ECE 414 - Josh Andrews - PID Search HW

Table of Contents

Testing the PI controller with PIDSEARCH function

Find good PI controller starting point using rlocus

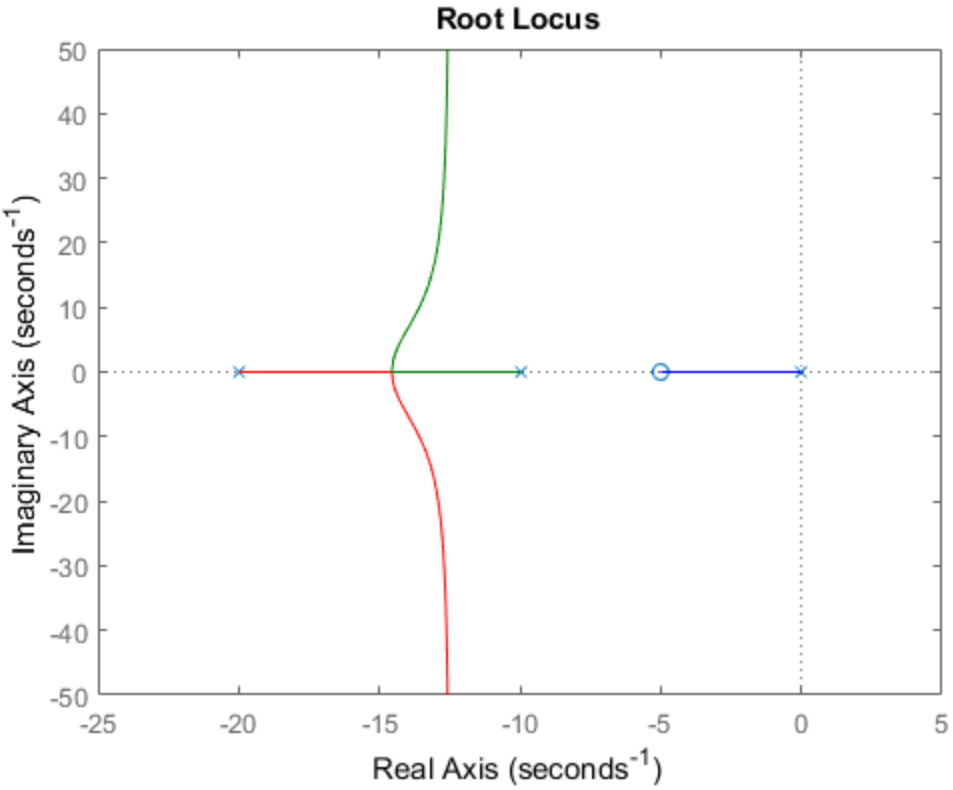
```
Kpi = 2;
Zpi = 5;
G = tf(40, [1 30 200]);
C = tf([Kpi Zpi*Kpi], [1 0])
H_locus = G*C;

figure(1); clf;
rlocus(H_locus)
```

```
C =

  2 s + 10
  --------
     s

Continuous-time transfer function.
```



Use PID Search for minimizing ITAE

Start with the root locus found controller

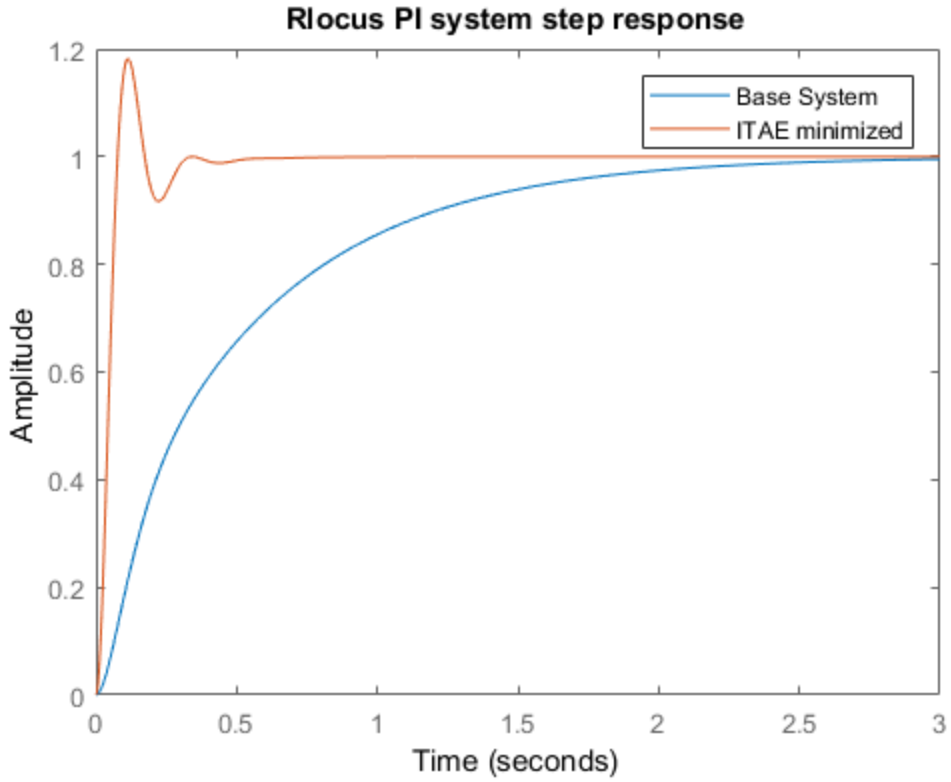
```
Urbase = C/(1+(C*G));
Trbase = (C*G)/(1+(G*C));

Sr = pidsearch(G, C, 'ITAE');

Ursearch = Sr/(1+(Sr*G));
Trsearch = (Sr*G)/(1+(G*Sr));
```

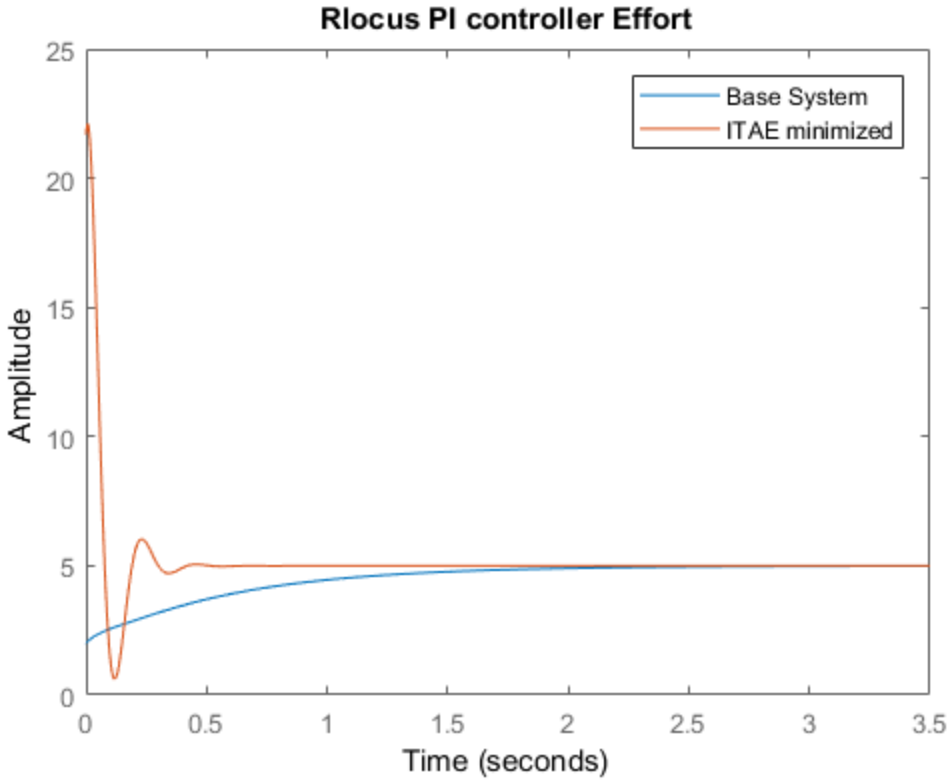
Plot the system stem response

```
figure(2); clf;
step(Trbase)
hold on
step(Trsearch)
title('Rlocus PI system step response')
legend('Base System', 'ITAE minimized')
```



Plot the system controller effort

```
figure(3); clf;
step(Urbase)
hold on
step(Ursearch)
title('Rlocus PI controller Effort')
legend('Base System', 'ITAE minimized')
```

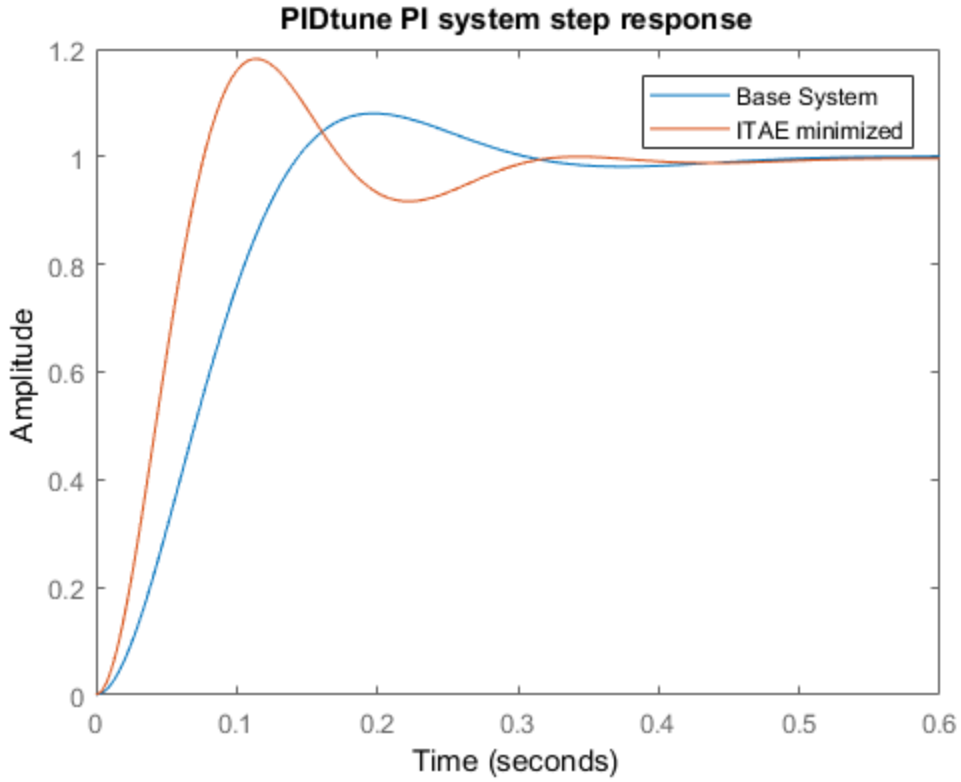


Now for the pidtune controller

```
Ct = pidtune(G, 'PI');
St = pidsearch(G, Ct, 'ITAE');
Utbase = Ct/(1+(Ct*G));
Ttbase = (Ct*G)/(1+(G*Ct));
Utsearch = St/(1+(St*G));
Ttsearch = (St*G)/(1+(G*St));
```

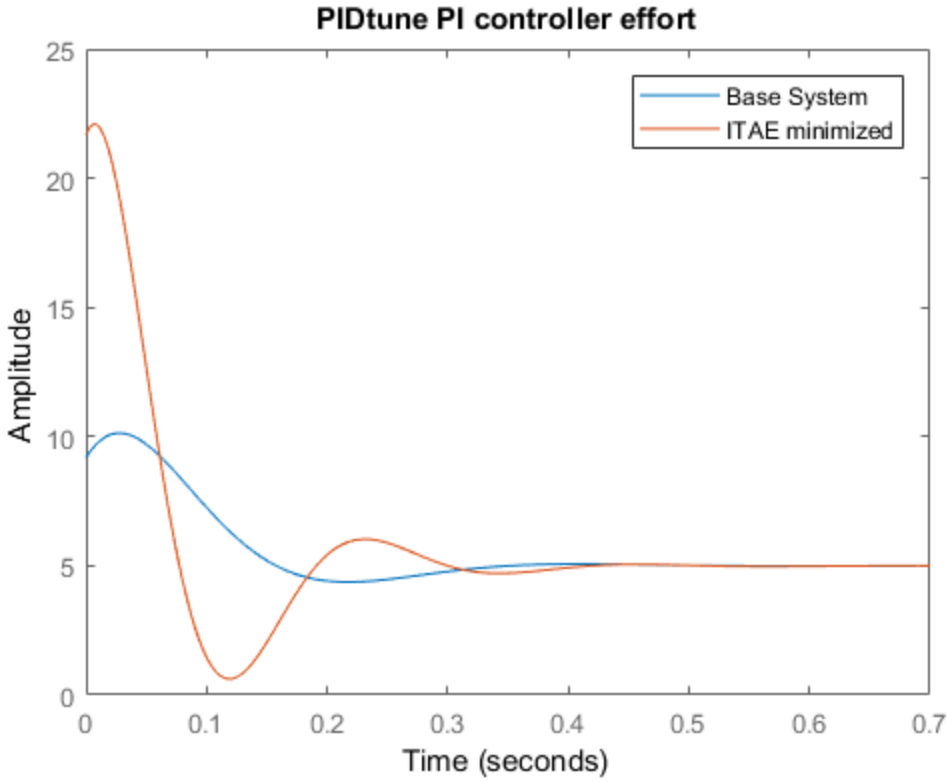
Plot the system stem response

```
figure(4); clf;
step(Ttbase)
hold on
step(Ttsearch)
title('PIDtune PI system step response')
legend('Base System', 'ITAE minimized')
```



Plot the system controller effort

```
figure(5); clf;
step(Utbase)
hold on
step(Utsearch)
title('PIDtune PI controller effort')
legend('Base System', 'ITAE minimized')
```



Use PID Search for minimizing ISE

Start with the root locus found controller

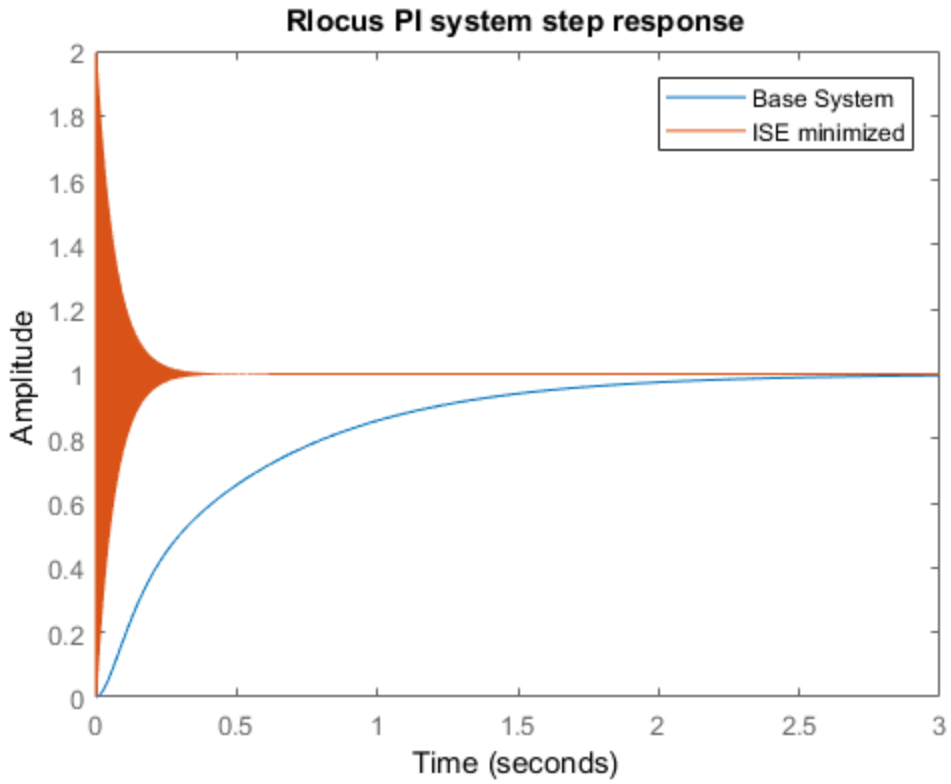
```
Urbase = C/(1+(C*G));
Trbase = (C*G)/(1+(G*C));

Sr = pidsearch(G, C, 'ISE');

Ursearch = Sr/(1+(Sr*G));
Trsearch = (Sr*G)/(1+(G*Sr));
```

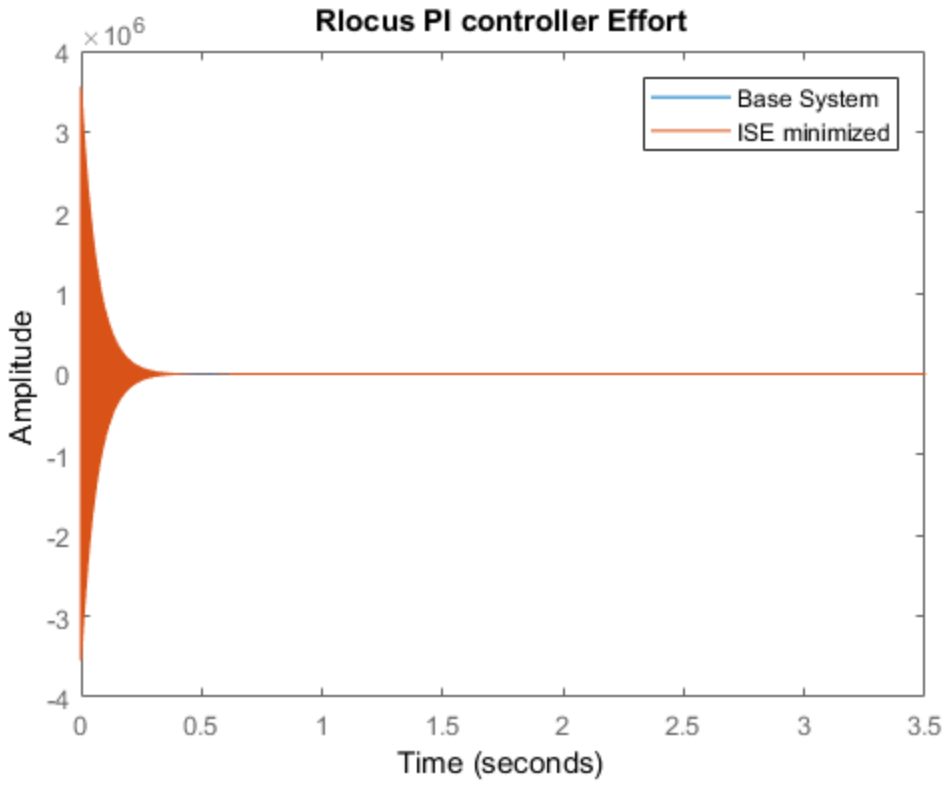
Plot the system stem response

```
figure(6); clf;
step(Trbase)
hold on
step(Trsearch)
title('Rlocus PI system step response')
legend('Base System', 'ISE minimized')
```



Plot the system controller effort

```
figure(7); clf;
step(Urbase)
hold on
step(Ursearch)
title('Rlocus PI controller Effort')
legend('Base System', 'ISE minimized')
```

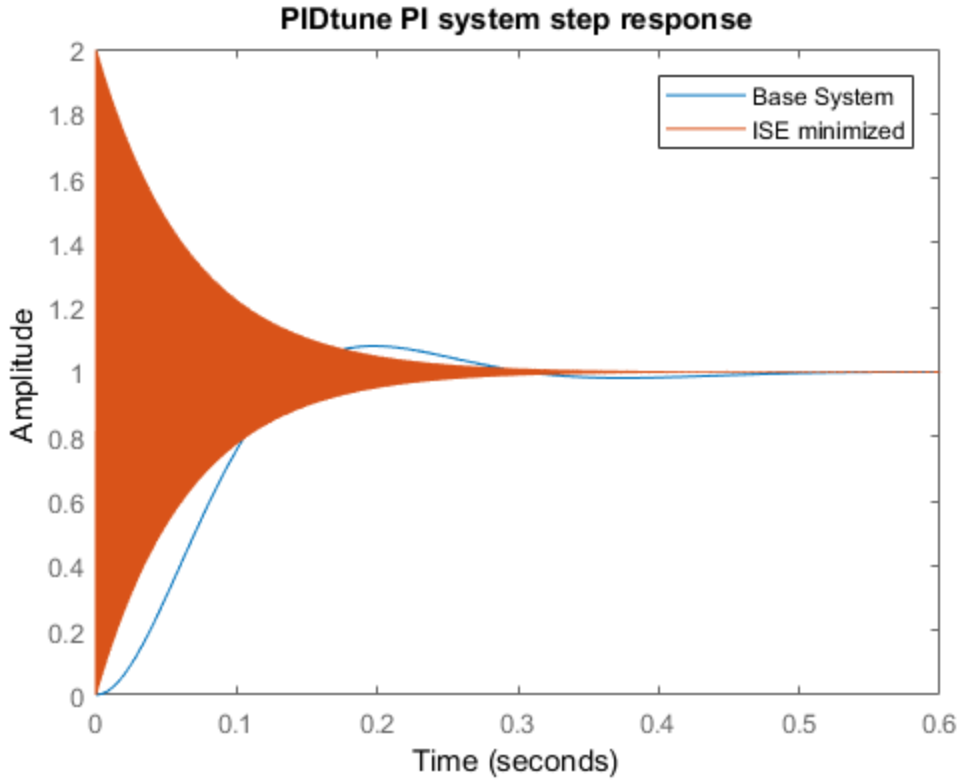


Now for the pidtune controller

```
Ct = pidtune(G, 'PI');
St = pidsearch(G, Ct, 'ISE');
Utbase = Ct/(1+(Ct*G));
Ttbase = (Ct*G)/(1+(G*Ct));
Utsearch = St/(1+(St*G));
Ttsearch = (St*G)/(1+(G*St));
```

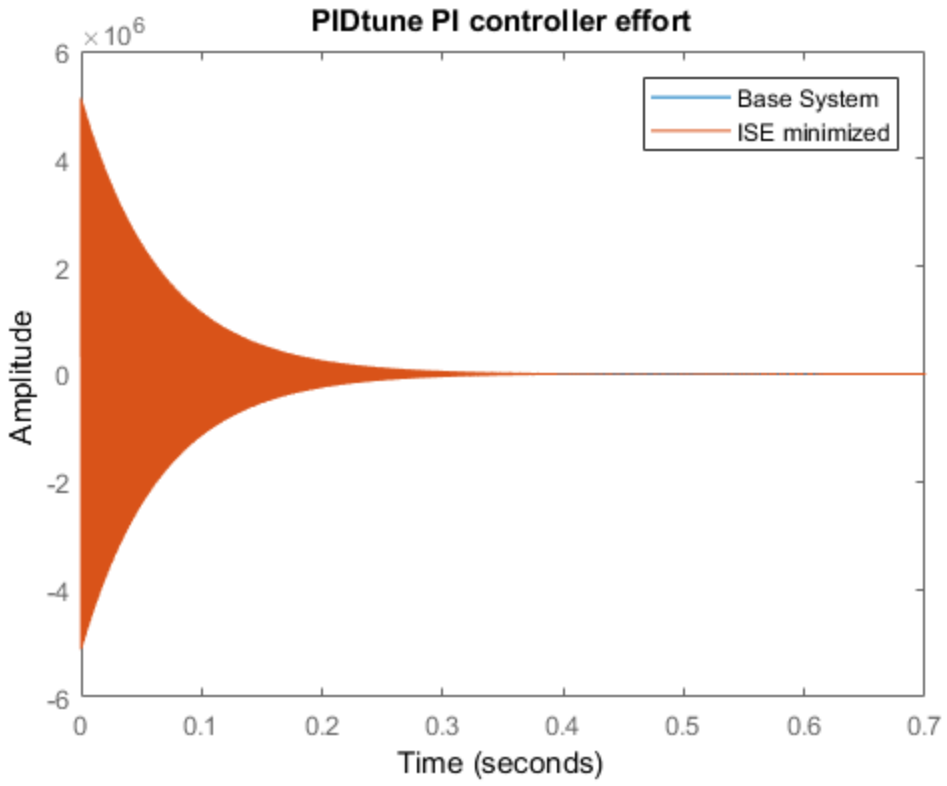
Plot the system stem response

```
figure(8); clf;
step(Ttbase)
hold on
step(Ttsearch)
title('PIDtune PI system step response')
legend('Base System', 'ISE minimized')
```



Plot the system controller effort

```
figure(9); clf;
step(Utbase)
hold on
step(Utsearch)
title('PIDtune PI controller effort')
legend('Base System', 'ISE minimized')
```



Use PID Search for minimizing %OS

Start with the root locus found controller

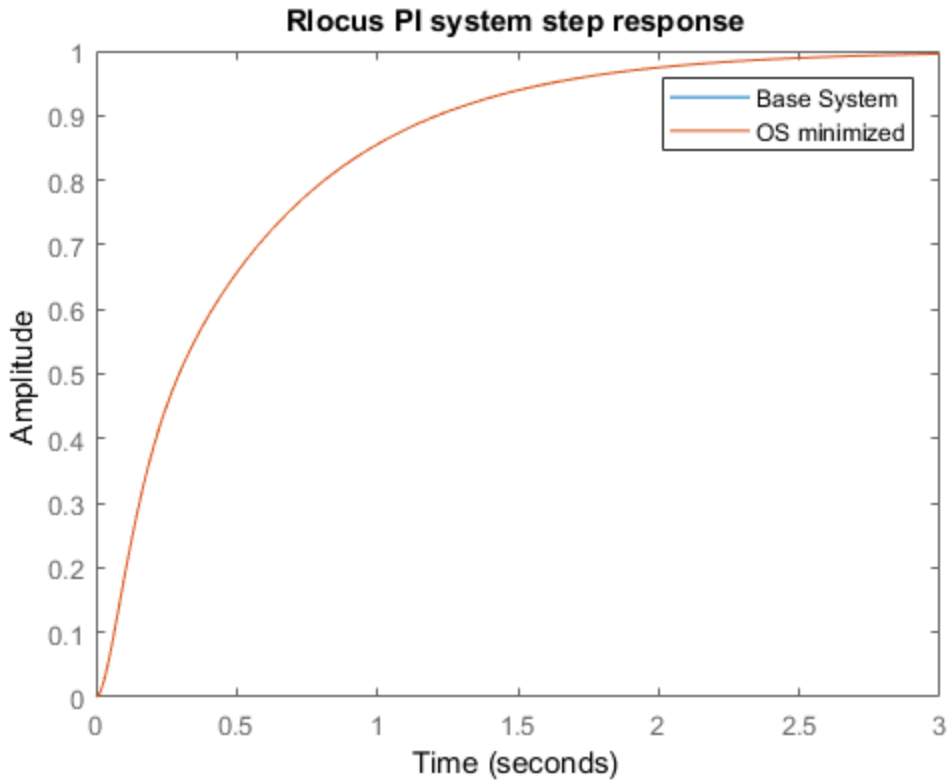
```
Urbase = C/(1+(C*G));
Trbase = (C*G)/(1+(G*C));

Sr = pidsearch(G, C, 'OS');

Ursearch = Sr/(1+(Sr*G));
Trsearch = (Sr*G)/(1+(G*Sr));
```

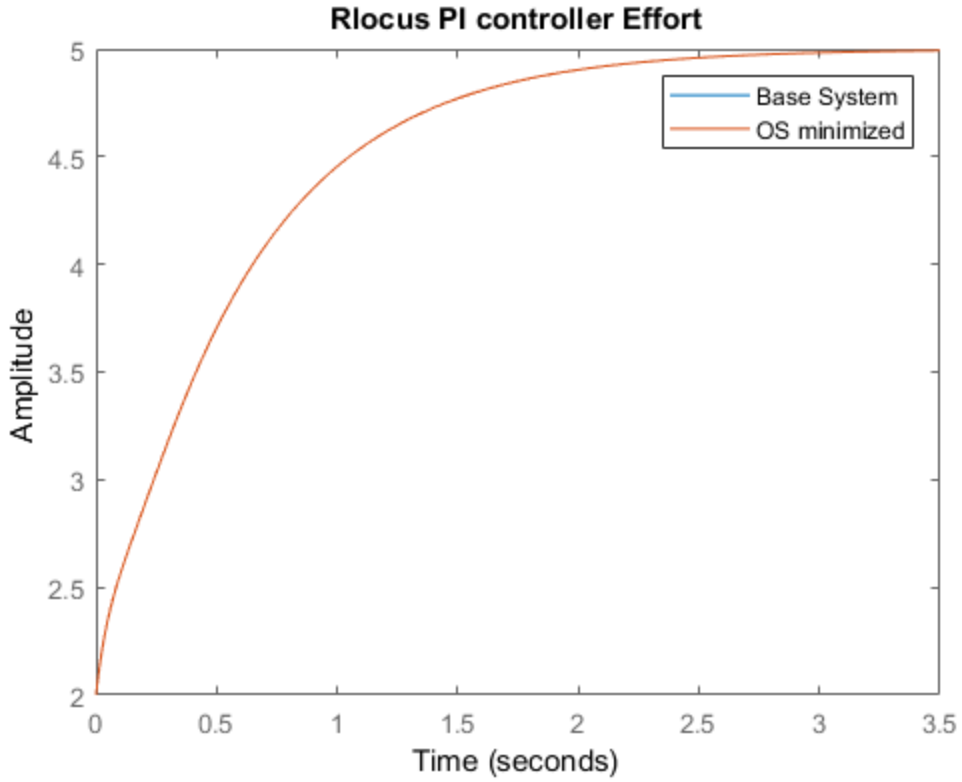
Plot the system stem response

```
figure(10); clf;
step(Trbase)
hold on
step(Trsearch)
title('Rlocus PI system step response')
legend('Base System', 'OS minimized')
```



Plot the system controller effort

```
figure(11); clf;
step(Urbase)
hold on
step(Ursearch)
title('Rlocus PI controller Effort')
legend('Base System', 'OS minimized')
```

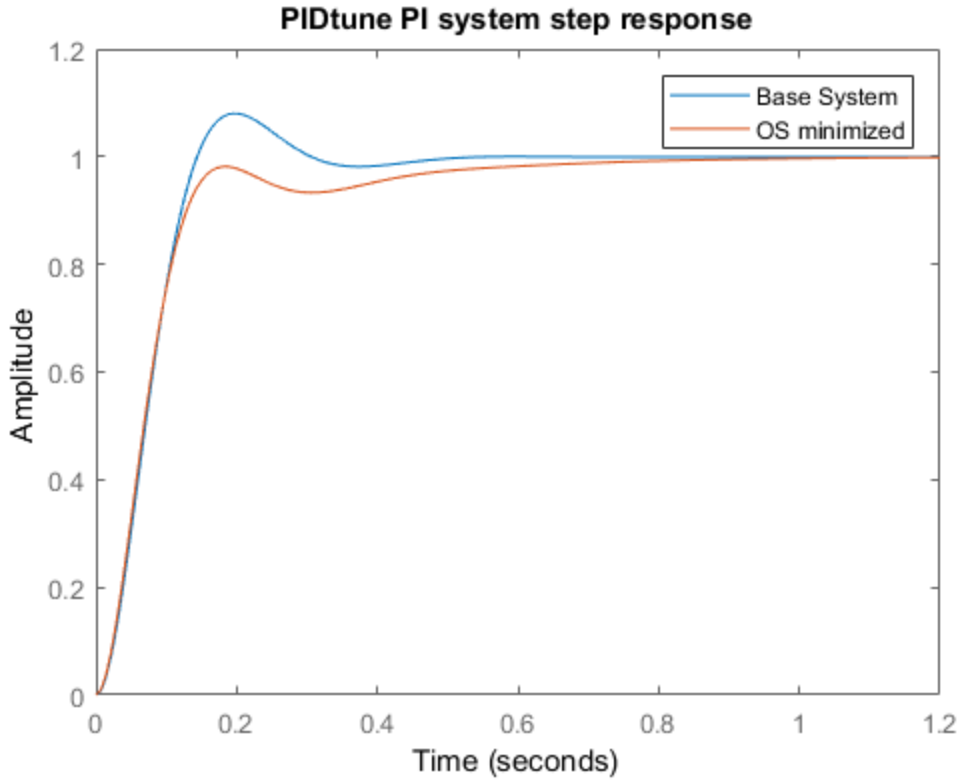


Now for the pidtune controller

```
Ct = pidtune(G, 'PI');
St = pidsearch(G, Ct, 'OS');
Utbase = Ct/(1+(Ct*G));
Ttbase = (Ct*G)/(1+(G*Ct));
Utsearch = St/(1+(St*G));
Ttsearch = (St*G)/(1+(G*St));
```

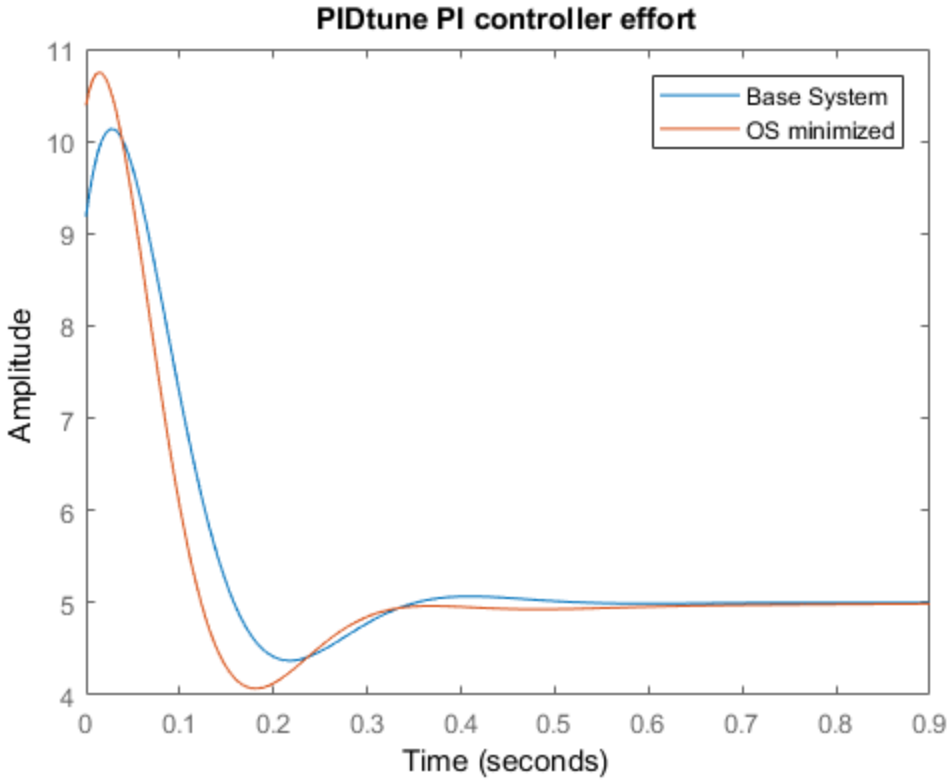
Plot the system stem response

```
figure(12); clf;
step(Ttbase)
hold on
step(Ttsearch)
title('PIDtune PI system step response')
legend('Base System', 'OS minimized')
```



Plot the system controller effort

```
figure(13); clf;
step(Utbase)
hold on
step(Utsearch)
title('PIDtune PI controller effort')
legend('Base System', 'OS minimized')
```



Use PID Search for minimizing Settling Time

Start with the root locus found controller

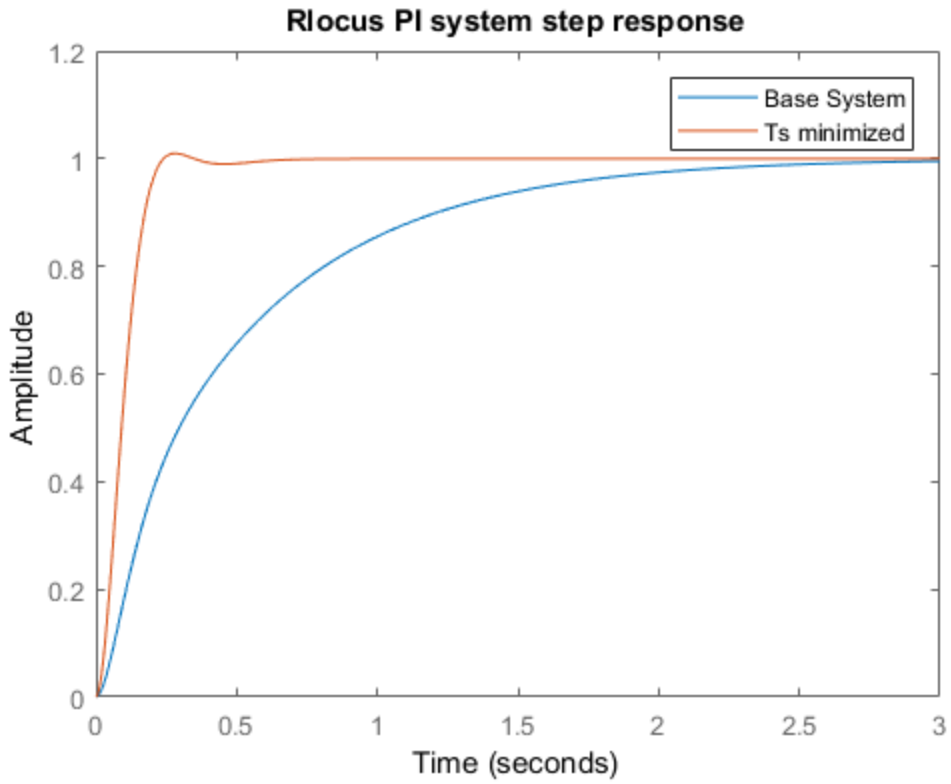
```
Urbase = C/(1+(C*G));
Trbase = (C*G)/(1+(G*C));

Sr = pidsearch(G, C, 'Ts');

Ursearch = Sr/(1+(Sr*G));
Trsearch = (Sr*G)/(1+(G*Sr));
```

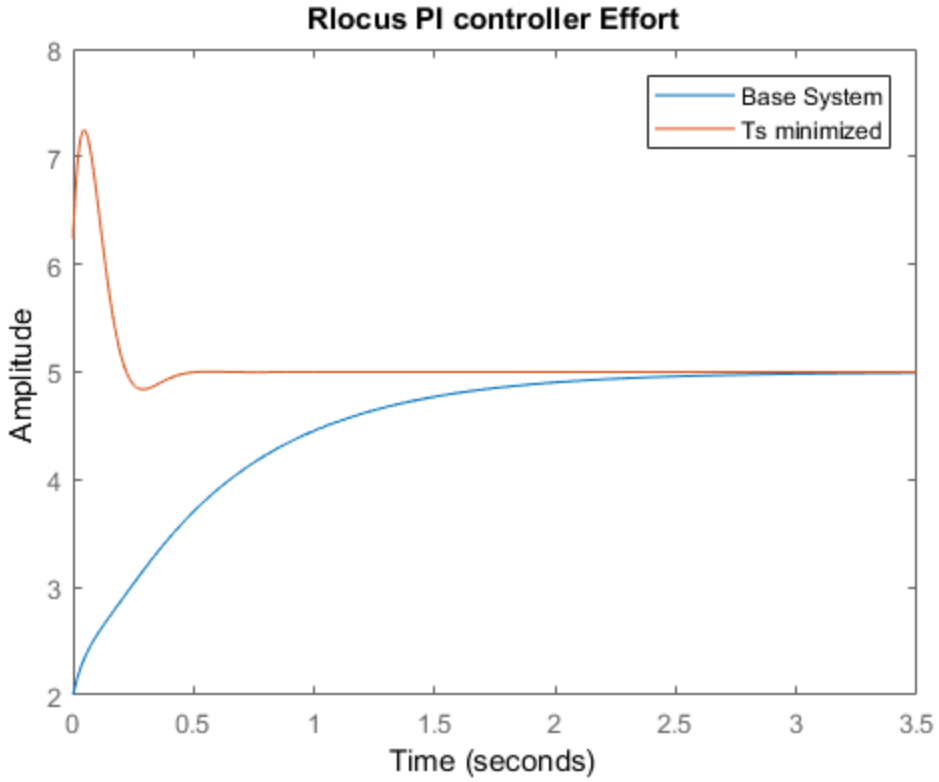
Plot the system stem response

```
figure(14); clf;
step(Trbase)
hold on
step(Trsearch)
title('Rlocus PI system step response')
legend('Base System', 'Ts minimized')
```



Plot the system controller effort

```
figure(15); clf;
step(Urbase)
hold on
step(Ursearch)
title('Rlocus PI controller Effort')
legend('Base System', 'Ts minimized')
```

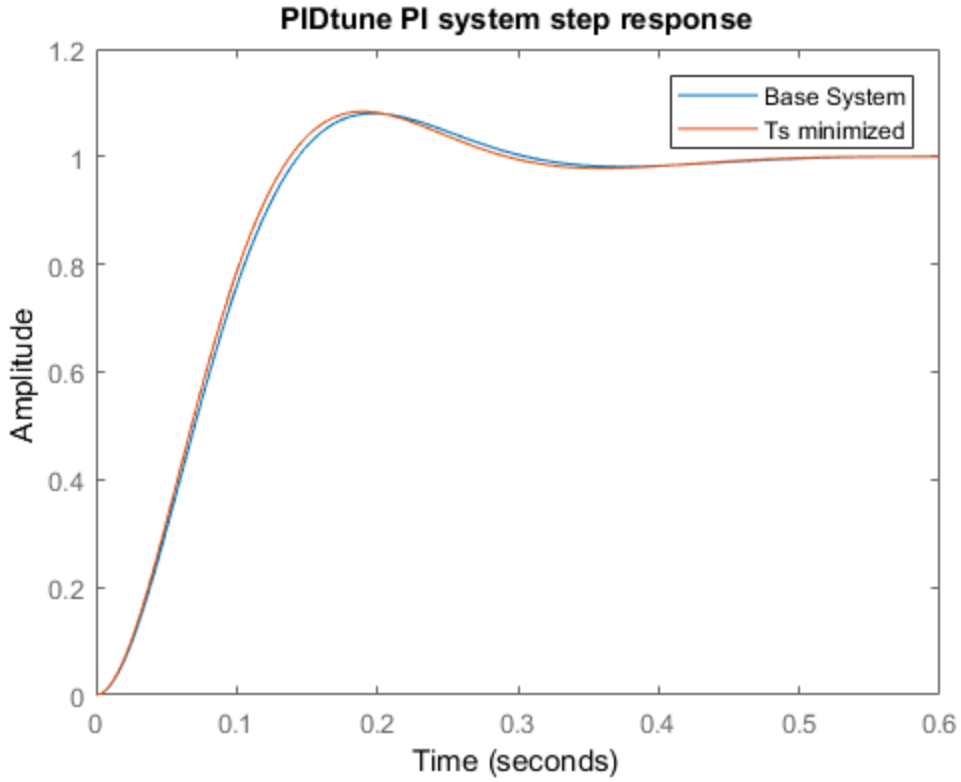


Now for the pidtune controller

```
Ct = pidtune(G, 'PI');
St = pidsearch(G, Ct, 'Ts');
Utbase = Ct/(1+(Ct*G));
Ttbase = (Ct*G)/(1+(G*Ct));
Utsearch = St/(1+(St*G));
Ttsearch = (St*G)/(1+(G*St));
```

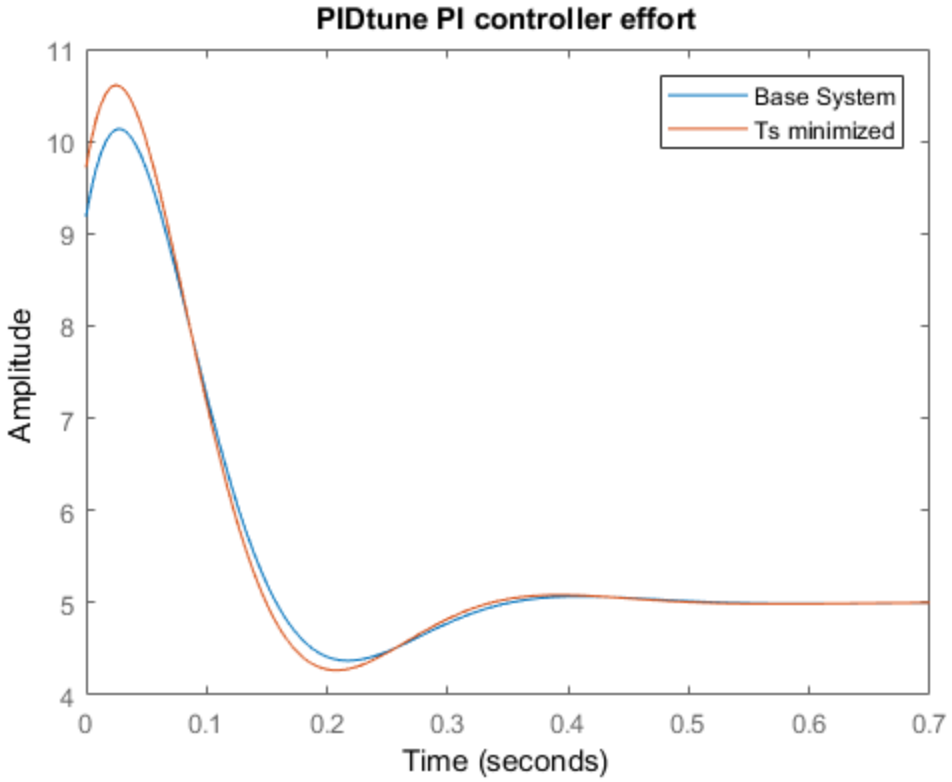
Plot the system stem response

```
figure(16); clf;
step(Ttbase)
hold on
step(Ttsearch)
title('PIDtune PI system step response')
legend('Base System', 'Ts minimized')
```



Plot the system controller effort

```
figure(17); clf;
step(Utbase)
hold on
step(Utsearch)
title('PIDtune PI controller effort')
legend('Base System', 'Ts minimized')
```



Use PID Search for minimizing OSTs

Start with the root locus found controller

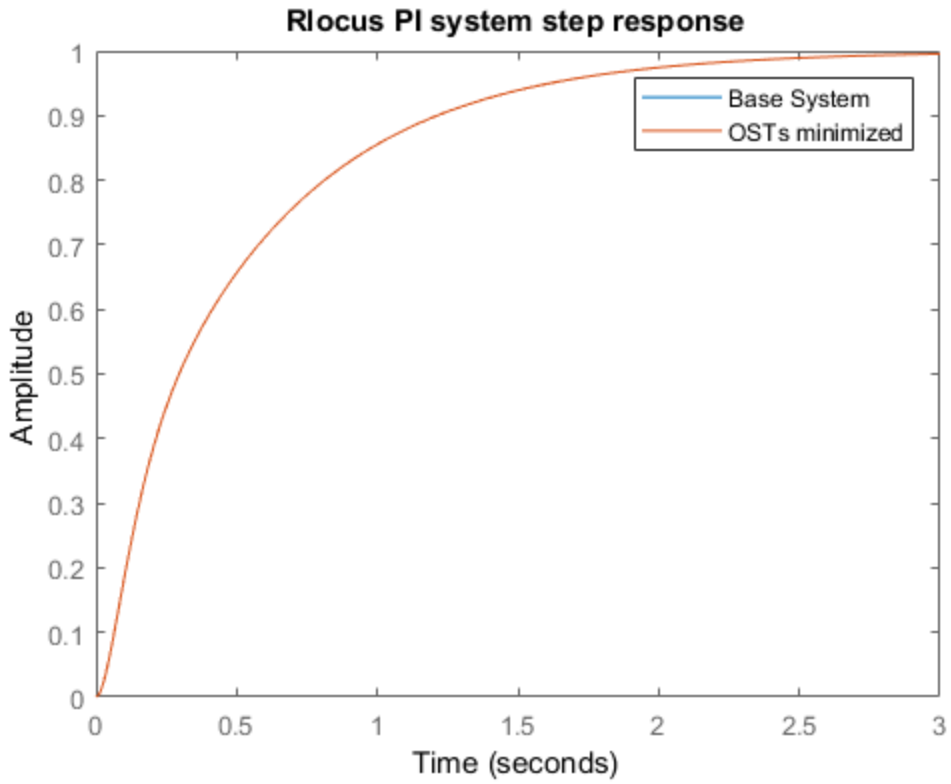
```
Urbase = C/(1+(C*G));
Trbase = (C*G)/(1+(G*C));

Sr = pidsearch(G, C, 'OSTs');

Ursearch = Sr/(1+(Sr*G));
Trsearch = (Sr*G)/(1+(G*Sr));
```

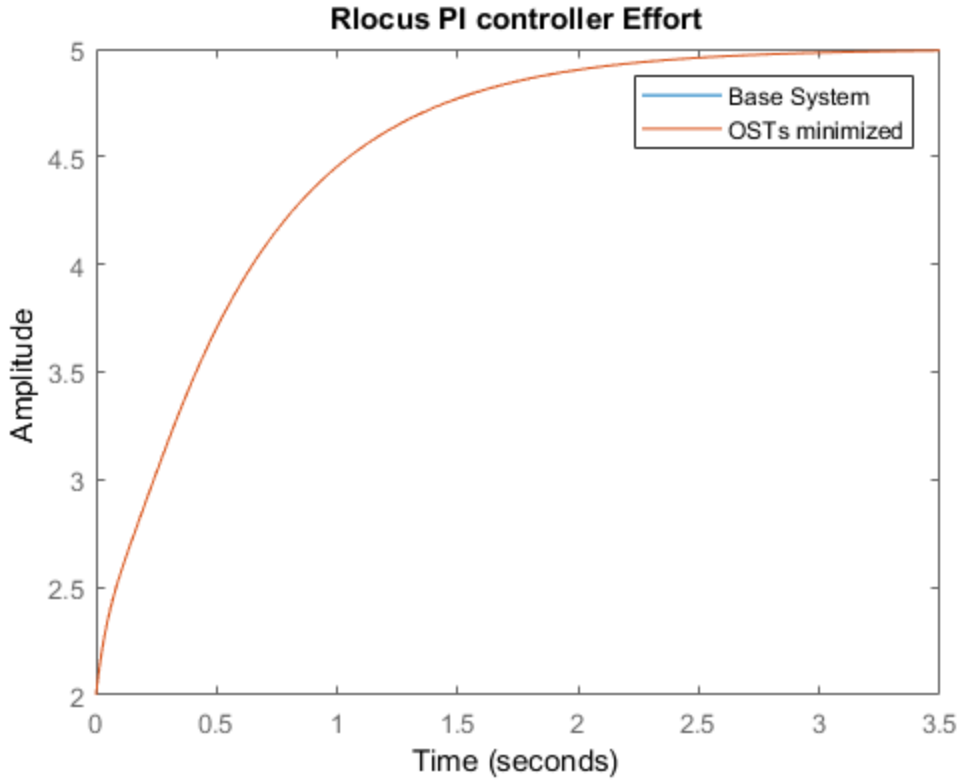
Plot the system stem response

```
figure(18); clf;
step(Trbase)
hold on
step(Trsearch)
title('Rlocus PI system step response')
legend('Base System', 'OSTs minimized')
```



Plot the system controller effort

```
figure(19); clf;
step(Urbase)
hold on
step(Ursearch)
title('Rlocus PI controller Effort')
legend('Base System', 'OSTs minimized')
```

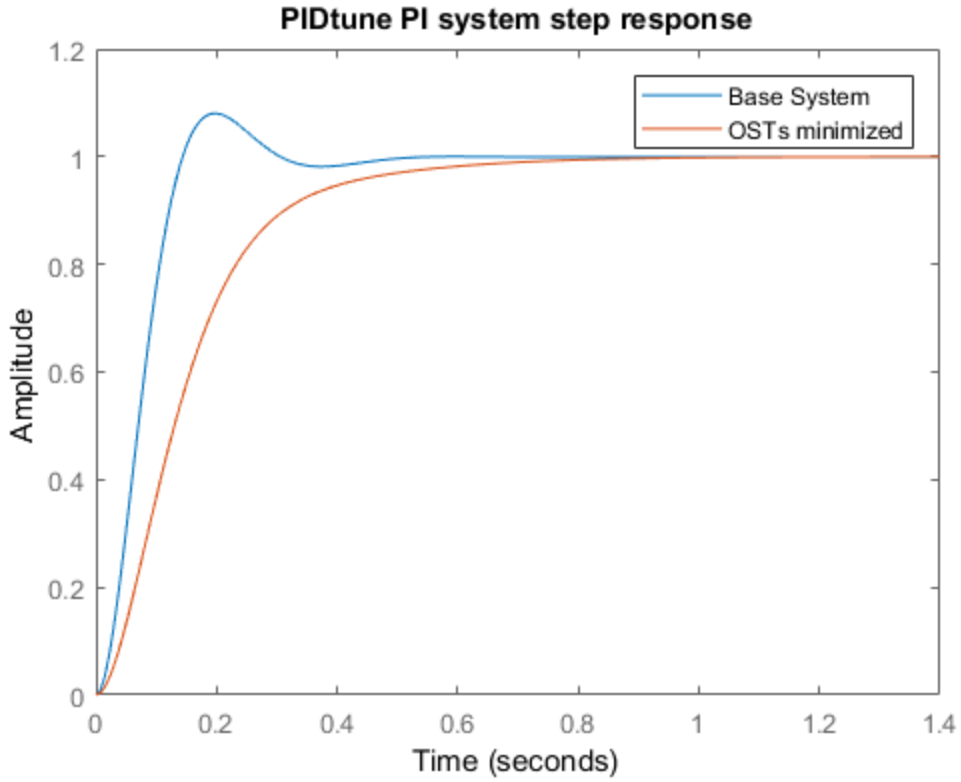


Now for the pidtune controller

```
Ct = pidtune(G, 'PI');
St = pidsearch(G, Ct, 'OSTs');
Utbase = Ct/(1+(Ct*G));
Ttbase = (Ct*G)/(1+(G*Ct));
Utsearch = St/(1+(St*G));
Ttsearch = (St*G)/(1+(G*St));
```

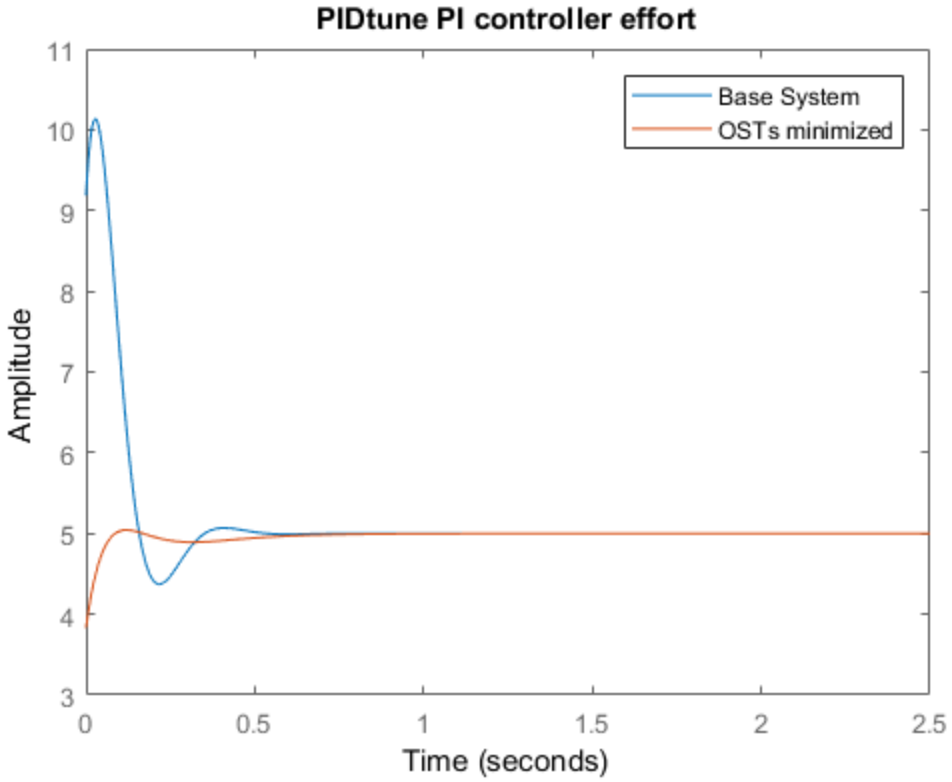
Plot the system stem response

```
figure(20); clf;
step(Ttbase)
hold on
step(Ttsearch)
title('PIDtune PI system step response')
legend('Base System', 'OSTs minimized')
```



Plot the system controller effort

```
figure(21); clf;
step(Utbase)
hold on
step(Utsearch)
title('PIDtune PI controller effort')
legend('Base System', 'OSTs minimized')
```



Use PID Search for minimizing UTs

Start with the root locus found controller

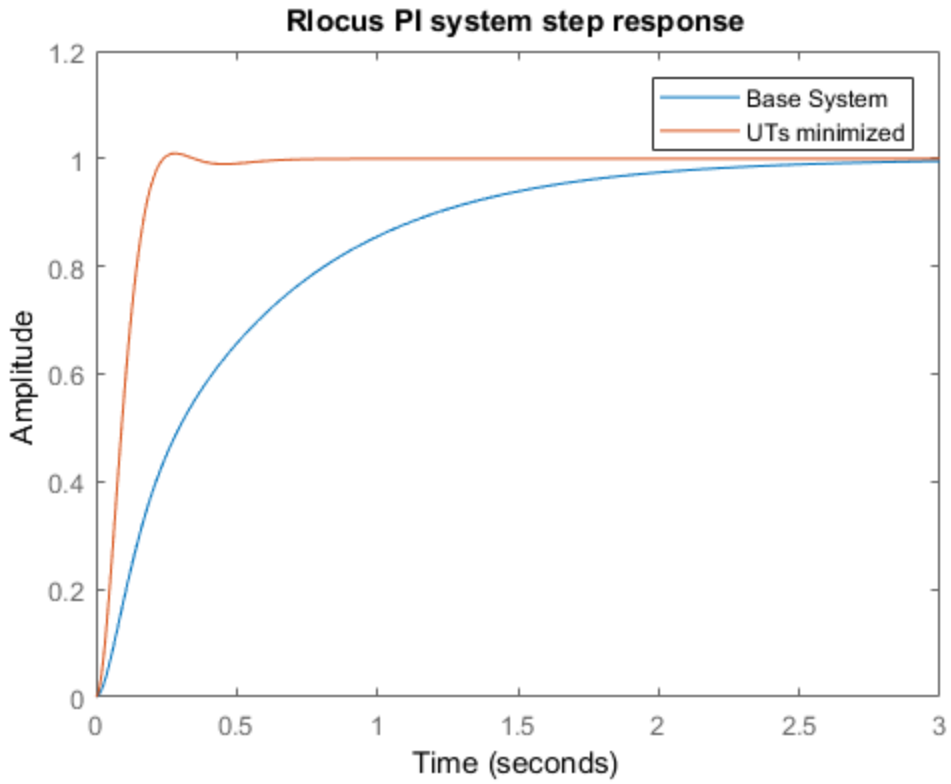
```
Urbase = C/(1+(C*G));
Trbase = (C*G)/(1+(G*C));

Sr = pidsearch(G, C, 'UTs');

Ursearch = Sr/(1+(Sr*G));
Trsearch = (Sr*G)/(1+(G*Sr));
```

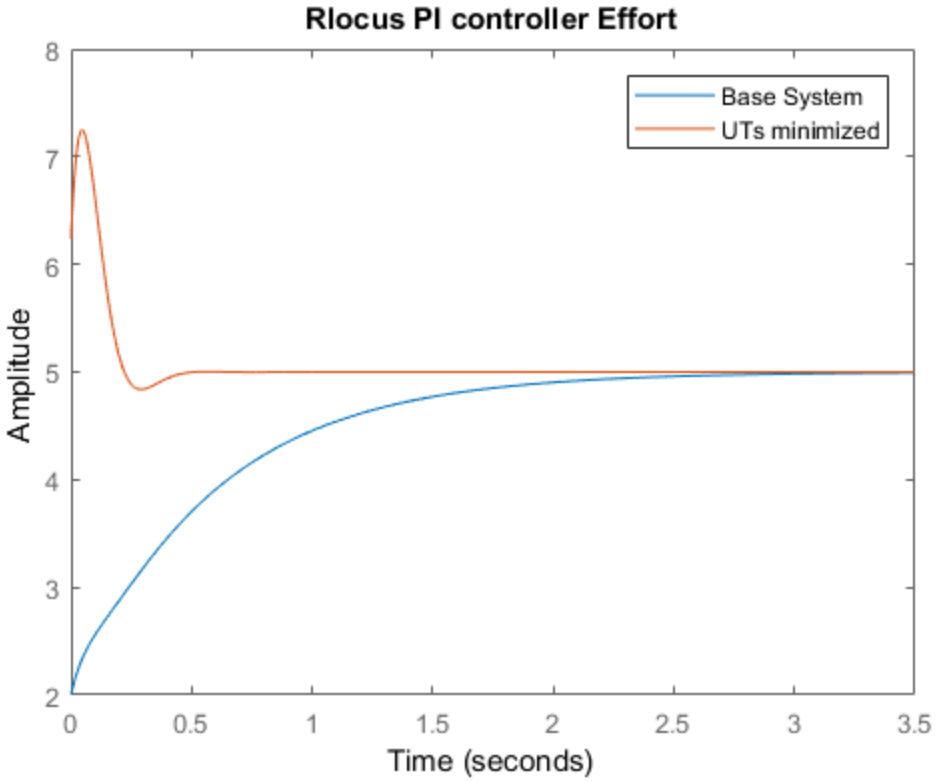
Plot the system stem response

```
figure(22); clf;
step(Trbase)
hold on
step(Trsearch)
title('Rlocus PI system step response')
legend('Base System', 'UTs minimized')
```



Plot the system controller effort

```
figure(23); clf;
step(Urbase)
hold on
step(Ursearch)
title('Rlocus PI controller Effort')
legend('Base System', 'UTs minimized')
```

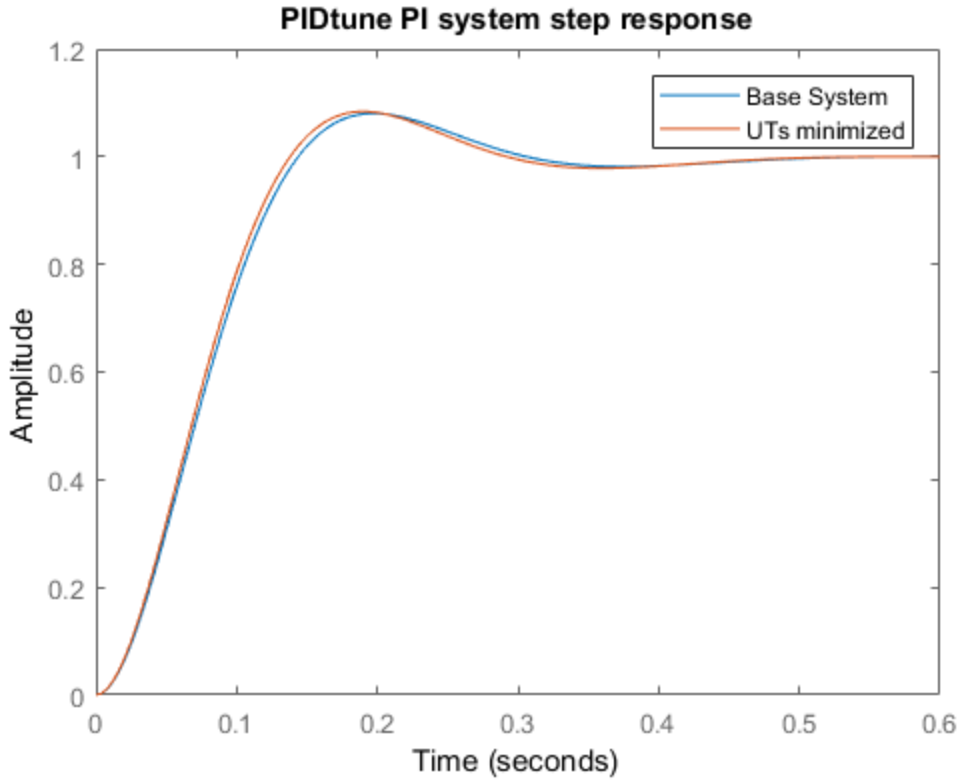


Now for the pidtune controller

```
Ct = pidtune(G, 'PI');
St = pidsearch(G, Ct, 'UTs');
Utbase = Ct/(1+(Ct*G));
Ttbase = (Ct*G)/(1+(G*Ct));
Utsearch = St/(1+(St*G));
Ttsearch = (St*G)/(1+(G*St));
```

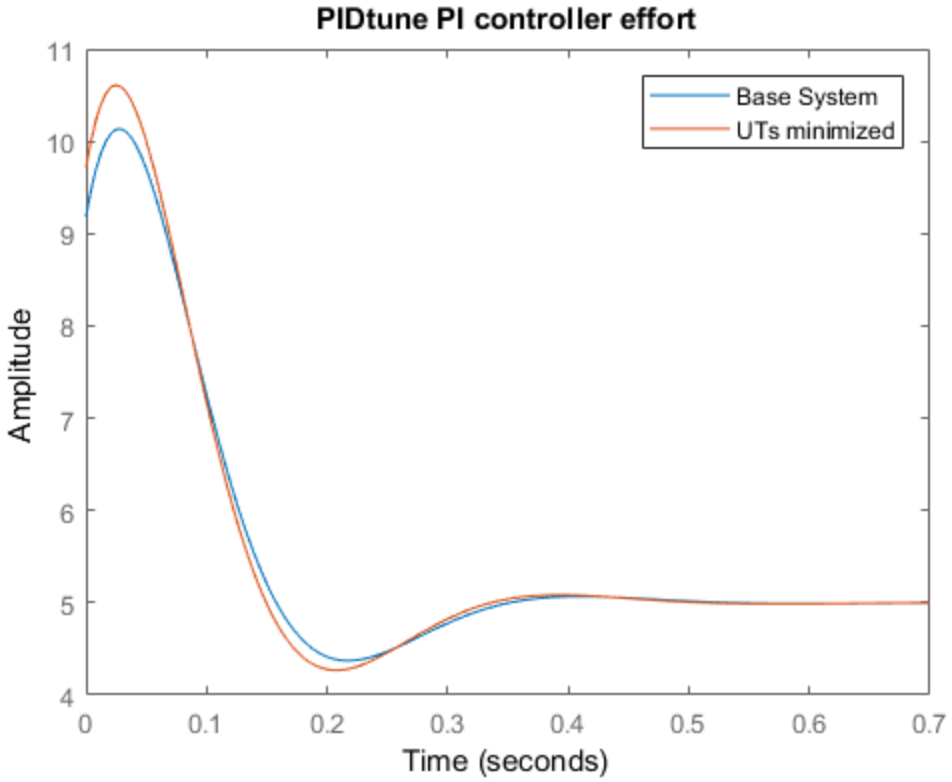
Plot the system stem response

```
figure(24); clf;
step(Ttbase)
hold on
step(Ttsearch)
title('PIDtune PI system step response')
legend('Base System', 'UTs minimized')
```



Plot the system controller effort

```
figure(25); clf;
step(Utbase)
hold on
step(Utsearch)
title('PIDtune PI controller effort')
legend('Base System', 'UTs minimized')
```



Use PID Search for minimizing LQG

Start with the root locus found controller Could not get this one to work

```
% Urbase = C/(1+(C*G));
% Trbase = (C*G)/(1+(G*C));
%
% Sr = pidsearch(G, C, 'LQG', 0.99);
%
% Ursearch = Sr/(1+(Sr*G));
% Trsearch = (Sr*G)/(1+(G*Sr));
%
% %%
% % Plot the system stem response
% figure(26); clf;
% step(Trbase)
% hold on
% step(Trsearch)
% title('Rlocus PI system step response')
% legend('Base System', 'LQG minimized')
%
% %%
% % Plot the system controller effort
% figure(27); clf;
% step(Urbase)
% hold on
```

```
% step(Ursearch)
% title('Rlocus PI controller Effort')
% legend('Base System', 'LQG minimized')
% %%
% % Now for the pidtune controller
%
% Ct = pidtune(G, 'PI');
% St = pidsearch(G, Ct, 'LQG');
% Utbase = Ct/(1+(Ct*G));
% Ttbase = (Ct*G)/(1+(G*Ct));
% Utsearch = St/(1+(St*G));
% Ttsearch = (St*G)/(1+(G*St));
%
% %%
% % Plot the system stem response
% figure(28); clf;
% step(Ttbase)
% hold on
% step(Ttsearch)
% title('PIDtune PI system step response')
% legend('Base System', 'LQG minimized')
%
% %%
% % Plot the system controller effort
% figure(29); clf;
% step(Utbase)
% hold on
% step(Utsearch)
% title('PIDtune PI controller effort')
% legend('Base System', 'LQG minimized')
```

Published with MATLAB® R2017a