

Diseño de controladores

Entrega 3

Jaime Pérez Sánchez

Sergio Pérez Morillo

UPM – ETSIT – Sistemas electrónicos de control (SECO)
Curso 2017-2018

Índice

1. Diferencias entre controladores desde la perspectiva del seguimiento de señales de referencia monómicas y el comportamiento en régimen transitorio	2
Introducción	2
Controlador P	2
Controlador PD	3
Controlador P-D	5
Controlador PI	6
Controlador PID	8
Controlador PI-D	10
Controlador PID-D	12
Controlador D PID	14
2. Análisis, diseño e implementación de un sistema de control de posición angular de un motor DC utilizando un controlador D PID	17
Introducción	17
Análisis de estabilidad	17
Sintonización de parámetros	21
Experimento en Telelaboratorio	23
3. Análisis, diseño e implementación de un sistema de control de posición angular de un motor DC utilizando un controlador PID, sintonizado con el método de Ziegler-Nichols Modificado	26
Introducción	26
Análisis del dominio de la frecuencia	26
Diseño del controlador mediante el método de Ziegler-Nichols Modificado	28
Experimento en Telelaboratorio	32
4. Conclusiones	33
5. Bibliografía	35

1. Diferencias entre controladores desde la perspectiva del seguimiento de señales de referencia monómicas y el comportamiento en régimen transitorio

Introducción

Los sistemas de control son conjuntos de dispositivos encargados de administrar y regular el comportamiento de otros sistemas, con el objetivo de obtener una salida deseada reduciendo las probabilidades de fallo. Además estos controladores deben ser estables y robustos frente a perturbaciones y errores.

En este apartado vamos a estudiar las diferencias entre los distintos controladores vistos en el aula en el ámbito de seguimiento de señales monómicas y su comportamiento en el régimen transitorio.

Utilizaremos como ejemplo a controlar la función simplificada del motor DC:

$$G(s) = \frac{K}{s(s + p)}$$

Dando los siguientes valores a las constantes: $K = 2500$, $p = 68$

Controlador P

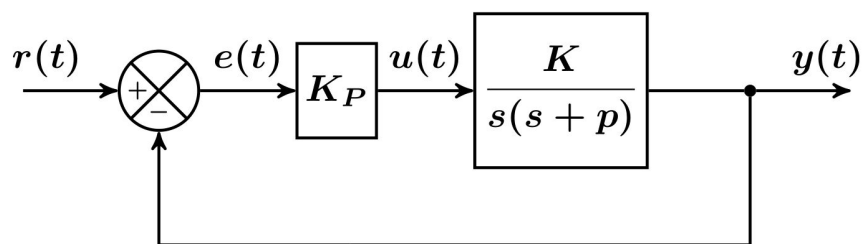


Figura 1.1: Controlador P para un sistema continuo de segundo orden y tipo uno

En este controlador la señal de salida es proporcional (P) a la señal de error con la que se realimenta el sistema. Dicha señal error se define como la diferencia entre la señal de salida actual y la de referencia.

La función de transferencia de lazo cerrado es:

$$H_P(s) = \frac{K_p K}{s^2 + ps + K_p K}$$

En el documento de diseño de controladores proporcionado en la asignatura **[1.1]** se demuestra que este controlador tendrá un error de seguimiento nulo para señales monómicas de grado menor que 1, es decir la señal escalón.

Simulando el sistema en Matlab, con la señal escalón como estímulo, obtenemos la siguiente gráfica:

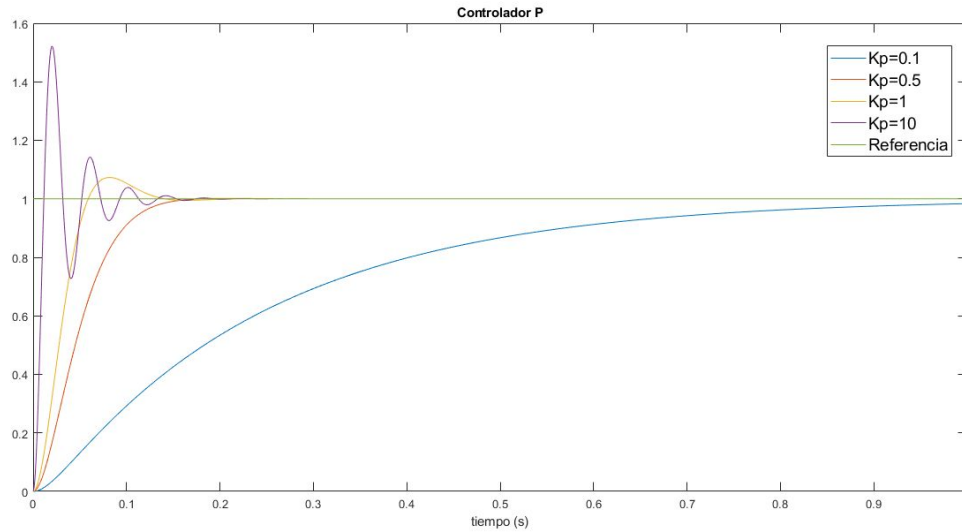


Figura 1.2: Salida con controlador P variando K_p

Podemos observar que a medida que K_p aumenta, disminuyen el tiempo de establecimiento (t_s) y el tiempo de subida (t_r). Pero si aumenta demasiado su valor comienzan a aparecer oscilaciones y aumenta la sobreelongación máxima (M_p).

Controlador PD

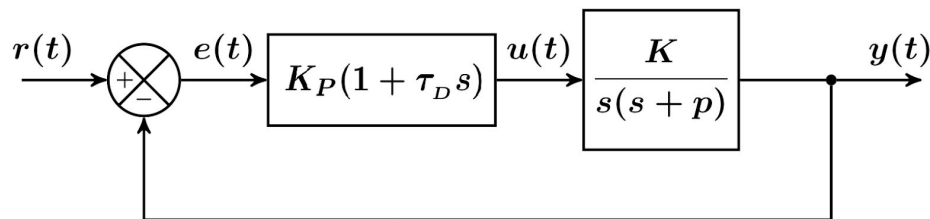


Figura 1.3: Controlador PD para un sistema continuo de segundo orden y tipo uno

Este controlador se opone a desviaciones de la señal de entrada, con una respuesta proporcional a la derivada de éstas, es decir, proporcional a la rapidez con la que varían.

Es una acción predictiva ya que actúa antes de que la magnitud del error se vuelva demasiado grande.

La función de transferencia de lazo cerrado es:

$$H_{PD}(s) = \frac{K_p K (1 + \tau_D s)}{s^2 + (p + K_p K \tau_D) s + K_p K}$$

En el documento de diseño de controladores proporcionado en la asignatura **[1.2]** se demuestra que este controlador tendrá un error de seguimiento nulo para señales monómicas de grado menor que 1, es decir la señal escalón.

Simulando el sistema en Matlab, con la señal escalón como estímulo, obtenemos las siguientes gráficas:

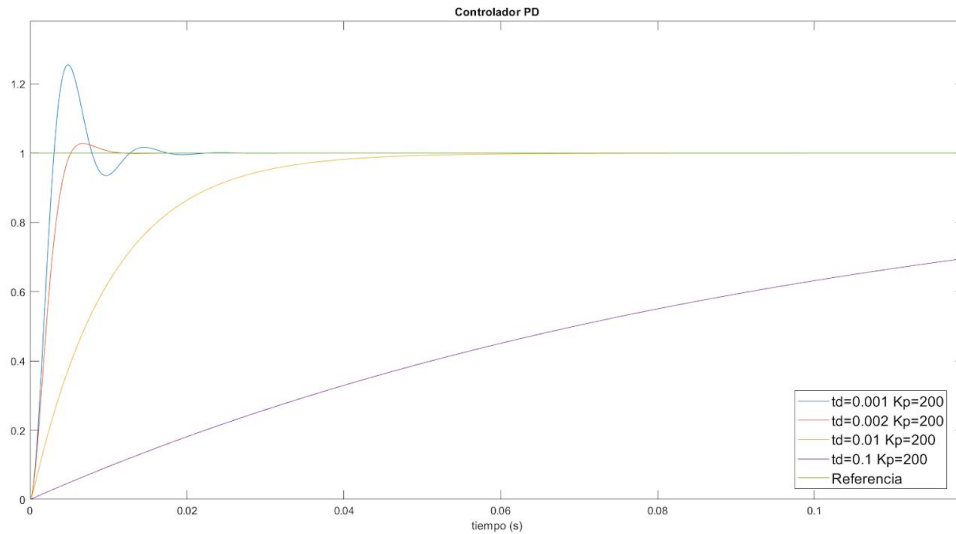


Figura 1.4: Salida con controlador PD variando τ_D

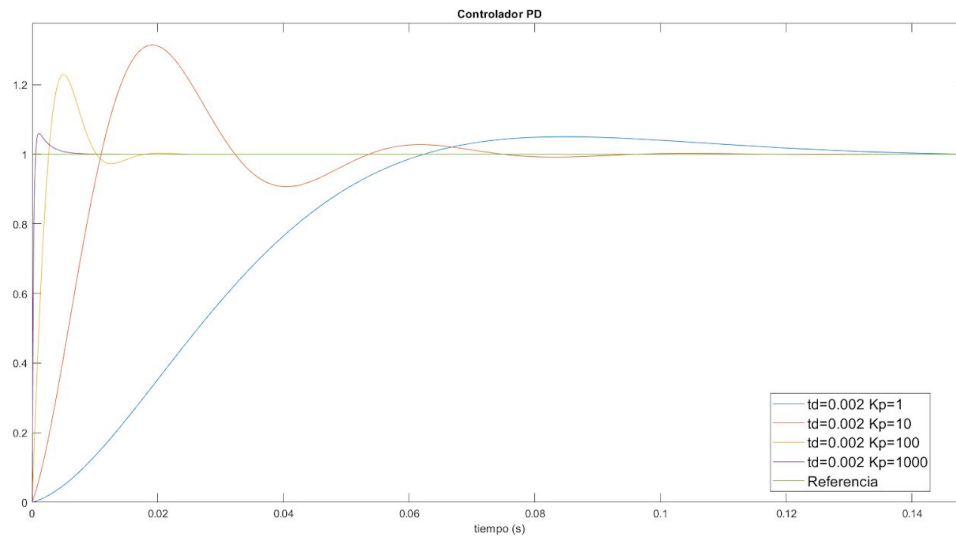


Figura 1.5: Salida con controlador PD variando K_p

Podemos observar que si aumenta K_p , disminuyen el tiempo de establecimiento (t_s) y el tiempo de subida (t_r).

Si K_p tiene un valor mucho mayor que τ_D , al igual que en el controlador P, comienzan a aparecer oscilaciones y aumenta la sobreelongación máxima (M_p).

La función del factor derivativo τ_D es eliminar las oscilaciones. Cuanto mayor es el valor de τ_D , más oscilaciones compensaremos, pero a cambio el sistema será más lento a la hora de alcanzar la referencia.

Podemos ver también que este factor derivativo no soluciona de manera suficientemente rápida el error en régimen permanente. En un sistema real tendríamos un error constante que seríamos incapaces de corregir con este controlador.

Controlador P-D

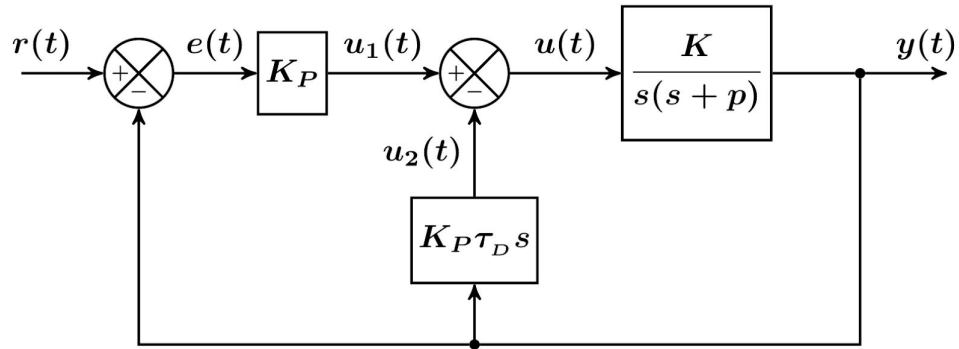


Figura 1.6: Controlador P-D para un sistema continuo de segundo orden y tipo uno

Este controlador se comporta de forma similar al de tipo PD, excepto en que el factor derivativo (D) se encuentra en la rama paralela en lugar de en la rama directa.

La función de transferencia de lazo cerrado es:

$$H_{P-D}(s) = \frac{K_p K}{s^2 + (p + K_p K \tau_D)s + K_p K}$$

Al igual que el controlador PD, en el documento de diseño de controladores proporcionado en la asignatura **[1.3]** se demuestra que tendrá un error nulo para el seguimiento de señales monómicas de grado menor que 1, es decir, la señal escalón.

Simulando el sistema en Matlab, con la señal escalón como estímulo, obtenemos las siguientes gráficas:

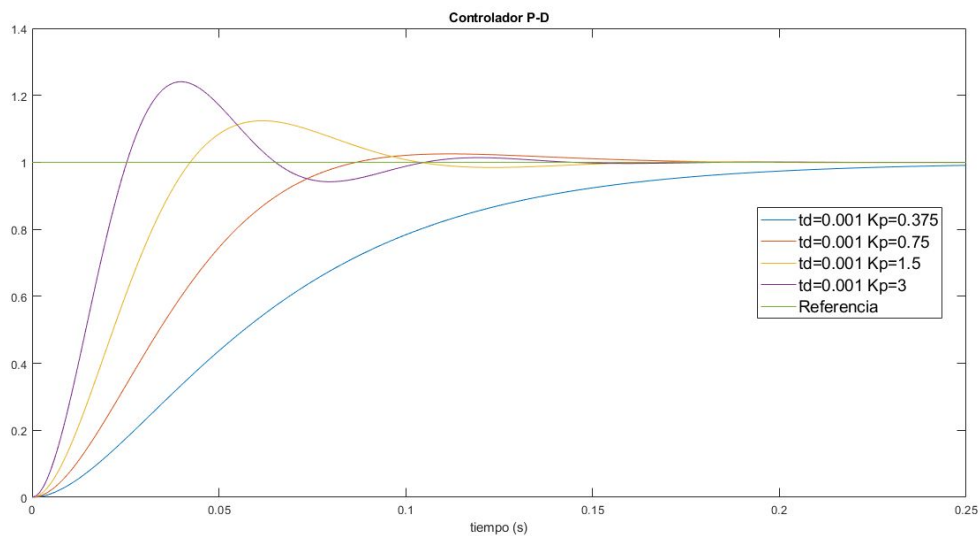


Figura 1.7: Salida con controlador P-D variando K_p

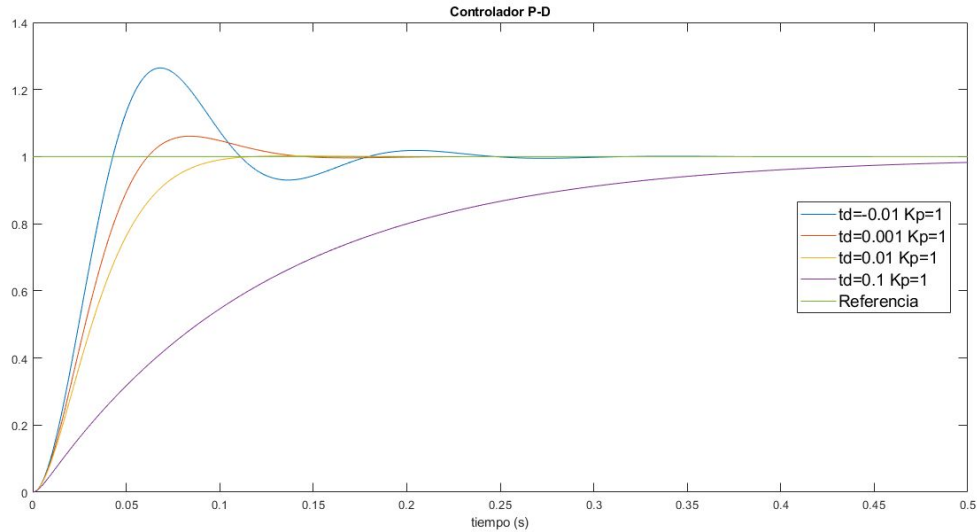


Figura 1.8: Salida con controlador P-D variando τ_D

Podemos observar se comporta de manera muy similar al controlador PD. Por lo que extraemos las mismas conclusiones que en el punto anterior.

Controlador PI

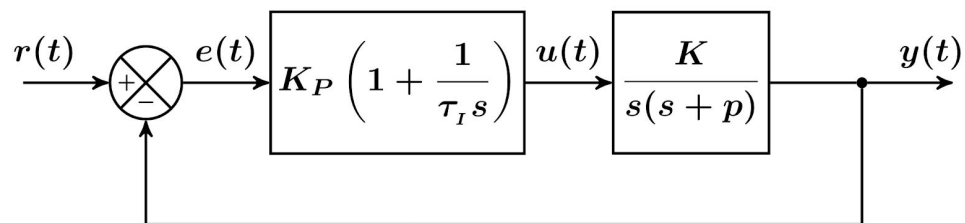


Figura 1.9: Controlador PI para un sistema continuo de segundo orden y tipo uno

Este controlador tiene como objetivo disminuir el error en régimen estacionario provocado por el factor proporcional (P). Para ello el factor integral (I) suma el error integrado durante un periodo determinado de tiempo.

La respuesta del factor proporcional (P) es instantánea, sin embargo el factor integral (I) actúa con cierto retardo.

La función de transferencia de lazo cerrado es:

$$H_{PI}(s) = \frac{K_p K \left(s + \frac{1}{\tau_I} \right)}{s^2(s + p) + K_p K \left(s + \frac{1}{\tau_I} \right)}$$

Según se demuestra en el documento de diseño de controladores proporcionado en la asignatura [1.4], este controlador podrá resolver con error nulo el seguimiento de las señales monómicas de grado menor que 2, es decir, la señal escalón y rampa.

Simulando el sistema en Matlab, con la señal escalón como estímulo, obtenemos las siguientes gráficas:

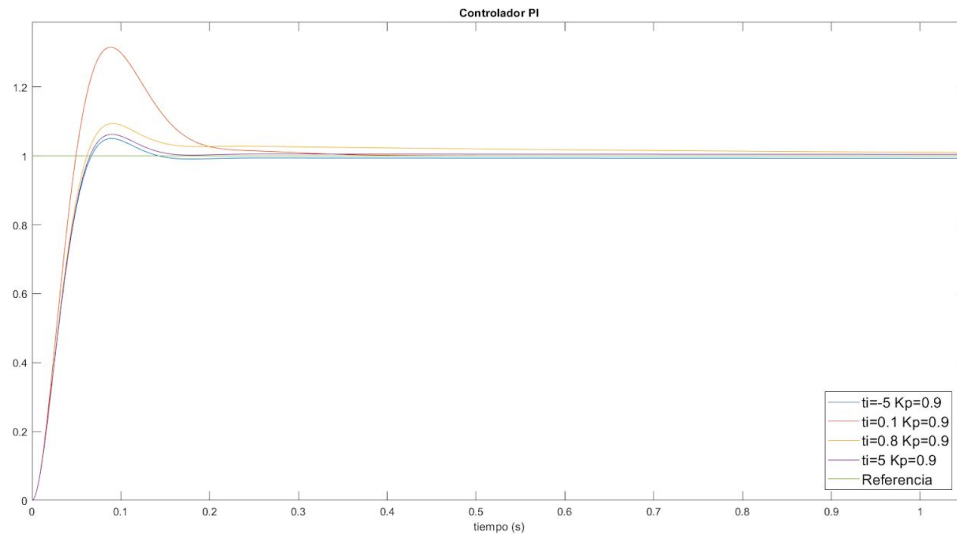


Figura 1.10: Salida con controlador PI variando τ_i

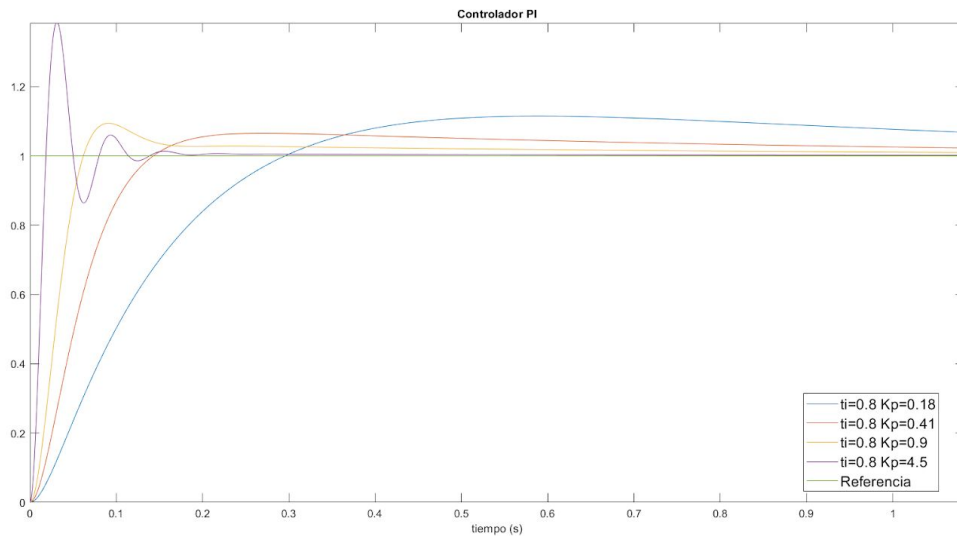


Figura 1.11: Salida con controlador PI variando K_p

Podemos observar que si aumenta K_p , disminuyen el tiempo de establecimiento (t_s) y el tiempo de subida (t_r). Pero si aumentamos demasiado, comienzan a aparecer oscilaciones y aumenta la sobreelongsación máxima (M_p).

La función del factor integral τ_i es eliminar el error en régimen permanente provocado por el controlador P. Cuanto menor sea τ_i más rápido se eliminará el error, pero si lo disminuimos demasiado, el sistema se vuelve más inestable y comienzan a aparecer oscilaciones.

Este factor integral, soluciona el error en régimen permanente pero no las oscilaciones del régimen transitorio. La solución es, como se verá a continuación, introducir los tres factores (Proporcional, Integral y Derivativo) en el controlador para obtener un PID.

Controlador PID

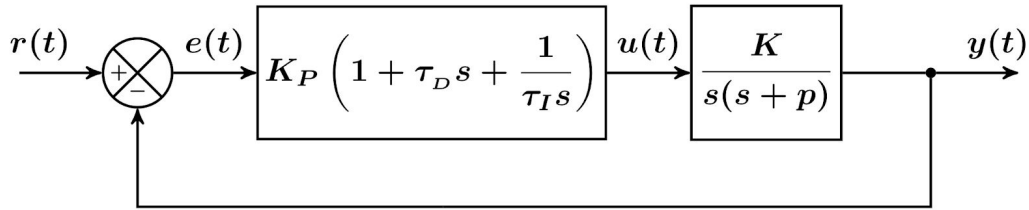


Figura 1.12: Controlador PID

Este controlador intenta aprovechar las ventajas de los controladores de acciones básicas explicados anteriormente.

Si la señal de error varía rápidamente, predomina la acción derivativa. En cambio, si varía de forma lenta o es constante, predomina la acción integral.

Este controlador ofrece una respuesta muy rápida y una compensación del error casi inmediata en caso de perturbaciones. Sin embargo, es propenso a oscilar y la dificultad para ajustar los parámetros es mucho mayor que en los casos anteriores.

La función de transferencia de lazo cerrado es:

$$H_{PID}(s) = \frac{K K_P \tau_D \left(s^2 + \frac{s}{\tau_D} + \frac{1}{\tau_D \tau_I} \right)}{s^2(s+p) + K K_P \tau_D \left(s^2 + \frac{s}{\tau_D} + \frac{1}{\tau_D \tau_I} \right)}$$

Según se demuestra en el documento de diseño de controladores proporcionado en la asignatura **[1.5]**, este controlador podrá resolver con error nulo el seguimiento de las señales monómicas de grado menor que 2, es decir, la señal escalón y rampa.

Simulando el sistema en Matlab, con la señal escalón como estímulo, obtenemos las siguientes gráficas:

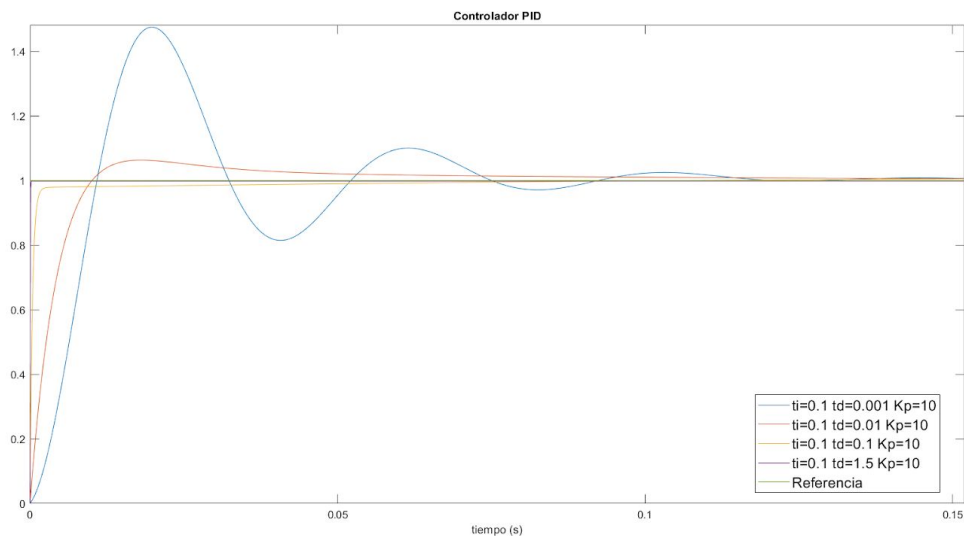


Figura 1.13: Salida con controlador PID variando τ_D

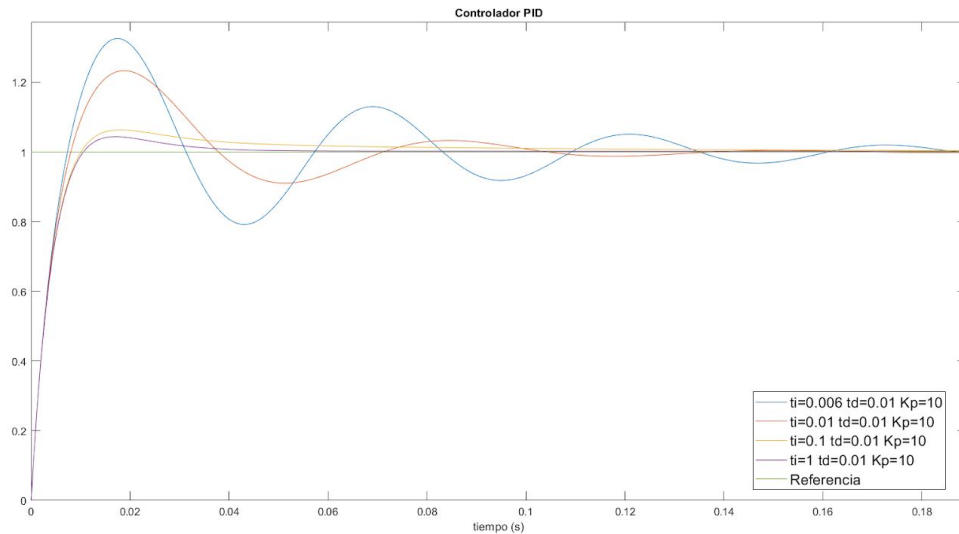


Figura 1.14: Salida con controlador PID variando τ_i

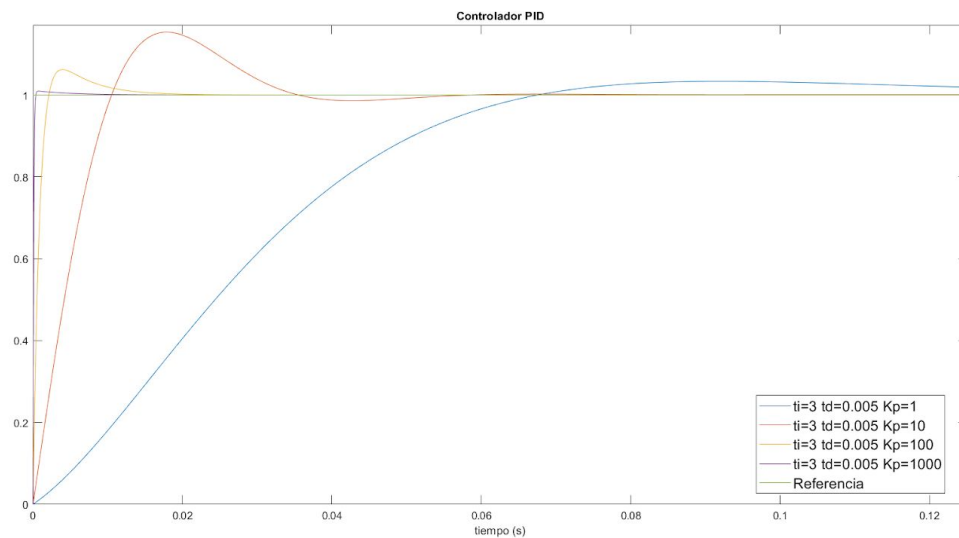


Figura 1.15: Salida con controlador PID variando K_p

Como podemos observar, y ya hemos deducido de los controladores anteriores:

A medida que K_p aumenta su valor, disminuyen el tiempo de establecimiento (t_s) y el tiempo de subida (t_r). Pero también comienzan a aparecer oscilaciones y aumenta la sobreelongación máxima (M_p).

El factor integral τ_i elimina el error en régimen permanente. Cuanto menor sea τ_i más rápido se eliminará el error, pero si lo disminuimos demasiado, el sistema se vuelve más inestable y comienzan a aparecer oscilaciones.

El factor derivativo τ_d elimina las oscilaciones. Cuanto mayor es el valor de τ_d , más oscilaciones compensaremos, pero a cambio el sistema será más lento a la hora de alcanzar la referencia.

Controlador PI-D

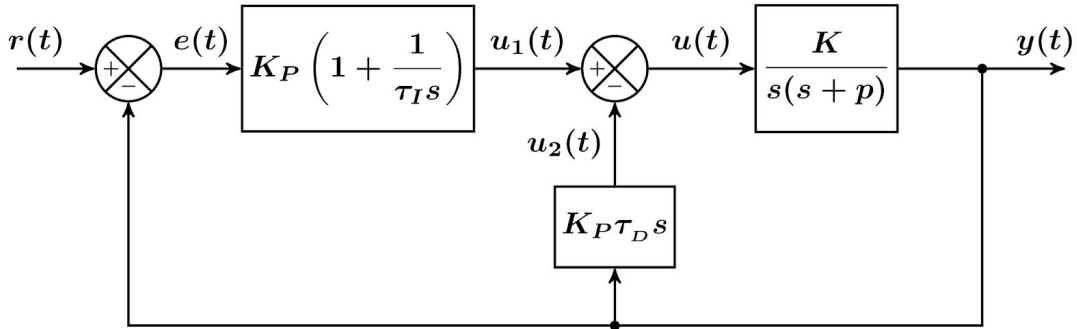


Figura 1.16: Controlador PI-D

Este controlador se comporta de forma similar al de tipo PID, excepto en que el factor derivativo (D) se encuentra en la rama paralela en lugar de en la rama directa. La función de transferencia de lazo cerrado es:

$$H_{PI-D}(s) = \frac{KK_P \left(s + \frac{1}{\tau_I} \right)}{s^2(s+p) + KK_P \tau_D \left(s^2 + \frac{s}{\tau_D} + \frac{1}{\tau_D \tau_I} \right)}$$

Al igual que el controlador PID, en el documento de diseño de controladores proporcionado en la asignatura **[1.5]** se demuestra que tendrá un error nulo para el seguimiento de señales monómicas de grado menor que 2, es decir, la señal escalón y rampa.

Simulando el sistema en Matlab, con la señal escalón como estímulo, obtenemos las siguientes gráficas:

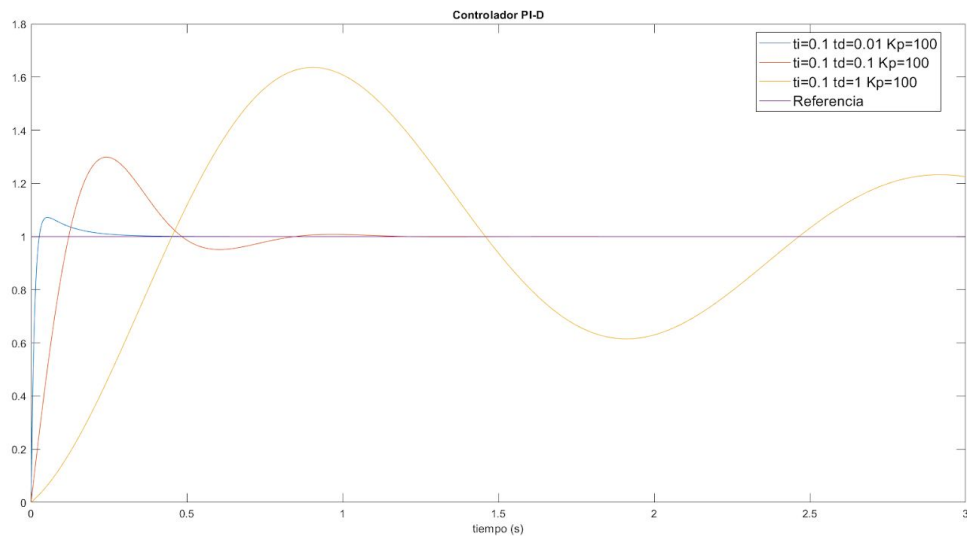


Figura 1.17: Salida con controlador PI-D variando τ_D

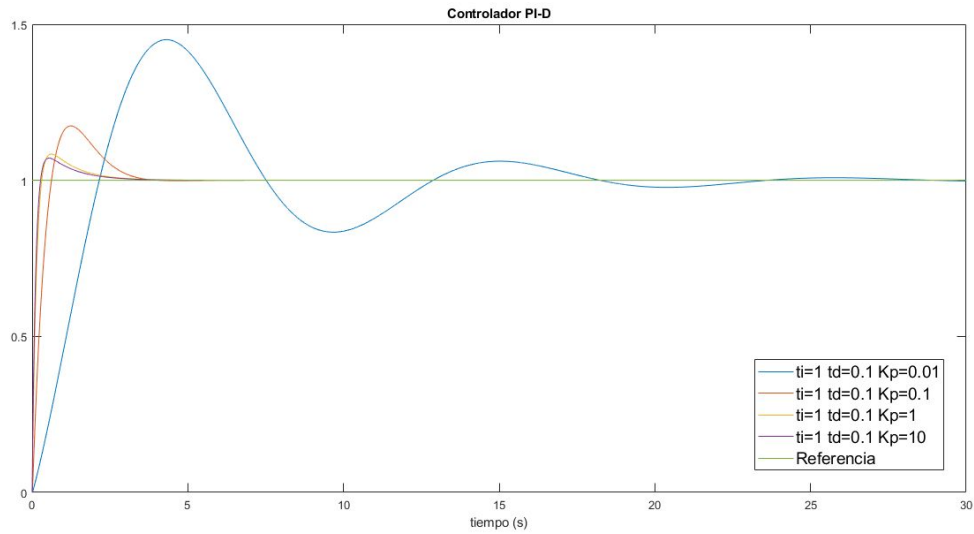


Figura 1.18: Salida con controlador PI-D variando K_p

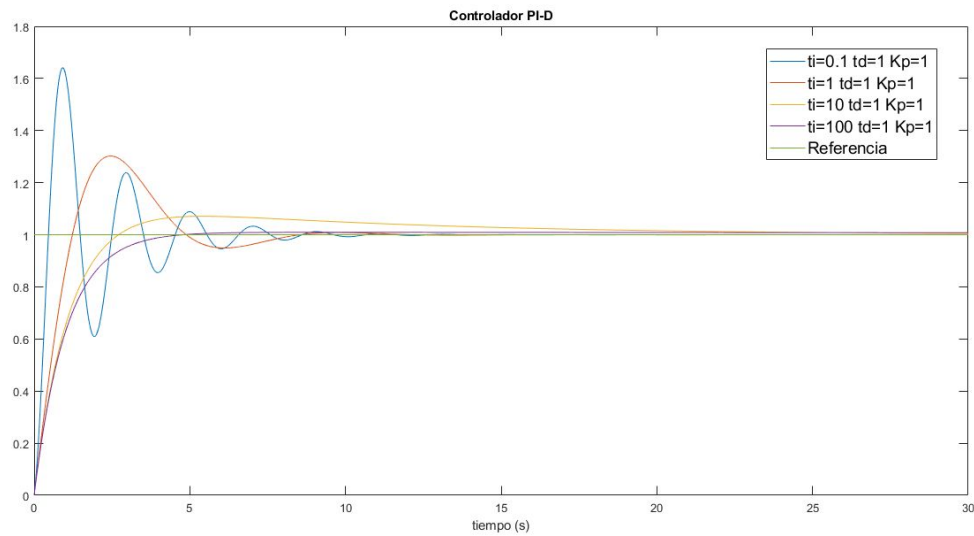


Figura 1.19: Salida con controlador PI-D variando t_i

Podemos observar se comporta de manera muy similar al controlador PID. Excepto por que el factor derivativo al estar en la rama paralela, infiere de forma opuesta que en el caso del PID. Es decir, a mayor valor de τ_D , más oscilaciones aparecen. Del resto de parámetros extraemos las mismas conclusiones que en el punto anterior.

Controlador PID-D

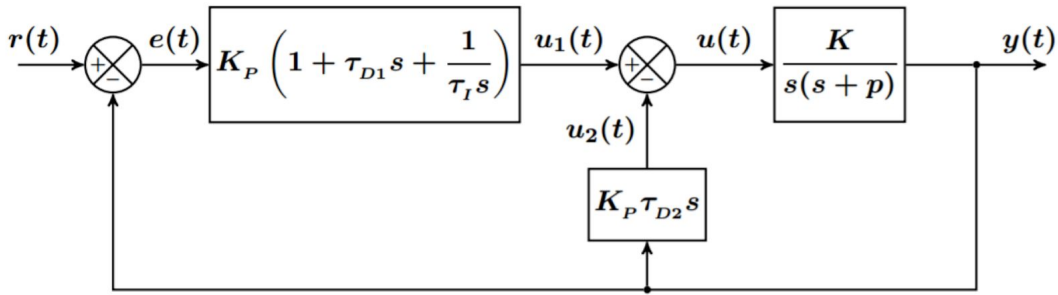


Figura 1.20: Controlador PID-D

Este controlador es un PID al que se le ha añadido un factor derivativo (D) extra en la rama paralela. Con ello se pretende solucionar el problema de supresión de una señal de perturbación constante.

La función de transferencia de lazo cerrado es:

$$H_{PID-D}(s) = \frac{K K_P \tau_{D1} \left(s^2 + \frac{s}{\tau_{D1}} + \frac{1}{\tau_{D1} \tau_I} \right)}{s^2(s+p) + K K_P \tilde{\tau}_D \left(s^2 + \frac{s}{\tilde{\tau}_D} + \frac{1}{\tilde{\tau}_D \tau_I} \right)}$$

Donde:

$$\tilde{\tau}_D = \tau_{D1} + \tau_{D2}$$

Según se demuestra en el documento de diseño de controladores proporcionado en la asignatura **[1.6]**, este controlador podrá resolver con error nulo el seguimiento de las señales monómicas de grado menor que 3, es decir, las señales escalón, rampa y parábola.

Simulando el sistema en Matlab, con la señal escalón como estímulo, obtenemos las siguientes gráficas:

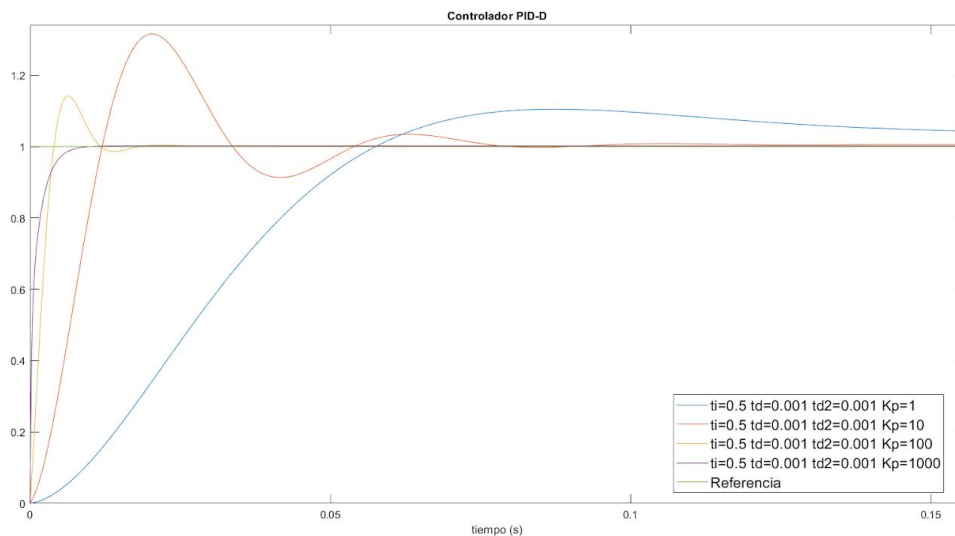


Figura 1.21: Salida con controlador PID-D variando K_p

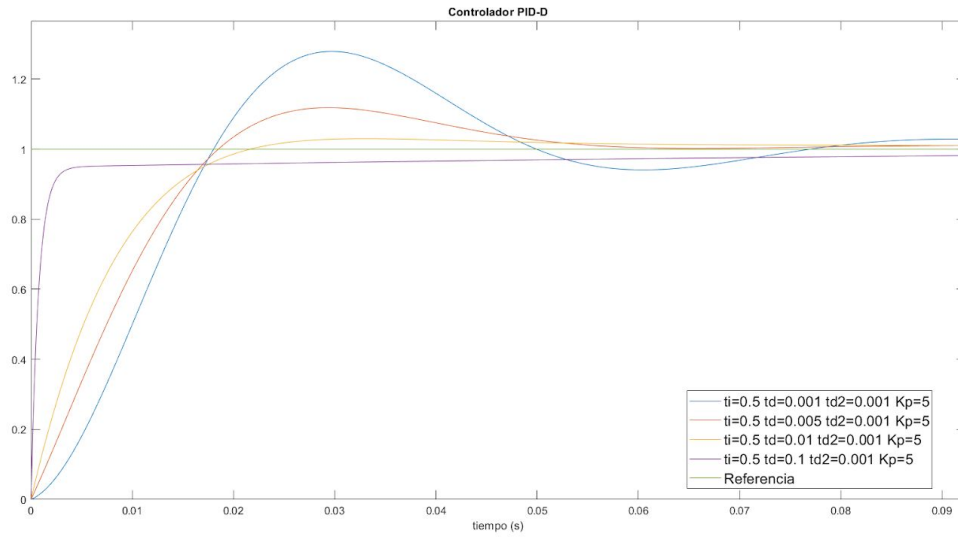


Figura 1.22: Salida con controlador PID-D variando τ_{D1}

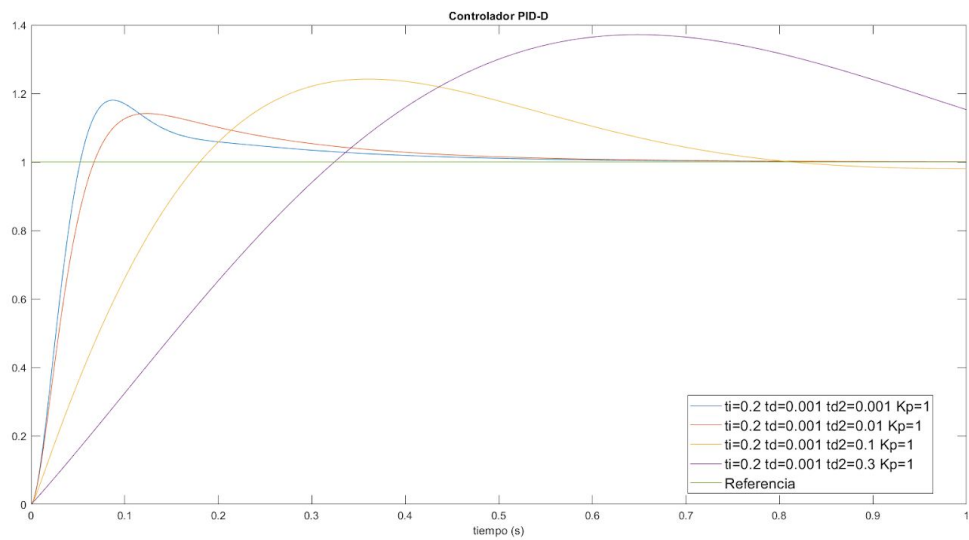


Figura 1.23: Salida con controlador PID-D variando τ_{D2}

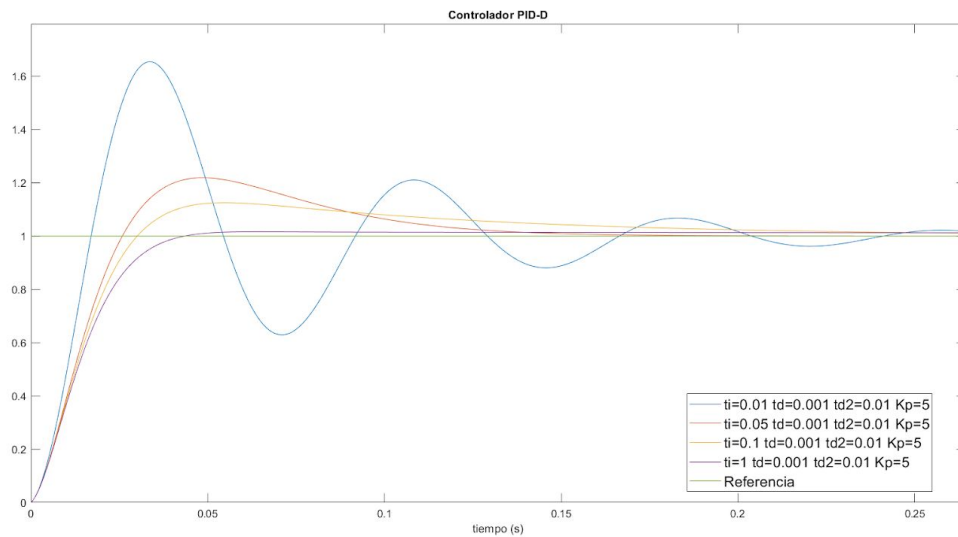


Figura 1.24: Salida con controlador PID-D variando τ_I

Podemos observar que la rama directa del controlador se comporta igual que en el caso del PID, y la rama paralela se comporta igual que la misma rama del caso PI-D. Es decir, el factor derivativo de la rama directa y de la rama paralela se comportan de forma inversa.

Controlador D|PID

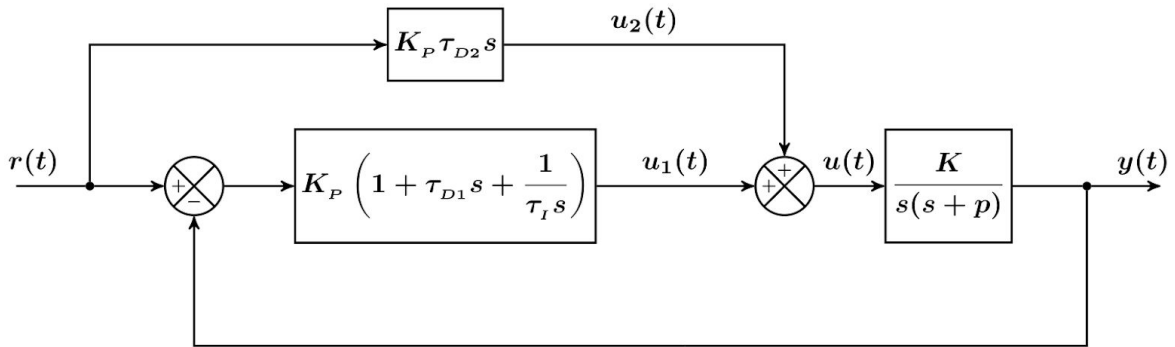


Figura 1.25: Controlador D|PID

Este controlador es un PID al que se le ha añadido un factor derivativo (D) extra en la rama prealimentada. Con ello se pretende solucionar el problema de supresión de una señal de perturbación constante.

La función de transferencia de lazo cerrado es:

$$H_{D|PID}(s) = \frac{K K_P \tilde{\tau}_D \left(s^2 + \frac{s}{\tilde{\tau}_D} + \frac{1}{\tilde{\tau}_D \tau_I} \right)}{s^2(s + p) + K K_P \tau_{D1} \left(s^2 + \frac{s}{\tau_{D1}} + \frac{1}{\tau_{D1} \tau_I} \right)}$$

Donde:

$$\tilde{\tau}_D = \tau_{D1} + \tau_{D2}$$

Según se demuestra en el documento de diseño de controladores proporcionado en la asignatura **[1.7]**, este controlador podrá resolver con error nulo el seguimiento de las señales monómicas de grado menor que 3, es decir, las señales escalón, rampa y parábola.

Simulando el sistema en Matlab, con la señal escalón como estímulo, obtenemos las siguientes gráficas:

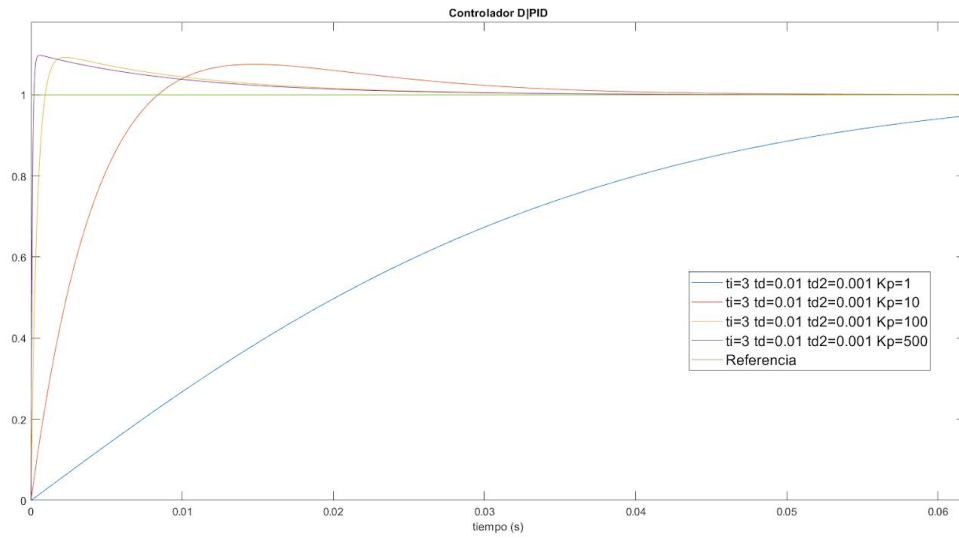


Figura 1.26: Salida con controlador D|PID variando K_p

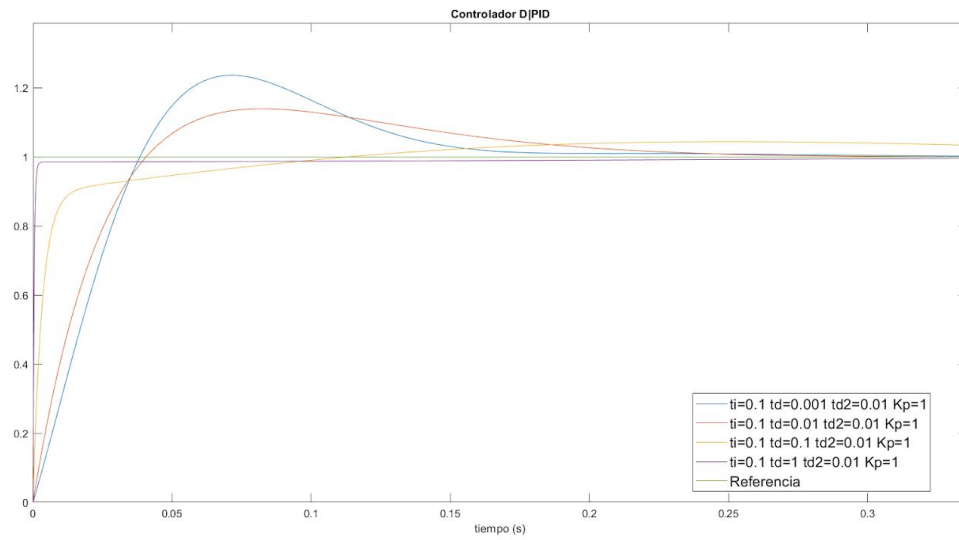


Figura 1.27: Salida con controlador D|PID variando τ_{D1}

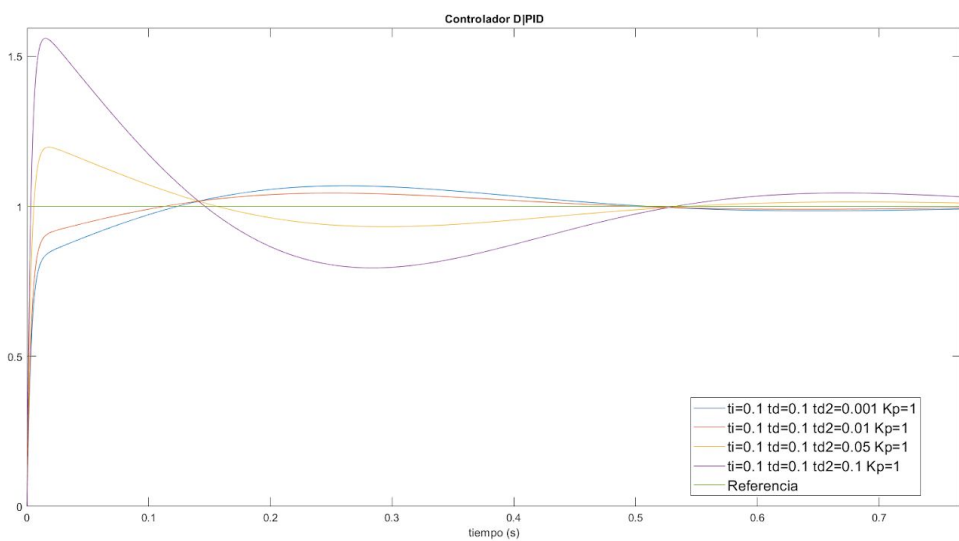


Figura 1.28: Salida con controlador D|PID variando τ_{D2}

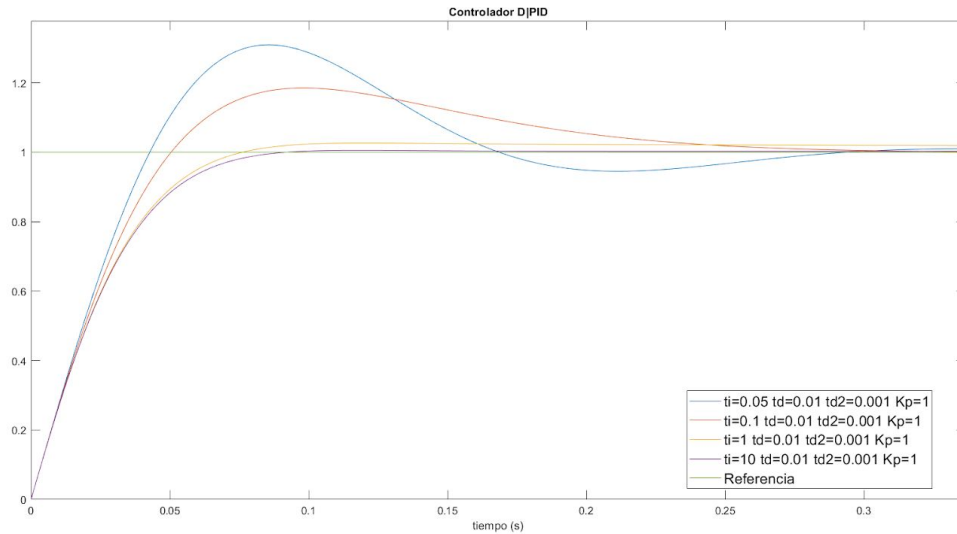


Figura 1.29: Salida con controlador DJPID variando τ_i

Podemos observar que el controlador es propenso a alcanzar una sobreelongación máxima máxima (M_p) grande si no se sintonizan bien los parámetros de la rama prealimentada.

Concluimos también que la rama directa se comporta de la misma manera que en el caso PID. Sin embargo el factor derivativo de la rama prealimentada se comporta de forma inversa al factor derivativo de la rama directa. Es decir, a mayor valor de τ_{D2} mayores oscilaciones obtenemos, pero el tiempo de subida (t_r) es menor.

2. Análisis, diseño e implementación de un sistema de control de posición angular de un motor DC utilizando un controlador D|PID

Introducción

En este apartado analizaremos la estabilidad y los problemas de seguimiento de señales monómicas, del controlador con estructura prealimentada D|PID **[Figura 1.25]**. Después diseñaremos un controlador sintonizando los parámetros para cumplir las siguientes especificaciones:

- Sobreelongación máxima de la salida al escalón $M_p \in [5 \%, 12 \%]$.
- Tiempo de establecimiento $t_s \leq 0,3$ segundos para una tolerancia $v = 5 \%$.
- Tiempo de subida $t_r \leq 0,15$ segundos.

Por último probaremos los parámetros escogidos en el Telaboratorio, comparando la salida del sistema con el cálculo teórico. Utilizaremos la función de transferencia simplificada del motor DC del Telaboratorio:

$$G(s) = \frac{2652,28}{s(s + 64,986)}$$

A. Análisis de estabilidad

Para estudiar la estabilidad del sistema utilizaremos los métodos explicados en el documento “Análisis en el dominio del tiempo y dominio de la frecuencia” **[2.1]** proporcionado en la asignatura. En concreto se va a comprobar la estabilidad en el sentido de Hurwitz, y para ello se realizará el test de Routh. El polinomio característico de este sistema es:

$$P(s) = s^3 + (p + KK_P\tau_{D1})s^2 + KK_P\left(s + \frac{1}{\tau_I}\right)$$

Construimos la tabla de Routh:

s^3	1	KK_p	0
s^2	$p + KK_p\tau_{D1}$	KK_p/τ_I	0
s^1	b_1	b_2	0
1	c_1	c_2	0

Donde:

$$b_1 = \frac{a_1 a_2 - a_3}{a_1} = \frac{p + KK_p \tau_{D1} - \frac{1}{\tau_I}}{\frac{p}{KK_p} + \tau_{D1}}$$

$$b_2 = \frac{a_1 a_4 - a_5}{a_1} = 0$$

$$c_1 = \frac{b_1 a_3 - a_1 b_2}{b_1} = a_3 = \frac{KK_p}{\tau_I}$$

$$c_2 = \frac{b_1 a_5 - a_1 b_3}{b_1} = 0$$

El teorema de Routh afirma que el polinomio $P(s)$ es estable en el sentido de Hurwitz si todos los coeficientes de la primera columna son positivos. Entonces, para que sea estable se debe cumplir:

$$\textcircled{1} \quad a_1 > 0 \Rightarrow p + KK_p \tau_{D1} > 0 \Rightarrow \boxed{K_p \tau_{D1} > -0,0245}$$

$$\textcircled{2} \quad b_1 > 0 \Rightarrow a_2 - \frac{a_3}{a_1} > 0 \Rightarrow KK_p > \frac{KK_p / \tau_I}{p + KK_p \tau_{D1}} \Rightarrow \boxed{64,986 + 2652,28 \cdot KK_p > \frac{1}{\tau_I}}$$

$$\textcircled{3} \quad c_1 > 0 \Rightarrow a_3 > 0 \Rightarrow \boxed{\frac{K_p}{\tau_I} > 0}$$

B. Análisis del problema de seguimiento

Según se demuestra en el documento de diseño de controladores proporcionado en la asignatura **[2.2]**, este controlador podrá resolver con error nulo el seguimiento de las señales monómicas de grado menor que 3, es decir, las señales escalón, rampa y parábola. Simulando el sistema en Matlab, con estas señales como estímulos, obtenemos las siguientes gráficas:

I. Escalón unidad

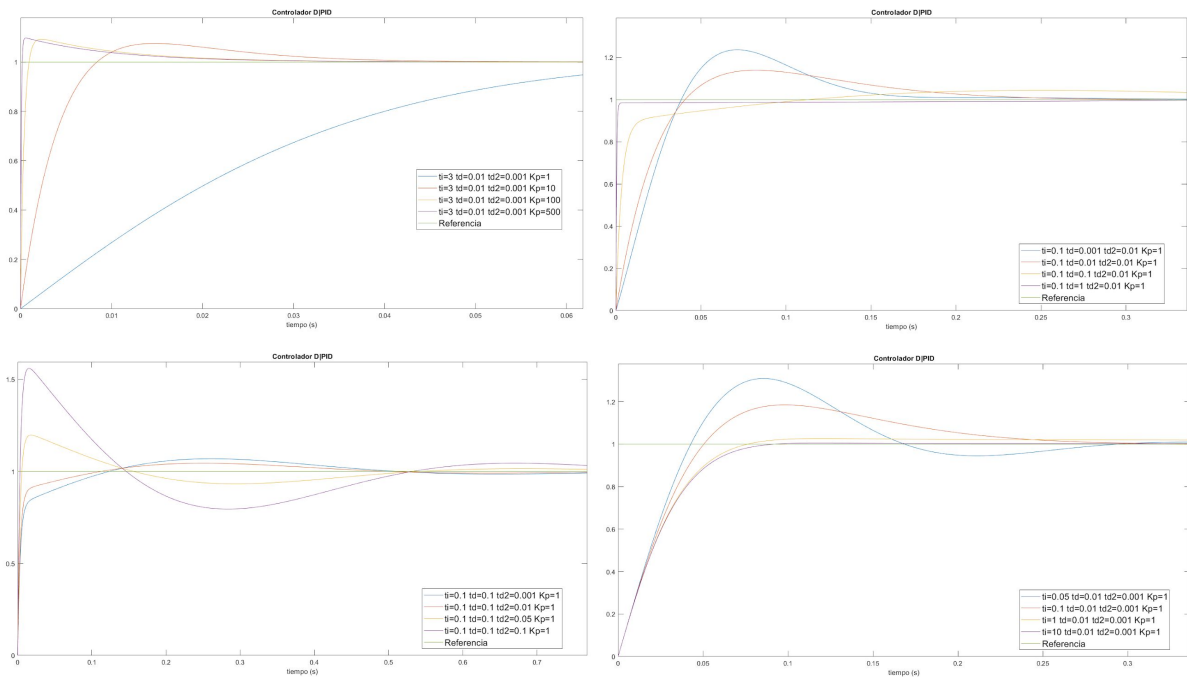
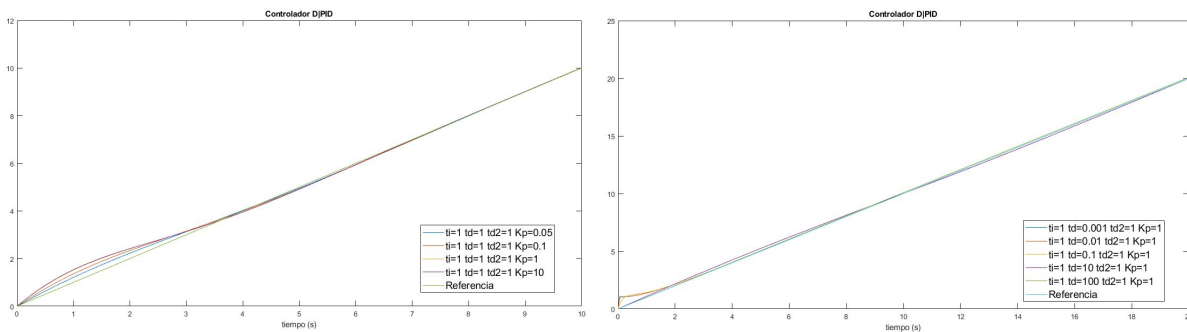


Figura 2.1: Salida con controlador D/PID al escalón unidad

II. Rampa



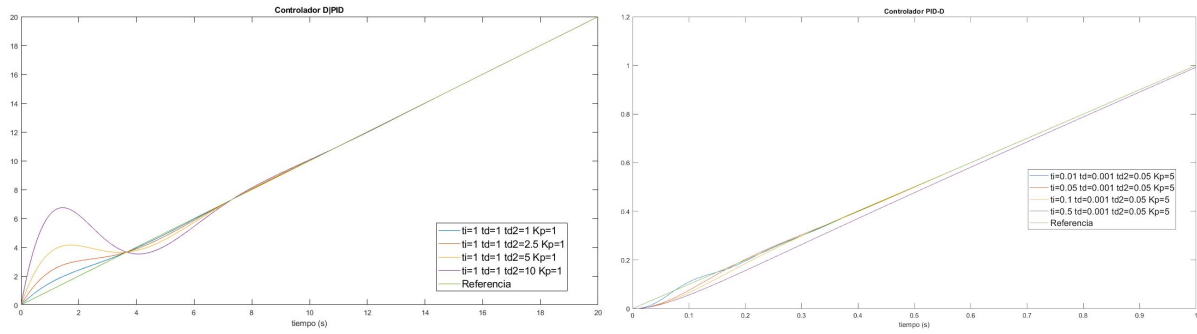


Figura 2.2: Salida con controlador DJPID a la rampa

III. Parábola

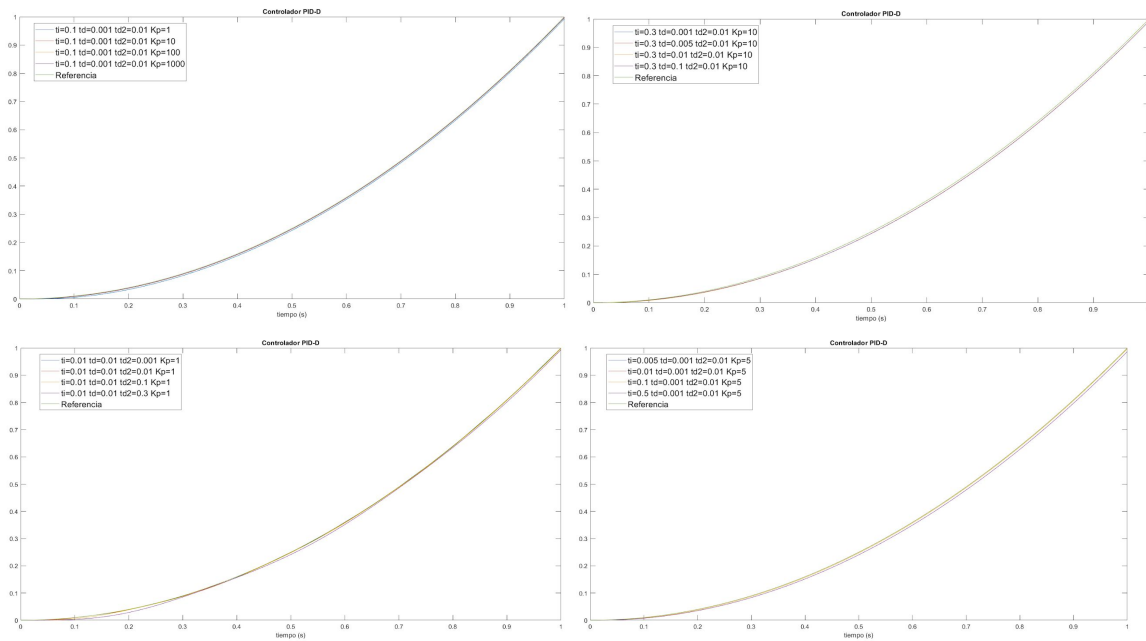


Figura 2.3: Salida con controlador DJPID a la parábola

Podemos observar en las gráficas, que mientras los parámetros introducidos cumplan las condiciones de estabilidad, el sistema podrá resolver con error nulo el problema de seguimiento del escalón, la rampa y la parábola.

C. Sintonización de parámetros

En este apartado vamos a estudiar el problema de diseño del controlador D|PID atendiendo a la sobreelongación máxima (M_p), al tiempo de establecimiento (t_s) y al tiempo de subida (t_r). Todo ello diseñado para el seguimiento de la señal escalón de 3.14 (π) radianes.

Según se demuestra en el documento de diseño de controladores proporcionado en la asignatura [2.3], M_p es independiente del β_2 y depende de manera no lineal de β y ξ . Y también se demuestra que t_s depende de manera lineal de β_2 y de manera no lineal de β y ξ .

Debido a la dificultad de un cálculo analítico de estos parámetros, simularemos con la herramienta Matlab las respuestas al escalón del controlador variando β y ξ , dejando β_2 fija (con un valor arbitrario).

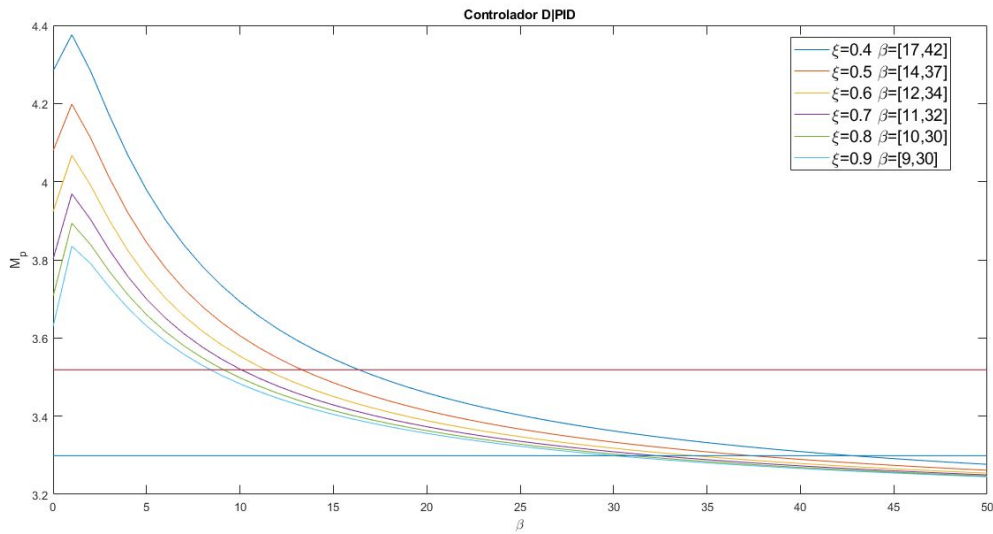


Figura 2.4: Valor de M_p en función de β y ξ

Se puede observar que las curvas de M_p tienen un máximo para un valor de β dependiente de ξ . De esta gráfica podemos sacar un conjunto de pares de valores de β y ξ que satisfacen la especificación de M_p .

Los valores que cumplen las especificaciones corresponden con los puntos de las curvas que están entre las dos líneas de referencia en 3.297 (5%) y 3.5168 (12%) radianes. Vamos a escoger valores de β que proporcionen un M_p cercano a 3.297 (5%).

Ahora volveremos a simular con los β y ξ obtenidos, variando β_2 para encontrar valores que satisfacen la especificación de t_s .

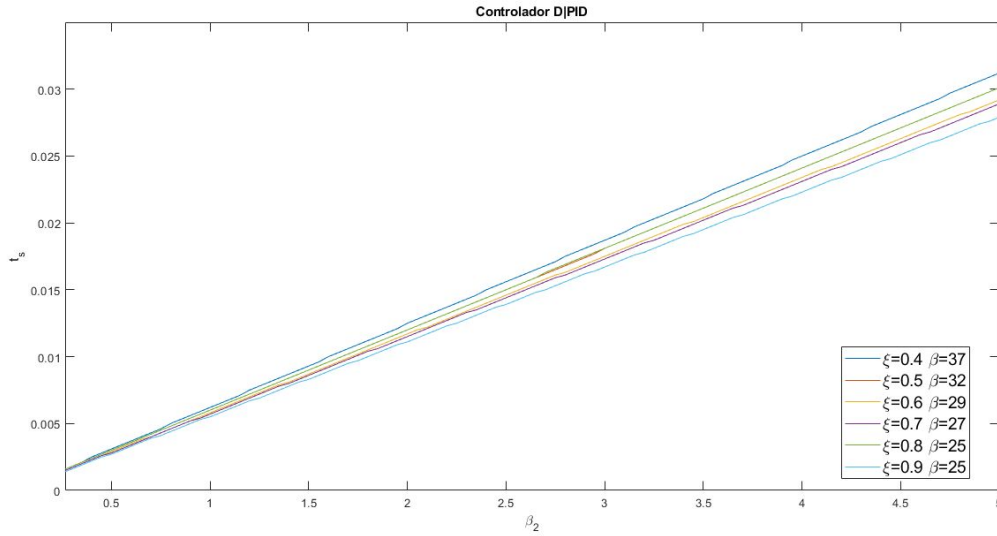


Figura 2.5: Valor de t_s en función de β , β_2 y ξ

Observamos que teóricamente el tiempo de establecimiento se satisface sin problemas. Ahora volvemos a simular con los β y ξ obtenidos anteriormente, variando β_2 para encontrar valores que satisfacen la especificación de t_r .

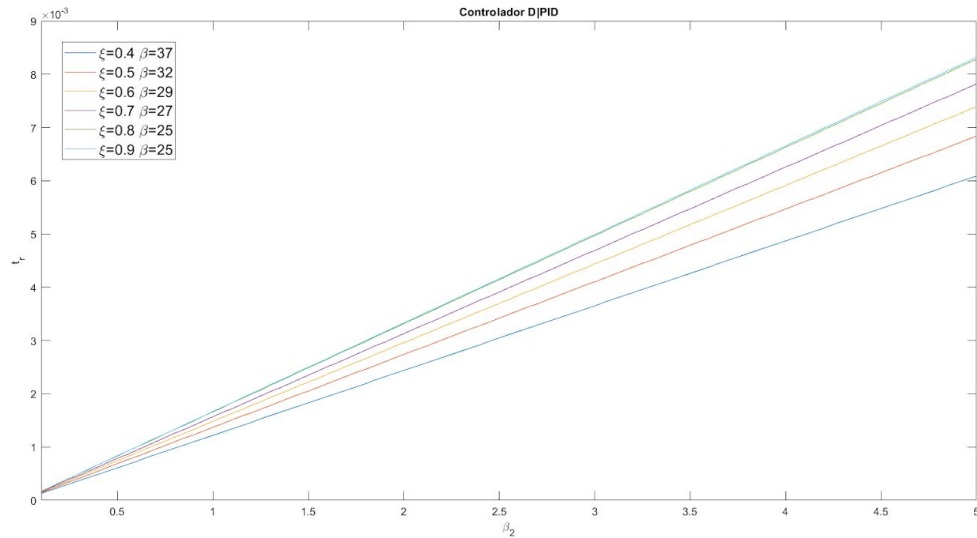


Figura 2.6: Valor de t_r en función de β , β_2 y ξ

Se observa que cualquier combinación de los valores escogidos anteriormente satisface teóricamente el tiempo de subida t_r deseado. Escogemos un valor de $\beta_2 = 1$.

Por último vamos a simular la salida del controlador D|PID con los valores escogidos:

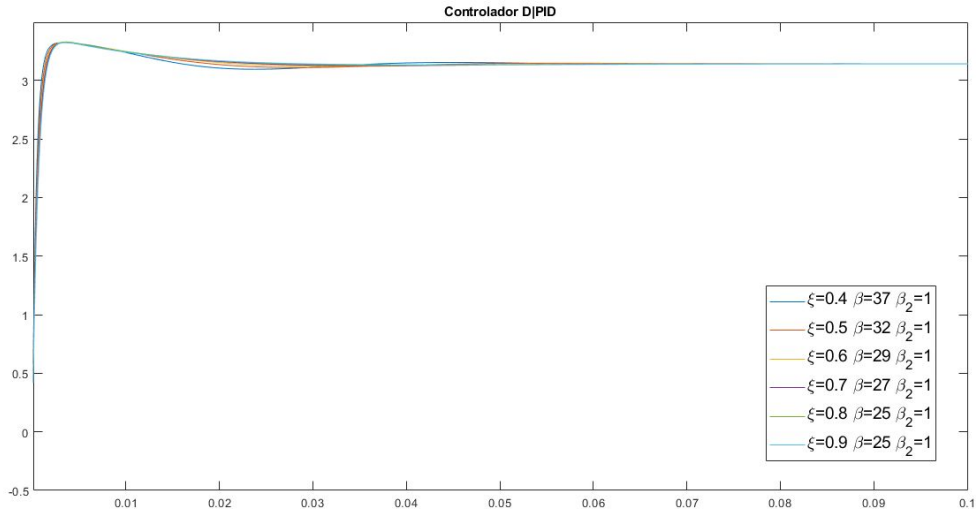


Figura 2.7: Salida con controlador D|PID al escalón de 3.14 rad

Finalmente los valores para los parámetros de diseño elegidos son:

$$\xi = 0.9$$

$$\beta = 25$$

$$\beta_2 = 1$$

D. Experimento en Telelaboratorio

Ahora vamos a comprobar el funcionamiento experimental del controlador D|PID que hemos diseñado. Lo realizaremos mediante la aplicación telemática proporcionada en la asignatura **[2.4]** que nos da acceso remoto a un motor DC.

A partir de los parámetros obtenidos en el apartado anterior, con las ecuaciones descritas en el documento de diseño de controladores **[2.2]**, calculamos:

$$K_p = \frac{p^2 \left(2\beta + \frac{1}{\xi^2} \right)}{\beta_2^2 K} = 81.58$$

$$\tau_{D1} = \frac{\beta_2 (\beta - \beta_2 + 2)}{p \left(2\beta + \frac{1}{\xi^2} \right)} = 0.0078$$

$$\tau_I = \frac{\beta_2 \xi^2 \left(2\beta + \frac{1}{\xi^2} \right)}{p\beta} = 0.0255$$

$$\tau_{D2} = \frac{p}{KK_p} = 0.0003$$

Teniendo en cuenta que trabajamos con un sistema discretizado con $T = 2 \text{ ms}$, calculamos los valores a introducir en el Telaboratorio:

$$K_P = 81.58$$

$$K_{D1} = \frac{K_P \tau_{D1}}{T} = 318.525$$

$$K_I = \frac{K_P}{\tau_{D1}} T = 6.387$$

$$K_{D2} = \frac{K_P \tau_{D2}}{T} = 12.251$$

Obtenemos los siguientes resultados:

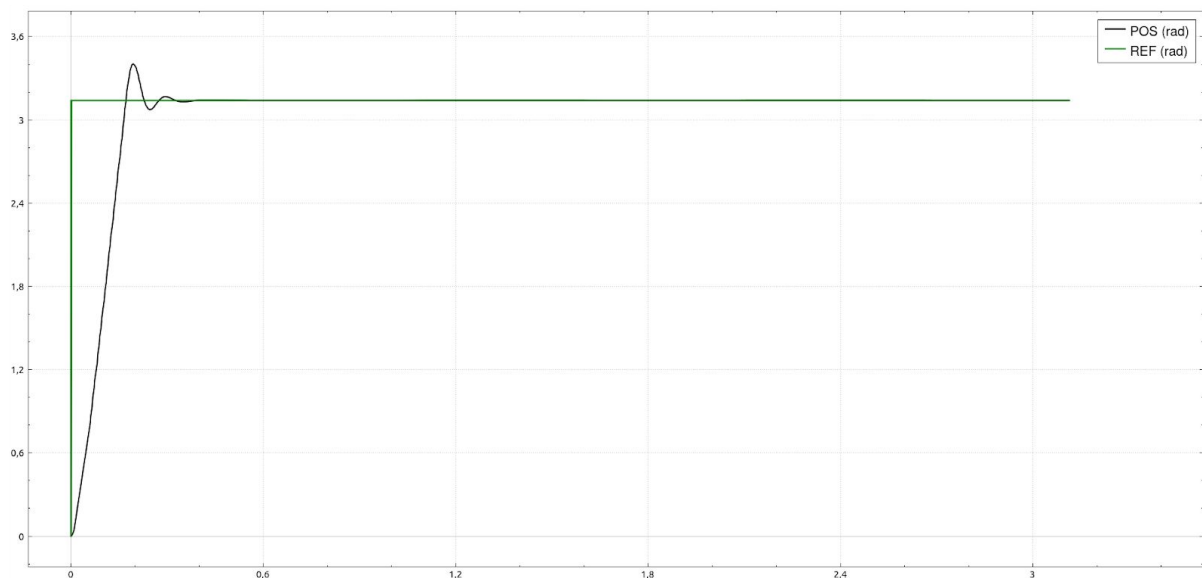


Figura 2.8: Posición del motor DC con controlador DJPID al escalón de 3.14 rad en Telaboratorio



Figura 2.9: Velocidad, posición y error del motor DC con controlador DJPID al escalón de 3.14 rad en Telaboratorio



Figura 2.10: Tensión entregada al motor DC con controlador DJPID al escalón de 3.14 rad en Telaboratorio

Observamos que se cumplen dos de las especificaciones siendo $M_p = 3.403$ rad y $t_s = 0.2115$ segundos. Pero el sistema es incapaz de cumplir el tiempo de subida, siendo en nuestro caso $t_r = 0.1724$ segundos.

Aun así, viendo que esta especificación incumplida solo se desvía 0.224 segundos, creemos que es un diseño correcto para el controlador. Ya que consideramos que el parámetro del tiempo de subida es menos importante que los otros dos, y por lo tanto no es necesario que su cumplimiento sea tan estricto.

Cabe destacar observando la figura 2.10, que la tensión de entrada al motor DC teórica es muy elevada pero la tensión máxima que se le puede entregar (saturación) son 12V. Deducimos que este hecho junto con ser un sistema discreto y no continuo, son las causas por las que hay diferencias entre la respuesta teórica y la real.

3. Análisis, diseño e implementación de un sistema de control de posición angular de un motor DC utilizando un controlador PID, sintonizado con el método de Ziegler-Nichols Modificado

Introducción

En este apartado se pretende diseñar un controlador tipo PID de posición angular para un motor DC. Para ello, primero analizaremos en el dominio de la frecuencia la función de transferencia sin simplificar del motor DC del Telaboratorio definida como:

$$G(s) = \frac{22233201,58}{s(s + 8382,70)(s + 64,986)}$$

Después diseñaremos el controlador, sintonizando los distintos parámetros mediante el método de Ziegler-Nichols Modificado. Y por último probaremos el controlador diseñado en el Telaboratorio comparando la salida real con el comportamiento teórico.

A. Análisis del dominio de la frecuencia

El motor DC se trata de un sistema de tercer orden ya que tiene tres polos, uno de ellos situado en el origen (con lo que podrá resolver el seguimiento de señales escalón) y otros dos dominantes y conjugados.

Lo primero será aplicar la reductora del motor a la función (1:23,04) **[3.1]** y pintar el diagram de Bode y de Nyquist en lazo abierto y calcular el margen de ganancia (MG) y de fase (MF) del sistema:

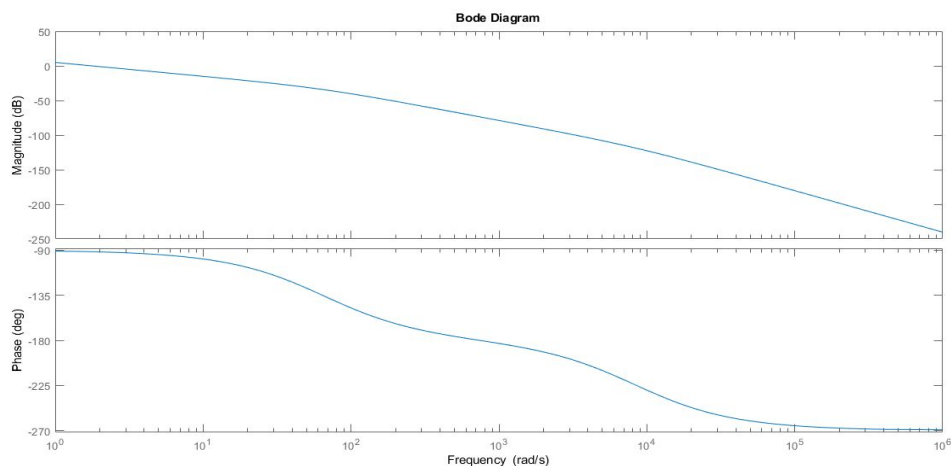


Figura 3.1: Diagramas de bode de la función del motor DC aplicando la reductora

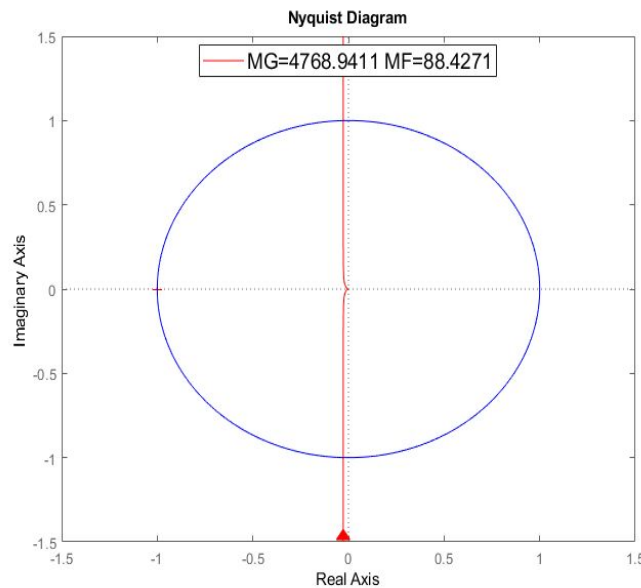


Figura 3.2: Diagramas de Nyquist de la función del motor DC aplicando la reductora

A primera vista vemos que se asemeja a un sistema de tipo 1 [3.2], ya que empieza la fase en -90° (se puede ver tanto en Bode como en Nyquist) y luego en infinito es paralelo a uno de los ejes. Si miramos su estabilidad aplicando el criterio de estabilidad de Nyquist [3.3] tenemos que:

- $P = 0$ Número de polos en el semiplano derecho.
- $N = 0$ Número de vueltas que rodea al punto $-1+j0$ en sentido antihorario).
- $Z = N + P = 0$ Número de polos en semiplano derecho en sistema de lazo cerrado.

Dado que $Z = 0$ podemos asegurar que el sistema es estable en lazo cerrado.

Por último miramos los valores del margen de ganancia y margen de fase, observamos que el que más nos llama la atención es el margen de ganancia al ser muy elevado, como tenemos un sistema de orden mayor a dos esto indica que en lazo cerrado el sistema tendrá un tiempo de subida alto ya que están directamente relacionados [3.4].

Sobre el margen de fase destacar que también es alto, con lo que podemos esperar un sistema sobreamortiguado ya que se relacionan directamente con el coeficiente de amortiguamiento.

Para comprobar los diferentes aspectos mencionados del sistema, lo simulamos para una señal escalón en lazo cerrado en el tiempo:

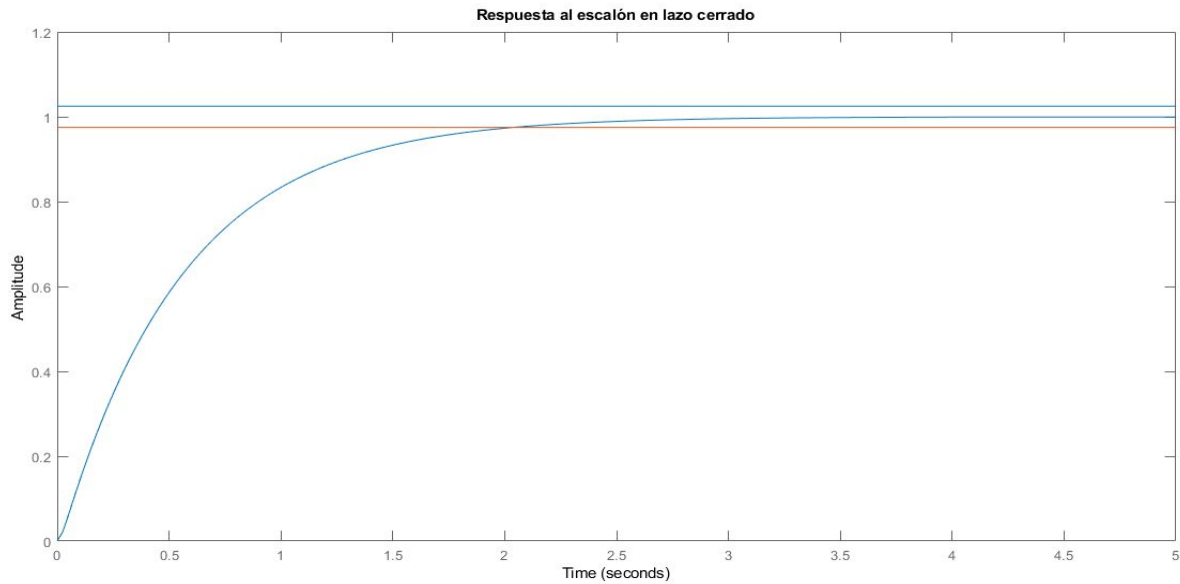


Figura 3.3: Respuesta al escalón del motor DC en lazo cerrado

Podemos observar los cuatro aspectos más relevantes mencionadas anteriormente: el sistema es estable, en régimen permanente no tiene error ya que sigue perfectamente la señal escalón, es un sistema lento ya que el tiempo de subida y establecimiento es muy alto (se está considerando una tolerancia del 2,5%) , y por último, se observa con claridad que el sistema está sobreamortiguado.

B. Diseño del controlador mediante el método de Ziegler-Nichols Modificado

Una vez hecho el análisis del motor DC realizaremos un controlador para mejorar el rendimiento del sistema. Como hemos visto, el sistema es muy lento (tiene un alto margen de ganancia) por lo que primero vamos a intentar que el sistema sea más rápido. Para ello lo más fácil es implementar un controlador tipo P que calculamos con el método Ziegler-Nichols Modificado **[3.5]**.

En este caso, usamos el método para mover el punto de corte del sistema con el semieje real negativo, (por definición, es el margen de ganancia) en dirección al punto $-1 + j0$:

$$Kp = r_c * \cos \phi_c, \quad r_c = \frac{MG_a}{MG_b}$$

$$\phi_c = 0 \Rightarrow Kp = \frac{MG_a}{MG_b}$$

Donde MG_b es el margen de ganancia deseado y MG_a el del motor DC.

Para elegir la K_p del controlador P tendremos en cuenta dos factores: primero, aunque el margen de ganancia es un valor relativo, generalmente se considera adecuado un valor mayor que dos [3.6], y segundo, un valor muy alto de K_p podría saturar el sistema en un caso real, por lo que hay que encontrar un término medio entre los dos factores.

Simulando varios márgenes de ganancia en Matlab al final nos decidimos por un $MG=20$, que satisface los dos factores mencionados anteriormente. Los resultados de la simulación son:

$$MG_b = 20, MG_a = 4768.94 \Rightarrow K_p = 238.44$$

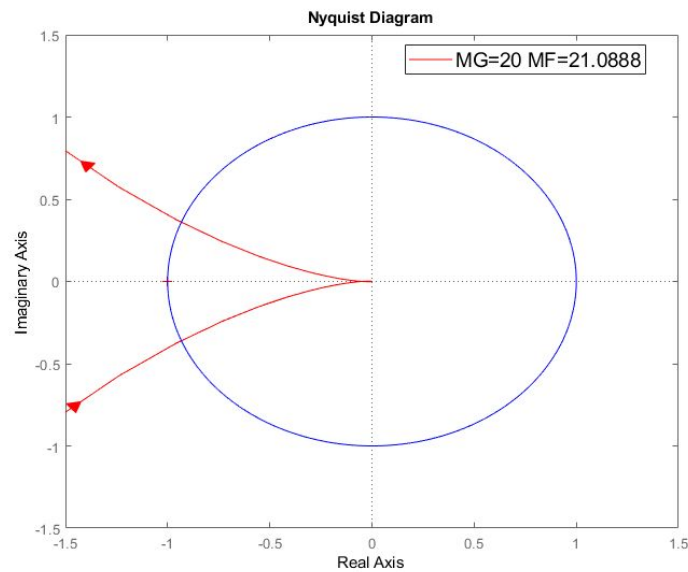


Figura 3.4: Diagrama de Nyquist del sistema con controlador P

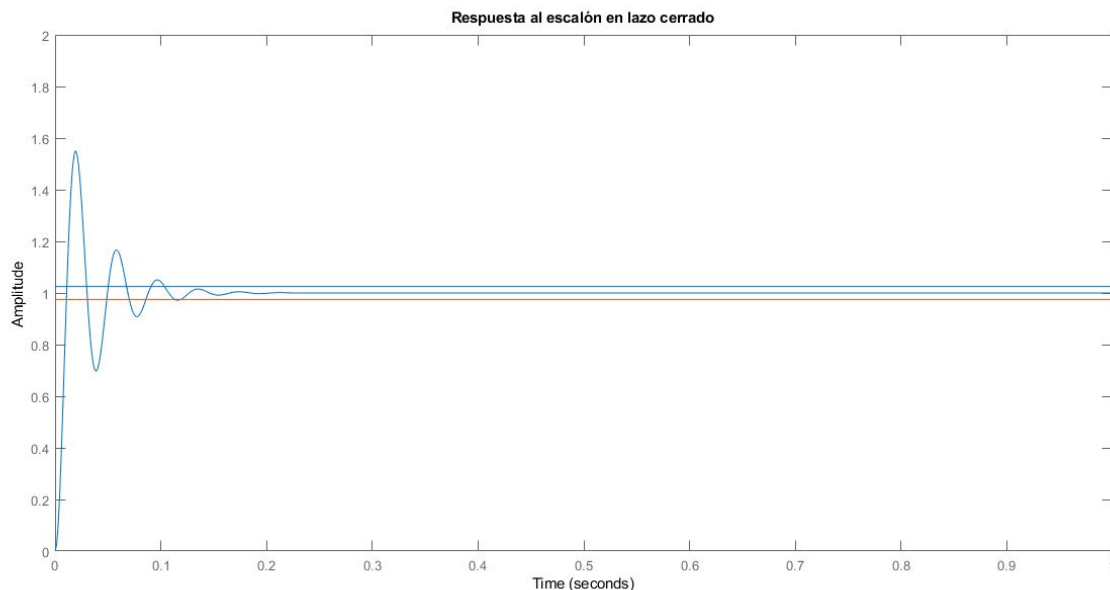


Figura 3.5: Respuesta al escalón del sistema con controlador P

Podemos ver que el tiempo de subida se ha reducido considerablemente, pero han aparecido oscilaciones, el sistema ahora está subamortiguado y la sobreelongación máxima es muy elevada.

Observamos esto también en el margen de fase, que ha sido reducido a 21.09°. Como pasaba con el margen de ganancia, el margen de fase generalmente tiene unos valores adecuados, en este caso a partir de los 30°. Por ello decidimos implementar un controlador que nos permita aumentar el margen de fase, y en definitiva, reducir las oscilaciones del sistema (recordemos que estaba directamente relacionado con el amortiguamiento).

Para ello habrá que hacer uso de un controlador PD o PID, como el enunciado nos especifica un PID nos decantamos por este, que además al ser de tercer orden nos permitirá resolver el problema de seguimiento de señales rampa.

Para calcular los parámetros, hacemos uso otra vez del método de Ziegler-Nichols Modificado, pero esta vez para desplazar el punto de corte del sistema con la circunferencia unidad:

$$K_{pid} = K_p * r_c * \cos \phi_c ; \phi_c = \phi_b - \phi_a$$

$$\tau_i = \frac{1}{2\omega_0 \alpha} * (\tan \phi_c + \sqrt{4\alpha + (\tan \phi_c)^2})$$

$$\tau_d = \alpha * \tau_i$$

Como el controlador PID está seguido del P el factor K_{PID} depende de K_p .

Para encontrar el margen de fase adecuado fijamos $\alpha = 0.25$ como en el ejemplo proporcionado [3.7], y probamos distintos teniendo en cuenta dos factores: que la sobreelongación máxima no sobrepase 1.3, y que el tiempo de subida y establecimiento sea el mínimo posible para conseguir que el sistema responda rápido.

Probando varios márgenes de fase en la herramienta Matlab, al final nos decidimos por MF = 60. Los resultados obtenidos son los siguientes:

$$\phi_b = 60^\circ ; \phi_a = 21.09^\circ \Rightarrow \phi_c = 38.91^\circ$$

$$K_{pid} = 238.44 * 1 * \cos 38.91 = 185.54$$

$$\tau_i = \frac{1}{2 * 159.42 * 0.25} * (\tan 38.91 + \sqrt{4 * 0.25 + (\tan 38.91)^2}) = 2.63 * 10^{-2}$$

$$\tau_d = 0.25 * 2.63 * 10^{-2} = 6.6 * 10^{-3}$$

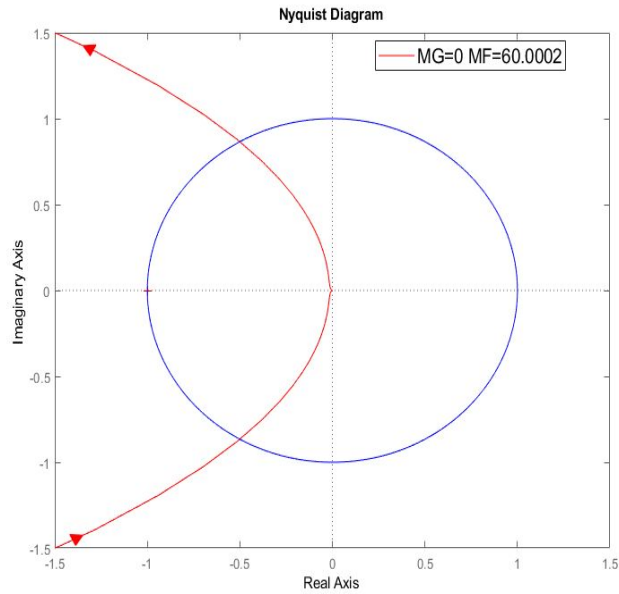


Figura 3.6: Diagrama de Nyquist del sistema con controlador PID

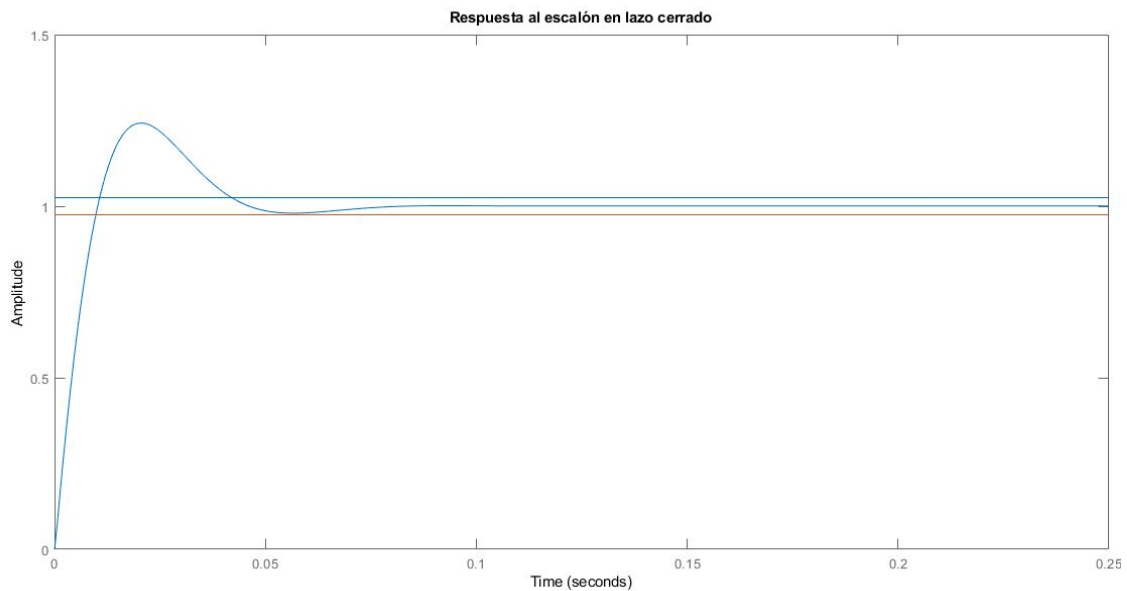


Figura 3.7: Respuesta al escalón del sistema con controlador PID

Vemos que el resultado es el deseado, un sistema que responde rápidamente al estímulo de entrada ($t_r = 0.01$), con cierto subamortiguamiento ($M_p = 1.24$).

Cabe destacar también que el margen de ganancia se ha reducido a cero (el sistema no corta con el semieje real y negativo), esto podría preocuparnos en un principio ya que el adecuado está por encima de dos, pero comprobamos rápidamente en la gráfica temporal que el sistema, aun así, sigue siendo estable.

C. Experimento en Telelaboratorio

Ahora vamos a comprobar el funcionamiento experimental del controlador que hemos diseñado. Lo realizaremos mediante la aplicación telemática proporcionada en la asignatura **[3.8]** que nos da acceso remoto a un motor DC. Calculamos primero los parámetros del PID (rama directa) para un período $T = 2 \text{ ms}$:

$$Kp^* = Kpid = 185.54$$

$$Ki = Kpid * \frac{T}{\tau_i} = 14.14$$

$$Kd = Kpid * \frac{T}{\tau_d} = 608.81$$

Obtenemos los siguientes resultados:

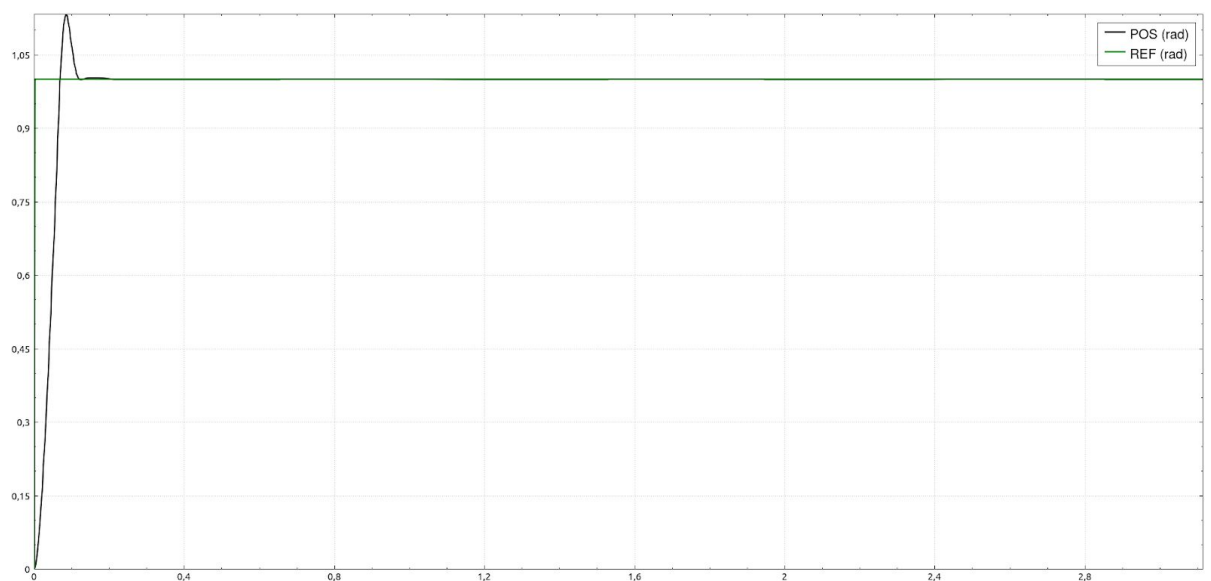


Figura 3.8: Respuesta al escalón del sistema con controlador PID en el motor de Telelabo.

El resultado obtenido es un sistema con una sobreelevación algo menor ($M_p = 1,13$) y tiempo de subida mayor ($t_r = 0.07$) ya que se trata de un sistema real.

4. Conclusiones

En el primer apartado realizamos un análisis teórico y de simulación de los distintos controladores explicados en el aula. Podemos concluir que en general:

- El factor proporcional (P) corrige el error del momento actual, y a medida que aumenta su valor disminuyen el tiempo de establecimiento (t_s) y el tiempo de subida (t_r). Pero si su valor aumenta demasiado comienzan a aparecer oscilaciones y sube la sobreelongación máxima (M_p).
- El factor derivativo (D) realiza una corrección predictiva que actúa antes de que el error aumente demasiado, provocando en general que se den menos oscilaciones. Pero si su valor aumenta demasiado el sistema se volverá más lento aumentando tanto el tiempo de establecimiento (t_s) como el tiempo de subida (t_r).
- El factor integral (I) corrige el error acumulado en los instantes anteriores. Cuanto más disminuya su valor antes se corrige el error, pero el sistema se volverá más inestable.

Desde la perspectiva de seguimiento de señales monómicas no se puede generalizar tanto, ya que dependiendo del tipo de controlador, este será capaz de resolver con error nulo, error constante o error infinito las distintas funciones estímulo (escalón, rampa y parábola). Hay que estudiar cada caso en concreto.

En el segundo apartado hemos tenido que diseñar un controlador D|PID cumpliendo unas especificaciones dadas. Realizando el análisis teórico hemos comprobado cómo dependen unos parámetros de otros.

Llevando a la práctica en el Telaboratorio el controlador diseñado, hemos podido ver cómo se diferencian la salida teórica de la práctica. Entre otros factores, deducimos que se debe la discretización de las señales del controlador, y a la imposibilidad de proporcionar al motor DC la tensión calculada por saturación a 12V.

En el tercer apartado hemos realizado un análisis en frecuencia del motor y diseñado un controlador PID sintonizando los parámetros mediante el método de Ziegler-Nichols Modificado.

Hemos podido observar como los parámetros en el dominio del tiempo están relacionados con los parámetros del dominio de la frecuencia, el margen de fase y el margen de ganancia. Estos no pueden ser modificados a la vez, ya que en un principio probamos esta estrategia y no se consiguieron los resultados esperados, por eso nos decidimos a hacer primero un controlador P y luego un PID.

Una vez realizado el PID, nos dimos cuenta de que no era necesario el controlador P, ya que el margen de ganancia que habíamos diseñado con este se modificó al cambiar el de fase. Aun así, lo dejamos en la memoria para que se viera el procedimiento que seguimos hasta el resultado final.

Cabe destacar también que los resultados obtenidos con el método de Ziegler-Nichols Modificado en el sistema real son mejores (más parecido a la simulación teórica en MATLAB) que los obtenidos con el método utilizado en el segundo apartado.

5. Bibliografía

Apartado 1:

[1.1 - 1.7] Apuntes de la asignatura: "Diseño de controladores"

- [1.1] Páginas 5-6
- [1.2] Páginas 9-11
- [1.3] Páginas 6-8
- [1.4] Páginas 11-12
- [1.5] Páginas 18-19
- [1.6] Páginas 33-36
- [1.7] Páginas 37-39

Apartado 2:

[2.1] Apuntes de la asignatura: "Análisis en el dominio del tiempo y de la frecuencia"

[2.2 - 2.3] Apuntes de la asignatura: "Diseño de controladores"

- [2.2] Páginas 37-39
- [2.3] Página 23

[2.4] Aplicación de la asignatura: "Telelaboratorio"

Apartado 3:

[3.1] Reductora del motor

[3.2 - 3.7] Apuntes de la asignatura: "Dominio de la frecuencia"

- [3.2] Páginas 7-8
- [3.3] Páginas 9-11
- [3.4] Página 14
- [3.5] Páginas 18-21
- [3.6] Página 13
- [3.7] Página 20