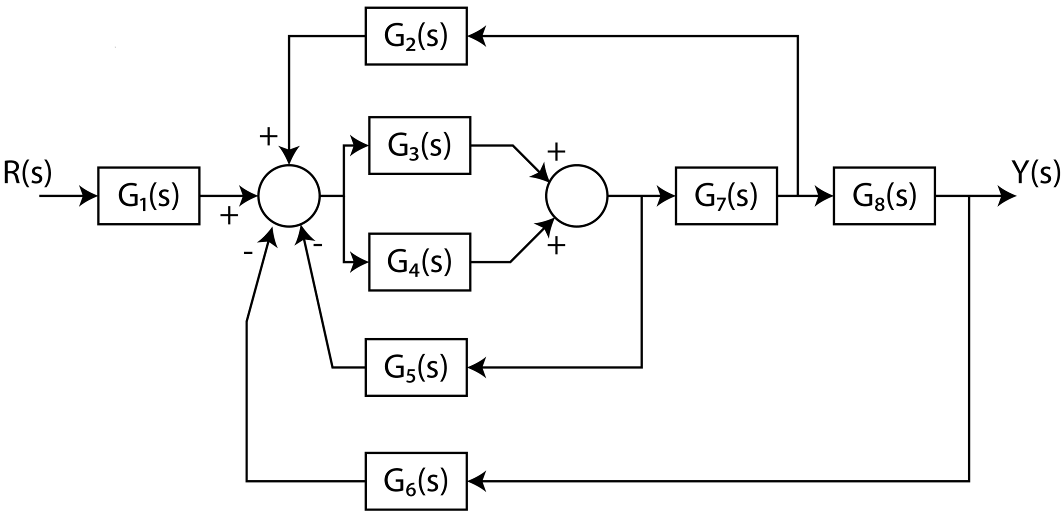
Práctica 4 - Ejercicio 3

Dado el sistema representado por el diagrama de bloques de la figura:



- ***Reduce el diagrama de bloques a la expresión G)s)=Y(s)/X(s) (0.75 ptos)***

Defino las funciones de transferencia que intervienen en el sistema:

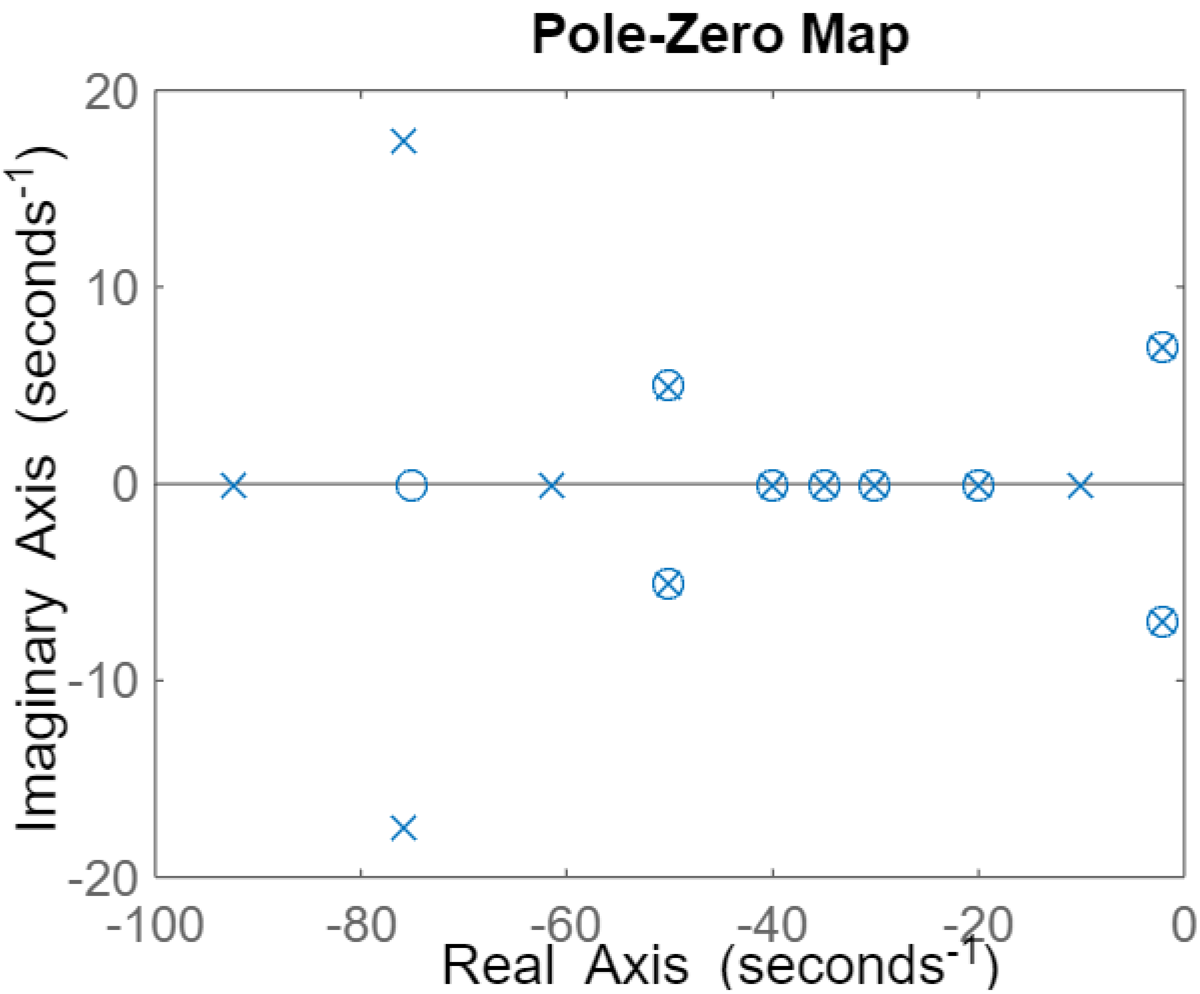
```
G1=tf(200*[1 70 1200], [1 45 350]);
G2=tf(1, [1 24 133 1060]);
G3=tf(1, [1 30]);
G4=tf(1, [1 40]);
G5=tf(1, [1 100 2525]);
G6=tf([1 30], [1 75]);
G7=tf([700], [1 70]);
G8=tf([1 30], [1 160 6500]);

%Sistema parelelo G3 y G4
H1=parallel(G3, G4);

%Realimentacion negativa del anterior con G5
H2=feedback(H1, G5);

%Serie del anterior con G7
H3=series(H2, G7);

%Feedback positivo del anterior con G2
H4=feedback(H3, G2, +1);

%Series del anterior con G8
H5=series(H4, G8);

%Feedback negativo del anterior con G6
H6=feedback(H5, G6, -1);

%Series de G1 con el anterior
```

```
G_final=series(G1, H6)
```

```
G_final =

  280000 s^10 + 9.352e07 s^9 + 1.345e10 s^8 + 1.091e12 s^7 + 5.5e13 s^6

          + 1.784e15 s^5 + 3.729e16 s^4 + 4.933e17 s^3 + 4.041e18 s^2

                                                        + 2.115e19 s + 7.082e19

  -----------------------------------------------------------------------------

  s^13 + 544 s^12 + 131863 s^11 + 1.878e07 s^10 + 1.745e09 s^9 + 1.11e11 s^8

          + 4.945e12 s^7 + 1.548e14 s^6 + 3.381e15 s^5 + 5.062e16 s^4

                        + 5.097e17 s^3 + 3.435e18 s^2 + 1.552e19 s + 3.838e19

Continuous-time transfer function.
```

- ***¿Es posible aproximar el sistema por otro más simple de orden reducido? Representa y compara la respuesta ante una entrada escalón unitario para ambos sistemas (0.75 pto)***

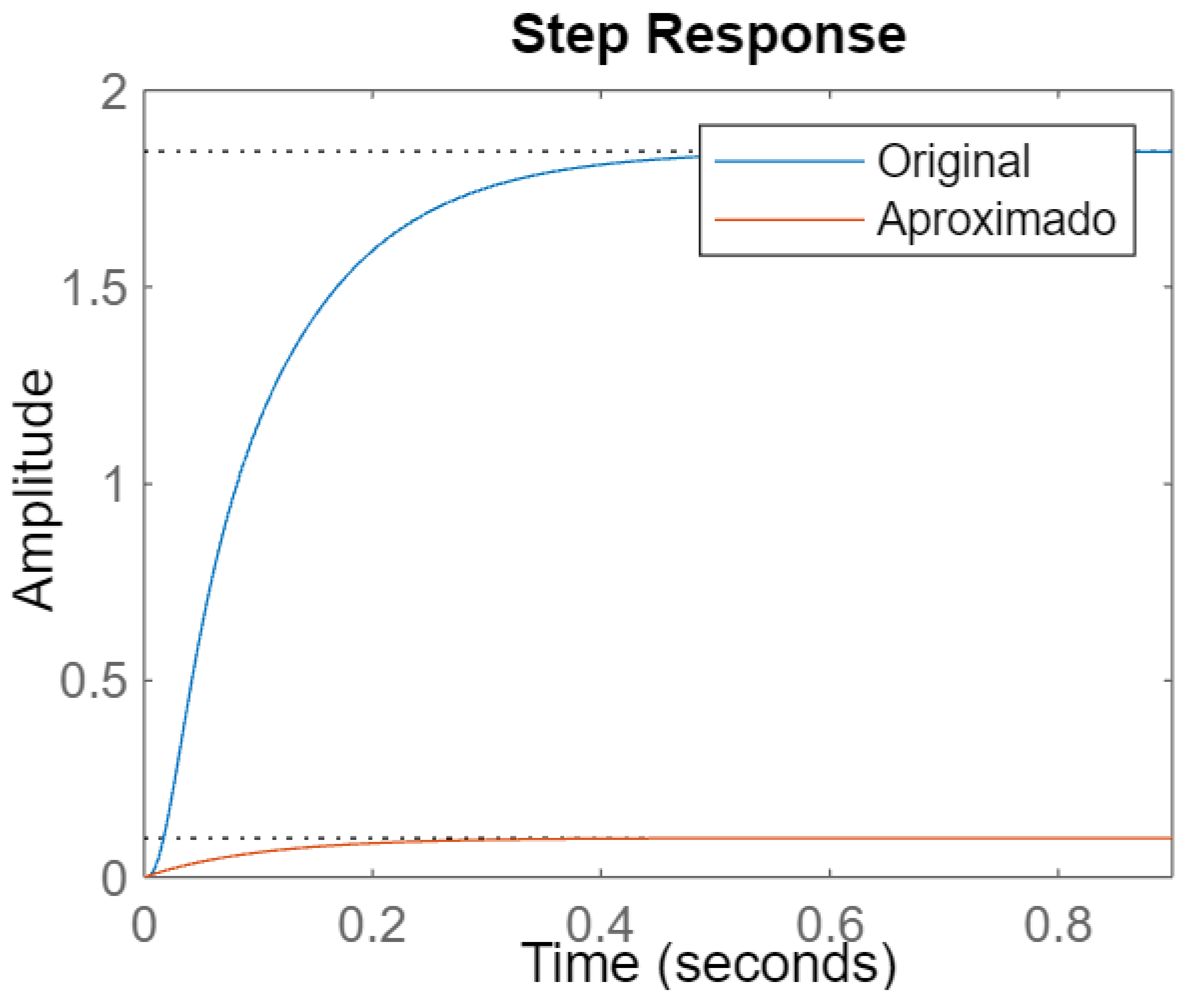
```
figure();
pzmap(G_final); %Representación del mapa de polos y ceros
```



De todos los polos del sistema, el único dominante es el que se encuentra en s=-10. El resto de polos con parte real mayor que -60 está compensado por los ceros correspondientes. Los polos y ceros que no están compensados tienen parte real que es al menos 6 veces la parte real del polo dominante, por ello su dinámica también es despreciable.

En conclusión, la dinámica del sistema se puede aproximar por el sistema G=1/(s+10)

```
G_aprox=zpk([], -10, 1);
figure();
step(G_final, G_aprox)
legend('Original', 'Aproximado')
```

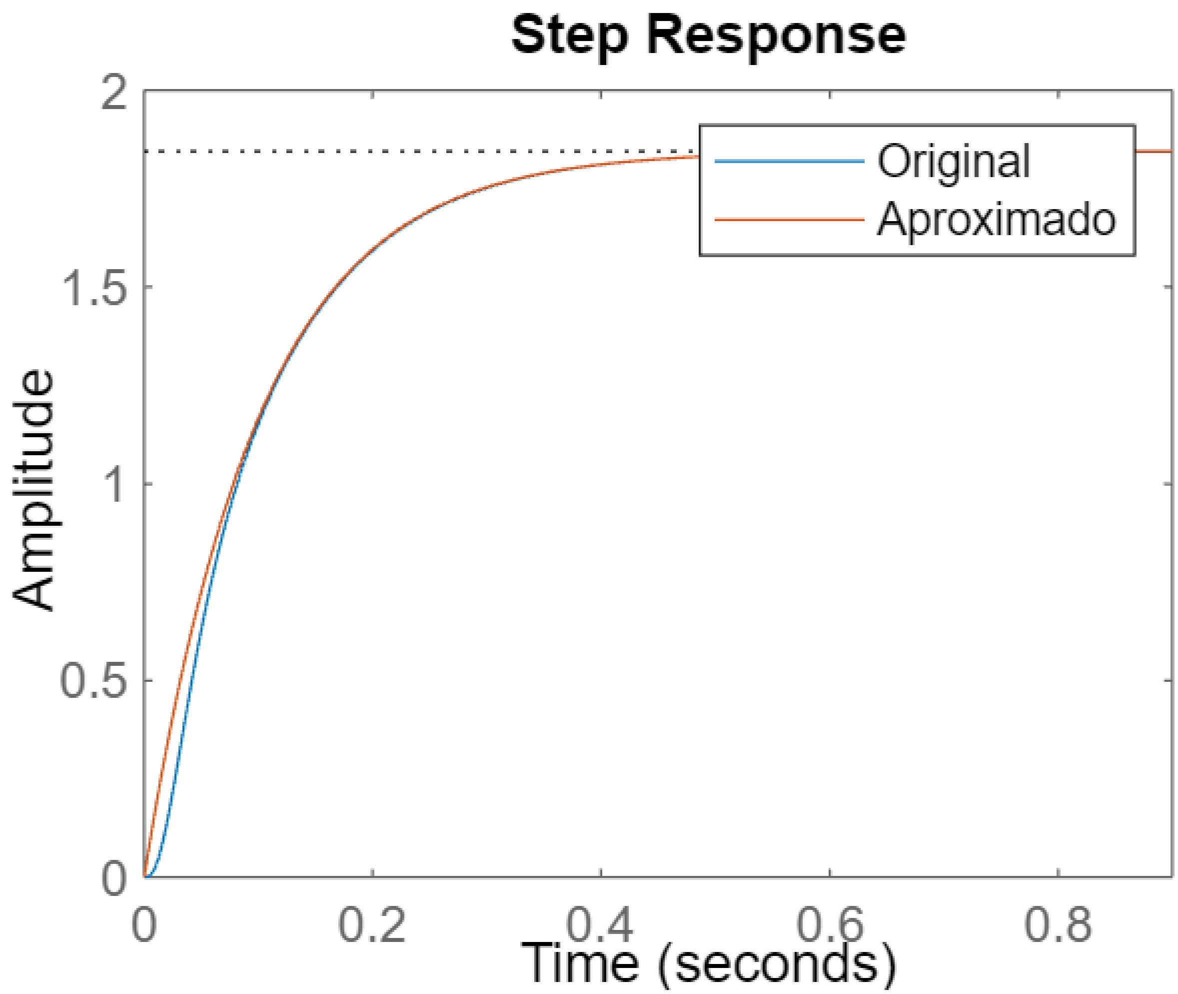


Para que el sistema tenga el mismo valor final, hay que ajustar también la ganancia estática del sistema aproximado. Para ello, calculo en primer lugar la ganancia estática del sistema original y del sistema aproximado. Aplicando el teorema del valor final, esta ganancia estática se obtiene cuando se evalúa la función de transferencia en s=0.

```
kss_G=abs(evalfr(G_final,0));
kss_Gaprox=abs(evalfr(G_aprox, 0));
```

Conocidad estas ganancias, se corrige la función de transferencia del sistema aproximado para forzarle a tener la misma ganancia estática del sistema original. Esto se consigue dividiendo la función de transferencia aproximada por su ganancia estática (así se consigue una ganancia unitaria) y multiplicandola a continuación por la ganancia estática del sistema final.

```
G_aprox=G_aprox*kss_G/kss_Gaprox;
figure();
step(G_final, G_aprox)
legend('Original', 'Aproximado')
```



Como se puede ver en la figura, ambos sistemas se comportan practicamente igual. Sin embargo, hay diferencia al comienzo de la respuesta transitoria debido a los polos y ceros rápidos no dominantes del sistema.

- ***Diseñe los controladores proporcional necesarios para que el sistema de orden reducido tal que su tiempo de asentamiento sea de 0.05s y de 0.2s. Aplique los controladores diseñados sobre la planta original y discuta los resultados. (1.5 pto)***

Para aplicar las condiciones de diseño, es preciso determinar la posición en la que habrá que colocar los polos en lazo cerrado para obtener los requisitos temporales pedidos. Para ello, al tratarse de sistemas de primer orden, se puede utilizar la expresión: $t_s = \frac{\pi}{\sigma}$

```
sigma_02=-pi/0.2
```

```
sigma_02 = -15.7080
```

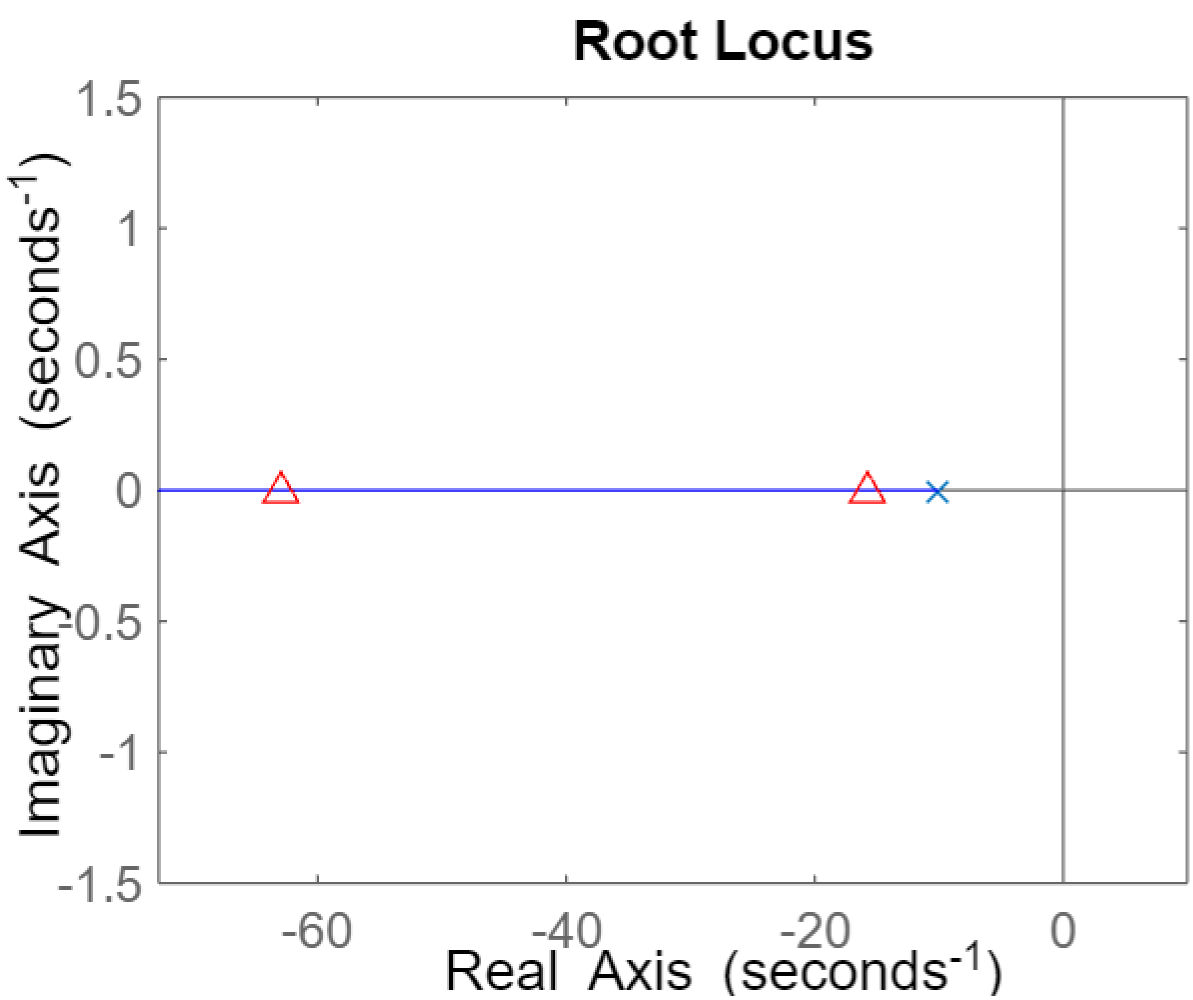
```
sigma_005=-pi/0.05
```

```
sigma_005 = -62.8319
```

La posición de los polos en lazo cerrado, por tanto, será: s=-15.7 (para obtener un tiempo de asentamiento de 0.2s) y de s=-62.83 (para obtener un tiempo de asentamiento de 0.05s).

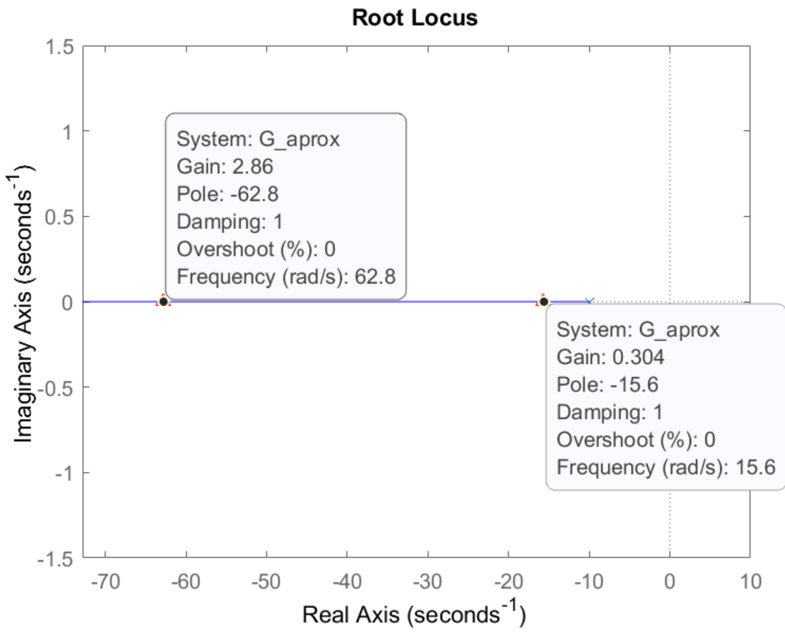
Para determinar los controladores proporcionales necesarios, se utilizará el lugar de las raices del sistema reducido. Puesto que no se indica lo contrario, se utilizará realimentación unitaria para conseguir el comportamiento requerido.

```
rlocus(G_aprox)
hold on
plot([sigma_02, sigma_005], [0 0], '^r');
hold off
xlim([sigma_005-10, 10])
```




```
%Dibuja la posición de los polos en lazo cerrado necesaria
```

Analizando el lugar de las raíces, se pueden obtener las ganancias del controlador proporcional necesarias para conseguir el comportamiento deseado

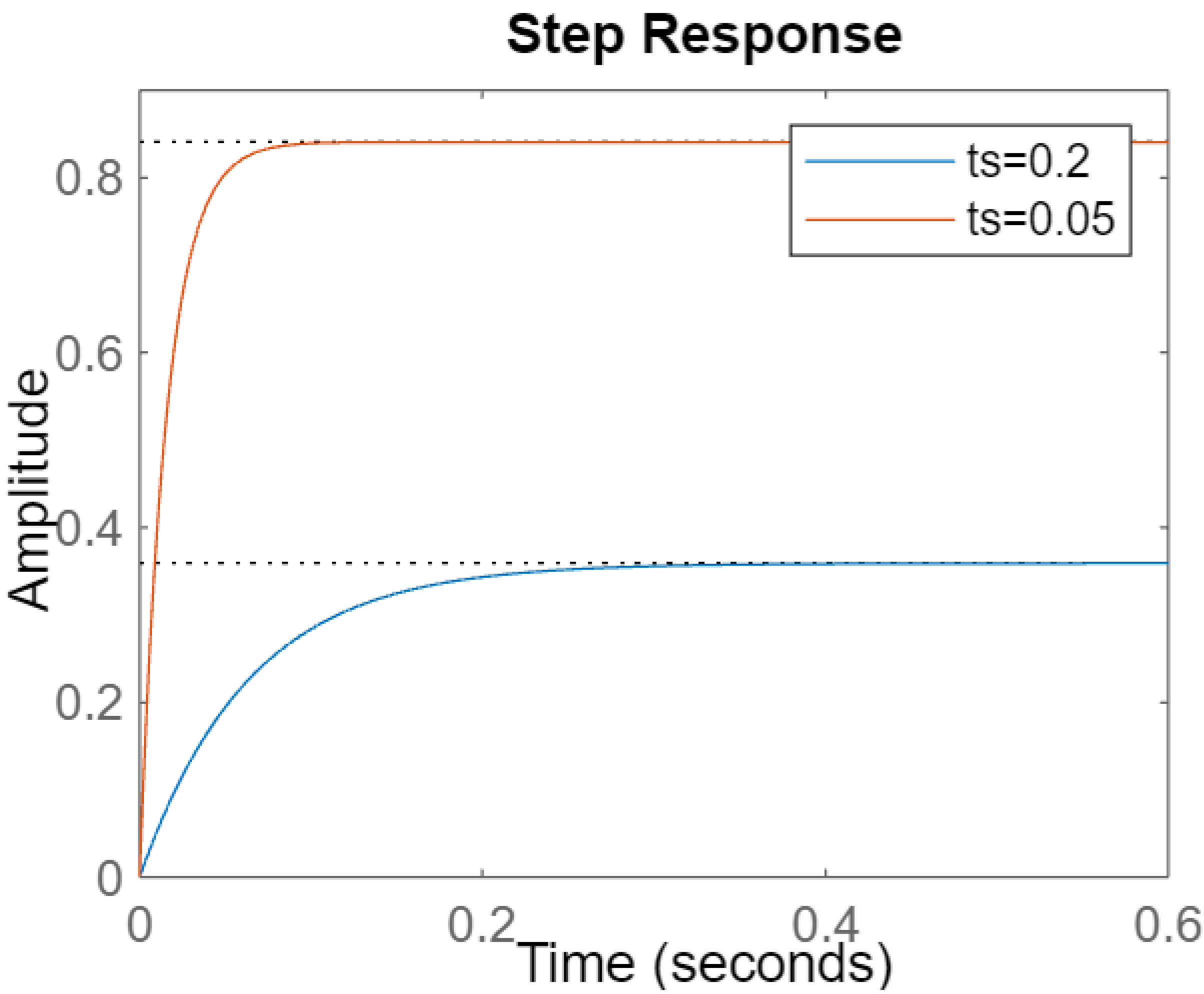


Por tanto, los controladores proporcionales necesarios son:

- P=0.304 para un tiempo de asentamiento de 0.2s
- P=2.86 para un tiempo de asentamiento de 0.05s

Lo comprobamos con la siguiente figura

```
P_02=0.304;
P_005=2.86;
G_aprox_P02=feedback(P_02*G_aprox, 1);
G_aprox_P005=feedback(P_005*G_aprox, 1);
step(G_aprox_P02, G_aprox_P005)
legend('ts=0.2', 'ts=0.05')
```




```
ts02_aprox=stepinfo(G_aprox_P02).SettlingTime
```

```
ts02_aprox = 0.2506
```

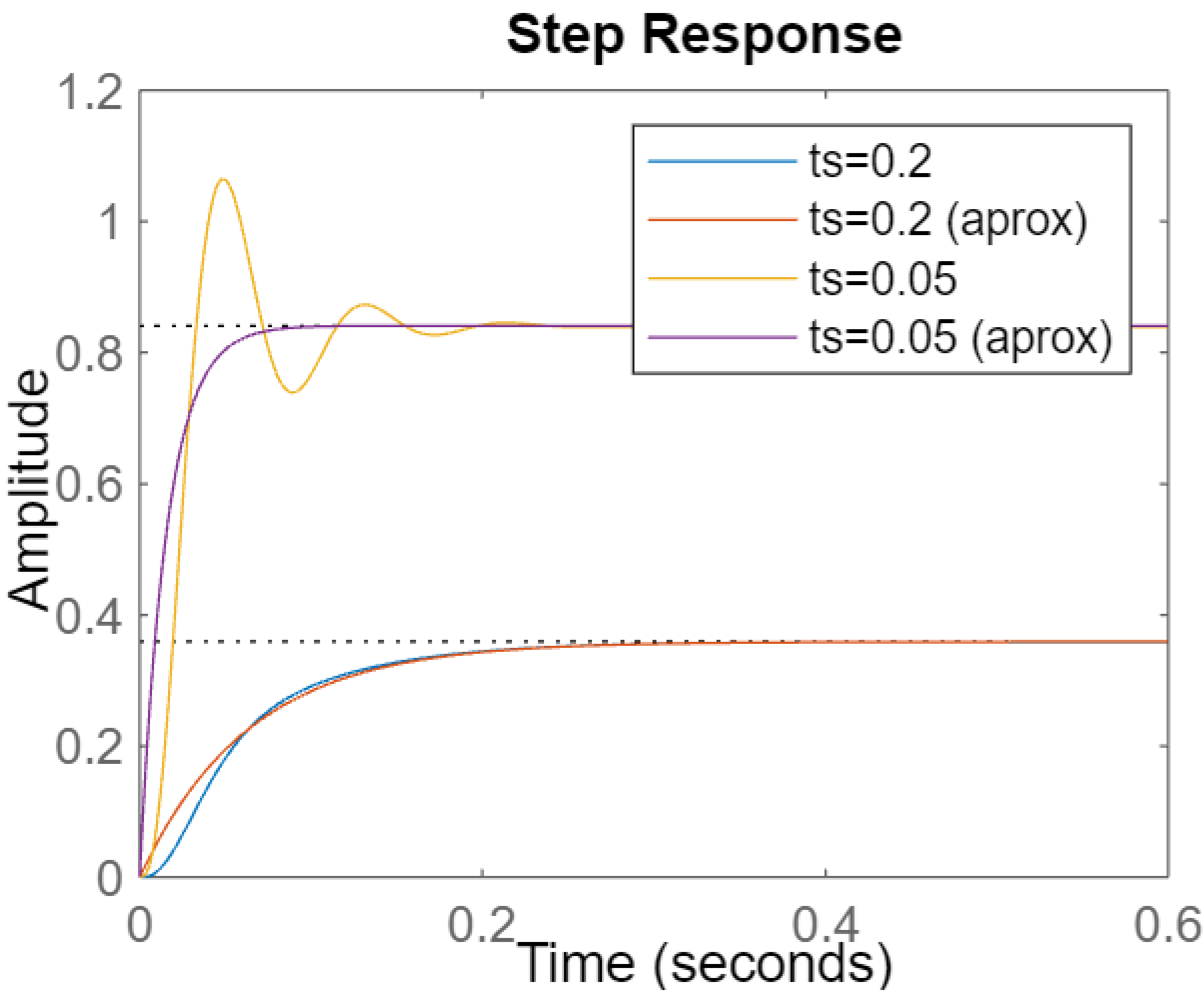
```
ts005_aprox=stepinfo(G_aprox_P005).SettlingTime
```

```
ts005_aprox = 0.0623
```

Los resultados obtenidos coinciden razonablemente con las especificaciones requeridas.

A continuación, se van a aplicar los mismos controladores sobre el sistema sin reducir:

```
G_P02=feedback(P_02*G_final, 1);
G_P005=feedback(P_005*G_final, 1);
step(G_P02, G_aprox_P02, G_P005, G_aprox_P005)
legend('ts=0.2', 'ts=0.2 (aprox)', 'ts=0.05', 'ts=0.05 (aprox)')
```




```
ts02=stepinfo(G_P02).SettlingTime
```

```
ts02 = 0.2473
```

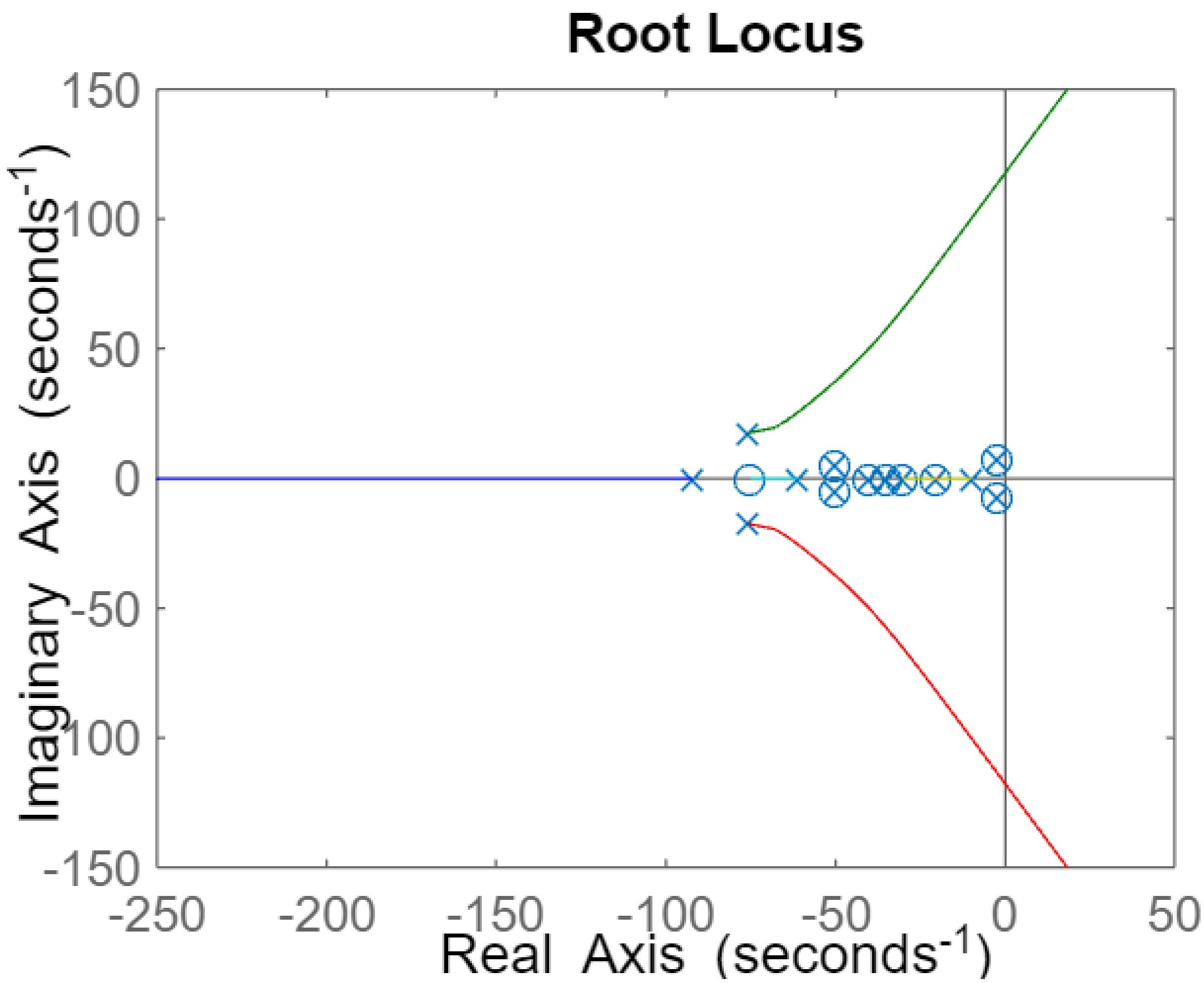
```
ts005=stepinfo(G_P005).SettlingTime
```

```
ts005 = 0.1452
```

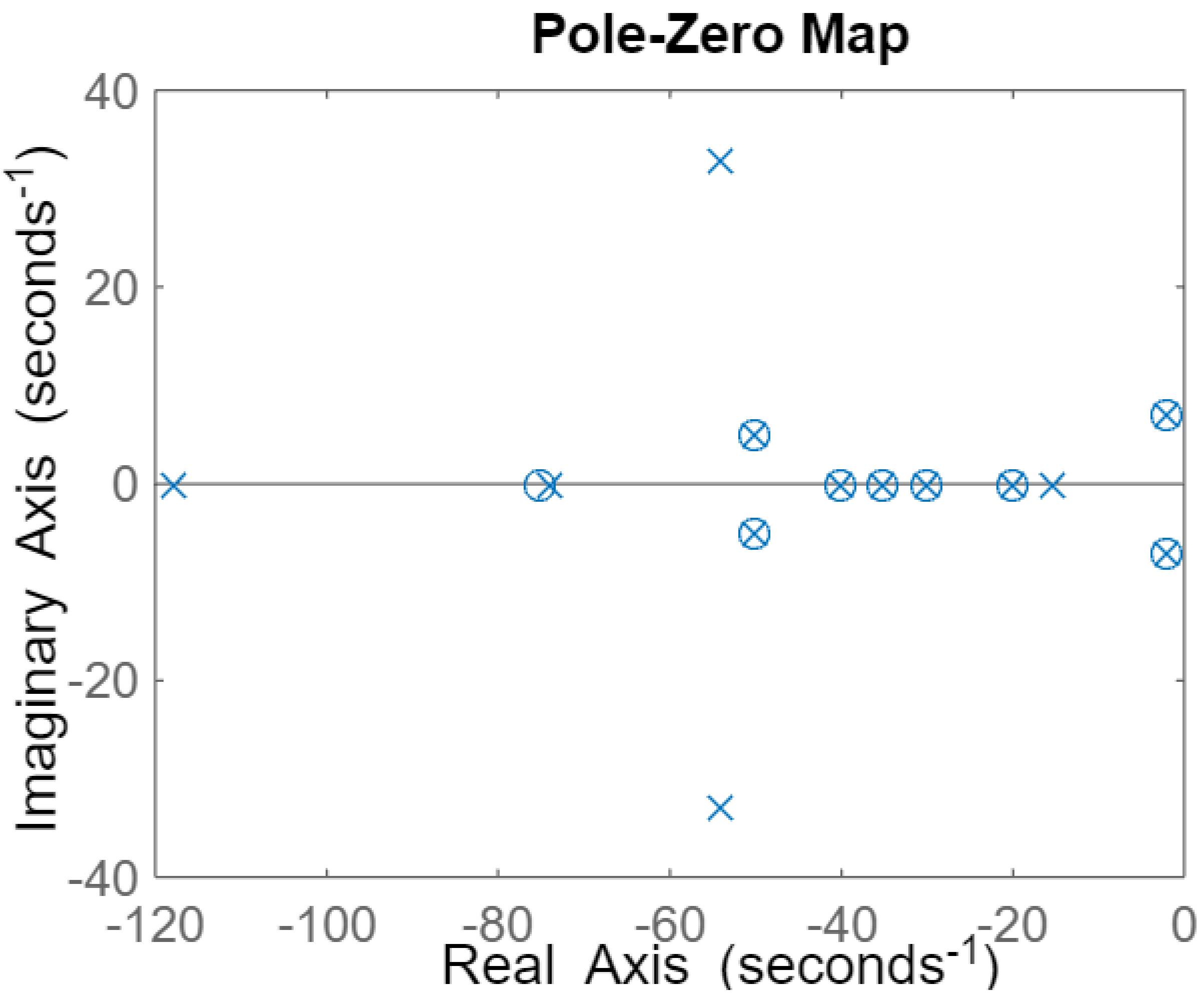
En este caso, se observa como el sistema real se comporta de igual manera que el aproximado en el caso del tiempo de asentameinto de 0.05s. Sin embargo, en el caso del tiempo de asentamiento de 0.2s, vemos como la dinámica de ambos sistemas es diferente, pasándo el sistema original a comportarse como un sistema subamortiguado.

Vamos a analizar el lugar de las raices para comprobar el motivo de estas discrepancias.

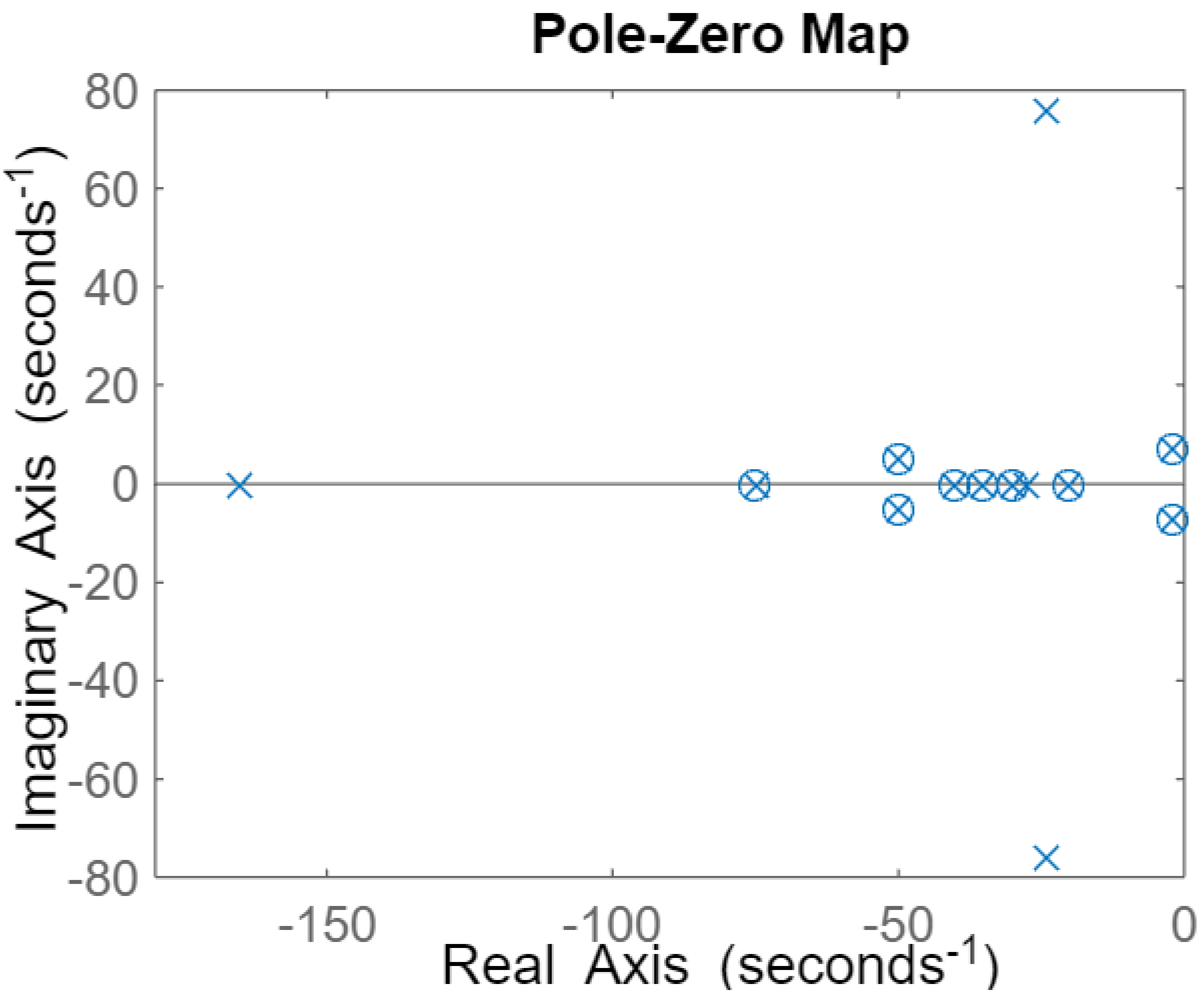
```
rlocus(G_final)
```


Root Locus
150
100
50
0
-50
-100
-150
Imaginary Axis (seconds^{-1})
-250
-200
-150
-100
-50
0
50
Real Axis (seconds^{-1})

```
pzmap(G_P02)
```


Pole-Zero Map
40
20
0
-20
-40
-120
-100
-80
-60
-40
-20
0
Real Axis (seconds-1)
Imaginary Axis (seconds-1)

```
pzmap(G_P005)
```



A la vista del lugar de las raíces del sistema original y de los mapas de polos y ceros del sistema con ambos controladores se encuentra el origen de esta discrepancia.

Este sistema tiene en el lugar de las raíces un par de asíntotas que nacen en los polos complejos conjugados y siguen asíntotas hacia el infinito en +/-60º.

Manteniendo ganancias bajas para el controlador P (P_02), los polos en lazo cerrado que se encuentran en estas ramas continuan siendo despreciables frente al polo en lazo cerrado del sistema. Sin embargo, al aumentar la ganancia del controlador (P_005), los polos en lazo cerrado se desplazan por estas ramas, llegando a una posición en la cual se vuelven dominantes frente al polo real. Esto hace que aparezcan las oscilaciones en el transitorio que difieren del comportamiento del sistema aproximado.

En conclusión, la aproximación que se ha hecho, sólo permanece válida mientras se tengan ganancias bajas, puesto que los polos en lazo cerrado permanecerán cercanos a los polos en lazo abierto. Cuando se aumenta la ganancia, estos polos en lazo cerrado se desplazarán por el lugar de las raíces hacia los ceros en lazo abierto o hacia las asíntotas, haciendo que pueda no seguir siendo aplicable la aproximación que se hizo en el lazo abierto.