UNIVERSIDAD DE SANTIAGO DE CHILE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA INFORMÁTICA

Laboratorio 1 - Análisis de sistemas lineales

Integrantes:	Vanina Correa Chávez
	John Serrano Carrasco
Curso:	Modelos y Simulación
Sección:	13321-0-A-1
Profesor:	Gonzalo Acuña Leiva
Ayudante:	Tomas Lopez Aleuy

26 de Mayo de 2023

Tabla de contenidos

1. Introducción

El análisis de sistemas y modelos es una herramienta fundamental en la ingeniería y en diversas áreas de la ciencia. En este informe, se abordarán conceptos clave de la teoría de sistemas y modelos, junto con el uso de la Transformada de Laplace para así obtener la Función de Transferencia de un sistema especifico.

Las funciones de transferencia son una herramienta muy útil a la hora de analizar sistemas, ya que permiten describir la relación entre la entrada y la salida del sistema de forma matemática. Además, el análisis de sistemas lineales es fundamental para entender cómo un sistema responde a diferentes señales de entrada, y cómo se puede mejorar su desempeño.

Como objetivo principal, se tiene realizar el análisis de sistemas lineales por medio de sus funciones de transferencia, apoyándose con MATLAB, donde a lo largo de la experiencia se utilizarán distintas funciones, como feedback(), step(), zero(), pole(), entre otras.

Como estructura del informe, se tiene una definición de varios conceptos claves e importantes para la comprensión del texto, correspondiente a un Marco Teórico donde se definen conceptos como Sistema, Lazo Abierto, Lazo Cerrado, Función de Transferencia, entre otros. Luego se muestra el desarrollo de cada parte del laboratorio, indicando el paso a paso para obtener los distintos resultados junto con los gráficos correspondientes. Para finalizar, se realiza una conclusión de todo el trabajo realizado, teniendo en cuenta varios aspectos del informe como también de los códigos realizados en MATLAB.

2. Marco Teórico

2.1. Sistema

Un sistema corresponde a un conjunto de elementos que interactúan entre si para lograr una cierta meta. (Ogata (2010)). También es conocido como una porción de la realidad que se busca aislar para estudiar. Los sistemas pueden ser distintos tipos, como sistemas físicos, biológicos, informáticos, matemáticos, entre otros. Un sistema está compuesto por varias variables, como por ejemplo, variables de entrada y variables de salida. Comúnmente, una herramienta utilizada en los Sistemas Lineales Invariantes (SLI) corresponde a la Función de Transferencia, donde se utiliza la ecuación:

$$Y(s) = H(s) \cdot U(s) \tag{1}$$

Donde Y es el sistema, H es la Función de Transferencia y U es la entrada al sistema.

2.2. SLI de Primer Orden

Los Sistemas Lineales Invariantes de Primer Orden corresponde a un tipo de sistemas que se pueden identificar cuando su respuesta al escalón forman una curva que inicialmente comienza en un punto y crece hasta estabilizarse en un valor límite. También puede darse un caso contrario, donde la curva inicialmente comienza en un valor correspondiente a una amplitud y luego decrece hasta estabilizarse a un valor límite. Un SLI de primer orden es estable si cumplen las siguientes características:

- Ante una entrada acotada, la salida también es acotada.
- La respuesta al impulso tiende al infinito.
- Los polos de la Función de Transferencia tienen parte real estrictamente negativa.

2.3. SLI de Segundo Orden

Los Sistemas Lineales Invariantes de Segundo Orden corresponde a un tipo de sistemas que pueden ser identificados cuando provienen de una ecuación de orden 2. Existen dos tipos de Sistemas de Segundo Orden: Los sistemas sobreamortiguados y los sistemas subamortiguados.

2.4. Función de Transferencia

Una función de Transferencia, la cual comúnmente es denominada como $H(s)$, corresponde a una herramienta muy esencial a la hora de analizar la respuesta a un sistema. En palabras simples, las Funciones de Transferencia suelen ser ecuaciones matemáticas que describen como un sistema se comporta ante una cierta entrada. Cuando se trabaja con Funciones de Transferencia también se trabaja con Transformadas de Laplace.

2.5. Transformada de Laplace

La Transformada de Laplace es una expresión matemática utilizada para convertir funciones de dominio en el tiempo en funciones del dominio complejo de Laplace. Es una transformación bastante útil a la hora de trabajar con ecuaciones diferenciales lineales y sistemas lineales de dominio en el tiempo. Principalmente, las Transformadas de Laplace *son utilizadas ya que permiten convertir ecuaciones diferenciales complejas en ecuaciones algebraicas que no suelen ser complicadas, permitiendo realizar cálculos más simples y usando menos tiempo.* (Zill (2009)).

2.6. Lazo abierto

El lazo abierto es un sistema de control en el que la salida del sistema no tiene influencia sobre la entrada o la acción de control. En otras palabras, *es aquél que la salida del proceso no es comparada con la señal de referencia.* (Castaño (2019)).

2.7. Lazo cerrado

El lazo cerrado es un sistema de control en el que la salida del sistema se utiliza para ajustar la entrada o la acción de control. Es decir, *toma la salida del proceso y la compara con la señal de referencia para conocer en todo momento la evolución de la variable.* (Castaño (2019)).

2.8. Ceros de la Función de Transferencia

Los ceros de una Función de Transferencia son aquellos valores de la variable compleja en los que la Función de Transferencia se hace cero. (Castaño (2020a)) En palabras simples, si consideramos a $H(s)$ como la Función de Transferencia, los ceros son aquellos valores de s que hacen que $H(s)$ sea cero, es decir, las raíces del numerador.

2.9. Polos de la Función de Transferencia

Los polos de una Función de Transferencia son aquellos valores de la variable compleja en los que la Función de Transferencia se indefine. En palabras simples, si consideramos a $H(s)$ como la Función de Transferencia, los polos son aquellos valores de s que hacen que $H(s)$ se indefine, es decir, las raíces del denominador.

2.10. Ganancia Estática

El concepto de ganancia estática se refiere a la relación entre la amplitud de la señal de entrada y la amplitud de la señal de salida del sistema cuando la frecuencia de la señal de entrada es cero. En otras palabras: *Se denomina ganancia estática de un sistema a la la relación de ganancia entre la entrada y la salida del proceso. Es decir, cuando la entrada es constante (escalón) y la salida se estabiliza (régimen permanente), la razón del cambio de la salida entre el cambio de la entrada nos da la ganancia estática del sistema.* (Castaño (2020b)).

2.11. Tiempo de Estabilización

El tiempo de estabilización indica la rapidez con la que el sistema responde a las señales de entrada y se estabiliza en condiciones de equilibrio. El criterio de aceptación va a depender mucho del contexto del sistema. *Un tiempo de estabilización más rápido puede ser deseable en ciertas aplicaciones, mientras que en otras, un tiempo de estabilización más lento puede ser aceptable.* (Ogata (2010)).

3. Desarrollo

3.1. Primera parte

Para el desarrollo de la primera parte se busca graficar las respuestas de lazo abierto y cerrado para las siguientes funciones:

1. $6y'(t) + 2y(t) = 8u'(t);\ y(0) = 3;\ u(0) = 1$
2. $y''(t) + 6y'(t) + 3y(t) - 5u''(t) - 7u'(t) - u(t) = 0$; $y(0) = 0$; $y'(0) = 0$; $u'(0) = 1$; $u(0) = 0$

3.1.1. Primera función

Comenzando con la primera función, se aplica la Transformada de Laplace a toda la función, quedando de la siguiente forma:

$$6\mathcal{L}(y'(t)) + 2\mathcal{L}(y(t)) = 8\mathcal{L}(u'(t)) \tag{2}$$

Aplicando propiedades y utilizando transformadas conocidas, se llega a la siguiente ecuación:

$$6(sY(s) - y(0)) + 2Y(s) = 8(sU(s) - u(0)) \tag{3}$$

De la ecuación anterior se pueden notar algunos términos que ya son conocidos, gracias a que corresponden a términos iniciales. Reemplazando:

$$6sY(s) - 18 + 2Y(s) = 8sU(s) - 8 \tag{4}$$

Para poder eliminar el 18 del lado izquierdo de la ecuación, se suma 18 a ambos lados:

$$6sY(s) + 2Y(s) = 8sU(s) + 10 \tag{5}$$

Como en el lado izquierdo de la ecuación ambos términos tienen a Y(s), se factoriza por Y(s), por lo que la ecuación queda de la siguiente forma:

$$Y(s)(6s+2) = 8sU(s) + 10 \tag{6}$$

Para despejar $Y(s)$, se puede dividir la ecuación por $(6s+2)$, quedando la ecuación de la siguiente forma:

$$Y(s) = \frac{8sU(s)+10}{6s+2} \tag{7}$$

Nótese que se ha llegado a una ecuación de forma bastante agradable, ya que de la teoría, se sabe que:

$$Y(s) = RESC + RENC \tag{8}$$

Si se aplica lo anterior, es posible identificar RESC y RENC en la ecuación 6. Debido a que se busca obtener la Función de Transferencia $H_1(s)$, se puede ignorar el RENC y despejar $H_1(s)$ desde el RESC.

$$H_1(s) = \frac{Y(s)}{U(s)} = \frac{8s}{6s+2} = \frac{4s}{3s+1} \tag{9}$$

Ahora que se tiene la función de transferencia $H_1(s)$, es necesario graficar las respuestas de lazo abierto y lazo cerrado.

Dada la naturaleza del sistema y de la función de Transferencia, $H_1(s)$ por si solo ya corresponde a la misma expresión racional del lazo abierto. Para poder obtener la expresión racional del lazo cerrado, se puede utilizar la función **feedback** de MATLAB, utilizando como entradas la función de Transferencia $H_1(s)$ y 1. La expresión racional correspondiente al lazo cerrado es:

$$\frac{4s}{7s+1}$$

Esto se comprueba con el siguiente código que realiza lo descrito anteriormente en MATLAB, como se puede ver en la figura a continuación:

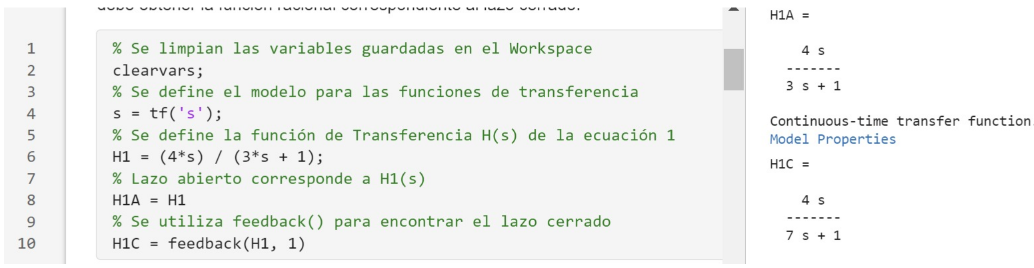

Figura 1: Código realizado en MATLAB para obtener el lazo abierto y cerrado de $H_1(s)$.

Donde H1A representa al lazo abierto y H1C representa al lazo cerrado. Con la función **step** de MATLAB se puede obtener un gráfico el cual muestra la respuesta de lazo abierto o cerrado a un escalón a la función de transferencia $H_1(s)$. Realizando esto para ambos lazos, se obtienen los siguientes gráficos.

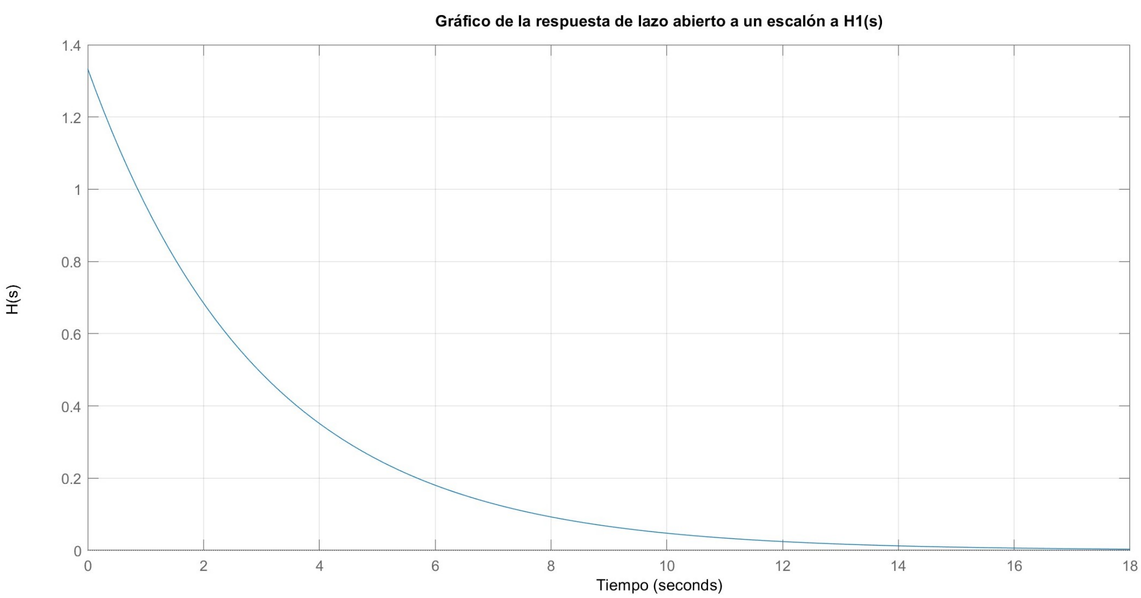

Figura 2: Gráfico para la respuesta de lazo abierto ante un escalón a $H_1(s)$.

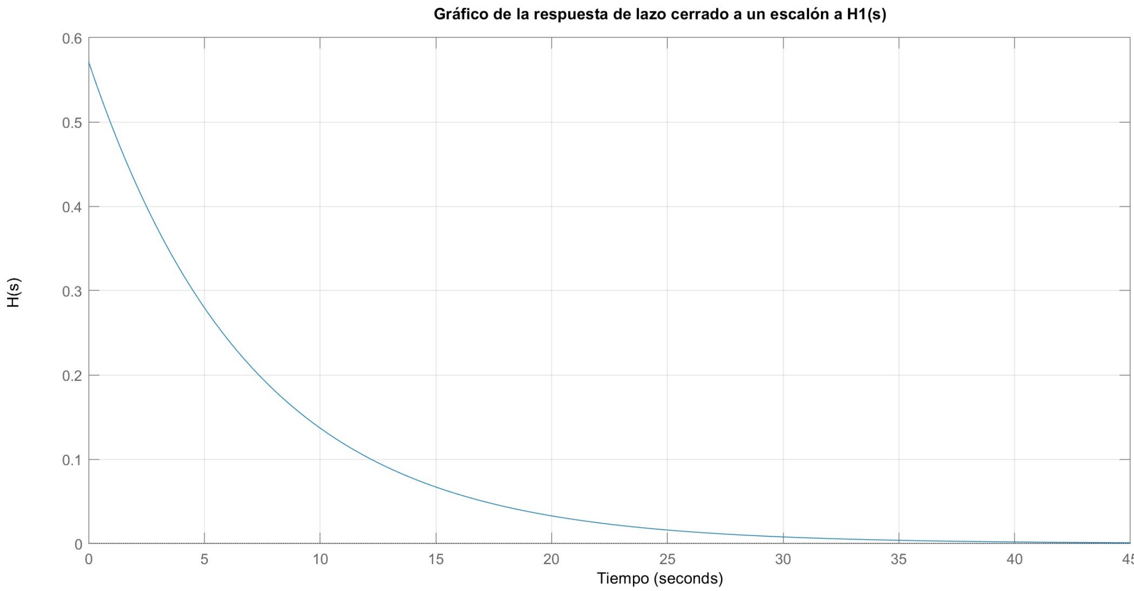

Figura 3: Gráfico para la respuesta de lazo cerrado ante un escalón a $H_1(s)$.

El código realizado en MATLAB para generar ambos gráficos es el siguiente:

```
% Gráfico para el lazo abierto usando step()
figure;
step(H1A);
title("Gráfico de la respuesta de lazo abierto a un escalón a H1(s)")
xlabel("Tiempo")
ylabel("H(s)")
grid("on");
```

Ahora se repite el mismo proceso para la respuesta de lazo cerrado.

```
% Gráfico para el lazo cerrado usando step()
figure;
step(H1C);
title("Gráfico de la respuesta de lazo cerrado a un escalón a H1(s)")
xlabel("Tiempo")
ylabel("H(s)")
grid("on");
```

Figura 4: Código realizado en MATLAB para obtener los gráficos anteriores.

La ganancia estática de una función de transferencia se obtiene cuando la variable s es igual a cero, entonces se puede obtener mediante la función **dcgain** de MATLAB o al reemplazar en la respuesta de lazo abierto:

$$\frac{0}{1} \Leftrightarrow 0 \tag{10}$$

Mientras que para la respuesta de lazo cerrado:

$$\frac{0}{1} \Leftrightarrow 0 \tag{11}$$

Como se mencionó anteriormente, se puede comprobar los resultados anteriores utilizando las funciones dcgain de MATLAB, tal como se ve en la figura a continuación:

```
% Se calculan la ganancia estática de cada lazo.
K_H1A = dcgain(H1A)
K_H1C = dcgain(H1C)
```

```
K_H1A = 0
K_H1C = 0
```

Figura 5: Código realizado en MATLAB para obtener las ganancias estáticas.

Donde K_H1A corresponde a la ganancia estática del lazo abierto y K_H1C corresponde a la ganancia estática del lazo cerrado. Utilizando las funciones **zero** y **pole** de MATLAB se pueden obtener los ceros y polos para cada lazo. Estos valores se pueden ver en la Tabla N°1 más adelante y también en el código de MATLAB a continuación.

```
% Se obtienen los ceros y polos para cada lazo
ceros_H1A = zero(H1A)
polos_H1A = pole(H1A)

ceros_H1C = zero(H1C)
polos_H1C = pole(H1C)
```

```
ceros_H1A = 0
polos_H1A = -0.3333

ceros_H1C = 0
polos_H1C = -0.1429
```

Figura 6: Código realizado en MATLAB para obtener los ceros y polos.

Donde ceros_H1A corresponde a los ceros del lazo abierto, ceros_H1C son los ceros del lazo cerrado, polos_H1A son los polos del lazo abierto y polos_H1C son los polos del lazo cerrado. Por último, para obtener el tiempo de estabilización de cada lazo, se utiliza la función **stepinfo** de MATLAB. Esta función nos entrega varios atributos, pero el atributo correspondiente al tiempo de estabilización es **SettlingTime**. En MATLAB se obtiene el siguiente resultado:

```
T_H1A = stepinfo(H1A).SettlingTime
T_H1C = stepinfo(H1C).SettlingTime
```

```
T_H1A = NaN
T_H1C = NaN
```

Figura 7: Código realizado en MATLAB para obtener los tiempos de estabilización.

Donde T_H1A es el tiempo de estabilización del lazo abierto y T_H1C es el tiempo de estabilización del lazo cerrado. Como se puede observar en la figura anterior, ambos dan como resultado **NaN**. Esto se debe a un problema que existe con la función, ya que de acuerdo a la documentación de la misma, *SettlingTime es calculado con un rango de error* (MathWorks (2022)) que nunca cumple cuando es un valor cercano al cero, haciendo que de como resultado NaN. Esto sucede tanto para lazo abierto y lazo cerrado (en consecuencia, se procede a obtener el tiempo de estabilización durante el análisis de los gráficos más abajo).

Respecto a los gráficos, se puede ver que el gráfico para el lazo abierto, correspondiente a la figura 2, tiene una forma de un SLI de primer orden, ya que comienza desde un punto alto hasta que se estabiliza cercano a un valor límite. En este caso, el gráfico comienza desde una amplitud cercana a 1.4, para luego bajar hasta estabilizarse cercano a 0. El sistema cumple las condiciones para ser estable, ya que sus polos son estrictamente negativos y reales, además de que tiende a 0 a medida que va pasando el tiempo, por lo que se podría decir que el tiempo de estabilización correspondiente al lazo abierto es de 15 segundos, donde la curva ya se estabiliza muy cercana a 0.

Para el caso del gráfico del lazo cerrado, correspondiente a la figura 3, nuevamente tiene la forma de un SLI de primer orden, pero la principal diferencia comparado con el lazo abierto es que su amplitud es de 0.6 y si bien la gráfica también baja hasta estabilizarse cercano a 0, lo hace mucho más después, teniendo un tiempo de estabilización cercano a los 35 segundos. Este sistema también cumple las condiciones para ser estable, ya que los polos también son estrictamente negativos y reales y tiende a 0 a medida que pasa el tiempo.

También es posible graficar los polos y ceros de ambos lazos en el plano complejo. Para ello, debemos utilizar la función pzplot() de MATLAB, la cual recibe las funciones de transferencia de ambos lazos. Los polos corresponden se representan con la letra x, mientras que los ceros con la letra o. Además, el polo y cero del lazo abierto están de color azul, mientras que del polo cerrado están en color rojo.

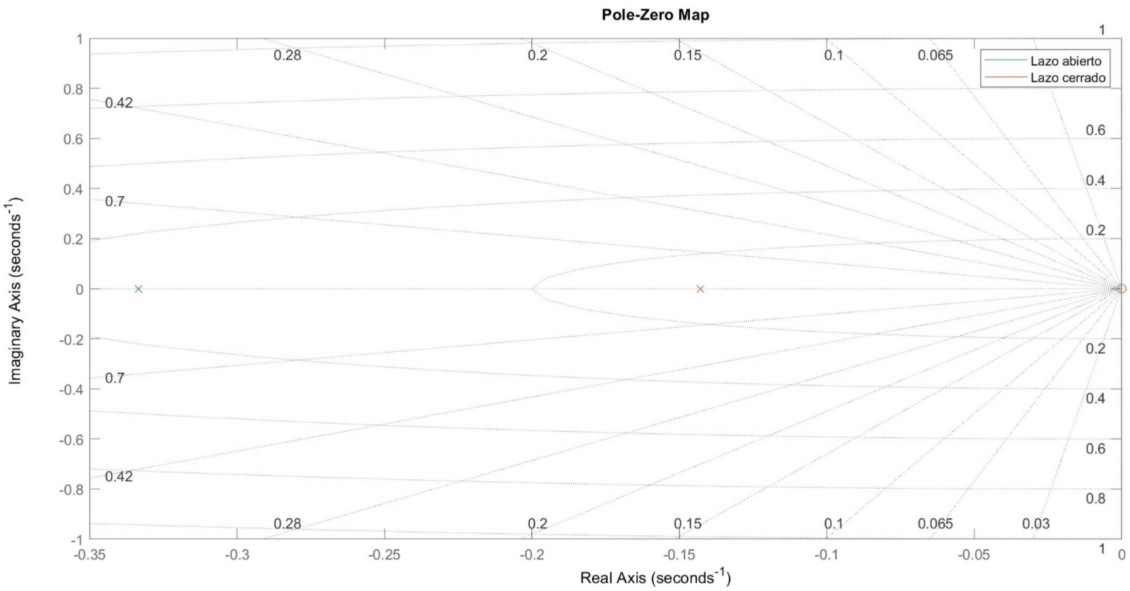

Figura 8: Gráfico de polos y ceros de ambos lazos de $H_1(s)$.

La variable H1pz corresponde al gráfico generado gracias a la función pzplot, tal como se puede observar en el código a continuación:

```
% Gráfico de polos y ceros de ambos lazos
H1pz = pzplot(H1A, H1C);
legend("Lazo abierto", "Lazo cerrado")
grid on
```

Figura 9: Código realizado en MATLAB para graficar polos y ceros de ambos lazos de $H_1(s)$.

3.1.2. Segunda función

Luego, de manera similar se desarrolla la segunda función y se aplica la Transformada de Laplace a ambos lados de la ecuación:

$$\mathcal{L}(y''(t)) + 6\mathcal{L}(y''(t)) + 3\mathcal{L}(y(t)) = 5\mathcal{L}(u''(t)) + 7\mathcal{L}(u'(t)) + \mathcal{L}(u(t)) \tag{12}$$

Con la aplicación de las propiedades de derivación queda de la siguiente forma:

$$s^2Y(s) - sy(0) - y'(0) + 6(sY(s) - y(0)) + 3Y(s) = 5(s^2U(s) - su(0) - u'(0)) + 7(sU(s) - u(0)) + U(s) \tag{13}$$

Al reemplazar los valores de las condiciones iniciales dados en el enunciado:

$$s^2Y(s) + 6sY(s) + 3Y(s) = 5(s^2U(s) - 1) + 7sU(s) + U(s) \tag{14}$$

Se distribuyen y factorizan términos para lograr despejar $Y(s)$:

$$s^2Y(s) + 6sY(s) + 3Y(s) = 5s^2U(s) - 5 + 7sU(s) + U(s) \tag{15}$$

$$Y(s)(s^2 + 6s + 3) = U(s)(5s^2 + 7s + 1) - 5 \tag{16}$$

$$Y(s) = \frac{U(s)(5s^2 + 7s + 1) - 5}{s^2 + 6s + 3} \tag{17}$$

Considerando la ecuación (7), se separan los términos para encontrar las respuestas RESC y RENC:

$$Y(s) = \frac{5s^2 + 7s + 1}{s^2 + 6s + 3}U(s) - \frac{5}{s^2 + 6s + 3} \tag{18}$$

Luego, la respuesta RESC es aquella que depende de U(s), entonces la función de transferencia del sistema de la segunda función:

$$H_2(s) = \frac{5s^2 + 7s + 1}{s^2 + 6s + 3} \tag{19}$$

De la misma manera, debido a la naturaleza, la función de transferencia racional corresponde a la respuesta de lazo abierto:

$$\frac{5s^2 + 7s + 1}{s^2 + 6s + 3} \tag{20}$$

Entonces, se debe encontrar la respuesta de lazo cerrado mediante la función **feedback** de MATLAB, con las entradas de a función de Transferencia $H_2(s)$ y 1. Por lo tanto:

$$\frac{5s^2 + 7s + 1}{6s^2 + 13s + 4} \tag{21}$$

Esto se comprueba con el siguiente código que realiza lo descrito anteriormente en MATLAB, como se puede ver en la figura a continuación:

```
% Se define la función de transferencia H(s) de la ecuación 2
H2 = (5*s^2 + 7*s + 1)/(s^2 + 6*s + 3);
% Lazo abierto corresponde a H(s)
H2A = H2
% Se utiliza feedback para encontrar el lazo cerrado
H2C = feedback(H2, 1)
```

Figura 10: Código realizado en MATLAB para obtener el lazo abierto y cerrado de $H_2(s)$.

Donde H2A representa al lazo abierto y H2C representa al lazo cerrado. Mediante la función **step** de MATLAB se grafican ambas respuestas de la función de transferencia $H_2(s)$ frente al escalón unitario.

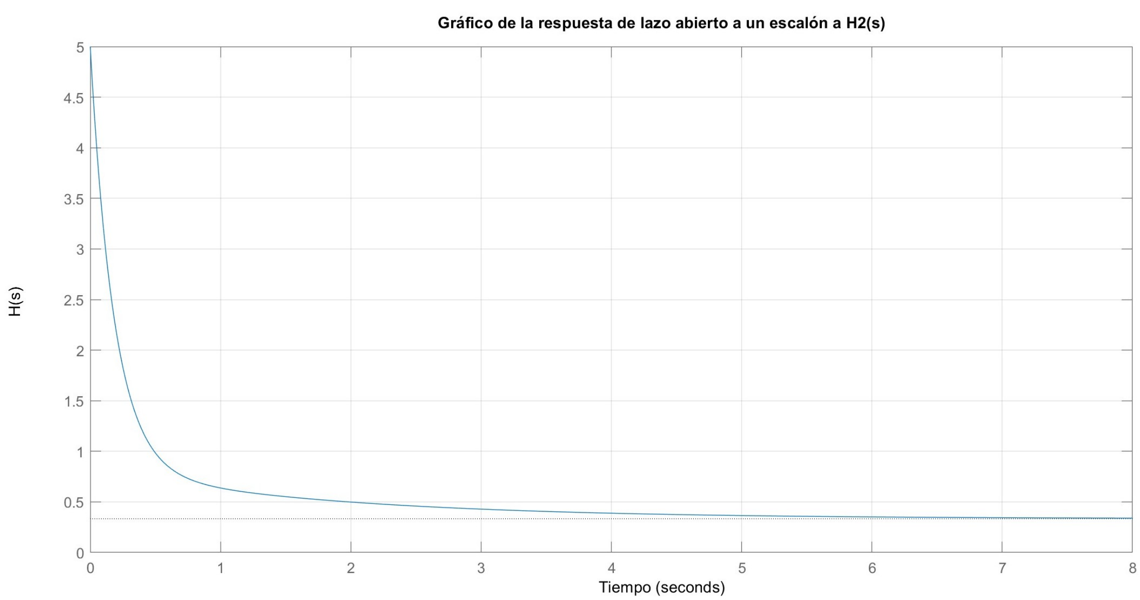

Figura 11: Gráfico para la respuesta de lazo abierto ante un escalón a $H_2(s)$.

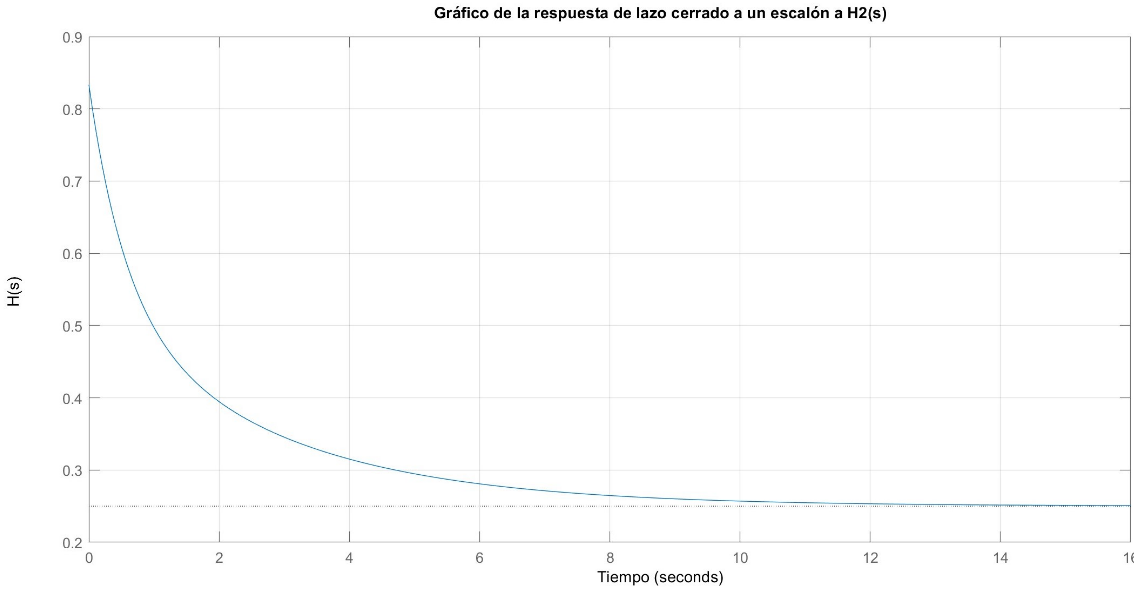

Figura 12: Gráfico para la respuesta de lazo cerrado ante un escalón a $H_2(s)$.

El código realizado en MATLAB para generar ambos gráficos es el siguiente:

```
% Gráfico para el lazo abierto usando step()
figure;
step(H2A);
title("Gráfico de la respuesta de lazo abierto a un escalón a H2(s)")
xlabel("Tiempo")
ylabel("H(s)")
grid("on");
```

Se repite el mismo proceso para la respuesta del lazo cerrado.

```
% Gráfico para el lazo cerrado usando step()
figure;
step(H2C);
title("Gráfico de la respuesta de lazo cerrado a un escalón a H2(s)")
xlabel("Tiempo")
ylabel("H(s)")
grid("on");
```

Figura 13: Código realizado en MATLAB para obtener los gráficos anteriores.

Se obtiene la ganancia estática al reemplazar la variable s por cero en la respuesta de lazo abierto:

$$\frac{1}{3} \Leftrightarrow 0,3333 \tag{22}$$

Luego, para la respuesta de lazo cerrado:

$$\frac{1}{4} \Leftrightarrow 0,25 \tag{23}$$

Se puede comprobar los resultados anteriores con MATLAB, tal como se ve en la figura a continuación:

```
% Se calcula la ganancia estática para ambos lazos
K_H2A = dcgain(H2A)
K_H2C = dcgain(H2C)
```

```
K_H2A = 0.3333
K_H2C = 0.2500
```

Figura 14: Código realizado en MATLAB para obtener las ganancias estáticas.

Donde K_H2A corresponde a la ganancia estática del lazo abierto y K_H2C corresponde a la ganancia estática del lazo cerrado. Además, se utilizan las funciones **zero** y **pole** de MATLAB para obtener los ceros y polos de las respuestas de lazo abierto y lazo cerrado. El código de MATLAB es el siguiente:

```
% Se obtienen los ceros y polos para cada lazo
ceros_H2A = zero(H2A)
polos_H2A = pole(H2A)

ceros_H2C = zero(H2C)
polos_H2C = pole(H2C)
```

Figura 15: Código realizado en MATLAB para obtener los ceros y polos.

Donde ceros_H2A corresponde a los ceros del lazo abierto, ceros_H2C son los ceros del lazo cerrado, polos_H2A son los polos del lazo abierto y polos_H2C son los polos del lazo cerrado. Estos valores se pueden ver en la tabla N°1 más adelante. Por último, se utiliza stepinfo de MATLAB para obtener el tiempo de estabilización, donde se obtiene el siguiente resultado:

```
T_H2A = stepinfo(H2A).SettlingTime
T_H2C = stepinfo(H2C).SettlingTime
```

```
T_H2A = 7.8289
T_H2C = 10.9049
```

Figura 16: Código realizado en MATLAB para obtener los tiempos de estabilización.

Donde T_H2A es el tiempo de estabilización del lazo abierto y T_H2C es el tiempo de estabilización del lazo cerrado. Se puede ver que esta vez stepinfo si entrega resultados, donde el tiempo de estabilización del lazo abierto es 7,83 segundos, lo cual, si se observa gráfico correspondiente a la figura 11, tiene bastante sentido, especialmente si se utiliza MATLAB para mirar el gráfico con más detalle, ya que se puede observar que la gráfica si se estabiliza cercano a los 7 segundos. Para el caso del lazo cerrado, se tiene que el tiempo de estabilización corresponde a 11 segundos, lo cual nuevamente tiene mucha relación con el gráfico obtenido, correspondiente a la figura 12.

Respecto a los gráficos, se puede observar que el gráfico para el lazo abierto, tiene una forma de un SLI de segundo orden, específicamente, un SLI de segundo orden sobre-amortiguado, ya que comienza desde un punto alto hasta que se estabiliza cercano a un valor límite y no tiene oscilaciones. En este caso, el gráfico comienza desde una amplitud cercana a 5, para luego bajar hasta estabilizarse cercano a 0.4. El sistema cumple las condiciones para ser estable, ya que ambos polos son estrictamente negativos y reales y si bien no tiende estrictamente a 0, tiende a un valor bastante cercano a 0, por lo que se podría decir que efectivamente corresponde a un SLI de segundo orden estable, cuyo tiempo de estabilización es 7,83 segundos.

Para el caso del gráfico del lazo cerrado, este también tiene la forma de un SLI de segundo orden sobreamortiguado, pero la principal diferencia comparado con el lazo abierto es que su amplitud es de un valor cercano a 0.9 y además, la gráfica baja a estabilizarse a un valor más bajo que la del lazo abierto, bajando hasta 0.25 aproximadamente y se estabiliza mucho después, aproximadamente en los 11 segundos, lo cual tiene relación con el resultado obtenido mediante MATLAB. Este sistema también cumple las condiciones para ser estable, ya que los polos también son estrictamente negativos y reales y tiende a un valor cercano a 0 a medida que pasa el tiempo.

Nuevamente, se pueden graficar tanto los polos y ceros del lazo abierto como del lazo cerrado. Los polos se representan con la letra x y los ceros con la letra o. El lazo abierto es representado con el color azul mientras que el lazo cerrado se representa con el color rojo.

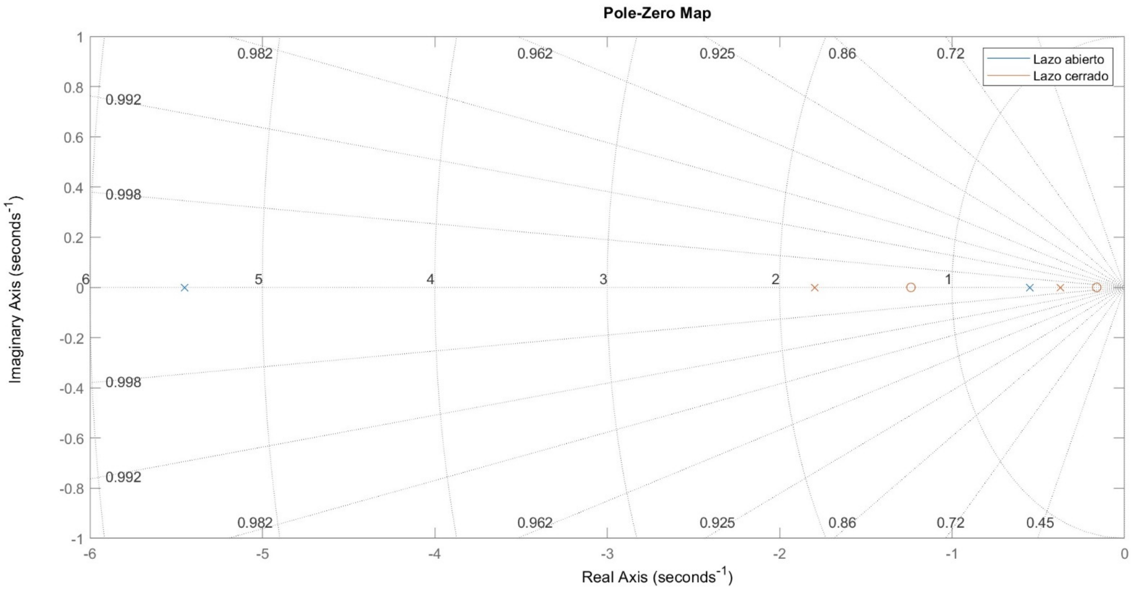

Figura 17: Gráfico de polos y ceros de ambos lazos de $H_2(s)$.

La variable H2pz corresponde al gráfico generado gracias a la función pzplot, tal como se puede observar en el código a continuación:

```
% Gráfico de polos y ceros de ambos lazos
H2pz = pzplot(H2A, H2C);
legend("Lazo abierto", "Lazo cerrado")
grid on
```

Figura 18: Código realizado en MATLAB para graficar polos y ceros de ambos lazos de $H_2(s)$.

3.1.3. Comparaciones entre los resultados de ambas funciones

A continuación, se realiza una tabla comparativa para los valores de ganancia estática, tiempo de estabilización, ceros y polos para las respuestas de lazo abierto y lazo cerrado de ambas funciones.

Respuesta	Ganancia estática	Tiempo de estabilización [s]*	Ceros	Polos
$H_1(s)$ L. A.	0	15	0	-0,33
$H_1(s)$ L. C.	0	35	0	-0,14
$H_2(s)$ L. A.	0,33	7,83	-1,24 y -0,16	-5,45 y -0,55
$H_2(s)$ L. C.	0,25	11	-1,24 y -0,16	-1,80 y -0,37

Tabla N°1: Comparaciones entre lazos abiertos y lazos cerrados.

*: Se debe tener en consideración que, debido al hecho de que stepinfo entrega como resultado NaN para $H_1(s)$, los tiempos de estabilización obtenidos corresponden a un aproximado.

De la tabla anterior se pueden desprender varios datos importantes:

- La ganancia estática para $H_1(s)$ es 0 para ambos lazos, en cambio para $H_2(s)$ es mayor la del lazo abierto.

- Si se comparan los tiempos de estabilización de $H_1(s)$ y $H_2(s)$, se puede notar que los tiempos de $H_1(s)$ son mayores. Sin embargo, si se compara lazo abierto con lazo cerrado, se puede observar que tanto en $H_1(S)$ como en $H_2(s)$ el tiempo del lazo cerrado es mayor que el del lazo abierto.

- $H_1(s)$ no tiene ceros, en cambio $H_2(s)$ tanto en lazo abierto como en lazo cerrado tiene dos ceros.

- $H_1(s)$ tiene un solo polo tanto en lazo abierto como en lazo cerrado, en cambio, $H_2(s)$ tiene dos polos en ambos lazos.

Como se mencionó anteriormente, los datos de la tabla fueron obtenidos gracias a un código realizado en MATLAB, correspondiente a la figura 1 hasta la figura 18.

3.2. Segunda parte

Se pide graficar la respuesta de un sistema frente al escalón con el uso de la función **step** de MATLAB y encontrar el valor de la función de transferencia. Por lo tanto, se debe analizar el diagrama de bloque de la Figura 19 para identificar el tipo de conexión entre las funciones de transferencia.

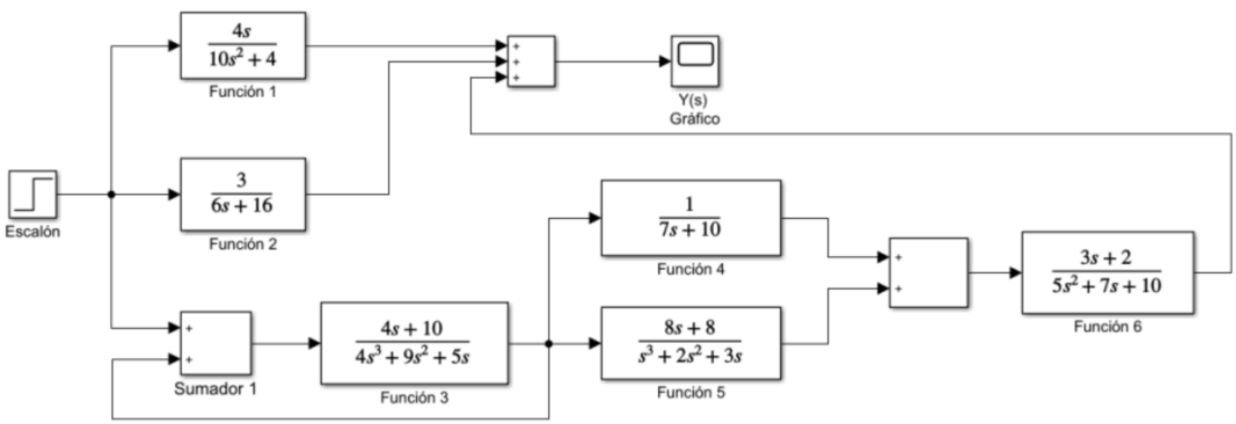

Figura 19: Diagrama de bloque.

En primer lugar, la salida del sistema tiene un sumador entre la función 1, la función 2 y la salida de la función 6, por lo que poseen una conexión en paralelo y se deben sumar los valores de sus funciones de transferencia, pero dado que las dos primeras son conocidas, se debe enfocar en calcular el valor de la función del tercer parámetro del sumador.

Se puede notar que la función 3 recibe como entrada un sumador con una conexión de retroalimentación y la función de escalón unitario, por lo que la retroalimentación es positiva y se puede reemplazar con 1. Entonces, en MATLAB se hace uso de la función **feedback** con los parámetros $F_3(s) = \frac{4s+10}{4s^3+9s^2+5s}$ y 1.

```
% La función 3 recibe una retroalimentación positiva
H_3 = feedback(F_3,1);
```

Figura 20: Línea de código para la retroalimentación en F_3.

Luego, para la función 4 y la función 5 se tiene una conexión en serie, cuya entrada es la retroalimentación de la función 3, por lo tanto se debe multiplicar este valor con la función de transferencia respectiva.

```
% Función 4 y función 5 poseen una conexión en serie, cuya entrada es la
% retroalimentación de la función 3
H_4 = F_4 + F_5;
H_5 = H_4 * H_3;
```

Figura 21: Líneas de código para conexión en serie.

Además, la transferencia anterior tiene una conexión en serie con la función 6.

```
% La función 6 posee una conexión en paralelo, cuyas entradas son la
% función 4 y 5.
H_6 = H_5 * F_6;
```

Figura 22: Línea de código para conexión en serie.

Por lo tanto, queda la siguiente expresión:

$$H(s) = f_1 + f_2 + (feedback(f_3, 1) \cdot f_4 + feedback(f_3, 1) \cdot f_5) \cdot f_6 \tag{24}$$

Y en el código de MATLAB:

```
H = F_1 + F_2 + H_6
```

Figura 23: Línea de código para calcular función de transferencia del diagrama de bloque.

Por lo tanto, la Función de Transferencia es:

```
H =

  7560 s^11 + 62474 s^10 + 259936 s^9 + 706088 s^8 + 1.407e06 s^7 + 2.3e06 s^6 + 3.176e06 s^5 + 3.325e06 s^4 + 2.366e06 s^3
                                                                              + 1.168e06 s^2 + 446880 s + 102400
  ----------------------------------------------------------------------------------------------------------------
  8400 s^12 + 81860 s^11 + 378400 s^10 + 1.124e06 s^9 + 2.376e06 s^8 + 3.755e06 s^7 + 4.562e06 s^6 + 4.269e06 s^5 + 3.073e06 s^4
                                                                            + 1.657e06 s^3 + 641600 s^2 + 192000 s
```

Figura 24: Función de Transferencia para el diagrama de bloques.

Para obtener la salida del sistema cuando la entrada es el escalón, se multiplica por la transformada de Laplace de dicha función.

```
Y = H * 1/s
```

Figura 25: Línea de código para la salida del sistema frente al escalón unitario

Cuyo resultado corresponde a:

```
Y =

  7560 s^11 + 62474 s^10 + 259936 s^9 + 706088 s^8 + 1.407e06 s^7 + 2.3e06 s^6 + 3.176e06 s^5 + 3.325e06 s^4 + 2.366e06 s^3
                                                                                      + 1.168e06 s^2 + 446880 s + 102400
  ------------------------------------------------------------------------------------------------------------------------
  8400 s^13 + 81860 s^12 + 378400 s^11 + 1.124e06 s^10 + 2.376e06 s^9 + 3.755e06 s^8 + 4.562e06 s^7 + 4.269e06 s^6 + 3.073e06 s^5
                                                                                  + 1.657e06 s^4 + 641600 s^3 + 192000 s^2
```

Figura 26: Salida del sistema frente al escalón unitario

Finalmente, con la función **step()** de MATLAB se gráfica la respuesta del sistema frente al escalón unitario.

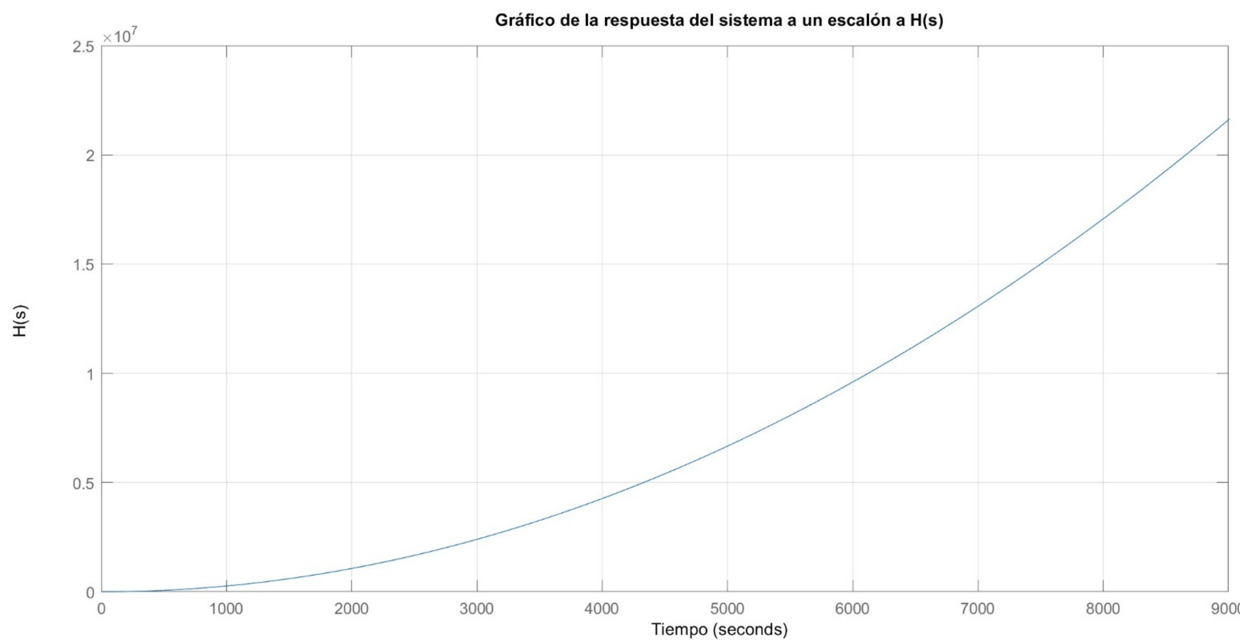

Figura 27: Gráfico para la respuesta del sistema frente al escalón.

Se puede apreciar que el gráfico es similar a una exponencial positiva, la cual parte desde 0 y a medida que va avanzando el tiempo esta linea crece hacia el infinito. Claramente, tanto por la forma del gráfico como por la Función de Transferencia obtenida, este sistema no corresponde a un sistema de primer orden ni de segundo orden, a diferencia de los sistemas obtenidos en la parte 1. Sin embargo cabe destacar de que el gráfico si tiene una forma similar a un gráfico de un sistema de primer orden que no logra estabilizarse y al inicio de un gráfico de un sistema de segundo orden sobreamortiguado.

4. Conclusión

En conclusión, se cumple completamente el objetivo de la actividad, dado que se realiza el análisis de tres sistemas lineales por medio de sus funciones de transferencia, mediante cálculos y el uso de funciones de MATLAB.

Además, cabe añadir que se logra determinar el valor de la función de transferencia mediante la aplicación de la Transformada de Laplace, analizando sus respuestas de lazo abierto y lazo cerrado, para posteriormente graficar su comportamiento frente al escalón unitario. También se comparan los valores de ganancia estática, tiempo de estabilización, ceros y polos para las respuestas de lazo abierto y cerrado de ambas funciones entregadas en el enunciado.

Por otro lado, se logra analizar un diagrama de bloque para calcular el valor de su función de transferencia a través del tipo de conexiones que posee, ya sea en serie, con retroalimentación o paralelo, y después graficar su respuesta al escalón.

Respecto al futuro, se espera que los conocimientos adquiridos en esta actividad puedan ser aplicados en los demás laboratorios, sean útiles para tener una mejor base para comprender los próximos contenidos del semestre y poder aplicarlos en la primera evaluación de la asignatura.

Bibliografía

Castaño, S. (2019). Lazo abierto y lazo cerrado. https://controlautomaticoeducacion.com/control-realimentado/lazo-abierto-y-lazo-cerrado/. Artículo Online. Recuperado el 10 de abril de 2023.

Castaño, S. (2020a). Cero de una función de transferencia. https://controlautomaticoeducacion.com/analisis-de-sistemas/cero-de-una-funcion-de-transferencia/. Artículo Online. Recuperado el 10 de abril de 2023.

Castaño, S. (2020b). Sistemas dinámicos de primer orden. https://controlautomaticoeducacion.com/control-realimentado/sistemas-dinamicos-de-primer-orden/. Artículo Online. Recuperado el 10 de abril de 2023.

MathWorks (2022). Stepinfo. https://es.mathworks.com/help/control/ref/lti.stepinfo.html/. Documentación online. Recuperado el 24 de abril de 2023.

Ogata, K. (2010). *Ingeniería de Control Moderna*. Pearson Prentice Hall, 3 edition.

Zill, D. G. (2009). *Ecuaciones Diferenciales con aplicaciones de modelado*. CENGAGE Learning, 9 edition.