PID controller Simulator – V1.0

Introduction – PID controller is a name commonly given to a three-term controller. ***P*** stands for the proportional term, ***I*** for the integral term, and ***D*** for the derivative term in the controller [1]. PID controllers are probably the most widely used industrial controller. Even complex industrial control systems may comprise a control network whose main control building block is a PID control module [1].

Overview – This repository provides a MATLAB application (GUI) that can be used as an educational tool to understand, visualize, and test the concept of PID controllers in a real-life example. The application is inspired by the system built in [2]. In this tool, the PID controller will attempt to balance a green ball on a red moving bar. The position of the ball is measured, and the slope of the bar is modified (increased or decreased) in order to balance the green ball at a position defined by the user.

In the simulation provided here, the user can control:

- the initial position of the ball
- the initial slope of the bar
- the desired position at which the ball should stop
- the PID controller's gains, namely, Kp, Ki, and Kd gains that correspond to ***P***, ***I***, and ***D*** terms, respectively.

In Fig. 1, we show the red moving bar and the green ball. For this illustration, the initial position of the ball is set to 3 and the initial slope of the moving bar is set to 10 degrees. Once we run the program, the PID controller will balance the ball at the desired position (defined by the user).

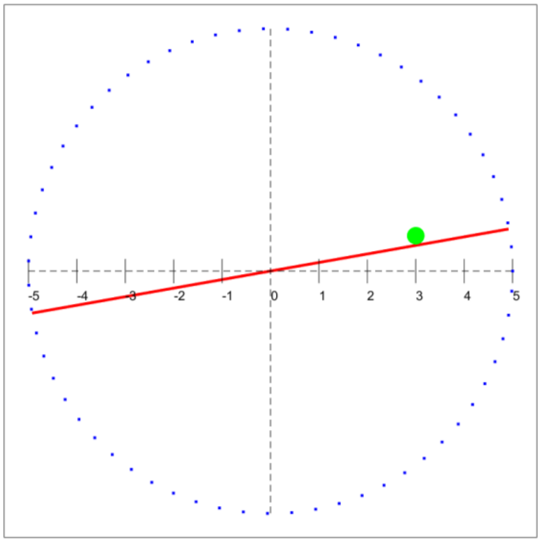

Fig. 1 Green ball and red moving bar

The ball's new position, $x(n+1)$, depends on the ball's previous position, $x(n)$, and step size, $S(n)$, which is a function of the slope of the moving bar. The ball's new position can be expressed as follows:

$$x(n+1) = x(n) + S(n) \qquad (1)$$

$$S(n) = \begin{cases} -3m(n)^2, & if\ m(n) \geq 0 \\ +3m(n)^2, & if\ m(n) < 0 \end{cases} \qquad (2)$$

$S(n)$ is shown in Fig. 2. For instance, if the slope of the bar is 10 degrees and the current ball position is 2, then the new ball position is 2 + 0.1 = 2.1.

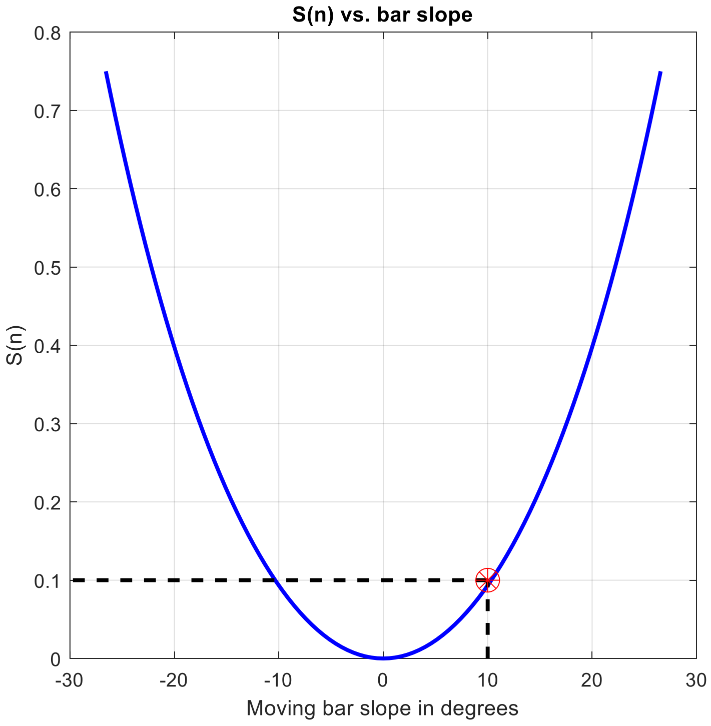

Fig. 2 Ball position variation as function of the moving bar's slope

Understanding PID Controller – We will now try to understand how the PID controller updates the slope of the moving bar in order to balance the ball at the desired position. We will assume that the desired position of the ball, D_0, is set to 0 and that the PID controller will update the slope of the bar so that the ball is balanced at D_0.

Proportion term: To better understand the proportion term, let's set $k_p = c_1$, $k_i = 0$, and $k_d = 0$. At each iteration, the controller will measure the distance between the ball and the D_0, denoted by Δ_0. We will assume that the initial slope of the bar is 10 degrees. If Δ_0 is positive, this implied that the ball is to the right of D_0. Therefore, the k_p will update the slope value of the bar in order to move the ball by value c_1 to the left. This will result in a smaller Δ_0. Since the proportional value is fixed and does not adaptively change, Δ_0 will fluctuate. This could typically cause the ball to oscillate around the desired position. The reason for the oscillation is that, if the ball is to the right of D_0, then the controller will force it to the left. Now that the ball is to the left of D_0, the controller will force it back to the right. Please note that the proportion term only reacts to the distance of the ball with respect to the desired position. It will not react to the ball's velocity or acceleration.

Derivative term: As we saw above that using the proportion term could cause the system to oscillate around the desired position but will not balance the ball. To resolve this matter, we introduce a derivative term to the controller. First, let's understand the effect that the derivative term has on the controller. We assume that $k_p = 0$, $k_i = 0$, and $k_d = c_2$.

When we only use the derivative term to control the system, the system will then measure the velocity of the ball and tries to set it to zero. In this case, the controller will keep on modifying the value of the slope of the bar until the velocity of the ball is zero. An artifact that could result when we only use the derivative term is that the system might balance the ball at a wrong desirable position. This is because the controller only aims to bring the ball's velocity to zero regardless of where it stops. To solve the problem, we should introduce the proportion term into the control. By doing that, we will ensure that the ball will balance and stop as close as possible to the desired position.

Proportion and derivative terms will usually suffice for this example if we are okay with the ball balancing close to the desired position. However, when the ball is too close to the desired position (but not fully centered at the desired position) the system will not be affected by the proportion term (since the ball is very close to the desired point) nor the derivative term (since the velocity of the ball will be zero at this point). If our application requires a high level of precision, then introducing the integration term will solve the problem.

Integration term: The integration term adds up the error values over a time period. The more time elapses, the bigger the error measured by the integration term gets (if it was not corrected by the controller). In the example where the ball very stops close to the desired position (but not exactly on it), the proportion and derivative terms will remain unchanged. However, since the system is still measuring a small error, the error introduced by the integral term will slowly increase with time. To compensate for this error (introduced by the integration term) the controller will slowly update the slope of the bar so that the ball is fully centered at the desired position.

Now that we understand the PID controller concept, let’s use the application in order to test different values of proportion, derivation, and integration gains in order to balance the ball at a set desired position.

Reference:

[1] Crowe, J., Chen, G. R., Ferdous, R., Greenwood, D. R., Grimble, M. J., Huang, H. P., ... & Zhang, Y. (2005). *PID control: new identification and design methods*. Springer-Verlag London Limited.

[2] Electronoobs. (2019, July 20). *PID Balance+Ball | full explanation & tuning* [Video]. YouTube. https://www.youtube.com/watch?v=JFTJ2SS4xyA