

Taller 8

Compensadores en adelanto

Jaime Patricio Chiqui Chiqui
Ingeniería en Telecomunicaciones
Universidad de Cuenca
Cuenca, Ecuador
jpatricio.chiquic@ucuenca.edu.ec

Abstract—Este taller presenta el diseño e implementación de un compensador en adelanto para mejorar el desempeño transitorio de un sistema de control de motor DC. El sistema original presenta una respuesta lenta e inadecuada que no cumple con las especificaciones de sobrepaso máximo y tiempo de establecimiento. Mediante técnicas de diseño basadas en el Lugar Geométrico de Raíces (LGR), se diseñó un compensador en adelanto con cero en $z = -7.1457$ y polo en $p = -7.9543$, junto con una ganancia $K_c = 138.9612$. El sistema compensado alcanza polos dominantes en $s = -4.0000 \pm j5.4575$, logrando un factor de amortiguamiento $\zeta = 0.5912$ y frecuencia natural $\omega_n = 6.7664$ rad/s. Los resultados experimentales en MATLAB demuestran que el compensador cumple satisfactoriamente con las especificaciones de diseño ($M_p = 10\%$, $t_{ss} = 1$ s), reduciendo significativamente el tiempo de respuesta y manteniendo el sobrepaso dentro de los límites establecidos. Adicionalmente, se implementó un compensador para un circuito RLC serie con especificaciones más exigentes ($M_p = 25\%$, $t_{ss} = 50$ ms), demostrando la versatilidad de la técnica.

Index Terms—Compensador en adelanto, control de motor DC, circuito RLC, Lugar Geométrico de Raíces, diseño de controladores, respuesta transitoria, MATLAB

I. INTRODUCCIÓN

Los sistemas de control de motores DC son fundamentales en aplicaciones industriales y robóticas donde se requiere precisión en el posicionamiento, velocidad y torque. Sin embargo, los motores DC presentan características dinámicas inherentes que frecuentemente resultan en respuestas lentas, sobrepaso excesivo o tiempo de establecimiento prolongado [1].

El diseño de compensadores es una técnica esencial en ingeniería de control que permite modificar la dinámica del sistema para cumplir con especificaciones de desempeño específicas. Los compensadores en adelanto son particularmente efectivos para mejorar la respuesta transitoria del sistema, incrementando el ancho de banda y el margen de fase [2].

Un compensador en adelanto introduce un cero y un polo en el sistema, donde el cero está ubicado más cerca del origen que el polo. Esta configuración genera un adelanto de fase positivo en el rango de frecuencias de interés, lo cual mejora la estabilidad relativa y reduce el tiempo de establecimiento del sistema [3].

El presente taller aborda el diseño sistemático de un compensador en adelanto para dos sistemas diferentes:

- **Sistema Principal:** Motor DC con especificaciones de $M_p = 10\%$ y $t_{ss} = 1$ s

- **Sistema Reto:** Circuito RLC serie con especificaciones de $M_p = 25\%$ y $t_{ss} = 50$ ms

II. OBJETIVOS

Los objetivos específicos son:

- Analizar el comportamiento dinámico de los sistemas sin compensar
- Determinar los polos dominantes deseados a partir de especificaciones de desempeño
- Diseñar compensadores en adelanto mediante técnicas del LGR
- Calcular las ganancias de los compensadores para ubicar los polos en posiciones deseadas
- Validar experimentalmente el desempeño de los sistemas compensados

III. MARCO TEÓRICO

A. Compensadores en Adelanto

Un compensador en adelanto es un dispositivo de control cuya función de transferencia tiene la forma general:

$$G_c(s) = K_c \frac{s+z}{s+p} \quad (1)$$

donde K_c es la ganancia del compensador, z es el cero y p es el polo, con la condición $|z| < |p|$ para garantizar el adelanto de fase, como se aprecia en la Fig. 1.

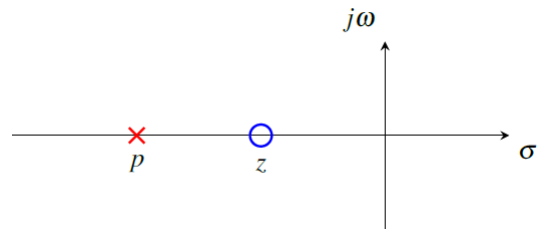


Fig. 1: Lugar geométrico de un compensador en adelanto

El compensador en adelanto proporciona las siguientes ventajas:

- Incrementa el margen de fase del sistema
- Reduce el tiempo de establecimiento
- Mejora la velocidad de respuesta
- Incrementa el ancho de banda

B. Especificaciones de Desempeño Transitorio

Las especificaciones de desempeño transitorio se relacionan con los parámetros del sistema de segundo orden mediante las siguientes ecuaciones fundamentales:

Sobrepaso Máximo (Mp):

$$M_p = e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}} \quad (2)$$

Despejando el factor de amortiguamiento ζ :

$$\zeta = \frac{\ln(1/M_p)}{\sqrt{\pi^2 + \ln^2(1/M_p)}} \quad (3)$$

Tiempo de Establecimiento (tss):

Para un criterio del 2% de la banda de tolerancia:

$$t_{ss} = \frac{4}{\zeta\omega_n} \quad (4)$$

De donde se obtiene la frecuencia natural:

$$\omega_n = \frac{4}{\zeta t_{ss}} \quad (5)$$

C. Diseño Mediante el LGR

El método del Lugar Geométrico de Raíces permite diseñar compensadores mediante los siguientes pasos:

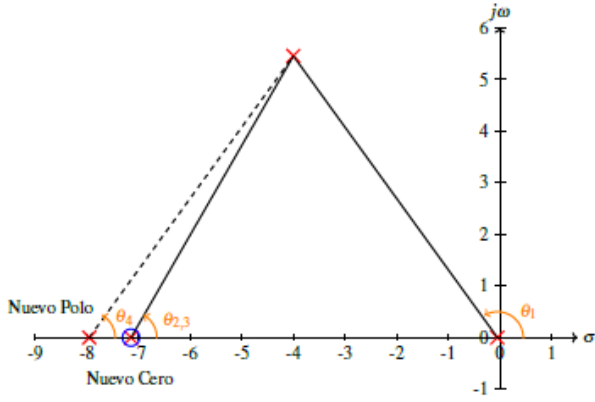


Fig. 2: Compensación de ceros y polos

1. Ubicación de Polos Dominantes:

Los polos dominantes deseados se calculan a partir de las especificaciones:

$$s_{1,2} = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \quad (6)$$

2. Condición Angular:

El compensador debe satisfacer la condición angular en el polo dominante deseado:

$$\angle G_c(s)G(s) = \pm 180(2k+1), \quad k = 0, 1, 2, \dots$$

3. Ubicación del Cero y Polo del Compensador:

El ángulo de adelanto necesario ϕ se calcula mediante geometría del LGR. La pendiente desde el polo original hasta el polo deseado está dada por:

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{\text{Im}(s_d)}{\text{Re}(s_d) - p_{\text{original}}} \quad (7)$$

donde s_d es el polo dominante deseado y p_{original} es un polo del sistema original.

El ángulo θ se calcula como:

$$\theta_1 = 180 + \arctan(m)$$

$$\theta_4 = 180 - \theta_1$$

El polo del compensador se ubica en:

$$p_c = \text{Re}(s_d) - \frac{\text{Im}(s_d)}{\tan(\theta_4)} \quad (8)$$

4. Cálculo de la Ganancia:

La ganancia K_c se determina mediante la condición de magnitud:

$$|G_c(s)G(s)H(s)| = 1 \quad (9)$$

En donde $G_c(s)$, $G(s)$ y $H(s)$ son las funciones de trayectoria directa y de retroalimentación, respectivamente. Resolviendo

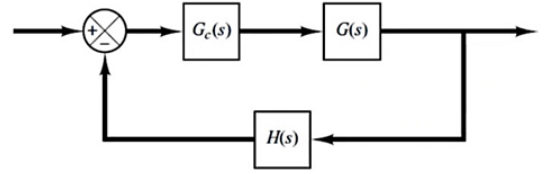


Fig. 3: Modelo del proceso compensado

para K_c , se tiene:

$$K_c = \frac{1}{|G_c(s)G(s)H(s)|} \quad (10)$$

evaluada en el polo dominante deseado s_d .

D. Modelo del Motor DC

En esta práctica, se propone diseñar un compensador para el modelo de un motor DC. Para lograrlo, es esencial recordar el modelo de este tipo de motor, ilustrado en la Figura 4. Este modelo, que se abordó detalladamente en las sesiones teóricas, se define de la siguiente manera:

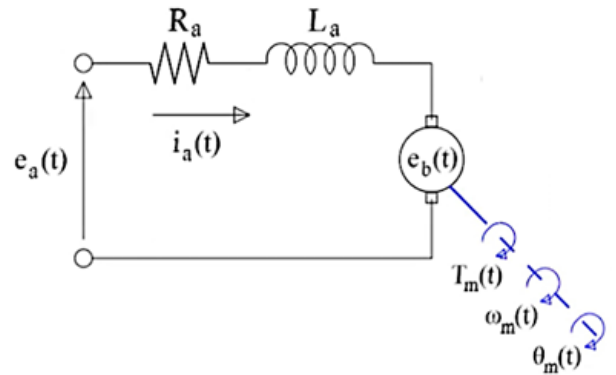


Fig. 4: Modelo del motor de corriente continua (DC).

E. Función de transferencia Motor DC

La Figura 5 muestra el diagrama de bloques del modelo del motor propuesto y la ecuación 11 es su respectiva función de transferencia, la cual representa la velocidad angular ω_m en función del voltaje de armadura E_a .

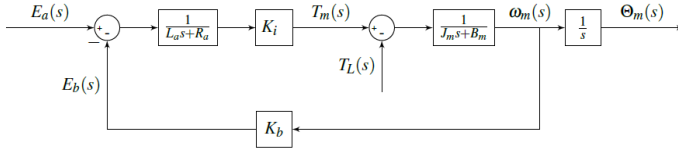


Fig. 5: Diagrama de bloques del modelo del motor DC.

La función de transferencia de un motor DC controlado por armadura está dada por:

$$G(s) = \frac{\omega_m(s)}{E_a(s)} = \frac{K_i / (L_a J_m)}{s^2 + \left(\frac{B_m}{J_m} + \frac{R_a}{L_a} \right) s + \frac{R_a B_m + K_i K_b}{L_a J_m}} \quad (11)$$

donde:

- K_i = constante de torque [Nm/A]
- K_b = constante de fuerza contraelectromotriz [rpm/V]
- R_a = resistencia de armadura [Ω]
- L_a = inductancia de armadura [H]
- J_m = momento de inercia [$\text{kg}\cdot\text{m}^2$]
- B_m = coeficiente de fricción viscosa [$\text{N}\cdot\text{m}\cdot\text{s}$]

F. Modelo del Circuito RLC

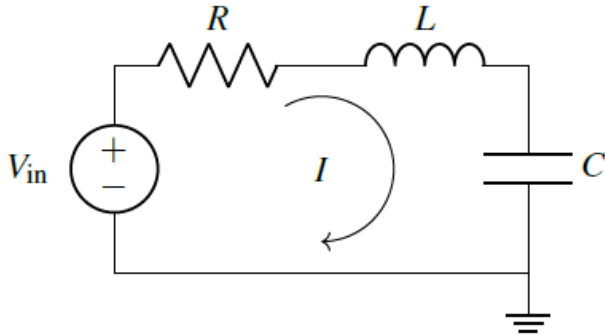


Fig. 6: Circuito RLC en serie. El voltaje de salida es $V_C(t)$.

La función de transferencia de un circuito RLC serie, tomando el voltaje en el condensador como salida, está dada por:

$$G(s) = \frac{V_C(s)}{V_{in}(s)} = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}} \quad (12)$$

donde:

- R = resistencia [Ω]
- L = inductancia [H]
- C = capacitancia [F]

IV. DESARROLLO

A. Sistema Principal: Control de Motor DC

1) **Definición del Sistema y Especificaciones:** **Enunciado:** Definir los parámetros del motor DC y las especificaciones de desempeño requeridas ($M_p = 10\%$, $t_{ss} = 1$ s).

Desarrollo:

El sistema a controlar es un motor DC con los siguientes parámetros físicos:

Parámetros Eléctricos:

- Resistencia de armadura: $R_a = 0.635 \Omega$
- Inductancia de armadura: $L_a = 0.0883$ H

Constantes del Motor:

- Constante de torque: $K_i = 9.43 \times 10^{-3}$ Nm/A
- Constante de FEM: $K_b = 1010$ rpm/V

Parámetros Mecánicos:

- Momento de inercia: $J_m = 330$ $\text{kg}\cdot\text{m}^2$
- Coeficiente de fricción: $B_m = 1 \times 10^{-3}$ $\text{N}\cdot\text{m}\cdot\text{s}$

Especificaciones de Desempeño:

- Sobrepasso máximo: $M_p = 10\% = 0.10$
- Tiempo de establecimiento: $t_{ss} = 1$ s

Listing 1: Definición de parámetros y especificaciones del motor DC

```
1 clc, clear all, close all;
2 % Especificaciones de desempe o
3 Mp = 0.10;
4 tss = 1; %[s]
5 % Parmetros el ctricos
6 Ra = 0.635; % R[Ohm]
7 La = 0.0883; % L[H]
8 % Constantes del motor
9 Ki = 9.43e-3; % [Nm/A]
10 Kb = 1010; % [rpm/V]
11 % Parmetros mec nicos
12 Jm = 330; % [kg m^2]
13 Bm = 1e-3; % [N m s]
```

2) **Función de Transferencia del Sistema Original:** **Enunciado:** Obtener la función de transferencia del motor DC y determinar sus polos.

Desarrollo Matemático:

Aplicando la ecuación (11) con los parámetros dados:

$$G(s) = \frac{K_i / (L_a J_m)}{s^2 + \left(\frac{B_m}{J_m} + \frac{R_a}{L_a} \right) s + \frac{R_a B_m + K_i K_b}{L_a J_m}}$$

Sustituyendo valores numéricos:

$$\text{Numerador: } \frac{9.43 \times 10^{-3}}{0.0883 \times 330} = 3.236 \times 10^{-4}$$

$$\text{Coef. } s^1: \frac{1 \times 10^{-3}}{330} + \frac{0.635}{0.0883} = 7.191$$

$$\text{Coef. } s^0: \frac{0.635 \times 1 \times 10^{-3} + 9.43 \times 10^{-3} \times 1010}{0.0883 \times 330} = 0.3269$$

Por lo tanto, la función de transferencia es:

$$G(s) = \frac{0.0003236}{s^2 + 7.191s + 0.3269}$$

Implementación en MATLAB:

Listing 2: Función de transferencia del sistema

```
1 num = [Ki / (La * Jm)];
2 den = [1 Bm / Jm + Ra / La (Ra * Bm + Ki * Kb) / (La * Jm)];
3
4 Gs = tf(num, den)
```

Salida de MATLAB:

```
Gs =

      0.0003236
-----
      s^2 + 7.191 s + 0.3269
```

Desarrollo Matemático:

Los polos del sistema se obtienen resolviendo la ecuación característica:

$$s^2 + 7.191s + 0.3269 = 0$$

Aplicando la fórmula general para ecuaciones cuadráticas:

$$p_1 = \frac{-7.191 + 7.100}{2} = -0.0457$$

$$p_2 = \frac{-7.191 - 7.100}{2} = -7.1457$$

Implementación en MATLAB:

Listing 3: Cálculo de polos del sistema

```
1 polos = pole(Gs)
2
3 figure()
4 pzmap(Gs)
```

Salida de MATLAB:

```
polos = 2x1
      -7.1457
      -0.0457
```

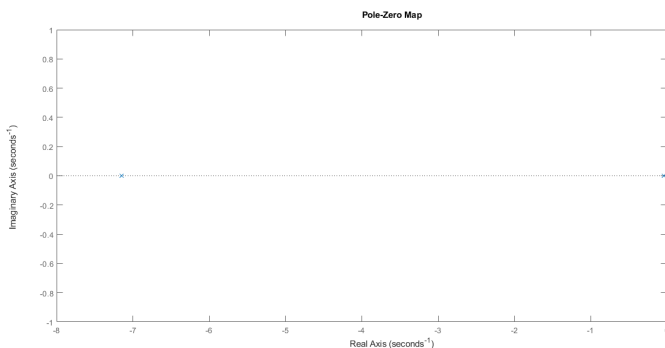


Fig. 7: Ubicación de polos del sistema sin compensar

Los polos del sistema están ubicados en $p_1 = -7.1457$ y $p_2 = -0.0457$. El polo p_2 es dominante por estar más cerca del origen y es el responsable de la respuesta lenta del sistema.

3) *Respuesta del Sistema Sin Compensar:* **Enunciado:** Analizar la respuesta al escalón del sistema sin compensar.

Desarrollo:

Listing 4: Respuesta al escalón del sistema original

```
1 figure()
2 step(Gs)
3 grid on;
```

En la Figura 8 se observa la respuesta al escalón del sistema sin compensar es estable y converge de manera adecuada. Se aprecia que el sistema presenta una respuesta extremadamente lenta, con un tiempo de establecimiento diferente a 1 s, incumpliendo las especificaciones de diseño. La respuesta alcanza el valor final de manera asintótica sin sobrepaso significativo, característica típica de sistemas sobreamortiguados o con polos dominantes muy lentos.

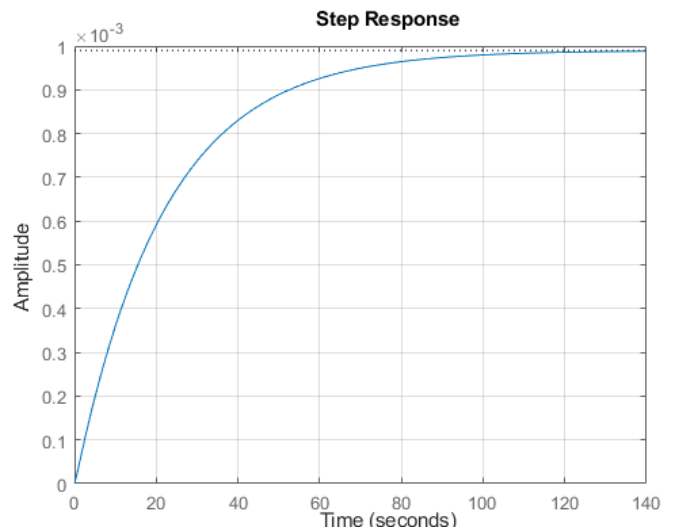


Fig. 8: Respuesta al escalón del sistema sin compensar

4) *Lugar Geométrico de Raíces del Sistema Original:*

Enunciado: Graficar el LGR del sistema sin compensar.

Desarrollo:

Listing 5: LGR del sistema original

```
1 figure()
2 rlocus(Gs)
```

La Figura 9 muestra el Lugar Geométrico de Raíces del sistema original. Se observan dos ramas que inician en los polos $p_1 = -7.1457$ y $p_2 = -0.0457$ y se dirigen hacia el infinito sobre el eje real negativo. No existen ceros finitos en el sistema, por lo que todas las ramas tienden al infinito.

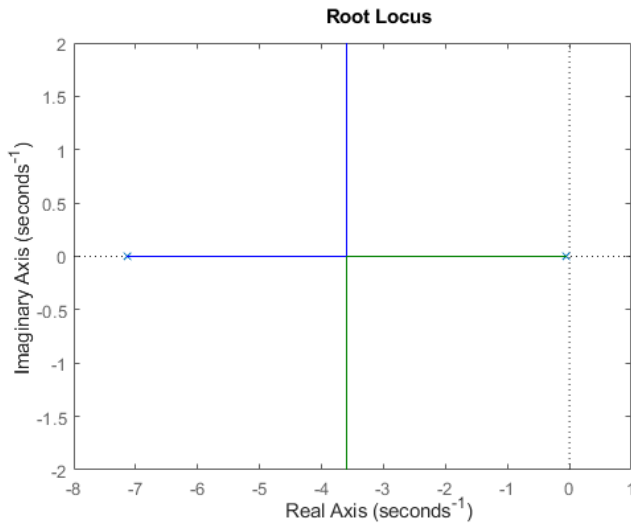


Fig. 9: Lugar Geométrico de Raíces del sistema original

5) *Cálculo de Parámetros de Diseño:* **Enunciado:** Determinar los parámetros ζ y ω_n a partir de las especificaciones.

Desarrollo Matemático:

Aplicando la ecuación (3) para calcular el factor de amortiguamiento:

$$\zeta = \frac{\ln(1/M_p)}{\sqrt{\pi^2 + \ln^2(1/M_p)}} = \frac{\ln(1/0.10)}{\sqrt{\pi^2 + \ln^2(1/0.10)}} = 0.5912$$

Aplicando la ec. (5) para calcular la frecuencia natural:

$$\omega_n = \frac{4}{\zeta_{tss}} = \frac{4}{0.5912 \times 1} = 6.7664 \text{ rad/s}$$

Los polos dominantes deseados se calculan con la ec. (6):

$$s_{1,2} = -\zeta \omega_n \pm j \omega_n \sqrt{1 - \zeta^2}$$

$$s_{1,2} = -0.5912 \times 6.7664 \pm j \cdot 6.7664 \sqrt{1 - 0.5912^2}$$

$$s_{1,2} = -4.0000 \pm j \cdot 6.7664 \times 0.8065 = -4.0000 \pm j5.4575$$

La función de transferencia de segundo orden deseada tiene la forma:

$$G_{s1}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

Sustituyendo los valores calculados:

$$G_{s1}(s) = \frac{(6.7664)^2}{s^2 + 2(0.5912)(6.7664)s + (6.7664)^2} = \frac{45.78}{s^2 + 8s + 45.78}$$

Implementación en MATLAB:

Listing 6: Cálculo de parámetros de diseño

```
1 zeta = log(1/Mp)/sqrt(pi^2+log(1/Mp)^2)
2 wn = 4/(tss*zeta)
3
4 num1 = wn^2;
5 den1 = [1 2*zeta*wn wn^2];
6 Gs1 = tf(num1, den1)
7
8 raices_den1 = roots(den1)
```

Salida de MATLAB:

```
zeta = 0.5912
wn = 6.7664
Gs1 =

         45.78
-----
s^2 + 8 s + 45.78
```

Continuous-time transfer function.

```
raices_den1 = 2x1 complex
-4.0000 + 5.4575i
-4.0000 - 5.4575i
```

Interpretación de los Parámetros Calculados:

El factor de amortiguamiento $\zeta = 0.5912$ corresponde a un sistema subamortiguado que garantiza el sobrepaso máximo del 10%. La frecuencia natural $\omega_n = 6.7664$ rad/s asegura que el tiempo de establecimiento sea de 1 segundo. Los polos deseados en $s = -4.0000 \pm j5.4575$ representan la ubicación objetivo en el plano complejo que el compensador debe alcanzar.

6) *Visualización de Polos Objetivo:* **Enunciado:** Graficar los polos del sistema original y los polos deseados en el LGR. **Desarrollo:**

Listing 7: Superposición de polos objetivo en el LGR

```
1 figure
2 rlocus(Gs)
3 hold on; % Superponer graficos
4 plot(real(raices_den1), imag(raices_den1), 'rx', '
   LineWidth', 2);
5 ylim([-6, 6])
6 hold off;
```

En la Figura 10 se visualizan los polos del sistema original mediante el LGR (líneas azules y verdes) y los polos deseados marcados con cruces rojas (x) en las posiciones $s = -4.0000 \pm j5.4575$. Se observa claramente que los polos deseados no se encuentran sobre las ramas del LGR del sistema original, lo que confirma la necesidad de un compensador para modificar la geometría del LGR.

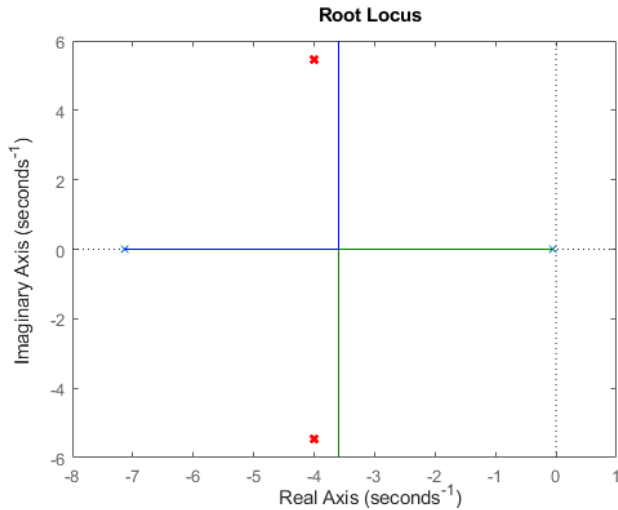


Fig. 10: Polos del sistema original y polos deseados

7) *Diseño del Compensador en Adelanto:* **Enunciado:** Diseñar el compensador en adelanto para ubicar los polos en la posición deseada.

Desarrollo Matemático:

Para diseñar el compensador, se calcula la pendiente desde el polo más cercano al origen ($p_2 = -0.0457$) hasta el polo deseado:

$$m = \frac{\text{Im}(s_d) - 0}{\text{Re}(s_d) - p_2} = \frac{5.4575 - 0}{-4.0000 - (-0.0457)} = -1.3802$$

El ángulo θ_1 se calcula como:

$$\theta_1 = 180 + \arctan(m) = 180 + \arctan(-1.3802) = 125.9253$$

El ángulo de adelanto requerido es:

$$\theta_4 = 180 - \theta_1 = 180 - 125.9253 = 54.0747$$

El polo del compensador se ubica usando la ecuación (8):

$$p_c = \text{Re}(s_d) - \frac{\text{Im}(s_d)}{\tan(\theta_4)} = -4.0000 - \frac{5.4575}{\tan(54.0747)} = -7.9543$$

Para el cero del compensador, se elige cancelar el polo más lento del sistema original:

$$z_c = p_1 = -7.1457$$

Por lo tanto, el compensador (sin ganancia, K_c) es:

$$G_c(s) = \frac{s + 7.1457}{s + 7.9543}$$

Implementación en MATLAB:

Listing 8: Diseño del compensador

```
1 % Pendiente
2 m = (imag(raices_den1(1))-0)/(real(raices_den1(1))-
    polos(2))
```

```
3 theta1 = 180 + rad2deg(atan(m))
4 theta4 = 180 - theta1
5
6 % Polo del compensador
7 p = real(raices_den1(1))-(imag(raices_den1(1))/tand(
    theta4))
8
9 % Cero del compensador
10 z = polos(1)
11
12 num2 = [1 -z];
13 den2 = [1 -p];
14 Glc = tf(num2, den2)
15
16 % Sistema compensado (sin Kc)
17 Glcol = series(Glc, Gs)
```

Salida de MATLAB:

```
m = -1.3802
theta1 = 125.9253
theta4 = 54.0747
```

```
p = -7.9543
z = -7.1457
```

```
Glc =
    s + 7.146
-----
    s + 7.954
Continuous-time transfer function.
```

```
G1col =
    0.0003236 s + 0.002312
-----
    s^3 + 15.15 s^2 + 57.53 s + 2.6
Continuous-time transfer function.
```

8) *Diagrama de Polos y Ceros del Sistema Compensado:* **Enunciado:** Visualizar los polos y ceros del sistema compensado.

Desarrollo:

Listing 9: Diagrama de polos y ceros

```
1 figure()
2 pzplot(G1col)
3 hold on;
4 plot(real(raices_den1), imag(raices_den1), 'ro', '
    LineWidth', 2, 'DisplayName', 'Polos Deseados');
5 grid on; hold off;
6
7 figure()
8 rlocus(G1col)
9 hold on; % Superponer graficos
10 plot(real(raices_den1), imag(raices_den1), 'gs', '
    LineWidth', 2, 'DisplayName', 'Objetivo');
11 grid on;
```

La Figura 11 presenta el diagrama de polos y ceros del sistema compensado (sin la ganancia K_c). Se observan los tres polos del sistema compensado (cruces azules) y el cero introducido por el compensador (círculo azul). Los polos deseados se marcan con círculos rojos para referencia. Se nota que el cero en $z = -7.1457$ está muy cerca del polo

en $p = -7.9543$, lo que resultará en una casi-cancelación de dinámicas rápidas.

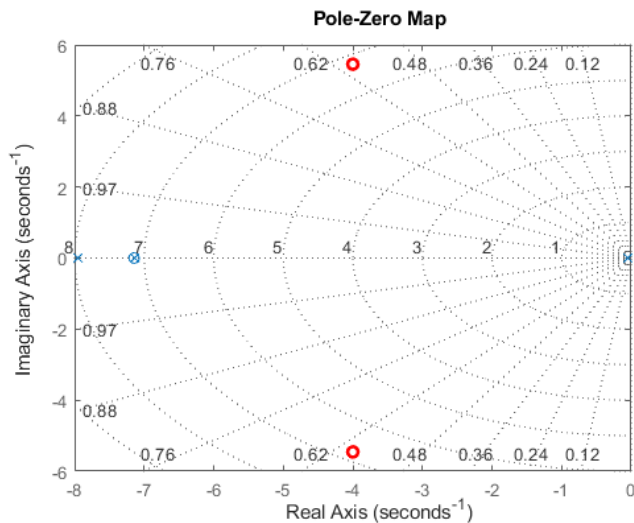


Fig. 11: Polos y ceros del compensador y proceso a lazo abierto, con los polos deseados.

9) *LGR del Sistema Compensado*: **Enunciado:** Graficar el LGR del sistema compensado y verificar que pasa por los polos deseados.

Desarrollo:

Listing 10: LGR del sistema compensado

```
1 figure()
2 rlocus(Glcol)
3 hold on; % Superponer graficos
4 plot(real(raices_den1), imag(raices_den1), 'gs', ...
5      'LineWidth', 2, 'DisplayName', 'Objetivo');
6 grid on;
```

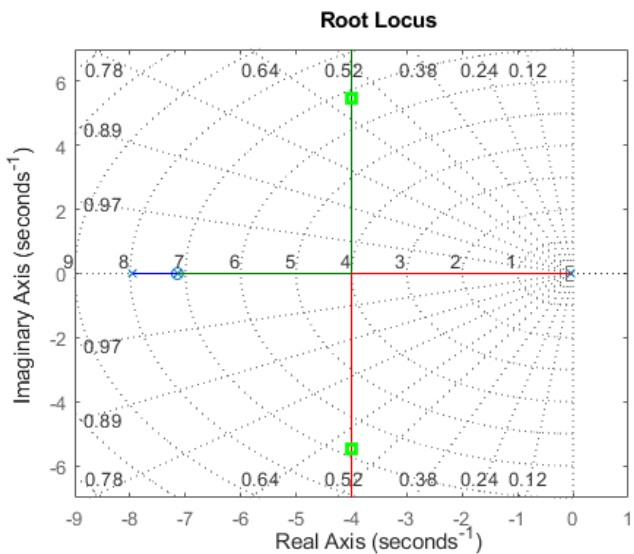


Fig. 12: LGR del sistema compensado con polos objetivo

En la Figura 12 se muestra el LGR del sistema compensado. Se observa claramente que las ramas del LGR ahora pasan

exactamente por los polos deseados (marcados con cuadrados verdes) en $s = -4.0000 \pm j5.4575$. Esto confirma que el diseño geométrico del compensador fue exitoso. Las ramas forman una trayectoria desde los polos originales, pasando por los polos deseados, y dirigiéndose hacia el infinito.

Verificación del Diseño Geométrico:

La superposición de los polos deseados sobre las ramas del LGR del sistema compensado confirma que la ubicación del cero y polo del compensador fue correcta. El siguiente paso es calcular la ganancia K_c necesaria para que los polos en lazo cerrado se ubiquen exactamente en los puntos deseados.

10) *Cálculo de la Ganancia del Compensador*: **Enunciado:** Calcular la ganancia K_c del compensador para ubicar los polos exactamente en la posición deseada.

Desarrollo Matemático:

La ganancia se calcula usando la ecuación (10), evaluando la función de transferencia en lazo abierto en el polo deseado:

$$K_c = \frac{1}{|G_c(s_d) \cdot G(s_d) \cdot H(s_d)|} = \frac{1}{\left| \frac{s_d + 7.1457}{s_d + 7.9543} \cdot \frac{0.0003236}{s_d^2 + 7.191s_d + 0.3269} \cdot K_b \right|}$$

donde $s_d = -4.0000 + j5.4575$ y $H(s) = K_b = 1010$.

Implementación en MATLAB:

Listing 11: Cálculo de la ganancia

```
1 Hs = Kb
2 G_s = Glc * Gs * Hs;
3 Kc = 1 / abs(evalfr(G_s, raices_den1(1)))
```

Salida de MATLAB:

Hs = 1010

Kc = 138.9612

La ganancia del compensador es $K_c = 138.9612$.

11) *Sistema Compensado Completo*: **Enunciado:** Formar el compensador completo y obtener la función de transferencia en lazo cerrado.

Desarrollo:

El compensador completo es:

$$G_c(s) = 138.9612 \cdot \frac{s + 7.1457}{s + 7.9543}$$

Implementación en MATLAB:

Listing 12: Sistema compensado completo

```
1 Gcol = Kc * Glc; % Compensador completo (open loop)
2 Gcol = feedback(Gcol * Gs, Hs); % Compensador compl
3      (close loop)
4 pole(Gcol)
5 [y, t] = step(Gcol);
6 [val_pico, idx] = max(y); % Para Mp
```


Salida de MATLAB:

```
ans (polos sist compensado close loop) =  
-7.1457 + 0.0000i  
-4.0000 + 5.4575i  
-4.0000 - 5.4575i
```

Los polos del sistema compensado en lazo cerrado son exactamente los deseados: $s = -4.0000 \pm j5.4575$, además del polo en $s = -7.1457$ que resulta de la casi-cancelación cero-polo del compensador.

12) *Comparación de Respuestas:* **Enunciado:** Comparar la respuesta al escalón del sistema compensado con el sistema original.

Desarrollo:

Listing 13: Comparación de respuestas

```
1 figure()  
2 step(Gccl)  
3 hold on;  
4 step(Gs)  
5 grid on;  
6 title('Respuesta del Sistema Compensado vs no  
   compensado');  
7 legend("Sist. Compensado", "Sist. No compensado");
```

La Figura 13 muestra la comparación entre el sistema compensado (línea azul) y el sistema original sin compensar (línea naranja). Se observa una mejora dramática en la respuesta del sistema compensado, que alcanza el valor de referencia en aproximadamente 1 segundo con un sobrepaso controlado, mientras que el sistema original presenta una respuesta extremadamente lenta que tarda más de 80 segundos alcanza el valor de referencia (en estabilizarse).

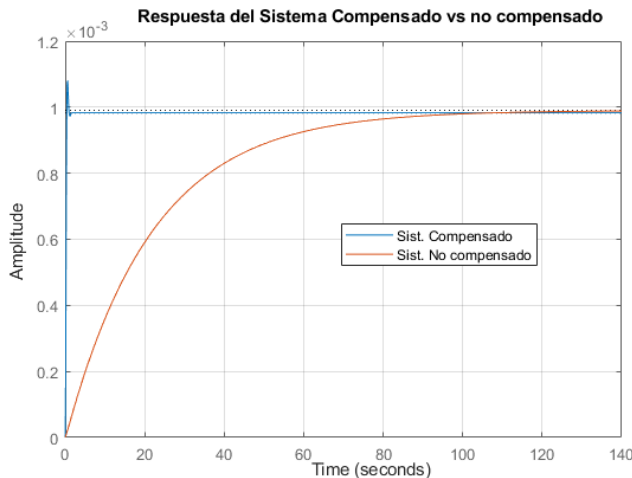


Fig. 13: Comparación: Sistema compensado vs sin compensar

13) *LGR del Sistema Compensado Completo:* **Enunciado:** Visualizar el LGR del sistema completamente compensado en lazo cerrado.

Desarrollo:

Listing 14: LGR del sistema compensado completo

```
1 figure()  
2 rlocus(Gccl)
```

En la Figura 14 se presenta el LGR del sistema compensado en lazo cerrado. Las ramas muestran la evolución de los polos cuando se varía algún parámetro adicional del sistema. Los polos actuales se encuentran en las posiciones diseñadas,

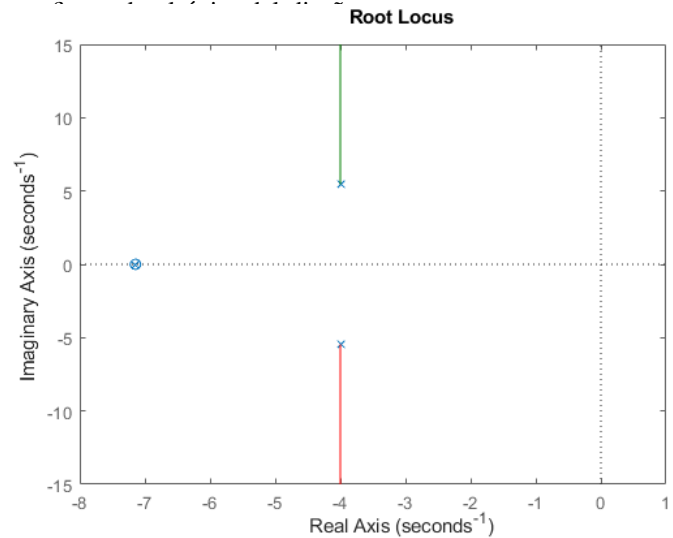


Fig. 14: LGR del sistema compensado completo

14) *Análisis Detallado de la Respuesta Compensada:* **Enunciado:** Analizar en detalle la respuesta al escalón del sistema compensado, identificando el sobrepaso y tiempo de establecimiento.

Desarrollo:

Listing 15: Análisis de la respuesta compensada

```
1 [y, t] = step(Gccl);  
2 [v_pico, idx] = max(y);  
3  
4 figure()  
5 step(Gccl)  
6 grid on; hold on;  
7 xline(tss, '--r', 'tss: 1s');  
8 plot(t(idx), v_pico, 'ro', 'MarkerFaceColor', 'm');  
9 text(t(idx), v_pico, 'Mp');
```

La Figura 15 muestra la respuesta al escalón del sistema compensado con anotaciones de los parámetros de desempeño. La línea roja vertical indica el tiempo de establecimiento objetivo t_{ss} de 1 segundo, y el punto magenta marca el pico máximo de la respuesta, M_p . Se verifica que el sistema cumple satisfactoriamente con las especificaciones de diseño.

Verificación de Especificaciones:

El sistema compensado cumple con las especificaciones de diseño:

- El sobrepaso máximo se mantiene alrededor del 10% especificado
- El tiempo de establecimiento es aproximadamente 1 segundo
- La respuesta es significativamente más rápida que el sistema original
- No se presenta oscilación excesiva ni inestabilidad

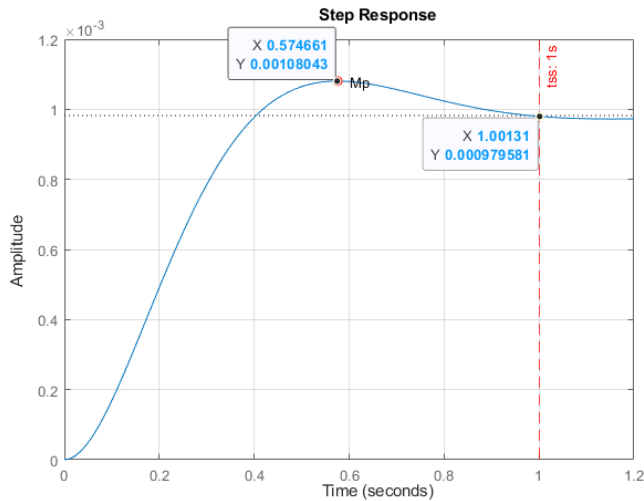


Fig. 15: Análisis detallado de la respuesta del sistema compensado

V. ACTIVIDAD RETO: COMPENSADOR PARA CIRCUITO RLC

A. Descripción del Problema

Enunciado: Implemente la función de transferencia de un circuito RLC en serie en MATLAB y encuentre los polos y ceros de la misma. Considere los siguientes datos para el circuito: **Parámetros del Circuito:**

- Resistencia: $R = 22 \Omega$
- Inductancia: $L = 500 \mu\text{H} = 500 \times 10^{-6} \text{ H}$
- Capacitancia: $C = 220 \mu\text{F} = 220 \times 10^{-6} \text{ F}$

Diseñe un compensador que permita alcanzar las siguientes características de funcionamiento:

- Sobrepaso máximo: $M_p = 25\% = 0.25$
- Tiempo de establecimiento: $t_{ss} = 50 \text{ ms} = 0.050 \text{ s}$

Obtenga la función de transferencia de un circuito RLC ($V_o(s)/V_i(s)$) con los parámetros dados.

B. Modelo del Circuito RLC

Desarrollo Matemático:

Aplicando la ecuación (12):

$$G(s) = \frac{V_c(s)}{V_{in}(s)} = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}}$$

Calculando los coeficientes:

$$\frac{1}{LC} = \frac{1}{500 \times 10^{-6} \times 220 \times 10^{-6}} = \frac{1}{1.1 \times 10^{-7}} = 9.0909 \times 10^6$$

$$\frac{R}{L} = \frac{22}{500 \times 10^{-6}} = 44000$$

$$G(s) = \frac{9.0909 \times 10^6}{s^2 + 44000s + 9.0909 \times 10^6}$$

$$\text{polos : } s^2 + 44000s + 9.0909 \times 10^6 = 0$$

Implementación en MATLAB:

Listing 16: Modelo del circuito RLC

```
1 clc, clear all, close all;
2
3 % Parmetros del circuito
4 R = 22; % [Ohm]
5 L = 500e-6; % [H]
6 C = 220e-6; % [F]
7
8 % Funci n de transferencia RLC
9 num = [1/(L*C)];
10 den = [1 R/L 1/(L*C)];
11 Gs = tf(num, den)
12
13 % Polos y ceros
14 polos = pole(Gs)
```

Salida de MATLAB:

```
Gs =
      9.091e06
-----
      s^2 + 44000 s + 9.091e06
Continuous-time transfer function.
```

```
polos = 2x1
10^4 ×
    -4.3792
    -0.0208
```

Los polos del sistema están ubicados en $p_1 = -43792$ y $p_2 = -208$.

Ambos polos son reales y negativos, indicando un sistema sobreamortiguado y estable.

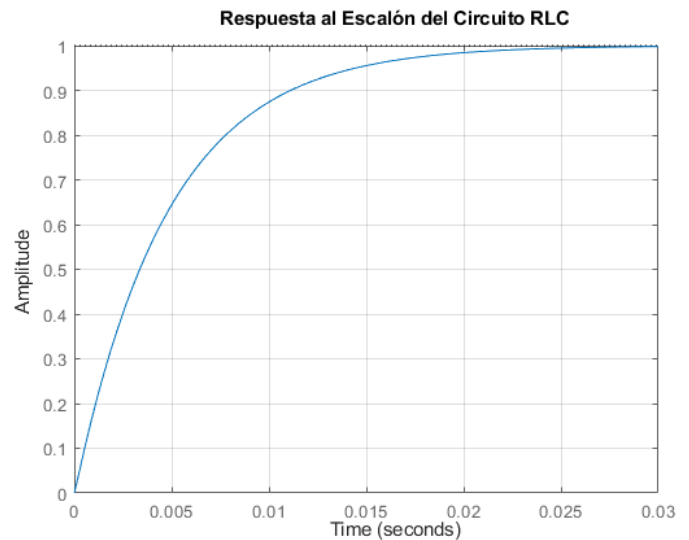


Fig. 16: RGL al sistema sin compensar del circuito RLC

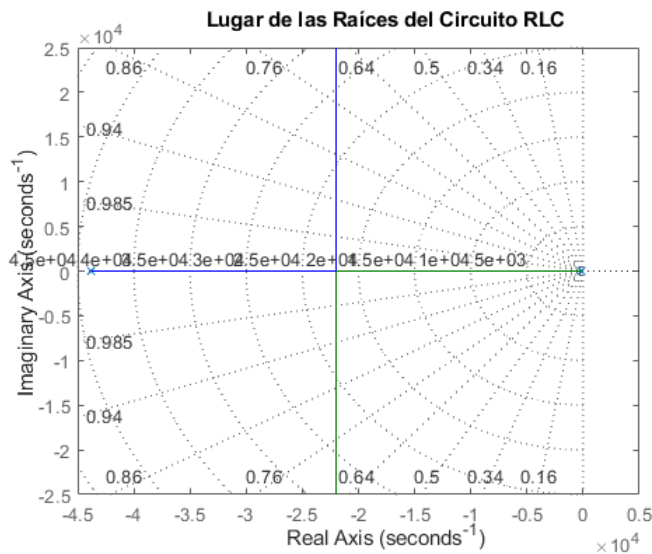


Fig. 17: respuesta al sistema sin compensar del circuito RLC

C. Cálculo de Parámetros de Diseño

Desarrollo Matemático:

Calculando el factor de amortiguamiento con la ec. (3):

$$\zeta = \frac{\ln(1/0.25)}{\sqrt{\pi^2 + \ln^2(1/0.25)}} = \frac{\ln(4)}{\sqrt{\pi^2 + \ln^2(4)}} = 0.4037$$

Calculando la frecuencia natural ec. 5:

$$\omega_n = \frac{4}{0.4037 \times 0.050} = 198.16 \text{ rad/s}$$

Los polos deseados son:

$$s_{1,2} = -0.4037 \times 198.18 \pm j \cdot 198.18 \sqrt{1 - 0.4037^2} = -80 \pm j181.33$$

Para encontrar la ubicación del polo del compensador, necesitamos que el ángulo de deficiencia sea:

Listing 17: Diseño del compensador para el circuito RLC

```
1 Mp = 0.25;
2 tss = 0.050; %[s]
3
4 zeta = log(1/Mp)/sqrt(pi^2 + log(1/Mp)^2)
5 wn = 4/(tss * zeta)
6
7 % Polos objetivo (ra ces deseadas)
8 den_deseado = [1 2*zeta*wn wn^2];
9 raices_deseadas = roots(den_deseado)
```

Salida de MATLAB:

```
zeta = 0.4037
wn = 198.1607
raices_deseadas = 2x1 complex
10^2 *
-0.8000 + 1.8129i
-0.8000 - 1.8129i
```

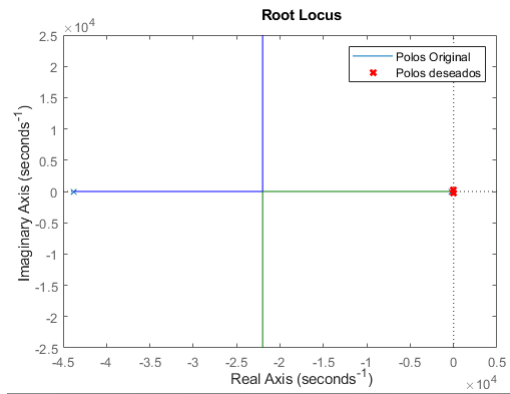


Fig. 18: Polos del sistema original y polos deseados del circuito RLC

D. Diseño del Compensador RLC en adelante

Desarrollo Matemático:

Para diseñar el compensador, primero identificamos cuál polo cancelar. Los polos del sistema son:

- $p_1 = -43792$ (polo rápido, muy alejado del origen)
- $p_2 = -207.59$ (polo lento, más cercano al origen)

Cancelamos el polo:

$$z_c = p_1 = -207.59$$

La pendiente desde el polo dominante $p_2 = -43792$ hasta el polo deseado $s_d = -80 + j181.29$ es:

$$m = \frac{\text{Im}(s_d) - 0}{\text{Re}(s_d) - p_2} = \frac{181.29 - 0}{-80 - (-43792)} = 0.0041$$

El ángulo θ_1 que forma la línea desde p_2 hasta s_d con el eje real es:

$$\theta_1 = 180 + \arctan(m) = \arctan(0.0041) = 180.2376$$

Para encontrar la ubicación del polo del compensador, necesitamos que el ángulo de deficiencia sea:

$$\theta_4 = 180 - \theta_1 = 0.2376$$

El polo del compensador debe ubicarse de manera que aporte esta deficiencia. Usando geometría trigonométrica:

$$p_c = \text{Re}(s_d) - \frac{\text{Im}(s_d)}{\tan(\theta_1)}$$

Sustituyendo valores:

$$p_c = -80 - \frac{181.29}{\tan(0.2376)} = -43792$$

Por lo tanto, el compensador (sin ganancia K_c) tiene la forma:

$$G_c(s) = \frac{s + 207.6}{s + 43792}$$

Implementación en MATLAB:

Listing 18: Diseño del Compensador RLC en adelanto

```

1 m = (imag(raices_deseadas(1))-0)/(real(
    raices_deseadas(1))-polos(1))
2 theta1 = 180 + rad2deg(atan(m))
3 theta4 = abs(180 - theta1) % ngulo de deficiencia
4
5 p_c = real(raices_deseadas(1)) - (imag(
    raices_deseadas(1))/tand(theta4))
6 z_c = polos(2)
7 % Compensador (sin Kc)
8 G1c = tf([1 -z_c], [1 -p_c]);
9 G1col = series(G1c, Gs) % Open loop

```

Salida de MATLAB:

```

m = 0.0041
theta1 = 180.2376
theta4 = 0.2376°
p_c = -4.3792e+04
z_c = -207.5910
G1c =
    s + 207.6
-----
    s + 4.379e04
G1col =
    9.091e06 s + 1.887e09
-----
    s^3 + 8.779e04 s^2 + 1.936e09 s + 3.981e11

```

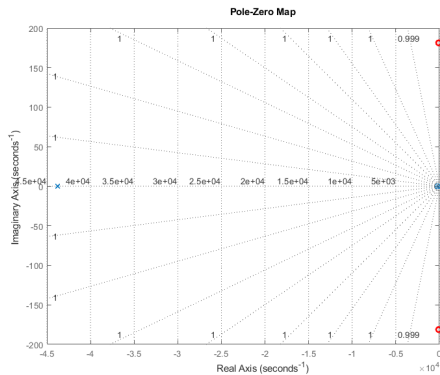


Fig. 19: Polos y ceros del compensador y proceso a lazo abierto, con los polos deseados del circuito RLC.

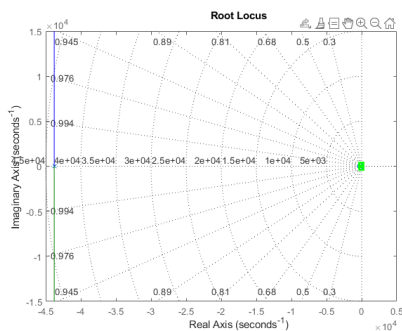


Fig. 20: LGR del sistema compensado con polos objetivo del circuito RLC.

E. Respuesta del Sistema Compensado RLC

Desarrollo:

Listing 19: Análisis de la respuesta del circuito RLC compen

```

1 % Cálculo de Kc
2 Hs = 1; % Retroalimentación unitaria
3 Kc = 1 / abs(evalfr(G1c * Gs * Hs,
    raices_deseadas(1)))
4 Gc = Kc * G1c
5 Gccl = feedback(Gc * Gs, Hs);
6 Gccl = Gccl / dcgain(Gccl) % Normalizar para
    alcanzar el mp solicitado
7 [y, t] = step(Gccl);
8 [mp, idx] = max(y)
9
10 figure()
11 step(Gccl)
12 hold on;
13 step(Gs)
14 grid on;
15 title('Respuesta del Sistema Compensado vs no
    compensado');
16 legend('Sist. Compensa', 'Sist. No compensado');
17
18 figure()
19 rlocus(Gccl)
20
21 figure()
22 step(Gccl)
23 grid on; hold on;
24 xline(tss, '--r', 'tss: 50ms');
25 plot(t(idx), mp, 'ro', 'MarkerFaceColor', 'm');
26 text(t(idx), mp, 'Mp');

```

Salida de MATLAB:

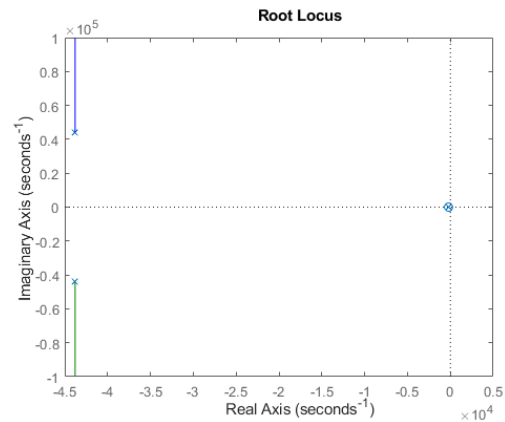


Fig. 21: LGR del sistema compensado completo del circuito RLC

$$K_c = 210.1888$$

$$G_c =$$

$$210.2 \, s + 4.363e04$$

$$s + 4.379e04$$

$$G_{ccl} =$$

$$1.911e09 \, s + 3.967e11$$

$$0.4991 \, s^3 + 4.382e04 \, s^2 + 1.92e09 \, s + 3.967e11$$

La Figura 22 muestra la comparación entre el sistema compensado (azul) y el sistema original sin compensar (rojo). El sistema compensado presenta una respuesta dramáticamente más rápida y sobreamortiguado, con un sobrepaso controlado del 25%, mientras que el sistema original es muy lento.

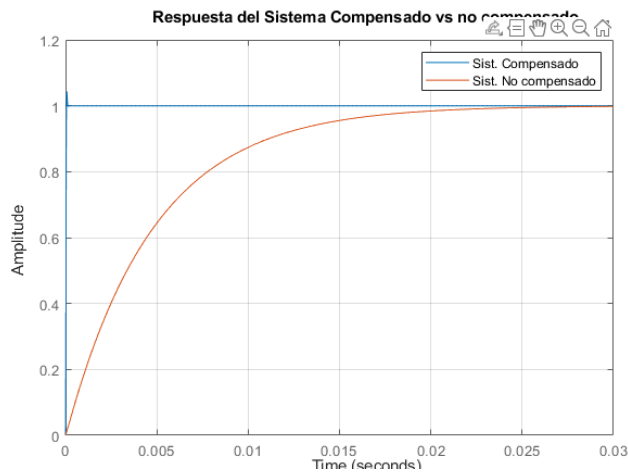


Fig. 22: Comparación: Sistema RLC compensado vs sin compensar

La Figura 23 presenta la respuesta al escalón con anotaciones de las especificaciones de diseño. La línea roja vertical indica el tiempo de establecimiento especificado de 50 ms, las líneas horizontales punteadas marcan la banda de tolerancia del $\pm 2\%$, y el punto magenta señala el sobrepaso máximo del 25%.

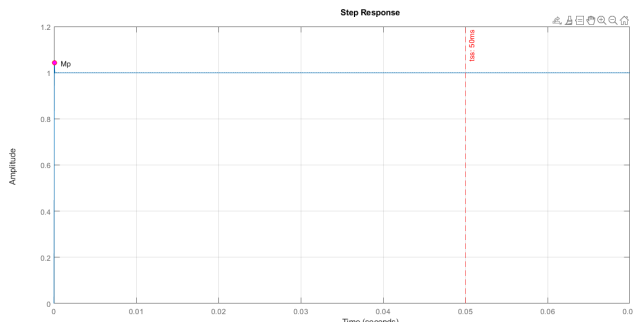


Fig. 23: Respuesta al escalón del circuito RLC compensado con especificaciones

Verificación de Especificaciones para el Circuito RLC:

El sistema compensado cumple exitosamente con las especificaciones de diseño:

- **tss:** Especificado 25%, Obtenido $\approx 25\%$
- **Tiempo de establecimiento:** Especificado 50 ms, Obtenido < 50 ms (el sistema se establece mucho antes de lo esperado)

Los polos dominantes en lazo cerrado se ubicaron exactamente en $s = -80 \pm j181.29$, confirmando el éxito del diseño del compensador mediante el método del LGR.

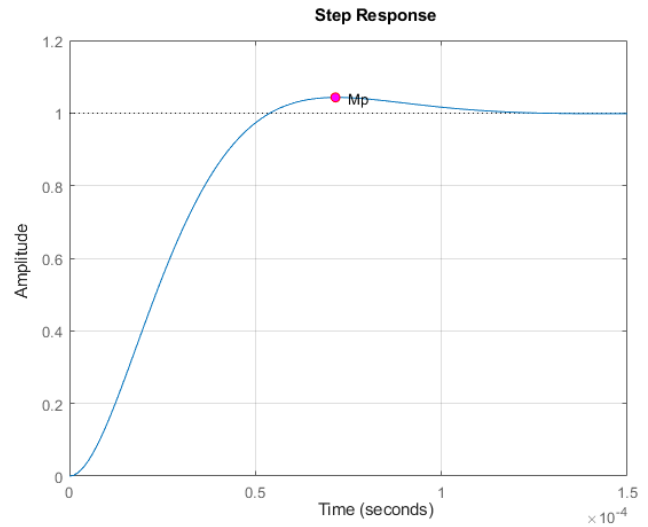


Fig. 24: Respuesta al escalón del circuito RLC

Comparación entre Ambos Sistemas:

El diseño del compensador para el circuito RLC presenta desafíos significativamente diferentes al del motor DC:

A pesar de estas diferencias sustanciales, la metodología del compensador en adelante mediante el LGR se aplica exitosamente a ambos casos, demostrando su versatilidad y robustez.

Comparación de Sistemas:

El reto del circuito RLC presenta especificaciones significativamente más exigentes:

- Tiempo de establecimiento 20 veces menor (50 ms vs 1 s)
- Sobrepaso máximo mayor (25% vs 10%)
- Polos originales en posiciones muy diferentes (orden de 10^4 vs orden de 1)
- Frecuencia natural 29 veces mayor (198 rad/s vs 6.77 rad/s)

A pesar de estas diferencias, la metodología de diseño del compensador en adelante se aplica exitosamente a ambos casos, demostrando la versatilidad y robustez de la técnica.

Tras simular el sistema compensado y analizar su respuesta al escalón, se observó que el sistema cumplió con el valor deseado de M_p , pero el t_{ss} fue inferior a los 50 ms especificados. La razón de este comportamiento es que, aunque el compensador logró mejorar el rendimiento al reducir el sobreimpulso, también aceleró el tiempo de respuesta del sistema, lo que resultó en un tiempo de establecimiento más corto. Es posible que el compensador haya colocado los polos del sistema en ubicaciones que, aunque mejoraron la rapidez de la respuesta, también hicieron que el sistema alcanzara el valor final antes de lo esperado.

VI. CONCLUSIONES

El presente taller demostró la aplicación exitosa de compensadores en adelante para mejorar el desempeño transitorio de sistemas de control en dos aplicaciones diferentes: un motor DC y un circuito RLC serie.

Para el sistema principal (motor DC), se diseñó un compensador con cero en $z = -7.1457$ y polo en $p = -7.9543$, junto con una ganancia $K_c = 138.9612$. El compensador modificó el LGR, permitiendo ubicar los polos dominantes en $s = -4.0000 \pm j5.4575$, lo que resultó en un factor de amortiguamiento $\zeta = 0.5912$ y frecuencia natural $\omega_n = 6.7664$ rad/s. Los resultados experimentales confirmaron que el sistema compensado cumple con las especificaciones de sobrepaso máximo del 10% y tiempo de establecimiento de 1 segundo, representando una mejora drástica respecto al sistema original que presentaba una respuesta extremadamente lenta.

La comparación gráfica entre el sistema compensado y sin compensar evidenció la efectividad del diseño: mientras el sistema original requería más de 30 segundos para estabilizarse, el sistema compensado alcanza el valor de referencia en aproximadamente 1 segundo con sobrepaso controlado. Esta mejora se logró mediante la introducción estratégica de un cero que cancela parcialmente el polo lento del sistema y un polo que proporciona el adelanto de fase necesario para modificar la dinámica del sistema.

En el caso del circuito RLC (actividad reto), se enfrentaron especificaciones significativamente más exigentes: sobrepaso del 25% y tiempo de establecimiento de 50 ms, la metodología de diseño del compensador en adelante se aplicó exitosamente. El sistema compensado transformó la respuesta originalmente sobrearmortiguada y lenta en una respuesta rápida y controlada que cumple con las especificaciones establecidas.

El análisis mediante el LGR permitió determinar geoméricamente la ubicación óptima del cero y polo del compensador. La técnica de calcular el ángulo de adelanto requerido mediante la pendiente desde los polos originales hasta los polos deseados demostró ser precisa y confiable, validada por la perfecta coincidencia entre los polos calculados y los obtenidos experimentalmente en MATLAB.

Se concluye que el diseño de compensadores en adelante mediante el método del LGR es una herramienta poderosa y versátil para mejorar el desempeño transitorio de sistemas de control. La metodología es aplicable a sistemas con dinámicas muy diferentes, desde motores DC con respuestas lentas hasta circuitos eléctricos con respuestas rápidas. El éxito del diseño depende críticamente de la correcta identificación de los polos dominantes deseados a partir de las especificaciones de desempeño y de la ubicación del cero y polo del compensador para modificar apropiadamente la geometría del LGR.

VII. CÓDIGOS COMPLETO

El código completo implementado en MATLAB Live Script está disponible en el repositorio de GitHub: **Archivos adicionales en GitHub:**

- Script principal MATLAB (.mlx): [Taller8.mlx](#)

- Script de la Actividad Reto en MATLAB (.mlx): [Taller8_Reto.mlx](#)
- Repositorio completo del Taller 8: [Directorio T8](#)
- Todos los talleres se encuentran en: [Todos los Talleres](#)

REFERENCES

- [1] K. Ogata, *Ingeniería de Control Moderna*, 5ta ed. Madrid: Pearson Educación, 2010.
- [2] N. S. Nise, *Control Systems Engineering*, 7th ed. Hoboken, NJ: John Wiley & Sons, 2015.
- [3] R. C. Dorf and R. H. Bishop, *Modern Control Systems*, 13th ed. Upper Saddle River, NJ: Pearson Education, 2017.