



UNIVERSIDAD POLITÉCNICA DE MADRID
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE
SISTEMAS INFORMÁTICOS

Máster en Software de Sistemas Distribuidos y
Empotrados

Sistemas de Control

Práctica de Controlador PID

Alejandro Casanova Martín

N.º de matrícula: bu0383

Madrid, 12 de diciembre, 2023

Índice

Introducción	3
Ejercicio 1	3
Ejercicio 2	3
Ejercicio 2.1	3
Ejercicio 2.2	4
Ejercicio 2.3	5
Ejercicio 2.4	5
Ejercicio 2.5	6
Ejercicio 3	7
Ejercicio 3.1	7
Ejercicio 3.2	8
Ejercicio 3.3	9
Ejercicio 3.4	10
Ejercicio 3.5	11
Conclusiones	12

Introducción

A partir del modelo matemático de un motor de DC, se ha desarrollado un controlador PID para estabilizarlo, ajustando sus parámetros a partir de sus índices de rendimiento.

Todos los ejercicios han sido resueltos en el Jupyter Notebook. Sin embargo, adicionalmente se han adjuntado los resultados obtenidos en este documento.

Ejercicio 1

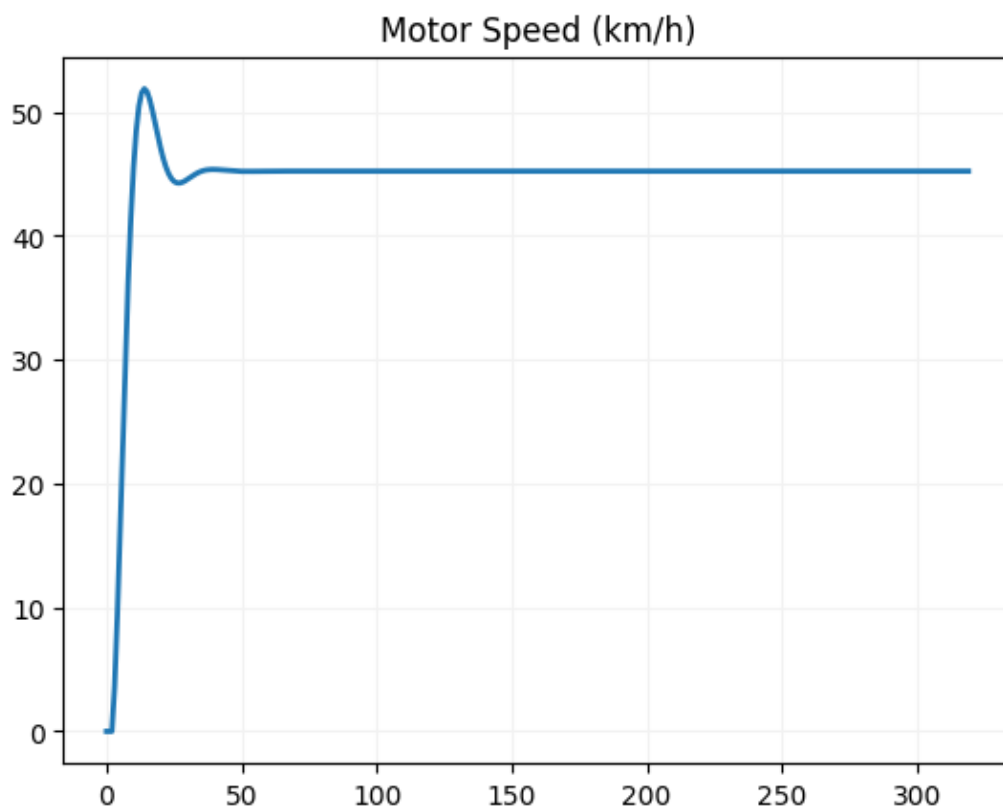
Se implementó la función *Exec_controller_cycle*, encargada de calcular la señal de control $u(t)$ para la siguiente iteración de control, a partir de los parámetros del controlador PID.

Ejercicio 2

Se ajustaron las ganancias del controlador para obtener distintas respuestas del sistema. Los resultados se muestran a continuación.

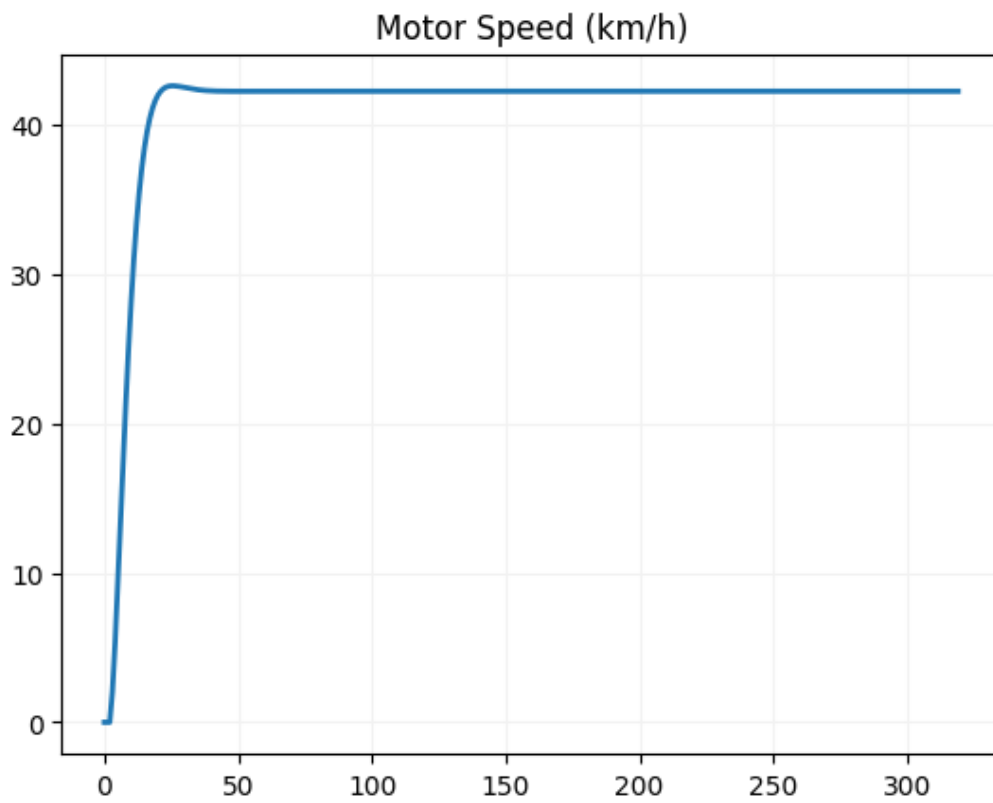
Ejercicio 2.1

Para lograr un overshoot del 10%: $K_p = 0.35$ | $K_i = 0$ | $K_d = 0$



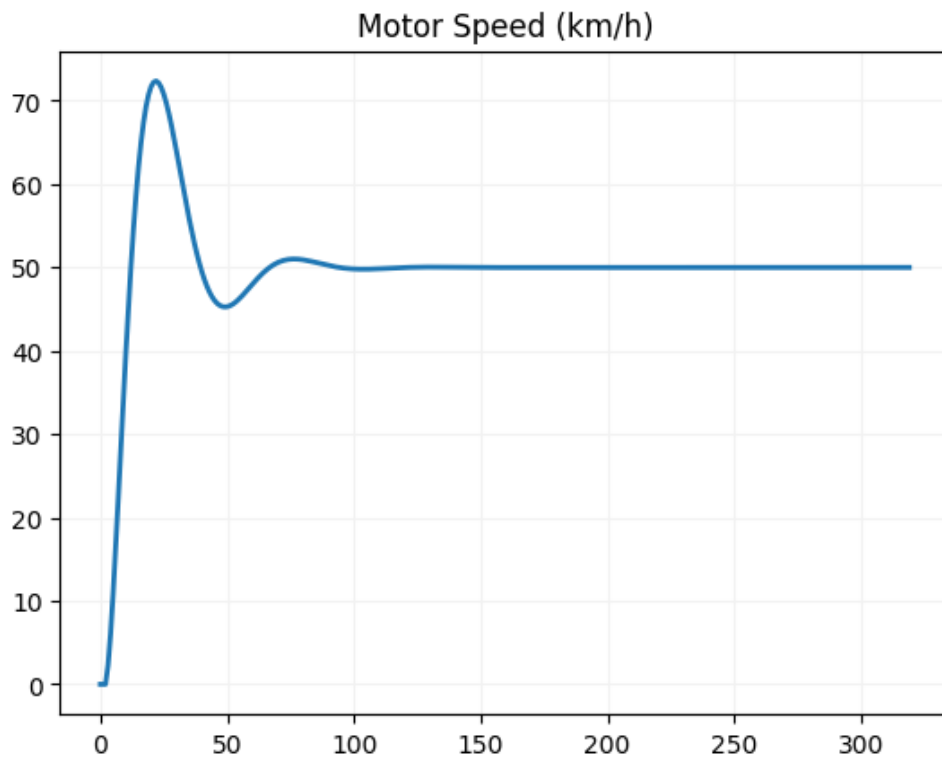
Ejercicio 2.2

Para lograr un tiempo de estabilización T_s mínimo: $K_p = 0.2$ | $K_i = 0$ | $K_d = 0$ (se redujo ligeramente K_p)



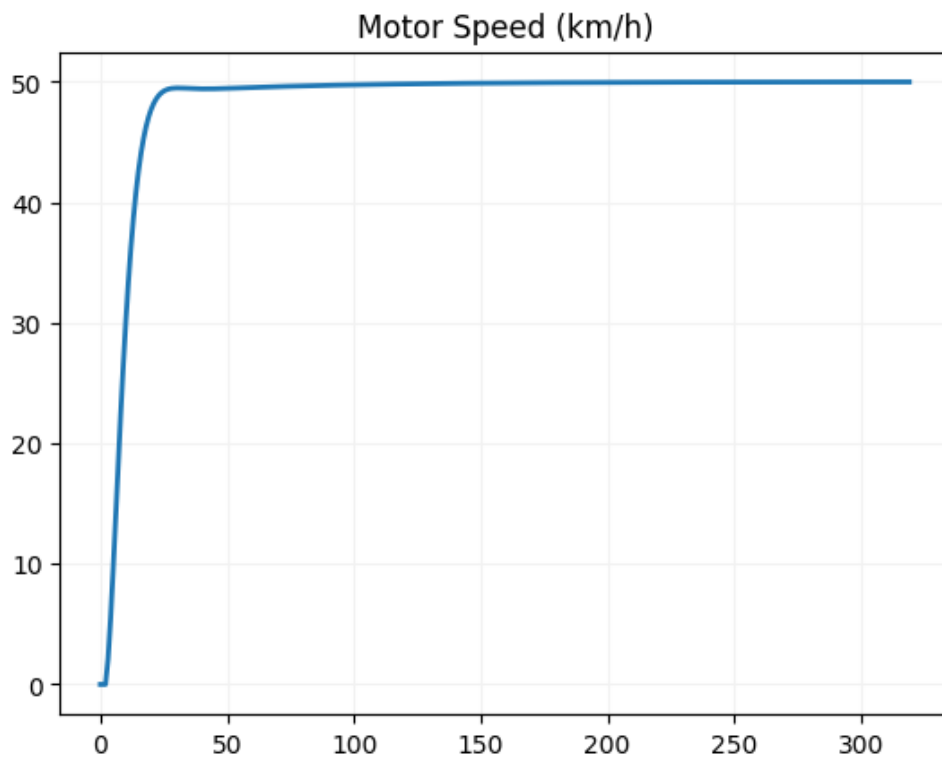
Ejercicio 2.3

Para lograr un error en régimen estacionario E_{ss} mínimo: $K_p = 0.2$ | $K_i = 0.01$ | $K_d = 0$ (se añadió K_i para eliminar el error en régimen estacionario)



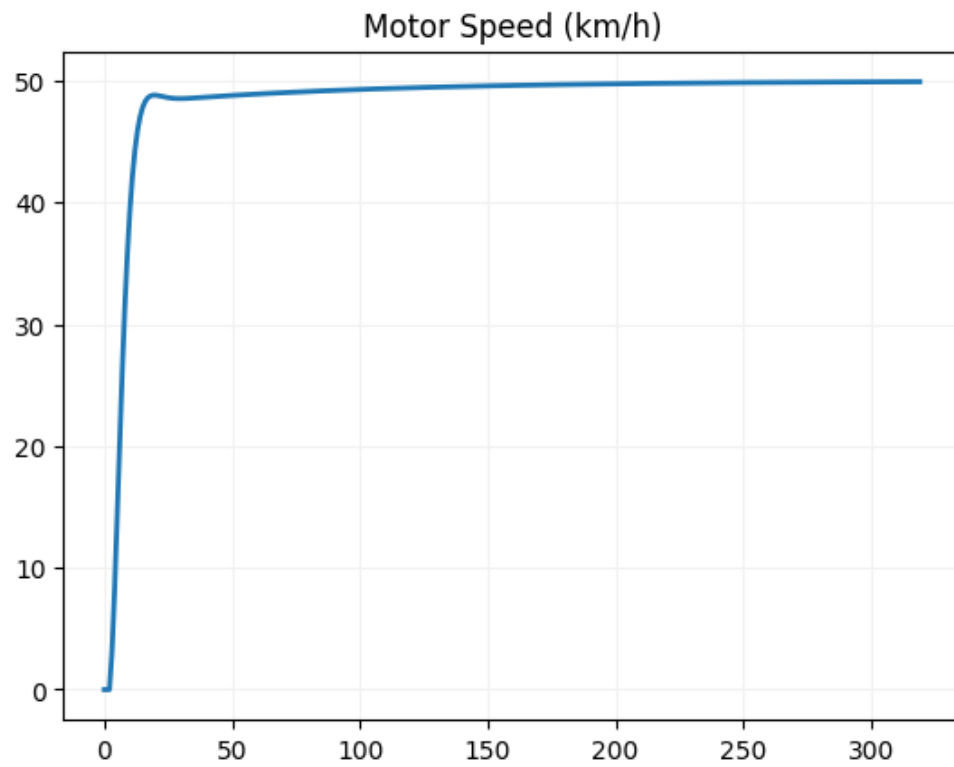
Ejercicio 2.4

Para eliminar el overshoot: $K_p = 0.2$ | $K_i = 0.06$ | $K_d = 0$ (Se aumentó K_i para eliminar el overshoot)



Ejercicio 2.5

Para reducir todos los errores: $K_p = 0.3$ | $K_i = 0.09$ | $K_d = 0.001$ (Se añadió K_d y se ajustaron las demás ganancias)

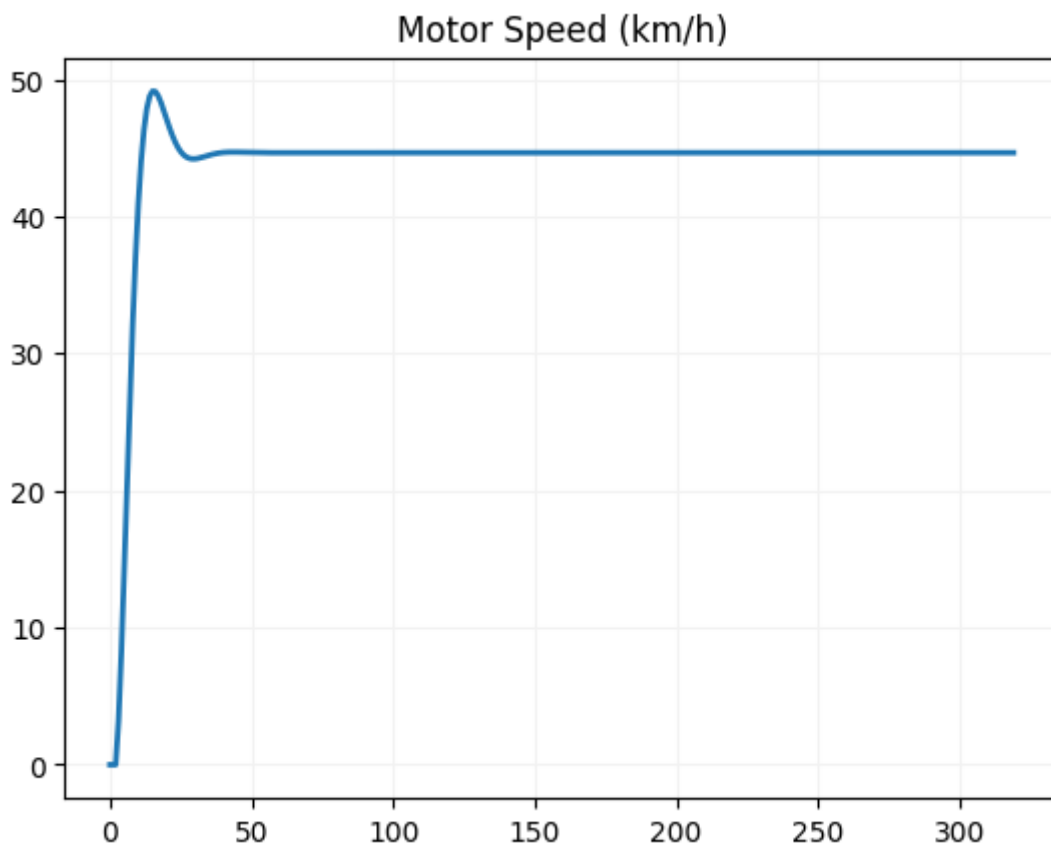


Ejercicio 3

Se implementó la función para calcular los 4 índices de rendimiento más relevantes: Overshoot, Ess, Ts y d. A continuación, se obtendrán los índices de rendimiento para las 5 configuraciones de ganancias del ejercicio 2. Finalmente, con la ayuda de dichos índices de rendimiento, se realizó un ajuste más fino del regulador para obtener mejores resultados en cada ejercicio. Los resultados se muestran a continuación.

Ejercicio 3.1

Para lograr un overshoot del 10%: $K_p = 0.308$ | $K_i = 0$ | $K_d = 0$



Overshoot: 0.1005228625493344 (10.05 %)

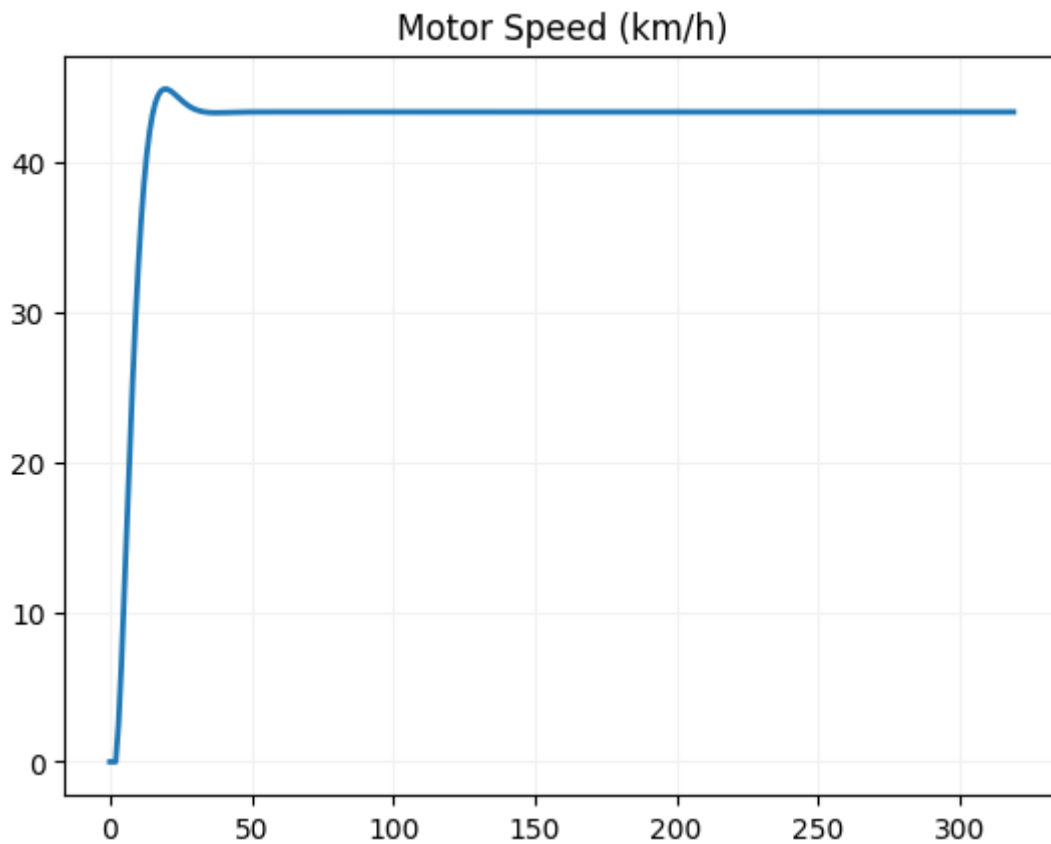
Decay Rate (d): 0.009971637499686898 (1.0 %)

Ess: -5.350135124590935 (11.98 %)

Ts: 0.0300 seconds (sample number 30)

Ejercicio 3.2

Para lograr un tiempo de estabilización T_s mínimo: $K_p = 0.24$ | $K_i = 0$ | $K_d = 0$ (se redujo ligeramente K_p)



Overshoot: 0.03526798790555552 (3.53 %)

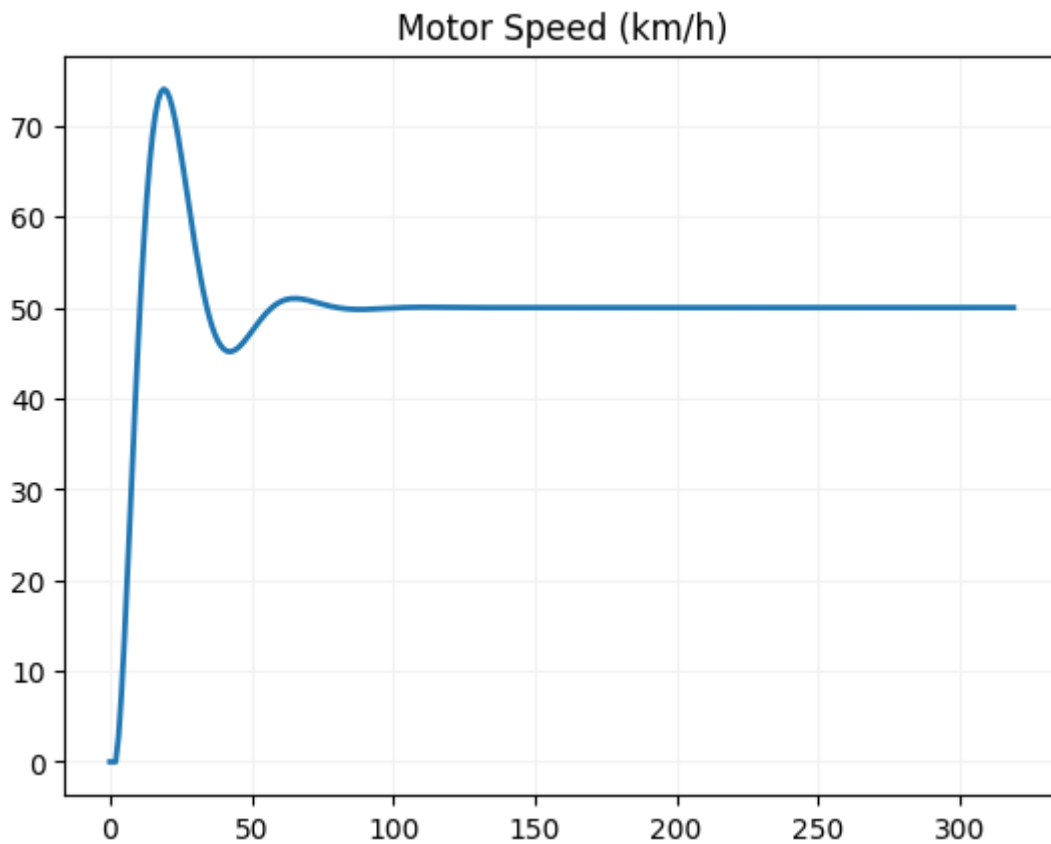
Decay Rate (d): 0.0012407633310925017 (0.12 %)

Ess: -6.663972216226199 (15.38 %)

Ts: 0.0280 seconds (sample number 28)

Ejercicio 3.3

Para lograr un error en régimen estacionario E_{ss} mínimo: $K_p = 0.24$ | $K_i = 0.01$ | $K_d = 0$ (se añadió K_i para eliminar el error en régimen estacionario)



Overshoot: 0.48101100225446425 (48.10 %)

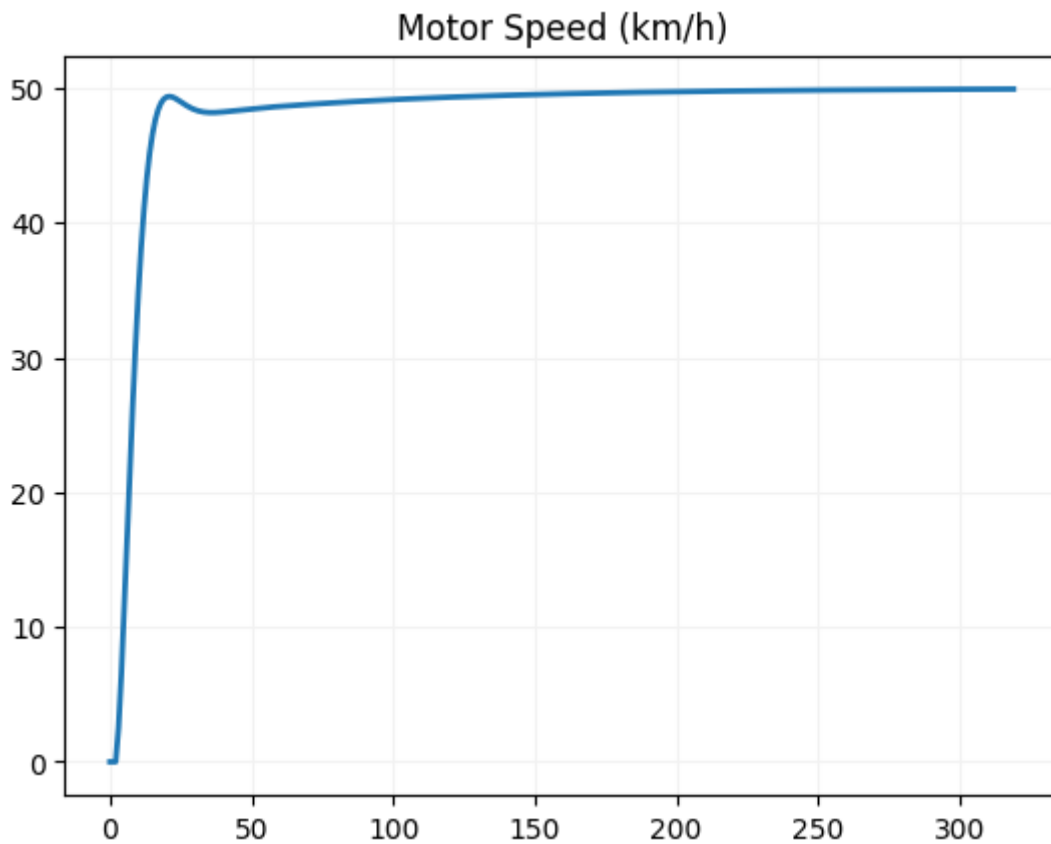
Decay Rate (d): 0.0416435808737424 (4.2 %)

Ess: -1.3951178701177014e-06 (0.00 %)

Ts: 0.0740 seconds (sample number 74)

Ejercicio 3.4

Para eliminar el overshoot: $K_p = 0.24$ | $K_i = 0.08$ | $K_d = 0$ (Se aumentó K_i para eliminar el overshoot)



Overshoot: 0 (0.00 %)

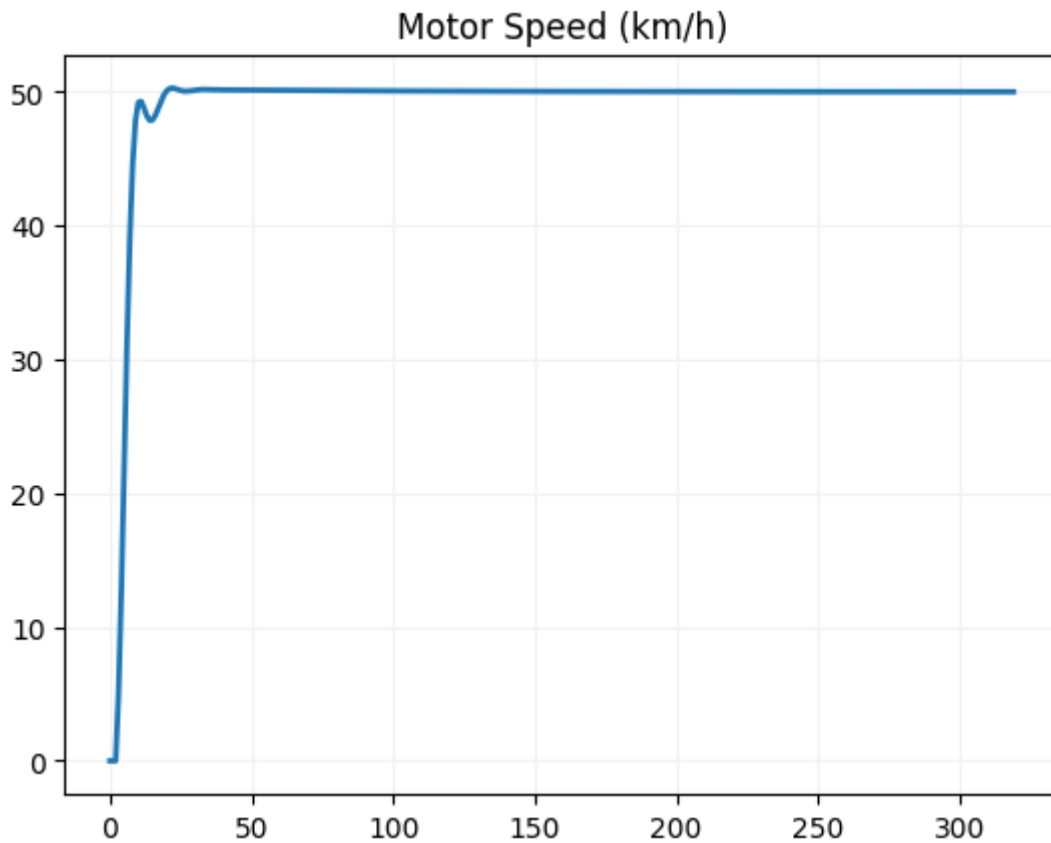
Decay Rate (d): 0.0 (0.0 %)

Ess: -0.11751474593017974 (0.24 %)

Ts: 0.1270 seconds (sample number 127)

Ejercicio 3.5

Para reducir todos los errores: $K_p = 0.5$ | $K_i = 0.08$ | $K_d = 0.002$ (Se añadió K_d y se ajustaron las demás ganancias)



Overshoot: 0 (0.00 %)

Decay Rate (d): 0.0 (0.0 %)

Ess: 0.008914187758257697 (0.02 %)

Ts: 0.0190 seconds (sample number 19)

Conclusiones

En esta práctica de sistemas de control se llevó a cabo el ajuste de los parámetros de un controlador PID, con el objetivo de controlar eficientemente un motor de corriente continua simulado.

Durante el desarrollo de la práctica, se pudo comprobar la importancia y utilidad de cada uno de los componente del regulador:

- El componente proporcional (P) se encarga de responder proporcionalmente al error actual (la diferencia entre la referencia y la salida del sistema). Un valor alto de ganancia proporcional (K_p) puede aumentar la velocidad de respuesta del sistema, pero también puede causar oscilaciones no deseadas o incluso inestabilidad.
- La componente integral (I) ayuda a reducir el error acumulado a lo largo del tiempo, y permite eliminar el error en régimen permanente.
- La componente derivativa (D) responde a la velocidad de cambio del error, proporcionando una amortiguación que evita oscilaciones excesivas y mejora la estabilidad del sistema. El componente derivativo es predictivo, es decir, puede ayudar a anticipar el comportamiento futuro del sistema, contribuyendo a una respuesta más suave y rápida ante cambios en la referencia o perturbaciones.

Durante la práctica se observó que el ajuste de los parámetros del PID es un proceso delicado y muy dependiente del sistema a controlar. Se comprobó que un valor de ganancia proporcional demasiado alto puede llevar a la inestabilidad mientras que una ganancia integral excesiva puede causar oscilaciones. La ganancia derivativa, por otro lado, juega un papel crucial en la amortiguación de oscilaciones pero un valor demasiado alto puede llevar a respuestas lentas. Por lo tanto, es importante encontrar un equilibrio adecuado entre estabilidad y velocidad de la respuesta.

A pesar de haber obtenido resultados aceptables, se ha comprobado que el ajuste manual de reguladores PID es una tarea tediosa y lenta. Además, resulta poco probable alcanzar un rendimiento óptimo mediante este método.