TUNING A THREE-PHASE SEPARATOR LEVEL CONTROLLER VIA PARTICLE SWARM OPTIMIZATION (PSO) ALGORITHM

By

LUVENRAJ SATHASIVAM

FINAL PROJECT REPORT

Submitted to the Department of Electrical & Electronic Engineering
in Partial Fulfillment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical & Electronic Engineering)

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

© Copyright 2013
by
LUVENRAJ SATHASIVAM, 2013

CERTIFICATION OF APPROVAL

TUNING A THREE-PHASE SEPARATOR LEVEL CONTROLLER VIA PARTICLE SWARM OPTIMIZATION (PSO) ALGORITHM

by

LUVENRAJ SATHASIVAM

A project dissertation submitted to the
Department of Electrical & Electronic Engineering
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronic Engineering)

Approved:

AP Dr. Irraivan Elamvazuthi
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

September 2013

CERTIFICATION OF ORIGINALITY

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

____

LUVENRAJ SATHASIVAM

Abstract

Three-Phase Separators are used to separate well crudes into three portions; water, oil, and gas. A suitable control system should be in place to ensure the optimum function of the Three-Phase Separator. The current PID tuning technique does not provide an optimum system response of the separator. Overshoot response, offset, steady-state error and system instability are some of the problems faced. Besides, the current method used is purely based on trial and error which is time consuming. There is room for improvement of the current PID tuning technique. An artificial intelligence (AI) PID tuning technique called Particle Swarm Optimization (PSO) is introduced to improve the system response of the Three-Phase Separator. The PSO algorithm mimics the behaviour of bird flocking and fish schooling striving for its global best position. In our case, the global best position is replaced with the optimized PID tuning parameters for the separator. The PSO algorithm has been used in several other applications such as the Brushless DC motor and in the Control Ball & Beam system. It has proven to be an effective tuning technique. Tuning of the Three-Phase Separator via PSO could prove to be an effective solution for Oil & Gas industries.

ACKNOWLEDGEMENT

I would like to convey my appreciation to my supervisor, AP. Dr. Irraivan Elamvazuthi for being very supportive and guiding me throughout the project time frame. Plenty of knowledge has been acquired via this project. Without the constant motivation and support from my Supervisor, a successful completion of this project would not have been possible. Finally I would like to thank my family, course mates, and peers who have constantly motivated me from the beginning through to the completion of this project.

TABLE OF CONTENTS

APPENDICES

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS AND NOMENCLATURES

PSO	Particle Swarm Optimization
IMC	Internal Model Control
BFD	Butterworth Filter Design
GUI	Graphical User Interface
PID	Proportional- Integral-Derivative
GA	Generic Algorithm
AI	Artificial Intelligence

Chapter 1

Introduction

1.1 Background Study

Control System plays an important part in aiding the function of a particular equipment, hardware or process. It ensures that a particular process is at its optimum functional level. Besides that, it helps a system compensate for disturbance be it externally or internally. There are two types of Control System, the Open-Loop Control System and the Closed-Loop Control System. General examples of the Open-Loop Control System include the remote controls, switches and etc. On the other hand, water level monitoring and temperature monitoring are typical examples of Closed-Loop Control Systems. Closed-Loop control systems come with a feedback loop equipped with a sensor in it. This feedback loop provides information which helps identify errors in a system by comparing the Process Variable and Set Point.

My study is related to the Control System of a Three-Phase Separator. A Three-Phase Separator is typically a vessel used in the Oil & Gas Industry for separation process. It can be found vertically, horizontally, and spherically. The most commonly used vessel is the vertical vessel as it occupies lesser ground space. Fluids (crude) from wells are flushed into the vessel via tentative pipelines. This fluid then separates accordingly due to its difference in densities. The gas occupies the top most-layer in the vessel followed by oil and water respectively. The uniqueness of a Three-Phase Separator compared to a Two-Phase Separator is that in a Three-Phase Separator, the separation of oil, gas, and water takes place simultaneously whereas in the Two-Phase Separator only crude gas is totally separated whereas there is still an element of liquid mix-up between the oil & water. In order to completely separate the mixture of oil & water, another separation process needs to takes place.

Industries have found it hard to completely separate the mixture of oil and water in recent times and this makes downstream refinery work even tougher. The separation process is a tough task due to the control systems inability to constantly adapt to internal changes such as pressure rise, temperature rise, and changes in the vessel water level. In addition to this, failure to overcome the internal system dynamics such as dead time also is a major contributor. Current PID tuning methods have highlighted certain limitations that can be overcome by developing a suitable and reliable control algorithm for the optimum function of the Three-Phase Separator. Further discussions on the proposed Control Algorithm would be done in Part (4) and Part (5).

1.2 Problem Statement

Figure 1.1 shows the Horizontal Three-Phase Separator.

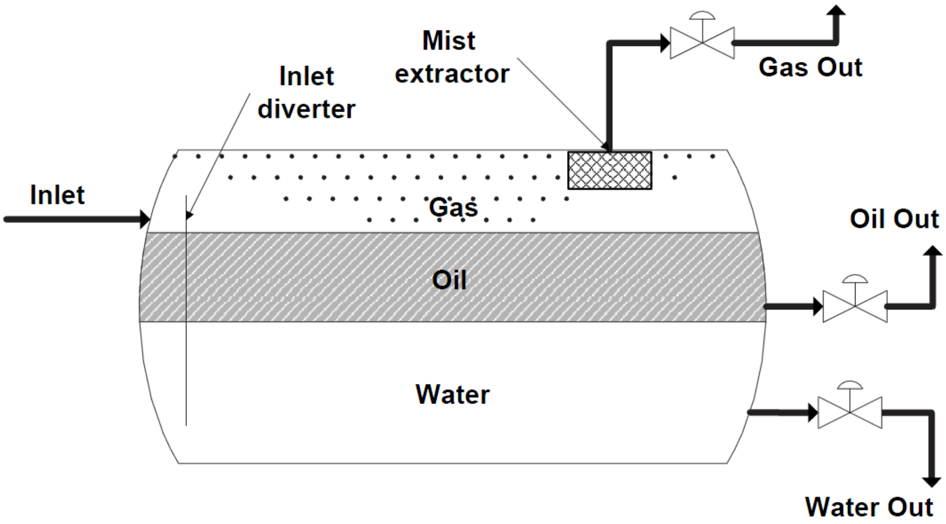


Figure 1.1: Horizontal Three-Phase Separator

The Three-Phase Separator is the most important vessel involved in the upstream environment as it does the preliminary separation of crude flowing in from wells. Liquid channelled into the separator hits the inlet diverter and the change in momentum drives the initial separation of gas, water, and oil. Problems faced by many Oil & Gas industries are to get the best performance out of the separator. Issues such as gas blow-by, liquid carryover, formation of emulsion, paraffin build up and etc are consistently observed within the separator. Two factors that contribute to such cases include improper separator design, and inadequate control system. This study focusses on the control aspects of the separator. An analysis was done to study on the reliability of the current PID tuning method used in the separator. Most separators used the Ziegler-Nichols (trial and error) PID tuning technique. A drawback of this method is that it is based on trial-and-error. Tuning parameters; Kp, Ki, and Kd are randomly assigned to get the best performance out of the controller. It is impossible to obtain the optimized PID tuning parameters via this method. Besides that, the Ziegler-Nichols method is also time consuming as it requires several trials before the best PID tuning parameters are chosen. Furthermore, the performance trend of the separator tuned using this method is not effective enough. High rise time, overshoot response, and offset are observed.

To counter this problem, industries have set a range of allowable deviation of process variables from the set point. This in turn did not give the best performance out of the separator. Although it seems a minute problem, Oil & Gas industries have found it hard to find a way to address this issue. Various methods of an effective PID controller tuning are still being researched on.

1.3 Objectives & Scope of Study

Figure 1.2 shows the aim and scope of study of the project in a flowchart.

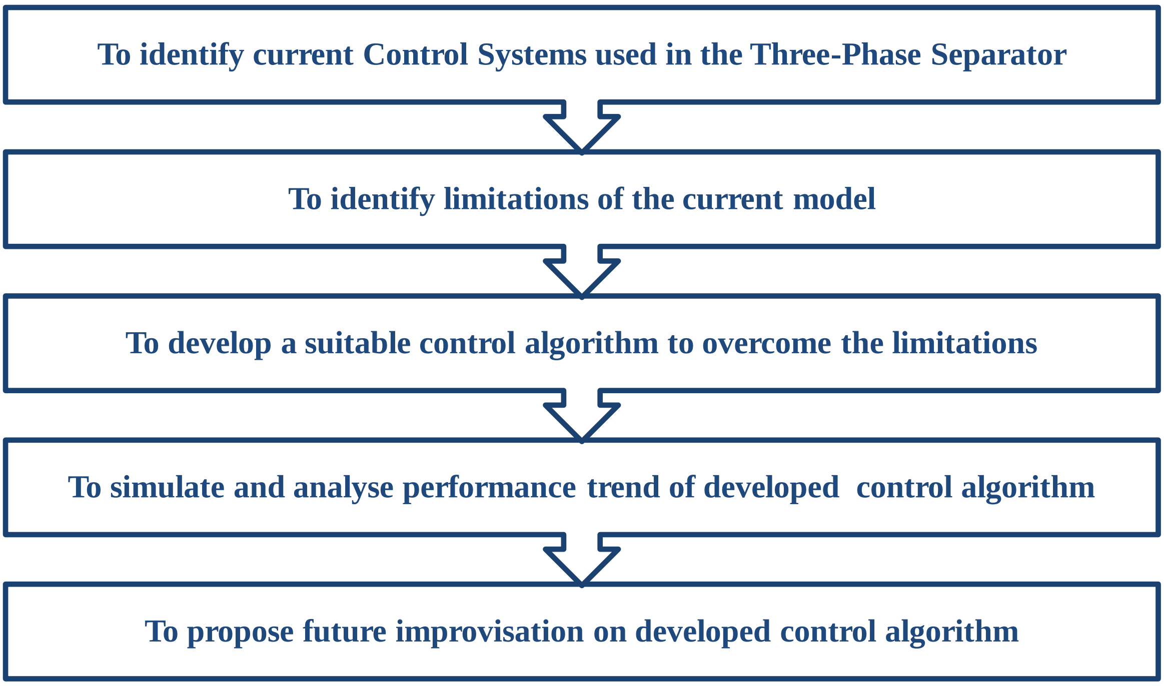


Figure 1.2: Objectives Flowchart

There are five main objectives to be covered throughout the course of the project. The first objective is to identify the current PID tuning technique used in the Three-Phase Separator. The next step would be to analyse the performance trend of the present technique. The limitations of the current model are then used as a benchmark to develop a new suitable control algorithm. Upon development, detailed analysis is done via simulations using MATLAB-Simulink to prove that the developed control algorithm produces the desired output. It should also be able to overcome limitations of the current algorithms used. The limitations of the developed model are then identified and future improvisations to overcome the limitations are recommended.

Chapter 2

Literature Review

2.1 Control System

Over the years, different methods have been implemented for the control action of the Three-Phase Separator. There has been an improvisation in terms of desired system performance between these methods. On the other hand, there also have been some limitations which can be further analysed and improved. This chapter discusses the limitations of a few control algorithms which are being used in the Three-Phase Separator.

In a Control System, there are four main blocks which are interdependent over one another. The four main blocks are the Controller, Final Element, the Process, and a typical feed-back block equipped with a sensor in it. Figure 2.1 shows the control actions of these elements;

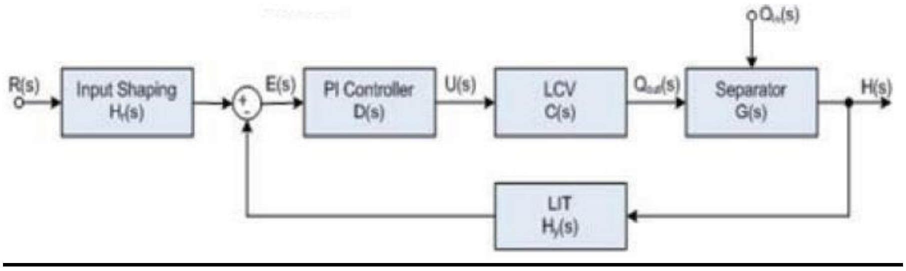


Figure 2.1: Three-Phase Separator Control Block diagram

There are three types of controllers namely the Proportional gain (P), Integral time (I), and Derivative time (D) controller. The (P) controller is used to estimate the present error of a system whereas the (I) controller is the sum of errors over a specific period of time. The Derivative (D) controller is used to predict the future error of a system based on the trend of errors occurring in the system. The need of each controller depends on the desired control action required for a system. Some Three-Phase Separators use the PID controller whereas in most cases only the PI controller is required.

2.2 Related Work

2.2.1 PID Tuning

Table 2.1 shows previous research related to PID tuning of three phase separator.

No	Author(s)	Year	Techniques Used	Advantages	Disadvantages
1	Mendes P. R. C. Normey-Rico J. E. Plucenio A. Carvalho R. L.	2012	• Practical non-linear model predictive controller (PNMPC) • Disturbance predictor-estimator via feed-forward action • Hammerstein model	• better disturbance damping • better performance in steady condition • only a simple model of separator and Quadratic Programming Algorithms are needed	• premature convergence • Feed-forward loop not sufficient enough for system. Requires an addition feedback loop
2	Zhenyu Yang	2010	• PI control • Trial and error method • Butterworth filter design method • Internal Model Control (IMC) method	• Tuning via trial and error method improves the overshoot value • BFD leads to smoother outflow-rate and better level control • IMC method results in no frequency distortion	• Improved but not optimized overshoot response • High rise time observed
3	Atalla F. Sayda James H. Taylor	2007	• Dynamic Mathematical Model	• Increased oil outflow • Decrease in the flashed gas amounts	• Increase in water discharge molar flow • Sophisticated model
4	Rosendo Monroy-Loperena Rocio Solar Jose Alvarez-Ramirez	2004	• balanced control scheme • parallel control structure • simultaneous feedback manipulations • concept of self-optimizing control	• Provides a stable, unitary, steady-state-gain • can deal better with input saturation • vapor boilup rate is significantly reduced	• rate of convergence to the desired set point is reduced, which can lead to reduction in robustness in control margin of the process

Table 2.1: Current PID Tuning Methods

Different types of control algorithm used for PID tuning are discussed in [1] . A study was done on first order, second order, and third order systems by comparing their Integral Absolute Error (IAE) values. The method was limited to Single Input Single Output (SISO) systems. Among the control algorithm studied was the Closed-Loop Ziegler Nichols (Z-N) method. Z-N method is the most commonly used control algorithm these days due to its near accuracy to a systems desired performance output. However the Z-N method possesses some limitations as it is not applicable for open loop systems which are unstable. Besides that, it only guarantees marginal close loop stability as this method does not compensate for external disturbance and set point changes. It is also time consuming as it involves trial-and-error method for parameter selection. The next method studied was the Chien-Hrones-Reswick (C-H-R) auto tuning method. This method was similar to the Open-Loop Ziegler Nichols method. This technique provides a fast response but it also presents an overshoot system response in the range 10%-20%. Some Three-Phase Separators are modeled with respect to the desired performance required from it. In such cases, a simple control system is sufficient enough to monitor its performance level. In [2] for an example, the modeling aspect of the separator focusses on two main elements; the liquid-liquid separation and the vapour-liquid separation. The American Petroleum Institute design guidelines were encrypted in the modeling aspect of liquid-liquid separation. In order to monitor the performance level of the modeled separator, a simple PI controller was introduced. The first phase (vapour-liquid separation), was designed to control the liquid level and pressure level in the vessel by adjusting the level control valve and controlling the amount of gas discharge. Two PI controllers were used in this case. The second phase (liquid-liquid separation) was designed to control the interface level of water/oil, vessel pressure, and oil level. This aspect was monitored by three PI controllers.

A comparative study was also done on the Three-Phase Separator to analyse the effectiveness of three different control approaches; the conventional PI controller, Butterworth Filter design (BFD), and Internal Model Control (IMC) [3]. A horizontal separator namely the V-3440 vessel was used. The piping and instrumentation diagram (P&ID) of the separator can be seen below;

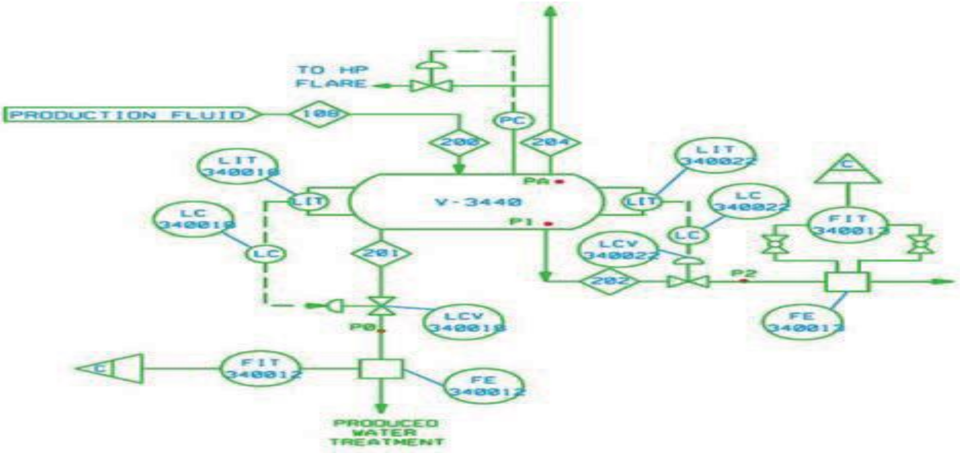


Figure 2.2: P&ID of V-3440

Source: Y. Zhenyu, M. Juhl, and B. Lohndorf, "On the innovation of level control of an offshore three-phase separator," in *Mechatronics and Automation (ICMA), 2010 International Conference on*, 2010, pp. 1348-1353

The two main control objectives were to ensure a smooth liquid outflow rate and to maintain a permissible range of water in the vessel. As can be seen typical flow indicator (FIT 340013) and level indicator (LIT 340022) are used to monitor the water outflow rate and water level in the vessel. The equations used to represent the Three-Phase Separator process (G2(s)) and disturbances (G1(s)) are as follow;

$$G1(s) = \frac{H(s)}{Qin(s)} = \frac{1}{47.55s+1.81} \qquad (1)$$

$$G2(s) = \frac{H(s)}{U(s)} = \frac{-10.82}{47.55s+1.81} \qquad (2)$$

The conventional PI algorithm and trial-and-error method proved ineffective as it produced high overshoot values and bandwidth. There was an improvement in system output when the BFD and IMC method was used. However, the bandwidth measured was still reasonably high due to zero's effect. The control block diagram of an IMC system is as shown;

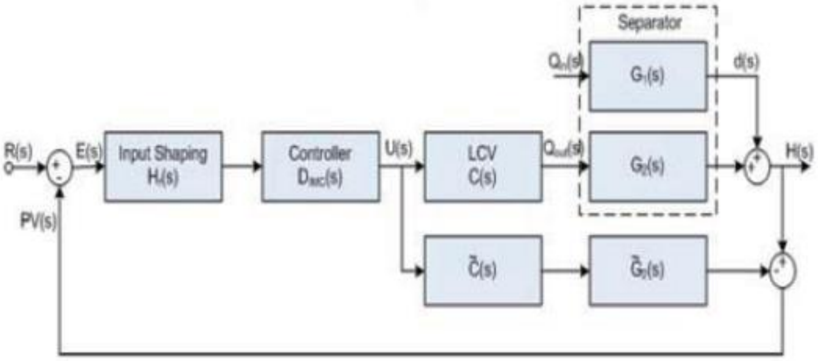


Figure 2.3: Block diagram of IMC

Source: Y. Zhenyu, M. Juhl, and B. Lohndorf, "On the innovation of level control of an offshore three-phase separator," in *Mechatronics and Automation (ICMA), 2010 International Conference on*, 2010, pp. 1348-1353

IMC is similar to a conventional PI approach; the only difference being there is an additional process model block present. The process model estimates internal system disturbance, combines it with the external disturbance detected before going through a summing junction and sending a feedback to the controller. This makes the IMC method applicable for non-linear systems. Applications of IMC method in the Reactor & Separator Process, Continuous Distillation Column, and Heat Exchanger System can be reviewed in [4-6] for a better overview of its control scheme.

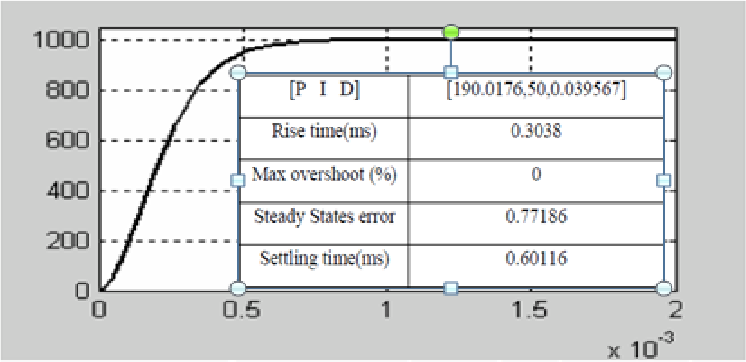
2.2.2 PID Tuning using Artificial Techniques

The applications of PID tuning using artificial intelligent techniques were also reviewed. Table 2.2 shows previous research using AI technique related to PID.

No	Author	Year	Techniques Used	Merits	Demerits
1	Rana M. A., et al	2011	Particle Swarm Optimization (PSO)	PSO has best control performance, negligible transient	Sufficiently high rise time (second scale)
2	F. Hongqing., et al	2008	Particle Swarm Optimization (PSO)	PSO has the fastest convergence speed for test PSO has a sufficiently small IAE value	High settling time
3	Nasri, M., et al	2007	Particle Swarm Optimization (PSO)	PSO has the best dynamic performance compare to generic algorithm and linear quadratic regulator Small rise time needed, no overshoot response, fast settling time(ms scale)	Steady-state error recorded
4	Kim & Cho	2005	Bacteria Foraging Algorithm (BFA)	BFA produced the best step response & ISE value (between 0.01-0.02)	Overshoot response observed

Table 2.2: Artificial Intelligence PID Tuning Methods

Among the intelligent techniques reviewed were Bacteria Foraging Algorithm (BFA) and Particle Swarm Algorithm (PSO). In [7] for an instance, the application of PSO in a Linear Brushless DC motor was reviewed. The optimized PID tuning parameters to control the speed of the DC motor was obtained using the PSO theory. MATLAB-SIMULINK was used to design the model and comparison of the model with Generic Algorithm (GA) and linear quadratic regulator (LQR) method was performed. The results obtained were as follow;



[P I D]	[190.0176,50,0.039567]
Rise time(ms)	0.3038
Max overshoot (%)	0
Steady States error	0.77186
Settling time(ms)	0.60116

Figure 2.4: Results of PSO in Brushless DC motor

The PID tuning parameters shown in the figure were computed using the PSO method. Results proved that PSO had a better performance trend compared to the GA and LQR method. Other application of PSO can be reviewed in [8, 9].

Chapter 3

Research Methodology

The research focuses on improving the current PID tuning technique of the 3-Phase Separator by introducing an Artificial Intelligence (AI) technique known as Particle Swarm Optimization (PSO). This technique would aid the PID tuning process and would be able to replace the current tuning methods such as the Ziegler-Nichols method, Butterworth filter design method, Internal Model Control (IMC) and etc. The current tuning methods are time consuming and based on trial and error. This would not provide an optimum system response of the Three-Phase Separator.

3.1 Particle Swarm Optimisation (PSO)

3.1.1 Technique

The generic concept of PSO was explained in [9, 10] . Particle Swarm Optimization mimics the behaviour of bird flocks and fish schooling striving for its best global position in a g-space environment based on its previous flying experience. Two comparisons are made, one being the particles personal best position (pbest) and the other being the best position of particles within the swarm (gbest). In an attempt to drive these particles to their global best positions, their velocity is adjusted until pbest or gbest is achieved. Several iterations are performed at particular time interval until the desired position (parameter) is obtained. The two equations related to the velocity and global positions of the particle are as follow;

$$\mathit{position}_{new}[\ ] = \mathit{position}_{old}[\ ] + \mathit{velocity}_{new}[\ ] \tag{3}$$

$$\mathit{velocity}_{new}[\ ] = w \times \mathit{velocity}_{old}[\ ] + c_1 \times \mathit{rand} \times (\mathit{pbest}[\ ] - \mathit{position}[\ ]) + c_2 \times \mathit{rand} \times (\mathit{pbest}[\mathit{gbest}] - \mathit{position}[\ ]) \tag{4}$$

where;

w=inertia factor

c1 and c2=learning factor

3.1.2 PSO Method in PID tuning

The PSO technique would be an ideal way of PID tuning for the optimum function of the Three-Phase Separator.

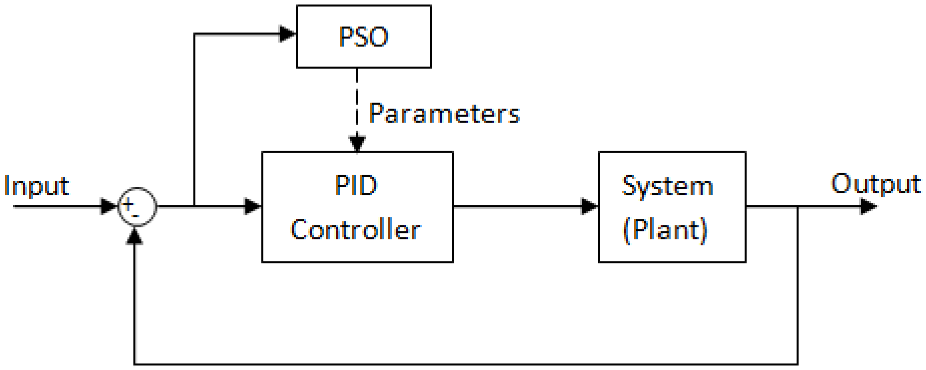


Figure 3.1: Implementing PSO in PID Controller

The block diagram above shows an overview of how PSO is to be implemented in the PID controller tuning. The measured process variable goes through the feedback loop into a summing junction. At the summing junction, the measured process variable is compared to the actual process variable known as the set point. Based on the error computed, the PID controller manipulates its’ Kp, Ki, and Kd values. The method in which the controller obtains these parameters is via the PSO technique. Based on the new controller tuning parameters, an action is taken on the final element, usually a Control Valve. The actual Process Variable is achieved at the output.

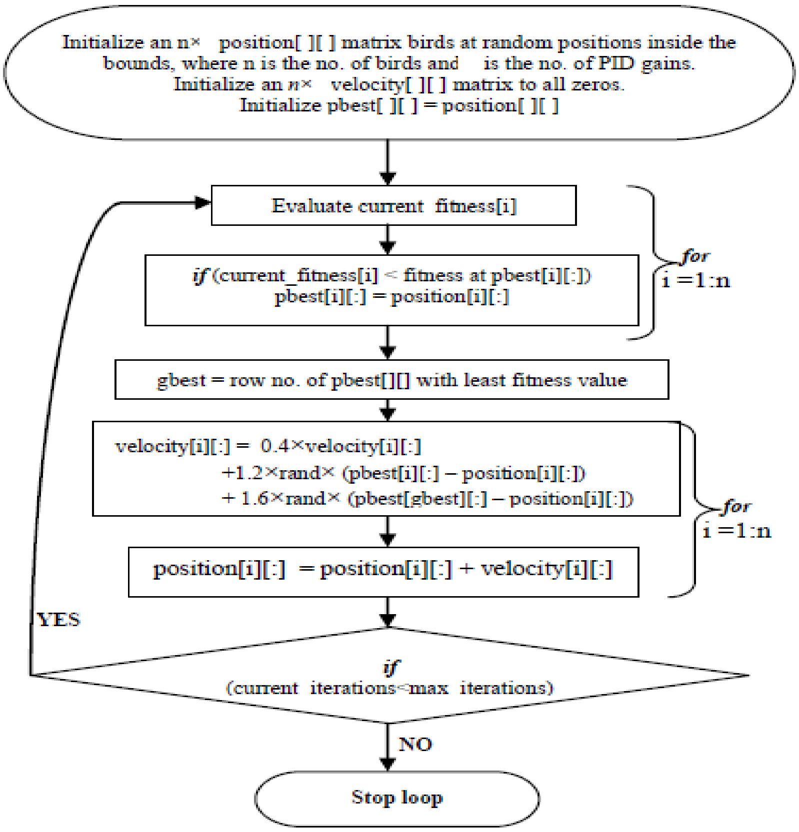


Figure 3.2: Flowchart of PSO algorithm

The implementation of PSO in MATLAB .m file is as follow. The first step involves the generation of the **n** × **m *position** matrix. **N** represents the number of birds (particles) and **M** represents the number of PID gains. If a typical PI controller is used then m=2 whereas if a PID controller is used m=3. The position can be a random number within the range of bounds.

The next step would be a replica of the first step, the only difference being the **n**×**m*position** matrix is replaced with the **n** × **m *velocity** matrix with zero as the initial condition.

The next equation generated would be to equate the pbest matrix to the velocity matrix. The particles current fitness is then evaluated. If the current fitness of the particle observed is lesser than the particles previous personal best fitness, then the new pbest of the particle would be at its current location. The groups fitness position is then evaluated. The gbest would be the row of the **n**th particle with the smallest fitness value. The particles are then assigned an arbitary velocity.

The new position of the particles is then computed by summing its current position with the particles new velocity. Finally if the current number of bird step is equal to the maximum number of bird step, the loop will be stopped and the tuning parameters are taken. If the number of iterations is lesser than the maximum number of iterations, the whole process mentioned has to be repeated.

3.2 Modeling of Three-Phase Separator

The Three-Phase Separator was modeled based on [3]. The control objective was to maintain the water level in the separator within its permissible range. The block diagram and the Piping & Instrumentation diagram of the model are shown in Figure 3.3 and Figure 3.4 respectively.

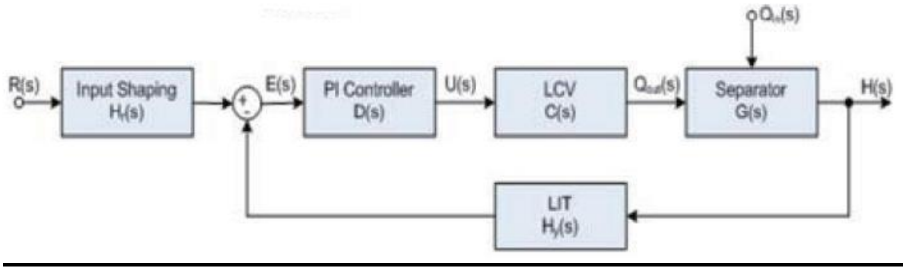


Figure 3.3: Block diagram of Three-Phase Separator

Source: Y. Zhenyu, M. Juhl, and B. Lohndorf, "On the innovation of level control of an offshore three-phase separator," in *Mechatronics and Automation (ICMA), 2010 International Conference on*, 2010, pp. 1348-1353

R(s) is the set point of the process. LCV and LIT are the level control valve and level indicator of the three phase separator. E(s) represents the error which is the deviation of water level from its actual level. Qin(s) is the unknown disturbance to the process. H(s) represents the current water level of the system.

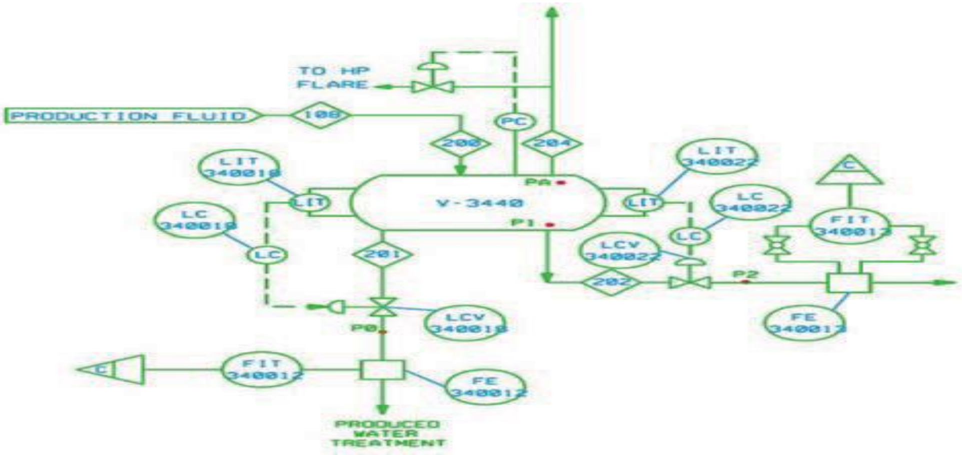


Figure 3.4: P&I diagram of the V-3440 Separator

Source: Y. Zhenyu, M. Juhl, and B. Lohndorf, "On the innovation of level control of an offshore three-phase separator," in *Mechatronics and Automation (ICMA), 2010 International Conference on*, 2010, pp. 1348-1353

The P&I diagram above shows the primary sensors, controllers and final elements associated with the V-3440 horizontal Three-Phase Separator. Since the process was modeled to control the water level within the separator, LIT 340018, LC 340018, and LCV 340018 are the primary sensor, controller and final element considered.

The mass balance principle was used to determine the water volume dynamics inside the separator.

$$\left(\frac{dV(t)}{dt}\right) \approx AL\frac{dH(t)}{dt} = Qin(t) - Qout(t) \qquad (5)$$

The rate of change in water volume within the separator is equal to the area (A) multiplied by the length (L) and the rate of change in height of the water in the separator.

The flow-dynamics theory was used to determine the water outflow rate over the valve. The equation is as follow;

$$Qout = Cvf(u)\sqrt{\frac{\Delta Pout}{\rho w}} \qquad (6)$$

Cv is the outlet valve discharge coefficient. Pout and rho w represent the pressure drop across the valve and density of water respectively.

The differential pressure over the valve was computed using the equation

$$\Delta Pout(t) = Pg(t) + \rho ogho(t) + \rho wgh(t) = Pw(t) \quad (7)$$

Finally the linearized model equation of the system was obtained by inserting specific system parameters.

$$47.55\ \frac{d\Delta h(t)}{dt} = Qin(t) - 1.81\Delta h(t) - 10.82\Delta u(t) \qquad (8)$$

The separator was modeled with a disturbance $Q_{in}(s)$ induced. The desired water level and actual water level in the separator are represented by R(s) and H(s) respectively. The transfer function of the process, G2(s) and the disturbance G1(s) are as shown below;

$$G1(s) \cong \frac{H(s)}{Qin(s)} = \frac{1}{47.55s+1.81} \qquad (9)$$

$$G2(s) \cong \frac{H(s)}{U(s)} = \frac{-10.82}{47.55s+1.81} \qquad (10)$$

3.3 Simulation Models

Simulink model’s on the current PID tuning techniques as well as the PSO technique was designed. Two separate models were designed to analyse the system response-one with no induced disturbance and one with an externally induced disturbance.

For the system with no induced disturbance, a unit step input signal was channelled into the system with transfer function G1(s) as shown. Out2 tracks the output variables used for the analysis of the system response. The Integral Squared Error (ISE) was tracked by comparing the measured process variable to the system’s set point.

A second model was designed to analyse the effectiveness of each PID tuning technique in response to an external disturbance G2(s).

3.3.1 Trial & Error Method

Figure 3.5 and Figure 3.6 shows the Simulink design for Trial & Error Method without induced disturbance and with induced disturbance respectively.

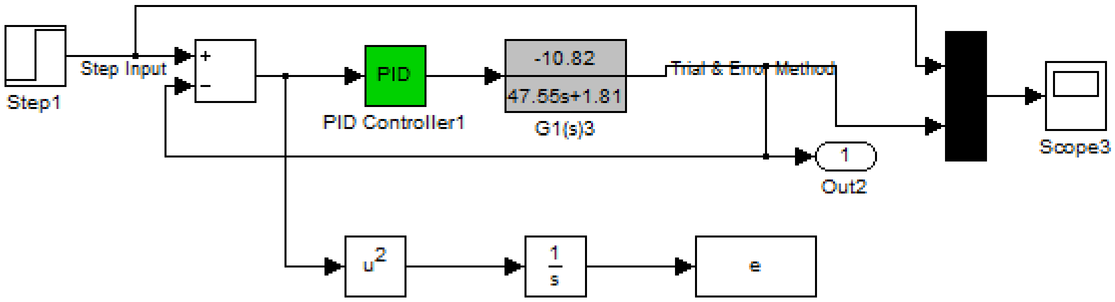


Figure 3.5: Simulink design for Trial & Error Method

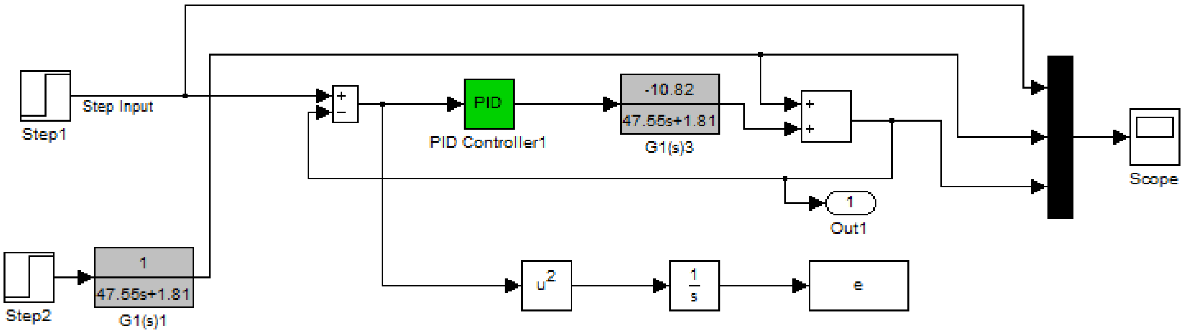


Figure 3.6: Simulink design for Trial & Error Method with Induced Disturbance

3.3.2 IMC Method

Figure 3.7 and Figure 3.8 shows the Simulink design for Internal Model Control Method without induced disturbance and with induced disturbance respectively.

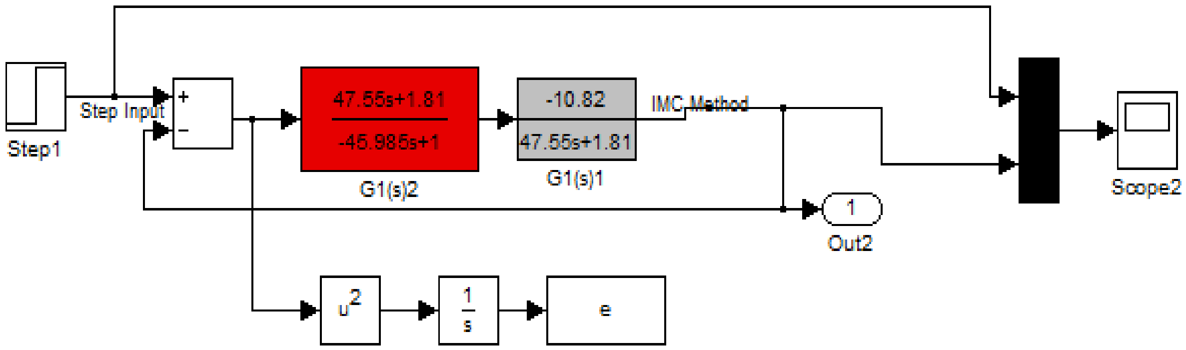


Figure 3.7: Simulink design for Internal Model Control Method

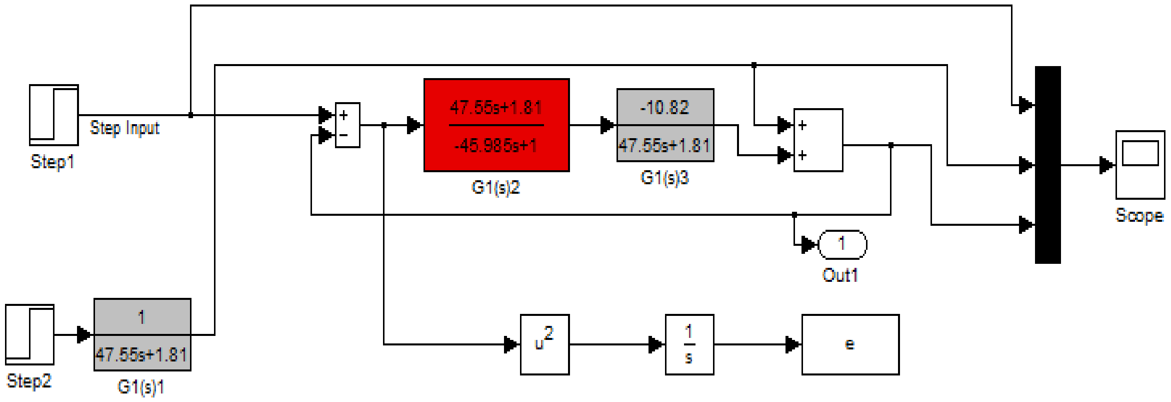


Figure 3.8: Simulink design for Internal Model Control Method with Induced Disturbance

3.3.3 Butterworth Filter Design Method

Figure 3.9 and Figure 3.10 shows the Simulink design for Butterworth Filter Design Method without induced disturbance and with induced disturbance respectively.

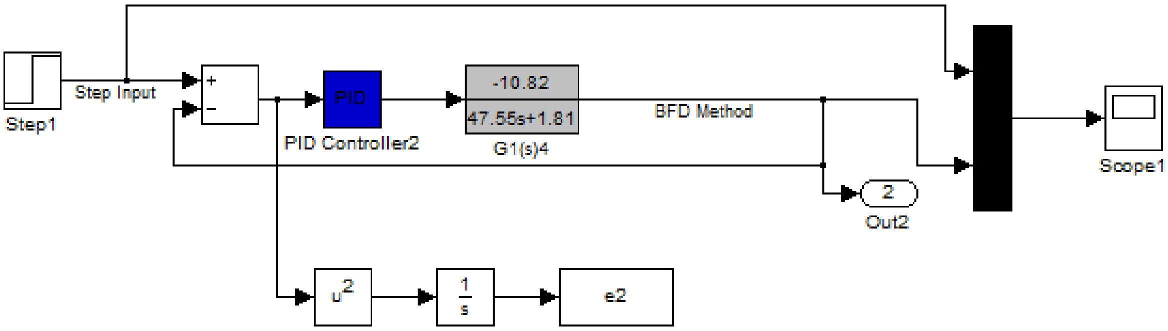


Figure 3.9: Simulink design for Butterworth Filter Design Method

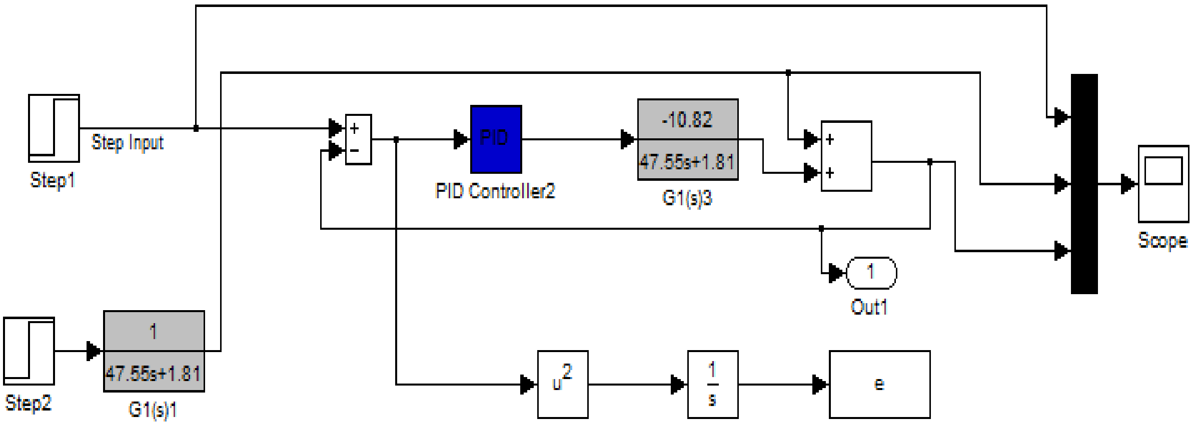


Figure 3.10: Simulink design for Butterworth Filter Design Method with Induced Disturbance

3.3.4 PSO Method

Figure 3.11 and Figure 3.12 shows the Simulink design for Particle Swarm Optimization Method without induced disturbance and with induced disturbance respectively.

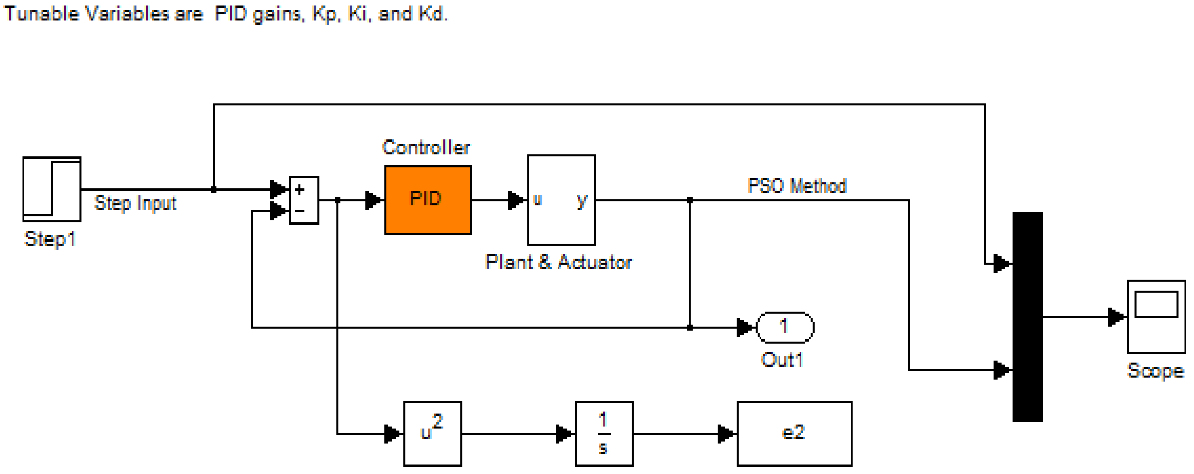


Figure 3.11: Simulink design for Particle Swarm Optimization Method

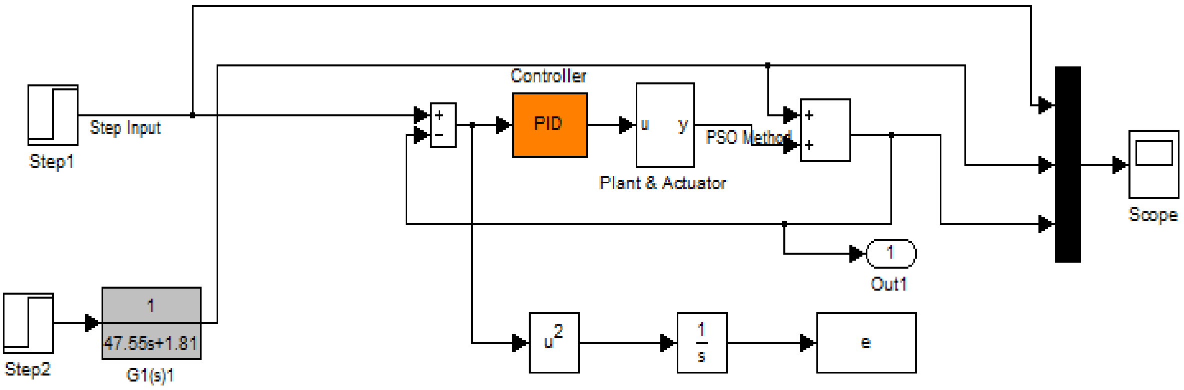


Figure 3.12: Simulink design for Particle Swarm Optimization Method with Induced Disturbance

3.3.5 Integrated Simulink Models

The integrated Simulink model for the four methods mentioned previously without an induced disturbance and with an induced disturbance is shown in Figure 3.13 and Figure 3.14 respectively.

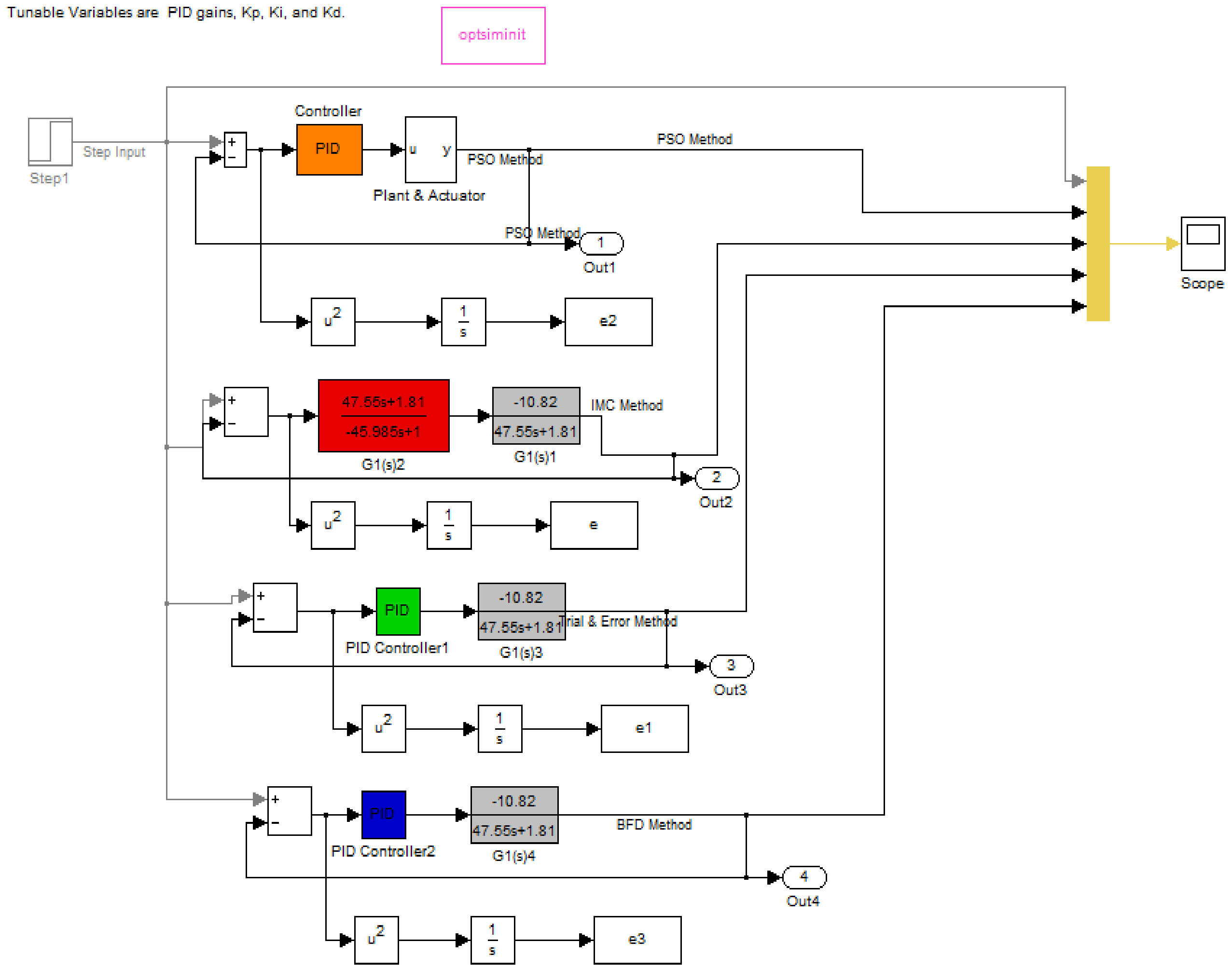


Figure 3.13: Integrated Simulink block

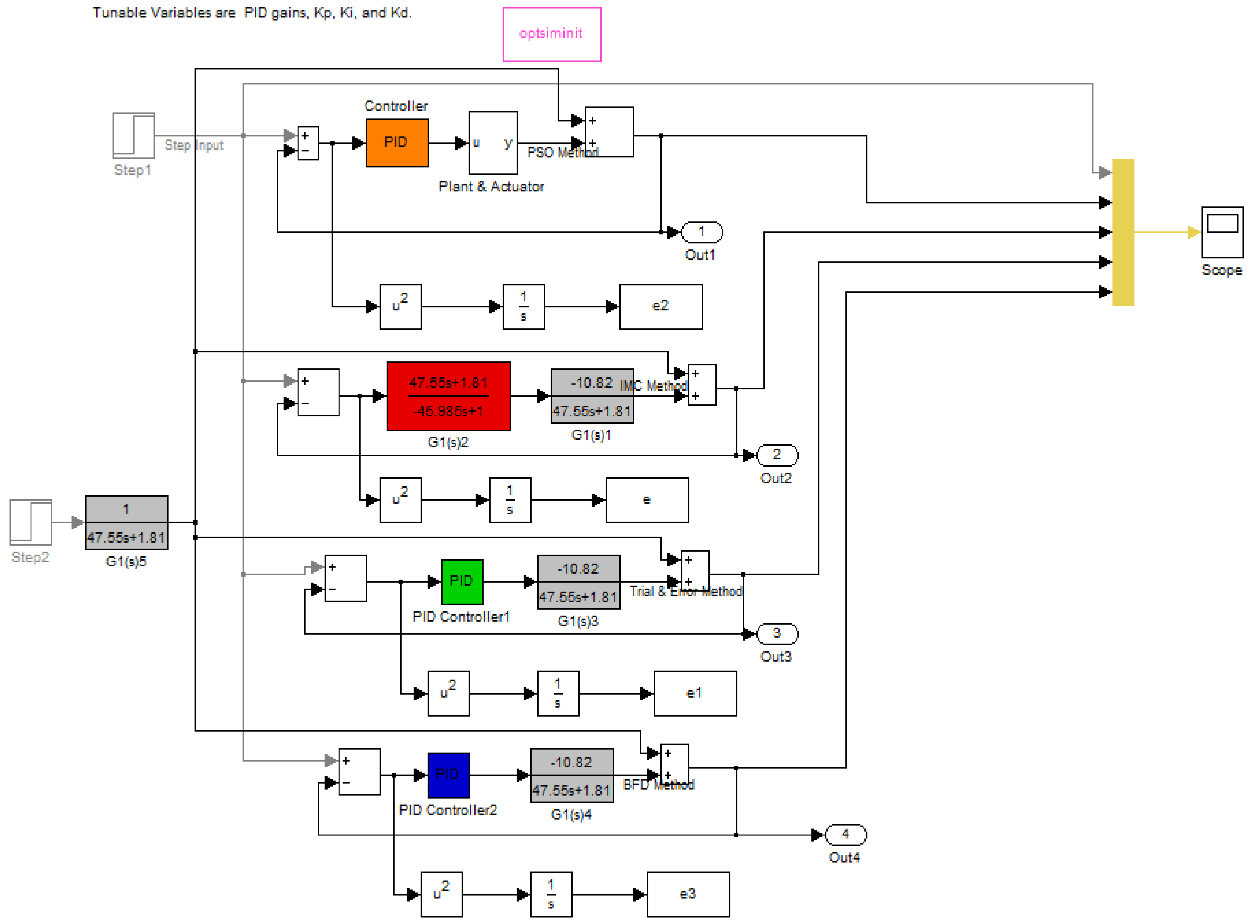


Figure 3.14: Integrated Simulink block with Induced Disturbance

The modeling aspect of the Three-Phase Separator was discussed in Chapter 3. The control objective was to maintain the water level within a permissible range in the V-3440 horizantal separator. Simulation models on the current tuning techniques as well as the developed PSO tuning technique was shown via MATLAB Simulink. The next chapter would provide a detailed analysis of the results of these models.

Chapter 4

Results and Discussions

4.1 Trial & Error Method

The results obtained from simulating the Three-Phase Separator Level Controller using Trial & Error method is discussed in Part 4.1.1 and 4.1.2.

4.1.1 Trial & Error Method without Induced Disturbance

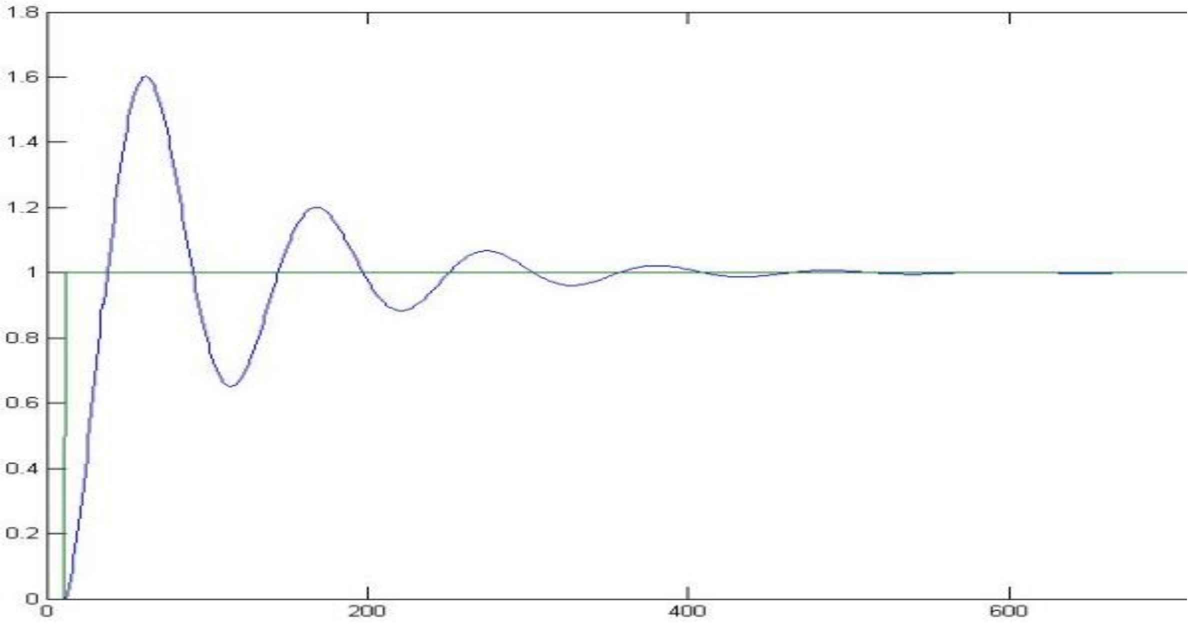


Figure 4.1: Response of Trial & Error Method

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
Trial & Error	2.4340	60%	5s	40s	0.02

Table 4.1.Results for Trial & Error Method

The trial and error method with no induced disturbance resulted in an overshoot response of 60% with an Integrated Squared Error (ISE) of 2.4340. Stability was attained in the end with a settling time of 40 seconds. A steady-state error of 0.02 was also attained.

4.1.2 Trial & Error Method with Induced Disturbance

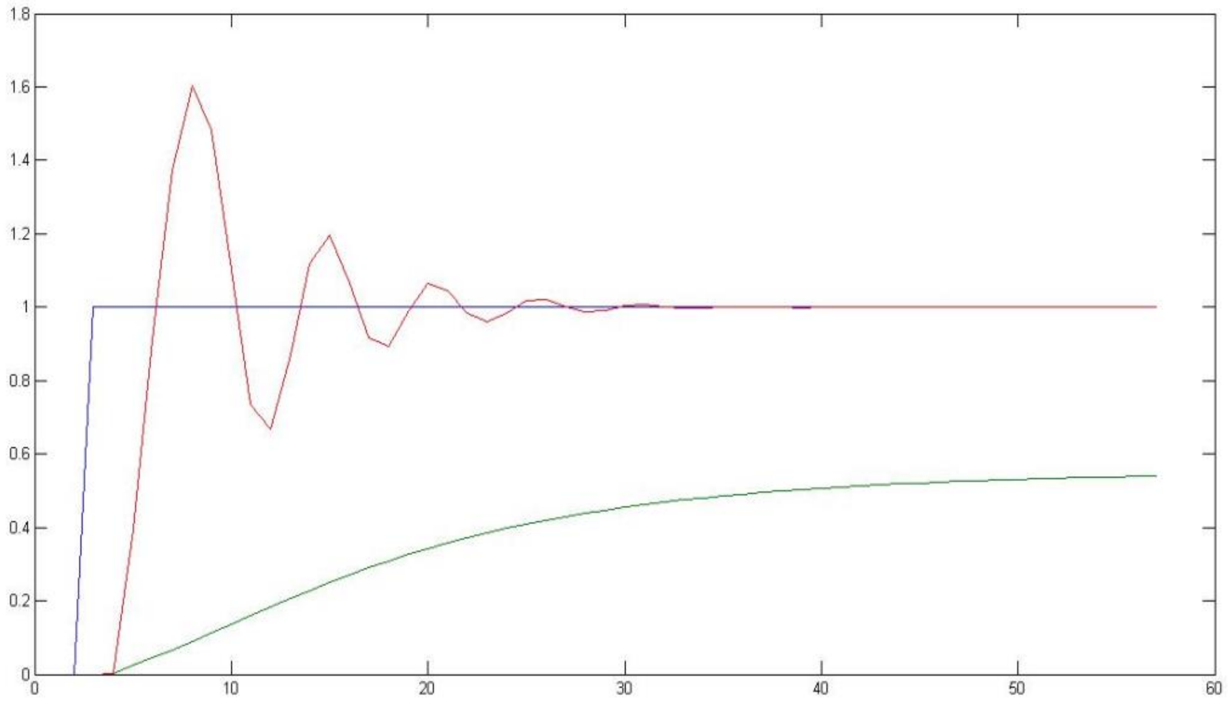


Figure 4.2: Response of Trial & Error Method with Induced Disturbance

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
Trial & Error	2.4262	60%	5s	42s	0.05

Table 4.2 Results for Trial & Error Method with Induced Disturbance

The trial and error method with induced disturbance resulted in an overshoot response of 60% with an Integrated Squared Error (ISE) of 2.4262. Stability was attained after 42seconds. A steady-state error of 0.05 was recorded.

4.2 IMC Method

The results obtained from simulating the Three-Phase Separator Level Controller using IMC method is discussed in Part 4.2.1 and 4.2.2.

4.2.1 IMC without Induced Disturbance

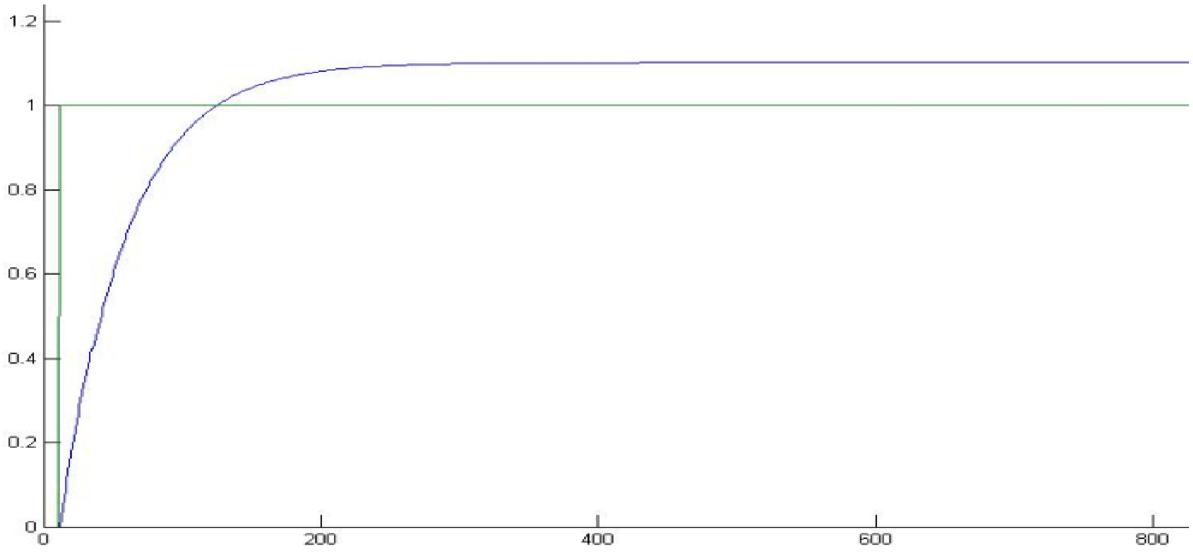


Figure 4.3: Response of IMC method

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
IMC	2.8183	0%	12s	70s	0.1

Table 4.3 Results for IMC method

The Internal Model Control method with no induced disturbance resulted in an Integrated Squared Error (ISE) of 2.8183. No overshoot response was recorded. Stability was attained after 70seconds.A steady-state error of 0.1 was recorded.

4.2.2 IMC Method with Induced Disturbance

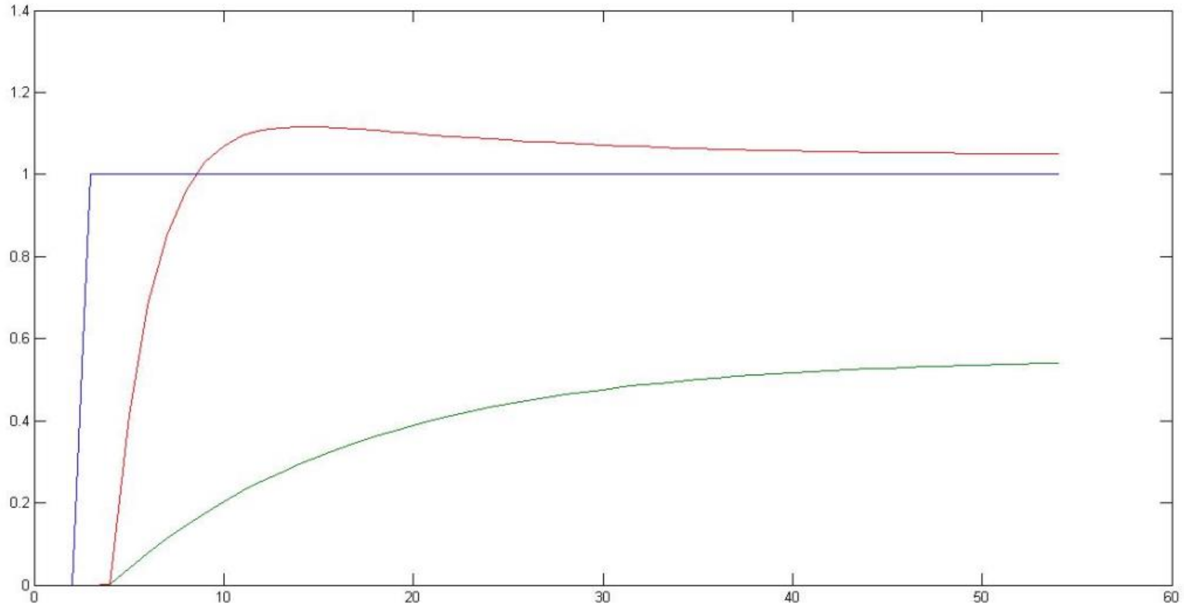


Figure 4.4: Response of IMC method with Induced Disturbance

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
IMC	2.3028	0%	12s	70s	0.1

Table 4.4 Results for IMC method with Induced Disturbance

The Internal Model Control method with induced disturbance resulted in an Integrated Squared Error (ISE) of 2.3028. No overshoot response was recorded. Stability was attained after 70seconds.A steady-state error of 0.1 was recorded.

4.3 Butterworth Filter Design Method

The results obtained from simulating the Three-Phase Separator Level Controller using BFD method is discussed in Part 4.3.1 and 4.3.2.

4.3.1 Butterworth Filter Design Method without Induced Disturbance

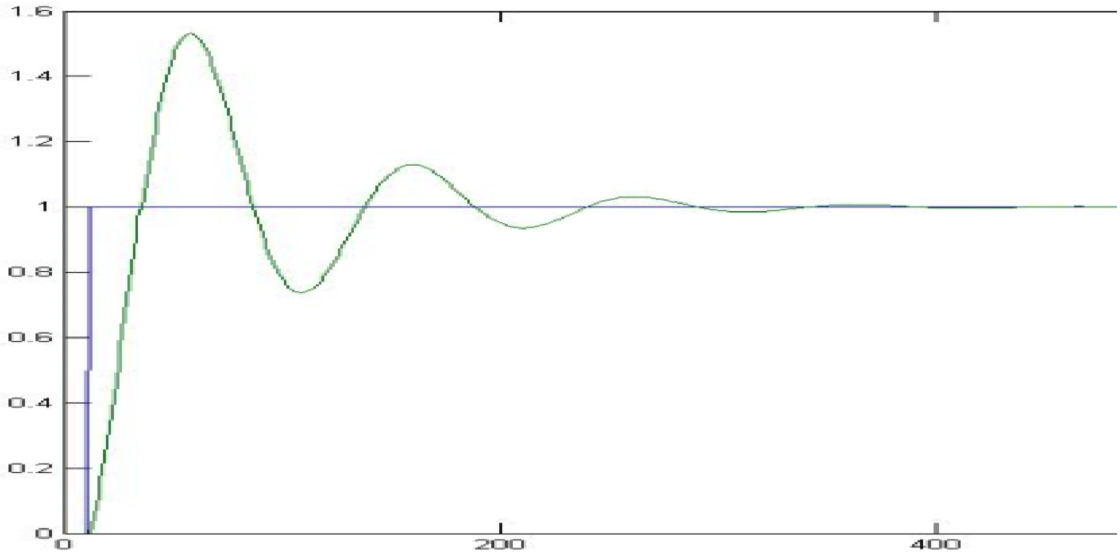


Figure 4.5: Response of BFD method

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
BFD	1.8116	55%	5s	40s	0.05

Table 4.5 Results for BFD method

The Butterworth Filter Design method with no induced disturbance resulted in an Integrated Squared Error (ISE) of 1.8116. An overshoot response of 55% was recorded. Stability was attained after 40seconds and a steady-state error of 0.05 was recorded.

4.3.2 BFD Method with Induced Disturbance

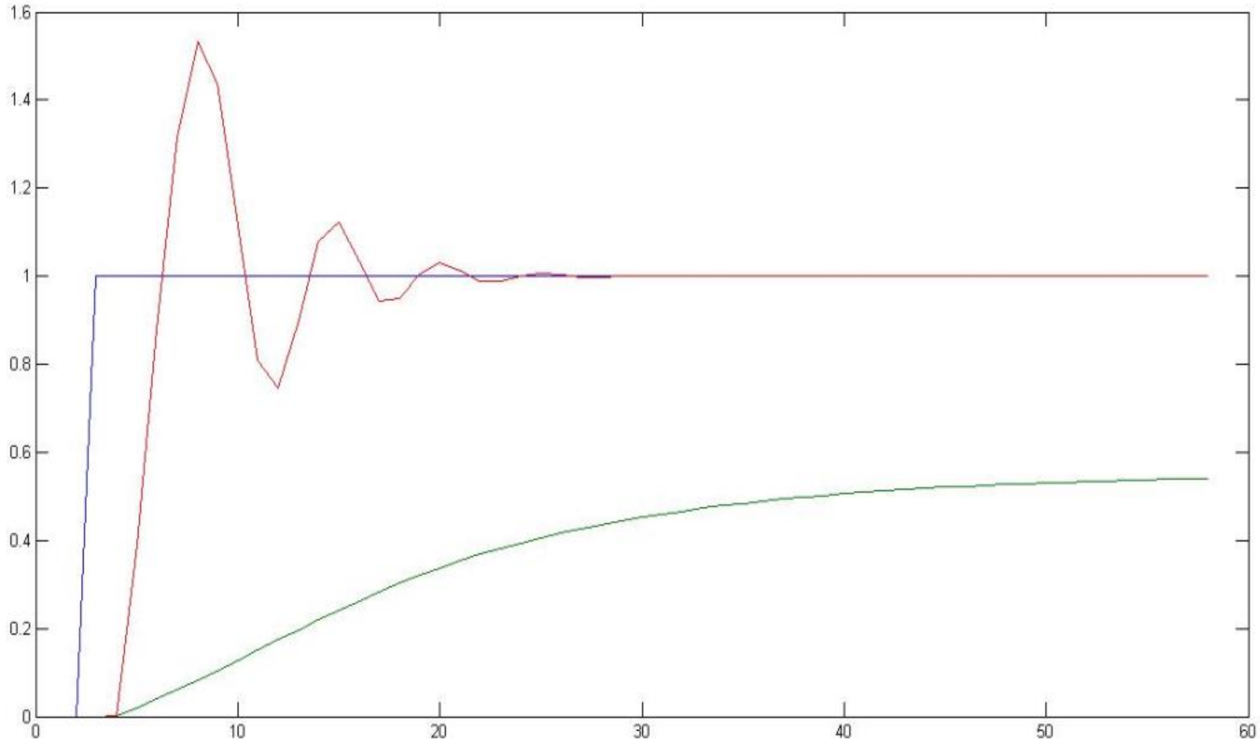


Figure 4.6: Response of BFD method with Induced Disturbance

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
BFD	1.8064	55%	5s	40s	0.05

Table 4.6 Results for BFD method with Induced Disturbance

The Butterworth Filter Design method with induced disturbance resulted in an Integrated Squared Error (ISE) of 1.8064. An overshoot response of 55% was recorded. Stability was attained after 40seconds and a steady-state error of 0.05 was recorded.

4.4 PSO Method

The results obtained from simulating the Three-Phase Separator Level Controller using PSO method is discussed in Part

4.4.1 PSO Method without Induced Disturbance (P-Controller)

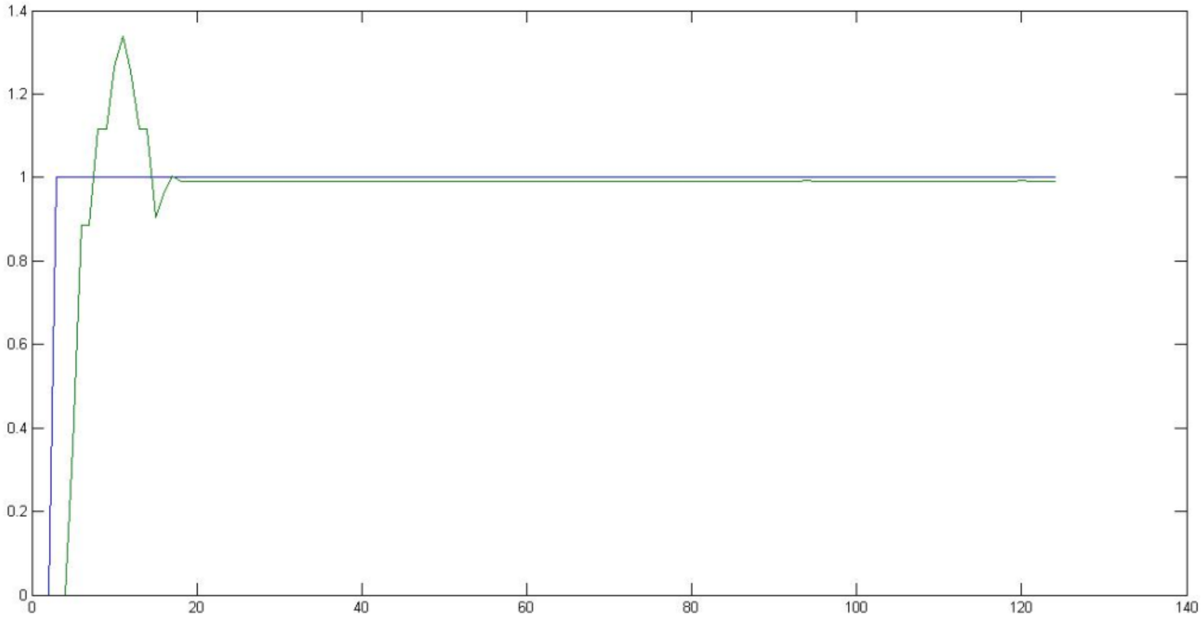


Figure 4.7: Response of PSO method (P-Controller)

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
PSO	2.0795	35%	5s	12s	0.05

Table 4.7 Results for PSO method (P-Controller)

The PSO method tuned with a P-Controller resulted in an Integrated Squared Error (ISE) of 2.0795. An overshoot response of 35% was recorded. Stability was attained after 12seconds and a steady-state error of 0.05 was recorded.

4.4.2 PSO Method with Induced Disturbance (P-Controller)

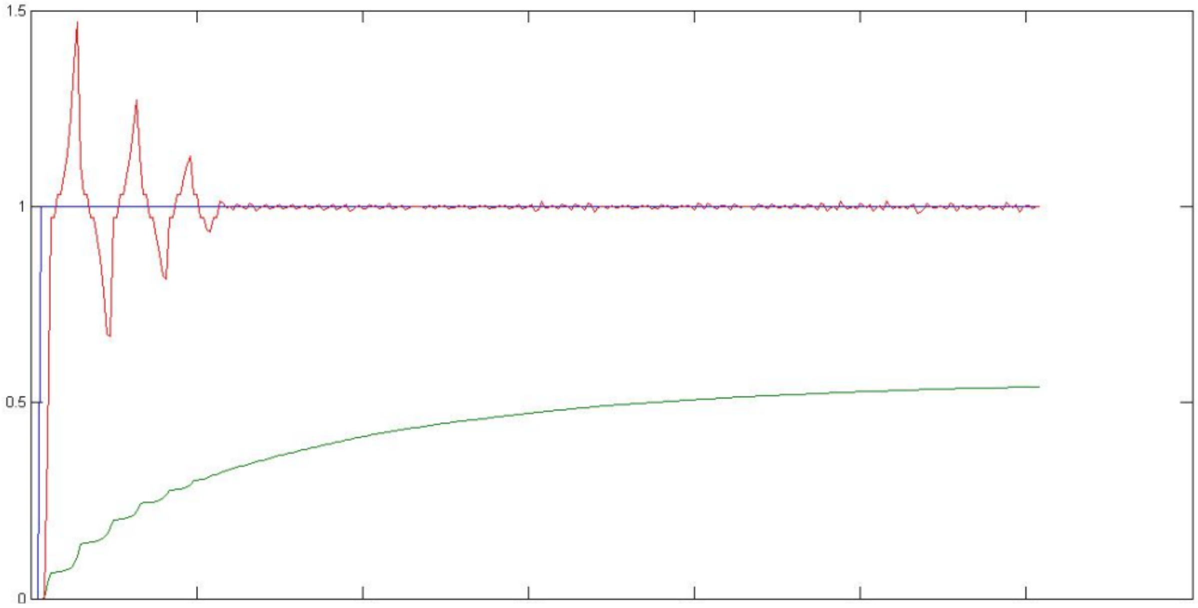


Figure 4.8: Response of PSO method with Induced Disturbance (P-Controller)

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
PSO	2.3185	40%	5s	25s	0.05

Table 4.8 Results for PSO Method with Induced Disturbance (P-Controller)

The PSO method tuned with a P-Controller with an induced disturbance resulted in an Integrated Squared Error (ISE) of 2.3185. An overshoot response of 40% was recorded. Stability was attained after 25seconds and a steady-state error of 0.05 was recorded.

4.4.3 PSO Method without Induced Disturbance (PI-Controller)

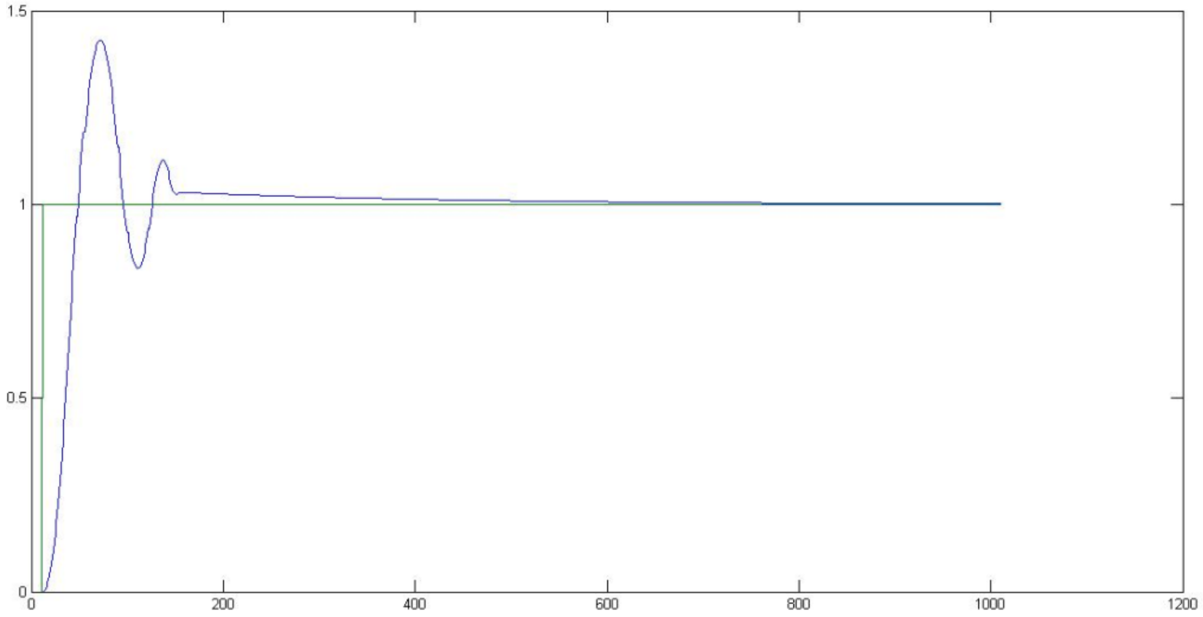


Figure 4.9: Response of PSO method (PI-Controller)

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
PSO	2.2809	53%	8s	45s	0.05

Table 4.9 Results for PSO Method (PI-Controller)

The PSO method tuned with a PI-Controller resulted in an Integrated Squared Error (ISE) of 2.2809. An overshoot response of 53% was recorded. Stability was attained after 45seconds and a steady-state error of 0.05 was recorded.

4.4.4 PSO Method with Induced Disturbance (PI-Controller)

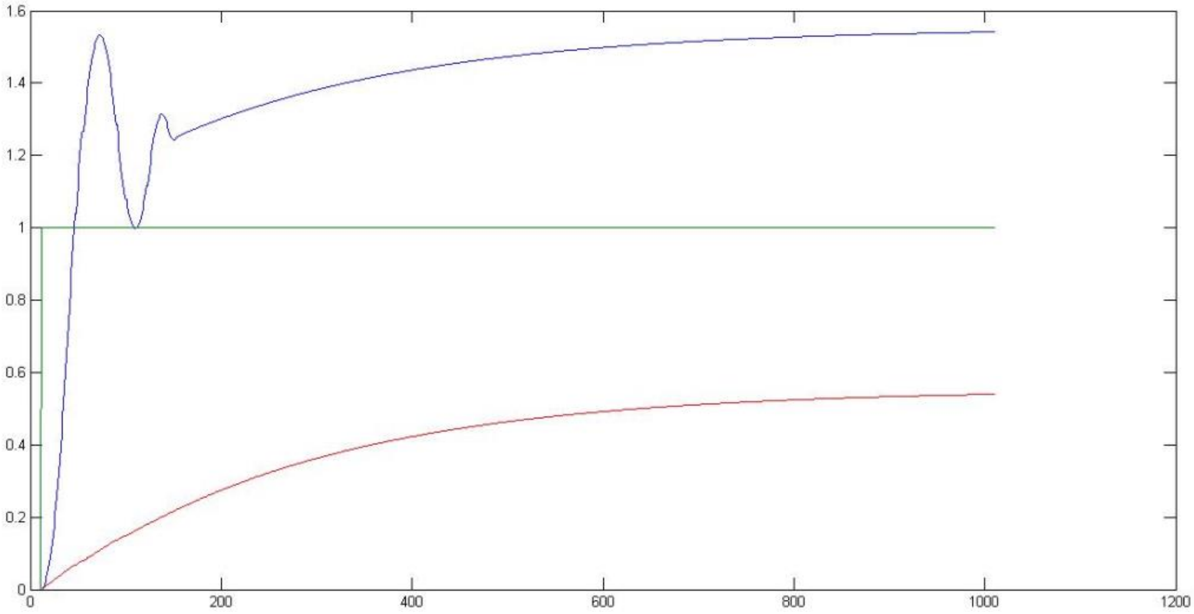


Figure 4.10: Response of PSO method with Induced Disturbance (PI-Controller)

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
PSO	3.2910	53%	7s	∞	∞

Table 4.10 Results for PSO Method with Induced Disturbance (PI-Controller)

The PSO method tuned with a PI-Controller with an induced disturbance resulted in an Integrated Squared Error (ISE) of 3.2910. An overshoot response of 53% was recorded. Stability was not attained and an infinite steady-state error was observed.

4.4.5 PSO Method without Induced Disturbance (PID-Controller)

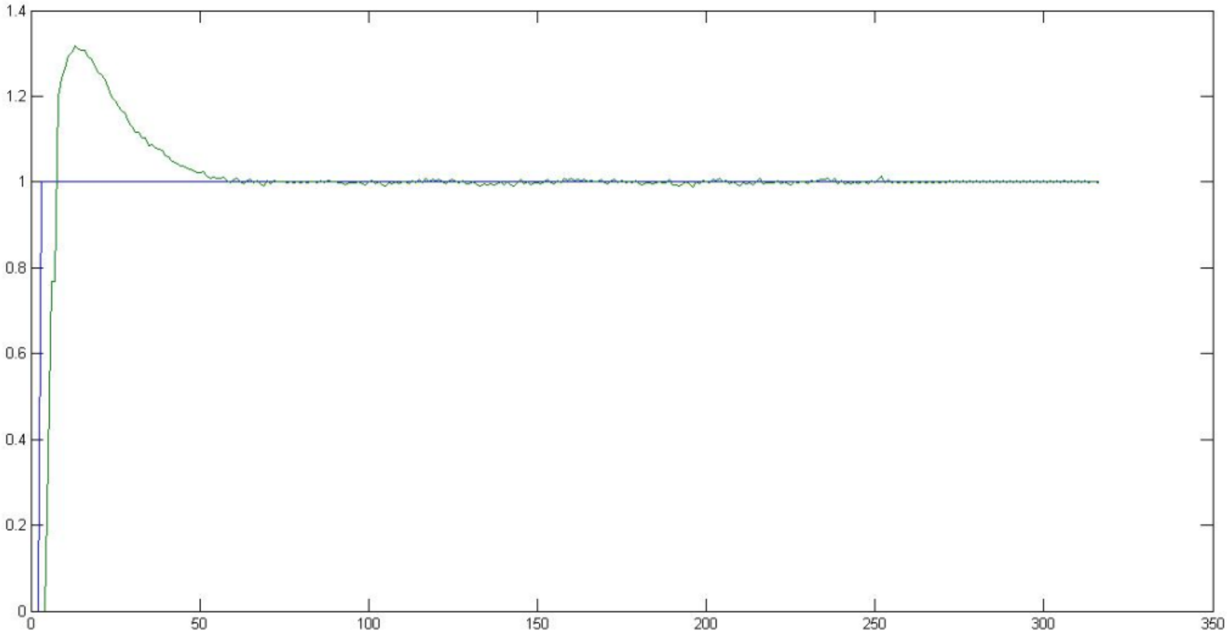


Figure 4.11: Response of PSO method (PID-Controller)

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
PSO	2.6536	25%	12s	35s	0.05

Table 4.11 Results for PSO Method (PID-Controller)

The PSO method tuned with a PID-Controller resulted in an Integrated Squared Error (ISE) of 2.6536. An overshoot response of 25% was recorded. Stability was attained after 35 seconds and a steady-state error of 0.05 was recorded.

4.4.6 PSO Method with Induced Disturbance (PID-Controller)

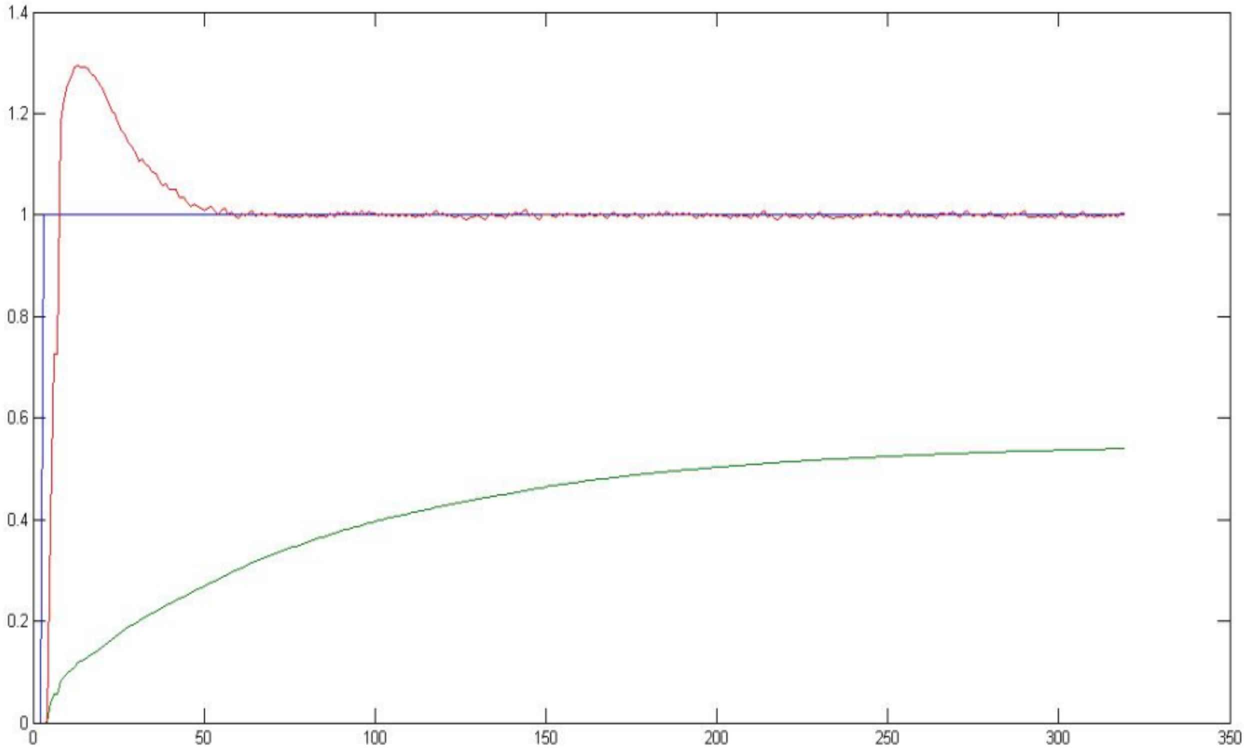


Figure 4.12: Response of PSO method with Induced Disturbance (PID-Controller)

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
PSO	2.8315	25%	12s	40s	0.10

Table 4.12 Results for PSO Method with Induced Disturbance (PID-Controller)

The PSO method tuned with a PID-Controller with an induced disturbance resulted in an Integrated Squared Error (ISE) of 2.8315. An overshoot response of 25% was recorded. Stability was attained after 40 seconds and a steady-state error of 0.05 was recorded.

4.4.7 PSO Method without Induced Disturbance (PD-Controller)

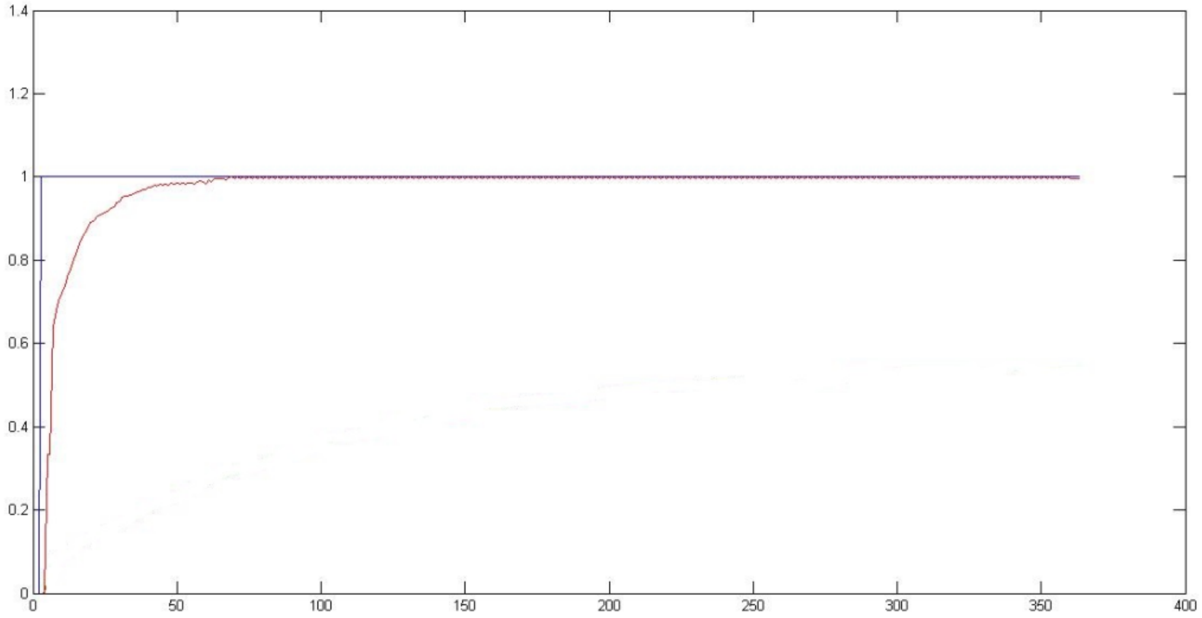


Figure 4.13: Response of PSO method (PD-Controller)

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
PSO	1.7695	0%	5s	12s	0

Table 4.13 Results for PSO Method (PD-Controller)

The PSO method tuned with a PD-Controller resulted in an Integrated Squared Error (ISE) of 1.7695. No overshoot response was recorded. Stability was attained after 12seconds and no steady-state error was recorded. The PD-Controller proved to be the best level controller used for tuning.

4.4.8 PSO Method with Induced Disturbance (PD-Controller)

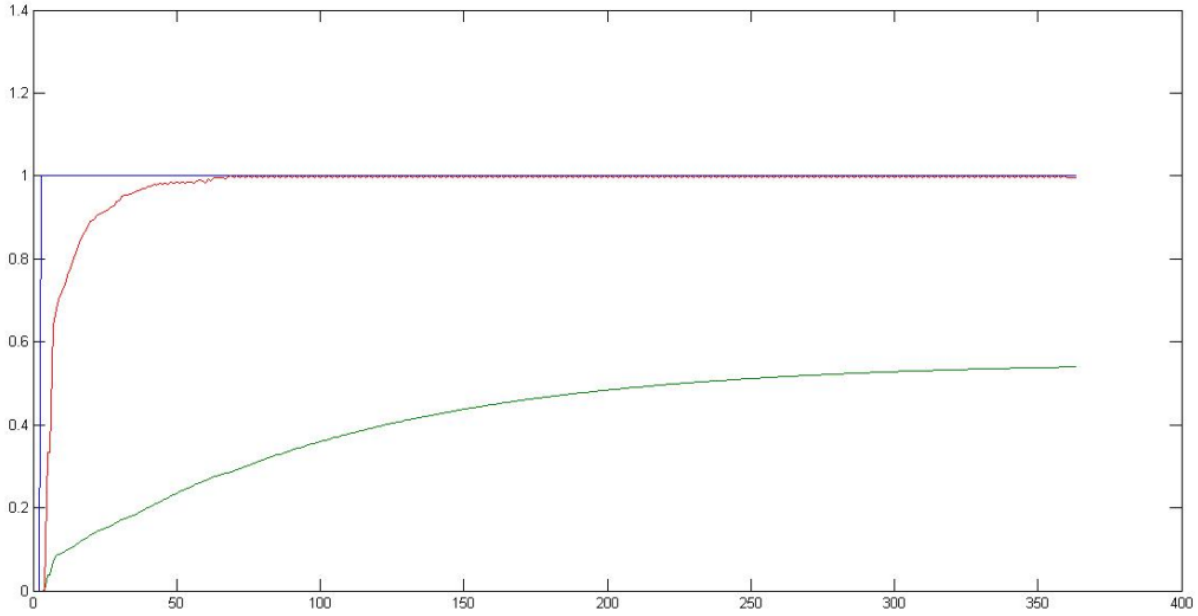


Figure 4.14: Response of PSO method with Induced Disturbance (PD-Controller)

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
PSO	1.9028	0%	5s	15s	0

Table 4.14 Results for PSO Method with Induced Disturbance (PD-Controller)

The PSO method tuned with a PD-Controller with an induced disturbance resulted in an Integrated Squared Error (ISE) of 1.9028. No overshoot response was recorded. Stability was attained after 15seconds and no steady-state error was recorded. The PD-Controller proved to be the best level controller used for tuning with an induced disturbance.

4.5 PSO Method (PD-Controller) with varying number of iterations

The number of bird steps, n (stopping criteria) for the PSO-PD Controller was varied in the region of ±20% from its default value of 50 to see the effect on the system response.

4.5.1 PSO Method without induced disturbance (PD-Controller)

n	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
40	1.7538	0%	5s	8s	0
45	1.7004	0%	5s	8s	0
50	1.7695	0%	5s	12s	0
55	1.9110	0%	5s	15s	0
60	1.7510	0%	5s	10s	0

Table 4.15 Results for PSO Method for variable iterations

4.5.2 PSO Method with induced disturbance (PD-Controller)

n	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
40	1.7391	0%	5s	8s	0
45	1.8466	0%	5s	12s	0
50	1.9028	0%	5s	15s	0
55	1.8470	0%	5s	12s	0
60	1.8364	0%	5s	10s	0

Table 4.16 Results for PSO Method with Induced Disturbance for variable iterations

The results prove that the best stopping criteria for the simulation would be at n=40 bird steps. It provides no overshoot response and a constant settling time of 8s with and without an induced disturbance. An integral squared error of 1.7391 and 1.7538 was produced with and without an induced disturbance respectively. The system also produces a rise time of 5s.

4.6 Comparative Analysis

Figure 4.15 and figure 4.16 shows the integrated response of each tuning method, without and with an induced disturbance respectively.

Table 4.17 and table 4.18 shows the results of the integrated response for each tuning method, without and with an induced disturbance respectively.

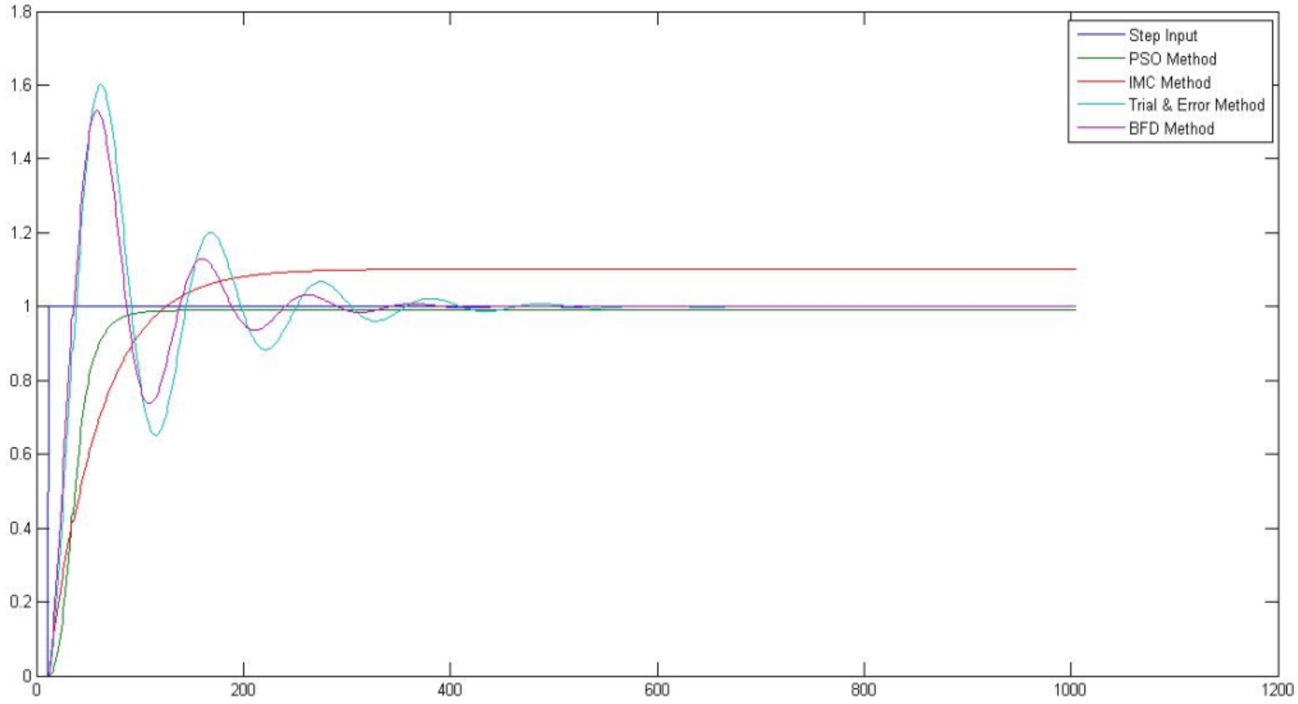


Figure 4.15: Integrated Plot

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
Trial & Error	2.4340	60%	5s	40s	0.02
BFD	1.8116	55%	5s	40s	0.05
IMC	2.8183	0%	12s	70s	0.10
PSO	1.7538	0%	5s	8s	0

Table 4.17 Comparative analysis of results

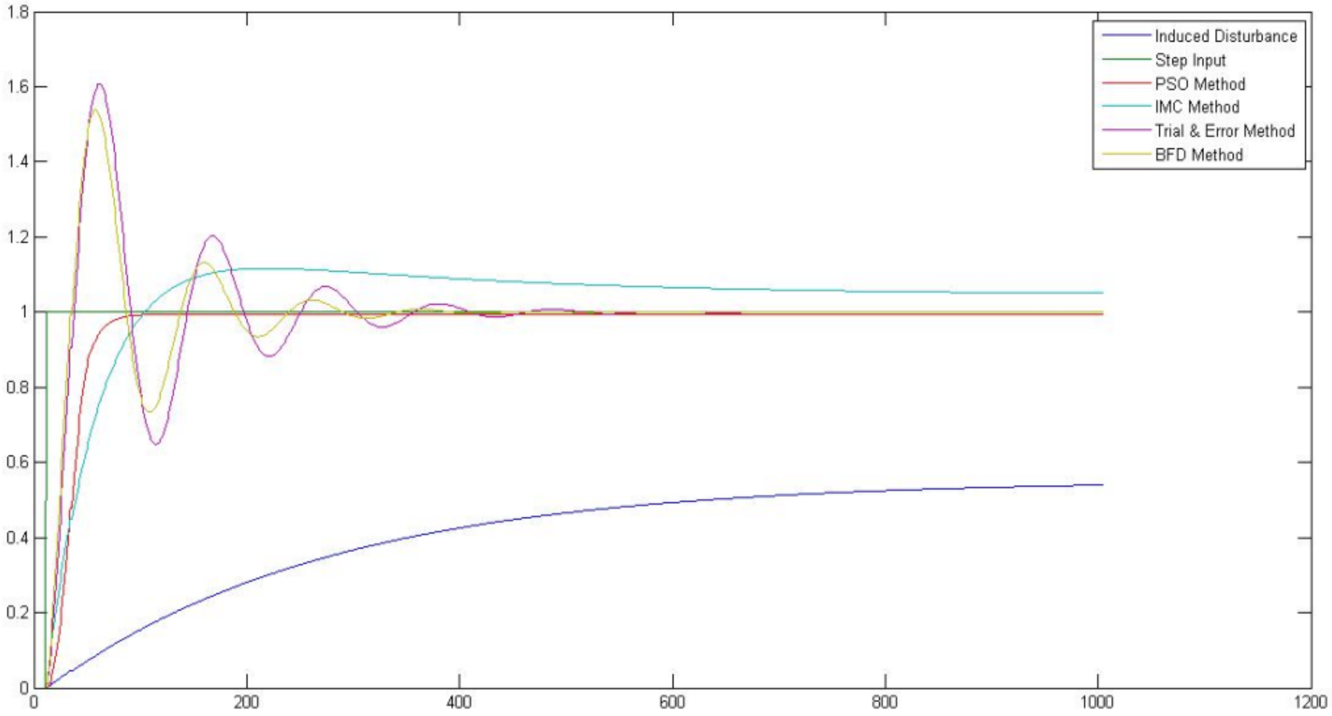


Figure 4.16: Integrated Plot with Induced Disturbance

Method	ISE	Overshoot	Rise Time(Tr)	Settling time(Ts)	Steady-state error
Trial & Error	2.4262	60%	5s	42s	0.05
BFD	1.8064	55%	5s	40s	0.05
IMC	2.3028	0%	12s	70s	0.10
PSO	1.7391	0%	5s	8s	0

Table 4.18 Comparative analysis of results with Induced Disturbance

Table 4.19 and table 4.20 shows the percentage of improvement the PSO tuning method has over the current tuning methods, without and with an induced disturbance respectively.

Method	ISE Improvement	Overshoot Improvement	Rise Time(Tr) Improvement	Settling time(Ts) Improvement	Average Improvement
PSO	-	-	-	-	-
Trial & Error	39%	60%	0%	400%	125%
BFD	3%	55%	0%	400%	115%
IMC	61%	0%	140%	775%	244%

Table 4.19 Table of PSO Improvement

Method	ISE Improvement	Overshoot Improvement	Rise Time(Tr) Improvement	Settling time(Ts) Improvement	Average Improvement
PSO	-	-	-	-	-
Trial & Error	40%	60%	0%	425%	131%
BFD	4%	55%	0%	400%	115%
IMC	32%	0%	140%	775%	237%

Table 4.20 Table of PSO Improvement with Induced Disturbance

4.7 Implementation of the PSO Control Algorithm in GUI

The methods of simulating the Three-Phase Separator Level Controller using PSO are shown using the Graphical User Interface (GUI) below.

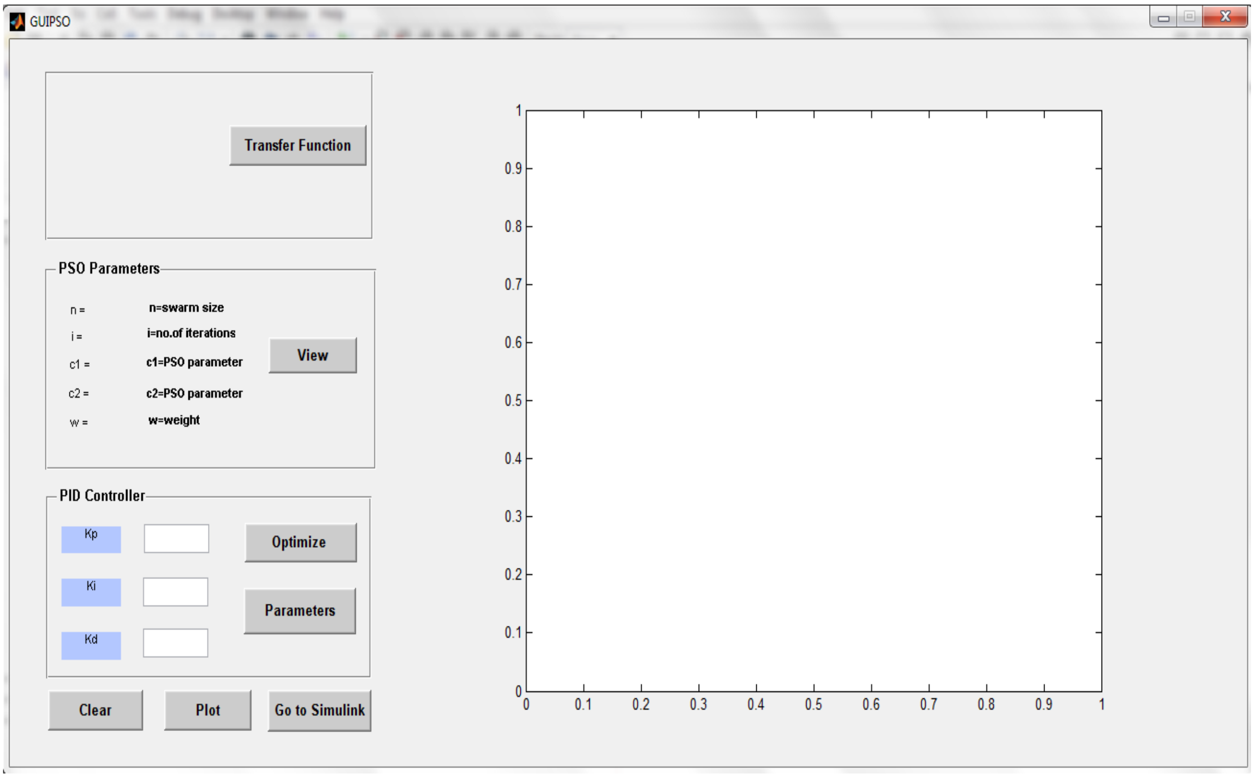
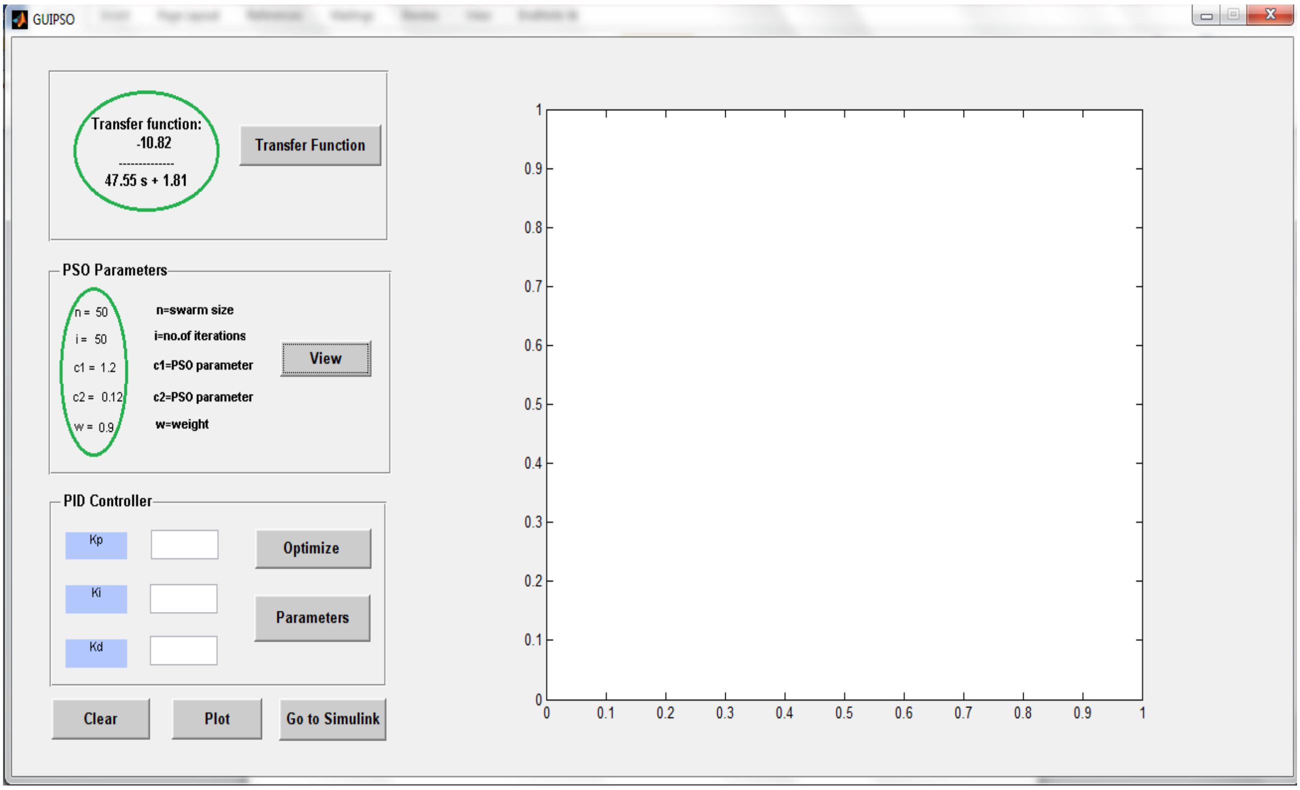


Figure 4.17 Default GUI display

The default GUI window when the GUI.m file is run is shown in Figure 4.17. There are three panels for the interface. The first panel shows the generated plant transfer function, the second panel displays the PSO parameters used in the simulation, and the third panel displays the tuning parameters Kp, Ki, and Kd of the level controller.



Transfer function:
-10.82

47.55 s + 1.81

PSO Parameters

n = 50
i = 50
c1 = 1.2
c2 = 0.12
w = 0.9

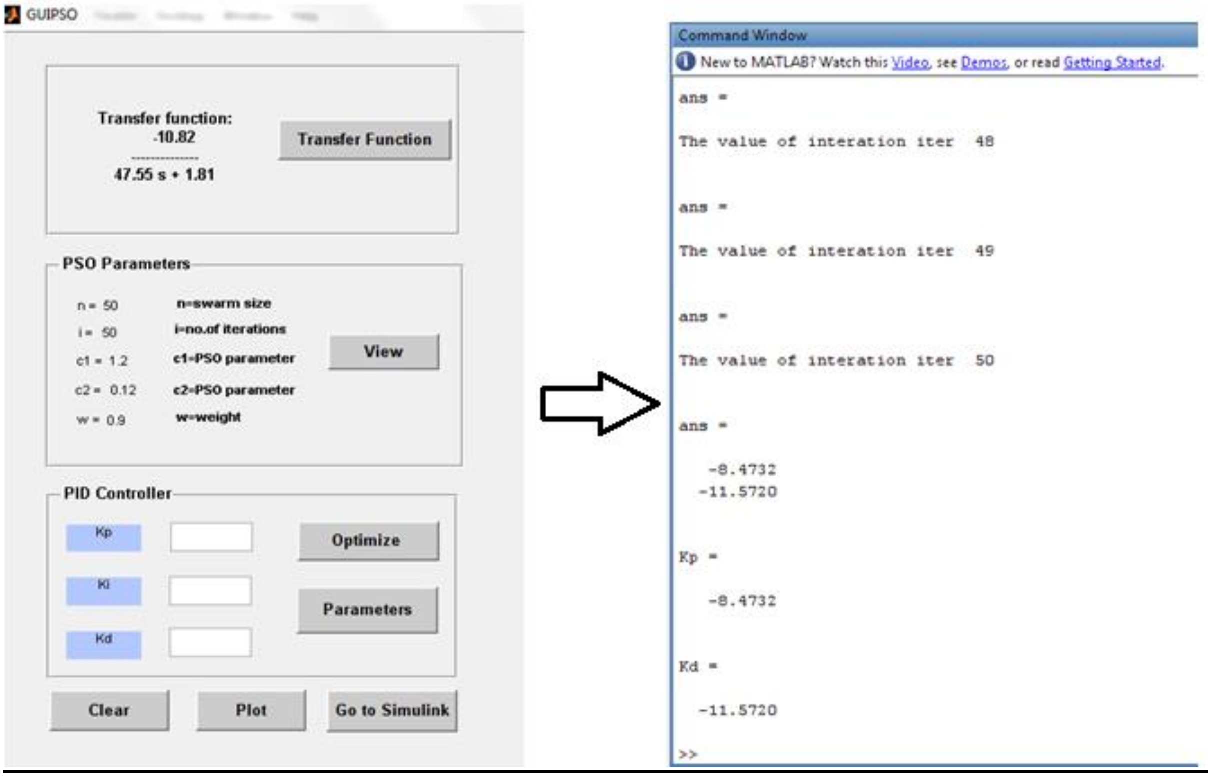
n=swarm size
i=no.of iterations
c1=PSO parameter
c2=PSO parameter
w=weight

PID Controller

Kp
Ki
Kd

Figure 4.18 Displaying Transfer Function and PSO parameters

In order to display the Transfer Function of the process, the *Transfer Function* push button is clicked. The plant's transfer function will then be displayed on the left of the push button. The next step would be to display the PSO default parameters. The *view* push button is now clicked and the parameters are then displayed as shown in the figure.



Transfer function:
-10.82

47.55 s + 1.81

PSO Parameters

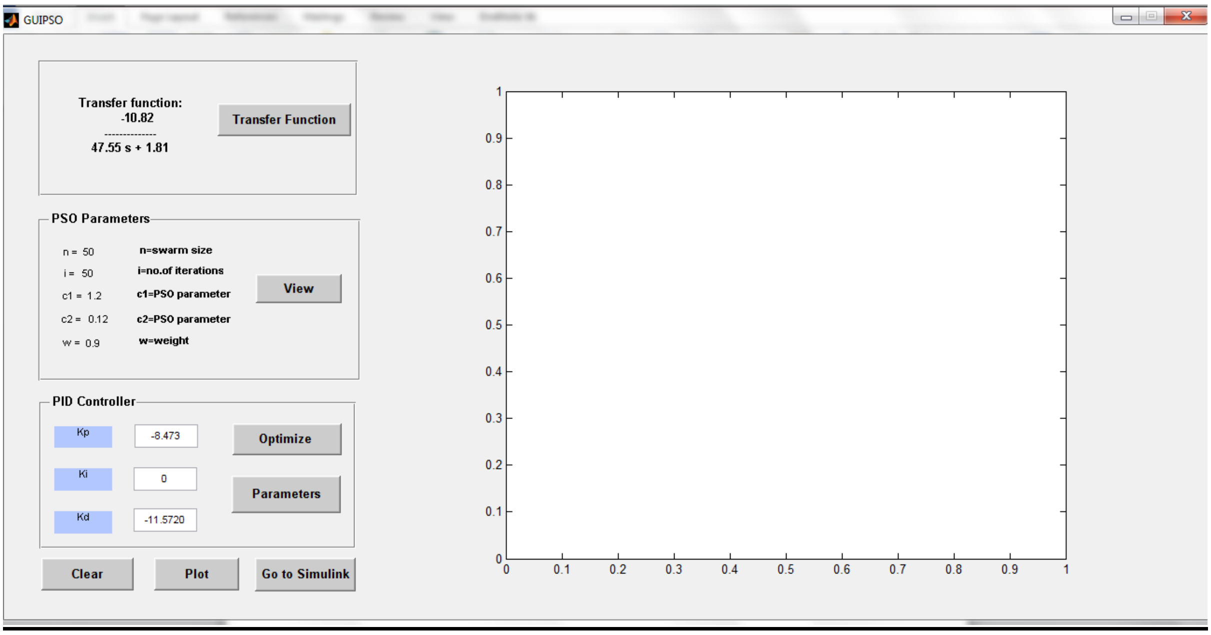
n= 50 n=swarm size
i= 50 i=no.of iterations
c1 = 1.2 c1=PSO parameter
c2= 0.12 c2=PSO parameter
w= 0.9 w=weight

PID Controller

```
ans =

The value of interation iter  48


ans =

The value of interation iter  49


ans =

The value of interation iter  50


ans =

  -8.4732
 -11.5720


Kp =

  -8.4732


Kd =

 -11.5720

>>
```

Figure 4.19 Running the PSO algorithm

After generating the transfer function and displaying the PSO parameters, the next step would be to tune the plant via PSO method. The *Optimize* push button is clicked and the tuning parameters Kp, Ki and Kd are computed as shown in the figure.



Transfer function:
-10.82

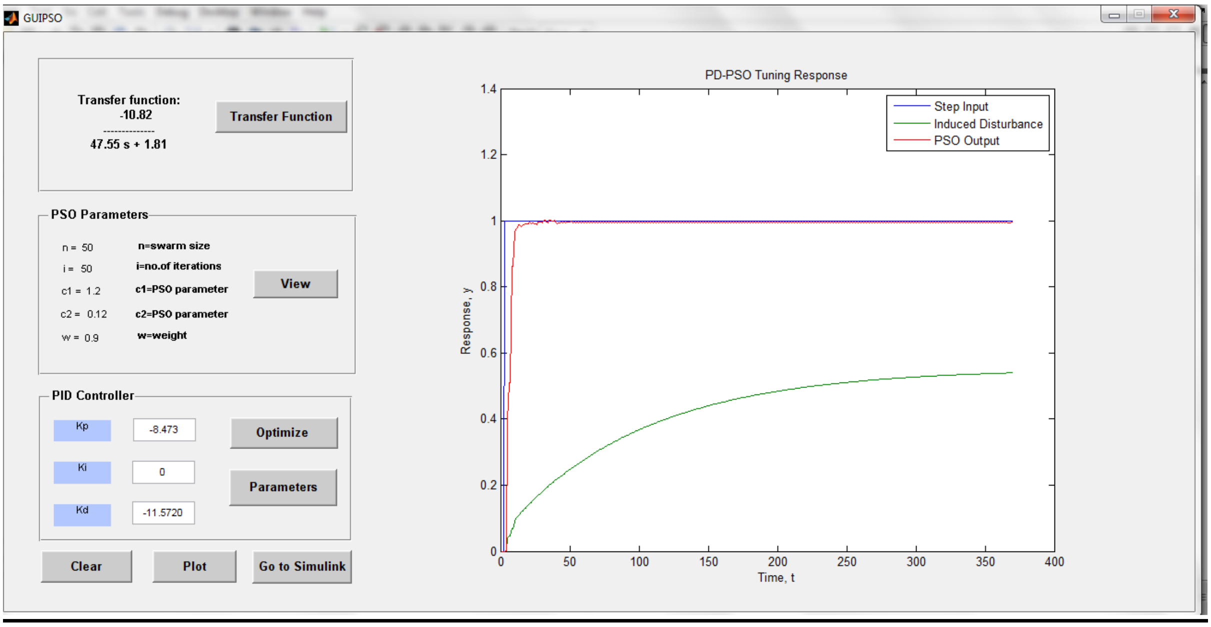
47.55 s + 1.81

n = 50 n=swarm size
i = 50 i=no.of iterations
c1 = 1.2 c1=PSO parameter
c2 = 0.12 c2=PSO parameter
w = 0.9 w=weight

Kp -8.473
Ki 0
Kd -11.5720

Figure 4.20 Displaying tuning parameters in GUI

In order to display the tuning parameters Kp, Ki, and Kd obtained from the common window into GUI, the *Parameters* push button is then clicked. The values are the displayed as shown in the figure above.


Transfer function:
-10.82

47.55 s + 1.81

PSO Parameters

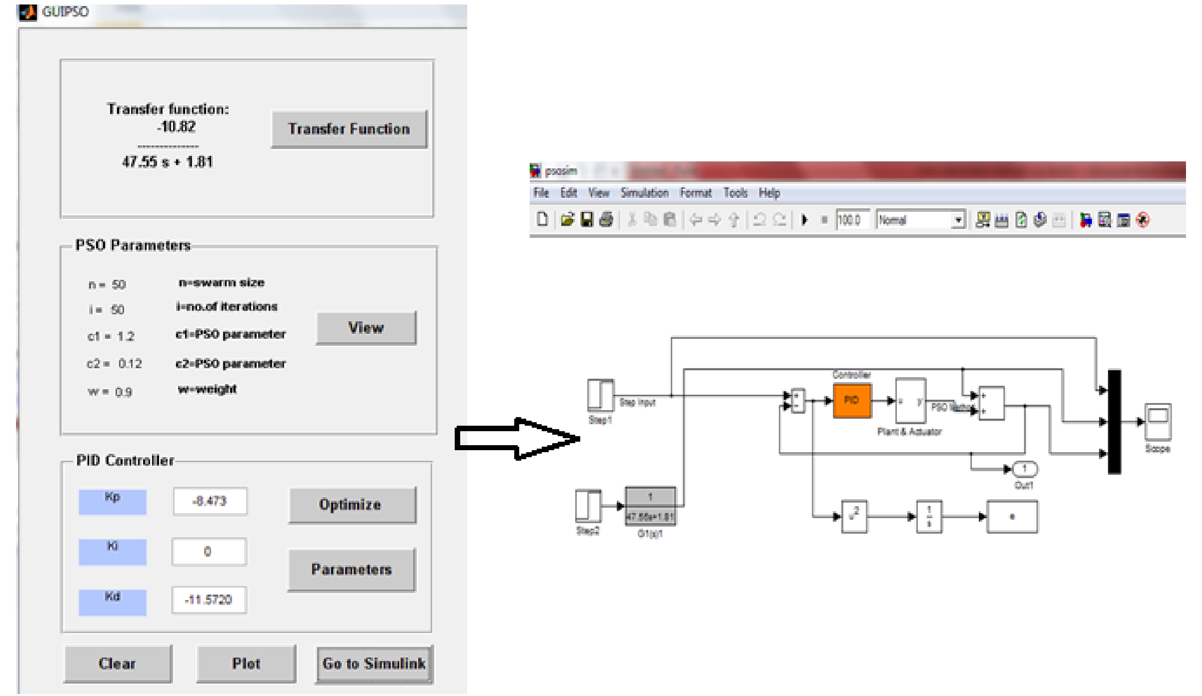
n = 50 n=swarm size
i = 50 i=no.of iterations
c1 = 1.2 c1=PSO parameter
c2 = 0.12 c2=PSO parameter
w = 0.9 w=weight

PID Controller

Kp -8.473
Ki 0
Kd -11.5720

Figure 4.21 Plotting response in GUI

Once the tuning parameters are displayed, a plot of the PSO-Three Phase Separator Level Controller simulation can be displayed in GUI. In order to display the plot, the *Plot* push button is clicked. The trending will be displayed on the axes as shown in the figure.


GUIPSO

Transfer function:
-10.82

47.55 s + 1.81

PSO Parameters

n = 50 n=swarm size
i = 50 i=no.of iterations
c1 = 1.2 c1=PSO parameter
c2 = 0.12 c2=PSO parameter
w = 0.9 w=weight

PID Controller

Kp -8.473
Ki 0
Kd -11.5720

Figure 4.22 Displaying the Simulink model from GUI

A push button called *Go to Simulink* enables users to directly view the designed Simulink model and make necessary changes if required. A *Clear* pushbutton is also present to enable users to return the GUI to the default state as shown in Figure 4.22

Chapter 5

Conclusion & Recommendation

Tuning the Three-Phase Separator Level Controller using Particle Swarm Optimization (PSO) proved to be an effective solution. The P-D Level Controller proved to be the best pair for tuning. It provided a system response with a reduced Integral Squared Error (ISE) and zero overshoot value. A faster process settling time was also attained and this helped in maintaining the water within a permissible level in the separator which met the control objective of the controller. This tuning method was also more effective than the other existing techniques modeled such as Internal Model Control, Trial & Error, and Butterworth Filter Design. A graphical user interface (GUI) was developed to run the PSO simulation of the Three-Phase Separator Level Controller.

Future recommendation would be for a test to be performed on a real time system using the simulated model. Besides that, other Artificial Intelligence (AI) techniques should be introduced to tune the Three-Phase Separator and any other suitable real time systems.

References

[1] M. Shahrokhi and A. Zomorrodi, "Comparison of PID Controller Tuning Methods," 2012.

[2] A. F. Sayda and J. H. Taylor, "Modeling and Control of Three-Phase Gravilty Separators in Oil Production Facilities," in *American Control Conference, 2007. ACC '07*, 2007, pp. 4847-4853.

[3] Y. Zhenyu, M. Juhl, and B. Lohndorf, "On the innovation of level control of an offshore three-phase separator," in *Mechatronics and Automation (ICMA), 2010 International Conference on*, 2010, pp. 1348-1353.

[4] R. Monroy-Loperena, R. Solar, and J. Alvarez-Ramirez, "Balanced control scheme for reactor/separator processes with material recycle," *Industrial & engineering chemistry research,* vol. 43, pp. 1853-1862, 2004.

[5] D. Muhammad, Z. Ahmad, and N. Aziz, "Implementation Of Internal Model Control (IMC) In Continuous Distillation Column," 2010.

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

[7] M. Nasri, H. Nezamabadi-Pour, and M. Maghfoori, "A PSO-based optimum design of PID controller for a linear brushless DC motor," *World Academy of Science, Engineering and Technology,* vol. 26, pp. 211-215, 2007.

[8] F. Hongqing, C. Long, and W. Wancheng, "Comparison of PSO algorithm and its variants to optimal parameters of PID controller," in *Control and Decision Conference, 2008. CCDC 2008. Chinese*, 2008, pp. 3100-3104.

[9] M. A. Rana, Z. Usman, and Z. Shareef, "Automatic control of ball and beam system using Particle Swarm Optimization," in *Computational Intelligence and Informatics (CINTI), 2011 IEEE 12th International Symposium on*, 2011, pp. 529-534.

[10] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J.-C. Hernandez, and R. G. Harley, "Particle swarm optimization: basic concepts, variants and applications in power systems," *Evolutionary Computation, IEEE Transactions on,* vol. 12, pp. 171-195, 2008.

APPENDIX A

PARTICIPATION IN SEDEX 31

31

SCIENCE&ENGINEERING
DESIGN EXHIBITION

UNIVERSITI
TEKNOLOGI
PETRONAS

Certificate of Participation

This Certificate is awarded to

LUVENRAJ SATHASIVAM
910220-05-5007

in recognition of his/her contribution as

Participant of
Open Innovation Challenge

"Bronze"

in the 31st edition of

Science and Engineering Design Exhibition
Universiti Teknologi PETRONAS

held on
21st-22nd August 2013

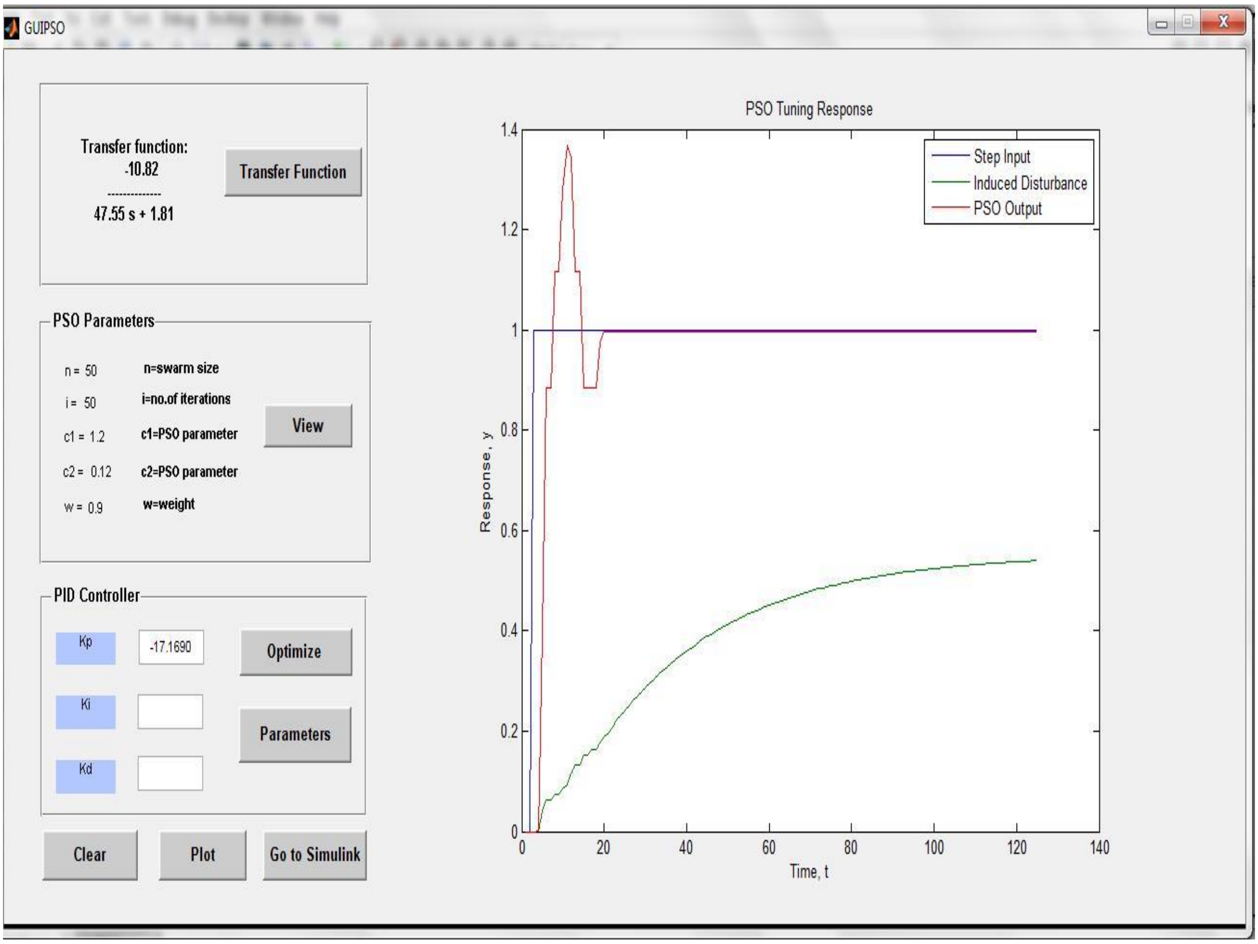
Dr Lukman Bin AB Rahim,
Chairman,
Science and Engineering
Design Exhibition 31

Prof.Ir.Dr.Ahmad Fadzil
Mohamad Hani,
FASc, FIEM
Deputy Vice Chancellor
(Academic)

SEDEX

APPENDIX B

Tuning Response of P-Controller PSO



Transfer function:
-10.82

47.55 s + 1.81

PSO Parameters

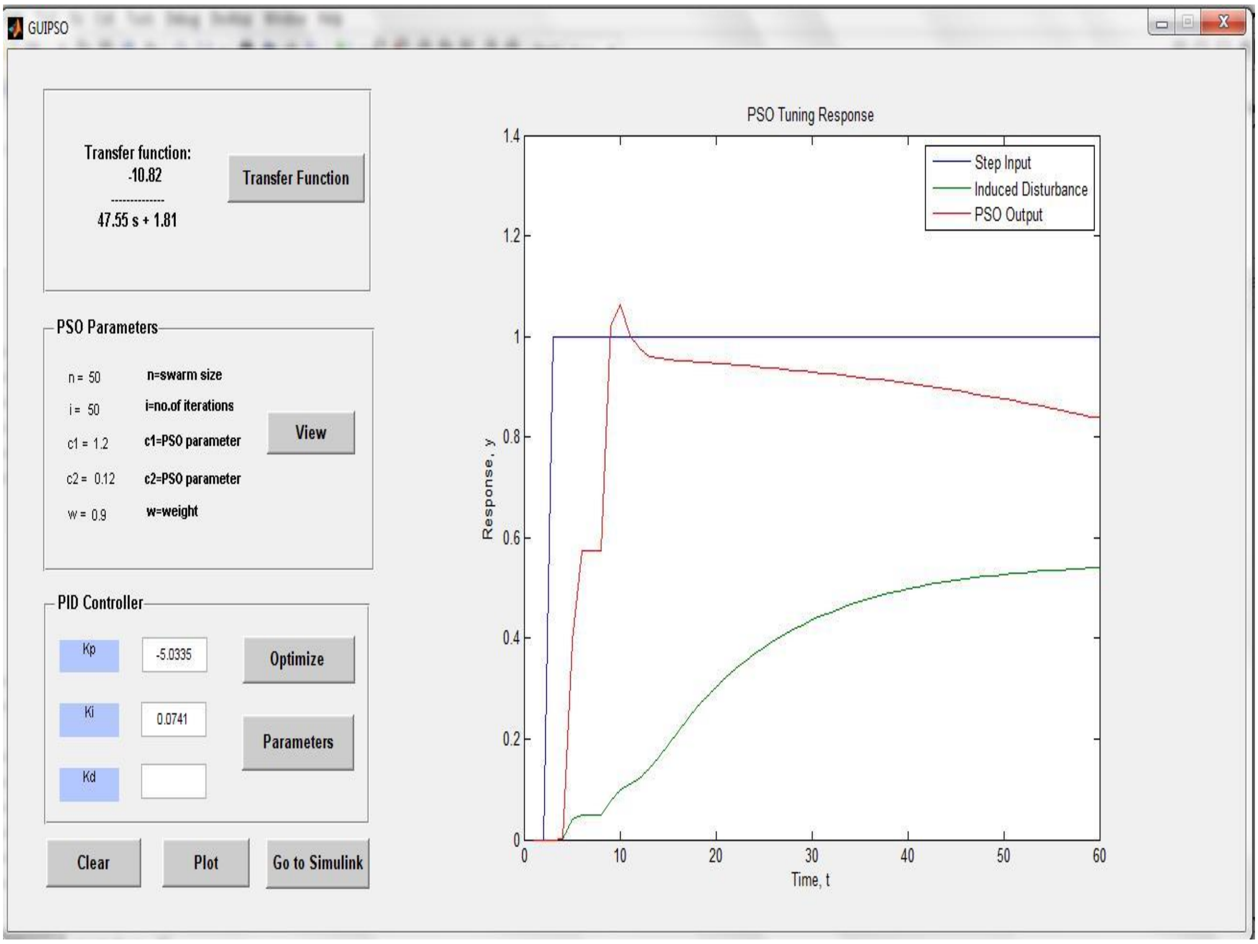
n = 50 n=swarm size
i = 50 i=no.of iterations
c1 = 1.2 c1=PSO parameter
c2 = 0.12 c2=PSO parameter
w = 0.9 w=weight

PID Controller

Kp -17.1690
Ki
Kd

APPENDIX C

Tuning Response of PI-Controller PSO



Transfer function:

-10.82

47.55 s + 1.81

PSO Parameters

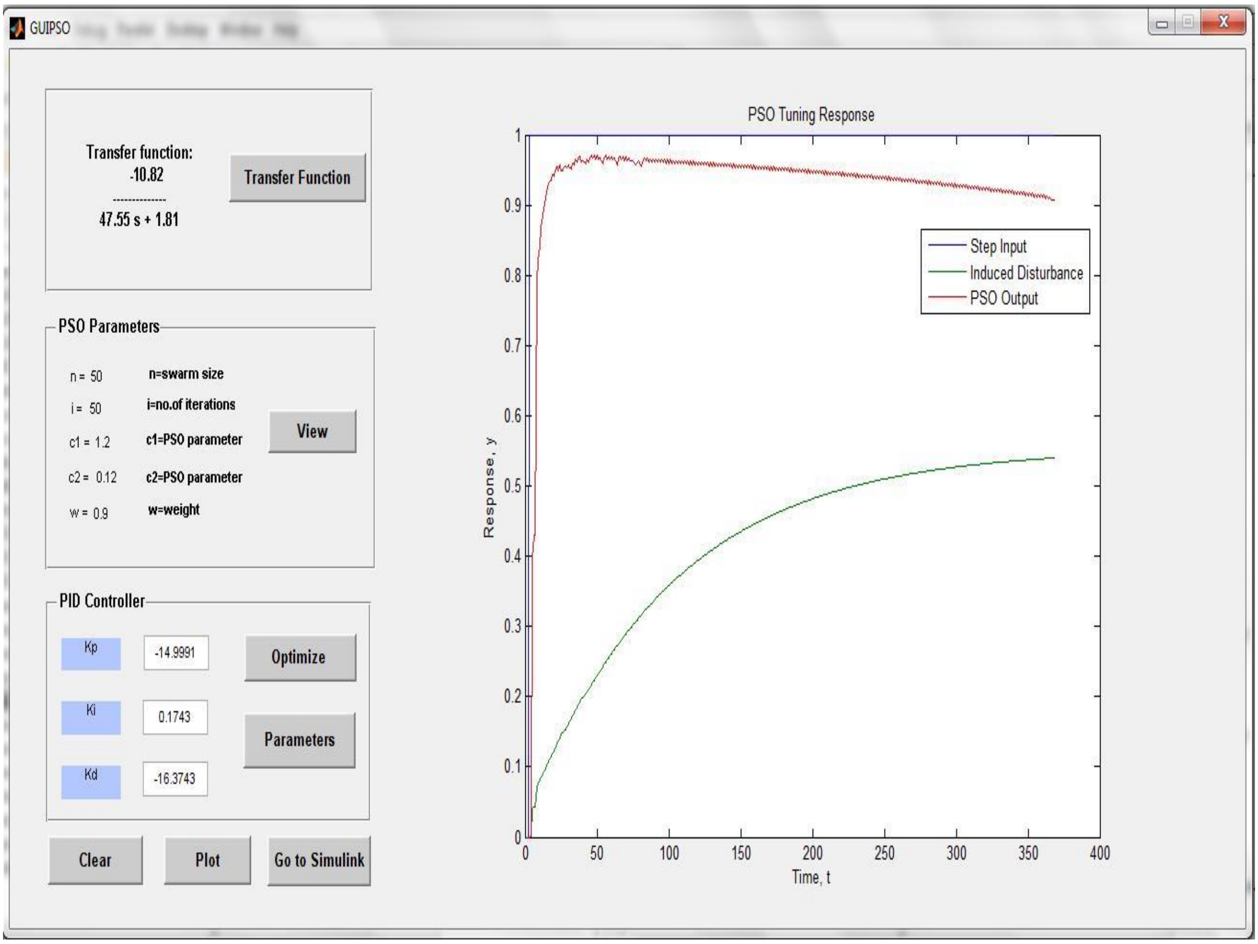
n = 50 n=swarm size
i = 50 i=no.of iterations
c1 = 1.2 c1=PSO parameter
c2 = 0.12 c2=PSO parameter
w = 0.9 w=weight

PID Controller

Kp	-5.0335
Ki	0.0741
Kd	

APPENDIX D

Tuning Response of PID-Controller PSO



Transfer function:

-10.82

47.55 s + 1.81

PSO Parameters

n = 50 n=swarm size

i = 50 i=no.of iterations

c1 = 1.2 c1=PSO parameter

c2 = 0.12 c2=PSO parameter

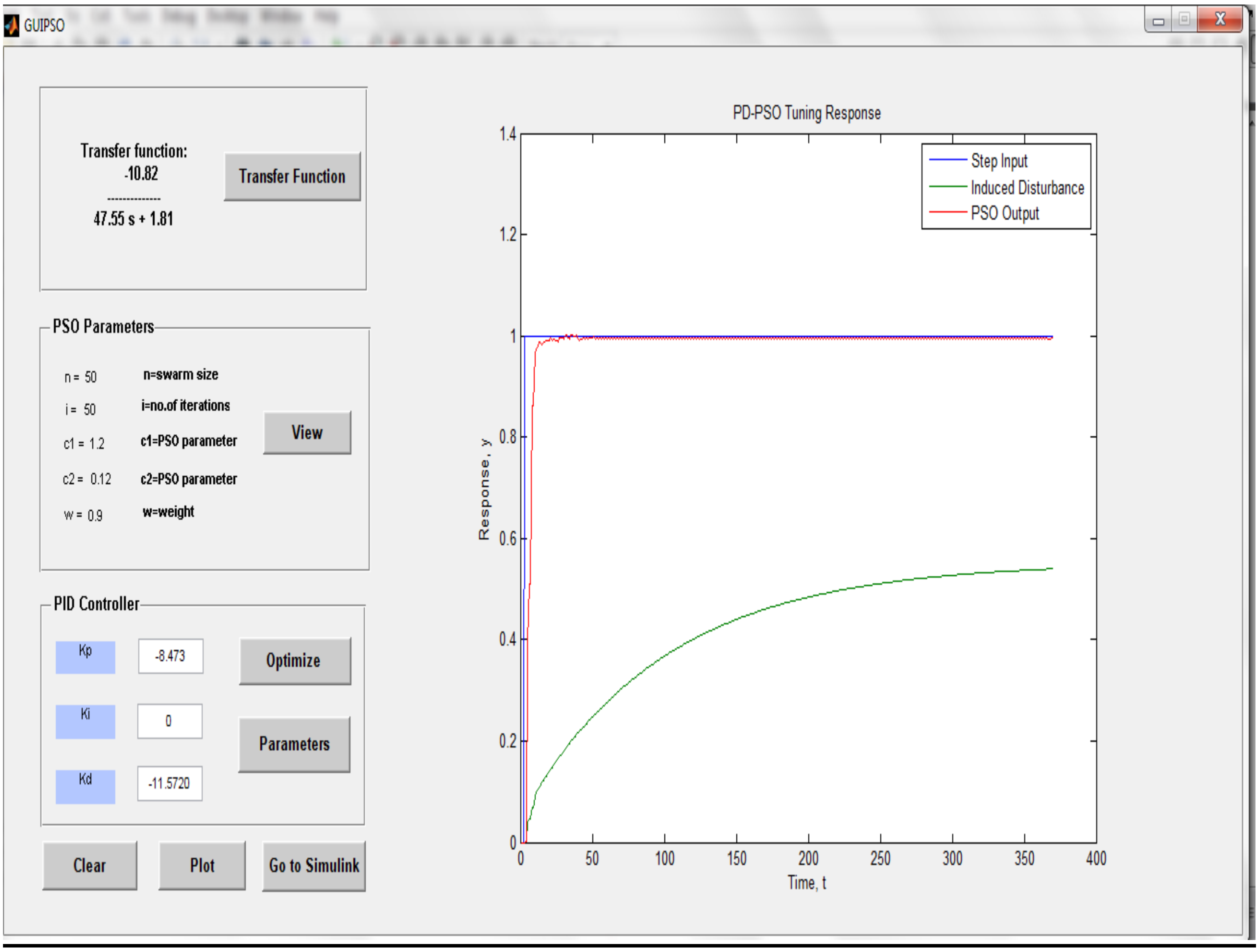
w = 0.9 w=weight

PID Controller

Kp	-14.9991
Ki	0.1743
Kd	-16.3743

APPENDIX E

Tuning Response of PD-Controller PSO



Transfer function:
-10.82

47.55 s + 1.81

PSO Parameters

n = 50 n=swarm size
i = 50 i=no.of iterations
c1 = 1.2 c1=PSO parameter
c2 = 0.12 c2=PSO parameter
w = 0.9 w=weight

PID Controller

Kp = -8.473
Ki = 0
Kd = -11.5720