International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Tuning of PID Controller using Glowworm Swarm Optimisation on a Satellite Attitude Control Reaction Wheel

Pulkit Sharma, Vishnu G Nair,

Abstract*: This paper attempts to use GSO algorithm to tune a PID controller that can be used to control a satellite using reaction wheels. These have a higher order transfer function and the controller will be more difficult to tune due to this. To do this, a satellite is chosen which controls its attitude using reactions wheels. An axis is chosen and the reaction wheel along this axis is taken into consideration. A PID controller is then attached to this system. PID (Proportional-Integral-Derivative) are one of the most popular controllers used due to their broad applications and easy to design nature. The PID controller is tuned using Glowworm Swarm Optimization (GSO) algorithm and then the system is checked against a step input. The optimization of the controller is done by minimizing the time weighed absolute error cost function.

***Index Terms*: Glowworm Swarm Optimisation (GSO) Algorithm, Optimisation, PID Tuning, Satellite attitude control**

I. INTRODUCTION

Satellites today carry a wide variety of loads and there are many launched each year with new methods and new hardware for attitude control. Reaction wheels have proven to be one of the best methods for attitude adjustment of a satellite using no extra fuel, but only electricity to run its motors. Electricity, which can be readily made available using solar power available on almost all satellites. In references [4] and [8], they have attempted to optimize a PID controller for 2 different cases using Fruit Fly Optimization algorithm. In the papers [12], [2] and [3], PSO algorithms are used to tune PID controllers used in different systems. The results they achieved are positive, however the system transfer functions were of a lower order than the ones in the case of a satellite. Reference [9] tests the PSO algorithm on various fitness functions and finds it to be on par with the other, though sometimes it could not arrive at the optimum value. It highlights the problems of choosing the correct fitness function and shows how increasing the generation size after a certain level only results in marginal improvements in the result. Reference [1] compares PSO with Zeigler Nichols and PSO obtains better results.

This paper is an attempt to control a satellite using GSO, which can find the global optima, as opposed to PSO, which may sometimes arrive on the local optima and miss the global optimum. This paper is an attempt to control a satellite's attitude using Glowworm Swarm Optimisation. The same is then done using Genetic Algorithm for comparison. The satellite is assumed to be a rigid body and the dynamics is expressed in the form of a MIMO system. One of the SISO systems is then selected from these. This SISO system is then connected with a classic PID controller. The optimization is carried out using GSO, which is used to find the optimal values of the PID constants.

II. METHODS

In a chosen satellite with reaction wheels, one axis is chosen and our PID controller is attached to it. This controller is then tuned using GSO algorithm to find the optimum values of the PID constants. This is then compared with the constants obtained using genetic algorithm. This is done by deriving the dynamics equation of the satellite and reaction wheels, the disturbance is modelled and then attached with it the transfer function of the PID controller.

A. Glowworm Swarm Optimisation (GSO)

This is an evolutionary optimization that is used to find the global maxima or minima of a given multi-modal function. It is based on the glowworm in which the less luminous glow worms move towards the more luminous glow worms that are nearby [6].

The algorithm of GSO consists of several steps defining how each glowworm behaves and how they are used to determine the peaks of a multi-modal function. There are three basic mechanisms at work [6].

Fitness broadcast is carried in the luminescent pigment of the glowworm, called luciferin. It is proportional to the value of the function at their current position and this works under the assumption that the luciferin sensed by other glowworms does not change with distance [6]. A glowworm tends to move towards its neighbour if the neighbour glows brighter than itself. In case of multiple such neighbours, a probabilistic mechanism is used [6].

Revised Manuscript Received on 30 July 2019.
***** Correspondence Author
Pulkit Sharma*, Department of Aeronautical and Automobile Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India 576104.
Vishnu G Nair, Department of Aeronautical and Automobile Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India 576104.

A glowworm will take another glowworm to be in its neighbourhood only if it is within its neighbourhood radius. This radius is modulated by using a heuristic and it can only vary within a certain range [6].

GSO starts by placing n well dispersed glowworms in the workspace. All the glowworms initially contain an amount l_o of luciferin. Each cycle consists of three phases, namely the luciferin update phase, the movement phase and the neighbourhood range update phase. These phases together form the GSO algorithm [6].

Luciferin update phase updates the luciferin value of glowworms depending upon the value of the given function at their present position, according to [6]. The rule for the update of luciferin is

$$li\ (t + 1)=(1-\rho)li\ (t) + \gamma\ J\ (xi\ (t + 1)) \quad (1)$$

Where $l_i(t)$ is the luciferin level of glowworm $i$at time t, ρ is the luciferin decay constant such that $(0< \rho<1)$, γ represents luciferin enhancement constant and the value of the function at the glowworm's position at the given time is $J(x_i(t))$.

During the movement phase, a probabilistic mechanism is used by the glowworm to decide which neighbour with a higher luciferin value to move towards, as determines by [6]. The probability is calculated using the following formula.

$$P_{ij}(t) = (l_j(t)-l_i(t))/(\Sigma_{k\in Ni(t)}l_k(t)-li(t)) \quad (2)$$

Where the glowworm $i$is to move towards the glowworm j, which has a higher luciferin value. Here, $j \in N_i\ (t)$, $N_i\ (t) = \{j : d_{ij}\ (t) <r_d^i(t); l_i\ (t) <l_j\ (t)\}$ are the neighbours of the glowworm $i$at time t, the Euclidean distance between the two glowworms is $d_{ij}(t)$ and $r_d^i(t)$ is the neighbourhood range, which is variable and associated with i and t. The discrete time model of glowworm movements for glowworm $i$that selects a glowworm $j \in N_i(t)$ with $p_{ij}(t)$ is

$$x_i(t+1)=x_i(t)+s(x_j(t)-x_i(t))/(||x_j(t)-x_i(t)||) \quad (3)$$

Here, $x_i\ (t) \in \boldsymbol{R}^m$ is the location in the m-dimensional real space of the glowworm $i$at time t, s is the step size (>0) and $(\|\cdot\|)$ is the Euclidean norm operator [6].

If the neighbourhood range is kept fixed, and in case this encompasses the entire range of the given function, all the glowworms will collect on the global peak and all the local peaks will be ignored. This is taken care of in the neighbourhood range update phase. Since no prior information is available as to what type of function will be used in this algorithm, an adaptive neighbourhood range remains the best option to make sure all peaks of a function are detected [6].

$$r_i^d(t + 1) = \min\{r_s , \max\{0, r_i^d(t) + \beta(n_t- |N_i\ (t)|)\}\} \quad (4)$$

The initial neighbourhood range of each glowworm is r_0, and the above rule is used to update the neighbourhood range of every glowworm. Here, β is a constant parameter and the parameter used to control the number of neighbours is n_t.

The quantities given by the symbols ρ, γ, s, β, n_t, and l_0 are parameters used in the algorithm whose values have been arrived on using extensive numerical experimentation and are given in table 1.

Table 1: Experimentally obtained values of glowworm optimisation parameters

Parameters	Values
ρ	0.4
γ	0.6
β	0.08
n_t	5
s	0.03
l_0	5

Only n and r s are to be varied for different cases to find the values which are most suitable for that particular case as stated by reference [6].This algorithm is given in figure 1.

B. Genetic Algorithm (GA)

Genetic algorithm is an evolutionary algorithm used to find solutions for optimisation problems. It generates a random population sample at each iteration and the best specimen form the sample, which is determined by the value of the cost function whose optimum is to be found, approaches the optimal solution [5]. It is a three-step process.

1. The selection rule which selects the individuals that will get to contribute to the next generation
2. The crossover step that combines two individuals to create children
3. The mutation step that applies random changes to the individual parents who will create the children for the next generation

C.PID Controller

A proportional integral derivative (PID) controller is a type of control loop feedback mechanism that is commonly used as a controller in various applications. It calculates the error between the feedback value and the desired value, using this error to determine the output using the proportional, integral and derivative terms.

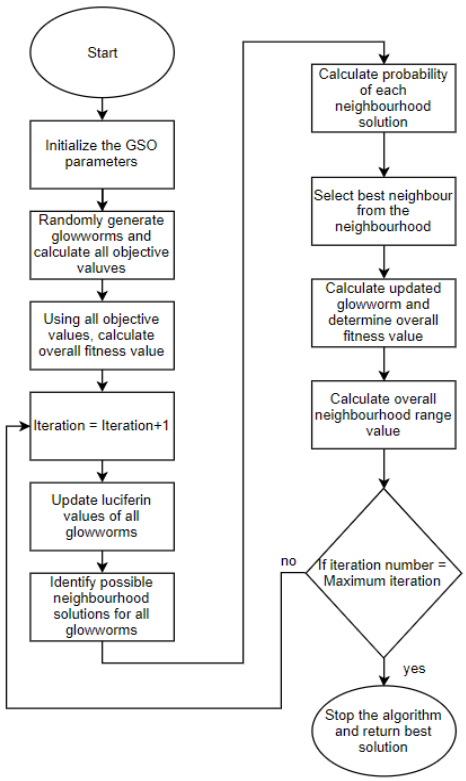

Figure 1: GSO algorithm block diagram

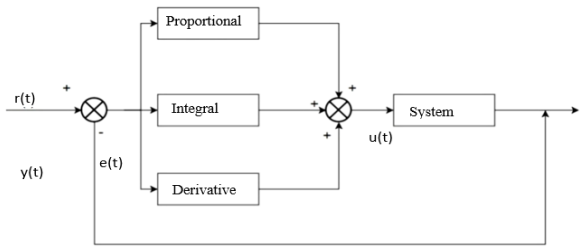

Figure 2: Typical PID controller

Figure 2 is the block diagram representation of a PID controller. The desired value to be achieved is r(t), which is compared with the current value of the system y(t). The difference between them, e(t), is used to compute the control variable u(t). The control variable is the sum all three components of the PID controller. The PID controller will be tuned by finding the appropriate values of K_p,$T_i$and T_d.

$$U(t) = K_p\left(P + \frac{1}{T_i}I + T_d D\right) \quad (4)$$

Here, P is the proportional component of the controller, I in the integral component of the controller and D is the derivative component of the controller. In equation (4), we substitute the tuned values for K_p ,T_i and T_d.

D. Cost Function

The GSO algorithm is used to find the maximums of a given function. The time weighed absolute error is used here. The cost function is denoted by J. The cost function J is given in equation 5.

$$J = \int_0^T t|e(t)|dt \quad (5)$$

Where, t is the time, e is the difference in required value of amplitude and actual value of amplitude at time t and dt is the size of the time step. This integration, when carried out over a period of time, can be denoted by equation 6.

$$J = \sum_{t=0}^{T} t|e|dt \quad (6)$$

The time weighed absolute error function is taken to minimize the deviation over a certain period of time T, thereby making sure that the transfer function has both low settling time as well as low maximum overshoot. Here, the values of the fitness function J for lower values of t will not hold much weight whereas a deviation later in time, i.e. for a higher value of t will hold more weightage. The absolute value of the difference in amplitudes e is multiplied to the size of the time step taken dt by the MATLAB^R step function, and this is then multiplied by t.

The open loop transfer function of the satellite attitude control system as give in figure 3 can be found by varying the K_p, T_i and T_d values and finding the resulting transfer function for each value of K_p, T_i and T_d and then solving the loop to find the open loop transfer functions of the system. A step input in then given to these open loop transfer functions and the response is plotted. The cost function J values are plotted against each value of K_p, T_i and T_d using MATLAB^R and minima of the resulting plot are found using the GSO algorithm. The values of the PID constants K_p, T_i and T_d, are varied from -3000 to 3000 and the minimum values of the cost function are found using GSO.

In the GSO algorithm, the bounds of workspace range is given as 3000 to encompass this and ρ, γ, β, n_t, and l_0 are left unchanged. The values of s,n and r_s are changed suitably to obtain the results. When the value of s was taken as 0.3, but the glowworms failed to converge on any values due to the large size of the workspace. To get around this problem, the value of s has been increased from 0.03 to 50 for this particular case. The glowworms collect about different minima values, and the values of all these glowworms is then checked to find out which corresponds to the lowest value of cost function J. The PID constants are found corresponding to this point and these are the tuned PID parameters. This is then compared with the result obtained by minimising the same cost function using genetic algorithm (GA).

Table 2: Glowworm optimisation parameters

Parameters	Values
ρ	0.4
γ	0.6
β	0.08
n_t	5
l_0	5
s	50
n	1000
r_s	1000

E. Satellite Dynamics and Disturbance Modelling

Satellite dynamics are modelled here in the form of transfer functions and the disturbances are suitably factored in.

The satellite is controlled by 3 reaction wheels in the 3 different axes. Where ϕ, θ and ψ represent roll pitch and yaw respectively [10].

$$\dot{\phi}=p+[q\sin\phi+r\cos\phi]\tan\theta$$
$$\dot{\theta}=q\cos\phi-r\sin\phi \quad (7)$$
$$\dot{\psi}=[q\sin\phi+r\cos\phi]\sec\theta$$

$\boldsymbol{\omega}=[p\ q\ r]^T$ is the angular velocity vector of orbital reference frame which is stated in the body frame and is with respect to body frame [10]. A body axis frame is selected so as to conform its axes to main axes of inertia. $\boldsymbol{\omega}^B{}_{RB}$ satisfies the following equation:

$$\boldsymbol{\omega} = \boldsymbol{\omega}^B{}_{RB} - \boldsymbol{\omega}_o \quad (8)$$

Here, $\boldsymbol{\omega}_o$ is used to denote the angular velocity of the orbit, C and S denote sine and cosine respectively and $\boldsymbol{\omega}=[\omega_x\omega_y\omega_z]^T$is the angular velocity of body frame with respect to inertial frame obtained from Euler's momentum equations as stated by reference [10]. Assume body frame axis conform to main inertial axis, then the Euler's momentum equations become [10]:

$$\boldsymbol{T_d}+\boldsymbol{T_G}=\begin{bmatrix} \dot{h}_x + \dot{h}_{wx} + (\omega_y h_z - \omega_z h_y) + (\omega_y h_{wz} - \omega_z h_{wy}) \\ \dot{h}_y + \dot{h}_{wy} + (\omega_z h_x - \omega_x h_z) + (\omega_z h_{wx} - \omega_z h_{wz}) \\ \dot{h}_z + \dot{h}_{wz} + (\omega_x h_y - \omega_y h_x) + (\omega x h_{wy} - \omega y h_{wx}) \end{bmatrix} \quad (9)$$

Where $\boldsymbol{T_d}$is the total disturbance momentum acting on the satellite, $\boldsymbol{h}$is angular momentum vector of rigid satellite body, $\boldsymbol{h_w}$ is angular momentum vector of reaction wheel and $\boldsymbol{T_c}$ is gravity gradient momentum [10].

$$\boldsymbol{T_G} = \begin{bmatrix} 1.5\omega_0 2(Iz - Iy)\sin^2\phi\cos 2\theta \\ 1.5\omega_0 2(Iz - Ix)\sin 2\theta\cos\phi \\ 1.5\omega_0 2(Ix - Iy)\sin 2\theta\sin\phi \end{bmatrix} \quad (10)$$

The equation (6) and the equation (9) are the nonlinear dynamic equations of satellite's attitude [10].

This satellite has 3 reaction wheels, one on each axis. These reaction wheels are used to control the attitude of the satellite. When rotated, these reaction wheels exert a moment on the satellite in the opposite direction of their own rotation. They are rotated in the direction of the body axis [10]. The transfer function of the reaction wheel is given in equation (10), where $\boldsymbol{u}$ is the output of the controller and $\dot{\boldsymbol{h}}_{\boldsymbol{w}}$ is the external torque to satellite in the direction of related axis [11]. Transfer function of the block diagram is the following:

$$\frac{\dot{h}_w}{u}(s) \approx 1 \qquad (11)$$

$$\text{tf} = \frac{-s^3 - (1.533 \times 10^{-17})s^2 - 0.0000001996\,s}{s^5 + 8.132 \times 10^{-20}s^4 + 0.000003588s^3 + 3.352 \times 10^{-25}s^2 + 1.267 \times 10^{-12}s} \qquad (12)$$

$$tf = \frac{-0.0003252\,s^2 - 1.004 \times 10^{-9}}{s^5 + 8.132 \times 10^{-20}s^4 + 0.000003588s^3 + 3.352 \times 10^{-25}s^2 + 1.267 \times 10^{-12}s} \qquad (13)$$

$$tf = \frac{0.0009435s^2 + 2.503 \times 10^{-10}}{s^5 + 8.132 \times 10^{-20}s^4 + 0.000003588s^3 + 3.352 \times 10^{-25}s^2 + 1.267 \times 10^{-12}s} \qquad (14)$$

Among these, the PID controller to be optimized is attached to the following transfer from input 1 to output 1 as given in (12). This denoted the output of ϕ with an input of $\dot{\boldsymbol{h}}_{\boldsymbol{wx}}$.

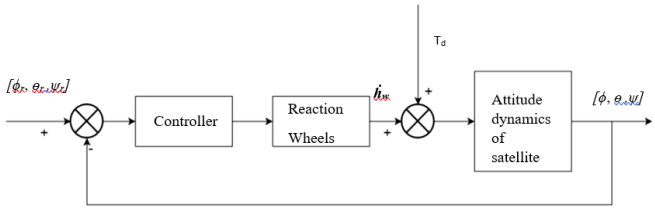

Figure 3: Block diagram of a satellite attitude control system

The satellite's specifications are given in table 3. The block diagram of the satellite is given in figure 3 as shown by reference [10].

In a space environment, a satellite has many outside torques acting on it, due to various disturbance forces that are present. The modelling of these torques in a satellite is very important as these disturbances are often non-negligible and omitting them can render a control system useless. They are of various types.

Table 3: Physical and orbital characteristics of the satellite [10]

Values	Parameters
0.75×0.75×0.75	Size of satellite body (m)
3×0.75	Size of solar panel (m)
15	Mass (kg)
[1 4 2]	Moment of inertia (kg·m^2)

Magnetic torque acts on the magnetic materials that are on the satellite and is due to Earth's magnetic field.

$$\boldsymbol{T_m}=\mathbf{M}\times\mathbf{B} \qquad (15)$$

Current in the satellite can generate magnetic moment through both induced magnetism a permanent magnetism. This is denoted by **M** and it is called residual magnetic moment. **B** is the geocentric magnetic flux density, which is:

$$\boldsymbol{B}=(\mu_f/|\boldsymbol{r}|^3)\begin{bmatrix} cos\omega_0 t\ sin\ i_m \\ \cos i_m \\ 2sin\omega_0\, tcos\ i_m \end{bmatrix} \qquad (16)$$

The equations that are derived here can be taken to be the actual satellite dynamics equations of the satellite, include both the structure of the satellite as well as the actuators. The controller of the reaction wheels that have been attached with this satellite is attached separately [10]. A PID controller will be used. The desirable angle which the satellite should track is around the zero angles. Using the concept of linear control and simplifying the analysis, the equations (7), (8), (9) and (10) can be linearized. The radius of the orbit is taken to be 10,000km. Various transfer functions available in the literature are given below in equations (12), (13) and (14).

Where $\mu_f = 7.9\times10^{15}$ Wb·m, **r** is beam of the satellite's orbit and i_m is the angle between satellite's orbit and the geomagnetic equator [7].

Atmospheric forces act on the body of the satellite when it is in the upper layers of the atmosphere and the aerodynamic torque it produces on the body of the satellite can be obtained by

$$\boldsymbol{T_a}=\frac{1}{2}\rho|\boldsymbol{v}|^2 C_d A_a(\boldsymbol{u_a}\times\boldsymbol{s_{cp}}) \qquad (17)$$

Where ρ is the atmospheric density, $\boldsymbol{v}$ is the velocity of the satellite, C_d is the drag coefficient, $\boldsymbol{u_a}$ is the unit vector along the direction of velocity, A_a is the vertical surface area on u_a and $\boldsymbol{s_{cp}}$ is the vector from centre of mass of satellite to centre of pressure [13].

The solar torque acts on the satellite due to solar radiation falling on the body of the satellite.

$$\boldsymbol{F_s} = \frac{1367}{c} A_s(1+q)\boldsymbol{u_s} cos\gamma \qquad (18)$$

Where c is the velocity of light, A_s is area of the satellite's surface that is exposed to solar radiation, q is the reaction coefficient, γ is the angle of radiation and $\boldsymbol{u_s}$ is the unit vector along sun radiation direction. Now, the dynamics equations of satellite's attitude and reaction wheel are connected to a PID controller [11].

III. RESULTS AND DISCUSSION

The PID was optimised using the GSO algorithm and a certain point was obtained for which the value of the cost function J is minimum. Most of the glowworms are collected around the optima values in figure 6. There are many optima values that the glowworms have found in the workspace. In figure 4, we can see that all the glowworms, that are scattered evenly all over the workspace volume, move about in the workspace volume, and, after a certain number of iterations, collect about the different optima values of the selected cost function J.

These points, which are shown in figure 5, are then all checked to find out which is the most optimum among them. This point is found and the values of PID constants corresponding to this point are taken as the corresponding K_p, T_i and T_d constants for the PID controller.

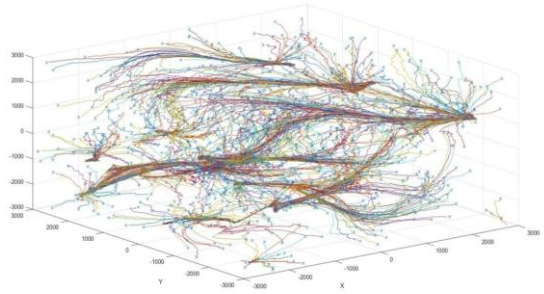

Figure 4: Evolution of Glowworms

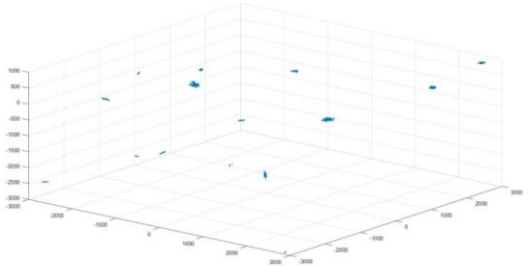

Figure 5: Final position of Glowworms

For comparison, the same cost function is then optimised from the value -3000 to the value 3000 using the Genetic Algorithm (GA). The values for various parameters and the results are then compared.GSO is found to have arrived at a much lower value of the cost function J as compared to GA. For GA, the bound is taken from 0 to 3000 for all three variables. The fraction and interval are taken as 0.2 and 20 respectively in the migration phase. In the reproduction phase, the elite count is taken as 0.05 times the population size and the crossover fraction is taken as 0.8. The initial penalty is 10 and the penalty factor is 100. The population is taken as 50.

Table 4: Results

Parameters	GSO-PID	GA-PID
Proportional constant	-2.9137×10^{-3}	-0.0029856
Integral constant	0.3970	-844.7360
Derivative constant	-97.7491	-387.8940
Settling time (sec)	1.9800	1.9818
Rise time (sec)	0.8000	0.8014
Peak time (sec)	1	5
Maximum overshoot (%)	1.8929×10^{-10}	0.066
Cost function	2.3541×10^{-4}	5.843661238236589
Gain margin	Infinite	728.4165
Phase margin	179.9925	-179.999
Gain crossover frequency	Not applicable	1.4757
Phase crossover frequency	0.2350	0.0974

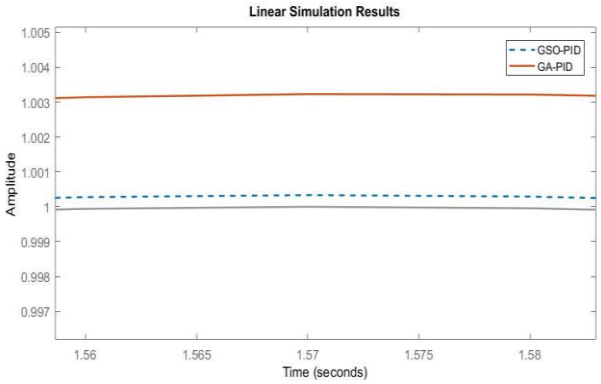

Figure 6: Sinusoidal response for GSO-PID and GA-PID

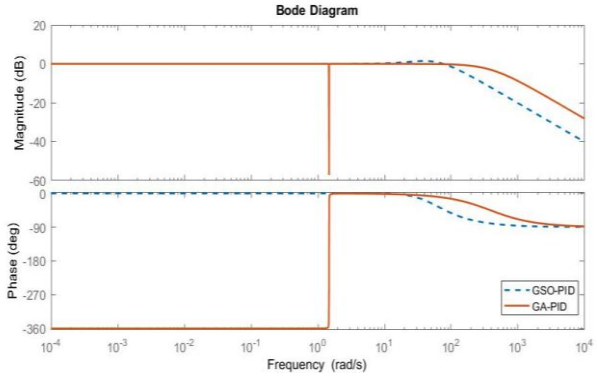

Figure 7: Bode plot for GSO-PID and GA-PID

In figure 6, the sinusoidal response plot for both GSO-PID and GA-PID is given and it has been enlarged at the top of the sine wave so that the oscillations can be more easily observed. The thick dotted line represents GSO-PID response and the GA-PID response is represented by the thick continuous line. The thinner line represents the input function which is given in the equation (20) where *t* represents time.

$$u = \sin(t) \quad (19)$$

The GA-PID response rises above 1.003 whereas the GSO-PID response remains closer to the input by being marginally above 1.000, and hence it can be observed that the response by GSO-PID is closer to the input function than the GA-PID controller.

In figure 7, the continuous line represents GA-PID whereas the dotted line represents GSO-PID. We can see that the bode plot for GSO-PID is constant for lower frequencies for both amplitude and the phase is 0 as well. Whereas for the GA-PID, the amplitude drops at a higher frequency than it does for GSO-PID, and the phase at all the lower frequencies, up until 1 rad/s is -360 degrees. For higher frequencies, the phase drops to -90 degrees for GSO-PID before it does for GA-PID. Both cases have significant differences in their control characteristics, with GSO-PID giving the system better control characteristics as compared to GA-PID. GA-PID has a much higher phase margin and the gain margin is approximately the same for both GA-PID as well as GSO-PID.

Phase margin is negative for GSO-PID as well as for GA-PID, therefore GSO-PID is comparatively less unstable as compared to GA-PID is unstable. The GA-PID is seen to cross both the zero decibels as well as the zero-degree line more than once.

IV. CONCLUSION

GSO is found to be a great method for tuning of a PID controller since it gives accurate and highly optimum solutions and is also easy to implement on code. Being tested for a high order transfer function, it can also be said to be very robust. GSO can be used to arrive at the global minima as well as the local minima, as opposed to GA, which is susceptible to getting stuck at a local minimum and never arriving at the global minima of the cost function, as can be observed in this case. It can be used to make satellite attitude adjustments much more precise and efficient. The GSO-PID achieves the desired state with less oscillation when compared to the GA-PID. GSO-PID is found to be stable, whereas the GA-PID is found to be unstable. Also, the different parameters of the GSO, such as the number of glowworms, the workspace range and step distance are given by research experience and can be improved upon. It shows promise, and more research should be done in this regard to achieve more efficient control of the satellite.

ACKNOWLEDGEMENT

The authors would like to thank Manipal Institute of Technology, Manipal Academy of Higher Education for providing the facilities needed for this research.

REFERENCES

1. Aranza, M.F., Kustija, J., Trisno, B., and Hakim, D.L., 2016, Tunning PID controller using particle swarm optimization algorithm on automatic voltage regulator system, IOP Conf. Series: Materials Science and Engineering, 128 (2016) 012038. DOI:10.1088/1757-899X/128/1/012038
2. Dashti, M., Shojaee, K., Seyedkashi, S.M.H., and Behnam, M., 2010, Tuning of Digital PID Controller Using Particle Swarm Optimization, Proceedings of the 29th Chinese Control Conference,July 29-31, 2010, Beijing, China
3. Gaing, Z.L., 2004, A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System, IEEE Transactions on Energy Conversion, VOL. 19, NO. 2, JUNE 2004. DOI: 10.1109/TEC.2003.821821
4. Han, J., and Wang, P., 2012, Tuning of PID Controller Based on Fruit Fly Optimization Algorithm, Proceedings of 2012 IEEE International Conference on Mechatronics and Automation, August 5 - 8, Chengdu, China, page 409-413. DOI: 10.1109/ICMA.2012.6282878
5. H Holland, John, 1975, Adaption in Natural and Artificial Systems, University of Michigan Press, January 1, Michigan, USA
6. Kaipa, K.N., and Ghose, D., 2017, Glowworm Swarm Optimization, Studies in Computational Intelligence 698, page 21-56, Springer International Publishing AG 2017. DOI 10.1007/978-3-319-51595-3_2
7. Kulkarni, J., Campbell, M., 2004, An approach to magnetic torque attitude control of satellites via 'H∞ ' control for LTV systems, 2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601), 2004, pp. 273-277 Vol.1. DOI: 10.1109/CDC.2004.1428642
8. Latha, K., Rajinikanth, V., and Surekha, P.M., 2013, PSO-Based PID Controller Design for a Class of Stable and Unstable Systems, ISRN Artificial Intelligence, vol. 2013, Article ID 543607, 11 pages, 2013. DOI:10.1155/2013/543607
9. Li, X.Z., Yu, F., and Wang, Y.B., 2007, PSO Algorithm Based Online Self-Tuning of PID Controller, 2007 International Conference on Computational Intelligence and Security (CIS 2007), Harbin, 2007, pp. 128-132. DOI: 10.1109/CIS.2007.194
10. Mohsenipour, R., Nemati, H., Nasirian, M., and Nia, A. K., 2013, Attitude Control of a Flexible Satellite by Using Robust Control Design Methods, Intelligent Control and Automation, Vol. 4 No. 3, 2013, pp. 313-326. DOI: 10.4236/ica.2013.43037.
11. Sidi, M.J., 1997. Spacecraft Dynamics and Control a Practical Engineering Approach, New York, Cambridge University Press.
12. Solihin, M.I., Tack, L.F., and Kean, M.L., 2011, Tuning of PID Controller Using Particle Swarm Optimization (PSO), Proceeding of the International Conference on Advanced Science, Engineering and Information Technology 2011 Hotel Equatorial Bangi-Putrajaya, Malaysia, 14 - 15 January 2011. ISBN 978-983-42366-4-9
13. Yang, C.D., Sun, Y.P., 2002, Mixed H2/H∞ State-Feedback Design for Microsatellite Attitude Control, Control Engineering Practice, Vol. 10, No. 9, 2002, pp. 951-970. DOI:10.1016/S0967-0661(02)00049-7

AUTHORS PROFILE

Pulkit Sharma, is an undergraduate student at the Department of Aeronautical and Automobile Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India. He is pursuing his studies in aeronautical engineering.

Vishnu G Nair is an Assistant Professor in the Department of Aeronautical and Automobile Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India. He holds a post-graduate degree in Space engineering.