

Programación en Matlab

Control PID

Ing. Eddie Ángel

Sobrado Malpartida

Introducción

En esta guía entenderemos el efecto de los parámetros del control PID: K_p , T_i , T_d .

Recordando que el PID expresado en FT es:

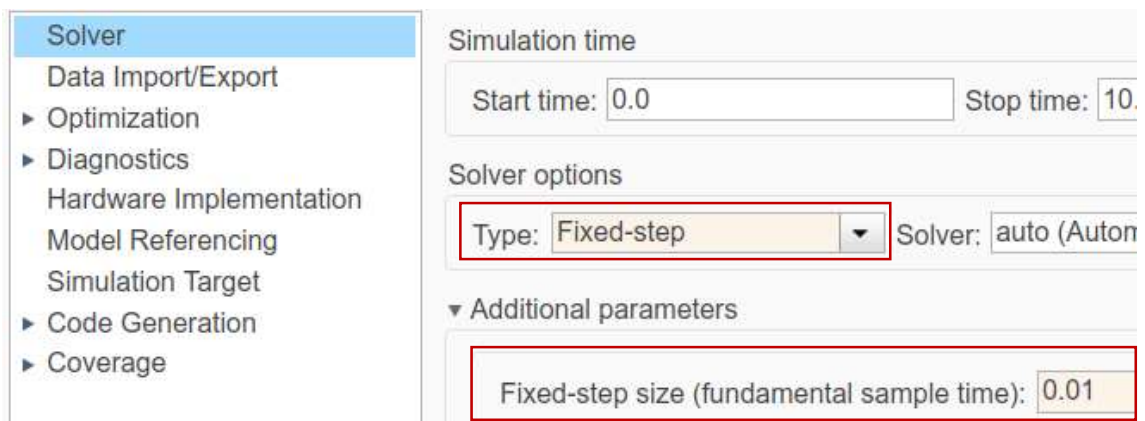
$$G(s) = k_p \left(1 + \frac{1}{T_i s} + T_d s \right)$$

Antes de emplear SIMULINK sería bueno modificar los pasos de simulación para tener una respuesta mas fina:

Presionar: 



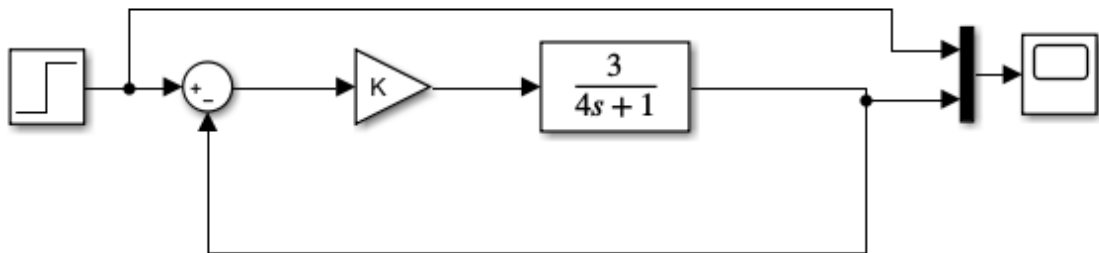
Modificar los parámetro indicados:



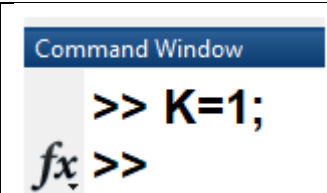
Control Proporcional: K

$$G(s) = kp$$

1. Implementar el siguiente programa:



Ir modificando el valor de K desde la línea de comandos antes de ejecutar el programa de SIMULINK

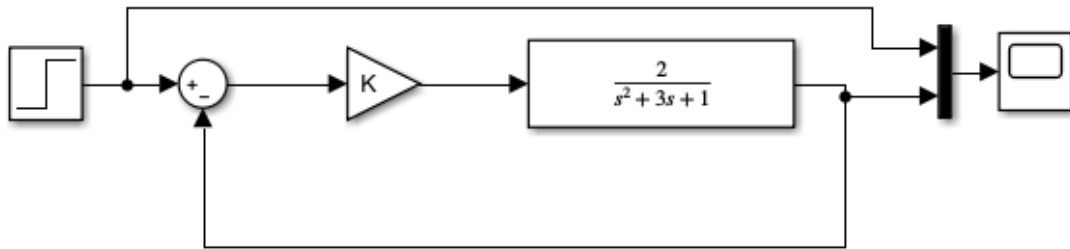
	Modificar: K=1, 3, 5, 7
--	----------------------------

Conclusión

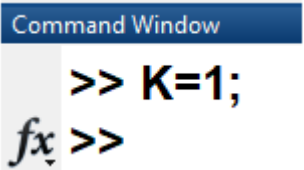
Observa la respuesta en el tiempo, y cada vez notara:

- A mayor ganancia el sistema **responderá más rápido**, es decir el tiempo de establecimiento es menor.
- A mayor ganancia de alguna manera se **reduce el error** en estado estable, pero **no lo elimina**

2. Ahora implementar el siguiente programa:



Ir modificando el valor de K desde la línea de comandos antes de ejecutar el programa de SIMULINK

	Modificar: K=1, 3, 5, 7, 10
---	--------------------------------

Conclusión

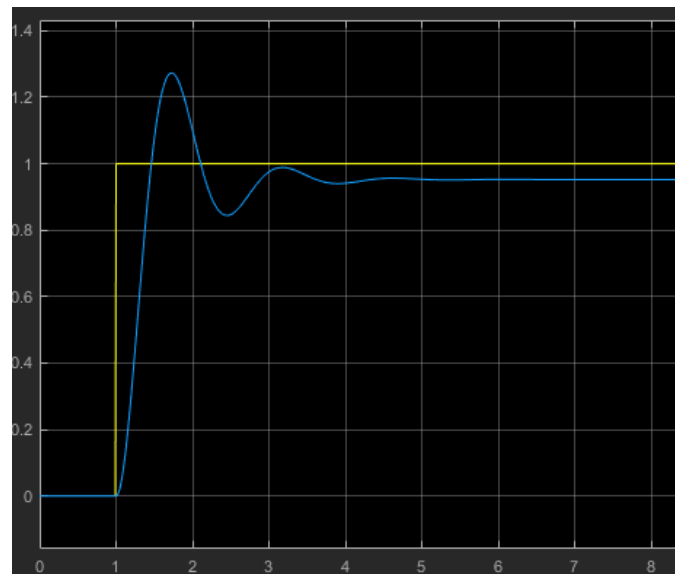
Observa la respuesta en el tiempo, y cada vez notara:

- A mayor ganancia el sistema **responderá más rápido**, es decir el tiempo de establecimiento es menor.
- A mayor ganancia de alguna manera se **reduce el error** en estado estable, pero **no lo elimina**
- Ahora también notamos que a medida que la ganancia aumente, se **genera sobre impulsos** u **oscilaciones** y en algunos casos podría llegar a la **inestabilidad**

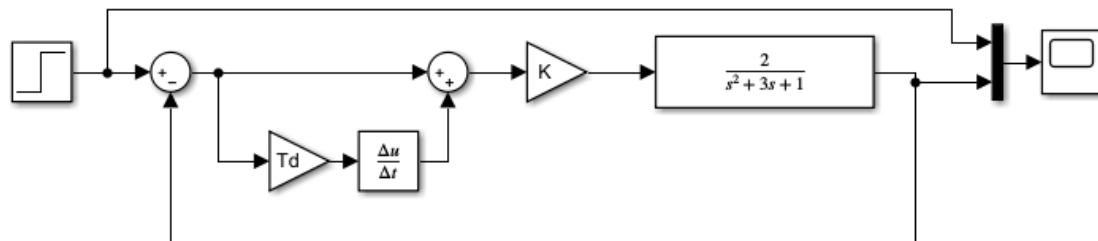
Control Proporcional Derivativo: **PD**

$$G(s) = kp(1 + T_d s)$$

Del programa anterior para un **K=10** se obtuvo:



Implementar el siguiente programa:



Manteniendo un valor de K=10 ir modificando valores de **Td** solamente desde la línea de comandos antes de ejecutar el programa de SIMULINK

Command Window	Modificar:
<code>fx >> Td=0.1;</code>	Td = 0.1, 0.3, 0.5

Conclusión

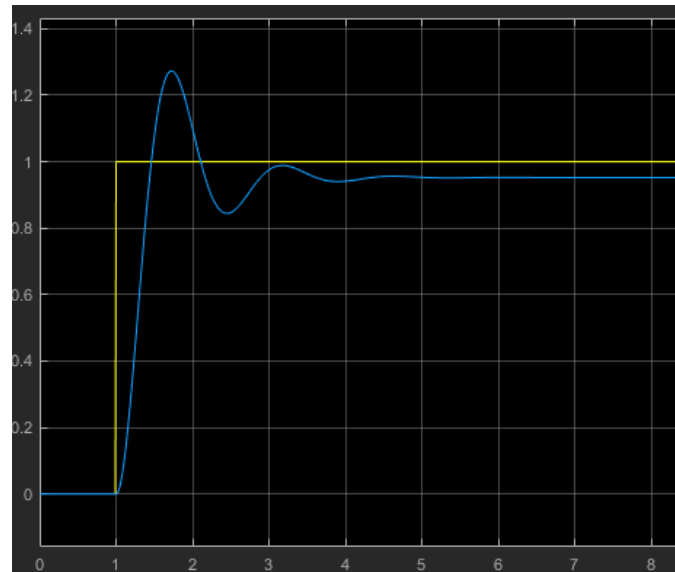
Observa la respuesta en el tiempo, y cada vez notara:

- a. El Td **atenúa** sobre impulso u oscilación.
- b. A mayor Td se logra atenuar más el sobre impulso u oscilación
- c. El error en estado estable no se ve afectado por el Td
- d. Normalmente **no se recomienda** usar un Td en procesos que tiene ruido

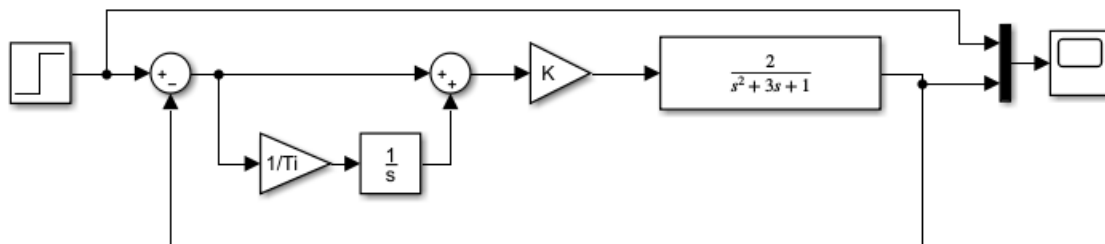
Control Proporcional Integral: **PI**

$$G(s) = kp \left(1 + \frac{1}{T_i s} \right)$$

Del primer programa para un **K=10** se obtuvo:



Implementar el siguiente programa:



Manteniendo un valor de $K=10$ ir modificando valores de **Ti** solamente desde la línea de comandos antes de ejecutar el programa de SIMULINK

Command Window	Modificar: Ti = 10, 3, 1, 0.5
<pre>>> Ti=10;</pre>	

Conclusión

Observa la respuesta en el tiempo, y cada vez notara:

- a. Con el T_i se logra $\text{error}=0$
- b. Si el T_i es grande la respuesta **es lenta** por ello demora en lograr error cero
- c. A medida que el T_i disminuye, la respuesta se hace menos lento, pero empieza a incrementarse el sobre impulso
- d. Si seguimos disminuyendo T_i , se **genera más sobre impulso, oscilaciones** y en algunos casos podría llegar a la **inestabilidad**