

Esercitazione di Controlli Automatici 1 – n°1

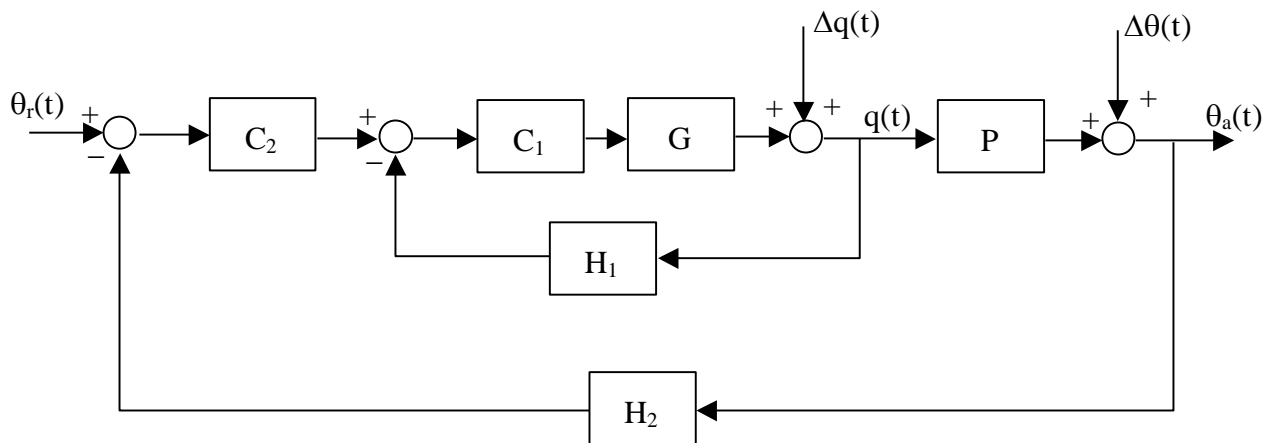
Un sistema di controllo della temperatura in un locale di piccole dimensioni è realizzato mediante un controllo in cascata in cui il loop interno controlla la portata di acqua calda/fredda circolante nella unità di condizionamento (vedi figura).

La dinamica dell'ambiente da condizionare, considerando come ingresso la portata di acqua a temperatura costante, può essere rappresentata dalla funzione di trasferimento $P(s)$ con due costanti di tempo; mentre il legame tra portata e segnale di comando alla valvola può essere schematizzato con un semplice sistema del primo ordine stabile $G(s)$.

Il controllo della portata è realizzato con un semplice controllore proporzionale $C_1(s)$, mentre il controllore esterno di temperatura $C_2(s)$, è un controllore industriale di tipo PI.

La misura della temperatura ambiente viene effettuata mediante una termocoppia $H_2(s)$, la cui costante di tempo non è, a priori, trascurabile.

Sul sistema possono agire dei disturbi sulla portata $\Delta q(t)$, legati a variazioni di pressione nelle tubazioni, e sulla temperatura ambiente $\Delta\theta(t)$, legati alle condizioni ambientali esterne.



$$P(s) = \frac{k_p}{(1 + \tau_{p1}s)(1 + \tau_{p2}s)} = \frac{0.1}{1000s^2 + 70s + 1}$$

$$G(s) = \frac{k_G}{1 + \tau_G s} = \frac{1}{1 + 10s}$$

$$C_1(s) = k_{c1} = 100$$

$$C_2(s) = k_{c2} \frac{(1 + \tau_I s)}{s} = k_{c2} \frac{1 + 100s}{s}$$

$$H_1(s) = 1$$

$$H_2(s) = \frac{k_H}{1 + \tau_H s} = \frac{1}{1 + s}$$

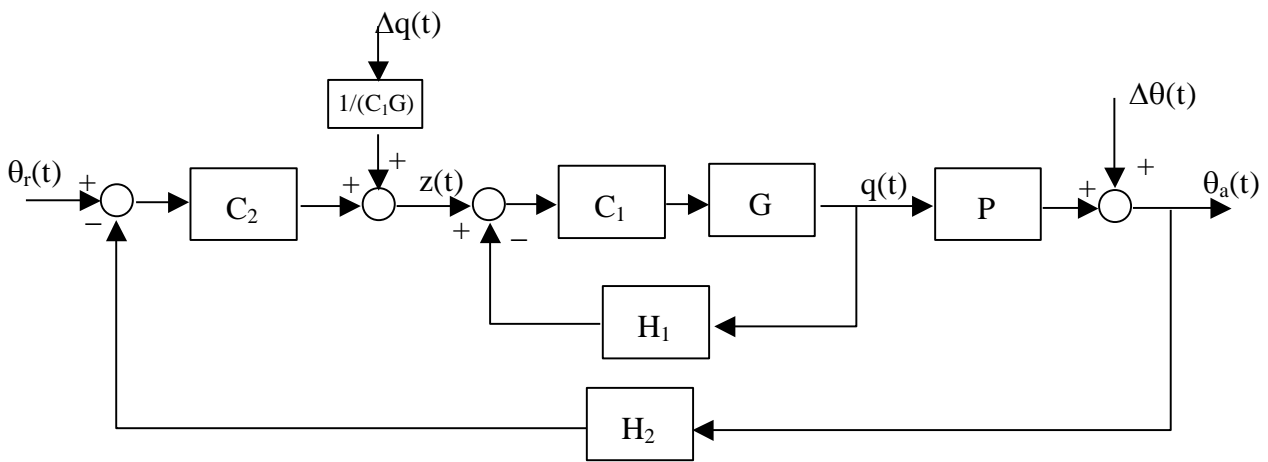
1. Valutare le funzioni di trasferimento ingresso-uscita e disturbi-uscita.
2. Verificare se sia possibile dare, nel caso di $k_{c2}=1$, una rappresentazione semplificata del legame ingresso uscita mediante un sistema del secondo ordine.

Soluzione Esercitazione n° 1

Il sistema oggetto dell'esercitazione è un classico esempio di sistema di controllo "in cascata" (*nested feedback control*) dove un controllore a monte fornisce il segnale di riferimento per un controllore a valle, inserito in un loop interno, sulla base di un segnale errore relativo alla variabile di uscita da controllare. In questo caso la variabile di uscita è la temperatura ambiente $\theta_a(t)$ [K], il controllore C_1 comanda una valvola che modifica la variabile manipolabile del processo $q(t)$ [m³/s], ovvero una portata.

- 1) Le funzioni di trasferimento ingresso-uscita e disturbi-uscita possono essere valutate mediante le regole di manipolazione dei blocchi.

Manipolazione del diagramma a blocchi



Funzione di trasferimento dell' anello interno:

$$W_1(s) = \frac{Q(s)}{Z(s)} = \frac{C_1 G}{1 + C_1 G H_1} = \frac{C_1 G}{1 + C_1 G}$$

Funzione di trasferimento ingresso-uscita:

$$W_r(s) = \frac{\Theta_a(s)}{\Theta_r(s)} = \frac{C_2 W_1 P}{1 + C_2 W_1 P H_2} = \frac{C_2 \frac{C_1 G}{1 + C_1 G} P}{1 + C_2 \frac{C_1 G}{1 + C_1 G} P H_2} = \frac{C_2 C_1 G P}{\Delta}$$

Dove $\Delta(s) = 1 + C_1(s)G(s) + C_2(s)C_1(s)G(s)P(s)H_2(s)$ è il polinomio caratteristico del sistema, che risulta lo stesso per ogni f.d.t. sia ingresso/uscita che disturbo/uscita.

Funzione di trasferimento disturbo-uscita (disturbo di temperatura):

$$W_{\Delta\theta}(s) = \frac{\Theta_a(s)}{\Delta\Theta(s)} = \frac{1}{1 + C_2 W_1 P H_2} = \frac{1}{1 + C_2 \frac{C_1 G}{1 + C_1 G} P H_2} = \frac{1 + C_1 G}{\Delta}$$

Funzione di trasferimento disturbo-uscita (disturbo di portata):

$$W_{\Delta q}(s) = \frac{\Theta_a(s)}{\Delta Q(s)} = \frac{1}{C_1 G} \cdot \frac{W_1 P}{1 + C_2 W_1 P H_2} = \frac{\frac{1}{C_1 G} \cdot \frac{C_1 G}{1 + C_1 G} P}{1 + C_2 \frac{C_1 G}{1 + C_1 G} P H_2} = \frac{P}{\Delta}$$

2) La dinamica del loop interno (Inner Loop) può essere trascurata in quanto caratterizzata da un guadagno praticamente unitario e da una costante di tempo trascurabile rispetto a quella degli altri blocchi.

$$W_1(s) = \frac{Q(s)}{Z(s)} = \frac{C_1 G}{1 + C_1 G H_1} = \frac{100}{10s + 101} = \frac{100}{101 \left(1 + \frac{s}{10.1} \right)} = \frac{0.990099}{1 + \frac{s}{10.1}} \cong 1$$

Comandi MatLab¹ per la creazione e l'analisi della f.d.t. $W_1(s)$:

```
% Loop interno di Portata
% P(s) f.d.t. Ambiente da condizionare
kp=0.1;
P=tf(kp,[1000 70 1]);
% G(s) f.d.t. Valvola per il controllo della Portata
G=tf(1,[10 1]);
% C1(s) f.d.t. Regolatore Loop di Portata
C1=tf(100);
% H1(s) f.d.t. Sensore di Portata
H1=1;
% W1(s) f.d.t. Loop di Portata
W1=feedback(C1*G,H1);

[numW1,denW1]=tfdata(W1,'v');
zeriW1=roots(numW1)
poliW1=roots(denW1)
```

Funzione di trasferimento ingresso-uscita

$$W_r(s) = \frac{\Theta_a(s)}{\Theta_r(s)} = \frac{C_2 W_1 P}{1 + C_2 W_1 P H_2} \cong \frac{C_2 P}{1 + C_2 P H_2}$$

La $W_r(s)$ risultante tenendo conto della approssimazione effettuata riguardo il loop interno ($W_1(s) \approx 1$) è:

$$W_r(s) = \frac{\Theta_a(s)}{\Theta_r(s)} \cong \frac{C_2 P}{1 + C_2 P H_2} = \frac{G}{1 + F} = \frac{k \cdot (s+1) \cdot (100+s)}{k \cdot (100s+1) + 10s \cdot (s+1) \cdot (1000s^2 + 70s + 1)}$$

Successivamente, assegnando al controllore del loop esterno guadagno unitario, otteniamo

$$W_r(s) \Big|_{k=1} = \frac{(s+1)(100s+1)}{10000s^4 + 10700s^3 + 710s^2 + 110s + 1}$$

¹ In Appendice 2 è riportata una breve descrizione dei comandi che vedremo in questa esercitazione. Per maggiori dettagli relativi ai comandi MatLab digitare nella “Command Window” il comando “help” o il comando “doc” seguito dal nome del comando di interesse. Per esempio: “help bode” oppure “doc step”.

Comandi MatLab per la creazione e l'analisi della f.d.t. $W_r(s)$:

```
% C2(s) f.d.t. Regolatore Loop di Temperatura
kc2=1;
C2=tf(kc2*[100 1],[1 0]);
% H1(s) f.d.t. Sensore di Temperatura
H2=tf(1,[1 1]);
% Wr(s) f.d.t. Loop di Temperatura
Wr=feedback(C2*W1*P,H2);
% Wr(s) nell'ipotesi di considerare istantaneo il loop di portata
Wr_approx1=feedback(C2*P,H2);
% Confronto diagrammi di bode
figure
bode(W1,Wr,Wr_approx1), grid
legend('W1(s)', 'Wr(s)', 'Wr_{approx1}(s)');
```

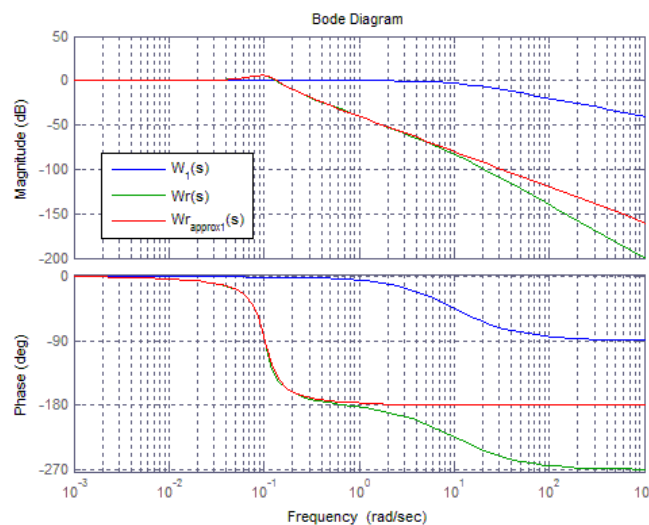


Figure 1. Risposte armoniche delle f.d.t. $W_1(s)$, $W_r(s)$ e $W_{r_approx1}(s)$.

Osservando la curva blu nel diagramma di bode in Figura 1 possiamo vedere come l'approssimazione in $W_1(s) \approx 1$ sia più che ragionevole per un range di frequenze sino ad 1 rad/sec. Per frequenze superiori invece un'attenuazione in dovuta all'effetto del polo in 10.1 inizia a mostrare i suoi effetti. Tali effetti sono ovviamente facilmente riscontrabili anche nel confronto tra la f.d.t. esatta del processo $W_r(s)$ e la sua approssimazione $W_{r_approx1}(s)$.

Zeri e poli della f.d.t. a ciclo chiuso $W_{r_approx1}(s)$, possono valutarsi utilizzando MatLab attraverso il comando `roots()` come segue:

```
[numWr_approx1,denWr_approx1]=tfdata(Wr_approx1,'v');
zeriWr_approx1=roots(numWr_approx1)
poliWr_approx1=roots(denWr_approx1)
```

$$z_1 = -1, \quad z_2 = -0.01$$

$$p_{1,2} = -0.025 \pm 0.098i; \quad p_3 = -0.0096; \quad p_4 = -1.0104$$

Osserviamo che le coppie z_1, p_4 e z_2, p_3 danno luogo a due “cancellazioni” polo-zero approssimate (*cancellazioni imperfette*); pertanto il sistema di controllo può essere ulteriormente approssimato da una funzione di trasferimento con guadagno unitario e due soli poli complessi e coniugati

$$W_r'(s) = \frac{1}{\left(1 - \frac{s}{-0.025 - 0.098i}\right) \left(1 - \frac{s}{-0.025 + 0.098i}\right)} = \frac{1}{97s^2 + 4.85s + 1}$$

A conferma delle approssimazioni eseguite, di seguito è riportato il codice MatLab e le figure relative al diagramma di bode ed alle risposte indicali delle $W_r(s)$ (f.d.t. esatta), $W_{r \text{ approx1}}(s)$ (f.d.t. ottenuta trascurando la dinamica del loop interno) e $W_{r \text{ approx2}}(s)$ (f.d.t. che tiene conto delle cancellazioni polo-zero approssimate).

```
% W_r(s) che tiene conto delle cancellazione polo-zero e di W_l(s)=1
denWr_approx2=conv([-1/(-0.025-0.098*j) 1],[-1/(-0.025+0.098*j) 1])
Wr_approx2=tf(1,denWr_approx2)

% Confronto diagrammi di bode
figure
bode(Wr,Wr_approx1,Wr_approx2), grid
legend('Wr(s)', 'Wr_{approx1}(s)', 'Wr_{approx2}(s)');
% Confronto risposte indicali
figure
step(Wr,Wr_approx1,Wr_approx2), grid
legend('Wr(s)', 'Wr_{approx1}(s)', 'Wr_{approx2}(s)');
axis([0 250 0 1.5])
```

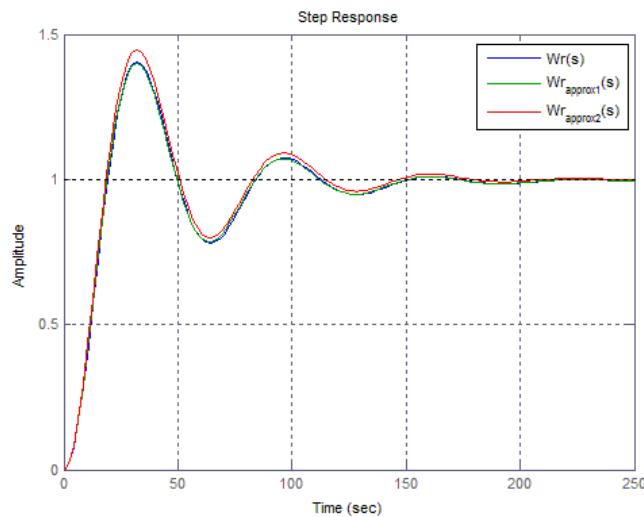


Figura 2. Confronto tra le risposte indicali delle f.d.t. $W_r(s)$, $W_{r \text{ approx1}}(s)$ e $W_{r \text{ approx2}}(s)$.

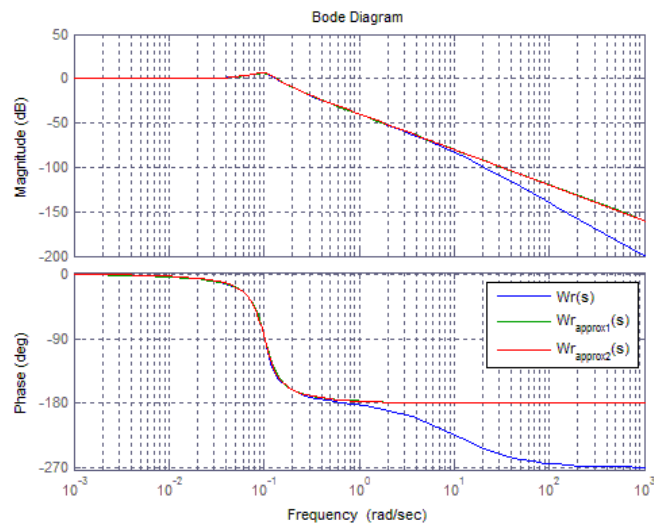


Figura 3. Confronto tra le risposte armoniche delle f.d.t. $W_r(s)$, $W_{r \text{ approx1}}(s)$ e $W_{r \text{ approx2}}(s)$.

Osservando l'andamento delle risposte indiciali in Figura 2 è immediato notare come le due approssimazioni rispettivamente sul loop interno e sulle cancellazioni garantiscono una buona corrispondenza rispetto all'andamento esatto di $W_r(s)$. Anche il confronto tra le tre risposte armoniche mostrato in Figura 3, conferma la buona corrispondenza in bassa frequenza tra sistema originale ed approssimato, mentre in alta frequenza tale corrispondenza va a ridursi, sia in modulo che in fase poiché in tali condizioni di lavoro il loop interno $W_1(s)$ non può più essere considerato istantaneo.

Per quanto riguarda invece le f.d.t. disturbo-uscita nell'ipotesi di considerare $W_1(s) \approx 1$ abbiamo:

$$W_{\Delta\theta}(s) = \frac{\Theta_a(s)}{\Delta\Theta(s)} = \frac{1}{1 + C_2 W_1 P H_2} \cong \frac{1}{1 + C_2 P H_2}$$

$$W_{\Delta q}(s) = \frac{\Theta_a(s)}{\Delta Q(s)} = \frac{1}{C_1 G} \cdot \frac{W_1 P}{1 + C_2 W_1 P H_2} \cong \frac{\frac{1}{C_1 G} \cdot P}{1 + C_2 P H_2}$$

Comandi MatLab per la creazione e l'analisi della f.d.t. disturbo/uscita

```
%% Analisi f.d.t. Disturbo/Uscita
Wdt=feedback(1,C2*W1*P*H2);
Wdt_approx1=feedback(1,C2*P*H2);

Wdq=1/(C1*G)*feedback(W1*P,C2*P*H2);
Wdq_approx1=1/(C1*G)*feedback(W1*P,C2*P*H2);

figure
subplot(1,2,1)
    step(Wdt,Wdt_approx1), grid
    legend('W \Delta \theta(s)', 'W \Delta \theta_{approx1}(s)')
subplot(1,2,2)
    step(Wdq,Wdq_approx1), grid
    legend('W \Delta q(s)', 'W \Delta q_{approx1}(s)')
figure
subplot(1,2,1)
    bode(Wdt,Wdt_approx1), grid
    legend('W \Delta \theta(s)', 'W \Delta \theta_{approx1}(s)')
subplot(1,2,2)
    bode(Wdq,Wdq_approx1), grid
    legend('W \Delta q(s)', 'W \Delta q_{approx1}(s)')
```

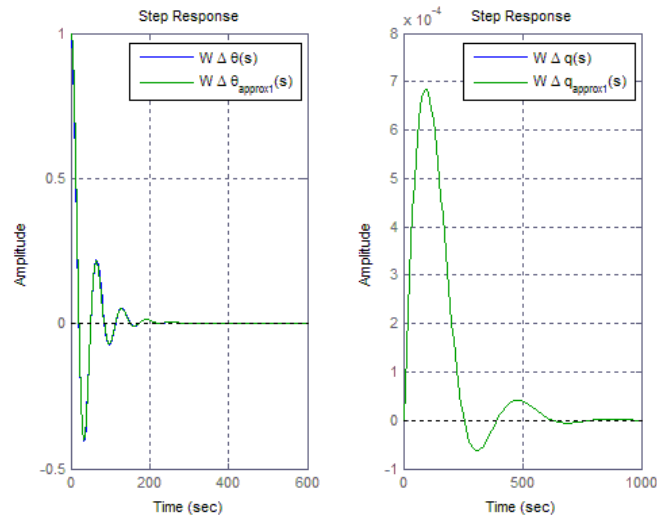


Figure 4. Confronto tra le risposte indicali delle f.d.t. disturbo/uscita.

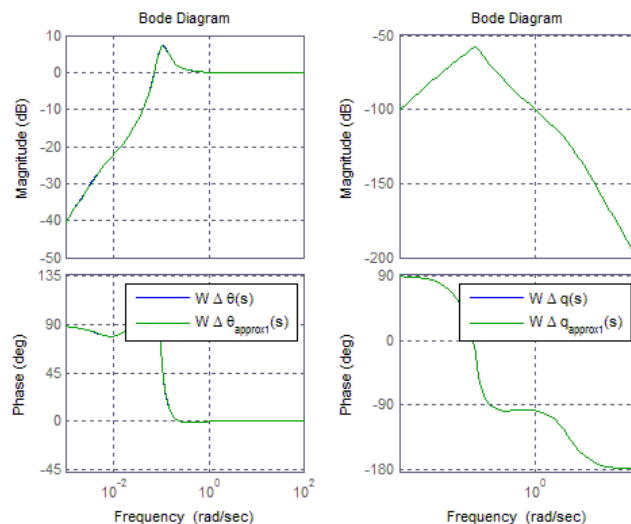


Figure 5. Confronto tra le risposte armoniche delle f.d.t. disturbo/uscita.

Per quanto riguarda Figura 4 e Figura 5 innanzitutto si può notare che il considerare $W_1(s) \approx 1$ garantisce in entrambi i casi una buona corrispondenza tra le f.d.t. esatte ed approssimate. Inoltre è importante sottolineare come entrambi il loop di controllo di temperatura sia in grado di reiettare completamente gli effetti di disturbi di tipo a gradino di temperatura $\Delta\theta = \delta_1(t)$ o di portata $\Delta q = \delta_1(t)$ grazie alla presenza di un polo nell'origine nel regolatore $C_2(s)$, il quale, essendo a monte di entrambi i disturbi è in grado di reiettarli completamente, nell'ipotesi che questi siano disturbi di tipo a gradino (Vedi Sistemi di Controllo di TIPO 1).

Per completezza di seguito è riportato l'intero codice relativo all'esercitazione appena discussa. Copiare ed incollare il seguente codice in un MatLab m-file per eseguire in blocco tutte le istruzioni. Vedi Appendice 1 per la creazione di un m-file.

```
% Esercitazione 1 Controlli Automatici

%% Analisi Loop interno di Portata
% P(s) f.d.t. Ambiente da condizionare
kp=0.1;
P=tf(kp,[1000 70 1]);
% G(s) f.d.t. Valvola per il controllo della Portata
```

```

G=tf(1,[10 1]);
% C1(s) f.d.t. Regolatore Loop di Portata
C1=tf(100);
% H1(s) f.d.t. Sensore di Portata
H1=1;
% W1(s) f.d.t. Loop di Portata
W1=feedback(C1*G,H1);

[numW1,denW1]=tfdata(W1,'v');
zeriW1=roots(numW1)
poliW1=roots(denW1)

%% Analisi Loop esterno di Temperatura
% C2(s) f.d.t. Regolatore Loop di Temperatura
kc2=1;
C2=tf(kc2*[100 1],[1 0]);
% H1(s) f.d.t. Sensore di Temperatura
H2=tf(1,[1 1]);
% Wr(s) f.d.t. Loop di Temperatura

Wr=feedback(C2*W1*P,H2);
% Wr(s) nell'ipotesi di considerare istantaneo il loop di portata
Wr_approx1=feedback(C2*P,H2);
% Confronto diagrammi di bode
figure(1)
    bode(Wr,Wr_approx1), grid
    legend('Wr(s)', 'Wr_{approx1}(s)');

[numWr_approx1,denWr_approx1]=tfdata(Wr_approx1,'v');

zeriWr_approx1=roots(numWr_approx1) '
poliWr_approx1=roots(denWr_approx1) '

% Wr(s) che tiene conto delle cancellazione polo-zero e di W1(s)=1
denWr_approx2=conv([-1/(-0.025-0.098*j) 1],[-1/(-0.025+0.098*j) 1])
Wr_approx2=tf(1,denWr_approx2)

% Confronto diagrammi di bode
figure(2)
bode(Wr,Wr_approx1,Wr_approx2), grid
legend('Wr(s)', 'Wr_{approx1}(s)', 'Wr_{approx2}(s)');
% Confronto risposte indiciali
figure(3)
    step(Wr,Wr_approx1,Wr_approx2), grid
    legend('Wr(s)', 'Wr_{approx1}(s)', 'Wr_{approx2}(s)');
    axis([0 250 0 1.5])

%% Analisi f.d.t. Disturbo/Uscita
Wdt=feedback(1,C2*W1*P*H2);
Wdt_approx1=feedback(1,C2*P*H2);

Wdq=1/(C1*G)*feedback(W1*P,C2*P*H2);
Wdq_approx1=1/(C1*G)*feedback(W1*P,C2*P*H2);

figure(4)
subplot(1,2,1)
    step(Wdt,Wdt_approx1), grid
    legend('W \Delta \theta(s)', 'W \Delta \theta_{approx1}(s)')
subplot(1,2,2)
    step(Wdq,Wdq_approx1), grid
    legend('W \Delta q(s)', 'W \Delta q_{approx1}(s)')

figure(5)

```



```
subplot(1,2,1)
    bode(Wdt,Wdt_approx1), grid
    legend('W \Delta \theta(s)', 'W \Delta \theta_{approx1}(s)')
subplot(1,2,2)
    bode(Wdq,Wdq_approx1), grid
    legend('W \Delta q(s)', 'W \Delta q_{approx1}(s)')
```

Appendice 1: Creare uno script in MatLab

MatLab consente di memorizzare una sequenza di istruzioni in un *file*; questo, per essere accessibile, deve avere l'estensione “.m” e pertanto si chiama *m-file*. Gli m-file possono essere di due tipi: *script* o *function* (vedi Figura 5).

Gli script contengono semplicemente una sequenza di istruzioni MatLab, nella forma in cui si scriverebbero dalla linea attiva della finestra di *Command*.

Utilizzano tutte le variabili già definite prima e, al termine dell'esecuzione, tutte le eventuali modifiche sono visibili all'esterno. In questo senso si dice che tutte le variabili sono GLOBALI. Accertarsi che la “Current Directory” di MatLab sia la stessa in cui è memorizzato lo *script*, per eseguirlo è sufficiente digitare, nella finestra *Command Window*, il nome del file (senza l'estensione .m), oppure premere il button “Run” evidenziato in Figura 6.

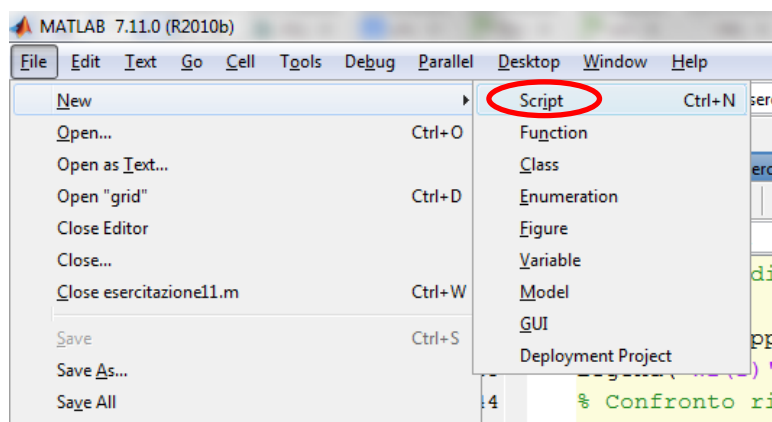


Figure 5. Creare un m-file MatLab.

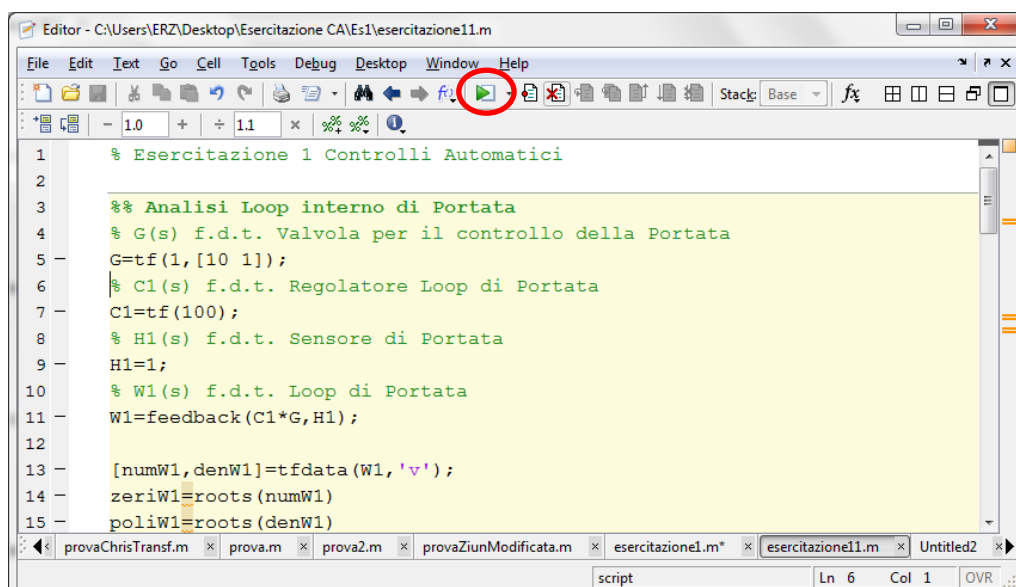


Figure 6. Eseguire un m-file Matlab.

Appendice 2: Comandi matlab:

Di seguito riportiamo una breve guida tratta dal *Documentation Center* della *MathWorks* (http://www.mathworks.it/it/help/symbolic/mupad_ref/solve.html) dove descriveremo brevemente i principali comandi MatLab visti in questa esercitazione.

- `clc` % Clear command window
- `close all` % Close figure
- `clear all` % Clear variables and functions from memory
- `tf` %Construction:
 `SYS = TF(NUM,DEN)` creates a continuous-time transfer function `SYS` with numerator `NUM` and denominator `DEN`. `SYS` is an object of class `@tf`. For example,
 `sys = tf([1 2],[1 0 10])`
 specifies the transfer function $(s+2)/(s^2+10)$
- `feedback` %Feedback connection of two input/output systems.
 `M = FEEDBACK(M1,M2)` computes a closed-loop model `M` for the feedback loop:

$$\begin{array}{c} u \text{ ---> } O \text{ ----> } [M1] \text{ ----+----> } y \\ | \hspace{10em} | \hspace{10em} y = M * u \\ + \text{ ----> } [M2] \text{ <----+} \end{array}$$
- `tfdata` % For a single SISO model `SYS`, the syntax
 `[NUM,DEN] = TFDATA(SYS,'v')`
 returns the numerator and denominator as row vectors than cell arrays.
- `roots` % Find polynomial roots.

Per maggiori dettagli relativi ai comandi MatLab digitare nella “*Command Window*” il comando “*help*” o il comando “*doc*” seguito dal nome del comando di interesse. Per esempio: “*help bode*” oppure “*doc step*”.

Authors: Prof. Elio Usai¹, Ing. Alessandro Pilloni¹.

E. Usai and A. Pilloni are with the Department of Electrical and Electronic Engineering (DIEE), University of Cagliari, Cagliari 09123, Italy. E-mail addresses: {eusai,alessandro.pilloni}@diee.unica.it



This work is licensed under the Creative Commons Attribuzione 3.0 Italia License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/it/>.