to windup and hence makes the controller more aggressive causing a simultaneous increase in both ISE and valve travel.

In Configuration B1, i.e. first-order filter on the MV, represented by “X” points, increase in filtering results in a slight increase in ISE value and gradual reduction in valve travel. It shows similar trends as configuration A1. Since there is no setpoint change and no disturbance, it is expected that the first-order filter on the MV and on the PV will show the same trend in ISE/Valve travel.

In Configuration B2, i.e. CUSUM filter on the MV, represented by solid triangles, as we increase the trigger values from 0 to 3.5, there is a reduction in valve travel and an increase in ISE value. Further increase in trigger results in simultaneous increase in both valve travel and ISE. Since the CUSUM filter holds the MV for a certain amount of time (sigma level of significance), for lower trigger values, the valve tampering is reduced causing lower valve travel. Higher level of filtering holds the MV for a larger amount of time causing a freeze in the MV value. This degrades the controller performance resulting in higher ISE and valve travel.

It can be pointed out that, similar to experimental results, there is no significant difference in ISE and valve travel characteristics for three of the configurations (A1, B1 and B2).

6.4: Closed Loop Testing – Change in Disturbance

Simulink setups for conducting disturbance rejection closed loop testing of filters were executed (Figure 5.24). The response was captured and the values of the following were obtained from the simulation

- e) Transient ISE
- f) Steady state ISE
- g) Transient valve travel
- h) Steady state valve travel

The “Disturbance ISE” is calculated as the difference between the transient ISE and steady state ISE. This “Disturbance ISE” will capture the properties of the transient response due to disturbance change. Similarly, the “Disturbance valve travel” is calculated as the difference between the transient valve travel and the steady state valve travel.

6.5: Disturbance Rejection Analysis

The simulation results of the tests conducted for disturbance rejection for all configurations are shown as a graphical plot in Figure 6.3. The horizontal and the vertical axis and the symbols used in this figure are similar to that of Figure 6.1.

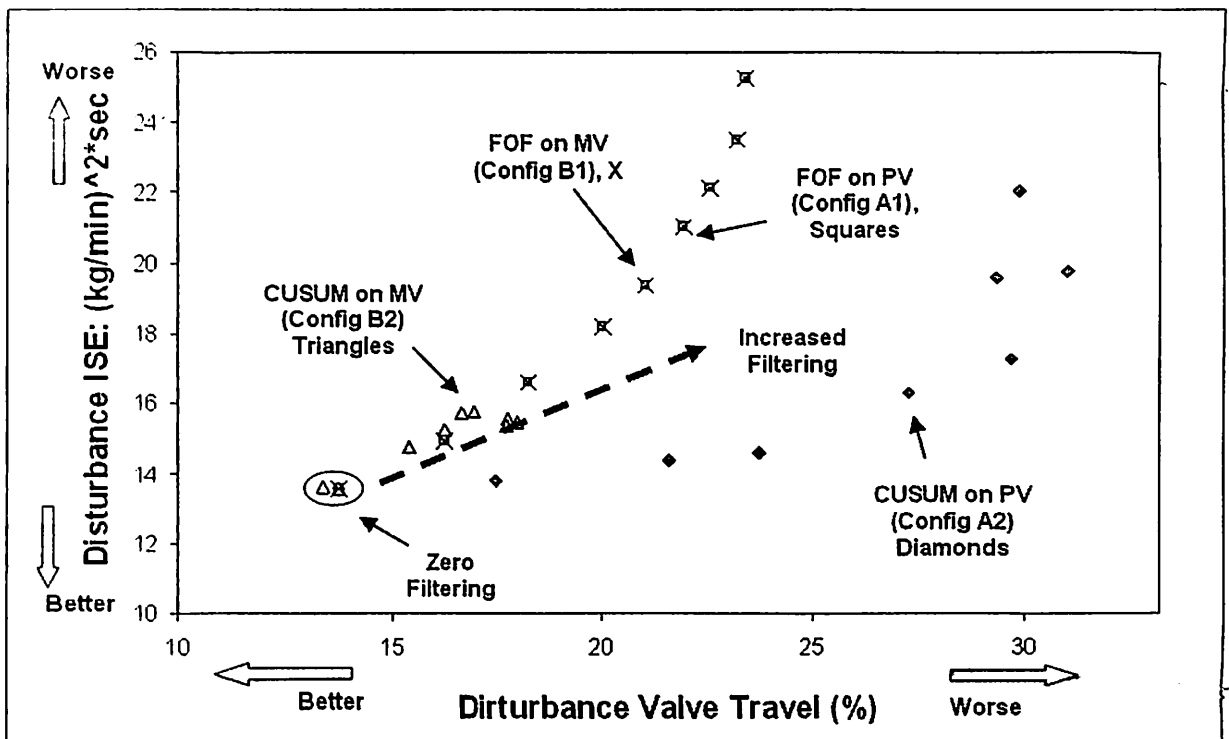


Figure 6.3: Simulation results for Config. A and B for disturbance change

Comments: It can be observed from Figure 6.3 that the ISE/Valve travel trends of all configurations for disturbance change are similar to that for the setpoint change. It can be concluded from Figure 6.3 that when there is a disturbance change, the best ISE/Valve travel characteristics are obtained when no filtering is done (since no filtering has the smallest ISE and valve travel).

6.5: Comparison of Experimental and Simulation Results

Servo Mode: Both the experimental and simulation results strengthen the fact that when the controller is tuned (with no filters on PV and MV) to be aggressive and having minimum ISE values, in such a case, during a setpoint change it is best not to use any filter. The use of CUSUM or first-order in any configuration always results in addition of

delay or lag in the closed loop and causes deterioration of controller performance. The idea being, adding a filter makes the controller more aggressive and hence resulting in oscillatory response. If a filter is desired, both the experimental and simulation indicate that the CUSUM filter on the PV gives best results.

Regulatory Mode: The relative trends of the different configurations obtained from the experimental analysis are slightly different from that of the simulation. The suspected reason being, the experimental setup was influenced by continual line pressure disturbances affecting the flow rate. Hence, an ideal noisy steady state condition was not obtained during experimentation. On the contrast, during simulation, an ideal noisy steady state condition was obtained, which did not have any disturbances.

It can be concluded from the experimental results that when there is no setpoint change and the controller is tuned (with no filter on PV and MV) to be aggressive, it is best to use the CUSUM filter on the MV, since it results in the lowest valve travel and good ISE values as well.

In the simulation results, all filters, except CUSUM filter on the PV, show similar performances. Hence, there is no preference of one configuration over another. But, it can be concluded that in simulation, the CUSUM filter on the PV is not a good option. The reason why the CUSUM filter on the MV did not have lowest valve travel in simulation could not be found.

6.7: Open Loop Testing

In order to conduct the open loop testing of filters, the Simulink setup shown in Figure 5.23 is used. As discussed in Section 2.6, the lag is quantified by measuring the cumulative sum of the differences between the noisy process variable (PV) and the filter output during the transient phase. Comparative terms like “more” or “less” lag are arrived by quantifying the lags using the mentioned method.

Effect of Sampling Rate on First-Order Filter and CUSUM Filter:

A particular value of λ and trigger has to be selected. The filtering factors (λ and trigger) are selected in such a manner that there is very less lag during a change in steady state, for a sampling time of 0.2 seconds.

The Trigger of CUSUM filter is set to 3.0

The λ of first-order filter is set to 0.9

The sampling time is selected as 0.2, 0.4 and 0.6 seconds and Figure 6.4, 6.5 and 6.6 show the corresponding filter response graphs. In all the three figures, the unfiltered data, the filter output of CUSUM filter and first-order filter are plotted with respect to time.

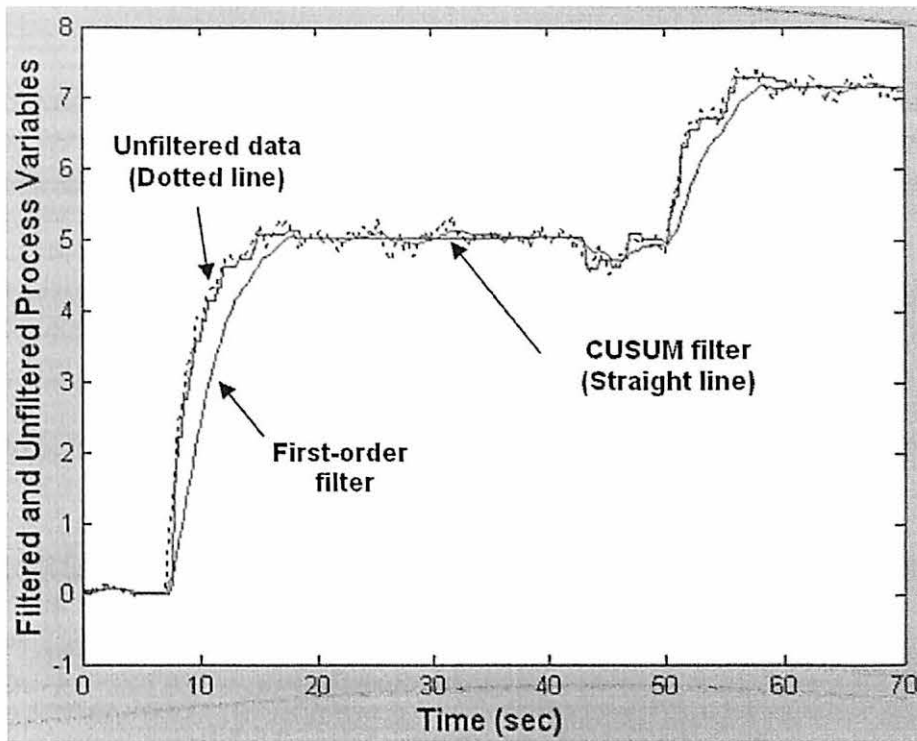


Figure 6.4: Open loop testing results: sampling time – 0.2 sec

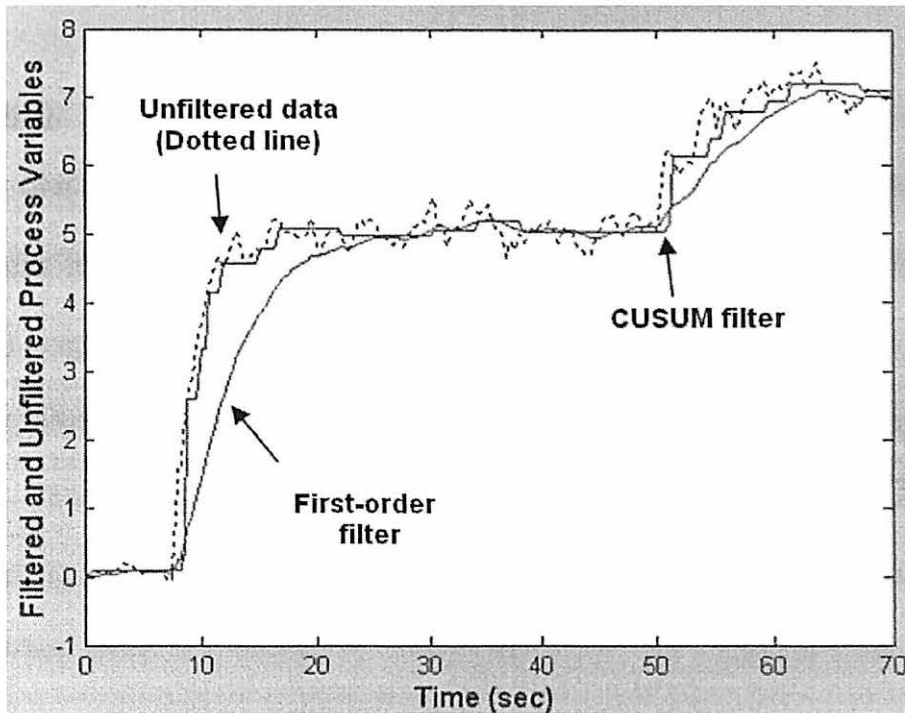


Figure 6.5: Open loop testing results: sampling time – 0.4 sec

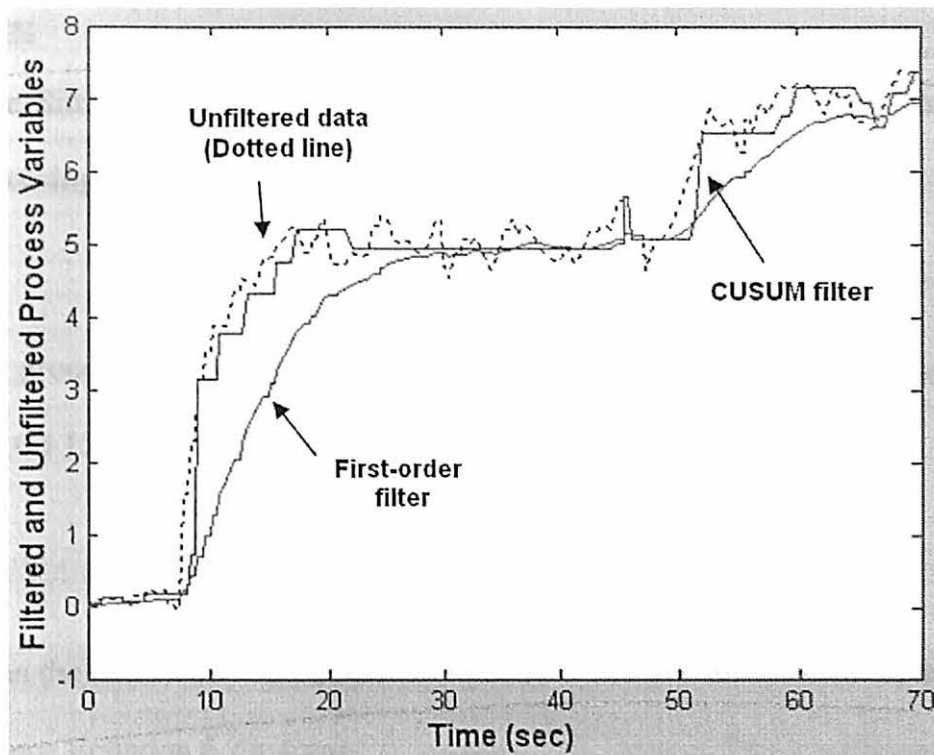


Figure 6.6: Open loop testing results: sampling time – 0.6 sec

Observations:

The first-order filter performance deteriorates as we increase the sampling time, since it results in more lag. The value of lag (calculated using the method described in Section 2.6) keeps on increasing for increase in sampling time. When there is a change in steady state value, the first-order filter takes a larger time to reach the new steady state.

The CUSUM filter performance is better than the first-order filter. Even for longer sampling times (Figure 6.6), the CUSUM filter does not introduce much lag in the response. When there is a change in steady state value, the CUSUM filter reaches the new steady state value almost instantaneously.

Comments:

The Simulink model of the first-order filter used in this analysis is characterized by the following equation

$$X_{fof_i} = \lambda * (X_{fof_{i-1}}) + (1 - \lambda) * X_i \quad (6.1)$$

In process industries, the first-order filters are mostly characterized by the Equations 6.1 [5,11]. But, if Equation 6.2 is used for the first-order filter

$$X_f(i) = \frac{\Delta t}{\tau_f + \Delta t} * X(i) + \frac{\tau_f}{\tau_f + \Delta t} X_f(i-1) \quad (6.2)$$

then the first-order filter output is expected to become independent of sampling time Δt , since Equation 6.2 automatically changes the value of filter coefficients.

CHAPTER 7

CONCLUSION AND RECOMMENDATIONS

7.1: Conclusion

1. When the controller is tuned (with no filters on PV and MV) to be aggressive and having minimum ISE values, in such a case, during a setpoint change and major disturbance change, it is best not to use any filter. The use of CUSUM or first-order in any configuration always results in addition of delay or lag in the closed loop and causes deterioration of controller performance. This conclusion was obtained from the experimental analysis as well as the simulation analysis. If the controller were tuned to be conservative, it is expected that adding a filter can actually improve the controller performance by reducing the ISE.
2. When there is no setpoint change and the controller is tuned (with no filter on PV and MV) to be aggressive, it is best to use the CUSUM filter on the MV, since it results in the lowest valve travel and good ISE values. This conclusion was obtained from the experimental analysis. The conclusion arrived from the simulation analysis was that, all filters, except CUSUM filter on the PV, show similar performances. Hence, there is no preference of one configuration over another. But, it can be concluded that in simulation, the CUSUM filter on the PV is not a good option

3. The CUSUM filter is relatively independent to sampling time as compared to the first-order filter (when λ or Trigger is not changed).
4. The study did not explore the effect of re-tuning the controller to accommodate changes in the filter.
5. In general, when the controller is tuned (with no filters on PV and MV) to be aggressive, and considering the fact that setpoint changes are rare in process control, using the CUSUM filter on the MV can be a good option since it causes very good reduction in valve travel and good ISE performance too.

7.2: Recommendation

1. When the controller is tuned (with no filters on PV and MV) to be aggressive and if the process does have frequent setpoint changes, then, it is good not to use any filter. Further research could investigate a mechanism to remove filtering during a transient.
2. Explore effects of disturbances.
3. Investigate use of ERF (External Reset Feedback) when filter on PV changes controller output.
4. Perform mathematical analysis to support experimental findings.
5. Explore adaptive controller tuning. Design the controller such that, it adjusts the tuning parameters to lag/delay of loop.
6. Compare with Kalman and Butterworth filters.

REFERENCES

1. “*Understanding Noise in Measurement*”(n.d.), Retrieved June 9th, 2004, from <http://www.capgo.com/Resources/Measurement/Noise/Noise.html>
2. Dorf, R. C., & Bishop, R. H. (1999), “*Modern Control Systems*”, Addison-Wesley, California.
3. Gopal, M. (1997), “*Digital Control and State Variable Methods*”, Tata McGraw-Hill Publishing Company Limited, New Delhi.
4. Mettu, K. R. (2004), “*Instruction Manual, Heat Exchanger Unit, Unit Operations Lab*”, Oklahoma State University, Stillwater, Oklahoma
5. Ogunnaike, B. A., & Ray, W. H. (1994), “*Process Dynamics, Modeling, and Control*”, Oxford University Press, Inc., New York
6. Pearson, R.(n.d.), “*Scrub data with scale-invariant nonlinear digital filters*”, Swiss Federal Institute of Technology, Retrieved June 12th, 2004, from <http://www.reed-electronics.com/ednmag/contents/images/191159.pdf>
7. Rhinehart, R. R. (1992), “A CUSUM-type of on-line Filter,” *Process Control and Quality*, Vol 2
8. Rhinehart, R. R. (2002), “*A Statistically-Based Filter*”, *ISA Transactions*, Vol. 41 No. 2.
9. Seborg, D. E., Edgar, T. F. & Mellichamp, D. A. (1989), “*Process Dynamics and Control*”, John Wiley & Sons, New York.
10. Smith, J. O. (2004), “*Introduction to Digital Filters*”, Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, Retrieved June 7th, 2004, from <http://www-ccrma.stanford.edu/~jos/filters/>
11. Stephanopoulos, G. (2001), “*Chemical Process Control, An Introduction to Theory and Practice*”, Prentice-Hall of India Private Limited, New Delhi.
12. Tham., M. T. (1998), “*DEALING WITH MEASUREMENT NOISE*”, Retrieved June 13th, 2004, from <http://lorien.ncl.ac.uk/ming/filter/filter.htm>
13. Whiteley, R. (2004), CHE 5853 Course Notes, Lecture 16 - Spring 2004, Oklahoma State University, Stillwater, Oklahoma.

APPENDIX A

CALIBRATION OF FLOW TRANSMITTER

The cooling water flowrate is measured using a differential pressure (DP) transmitter. The DP transmitter senses the pressure difference across the orifice and generates a proportional 4-20 mA signal. This signal is given to the Camile data acquisition system and is further converted into proportional flowrate in Camile.

A correlation between the current signal (mA) and the calculated mass flowrate (kg/min) is required [4].

The Bernoulli's Equation is given by

$$p + \frac{1}{2} \rho v^2 + \rho gh = \text{constant} \quad (\text{A.1})$$

Where p is the pressure, ρ is the fluid density, h is the elevation, v is the fluid velocity and g is gravitational constant. Considering horizontal flow, Equation A.1 can be written as

$$p_1 + \frac{1}{2} \rho v_1^2 = p_2 + \frac{1}{2} \rho v_2^2 = \text{constant} \quad (\text{A.2})$$

The continuity equation is given by

$$q = v_1 A_1 = v_2 A_2 \quad (\text{A.3})$$

Solving Equation A.2 and A.3 and introducing a multiplier C_d for more accurate representation, the equation of flow becomes

$$q = C_d A_2 \left(\frac{2\Delta p}{\rho \left(1 - \left(\frac{A_2}{A_1} \right)^2 \right)} \right)^{\frac{1}{2}} \quad (\text{A.4})$$

where C_d is the discharge coefficient.

Converting volumetric flow into mass flow

$$\dot{M} = \rho C_d A_2 \left(\frac{2\Delta p}{\rho \left(1 - \left(\frac{A_2}{A_1} \right)^2 \right)} \right)^{\frac{1}{2}} \quad (\text{A.5})$$

In the experimental setup, the pressure drop cannot be measured directly. Instead, the current signal, proportional to the pressure drop is obtained.

$$\Delta p = \alpha (i - i_o) \quad (\text{A.6})$$

Substituting Equation A.6 in A.5

$$\dot{M} = \rho C_d A_2 \left(\frac{2\alpha (i - i_o)}{\rho \left(1 - \left(\frac{A_2}{A_1} \right)^2 \right)} \right)^{\frac{1}{2}} \quad (\text{A.7})$$

Combining all the constants in Equation A.7 into a single factor “a”

$$\dot{M} = a (i - i_o)^{0.5} \quad (\text{A.8})$$

Certain assumptions in Bernoulli's equation are not valid. They are listed as follows.

- a) The cooling water does not have ideal in-viscid, potential flow.
- b) The flow profile is not flat, but "bullet" shaped.
- c) Flow lines are not ideally parallel due to pipeline elbows.
- d) The orifice is not perfectly thin, but has nozzle like attributes.

Due to the above mentioned reasons, replacing the square root relationship with a power law gives a more flexible representation of the system which is grounded in ideal expectations.

$$\dot{M} = a(i - i_o)^b \quad (\text{A.9})$$

Where \dot{M} is the mass flowrate in kg/min, i and i_o are current measurements in mA, "a" and "b" are constants. The constant "b" is expected to be between 0.4 and 0.6.

Taking logarithm on both sides of Equation A.9 gives Equation A.10 which represents the equation of a straight line.

$$\log_{10} \dot{M} = \log_{10} a + b \cdot \log_{10}(i - i_o) \quad (\text{A.10})$$

The values of \dot{M} and i can be measured at various control valve positions and a graph can be plotted with $\log_{10} \dot{M}$ on the vertical axis and $\log_{10}(i - i_o)$ on the horizontal axis. By performing a regression analysis of this graph, the values of intercept ($\log_{10} a$) and slope (b) can be found out. In this way, the values of constants "a" and "b" can be calculated.

Table A.1

Values of $\log_{10} \dot{M}$ and $\log_{10}(I - I_o)$ at various control valve positions

CO	M(w+v)	M(w)	M(w)	Time	Mdot	Mdot	ln Mdot	I	(I-Io)	ln(I-Io)
%	lb	lb	kg	Sec	kg/sec	kg/min	ln(kg/min)	mA	mA	ln(mA)
20%	6	5.25	2.380952	120	0.019841	1.190476	0.174353	4.19	0.16	-1.83258
40%	8.5	7.75	3.514739	30	0.117158	7.029478	1.950113	9.5	5.47	1.699279
60%	9.5	8.75	3.968254	25	0.15873	9.52381	2.253795	15.3	11.27	2.422144
70%	10.25	9.5	4.30839	20	0.21542	12.92517	2.559177	17.2	13.17	2.577942
80%	9.12	8.37	3.795918	20	0.189796	11.38776	2.432539	19.1	15.07	2.712706
90%	9.13	8.38	3.800454	20	0.190023	11.40136	2.433733	19.4	15.37	2.732418
100%	10	9.25	4.195011	20	0.209751	12.58503	2.532508	19.7	15.67	2.751748

Weight of Vessel M (v): 0.75 lb

Current at Zero Flow (Io): 4.03 mA

Figure A.1 represents the plot required for finding the values of constants “a” and “b”. $\log_{10} \dot{M}$ is plotted on the vertical axis and $\log_{10}(i - i_o)$ is plotted on the horizontal axis. The regression analysis is performed in MS Excel. The regression equation is shown in Figure A.1 and in Equation A.11.

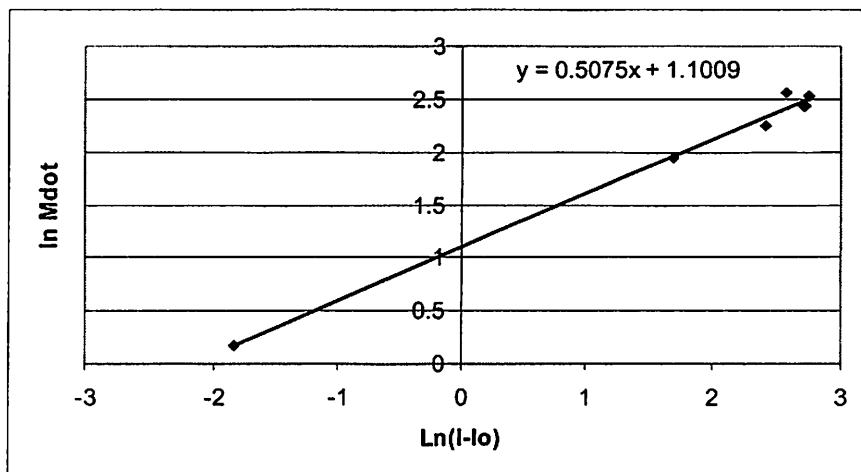


Figure A.1: Regression Analysis

Regression analysis yields us the following equation

$$Y=1.1009+0.5075.x \quad (A.11)$$

Comparing Equation A.11 with Equation A.10

$$b=0.5075 \quad (A.12)$$

$$a=\exp(1.1009)=3.006871 \quad (A.13)$$

These values of “a” and “b” are entered into the Camile software and in this way the flow transmitter is calibrated.

APPENDIX B

TABULATION OF EXPERIMENTAL RESULTS

Symbols used in Tables:

Trans	: Transient
SS	: Steady state
SSE	: Sum of square of error
ISE	: Integral of square of error – SSE * sampling time (0.2 sec)
VT	: Valve travel
Actual transient ISE	: (Trans SSE - SS SSE) * Sampling time
Actual valve travel	: (Trans valve travel – SS valve travel)

Table B.1

Experimental - Configuration A1 - First-order filter on PV

λ		Trans SSE	SS SSE	Actual	Actual	SS ISE	Trans	SS	Actual
				Trans SSE	Trans ISE		Valve Travel	Valve Travel	Tran VT
0		342.4015992	0.81254273	341.5890565	68.31781129	0.162509	56.4986	15.9686	40.53
0		340.4950147	0.53766714	339.9573476	67.99146951	0.107533	57.9334	14.2861	43.6473
0		337.8400957	0.66227804	337.1778177	67.43556353	0.132456	55.3919	16.5161	38.8758
0.1		350.9315578	0.95236296	349.9791948	69.99583897	0.190473	62.644	14.0986	48.5454
0.1		351.4597659	0.63733843	350.8224275	70.16448549	0.127468	61.3095	14.1227	47.1868
0.1		344.3506958	0.65681495	343.6938809	68.73877617	0.131363	59.9057	14.3873	45.5184
0.2		355.5517842	0.71034639	354.8414378	70.96828756	0.142069	62.283	11.0178	51.2652
0.2		357.7323529	0.51088313	357.2214698	71.44429395	0.102177	62.0165	11.2998	50.7167
0.2		353.6856495	0.65775774	353.0278918	70.60557835	0.131552	61.21	10.8316	50.3784
0.3		353.8951833	0.54620522	353.3489781	70.66979562	0.109241	61.648	9.5052	52.1428
0.3		355.7558228	0.66668992	355.0891329	71.01782658	0.133338	60.9733	9.9366	51.0367
0.3		354.7881876	0.72787913	354.0603085	70.81206169	0.145578	62.173	10.2262	51.9468
0.4		358.2621339	0.51957473	357.7425592	71.54851183	0.103915	62.2564	7.4544	54.802
0.4		357.9188401	0.60704337	357.3117967	71.46235935	0.121409	61.6639	7.5005	54.1634
0.4		358.102104	0.76703203	357.335072	71.46701439	0.153406	61.8991	7.4429	54.4562
0.5		368.0385179	1.52142415	366.5170938	73.30341875	0.304285	63.0249	8.3385	54.6864
0.5		366.6723644	0.75322996	365.9191344	73.18382689	0.150646	62.5575	6.6616	55.8959
0.5		366.2115511	0.6665315	365.5450196	73.10900392	0.133306	62.1685	6.5038	55.6647
0.55		367.9464881	0.5196755	367.4268126	73.48536252	0.103935	63.4574	5.5927	57.8647
0.55		373.3170447	0.65281839	372.6642263	74.53284526	0.130564	64.8373	5.7993	59.038
0.55		374.060254	0.68886541	373.3713886	74.67427772	0.137773	65.2144	6.0049	59.2095
0.6		381.3986225	0.43779543	380.9608271	76.19216541	0.087559	67.2229	5.7012	61.5217
0.6		383.7934461	0.49693397	383.2965121	76.65930243	0.099387	68.9683	5.1774	63.7909
0.6		376.6754388	0.77257422	375.9028646	75.18057292	0.154515	64.9971	5.7012	59.2959
0.65		388.5055711	0.64855889	387.8570122	77.57140244	0.129712	69.1685	4.5123	64.6562
0.65		386.1043733	0.62175394	385.4826194	77.09652387	0.124351	68.3245	4.6502	63.6743
0.65		384.9131969	0.55551456	384.3576823	76.87153647	0.111103	68.9879	4.2214	64.7665
0.7		406.4163677	0.52950167	405.886866	81.17737321	0.1059	74.2736	3.7592	70.5144
0.7		402.1737216	0.83139279	401.3423288	80.26846576	0.166279	72.7427	4.5842	68.1585
0.7		410.0060837	0.71542136	409.2906623	81.85813247	0.143084	74.1746	4.3373	69.8373
0.725		417.7209985	0.48977787	417.2312206	83.44624413	0.097956	76.2077	3.2126	72.9951
0.725		415.0797787	0.94125635	414.1385224	82.82770447	0.188251	75.2298	4.6815	70.5483
0.725		414.6829281	0.71435083	413.9685773	82.79371545	0.14287	75.4201	3.8658	71.5543
0.75		425.2401392	0.40272001	424.8374192	84.96748384	0.080544	78.1002	2.8045	75.2957
0.75		435.313006	0.90849914	434.4045069	86.88090137	0.1817	80.4847	4.3925	76.0922
0.75		432.2503581	0.62530103	431.6250571	86.32501141	0.12506	80.5894	3.511	77.0784
0.775		444.9059003	0.78658199	444.1193183	88.82386366	0.157316	80.7476	3.7403	77.0073
0.775		441.0154659	0.63478865	440.3806773	88.07613545	0.126958	81.0256	3.3953	77.6303
0.775		431.6179156	0.7936134	430.8243022	86.16486044	0.158723	78.6667	3.7643	74.9024

Table B.2

Experimental - Configuration A2 - CUSUM filter on PV

Trigger		Trans SSE	SS SSE	Actual Trans SSE	Actual Trans ISE	SS ISE	Trans Valve Travel	SS Valve Travel	Actual Tran VT
0		342.326328	0.623374	341.7029539	68.34059077	0.12467485	59.0364	14.0813	44.9551
0		352.256172	0.629571	351.6266015	70.3253203	0.125914142	59.7036	13.9629	45.7407
0		341.232922	0.499321	340.7336004	68.14672008	0.099864222	55.308	13.4721	41.8359
0.5		348.815828	0.814343	348.0014857	69.60029713	0.162868548	59.0918	12.6338	46.458
0.5		348.612546	0.502294	348.1102528	69.62205056	0.10045872	60.3449	10.594	49.7509
0.5		348.842805	0.63156	348.2112447	69.64224894	0.126312024	61.2743	10.6944	50.5799
1		351.082689	0.521345	350.5613433	70.11226865	0.10426907	59.7316	6.5309	53.2007
1		351.197808	0.506457	350.6913514	70.13827028	0.101291356	59.7145	5.8234	53.8911
1		352.351816	0.43394	351.9178752	70.38357504	0.086788056	60.8497	3.6872	57.1625
1.5		349.744203	0.782543	348.9616601	69.79233202	0.156508616	60.0162	4.2075	55.8087
1.5		352.82106	0.456396	352.3646637	70.47293274	0.091279264	60.8806	1.4098	59.4708
1.5		353.941851	0.543153	353.3986972	70.67973944	0.108630684	57.1737	5.6205	51.5532
2		357.879344	0.633227	357.2461168	71.44922336	0.126645458	62.2566	4.0169	58.2397
2		354.181299	0.800368	353.3809312	70.67618624	0.160073642	61.7763	5.1192	56.6571
2		357.641281	0.617347	357.0239341	71.40478682	0.123469402	64.3152	2.1943	62.1209
2.5		364.218549	0.937105	363.2814441	72.65628882	0.187421018	65.2378	3.0518	62.186
2.5		366.431478	0.661396	365.7700817	73.15401634	0.132279224	68.587	2.3313	66.2557
2.5		363.248564	0.967967	362.2805967	72.45611935	0.193593432	65.7935	5.0819	60.7116
3		367.669051	0.546798	367.1222529	73.42445057	0.10935953	68.5438	1.176	67.3678
3		372.876207	1.057736	371.8184705	74.3636941	0.211547276	73.8935	4.963	68.9305
3		373.169059	0.798264	372.3707745	74.47415491	0.159656832	75.8882	2.9918	72.8964
3.5		379.865211	0.95061	378.9146009	75.78292017	0.190122006	79.0607	5.0211	74.0396
3.5		377.831734	0.721945	377.1097896	75.42195792	0.144388938	75.0212	2.3488	72.6724
3.5		372.652389	1.011348	371.6410407	74.32820813	0.20226961	71.5871	4.0285	67.5586
4		387.244913	1.058394	386.1865186	77.23730372	0.211678824	80.6213	3.7053	76.916
4		386.817533	1.064372	385.7531604	77.15063208	0.212874424	81.6168	5.4618	76.155
4		382.492876	0.6255	381.8673757	76.37347513	0.125099986	76.0374	3.3163	72.7211

Table B.3

Experimental - Configuration B1 - First-order filter on PV

λ				Actual	Actual			Trans	SS	Actual
	Trans ISE	SS SSE	Trans SSE	Trans ISE	SS ISE	Valve Travel	Valve Travel	Tran VT		
0	342.4015992	0.81254273	341.5890565	68.31781129	0.16250855	56.4986	15.9686	40.53		
0	340.4950147	0.53766714	339.9573476	67.99146951	0.10753343	57.9334	14.2861	43.6473		
0	337.8400957	0.66227804	337.1778177	67.43556353	0.13245561	55.3919	16.5161	38.8758		
0.1	374.1772845	0.51688029	373.6604042	74.73208084	0.10337606	60.4217	11.9132	48.5085		
0.1	378.7653123	0.77210363	377.9932087	75.59864173	0.15442073	62.2671	13.4188	48.8483		
0.1	377.0335171	0.9037499	376.1297672	75.22595344	0.18074998	60.1549	13.2042	46.9507		
0.2	384.2050763	0.77581185	383.4292645	76.68585289	0.15516237	60.7791	10.9211	49.858		
0.2	382.0845365	0.49286815	381.5916684	76.3183367	0.09857363	61.1764	10.5773	50.5991		
0.2	384.031074	0.56941755	383.4616565	76.69233129	0.11388351	61.0908	11.1621	49.9287		
0.3	389.6256151	0.71294667	388.9126684	77.78253369	0.14258933	59.2521	9.3308	49.9213		
0.3	392.160007	0.4646163	391.6953907	78.33907814	0.09292326	60.9366	8.7567	52.1799		
0.3	391.5166334	0.47756589	391.0390675	78.2078135	0.09551318	60.3606	8.9182	51.4424		
0.4	400.3909538	0.51993997	399.8710138	79.97420277	0.10398799	62.0309	7.4856	54.5453		
0.4	402.0757073	0.98194427	401.093763	80.21875261	0.19638885	60.8648	8.1223	52.7425		
0.4	403.1321617	0.73986778	402.3922939	80.47845878	0.14797356	61.6552	7.7953	53.8599		
0.5	412.4467823	0.44921929	411.997563	82.3995126	0.08984386	62.0739	5.6809	56.393		
0.5	412.8996917	0.56350156	412.3361901	82.46723803	0.11270031	62.0241	5.9257	56.0984		
0.5	416.4329935	0.57141019	415.8615833	83.17231666	0.11428204	63.375	5.5874	57.7876		
0.6	439.0652358	0.63861552	438.4266203	87.68532406	0.1277231	66.293	5.2236	61.0694		
0.6	439.7386483	0.66150598	439.0771423	87.81542846	0.1323012	65.6822	5.2719	60.4103		
0.6	440.7566418	0.46192527	440.2947165	88.05894331	0.09238505	66.1416	4.7449	61.3967		
0.7	489.615247	0.48653714	489.1287099	97.82574197	0.09730743	74.3162	3.6964	70.6198		
0.7	487.8635113	0.43567086	487.4278404	97.48556809	0.08713417	72.9664	3.4152	69.5512		
0.7	489.1285039	0.69967988	488.428824	97.6857648	0.13993598	73.9562	4.01	69.9462		
0.725	502.2283059	0.59662214	501.6316838	100.3263368	0.11932443	73.8114	3.7381	70.0733		
0.725	504.6920529	0.60529221	504.0867607	100.8173521	0.12105844	75.4319	3.7906	71.6413		
0.725	504.6250899	0.58921691	504.035873	100.8071746	0.11784338	74.189	3.6205	70.5685		
0.75	527.5602063	0.52833306	527.0318732	105.4063746	0.10566661	78.5036	3.1546	75.349		
0.75	523.8347387	0.63092757	523.2038111	104.6407622	0.12618551	77.2498	3.7326	73.5172		
0.75	527.7173539	0.7264053	526.9909486	105.3981897	0.14528106	78.6611	3.8728	74.7883		
0.775	562.7752613	0.62183151	562.1534298	112.430686	0.1243663	84.1333	3.3732	80.7601		
0.775	559.6697688	0.57068256	559.0990862	111.8198172	0.11413651	82.4366	3.1149	79.3217		
0.775	557.1997357	0.79079045	556.4089453	111.2817891	0.15815809	82.9705	3.7321	79.2384		
0.8	592.7241811	0.84986303	591.8743181	118.3748636	0.16997261	87.7718	3.8522	83.9196		
0.8	583.8431998	0.61634369	583.2268561	116.6453712	0.12326874	85.1537	3.127	82.0267		
0.8	586.9744105	0.83415654	586.140254	117.2280508	0.16683131	86.366	3.6074	82.7586		

Table B.4

Experimental - Configuration B2 - CUSUM filter on MV

Trigger		Trans ISE	SS SSE	Actual Trans SSE	Actual Trans ISE	SSISE	Trans Valve Travel	SS Valve Travel	Actual Tran VT
0		342.4016	0.81254273	341.5890565	68.31781129	0.16250855	56.4986	15.9686	40.53
0		340.49501	0.53766714	339.9573476	67.99146951	0.10753343	57.9334	14.2861	43.6473
0		337.8401	0.66227804	337.1778177	67.43556353	0.13245561	55.3919	16.5161	38.8758
1		379.30146	0.58178784	378.7196743	75.74393485	0.11635757	56.6739	6.5371	50.1368
1		377.34598	0.61125745	376.7347235	75.34694469	0.12225149	55.8758	6.3672	49.5086
1		373.29194	0.66046673	372.6314722	74.52629443	0.13209335	55.9264	6.5989	49.3275
1.5		401.21336	0.62051896	400.5928431	80.11856863	0.12410379	57.9282	3.6786	54.2496
1.5		395.78512	0.84729499	394.9378209	78.98756418	0.169459	60.2159	5.6541	54.5618
1.5		395.25001	0.52377313	394.7262325	78.94524649	0.10475463	57.8119	4.6099	53.202
2		413.56176	0.86119838	412.7005587	82.54011174	0.17223968	60.7891	3.0045	57.7846
2		418.55801	0.72739835	417.8306091	83.56612181	0.14547967	62.7031	3.61	59.0931
2		414.88037	0.72386491	414.1565045	82.8313009	0.14477298	62.5302	3.9835	58.5467
2.5		414.4539	0.69075884	413.763143	82.75262859	0.13815177	61.8474	3.1537	58.6937
2.5		424.86163	0.57801553	424.2836192	84.85672383	0.11560311	65.0763	3.1107	61.9656
2.5		419.55691	0.79033581	418.7665757	83.75331514	0.15806716	63.8445	2.5682	61.2763
2.75		433.52489	0.46357364	433.0613209	86.61226417	0.09271473	63.333	1.4369	61.8961
2.75		428.6324	0.62785681	428.0045408	85.60090816	0.12557136	62.2681	3.3621	58.906
2.75		426.96211	0.67218259	426.2899224	85.25798448	0.13443652	63.5224	2.4944	61.028
3		429.30805	0.69771265	428.6103406	85.72206811	0.13954253	64.7597	3.151	61.6087
3		433.79474	0.61438057	433.1803589	86.63607179	0.12287611	61.5204	2.2604	59.26
3		432.17785	0.52589219	431.6519582	86.33039164	0.10517844	60.5533	2.0359	58.5174
3.25		434.92626	0.54165506	434.3846021	86.87692043	0.10833101	63.5757	1.646	61.9297
3.25		432.03441	0.63651021	431.3979011	86.27958022	0.12730204	60.7548	1.8869	58.8679
3.25		431.8087	0.58212791	431.2265704	86.24531408	0.11642558	63.1232	3.0093	60.1139
3.5		437.77669	0.64050109	437.1361931	87.42723862	0.12810022	61.5465	2.0161	59.5304
3.5		436.84473	0.47953966	436.3651917	87.27303835	0.09590793	63.227	1.0387	62.1883
3.5		438.70246	0.56373763	438.1387236	87.62774471	0.11274753	63.127	0.4895	62.6375
3.75		446.1125	0.62610283	445.486396	89.09727919	0.12522057	67.7967	1.7212	66.0755
3.75		443.87221	0.69978613	443.1724268	88.63448535	0.13995723	66.7023	3.5212	63.1811
3.75		442.23084	0.5458379	441.6850037	88.33700074	0.10916758	65.8735	1.7272	64.1463
4		445.24924	0.4323797	444.8168614	88.96337228	0.08647594	65.5141	0.5641	64.95
4		438.81311	0.50491354	438.3081999	87.66163997	0.10098271	62.412	1.2431	61.1689
4		441.01607	0.4580605	440.5580069	88.11160138	0.0916121	63.3879	1.4127	61.9752

APPENDIX C

TABULATION OF SIMULATION RESULTS

Symbols used in Tables:

Trans	: Transient
SS	: Steady state
ISE	: Integral of square of error
VT	: Valve travel
Actual transient ISE	: (Trans ISE - SS ISE)
Actual transient valve travel	: (Trans valve travel – SS valve travel)

Table C.1

Simulation - Configuration A1 - First-order filter on PV

				Actual		Tran	SS	Actual
λ		Trans ISE	SS ISE	Trans ISE		Valve Travel	Valve travel	Trans VT
0		62.1251	0.1462	61.9789		30.0237	8.3781	21.6456
0.4		62.7453	0.1596	62.5857		32.0272	6.3344	25.6928
0.6		64.269	0.1751	64.0939		35.0053	5.464	29.5413
0.7		66.6146	0.1902	66.4244		37.5502	4.958	32.5922
0.75		69.1259	0.1999	68.926		40.203	4.6856	35.5174
0.8		74.0502	0.2096	73.8406		45.4249	4.309	41.1159
0.825		78.448	0.2133	78.2347		49.3352	4.0445	45.2907
0.85		85.4584	0.2098	85.2486		54.3747	3.669	50.7057
0.875		97.4772	0.2241	97.2531		62.0831	3.4803	58.6028

Table C.2

Simulation - Configuration A2 – CUSUM filter on PV

				Actual		Trans	SS	Actual
Trigger		Trans ISE	SS ISE	Trans ISE		Valve Travel	Valve travel	Trans VT
0		62.1251	0.1462	61.9789		30.0237	8.378	21.6457
0.5		62.1326	0.1527	61.9799		30.1033	7.6747	22.4286
1		62.1976	0.1536	62.044		31.2345	7.1775	24.057
1.5		62.2046	0.1719	62.0327		30.5558	6.9779	23.5779
2		62.5149	0.2043	62.3106		36.7101	6.771	29.9391
2.5		62.7764	0.166	62.6104		38.9142	6.4482	32.466
2.75		62.779	0.2558	62.5232		39.4148	7.1716	32.2432
3		63.0263	0.2272	62.7991		39.0666	6.9251	32.1415
3.5		63.4729	0.2516	63.2213		39.8201	7.1092	32.7109

Table C.3

Simulation - Configuration B1 – First-order filter on MV

				Actual		Tran	SS	Actual
λ		Trans ISE	SS ISE	Trans ISE		Valve Travel	Valve travel	Trans VT
0		62.1357	0.1462	61.9895		30.0406	8.3781	21.6625
0.4		65.2947	0.1596	65.1351		31.5606	6.3344	25.2262
0.6		69.1568	0.1751	68.9817		33.35	5.464	27.886
0.7		72.9708	0.1902	72.7806		34.9755	4.9576	30.0179
0.75		76.0762	0.1999	75.8763		36.7898	4.6869	32.1029
0.8		80.8914	0.2095	80.6819		39.8696	4.3057	35.5639
0.825		84.4148	0.2109	84.2039		41.8568	4.0104	37.8464
0.85		89.1698	0.2102	88.9596		44.3859	3.6691	40.7168
0.875		95.8215	0.2423	95.5792		48.2322	3.6472	44.585

Table C.4

Simulation - Configuration B2 – CUSUM filter on MV

				Actual		Tran	SS	Actual
Trigger		Trans ISE	SS ISE	Trans ISE		Valve Travel	Valve travel	Trans VT
0		62.1322	0.1462	61.986		30.0182	8.3781	21.6401
0.5		62.1674	0.1527	62.0147		28.7148	6.8198	21.895
1		62.3153	0.1589	62.1564		28.9299	6.6952	22.2347
1.5		63.6343	0.1674	63.4669		29.7269	6.0222	23.7047
2		65.4517	0.1847	65.267		29.7393	5.8988	23.8405
2.5		65.5919	0.1861	65.4058		29.7733	5.5896	24.1837
2.75		66.0467	0.1842	65.8625		31.0771	5.6389	25.4382
3		66.8795	0.1921	66.6874		29.8719	5.8954	23.9765
3.5		70.7516	0.1989	70.5527		35.1054	4.7936	30.3118

VITA



Nagappan Muthiah

Candidate for the Degree of

Master of Science

Thesis: APPLICATION AND EVALUATION OF STATISTICALLY-BASED SIGNAL FILTERS ON A PILOT SCALE FLOW LOOP

Major Field: Control Systems Engineering

Biographical:

Personal: Born in Trichy, Tamil Nadu, India, On 6th February 1980, the son of Mr. M. Nagappan and Mrs. N. Susila.

Education: Graduated from D.A.V Public School, New Delhi, India in May 1997; received Bachelor of Engineering degree in Instrumentation and Control from University of Madras, India in April 2001. Completed the Requirements for the Master of Science degree with a major in Control Systems at Oklahoma State University in July 2004.

Professional Membership: Instrumentation Systems and Automation Society.