ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

PID Controller Design Using Particle Swarm Optimization for Servo Actuation System of Reusable Launch Vehicle

Remya S[1], Priya C Kurian[2], Priyanka C P[3]

M.Tech Scholar, Dept. of EEE, Lourdes Matha College of Science and Technology, Thiruvananthapuram, Kerala, India[1]

Scientist/ Engineer, CECG, Vikram Sarabhai Space Centre, ISRO, Thiruvananthapuram, Kerala, India[2]

Assistant Professor, Dept. of EEE,Lourdes Matha College of Science and Technology, Thiruvananthapuram, Kerala, India[3]

ABSTRACT:Reusable launch vehicle is a system which has the ability to carry a payload from the earth's surface to the outer space more than once. The actuator forms the main loop of the control system and it actuates the control surfaces of the RLV based on the command signal. Electro hydraulic actuators are used in RLV for vectoring the control surfaces about their axes. In order to meet the requirements of the system and to obtain a stable performance, a Proportional Integral Derivative (PID) controller using Particle swarm optimization technique (PSO) is designed for the actuator system. PSO technique is used to design the optimal controller parameters of PID controller taking into account the time domain specifications. The optimization technique is widely used for controllers because of its high computational efficiency. The optimization technique eliminates the trial and error complexity in the conventional design technique of the actuation system.

KEYWORDS: Reusable Launch Vehicle, Electro hydraulic Actuator, Particle Swarm Optimization, PID Controller.

I.INTRODUCTION

The reusable launch vehicles are designed to be recovered and launched more than once. The actuator forms the main loop of the control system and it actuates the control surfaces of the reusable launch vehicle based on the command signals. The actuator system will deflect the control surfaces to bring about the necessary control action. The electro hydraulic linear actuators are used for vectoring the control surfaces about their axis. Most of the flight control applications uses hydraulic servo systems. The hydraulic control elements can generate very high forces and can exhibit rapid responses as compared to other technologies [1].

Different control techniques are adopted for the design of the electro hydraulic actuation systems. Most of the industrial applications uses the PID controllers because of its reliability and robust performance. The tuning of PID controllers is the important part in the design. There are several methods for tuning the PID controller parameters including the traditional and the intelligent methods. The traditional method involves the Zeiglar-Nichols and the Cohen Coon tuning methods [2]. It is often hard to determine optimal values with this method in many industrial plants. Thus it is desirable to increase the capabilities of the controller by adding new features. The artificial intelligence techniques such as neural networks and fuzzy systems have been adopted to improve the controller parameter values [3], [4]. Many random search methods, such as genetic algorithm and simulated annealing have received much interest for achieving high efficiency and searching global optimal solution in the search space [5].

In this paper a particle swarm optimization technique is used to search the optimal controller parameters. Particle swarm optimization technique is a modern heuristic algorithm that has been developed from the behavior of organisms such as fish schooling and bird flocking [6]. The PSO technique can generate a high quality solution within shorter calculation time [7]. For getting the optimal controller values, the parameters of the controller are optimized using particle swarm optimization technique.

This paper is organized as follows. Section II deals with the system modelling and the conventional design method. Section III deals with the proposed particle swarm optimization technique. The implementation of PSO-PIDcontroller is done in Section IV. The simulation results are shown in Section V and the inferences and the conclusion is done in Section VI.

II. SYSTEM MODELLING

A. Mathematical Modelling

RLV consists of a booster stage and a fly back portion. During the ascent phase, it will be controlled by four fin actuators. During the re-entry and return flight, the altitude will be controlled by two primary control surfaces, ie, elevons and rudder. The RLV uses an electro hydraulic linear actuator for vectoring the control surfaces. It uses a hydraulic power unit (HPU).The HPU consists of a prime mover, an axial displacement pressure compensated pump, reservoir, check valve, high pressure relief valve and an isolation valve. The system uses petroleum based mineral oil as the power transfer medium. The hydraulic power unit is of closed circuit type and the oil is recirculated. The block diagram of the electro hydraulic actuator is shown in Fig.1. The system consists of a servo controller, servo amplifier, hydraulic power unit, servo valve, hydraulic actuator, control surface dynamics and a position sensor. In hydraulic actuator, a pressurized fluid is applied to the piston rod provides the power to move the external object. A low current is passed to the servo valve through an amplifier thereby providing the power to alter the position of the valve.

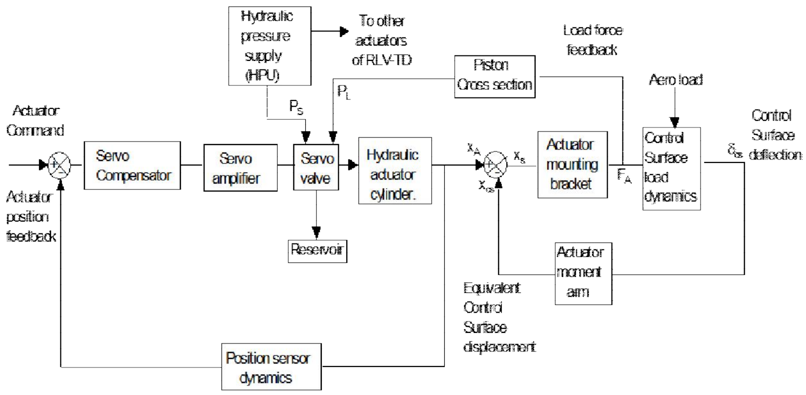

Fig.1 Functional block diagram of reusable launch vehicle actuation system

The hydraulic actuator is basically a piston cylinder mechanism. The magnitude and the direction of flow of the fluid is controlled by the servo valve. The load dynamics includes all the mechanical forces acting on the control surfaces. This dynamics along with the stiffness of all actuator mounting bracket and the hydraulic fluid constitutes the resonant frequency called as the hydro mechanical resonance. The position sensor will sense the actuator position is fed back to the position loop.

The servo valve is modelled as a second order equation. The relation between the spool movement y_v and the valve current is represented by,

$$\frac{y_v(s)}{I_v(s)} = \frac{K_v \omega_v^2}{s^2 + 2z_v w_v + w_v^2} \quad (1)$$

where K_v is the spool displacement sensitivity, w_v and z_v are the natural frequency and the damping factor of the servo valve spool. The various elements of the actuator chamber are shown in Fig.2. The servo valve consists of spools with lands machined on a cylindrical sleeve. The control ports of the servo valve are connected to the forward and return chambers of the actuator. When the spool valve moves in the forward direction, fluid enters into the forward chamber and it displaces the actuator piston in the positive direction. The linear movement of the piston is converted into mechanical motion and which in turn rotates the control surfaces about their hinge axis. The supply and the return flows are formulated based on the load flow Q_L, chamber pressures P_1 and P_2 .

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

The supply flow into the actuator chamber 1 is

$$Q_{vs} = Q_L + \frac{V_1}{\beta_e}\frac{dP_1}{dt} \quad (2)$$

The return flow from actuator chamber 2 is

$$Q_{vr} = Q_L - \frac{V_2}{\beta_e}\frac{dP_2}{dt} \quad (3)$$

where β_e is the effective bulk modulus of the hydraulic oil, V_1 and V_2 are the volume of actuator chamber.

The actuator chamber volumes are given by,

$$V_1 = V_0 + A_p x_A \quad (4)$$

$$V_2 = V_0 - A_p x_A \quad (5)$$

where V_0 is the half volume of actuator chamber, A_p is the area of cross section of the piston and x_A is the actuator displacement.

The fluid compression in the forward chamber is assumed to be equal to the fluid expansion in the return chamber,

$$\frac{V_1}{\beta_e}\frac{dP_1}{dt} = \frac{V_2}{\beta_e}\left(-\frac{dP_2}{dt}\right) = \frac{V_0}{2\beta_e}\frac{dP_L}{dt} \quad (6)$$

where $P_L = P_1 - P_2$ is the load pressure. Then combining equation (2), (3) and (4), it is obtained as,

$$Q_v = Q_L + \frac{V_0}{2\beta_e}\frac{dP_L}{dt} \quad (7)$$

The actuator piston velocity can be derived from the load flow Q_L as,

$$\dot{x}_A = \frac{Q_L}{A_P} \quad (8)$$

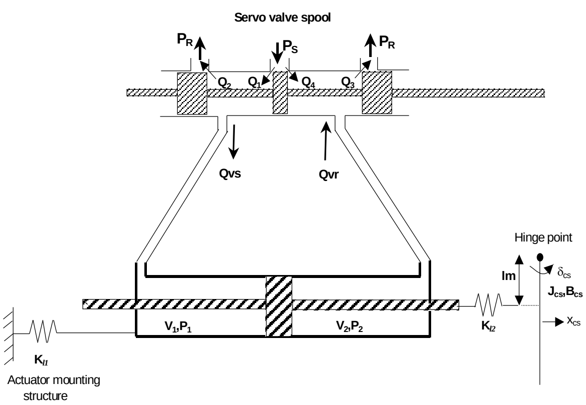

Fig.2 Dynamics of actuator chamber

The load dynamics is given by,

$$G = \frac{1}{J_{cs}s^2 + B_{cs}s} \quad (9)$$

where J_{cs} and B_{cs} are the moment of inertia and viscous damping coefficient of the control surface.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

The actuator displacement x_A is the sum of the equivalent control surface displacement, x_{cs} and the backward displacement of the actuator mounting arm bracket x_b and is given by,

$$x_A = x_{cs} + x_b \quad (10)$$

The control surface displacement x_{cs} is given by,

$$x_{cs} = \delta_{cs} L_m \quad (11)$$

where δ_{cs} is the control surface deflection and L_m is the actuator lever arm length.

The displacement of the actuator mounting arm bracket x_b includes all the mechanical flexible elements in cascade with the actuator on either side. Assuming that K_l as the equivalent stiffness of all such elements, then the actuator force is given by,

$$F_A = K_l x_b = K_l(x_A - L_m \delta_{cs}) \quad (12)$$

The control surface deflection is derived from the load dynamics equation as,

$$\frac{J_{cs}}{L_m}\frac{d^2\delta_{cs}}{dt^2} + \frac{B_{cs}}{L_m}\frac{d\delta_{cs}}{dt} = F_A + F_D - B_a\frac{dx_A}{dt} + F_r \quad (13)$$

where F_r is the coulomb friction, F_D is the total disturbance on the control surface, B_a is the viscous damping coefficient of the actuator.

The sensor has a dynamics equivalent to the second order system given by,

$$S(s) = \frac{{w_{d1}}^2}{s^2 + 2z_{d1}w_{d1} + {w_{d1}}^2} \quad (14)$$

where w_{d1} and z_{d1} are the natural frequency and the damping factor of the sensor.

The transducer output is given by,

$$V_{LVDT} = K_P \frac{1}{L_m} \delta_A \quad (15)$$

where K_p is the position sensor scale factor and δ_A is the actuator deflection.

Using all these equations the linear model of the actuation system is modelled as given in Fig.3.

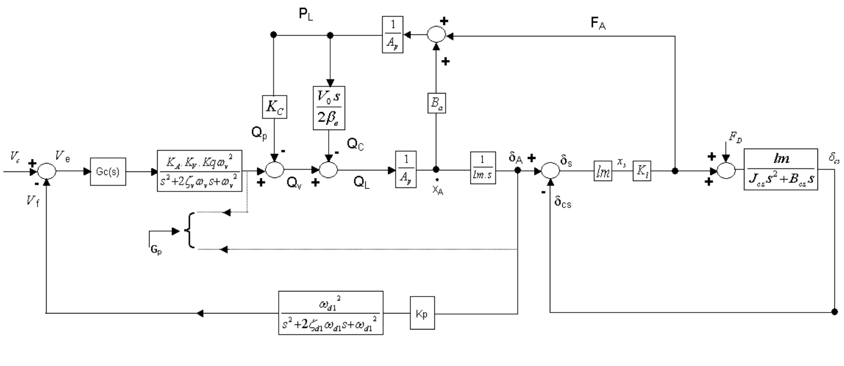

Fig.3. Linear mathematical model of actuation system

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

B. Conventional compensator design

The response analysis of the actuator is carried out. From the response analysis of the system it is found that even though the system is stable, the system specifications are not met. So in order to meet the specifications of the system a suitable compensation scheme is to be provided. The compensation scheme is developed based on the requirements of the system. It consists of a PI controller, a notch filter and a rate filter. The compensation scheme is shown in Fig.4.

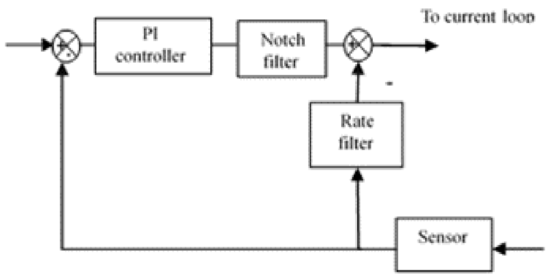

Fig.4. Compensation scheme

The PI controller is designed to offer maximum dynamic gain and to offer relative stability for systems. The transfer function for PI controller is designed as,

$$G_{PI}(s) = \frac{9s + 65}{s} \quad (16)$$

The notch filter is introduced in the forward path to attenuate the high frequency oscillations in the circuit [8]. The notch filter transfer function is given by,

$$N(s) = \frac{s^2 + 2\xi_n w_n s + {w_n}^2}{s^2 + 2\xi_d w_n s + {w_n}^2} \quad (17)$$

From the open loop frequency response the resonant peak was obtained at 21.2Hz. A notch filter centered at frequency 21.2Hz is used to attenuate the control structure oscillations. The notch filter is designed such that the ratio $\frac{\xi_d}{\xi_n}$, which is the depth of the notch is 10 and is designed as,

$$N(s) = \frac{s^2 + 13.32s + 1.774 \times 10^4}{s^2 + 133.2s + 1.774 \times 10^4} \quad (18)$$

The rate filter is used in the feedback path and the rate filter transfer function is given by,

$$R(s) = \frac{K_r s}{s + \omega} \quad (19)$$

where K_r is the rate gain and ω is the frequency of the rate loop.
The rate filter is designed with 100Hz frequency as,

$$R(s) = \frac{10s}{s + 100} \quad (20)$$

The simulation results of uncompensated and compensated systems are shown in below figures. The frequency response plots and the step response are shown in Fig.5 and 6 and 7 and the performance evaluation is given in Table.1.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

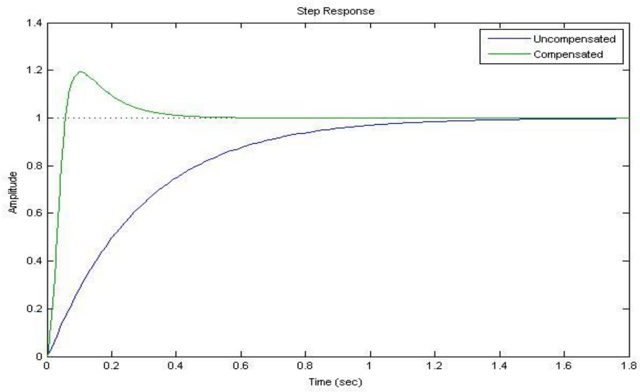

Fig.5. Open loop response of uncompensated and compensated system

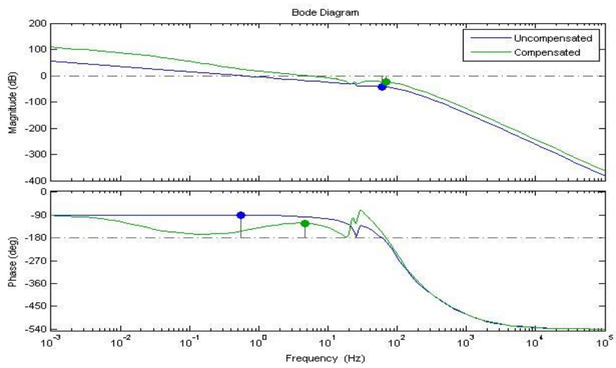

Fig.6. Closed loop response of uncompensated and compensated system

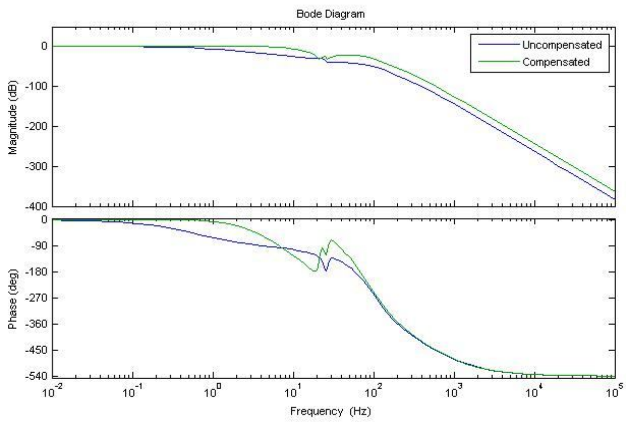

Fig.7. Step response of uncompensated and compensated system

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

Table.1. Performance evaluation of compensated system

Specification	Requirement	Compensated
Gain margin	>30degree	64.3 degree
Phase margin	>6dB	25.7dB
-3 dB bandwidth	6±0.5	7.84Hz
-90 deg bandwidth	7 ± 0.5	6.9Hz
Rise time	50±10msec	0.0414sec
Settling time	<600 msec	0.349sec
Overshoot	<20%	19.4%
Maximum Peak	<2dB	1.68dB

C. Proposed work

The proposed method designs a PID controller using a particle swarm optimization technique. The controller parameters are optimized using the proposed technique. PID controller have the combined effect of all the three control actions, ie, proportional, integral and derivative control actions. Hence the introduction of PID controller stabilizes the gain, reduces the peak overshoot of the system and also reduces the steady state error. The optimization of the PID controller yields better values for the parameters.

III. PARTICLE SWARM OPTIMIZATION

Particle swarm optimization is a modern technique used to tune optimally the gain values of a controller. Particle swarm optimization (PSO) is first introduced in 1995 by Russell Eberhart and James Kennedy and it is evolved from the social behaviour of birds and fishes. The method is highly robust in solving problems having non linearity and non-differentiability, multiple optima and high dimensionality through adaption. It has got stable convergence characteristics with high computational efficiency. Every PSO uses a population of particle [9], [10]. Each particle or the solution flies in the search space with a particular velocity. During each iteration of the algorithm, each of the candidate solution is being evaluated by the fitness function thus determining the fitness. Each candidate solution can be thought of as a particle flying through the search space finding the maximum or minimum value of the function. Initially, PSO chooses the candidate solutions randomly within the search space. The PSO algorithm has got three steps which are repeated until the stopping condition is met [11].

1. Generation of particles and their information.
2. Evaluating the fitness.
3. Updating the particles and forming new vectors.

The information of the particle refers to the position and velocity. A set of particles, n is first initialized having position and velocity and an evaluation function f is formulated based on the system requirements. Then the function is evaluated with each of the particle as the input vectors. The position and the velocity of the particle can be adjusted at each time step. When a particle discovers that any value obtained is better than any it has found previously, then it stores that value as $pbest_i$, the personal best value [12]. The overall best value which is tracked by the global version of the particle is the value represented by $gbest_i$.

Fitness evaluation is conducted by applying the candidate solution to the function formulated. Individual and global best positions are updated by comparing the new fitness value with the previously obtained value.

The velocity of each particle is updated using the formula [13],

$$v_i(t+1) = w * v_i(t) + c_1 * rand * \left(pbest_i - x_i(t)\right) + c_2 * rand * \left(pbest_i - x_i(t)\right) \quad (21)$$

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

where t is the number of iterations, $rand()$ refers to a random number between 0 and 1, w is the inertia weight which provides a balance between the personal and the global values.

The value of w ranges from 0.4 to 0.9 and it is calculated as,

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \tag{22}$$

where $iter_{max}$ is the maximum number of iterations. The constants c_1 and $c_2$are the acceleration constant terms that makes the solution near to or away from the personal and global best values. The acceleration constants are usually in the range of 1.5 or 2.Once the velocity of each particle is calculated, then the position is updated by applying the new velocity to the particles previous position as [13],

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1) \tag{23}$$

IV. IMPLEMENTATION OF PSO-PID CONTROLLER

A PID controller is designed using the PSO technique. The proportional, integral and derivative gains of the PID controller are optimized using the method. The open loop transfer function of the system is given by,

$$G(s) = \frac{4.767 \times 10^{15}(s^2 + 16.82s + 2.831 \times 10^4}{s^6(s+4414)(s+942.5)^2(s+12.555+158.2j)(s+12.555-158.2j)(s+314.15+544.15j)(s+314.15-544.15j)} \tag{24}$$

Considering there are n individuals in the population. The PID controller has got three parameters and hence the positionand velocities will be of the order $n \times 3$. The position represents the parameters of the PID controller and the initial values are given by,

$$\begin{aligned} K_p &= K_{pmin} + rand \times (K_{pmax} - K_{pmin}) \\ K_i &= K_{imin} + rand \times (K_{imax} - K_{imin}) \\ K_d &= K_{dmin} + rand \times (K_{dmax} - K_{dmin}) \end{aligned} \tag{25}$$

A fitness function is designed based on the time domain specifications and it includes rise time t_r, settling time t_s and overshoot M_p. The evaluation function is formulated as the sum of the ratios of settling time, rise time and the overshoot. The function is given by,

$$f = \frac{t_s}{t_{s0}} + \frac{t_r}{t_{r0}} + \frac{M_p}{M_{p0}} \tag{26}$$

where t_{s0},t_{r0} and M_{p0} are the settling time, rise time and overshoot values of the required system with the controller. For each of the iteration process the value of the parameters are changed and the fitness function is calculated. The searching procedure for the PSO-PID controller is as follows:

Step 1: Specify the lower and upper bounds of the controller parameters and initialize randomly the position, velocities and local best values. Enter the maximum number of iterations.

Step 2: For each of the individual solutions, evaluate the fitness value which is a function of rise time, settling time and peak overshoot.

Step3: Compare each individual's new fitness value with the best value initialized $pbest$. The best value among the $pbest$ is the $gbest$ value.

Step 4: Modify the velocity of the particle and update the position of the particle using the new velocity.

Step 5: When the number of iteration reaches the maximum, then it is stopped. The latest $gbest$ value is taken as the optimal controller parameters of the PI controller.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

The simulation parameters are taken as given in Table.2.

Parameter	Values
Population size	10
Number of iteration	50
Inertia weight factor w_{min} w_{max}	 0.4 0.9
Acceleration constants c_1, c_2	1.5

Table.2 Parameters of PSO algorithm

After optimization the values of the PID controller are obtained are shown in Table.3

Table.3 Optimal controller parameters

Parameter	K_p	K_i	K_d
Optimized value	8.2016	26.8658	0.0086

In order to meet the bandwidth requirements and to suppress the unwanted oscillations of the system a notch filter is designed. The filter centered at frequency 21.2 Hz is used to avoid the high frequency oscillations. The notch filter is designed as,

$$N(s) = \frac{s^2 + 13.32s + 1.774 \times 10^4}{s^2 + 133.2s + 1.774 \times 10^4} \quad (27)$$

V. SIMULATION RESULTS

The PID controller is designed using particle swarm optimization technique and optimal values of gain are obtained. The PSO-PID controller along with the notch filter is used for the actuation system of the RLV. The simulations are done and the frequency and step responses are plotted and the performance of the system with the controller is evaluated. The frequency response plots are given in Fig.8 and 9. The step response plot is shown in Fig.10.

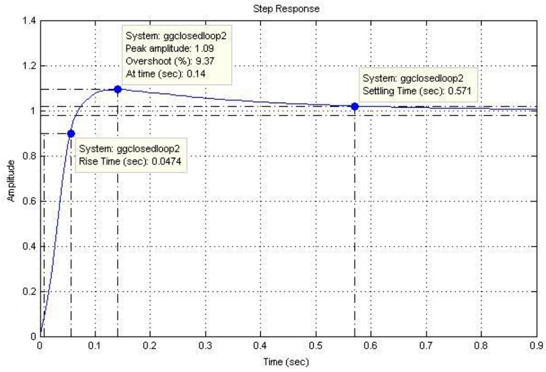

Fig.8 Open loop response with PSO-PID controller and notch filter

 DOI: 10.15662/ijareeie.2015.0408041

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

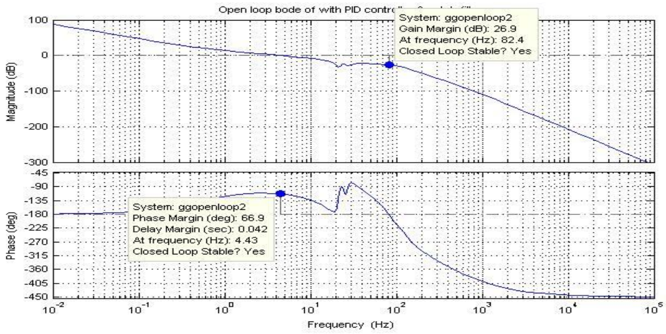

Fig.9 Closed loop response with PSO-PID controller and notch filter

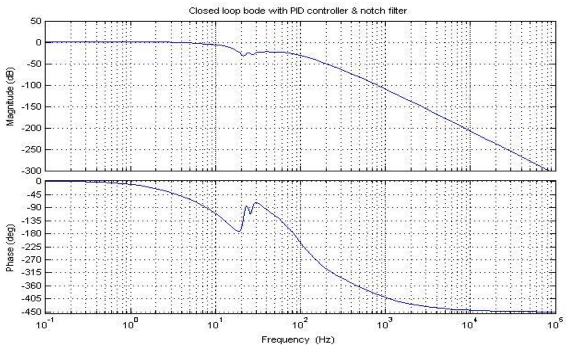

Fig.10 Step response with PSO-PID controller and notch filter

From the analysis, it is clear that the optimized PID controller shows good results. Thus by using PSO-PID controller and a notch filter, all the specifications are met and hence the use of rate filter can be avoided. The open loop, closed loop and the step response specifications are satisfied with the proposed design method and the system requirements are well within the specifications. The comparison of the proposed method and the conventional method is given in Table.4.

Table.4. Performance evaluation with the conventional method and the PSO-PID controller

Specification	Requirement	Conventional method	PSO-PID, notch filter
Phase margin	>30 degree	64.3 degree	66.9 degree
Gain margin	>6dB	25.7dB	26.9 dB
-3dB bandwidth	6±0.5	7.84Hz	6.51 Hz
-90degree bandwidth	7 ± 0.5	6.9Hz	7.48 Hz
Rise time	50±10msec	0.0414sec	0.0474 sec
Settling time	<600 msec	0.349sec	0.571 sec
Peak overshoot	<20%	19.4%	9.377 %
Maximum peak	<2dB	1.68dB	0.798 dB

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2015

VI. CONCLUSION

The optimized controller design of actuation system of RLV is considered. The conventional compensator design using trial and error method is much time consuming and it does not yield optimal results. Thus a PID controller is designed using particle swarm optimization technique. From the simulation results, it is shown that the PSO-PID controller with a notch filter showed better results as compared to the conventional method.

REFERENCES

[1] DSP Control of Electro hydraulic servo actuators, Texas Instruments, Jan.2005.

[2] K. Ogata , Modern Control Systems, University of Minnesota, Prentice Hall, 2002.

[3] A. Visioli, “Tuning of PID Controllers with Fuzzy Logic,” in Proceedings IEEE Control Theory and Applications, vol.148, no.1,pp. 1-8,Jan.2001.

[4] T. L. Seng, M.B. Khalid and R. Yusuf, “Tuning of Neuro-Fuzzy Controller by Genetic algorithm,”IEEE Trans. On Systems, Man and Cybernatics.B,vol.29,pp.226-326, Apr.1991.

[5] Ayman.A.Aly, “PID Parameters Optimization Using Genetic Algorithm Technique for Electro hydraulic Servo Control Systems,” Journal of Intelligent Control and Automation, no.2, pp.69-76, May 2011.

[6] R.C Eberhart and Y.Shi, “Comparison between genetic algorithm and particle swarm optimization,” in Proc. IEEE International Conference on Evolutionary Computing, pp. 611–616, May 1998.

[7] J.Kennedy and R. Eberhart, “ Particle Swarm optimization,”in Proc. IEEE Int. Conf. Neural Networks, Perth, Australia,vol.IV,pp.1942-1948,1995.

[8] P. Schmidt and T.Rehm, “ Notch Filter Tuning for Resonant Frequency Reduction in Dual Inertia Systems,” Proceedings of IEEE Industry Applications Conference, vol.3, pp.1730-1734, Oct.1999.

[9] Y. Shi and R. Eberhart, “A Modified Particle Swarm Optimizer,” in Proc. IEEE Int. Conf. Evol. Comput, Anchorage, AK,pp.69-73, May. 1998.

[10] Mahmud Iwan Solihin, Lee Fook Tack and Moey Leap Kean, “ Tuning of PID Controller Using Particle Swarm Optimization,” Proceedings of International Conference on Advanced Science, Engineering and Information Technology, pp.458-461, Jan.2011.

[11] Trias Andromeda, Azli Yahya, Nor Hisham Haji Khamis, Ameruddin Baharom and Muhammad Arif Abdul Rahim, “PID Controller Tuning by Particle Swarm Optimization on Electrical Discharge Machining Servo Control System,” International Conference on Intelligent and Advanced Systems,pp.51-55,2011.

[12] Zwe-Lee Gaing, “A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System,” IEEE Transactions on Energy Conversion.vol.19,no. 2, Jun.2004.

[13] James Blondin, Particle swarm optimization: A Tutorial, September 2009.