

## Spacecraft pitch control using SOPDT Method

Lama Mohammed<sup>1</sup>, Muawia Mohamed Ahmed<sup>2</sup>

P.G. Student, Department of Control Engineering, Faculty of Engineering, El.Neelain University, Khartoum, Sudan,<sup>1</sup>

Associate Professor, Department of control Engineering, Faculty of Engineering, El.Neelain University, Khartoum, Sudan,<sup>2</sup>

**Abstract:** High order transfer function is one of the most problems for controller's designers especially in critical applications fields which cannot accept a high rate of risk. This paper focuses on one of the useful techniques in control field to simplify the complicated higher order transfer function of the plant to be in a lower order form. This is more easier for designing the controller. This is done by converting higher orders to second orders with the same response but with a dead zone. This technique is called Second Order Plus Dead-Time which abbreviated in SOPDT. The plant taken as a model in this paper is the spacecraft pitch transfer function.

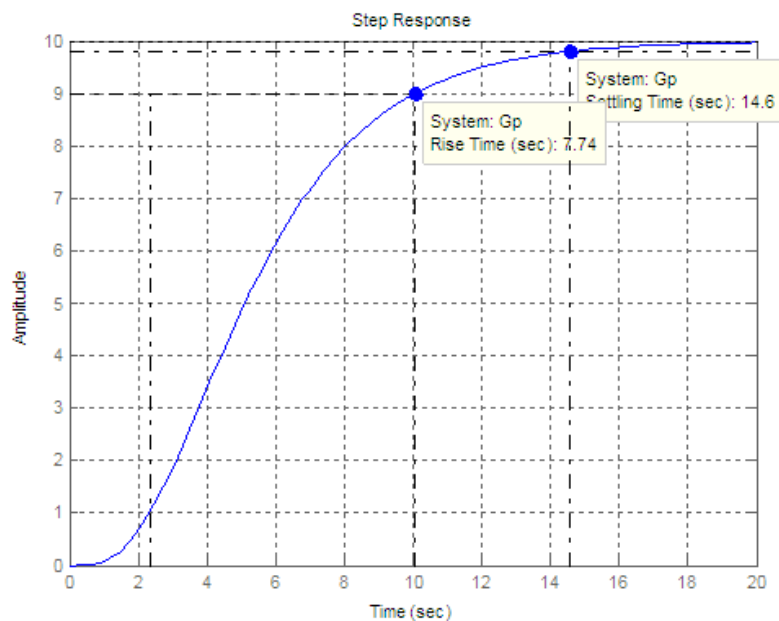
**Keywords:** space craft, pitch, TF, SOPDT, PID.

### I. Introduction

The pitch transfer function for high altitude is [1]:

$$TF1 = \frac{3.6}{S^4 + 3.36 s^3 + 4.08 s^2 + 2.08 s + 0.36} \quad \text{----- (1)}$$

The step response of the pitch TF is:



**Figure (1):** open loop step response for high altitude pitch spacecraft TF

When manipulating with third, fourth or any complicated TF for any system it's difficult to apply any type of controller to the system although there are many techniques to fix this problem.

SOPDT (second order plus dead-time) technique to convert the fourth order pitch spacecraft transfer function to second order TF with dead-time zone (delay element) is acceptable and provides good results. Applying the PID controller to control the pitch spacecraft system in a SOPDT form will be much easier.

## II. Methodology

The factorial form of the TF from equation (1) is:

$$TF = \frac{3.6}{(s+1)(s+1)(s+1)(s+0.36)} \quad \text{----- (2)}$$

When converting this TF into a SOPDT form the new TF will be as shown in equation (3) and its final form is shown in equation (7).

$$TF = \frac{3.6}{(s+1)(s+1)(s+1)(s+0.36) (2.7778/ 2.7778)} \quad \text{----- (3)}$$

$$TF = \frac{10}{(s+1) (s+1) (s+1) (2.7778s+1)} \quad \text{----- (4)}$$

The TF now matches with SOPDT form [2][3]:

$$G(s) = \frac{K_p}{(T_1s+1)(T_2s+1)(T_3s+1)(T_4s+1)} \quad \text{----- (5)}$$

Then we can transform that to:

$$\square \frac{K_p e^{-(T_3+T_4)s}}{(T_1s+1)(T_2s+1)} \quad \text{----- (6)}$$

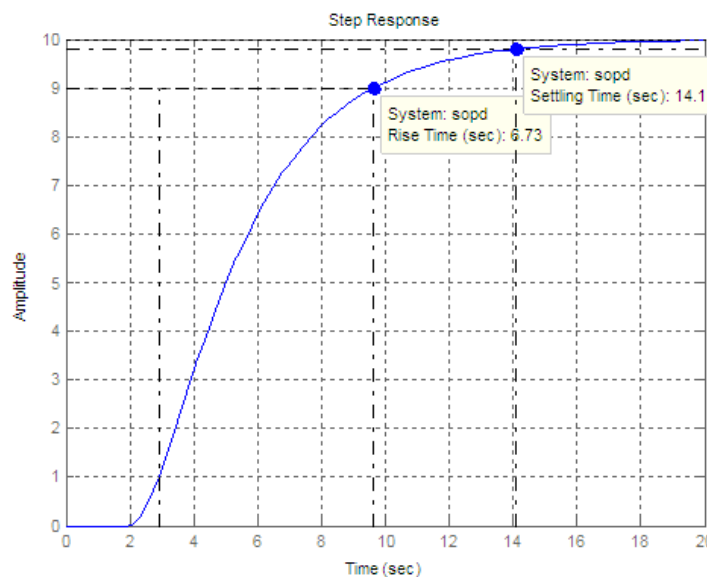
By choosing the largest T1 and T2 from equation (4) and then getting the final transfer function with delay element for the pitch spacecraft system:

$$TF = \frac{10 e^{-(1+1)s}}{(s+1)(2.7778s+1)}$$

$$TF = \frac{10 e^{-2s}}{(s+1)(2.7778s+1)}$$

$$TF_2 = \frac{10 e^{-2s}}{2.778 s^2 + 3.778 s + 1} \quad \text{----- (7)}$$

The step response of the TF is:



**Figure (2):** step response for delayed system

As shown in figure (1) and figure (2) the step responses before and after applying SOPDT techniques have almost the same responses but with dead zone equal two. Referring to equation (7) the system has a transfer function with delay element (equal 2 seconds) and it's somehow difficult to deal with.

To fix this problem, the delayed TF was converted to state space model [4]. The matrices of the state space are found to be as:

$$A = \begin{bmatrix} -1.36 & -0.72 \\ 0.5 & 0 \end{bmatrix}$$

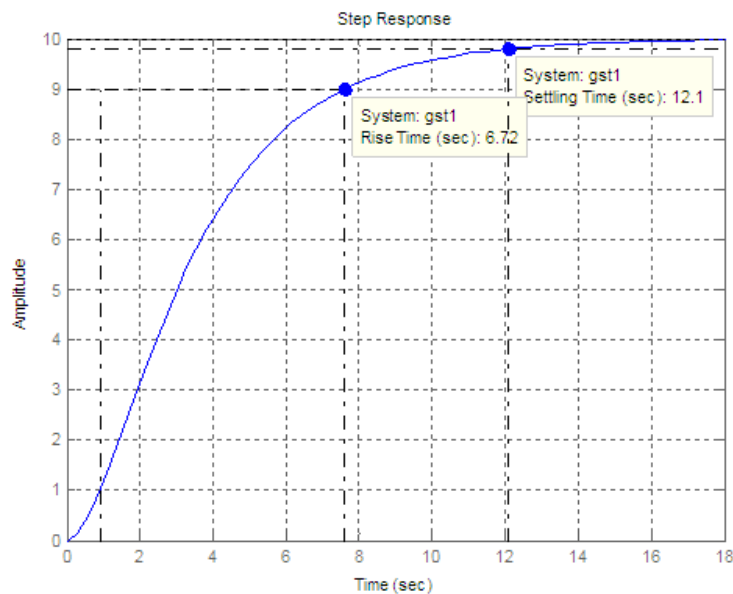
$$B = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 3.6 \end{bmatrix}$$

The corresponding TF of this state space is found to be:

$$TF3 = \frac{3.6}{S^2 + 1.36s + 0.36} \quad \text{----- (8)}$$

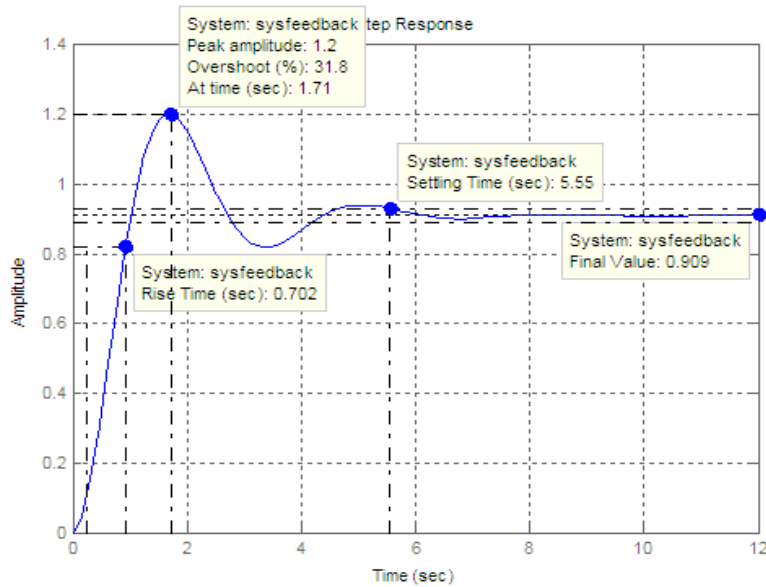
The step response of equation (8) is shown in figure (3):



**Figure (3):** open loop step response for the second order TF

The closed loop Transfer function and step response with a unity feedback before PID controller is:

$$TF4 = \frac{3.6}{S^2 + 1.36s + 3.96} \quad \text{----- (9)}$$



**Figure (4):** closed loop step response before PID controller

As shown in figure (4) the system has overshoot equal to 31.8% and rise time equal to 0.702 and settling time equal to 5.55 .

To choose the PID controller parameters (kp, ki, kd) must take those conditions into an account to get an optimal parameters for the PID controller that will obtain:

- Fast rise time
- Minimum overshoot
- No steady-state error.

A proportional controller (kp) will reduce the rise time and will reduce but never eliminate the steady state error. An integral control (ki) will eliminate the steady-state error, but it may make the transient response worse. A derivative control (kd) will increase the stability of the system, reducing the overshoot, and improving the transient response.

The effects of each of controller’s parameters kp, kd, and ki on a closed-loop system are summarized in table (1).

**Table (1)** the PID parameters [5]

Closed Loop RESPONSE	RISE TIME	OVERSHOOT	SETTLING TIME	S-S ERROR
<b>Kp</b>	Decrease	Increase	Small Change	Decrease
<b>Ki</b>	Decrease	Increase	Increase	Eliminate
<b>Kd</b>	Small Change	Decrease	Decrease	Small Change

The PID controller TF is:

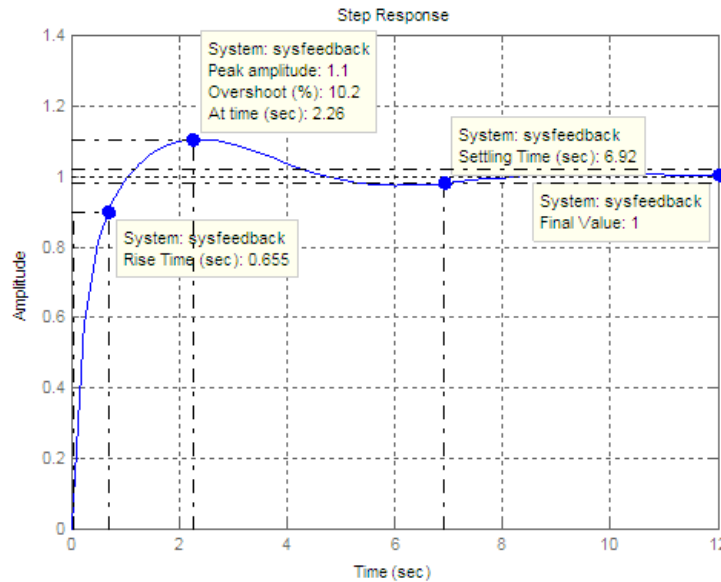
$$C(s) = k_p + \frac{k_i}{s} + k_d s = \frac{k_d s^2 + k_p s + k_i}{s} \text{----- (10)}$$

First assume that values for (kp, ki, kd) are equal to unity and then the TF of the PID controller is:

$$C(s) = \frac{s^2 + s + 1}{s} \text{----- (11)}$$

The system TF and closed loop Step response after a unity PID controller was injected with unity feedback will be:

$$TF5 = \frac{3.6 s^2 + 3.6 s + 3.6}{S^3 + 4.96 s^2 + 3.96 s + 3.6} \quad \text{----- (12)}$$



**Figure (5):** closed loop Step response after a unity PID controller

As shown in figure (5) that after entering the unity PID controller the system overshoot is decreased to 10.2% and the rise time is decreased to 0.655 and the settling time became 6.92 seconds.

By using the trial and error technique with table (1) the following proposed PID parameters were found to be equal to:

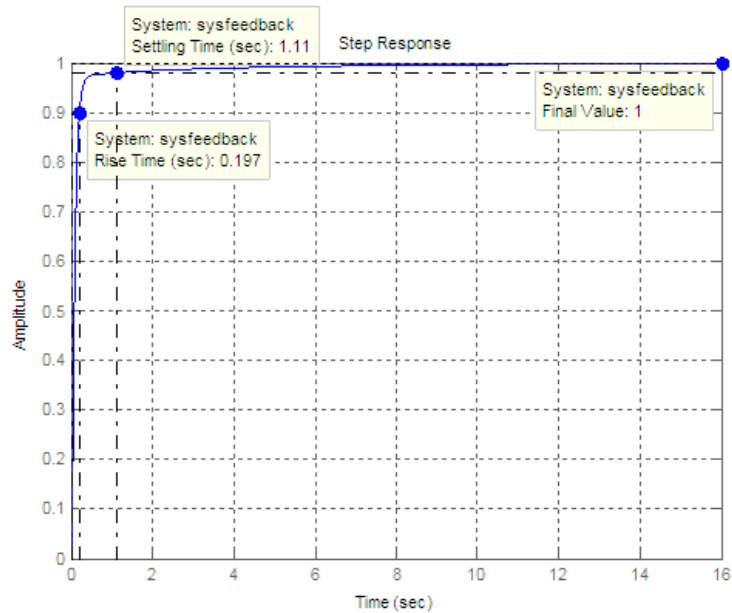
- Kp =0.5
- Ki =3.5
- Kd =3.5

Then the final PID transfer function will be:

$$C(s) = \frac{3.5 s^2 + 3.5 s + 0.5}{S} \quad \text{----- (13)}$$

The closed loop and step response for the final TF will be as shown in equation (14) and figure (6):

$$TF6 = \frac{12.6 s^2 + 12.6 s + 1.8}{S^3 + 13.96 s^2 + 12.96 s + 1.8} \quad \text{----- (14)}$$



**Figure (6):** closed loop Step response after tuning PID controller

### III. Results

The response of the spacecraft pitch system without a PID controller has an unstable performance as shown in figure (4). This requires entering the PID controller to improve its response characteristics and in this paper this is done by trial and error until the optimal PID parameter was found. When the optimal parameters were used with the controller the system has shown a better performance. From figure (4) and figure (5) it's seen that there was a big difference in performance before and after entering the PID controllers. Table (2) summarizes these results.

**Table (2):** The closed loop response characteristics with and without the PID controller

	RISE TIME	OVERSHOOT	SETTLING TIME
Closed Loop RESPONSE before PID controller	<b>0.702</b>	<b>31.8 %</b>	<b>5.55</b>
Closed Loop RESPONSE after tuning PID controller	<b>0.197</b>	<b>0%</b>	<b>1.11</b>

### IV. Conclusion

In this paper SOPDT method was used to convert the fourth order transfer function for the pitch spacecraft system into a second order system. The new second order system was used to design the suitable PID controller parameters to stabilize the pitch spacecraft system. The results obtained were consistent and meet the required condition.

### References

- [1]. <http://facultad.bayamon.inter.edu/arincon/pitch%20control%20system%20for%20high-altitude%20aircraft.pdf>.
- [2]. Katsuhiko Ogata, "Modern Control Engineering", Prentice-Hall, Inc., 1997 pp 60-63
- [3]. F. Haugen – 2010, "The Good Gain **method** for PI(D) controller tuning" Available at: [http://teachmean.com/publications/articles/good\\_gain\\_method/go](http://teachmean.com/publications/articles/good_gain_method/go)
- [4]. Hugh Jack, "Dynamic System Modelling and Control ", Copyright 1993-2003 Hugh Jack, pp 54-55.
- [5]. Richad C Dorf and Robert H. Bishop, Modern Control Systems, Pearson Education Inc, 2008.
- [6]. P.N Paraskevopoulos, Modern Control Engineering, Marcel Dekker Inc, 2002.