Soluzione dell'Esercizio sul Progetto Indiretto del Regolatore

Viene assegnato il sistema descritto dalla seguente funzione di trasferimento:

$$G(s) = 0.2\frac{(1-2s)}{s(1+10s)(1+0.1s)}$$

e per la quale si è progettata una rete correttrice a tempo continuo secondo la funzione di trasferimento:

$$R(s) = \frac{1+10s}{1+0.1s}$$

che garantisce un margine di fase al sistema controllato di almeno 60°.

Usando il tempo di campionamento assegnato pari a $T = 1s$, progettare i regolatori equivalenti a tempo discreto di $R(s)$ seconto il metodo di Tustin (TU) e l'Hold Equivalente (HE), e verificarne le proprietà di stabilità.

Nota: la funzione di trasferimento $G(s)$ caratterizza i cosiddetti *sistemi a fase non minima*. Sono modelli in cui sono presenti *zeri a parte reale positiva*. Tali impianti pongono diverse problematiche per quello che riguarda il progetto dei sistemi di controllo, dal momento che la risposta iniziale al gradino di riferimento segue una "direzione" opposta (definito anche comportamento in "controtendenza") rispetto a quella dell'andamento del segnale di riferimento. Si osservi a tal proposito il grafico della risposta del sistema come verrà riportato nel seguito.

Viene assegnata la funzione di trasferimento del sistema da controllare:

$$G(s) = 0.2\frac{(1-2s)}{s(1+10s)(1+0.1s)}$$

che si inserisce in Matlab con le istruzioni sotto:

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

Transfer function:
s

>> Gs=0.2*(1-2*s)/(s*(1+10*s)*(1+0.1*s))

Transfer function:
   -0.4 s + 0.2
------------------
s^3 + 10.1 s^2 + s

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

numGs =

         0         0   -0.4000    0.2000



denGs =

    1.0000   10.1000    1.0000         0

>>
```

e si assegna la rete correttrice (di tipo anticipatrice) secondo la funzione di trasferimento:

$$R(s) = \frac{1+10s}{1+0.1s}$$

che garantisce un margine di fase al sistema controllato di almeno 60°, secondo le seguenti istruzioni:

```
>> Rs=(1+10*s)/(1+0.1*s)

Transfer function:
10 s + 1
---------
0.1 s + 1

>> [numRs,denRs]=tfdata(Rs,'v')

numRs =

    10     1


denRs =

    0.1000    1.0000

>>
```

Si verifica poi il margine di fase del sistema non compensato, pari a 23°, che risulta migliorato dalla rete correttrice progettata, e risulta pari a circa 64°, come evidenziato attraverso le seguenti istruzioni:

```
>>
>> Gsa=Rs*Gs

Transfer function:
      -4 s^2 + 1.6 s + 0.2
---------------------------------
0.1 s^4 + 2.01 s^3 + 10.2 s^2 + s

>>

>> [Gm,Pm]=margin(Gs)

Gm =
```

```
    2.3821

Pm =

   23.0998

>> [Gm,Pm]=margin(Gsa)

Gm =

    2.4390
```

Pm =

63.9346

```
>>
```

Si definisce poi il tempo di campionamento pari a 1s con la variabile $T = 1s$:

>> **T=1**

```
T =

     1
```

e si progetta la prima rete correttrice a tempo discreto, $R_1(z)$, secondo il metodo di Tustin (TU), equivalente alla seguente sostituzione:

$$R_1(z) = R(s)\Big|_{s=\frac{2}{T}\frac{z-1}{z+1}}$$

che viene calcolata automaticamente dalla funzione Matlab `c2d` usando l'opzione `'tu'`, secondo le seguenti istruzioni:

>> **R1z=c2d(Rs,T,'tu')**

```
Transfer function:
17.5 z - 15.83
```

```
--------------
  z + 0.6667
```

Allo stesso modo, si calcola l'equivalente a tempo discreto di $R(s)$ secondo il metodo dell'Hold Equivalence (HE), ovvero:

$$R_2(z) = Z\left[\frac{1-e^{-sT}}{s}R(s)\right] = (1-z^{-1})Z\left[\frac{R(s)}{s}\right]$$

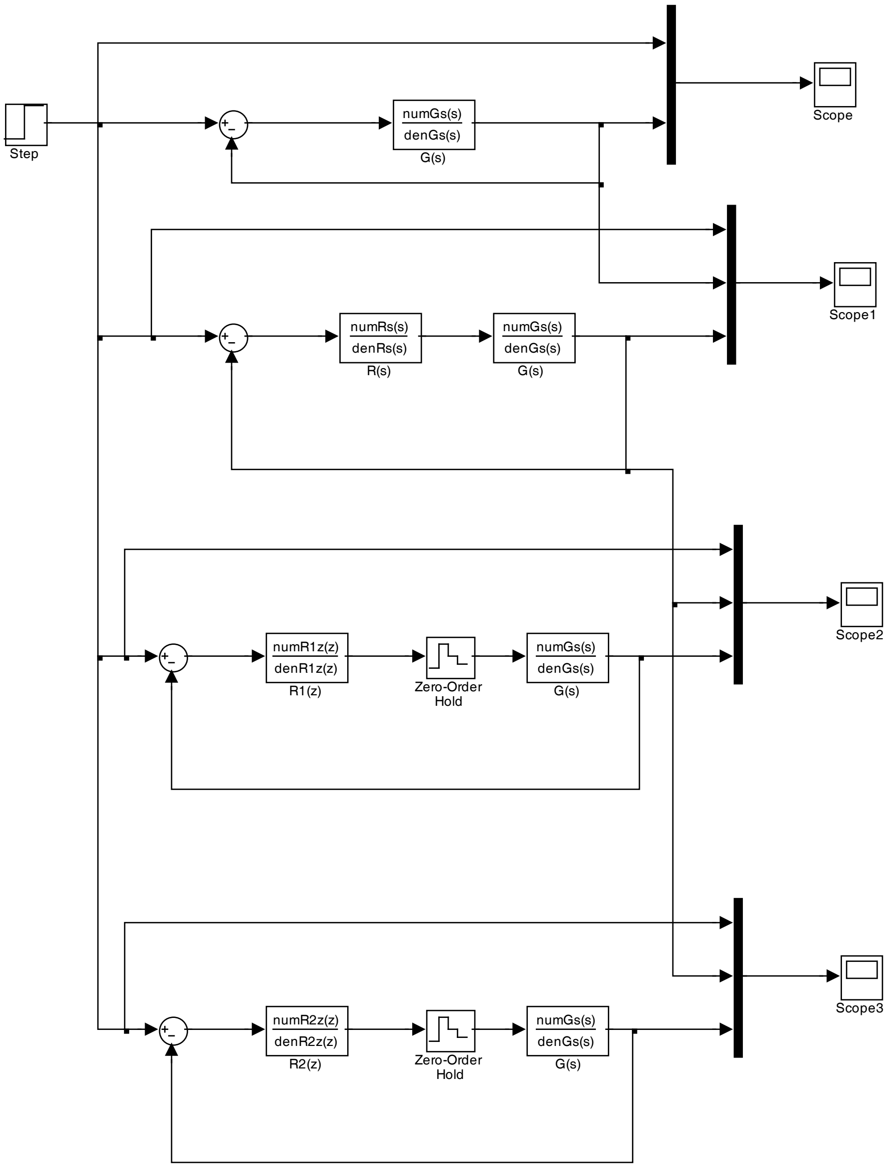
che si ottiene sempre dalla funzione Matlab `c2d` usando l'opzione (di default) `'zoh'`:

```
Sampling time (seconds): 1
>> R2z=c2d(Rs,T,'zoh')

Transfer function:
 100 z - 99
-------------
z - 4.54e-005

Sampling time (seconds): 1
>>


>> [numR1z,denR1z]=tfdata(R1z,'v')

numR1z =

   17.5000  -15.8333


denR1z =

    1.0000    0.6667

>> [numR2z,denR2z]=tfdata(R2z,'v')

numR2z =
```

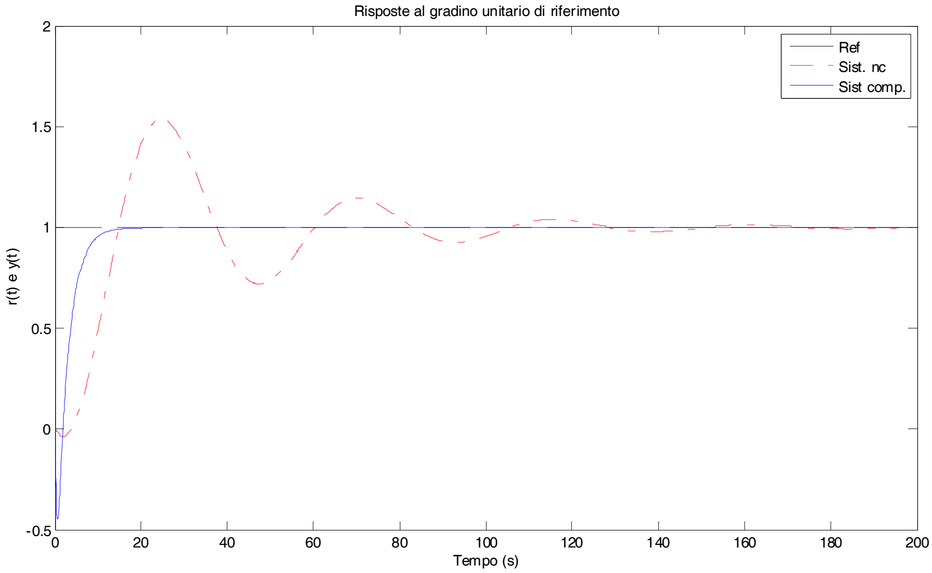
```
  100.0000  -99.0000

denR2z =

    1.0000   -0.0000

>>
```

Una volta assegnate tutte le variabili, si progetta il seguente schema Simulink, che comprende il sistema non compensato in retroazione unitaria (1), il sistema in retroazione con rete anticipatrice a tempo continuo (2), e i due schemi a tempo discreto, con funzione di trasferimento a tempo discreto e mantenitore di ordine zero, (3) e (4).

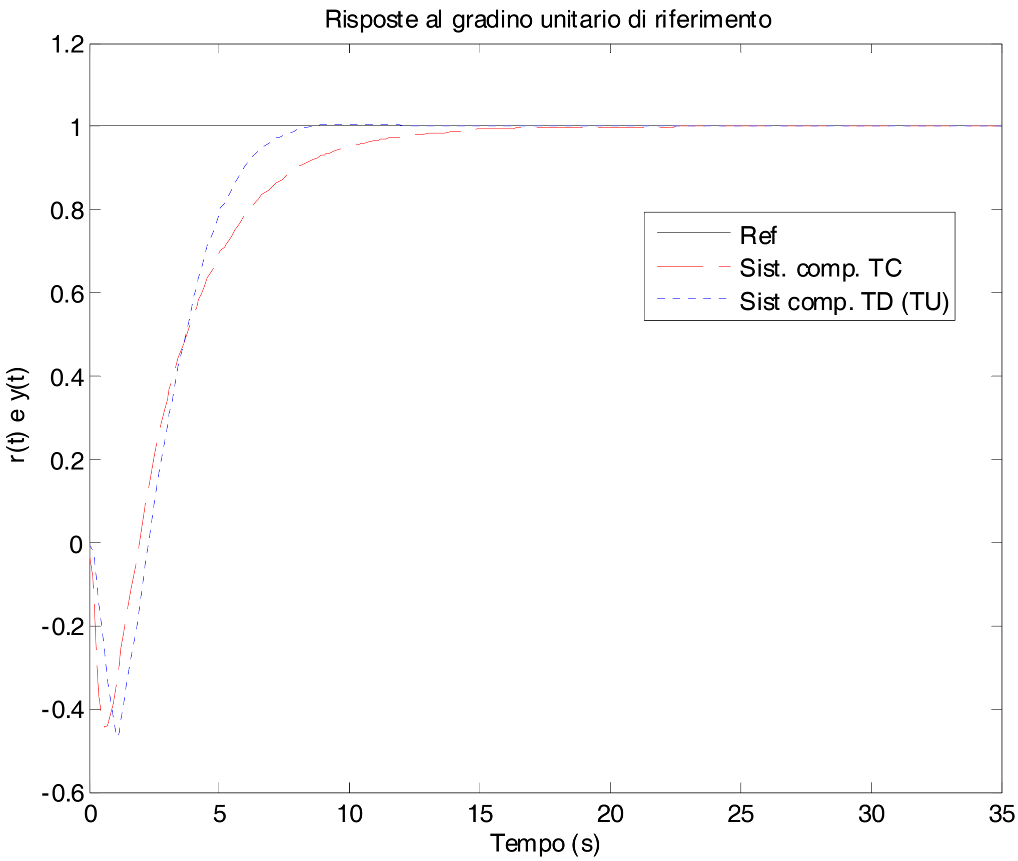
Si osservi come nei blocchi Simulink che implementano la funzione di trasferimento a tempo discreto, oltre ai coefficienti dei polinomi, occorre definire anche il tempo di campionamento. La stessa variabile deve essere anche definita nel blocco del dispositivo di tenuta di ordine zero.



Si osservi che lo schema proposto effettua inizialmente il confronto tra sistema non compensato e compensato, entrambi in retroazione unitaria negativa, e si hanno le seguenti risposte al gradino unitario di riferimento:



Successivamente si implementano e verificano gli schemi di regolatori a tempo discreto in cascata col mantenitore di ordine zero. Per quello che riguarda la risposta del sistema con il regolatore a tempo discreto ottenuto col metodo di Tustin, si verifica quanto segue:



Il sistema compensato dal regolatore a tempo discreto ottenuto col metodo di Tustin risulta stabile, come si può verificare anche andando a controllare i poli del sistema complessivo in retroazione:

$$G_{R1}(z) = \frac{R_1(z)G_{eq}(z)}{1+R_1(z)G_{eq}(z)}$$

dove $G_{eq}(z)$ è l'equivalente a tempo discreto della funzione di trasferimento $G(s)$ secondo il metodo dell'Hold Equivalence:

$$G_{eq}(z) = Z\left[\frac{1-e^{-sT}}{s}G(s)\right] = (1-z^{-1})Z\left[\frac{G(s)}{s}\right]$$

che si ottiene come segue:

```
>> Gz=c2d(Gs,T,'zoh')

Transfer function:
  -0.02645 z^2 + 0.04165 z + 0.003836
---------------------------------------
z^3 - 1.905 z^2 + 0.9049 z - 4.108e-005

Sampling time (seconds): 1
>>
>> Gr1=R1z*Gz/(1+R1z*Gz)

Transfer function:


-0.463 z^7 + 1.721 z^6 - 1.845 z^5 - 0.02551 z^4 + 0.9838 z^3

                       - 0.3352 z^2 - 0.03663 z + 1.663e-006

--------------------------------------------------------------
```

```
 z^8 - 2.939 z^7 + 2.524 z^6 + 0.2659 z^5 - 1.386 z^4

          + 0.5435 z^3 + 0.02874 z^2 - 0.03666 z + 1.664e-006


Sampling time (seconds): 1
>>

>> [numGr1,denGr1]=tfdata(Gr1,'v')

numGr1 =

  Columns 1 through 5

         0   -0.4630    1.7210   -1.8445   -0.0255

  Columns 6 through 9

    0.9838   -0.3352   -0.0366    0.0000


denGr1 =

  Columns 1 through 5

    1.0000   -2.9394    2.5242    0.2659   -1.3862

  Columns 6 through 9

    0.5435    0.0287   -0.0367    0.0000

>>

>> roots(denGr1)
```

```
ans =

  -0.6667
   1.0000
   0.9048
   0.9047
   0.5120 + 0.1818i
   0.5120 - 0.1818i
  -0.2275
   0.0000

>>
```

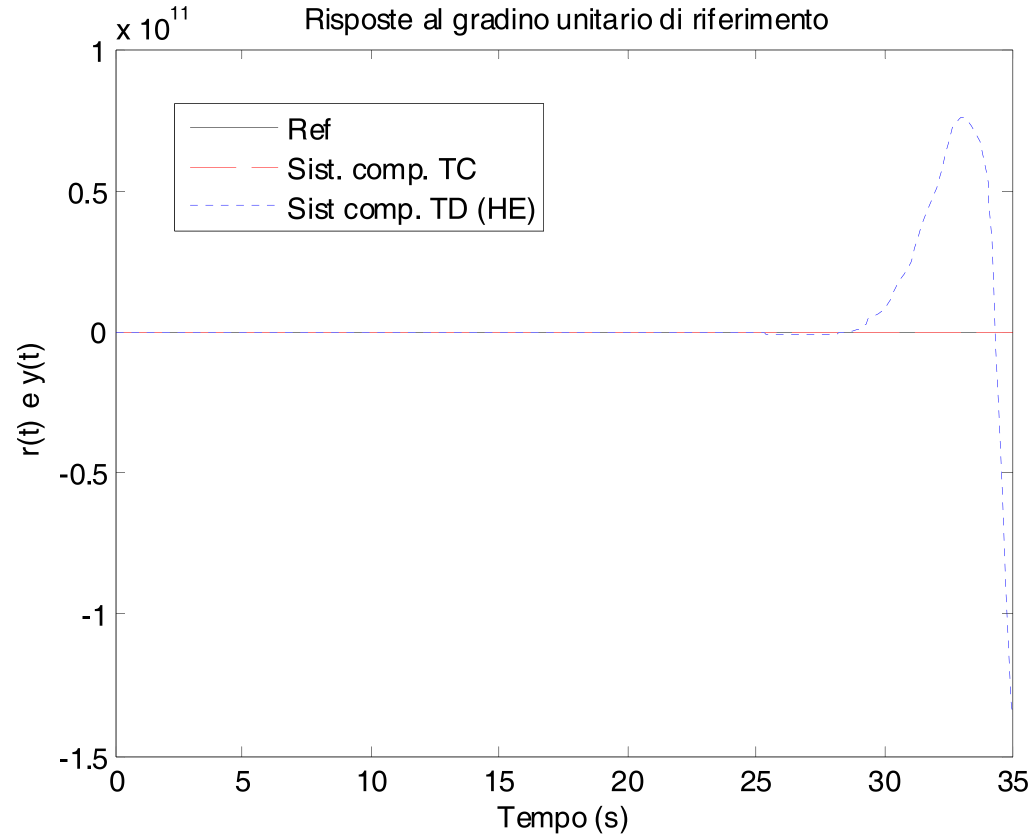
Si osserva quindi che tutte le radici del sistema in retroazione risultano all'interno del cerchio di raggio unitario ($|z<1|$), a parte una radice sul cerchio di raggio unitario ($z=1$, che garantisce un errore a regime nullo in risposta al gradino). Se si calcola inoltre il margine di fase del sistema in retroazione con il regolatore a tempo discreto ottenuto col metodo di Tustin, si verifica che:

```
>> Ga1z=R1z*Gz

Transfer function:
    -0.463 z^3 + 1.148 z^2 - 0.5923 z - 0.06074
----------------------------------------------------
z^4 - 1.238 z^3 - 0.365 z^2 + 0.6032 z - 2.739e-005

Sampling time (seconds): 1
>> [Gm1,Pm1]=margin(Ga1z)

Gm1 =

    2.0245



Pm1 =
```

```
   57.6624

>>
```

Ovvero il sistema complessivo ha un buon margine di stabilità, cioé un margine di fase di circa 57°.

Il sistema invece compensato dal regolatore ottenuto col metodo dell'Hold Equivalence (HE) produce questa risposta:



caratterizzante un sistema instabile. Infatti, il sistema complessivo in retroazione a tempo discreto in retroazione unitaria col regolatore ottenuto col metodo HE risulta:

```
>> Gr2=R2z*Gz/(1+R2z*Gz)

Transfer function:


-2.645 z^7 + 11.82 z^6 - 19.06 z^5 + 12.88 z^4 - 2.662 z^3
```

```
                       - 0.3434 z^2 + 3.12e-005 z - 7.083e-010

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

z^8 - 6.455 z^7 + 17.26 z^6 - 22.51 z^5 + 13.7 z^4

         - 2.662 z^3 - 0.3434 z^2 + 3.12e-005 z - 7.083e-010


Sampling time (seconds): 1
>>
```

e calcolando le radici del polinomio al denominatore di tale funzione, si ottiene:

```
>> [numGr2,denGr2]=tfdata(Gr2,'v')

numGr2 =

  Columns 1 through 6

         0   -2.6455   11.8235  -19.0571   12.8840   -2.6616

  Columns 7 through 9

   -0.3434    0.0000   -0.0000


denGr2 =

  Columns 1 through 6

    1.0000   -6.4554   17.2623  -22.5052   13.7034   -2.6617

  Columns 7 through 9
```

```
   -0.3434    0.0000   -0.0000

>>

>> roots(denGr2)

ans =

   1.8228 + 1.0737i
   1.8228 - 1.0737i
   1.0000
   0.9904
   0.9048
  -0.0857
   0.0000
   0.0000

>>
```

Ovvero due radici complesse coniugate risultano esterne al cerchio di raggio unitario, $|z>1|$.

Verificando inoltre il margine di fase, si ha la conferma che il sistema complessivo risulta instabile (anche la funzione Matlab `margin` lo segnala con un warning):

```
>> Ga2z=R2z*Gz

Transfer function:
       -2.645 z^3 + 6.784 z^2 - 3.74 z - 0.3798
-------------------------------------------------------
z^4 - 1.905 z^3 + 0.905 z^2 - 8.216e-005 z + 1.865e-009

Sampling time (seconds): 1
>>
```

```
>> [Pm2,Gm2]=margin(Ga2z)
Warning: The closed-loop system is unstable.
> In warning at 26
  In DynamicSystem.margin at 62

Gm2 =


    0.2105



Pm2 =


   Inf

>>
```

Si osserva infine che risulta possibile quindi la compensazione del sistema con il solo regolatore a tempo discreto ricavato col metodo di Tustin (TU).