

Esercitazione di laboratorio #4 - Controlli Automatici

Esercizio #2: Simulazione di un DC-motor comandato in armatura e controllato in posizione

Autori: M. Indri, M. Taragna (ultima modifica: 07/05/2020)

Contents

- [Introduzione](#)
- [Passo 0: definizione del sistema DC-motor comandato in armatura](#)
- [Passo 1: simulazione in catena aperta in assenza del disturbo Td](#)
- [Passo 2: simulazione in catena aperta in presenza del disturbo Td](#)
- [Passo 3: simulazione in catena chiusa in assenza del disturbo Td](#)
- [Passo 4: simulazione in catena chiusa in presenza del disturbo Td](#)
- [Passo 5: calcolo delle f.d.t. in catena chiusa e dei diagrammi di Bode](#)

Introduzione

Si puo' suddividere il programma in diverse sezioni di codice usando i caratteri "%%". Ogni sezione puo' essere eseguita separatamente dalle altre con il comando "Run Section" (nella toolbar dell'Editor, subito a destra del tasto "Run"). Si puo' ottenere lo stesso risultato selezionando la porzione di codice che si vuole eseguire e premendo il tasto funzione F9, risparmiando cosi' tempo rispetto all'esecuzione di tutto il programma. Si prenda questo script come esempio di riferimento.

```
clear all, close all, clc
```

Passo 0: definizione del sistema DC-motor comandato in armatura

```
% Parametri del motore elettrico
Ra=1; La=6e-3; Km=0.5; J=0.1; b=0.02; Ka=10;

s=tf('s');
F1=Ka*Km/((s*La+Ra)*(s*J+b)+Km^2)
F2=-((s*La+Ra)/((s*La+Ra)*(s*J+b)+Km^2))
```

F1 =

$$\frac{5}{0.0006 s^2 + 0.1001 s + 0.27}$$

Continuous-time transfer function.

F2 =

$$\frac{-0.006 s - 1}{0.0006 s^2 + 0.1001 s + 0.27}$$

Continuous-time transfer function.

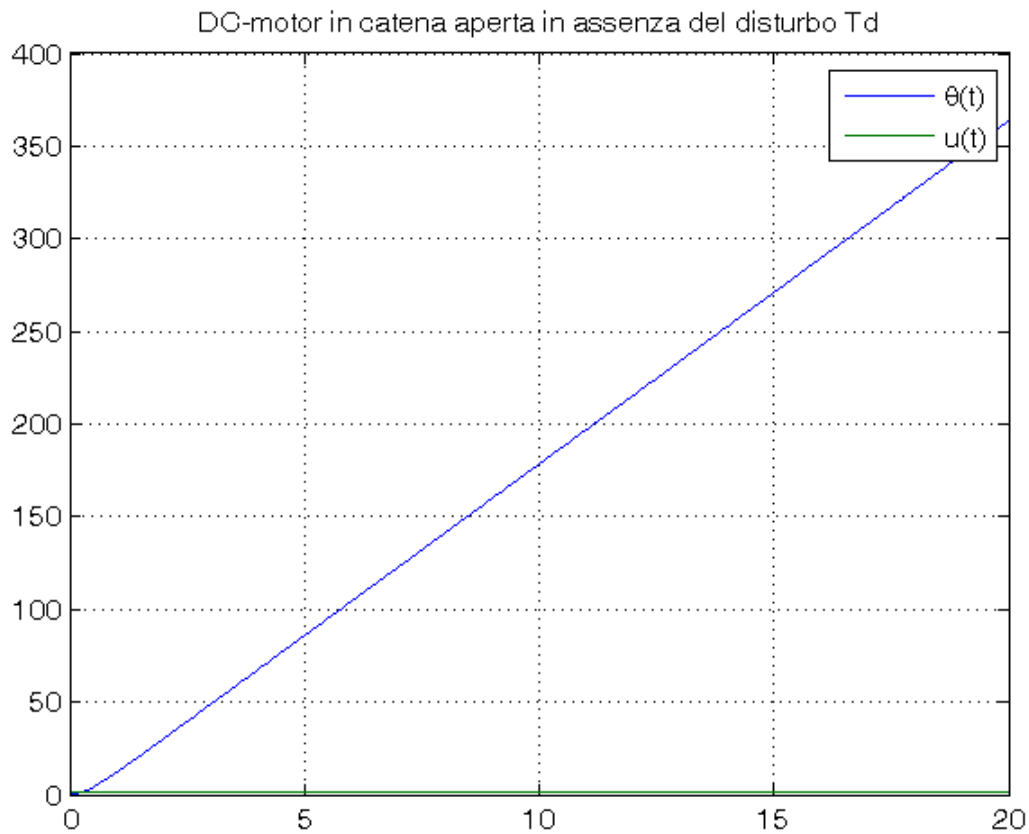
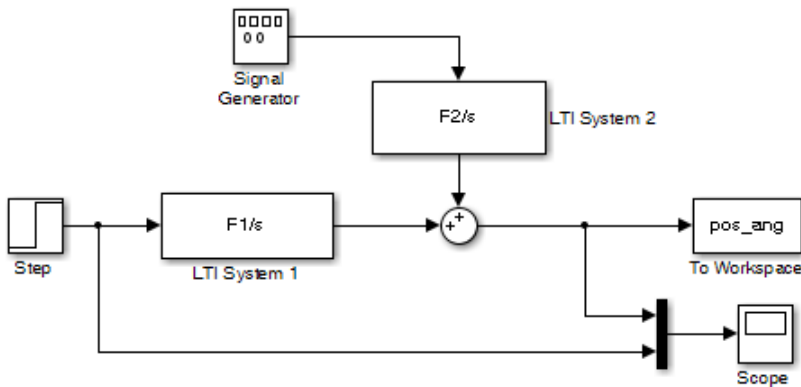
Passo 1: simulazione in catena aperta in assenza del disturbo Td

```
Td_amp=0

open_system('es_motore_no_controllo_posiz')
sim('es_motore_no_controllo_posiz')
pos_rif=ones(size(tout));
figure, plot(tout,pos_ang, tout,pos_rif), grid on,
```

```
title('DC-motor in catena aperta in assenza del disturbo Td'),
legend('\theta(t)', 'u(t)')
```

```
Td_amp =
0
```



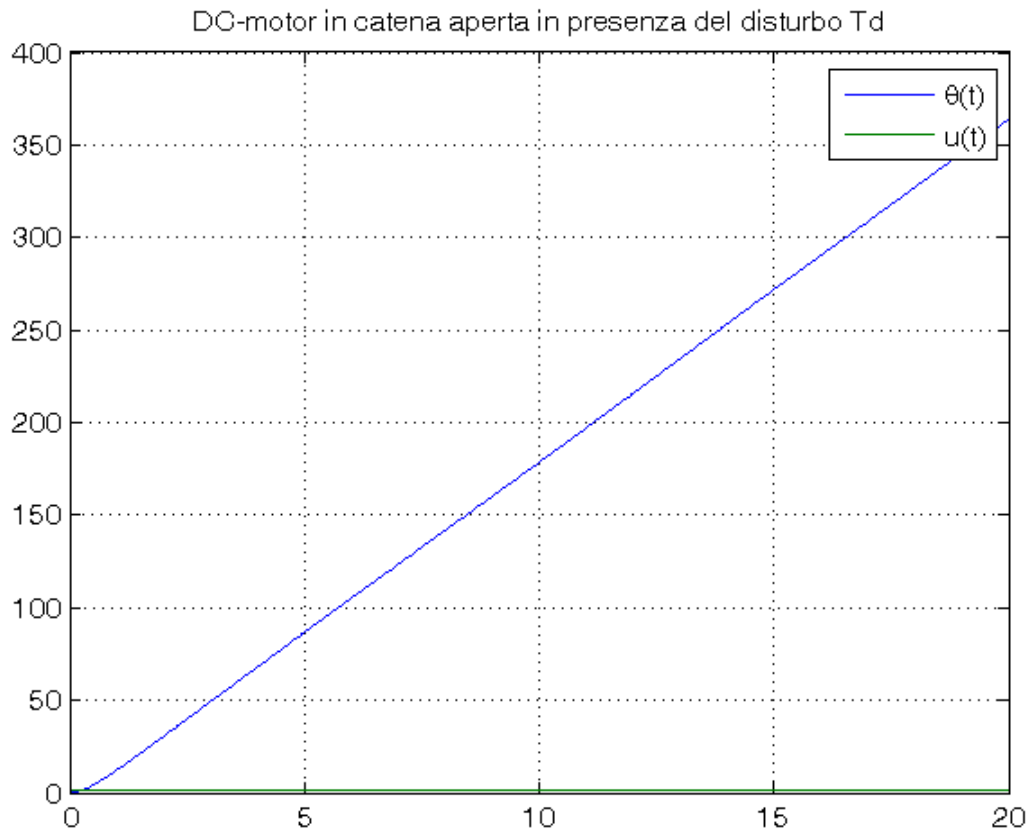
Passo 2: simulazione in catena aperta in presenza del disturbo Td

```
Td_amp=0.05

sim('es_motore_no_controllo_posiz')
pos_rif=ones(size(tout));
figure, plot(tout,pos_ang, tout,pos_rif), grid on,
title('DC-motor in catena aperta in presenza del disturbo Td'),
legend('\theta(t)', 'u(t)')
close_system('es_motore_no_controllo_posiz')
```

```
Td_amp =
```

0.0500

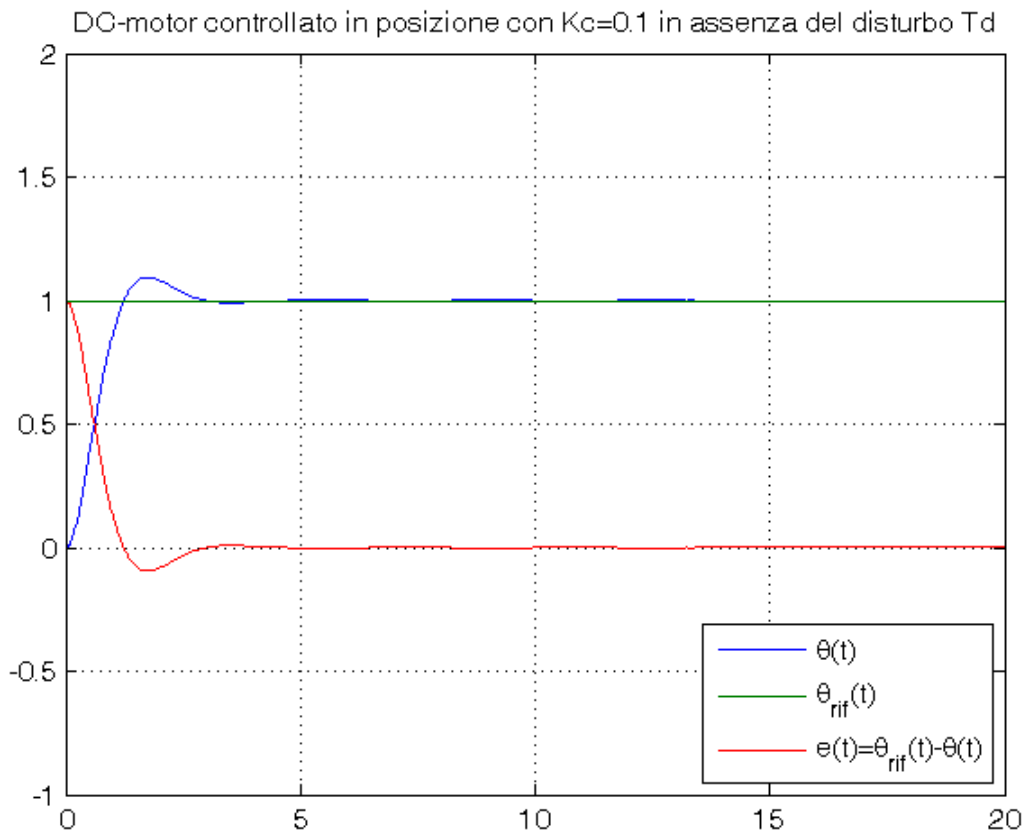
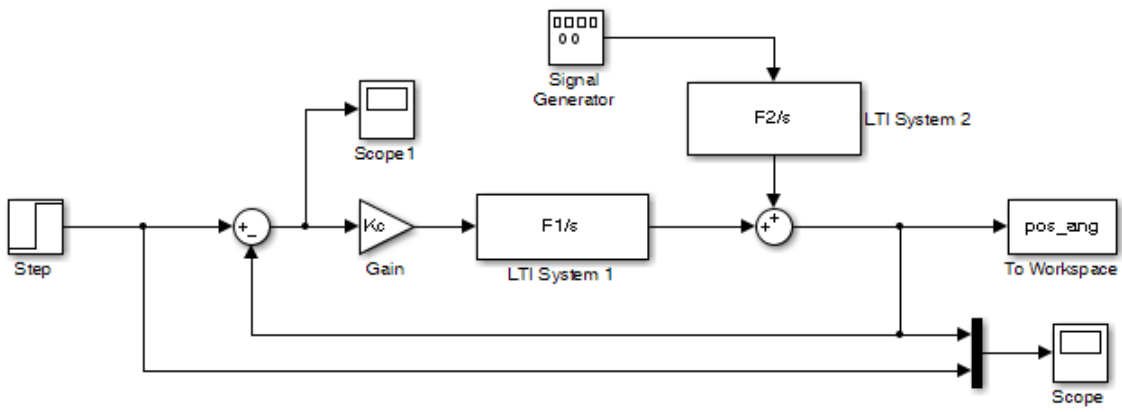


Passo 3: simulazione in catena chiusa in assenza del disturbo T_d

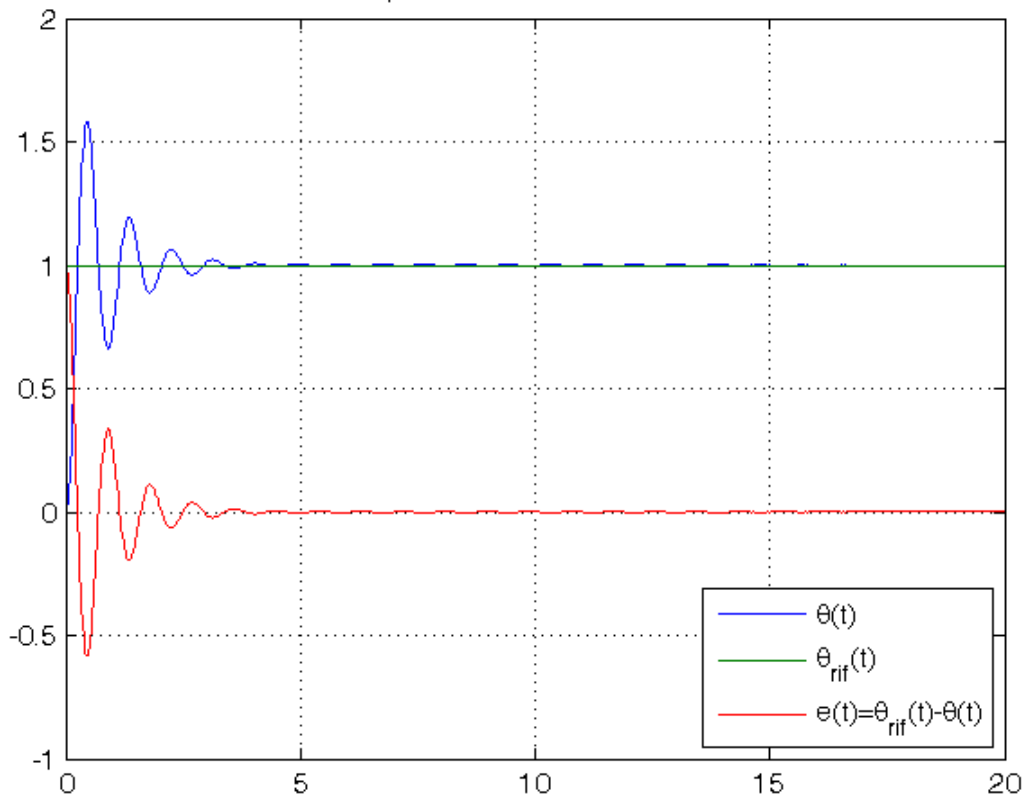
```
Td_amp=0
Kc_vec=[0.1, 1, 5];

open_system('es_motore_con_controllo_posiz')
for Kc=Kc_vec,
    sim('es_motore_con_controllo_posiz')
    pos_rif=ones(size(tout));
    errore=pos_rif-pos_ang;
    figure, plot(tout,pos_ang, tout,pos_rif, tout,errore), grid on, ylim([-1,2]),
    title(['DC-motor controllato in posizione con Kc=', num2str(Kc), ...
        ' in assenza del disturbo Td']),
    legend('\theta(t)', '\theta_{rif}(t)', 'e(t)=\theta_{rif}(t)-\theta(t)',4)
end
```

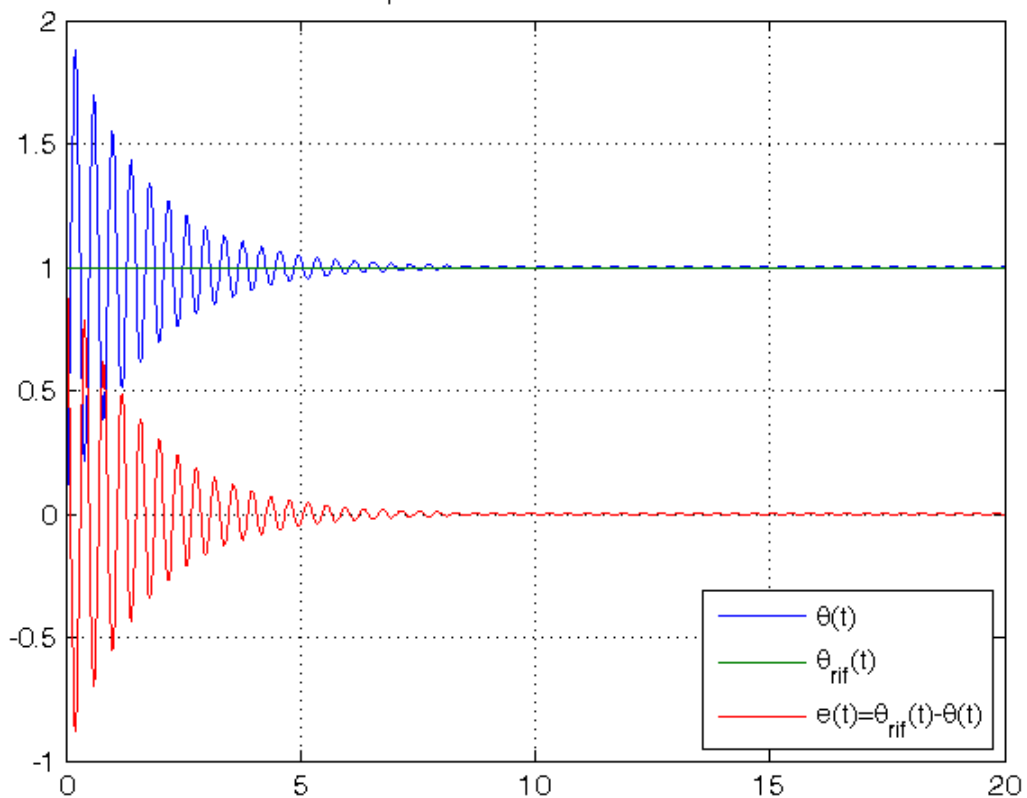
```
Td_amp =
    0
```



DC-motor controllato in posizione con $K_c=1$ in assenza del disturbo T_d



DC-motor controllato in posizione con $K_c=5$ in assenza del disturbo T_d



Passo 4: simulazione in catena chiusa in presenza del disturbo T_d

```
Td_amp=0.05  
  
for Kc=Kc_vec,  
    sim('es_motore_con_controllo_posiz')
```

```

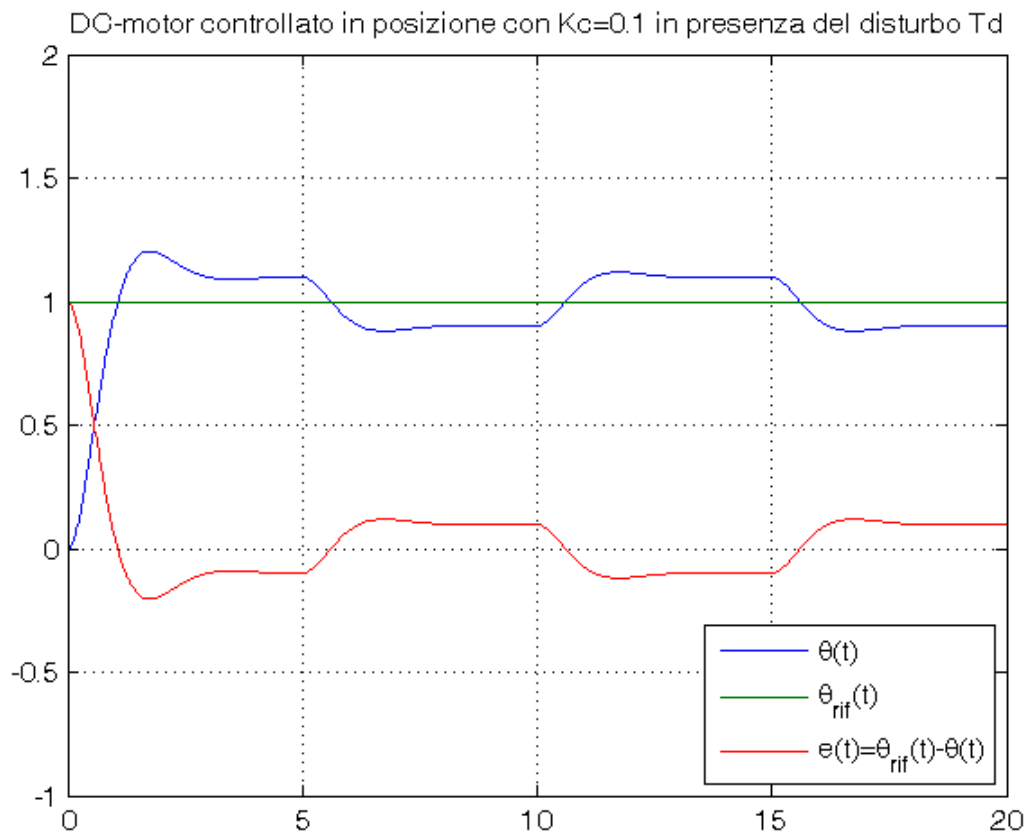
pos_rif=ones(size(tout));
errore=pos_rif-pos_ang;
figure, plot(tout,pos_ang, tout,pos_rif, tout,errore), grid on, ylim([-1,2]),
title(['DC-motor controllato in posizione con Kc=', num2str(Kc), ...
' in presenza del disturbo Td']),
legend('\theta(t)', '\theta_{rif}(t)', 'e(t)=\theta_{rif}(t)-\theta(t)',4)
end
close_system('es_motore_con_controllo_posiz')

```

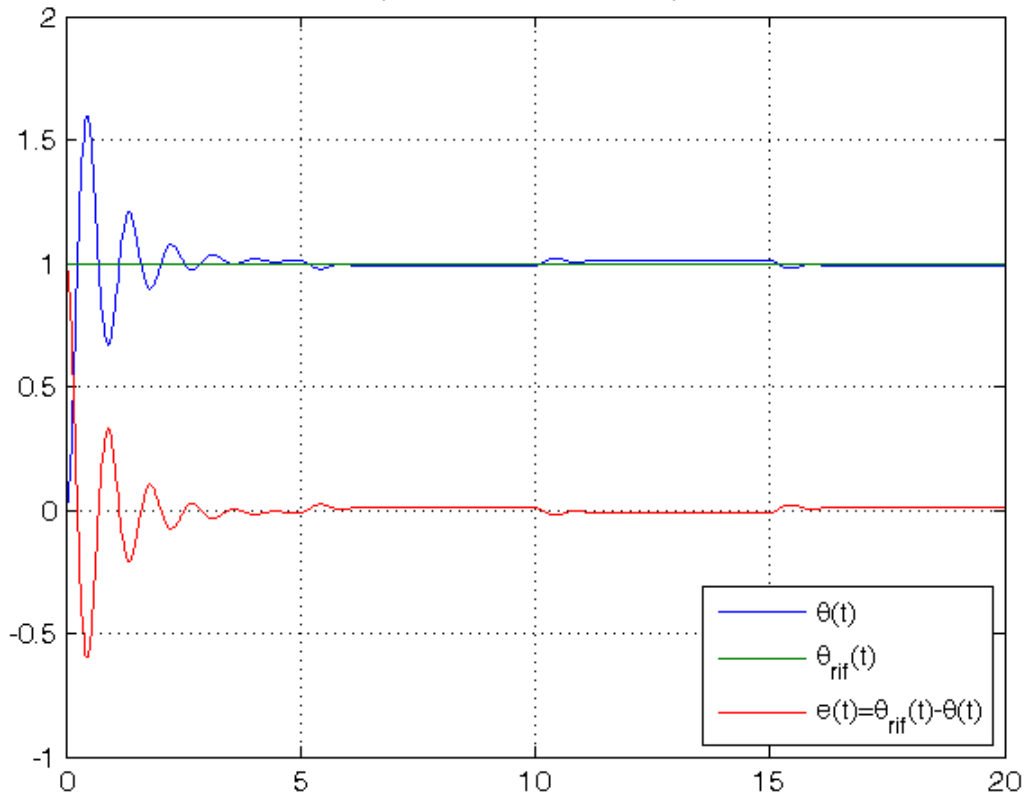
```

Td_amp =
    0.0500

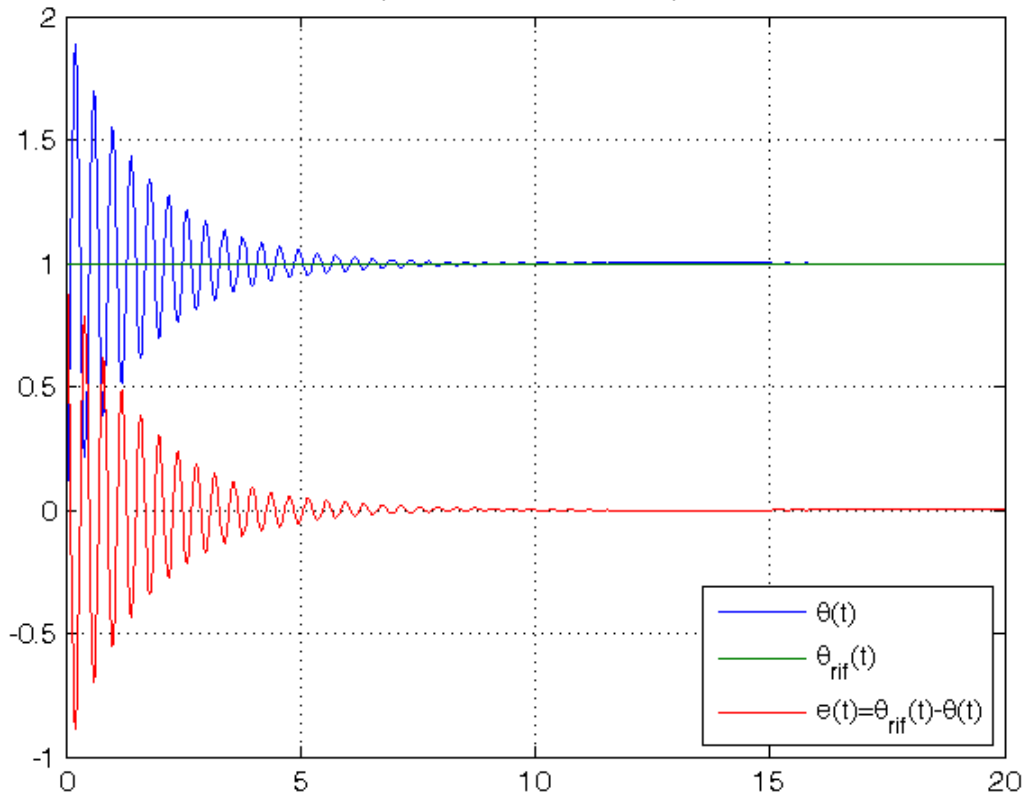
```



DC-motor controllato in posizione con $K_c=1$ in presenza del disturbo T_d



DC-motor controllato in posizione con $K_c=5$ in presenza del disturbo T_d



Passo 5: calcolo delle f.d.t. in catena chiusa e dei diagrammi di Bode

```
Kc_max=(b*La+Ra*J) * (Ra*b+Km^2) / (J*La*Km*Ka)
```

```
figure
for Kc=Kc_vec,
```

```

Kc
W=feedback(Kc*F1/s,1)
z_W=zero(W)
p_W=pole(W)
damp(W)
bode(W), grid on, xlim([1e-1, 1e4]), hold on,
title('DC-motor controllato in posizione')
end
legend(['Kc=',num2str(Kc_vec(1))], ['Kc=',num2str(Kc_vec(2))], ['Kc=',num2str(Kc_vec(3))])

```

```

Kc_max =
    9.0108
Kc =
    0.1000

```

```

W =
-----
                0.5
-----
0.0006 s^3 + 0.1001 s^2 + 0.27 s + 0.5

```

Continuous-time transfer function.

```

z_W =
Empty matrix: 0-by-1

```

```

p_W =
    1.0e+02 *
-1.6416 + 0.0000i
-0.0136 + 0.0180i
-0.0136 - 0.0180i

```

Pole	Damping	Frequency (rad/seconds)	Time Constant (seconds)
-1.36e+00 + 1.80e+00i	6.01e-01	2.25e+00	7.38e-01
-1.36e+00 - 1.80e+00i	6.01e-01	2.25e+00	7.38e-01
-1.64e+02	1.00e+00	1.64e+02	6.09e-03

```

Kc =
    1

```

```

W =
-----
                5
-----
0.0006 s^3 + 0.1001 s^2 + 0.27 s + 5

```

Continuous-time transfer function.

```

z_W =
Empty matrix: 0-by-1

```

```

p_W =
    1.0e+02 *
-1.6444 + 0.0000i
-0.0121 + 0.0701i
-0.0121 - 0.0701i

```

Pole	Damping	Frequency (rad/seconds)	Time Constant (seconds)
-1.21e+00 + 7.01e+00i	1.71e-01	7.12e+00	8.24e-01
-1.21e+00 - 7.01e+00i	1.71e-01	7.12e+00	8.24e-01
-1.64e+02	1.00e+00	1.64e+02	6.08e-03

```

Kc =
    5

```

```

W =

```


$$0.0006 s^3 + 0.1001 s^2 + 0.27 s + 25$$

Continuous-time transfer function.

z_W =

Empty matrix: 0-by-1

p_W =

```
1.0e+02 *
-1.6567 + 0.0000i
-0.0060 + 0.1585i
-0.0060 - 0.1585i
```

Pole	Damping	Frequency (rad/seconds)	Time Constant (seconds)
$-5.99e-01 + 1.58e+01i$	$3.78e-02$	$1.59e+01$	$1.67e+00$
$-5.99e-01 - 1.58e+01i$	$3.78e-02$	$1.59e+01$	$1.67e+00$
$-1.66e+02$	$1.00e+00$	$1.66e+02$	$6.04e-03$

