Viene assegnata la funzione di trasferimento del sistema da controllare $G(s)$:

$$G(s) = \frac{1}{s(s+5)^2}$$

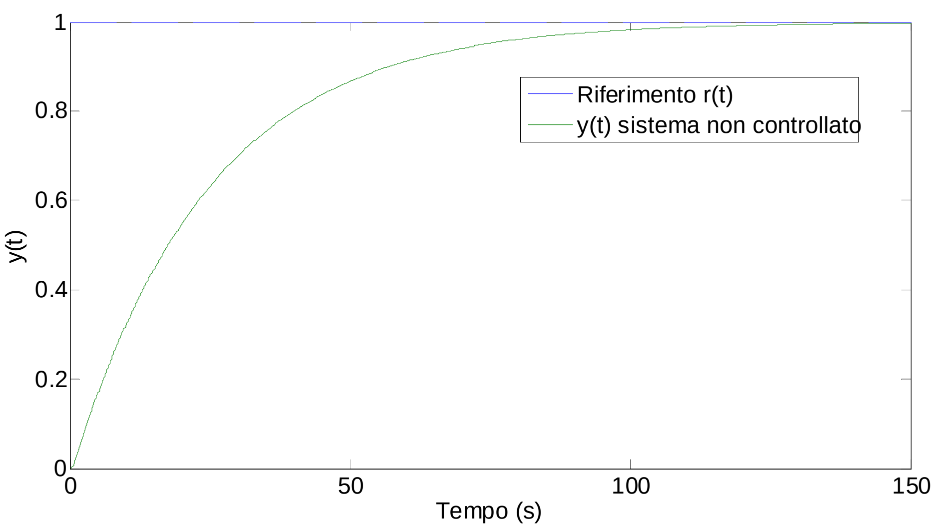
che è inserita nell'ambiente Matlab come segue:

```
>> s=tf('s')

Transfer function:
s

>> Gs=1/(s*(s+5)^2)

Transfer function:
         1
-------------------
s^3 + 10 s^2 + 25 s
```

Sono quindi definiti i vettori corrispondenti della funzione di trasferimento $G(s)$ per essere poi utilizzati in Simulink:

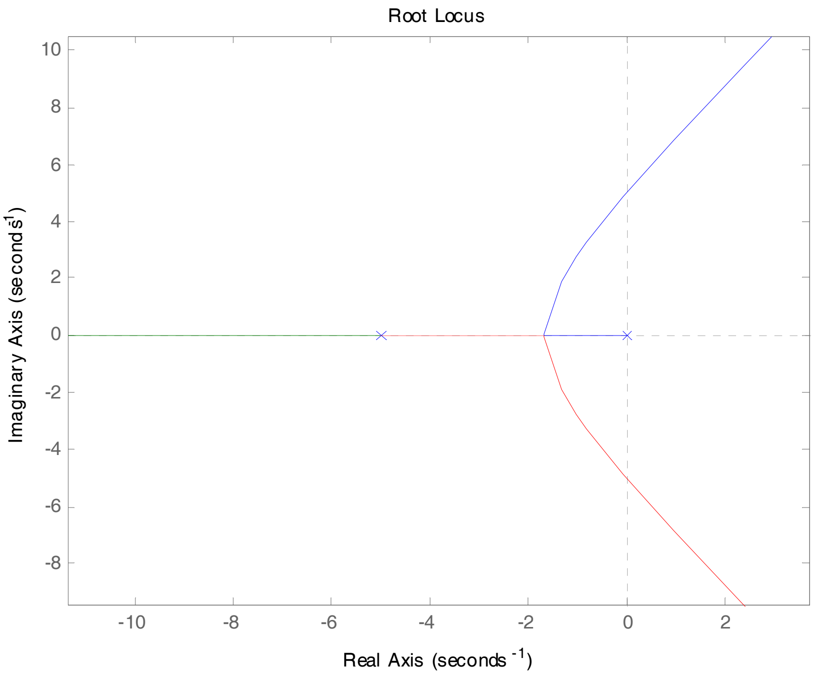
```
>> [numGs,denGs]=tfdata(Gs,'v')

numGs =

     0     0     0     1


denGs =

     1    10    25     0

>>
```

Il sistema non compensato $G(s)$ in retroazione unitaria in risposta al gradino di riferimento presenta una risposta riportata nella fugura seguente, caratteristica di un modello del 1° ordine, come già osservato in precedenza:

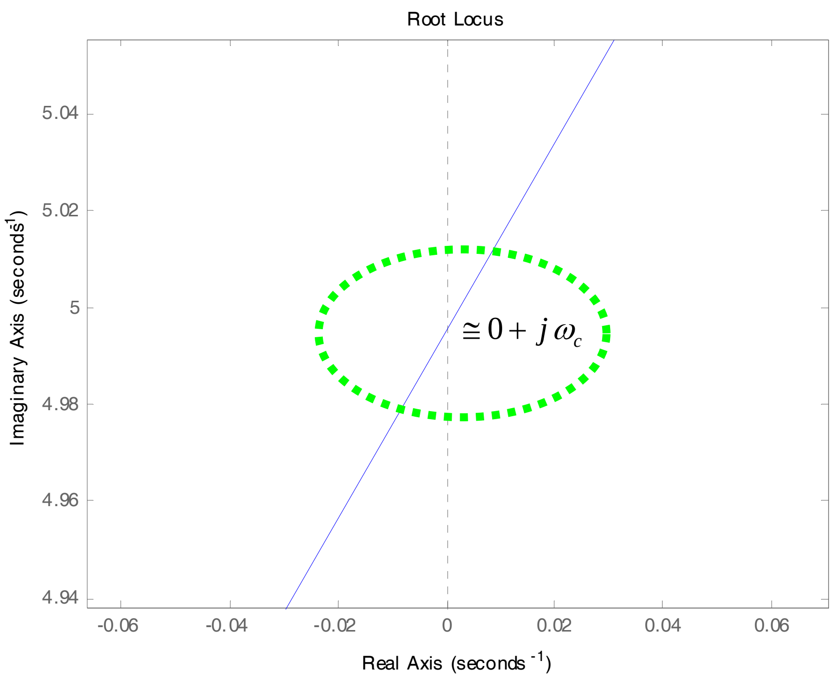


Si disegna quindi il luogo delle radici del sistema in retroazione unitaria attraverso la funzione Matlab:

```
>> rlocus(Gs)
```

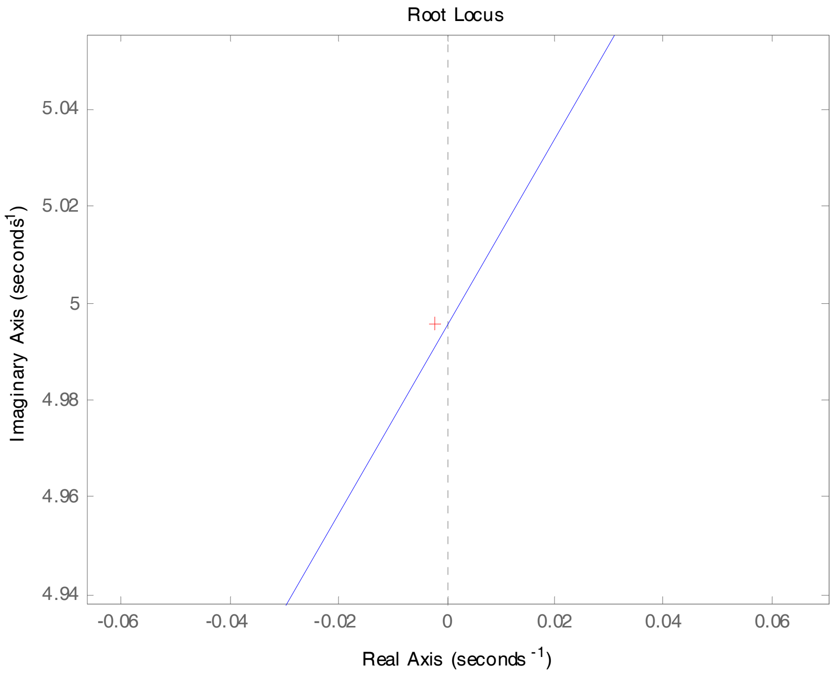


e si ingrandiche il luogo in prossimità del punto in cui intercetta l'asse delle ordinate. Tale punto corrisponde ad uno dei due poli complessi coniugati, e in particolare a quello con parte immaginaria positiva definito dal guadagno critico K_c da determinare come segue.



Usando la funzione Matlab, ed il corrispondente strumento interattivo, si determina col mouse la posizione di tale punto, come indicato dal '+' rosso nella figura successiva.

```
>> Kc=rlocfind(Gs)
Select a point in the graphics window

selected_point =

  -0.0001 + 4.9950i


Kc =

  249.4970

>>
```



Tale valore di guadagno $K_c \cong 249.5$ determina il punto evidenziato sopra, con parte reale circa nulla (0.0001), e parte immaginaria indicata da $\omega_c \cong 4.9950$, utilizzata poi per calcolare il periodo critico delle oscillazioni secondo la relazione $P_c = {2\pi}/{\omega_c}$:

```
>> Wc=4.9950

Wc =

    4.9950

>> Pc=2*pi/Wc

Pc =
```

```
    1.2579

>>
```

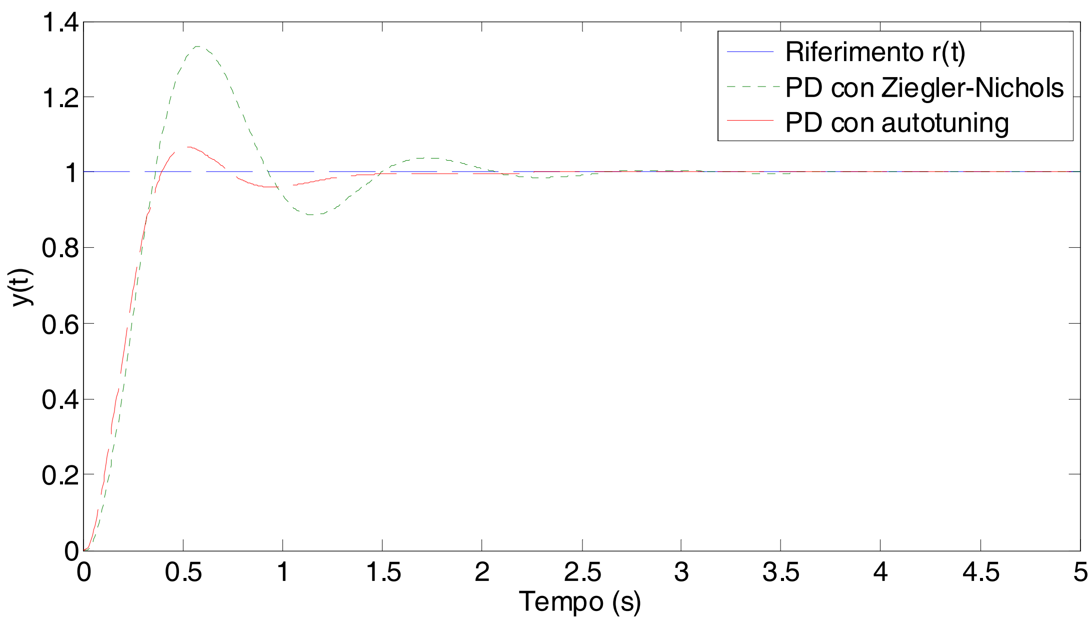
Il problema richiede quindi di calcolare i parametri di un regolatore con la sola azione proporzionale e derivativa, ovvero di tipo PD, dal momento che il sistema da controllare risulta già di tipo zero (cioè con già un polo nell'origine), e non sarebbe opportuno controllarlo mediante un regolatore PID completo. Le formule di Ziegler-Nichols, nel caso vengano utilizzate per determinare i parametri di un regolatore solo PD, risultano le seguenti:

$$\begin{cases} K_P = 0.8 \cdot K_c \\ K_d = {K_p \cdot P_c}/{8} \end{cases}$$

che in Matlab vengono facilmente impelmentate come segue:

```
>> Kp=0.8*Kc

Kp =

  199.5976

>> Kd=Kp*Pc/8

Kd =

   31.3841

>>
```

Una volta realizzato lo schema Simulink riportato alla fine del documento, la figura seguente mostra il confronto tra la risposta del sistema controllato dal PD i cui parametri vengono calcolati con le formule di Ziegler-Nichols, e quella ottenuta dal PD con taratura automatica.



Si noti l'efficacia del PD ottenuto con la taratura automatica rispetto a quello i cui parametri vengono calcolati in maniera empirica. Al fine di determinare il periodo di campionamento per l'implementazione dei corrispondenti PD digitali, viene valutato il tempo di assestamento T_a del sistema più veloce.

Il tempo di assestamento ottenuto dal PD con taratura empirica è il seguente:

```
>> lsiminfo(yc1,t)

ans =

    SettlingTime: 1.9222
             Min: 0
         MinTime: 0
             Max: 1.3326
         MaxTime: 0.5846
```

mentre quello con taratura automatica risulta:

```
>> lsiminfo(yc2,t)

ans =

    SettlingTime: 1.1918
             Min: 0
         MinTime: 0
             Max: 1.0681
         MaxTime: 0.5209

>>
```

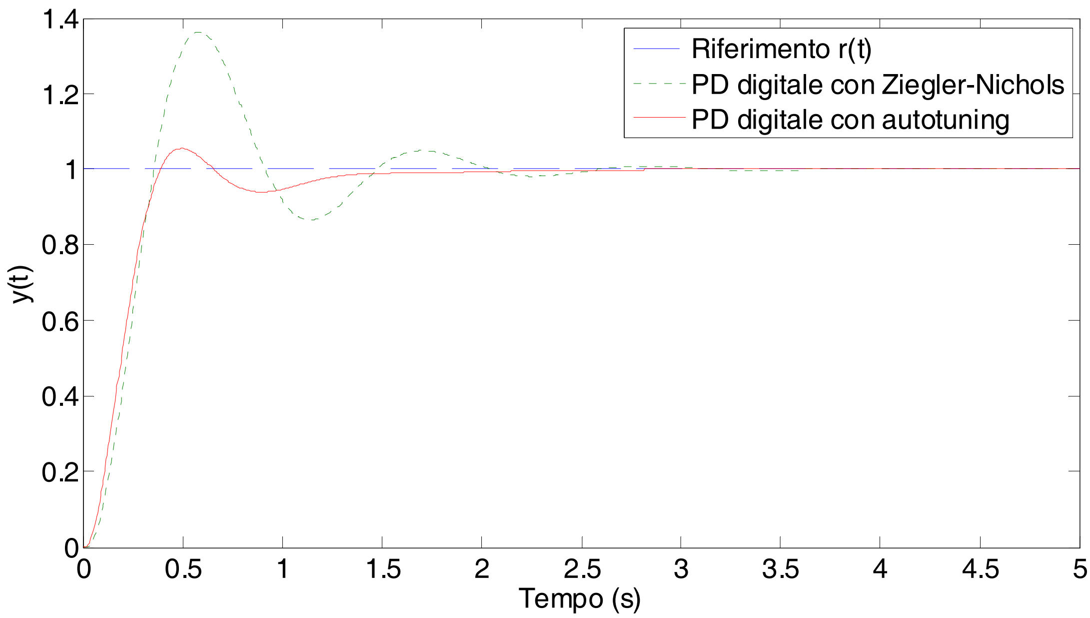
Si sceglie quindi il tempo di assestamento inferiore, arrotondandolo a circa $T_a \approx 1s$, e quindi il periodo di campionamento è determinato dalla relazione $T = {}^{T_a}\!/_{100}$:

```
>> Ta=1

Ta =

     1

>> T=1/100

T =

    0.0100

>>
```

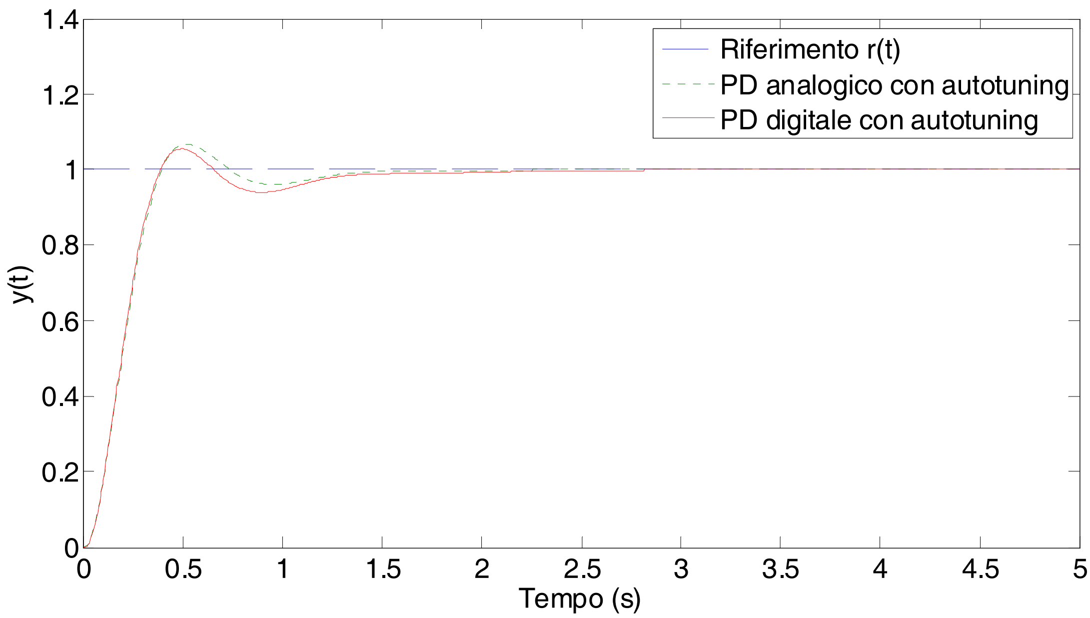
che porta alla scelta di un periodo di campionamento $T = 0.01s$. La figura seguente riporta perciò il confronto delle risposte ottenute dal PD i cui parametri sono stati ottenuti dalle formule di Ziegler-Nichols (le stesse usate per il PD analogico) e quella del PD digitale con parametri ottenuti dalla taratura automatica.

Si ricordi che il metodo di discretizzazione da utilizzare per la parte derivativa (voce ‘Filter Method’) e da impostare nel menù all’interno del blocco Simulink per il regolatore PD risulta quello di Eulero all’Indietro (‘Backward Euler’).

La figura seguente confronta la risposta del PD digitale con parametri calcolati con le formule di Ziegler-Nichols ed il PD digitale con taratura automatica. Si osservi il miglioramento ottenuto anche in questo caso impiegando il metodo di taratura automatica anziché il PD con le formule empiriche di Ziegler-Nichols.



La figura successiva riporta invece il confronto tra il PD analogico con autotuning e il PD digitale con autotuning. Si osserva che il PD digitale con autotuning ha caratteristiche lievemente migliori rispetto all’analogo PD analogico. Infatti, il metodo di taratura automatica, quando impiegato con lo schema che comprende anche il dispositivo di tenuta di ordine zero, tiene conto a priori della persenza di tale dispositivo di interfaccia D/A, cercando di contrastarne gli effetti negativi.



L'ultima figura nel seguito illustra invece l'implementazione in Simulink degli schemi impiegati per la simulazione dei PD digitale e analogico, i cui parametri sono ottenuti con le formule empiriche di Ziegler-Nichols e la metodologia di taratura automatica fornita dallo stesso blocco Simulink dei regolatori PID.