Tesina Sistemi Dinamici: Due serbatoi

Giuliano Ranauro

17/01/2024

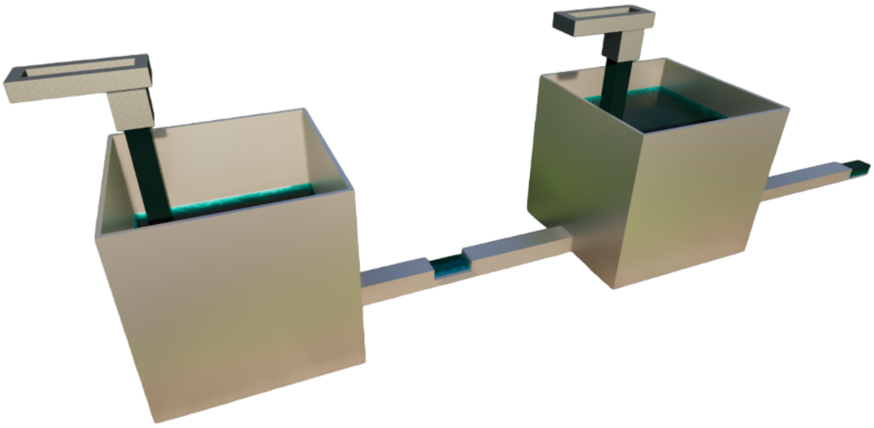


Figure 1: Enter Caption

Contents

Risposta nel tempo **2**
Schema del sistema 2
Circuito Equivalente 2
Equazioni del sistema 2
Modello nello spazio di stato 3
Funzione di trasferimento calcolata con matlab 4
Immissione delle matrici 4
Calcolo della funzione di trasferimento 4
Risposta del sistema nel tempo 5
Risposta ad un segnale custom 6
Plot dei singoli modi 7
Risposta del sistema al variare dei parametri 7

Risposta in frequenza **10**
Diagrammi di Bode 10
Lettura del diagramma dei moduli 11
Lettura del diagramma delle fasi 11
Uscita "Steady State" 11
Richieste 12

Risposta nel tempo

Schema del sistema

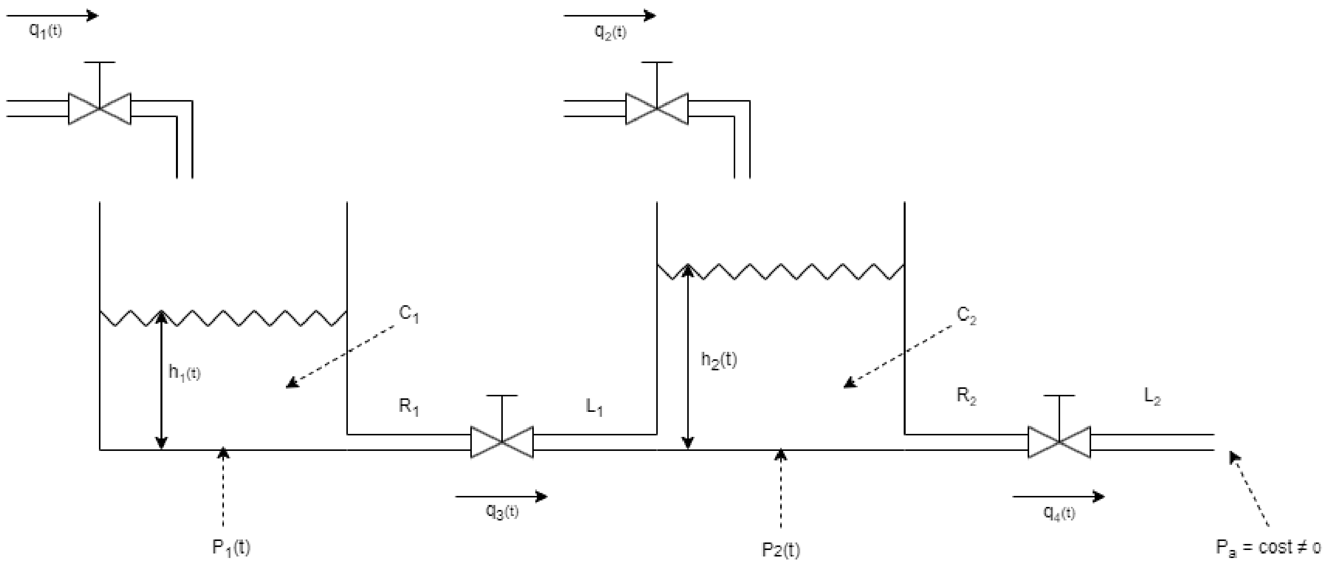


Figure 2: Enter Caption

Schema del sistema idraulico composto da due serbatoi.

Circuito Equivalente

E‘ possibile rappresentare lo schema idraulico in termini di **circuito elettrico**, con le componenti che conosciamo, andando a sostituire:

Schema Idraulico	Schema Elettrico
Serbatoio	Condensatore
Rubinetti	Generatori di corrente
Pressione atmosferica	Generatore di tensione
Resistenze idrauliche	Resistenze elettriche
Induttanze idrauliche	Induttanze idrauliche

Bisogna tenere a mente diversi accorgimenti:

- Come nello schema idraulico, le resistenze ed induttanze vanno poste tra due serbatoi (condensatori).
- Tra un serbatoio e l'altro, se connessi, scorre un flusso qn; nello schema elettrico questo si traduce nel flusso, elettrico, ovvero in *corrente elettrica.*

  Circuito equivalente

Equazioni del sistema

A partire dal circuito equivalente è possibile estrapolare le equazioni del sistema, che lo descrivono:

$$\begin{cases} C_1 \cdot \frac{\mathrm{d}}{\mathrm{d}t} P_1 = q_1 - q_3 \\ C_2 \cdot \frac{\mathrm{d}}{\mathrm{d}t} P_2 = q_2 + q_3 - q_4 \\ L_1 \cdot \frac{\mathrm{d}}{\mathrm{d}t} q_3 = -R_1 \cdot q_3 + p_1 - p_2 \\ L_2 \cdot \frac{\mathrm{d}}{\mathrm{d}t} q_4 = -R_1 \cdot q_4 + p_1 - p_a \end{cases} \tag{1}$$

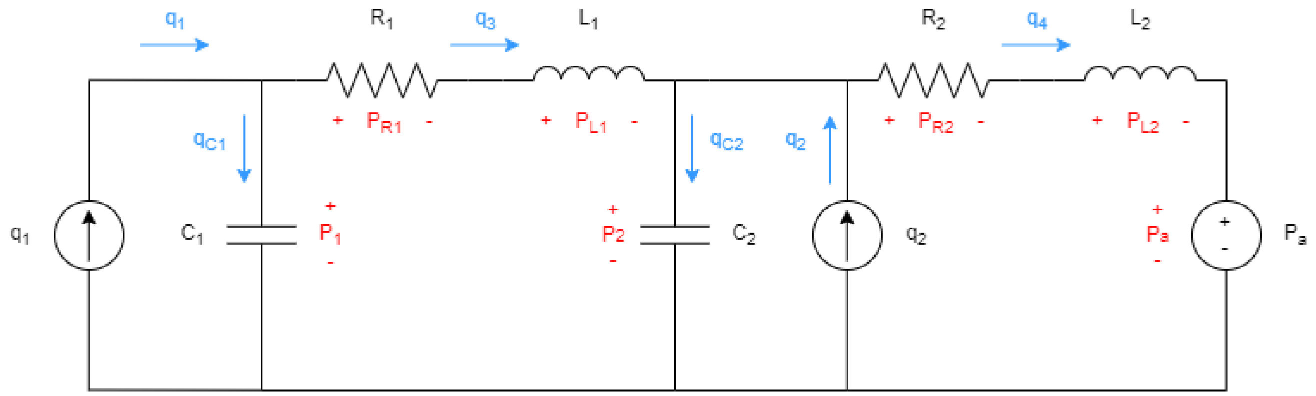


Figure 3: Enter Caption

Ricordando la natura del sistema, possiamo identificare i valori di input:

- **Input:**
  - q1 : flusso in immissione tank 1
  - q2 : flusso in immissione tank 2
  - Pa: Pressione atmosferica "a valle"; Se si considera questa pressione molto più piccola rispetto alla pressione alla base del tank 2, la si può ignorare.

Modello nello spazio di stato

Possiamo quindi partire dalle equazioni del sistema e rappresentarlo nello spazio di stato, andando a scegliere **Variabili di stato**, **Entrate** ed **uscite**:

$$Variabili\ di\ stato \begin{cases} x_1 = P_1 \\ x_2 = P_2 \\ x_3 = q_3 \end{cases} \tag{2}$$

$$input \begin{cases} u_1 = q_1 \\ u_2 = q_2 \\ u_3 = P_a \end{cases} \tag{3}$$

$$uscita \begin{cases} y_1 = P_1 \end{cases} \tag{4}$$

Possiamo scrivere sottoforma di matrice:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} = \begin{bmatrix} 0 & 0 & -\frac{1}{C_1} & 0 \\ 0 & 0 & \frac{1}{C_2} & -\frac{1}{C_2} \\ \frac{1}{L_1} & -\frac{1}{L_1} & -\frac{R_1}{L_1} & 0 \\ 0 & \frac{1}{L_2} & 0 & -\frac{R_2}{L_2} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} \frac{1}{C_1} & 0 & 0 \\ 0 & \frac{1}{C_2} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{L_2} \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

Figure 4: Spazio di stato in notazione matriciale

Possiamo inoltre "scegliere" l'uscita del sistema che vogliamo monitorare, per questo esempio scegliamo la *pressione P1*, che per semplicità coincide proprio con la variabile di stato *x1*:

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Figure 5: Uscita del sistema

Funzione di trasferimento calcolata con matlab

Siccome la funzione di questo sistema risulta complicata da calcolare "a mano", possiamo servirci di **MatLab**, che una volta immessi i parametri del sistema tramite spazio di stato, può restituirci la **Funzione Di Trasferimento** *numerica* del sistema:

Immissione delle matrici

Funzione di trasferimento calcolata con MatLab Dopo aver dichiarato i parametri, possiamo immettere le matrici appena trovate all'interno dell'ambiente MatLab; queste equazioni descrivono il sistema.

```
A = [   0,      0,      -1/C1,      0;
        0,      0,      1/C2,       -1/C2;
        1/L1,   -1/L1,  -R1/L1,      0;
        0,      1/L2,   0,          -R2/L2
    ];

B = [   1/C1,   0,      0;
        0,      1/C2,   0;
        0,      0,      0;
        0,      0,      -1/L2
    ];

C = [1, 0, 0, 0];
D = [0, 0, 0];
```

Calcolo della funzione di trasferimento

Con i seguenti comandi possiamo trovare ***le*** funzioni di trasferimento associate ai diversi input:

```
G = ss(A, B, C, D);           % Sistema a partire dallo spazio di stato

TFs = tf(G);                  % Funzioni di trasferimento per i diversi input
```

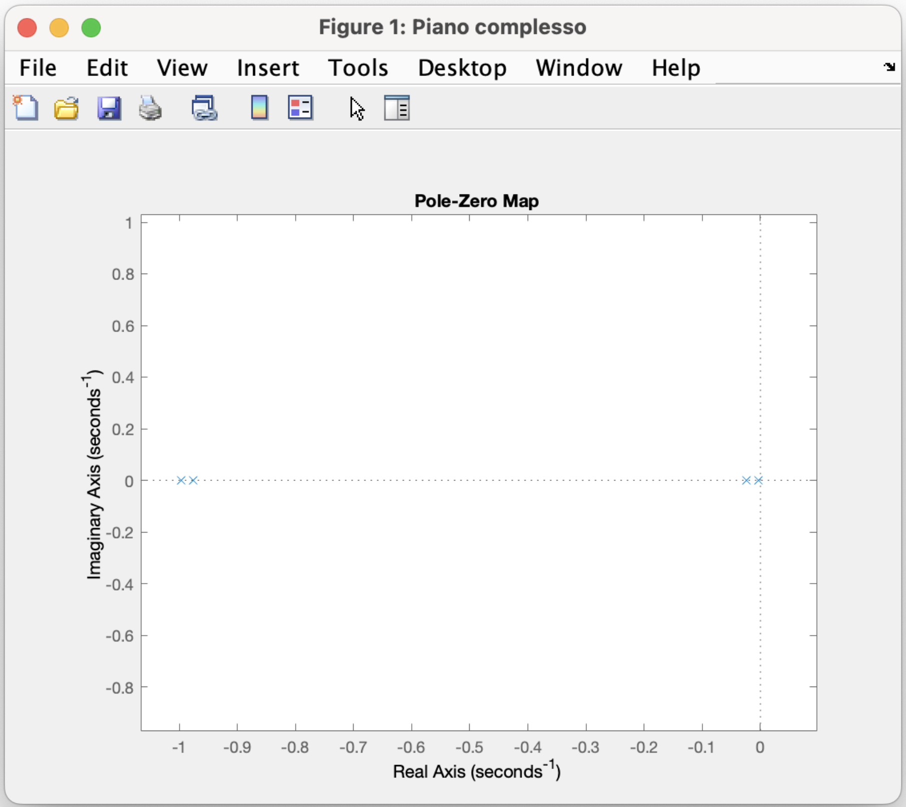
Possiamo accedere ad una delle funzioni di trasferimento con il comando `Tfs(k)`; viene di seguito riportato la FDT numerica associata all'input 1:

$$G(s)_1 = \frac{0,01s^3 + 0.02s^2 + 0.01s + 16 \cdot 10^{-4}}{s^4 + 2s^3 + 1.03s^2 + 26 \cdot 10^{-2} + 8.3 \cdot 10^5}$$

Risposta del sistema nel tempo

Dopo aver rappresentato il sistema nell'ambiente MatLab, possiamo iniziare ad ottenere informazioni sulle possibili risposte del sistema.

La prima cosa che possiamo fare è controllare la posizione dei **poli** e degli **zeri** nel piano complesso; questo è possibile tramite il comando `pzmap(G)`, che restituisce il seguente grafico:



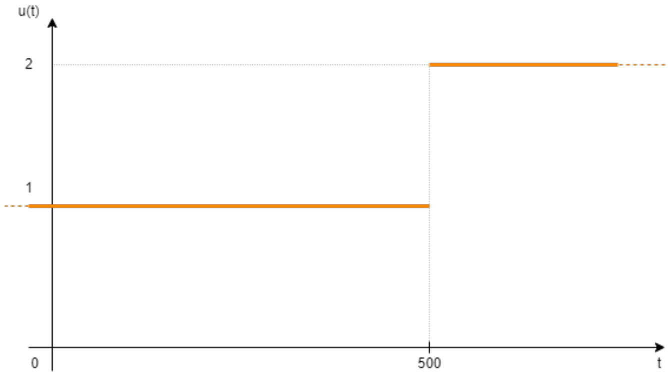
Dopo aver visualizzato i diversi poli e zeri del sistema, possiamo passare a trovare l'uscita del sistema ad un segnale; siccome il sistema è composto di 3 input, nel momento in cui calcoliamo `step(G)`, ovvero la risposta ad un segnale gradino, matlab "mette in ingresso" un gradino unitario **a tutti e 3 gli input**.

Capiamo quindi che questo modo è un po' troppo limitante per il calcolo dell'uscita.

Possiamo però "costruirci" un segnale custom da utilizzare come uno (o più) degli input del sistema; vediamo come fare:

Risposta ad un segnale custom

La prima cosa da fare è sicuramente quella di costrire l'input personalizzato; vogliamo realizzare un segnale gradino che parte da un'ampiezza 1, e che varia di ampiezza (arrivando a 2) dopo 500s:



Possiamo scrivere il segnale come somma di segnali elementari:

$$\begin{cases} u_1(t) = \mathbf{1}(t) \\ u_2(t) = \mathbf{1}(t-500) \end{cases}$$

$$u(t) = u_1(t) + u_2(t)$$

Possiamo trasformare il segnale nel dominio di Laplace ottenendo:

$$\begin{cases} u_1(t) \longleftrightarrow U_1(s) = \frac{1}{s} \\ u_2(t) \longleftrightarrow U_2(s) = \frac{1}{s} \cdot e^{-100s} \end{cases}$$

$$U(s) = U_1(s) + U_2(s)$$

Possiamo quindi costruire il segnale su MatLab "pezzo per pezzo" nel seguente modo:

```
U21 = 1/s;                                  % Gradino unitario
U22 = tf([0 1], [1 0], 'inputDelay', 500);  % Gradino ritardato di 100
U2 = U21 + U22;                             % Trovo il segnale risultante
```

Per trovare l'uscita ci basta quindi moltiplicare il segnale (nel dominio di Laplace) per la funzione di trasferimento (che abbiamo ottenuto tramite lo spazio di stato); bisogna stare attenti però a "selezionare" la funzione di trasferimento giusta, perchè come abbiamo già visto, le FDT sono 3, una per ogni input.

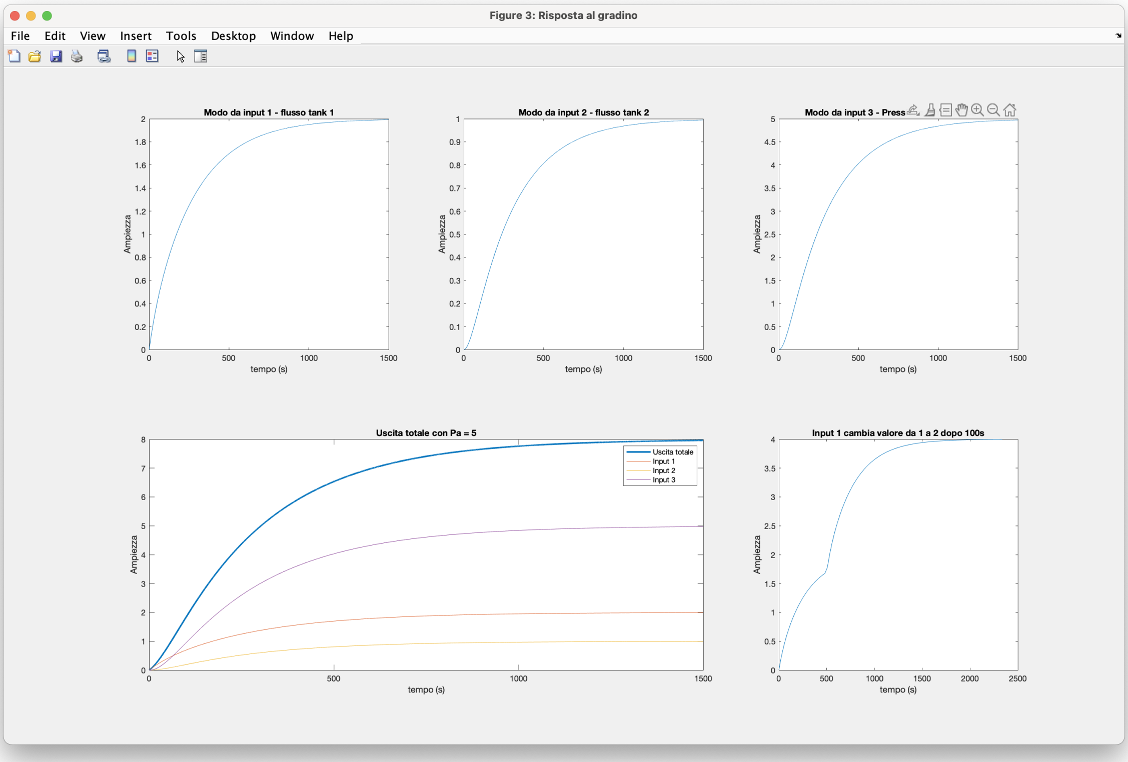
In questo caso vogliamo che il segnale sia attribuito all'input 1, ovvero al flusso del tank 1, quindi scegliamo la FDT 1:

```
Y3 = G_default(1) * U2;                     % Calcolo l'uscita nel dominio di Laplace
[y2, t2] = impulse(Y3);                     % Calcolo l'antitrasformata
```

Possiamo calcolare l'antitrasformta con `impulse(Y3)`, perché nel dominio di Laplace l'impulso corrisponde ad `1`, quindi moltiplicando per 1 il segnale *Y(s)* non cambia; successivamente provvede MatLab ad *antri trasformare* (sempre con impulse).

Plot dei singoli modi

Possiamo quindi effettuare il plot dei singoli modi e dei segnali totali:



- **Plot 1**: modo dall'input 1 gradino unitario (q1)
- Plot 2: modo dall'input 2 gradino unitario (q2)
- Plot 3: modo dall'input 3 gradino di ampiezza 5 (Pa)
- Plot 4:
  - In Blu: uscita totale dei segnali 1, 2 e 3.
  - Il resto dei plot sono i singoli modi.
- Plot 5: uscita totale agli input:
  - in 1: gradino che inizia unitario e cambia valore in t=500 arrivando ad un valore pari a 2.
  - in 2: gradino unitario.
  - in 3: gradino unitario.

Risposta del sistema al variare dei parametri

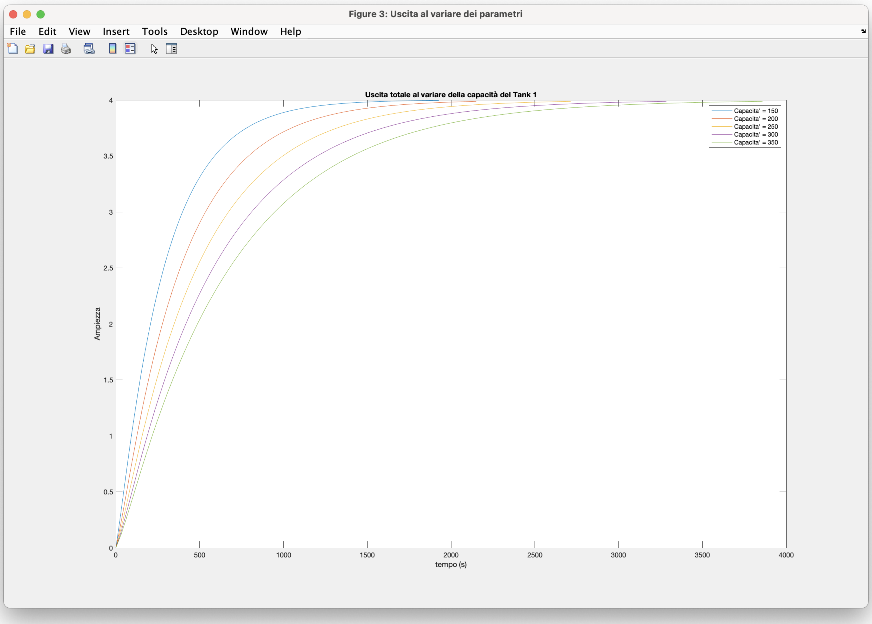
Risposta al variare della capacità In questo esempio facciamo variare la capacità del tank 1, con il seguente codice:

```
for i=1:n
    G_capcity_1 = ss(A, B, C, D);       % Rappresento il sistema con il valore di C1 corrente

    [yn, tn] = step(G_capcity_1);       % Calcolo la risposta al gradino unitario
    yn_sum = sum(yn, [2 3]);            % Sommo le uscite dei 3 ingressi
    plot(tn, yn_sum)                    % Disegno le diverse uscite
    hold on;

    C1 = C1 + 50;                       % Aggiorno il valore di C per la prossima iterazione
    [A, B, C, D] = updateMatrixes(C1, C2, R1, R2, L1, L2);  % Aggiorno le matrici con il valore di C corr
```

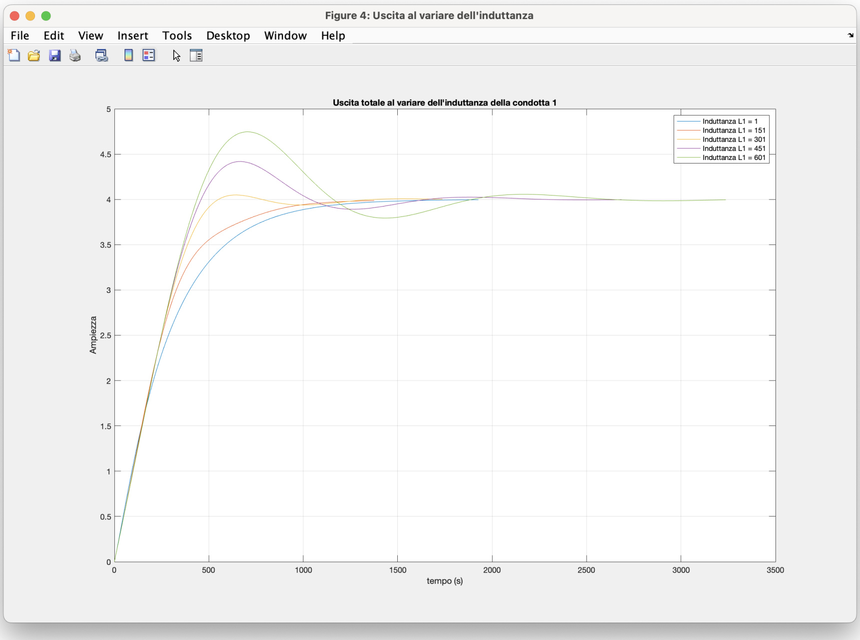
```
    legend_text{i} = sprintf("Capacita' = %i", C1);         % Attribuisco ad ogni curva un nome per la le
end
```

Andiamo quindi, ad ogni iterazione, ad aumentare la capacità del serbatoio; possiamo notare dalle curve, che man mano che la capacità aumenta, la pressione del sistema impiega sempre più tempo ad arrivare a regime.

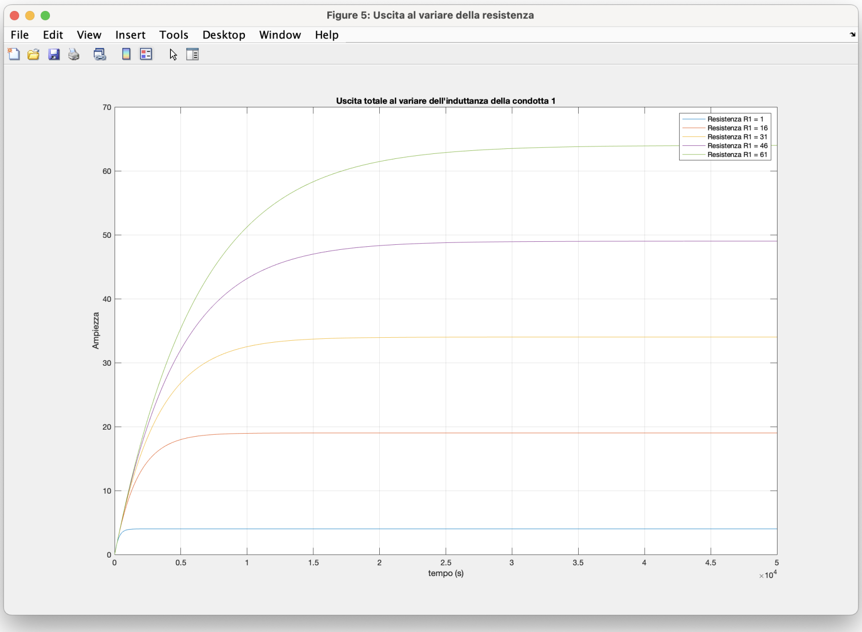
Notiamo, però, che il *valore* di regime resta invariato.



Risposta al variare dell'induttanza Allo stesso modo possiamo far variare l'induttanza; ad ogni iterazione variamo di un valore maggiore, in modo da ottenere un risultato più "interessante":



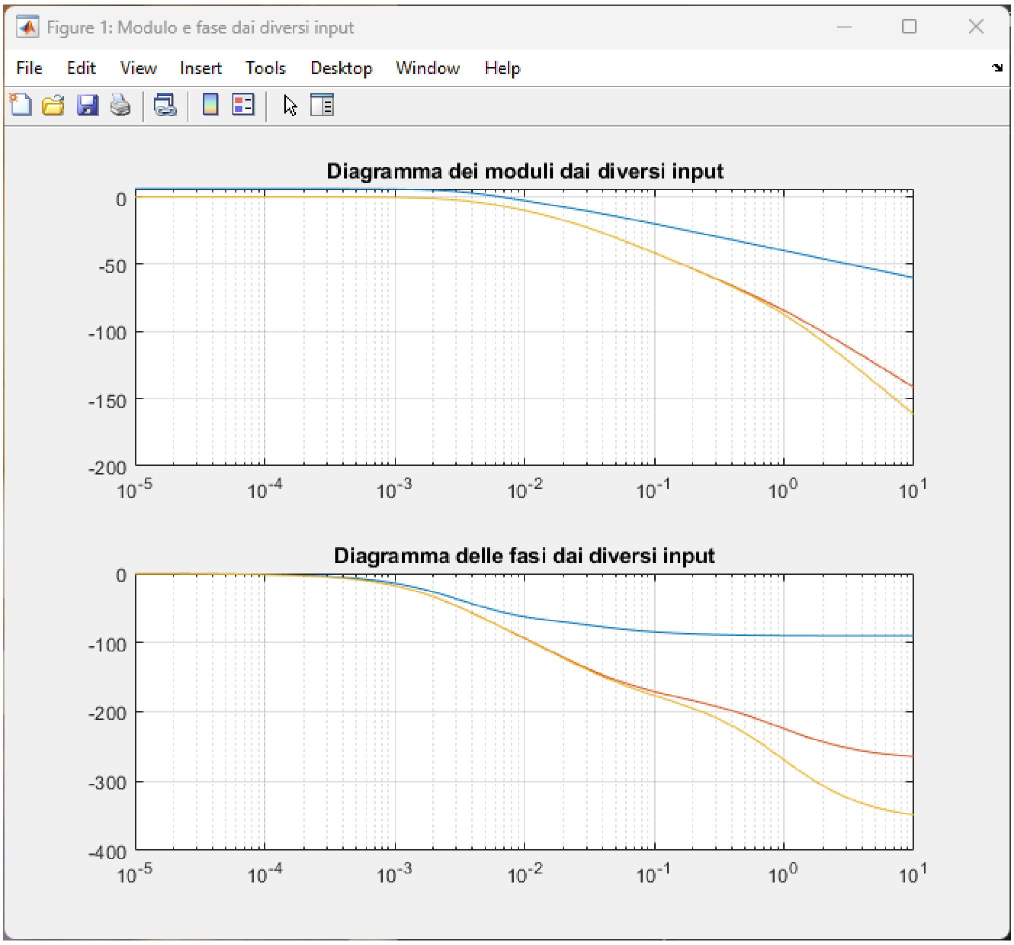
Risposta al variare della resistenza Se la resistenza della condotta 1 aumenta, allora la pressione sul tank 1 aumenta, perchè il liquido avrà maggiore difficoltà a spostarsi dal tank 1 al tank 2, infatti:



Risposta in frequenza

Diagrammi di Bode

Possiamo effettuare un'analisi anche dal punto di vista della frequenza, andando a tracciare i **diagrammi di Bode**, sia dei moduli che delle fasi:



I diagrammi di Bode sono in grado di darci molte informazioni sul sistema in esame; possiamo ad esempio vedere che tutte le frequenze inferiori alla frequenza

$$\omega < 10^{-3} \quad rad/s$$

vengono leggermente amplificate, mentre a partire da questa frequenza in poi verranno attenuate.

Ma cosa vuol dire?

Essenzialmente stiamo dicendo che, data una sinusoide di una certa ampiezza in ingresso del tipo:

$$u(t) = X \cdot sin(\omega_0 \cdot t)$$

avremo in output un segnale sempre sinusoidale, ma con modulo e fase cambiati, a seconda dei valori letti sui diagrammi di Bode:

$$y(t) = X \cdot |G(j\omega_0)|_{dB} \;\; \cdot sin(\omega_0 \cdot t \;\; + \angle G(j\omega_0))$$

Lettura del diagramma dei moduli

Di conseguenza, se sul diagramma dei moduli vedremo un valore positivo (in corrispondenza della frequenza dell'ingresso u(t)), sapremo che l'input verrà amplificato, mentre se vedremo un valore negativo l'input verrà attenuato.

Otteniamo questo comportamento per via del fatto che sul diagramma di Bode, i moduli vengono espressi in **deciBel**; un valore x0 può essere convertito in deciBel secondo la formula:

$$x_{dB} = 20 \cdot log_{10}(x_0)$$

Quindi, quando leggiamo il valore sul diagramma dei moduli, e lo convertiamo nuovamente in valore naturale, seguendo la formula:

$$x_{dB} = 20 \cdot log_{10}(x_0) \implies x_0 = 10^{\frac{x_{dB}}{20}}$$

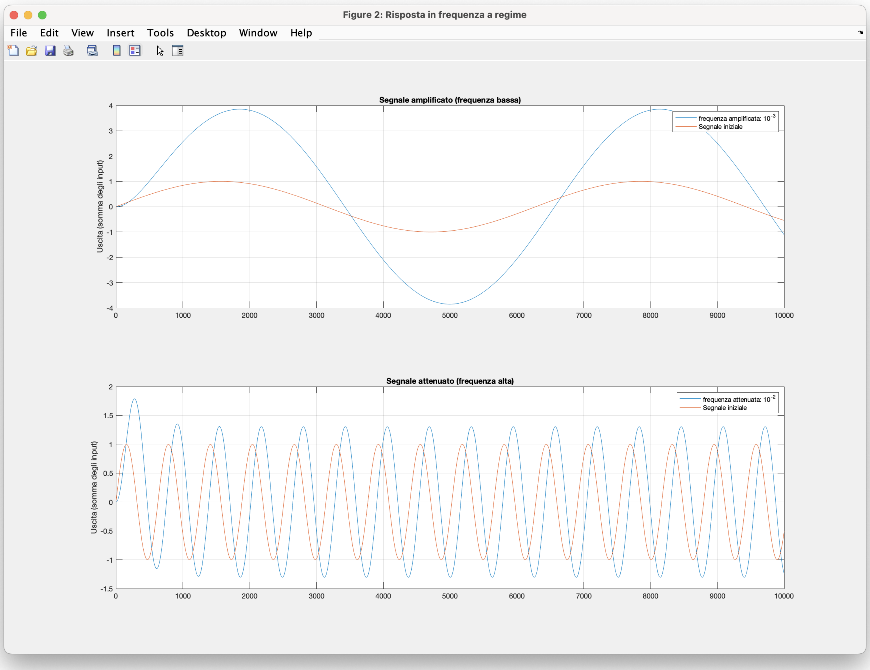
Lettura del diagramma delle fasi

Il diagramma delle fasi ci dice se la fase della sinusoide in uscita è **in anticipo o in ritardo** rispetto alla sinusoide in entrata; se leggiamo, ad esempio, un valore di *+90°*, questo vorrà dire che la sinusoide in uscita è in anticipo di 90° rispetto al segnale in entrata.

Il segnale in uscita può anche essere **in fase** col segnale in entrata, ovvero se, in corrispondenza di quella determinata frequenza, il diagramma delle fasi riporta come valore `0`.

Uscita "Steady State"

Possiamo quindi trovare l'uscita **Steady State**, ovvero la risposta del sistema dopo che il *transitorio* si è esaurito, e l'uscita **è a regime**.



In questo esempio possiamo vedere due tipi di output, che vengono calcolati a seconda dell'input. In blu osserviamo l'output, mentre l'input è rappresentato in arancione.

Nel primo caso, è stata scelta una frequenza tale che, a seconda delle specifiche del sistema, porta l'output ad essere **amplificato**; infatti possiamo osservare come il valore di uscita (bisogna sempre tenere presente che l'output è la somma di 3 contributi, visto che il sistema accetta 3 input) ha un'ampiezza maggiore rispetto al segnale di entrata.

Nel secondo caso, è stata scelta una frequenza **più alta**, in modo che l'output venisse **attenuato**. Continuiamo ad osservare un'ampiezza finale maggiore rispetto a quella del segnale di ingresso per il semplice motivo che il segnale di output è la somma di 3 contributi.

L'intervallo di frequenze (banda) scelto è lo stesso per entrambi i segnali, proprio per evidenziare le differenze di frequenza e di fase.

Richieste

La prova orale consiste nella presentazione di un esempio simulato in Matlab/Simulink e due domande dall'elenco allegato.

Un modello di traccia per la presentazione del modello da simulare (lo studente deve portare con sé il codice oppure un portatile per mostrare la simulazione) è il seguente:

- **Primo lucido**: disegno del sistema (si può scegliere tra quelli fatti a lezione eventualmente introducendo piccole varianti o un altro modello a scelta), equazioni, modello nello spazio di stato, funzione di trasferimento
- **Secondo lucido**: parametri, risposta impulsiva con singoli modi (espressione analitica e rappresentazione grafica)
- **Terzo lucido**: diagrammi di Bode e risposta a sinusoide/i
- **Quarto lucido**: risposta nel tempo a diverse condizioni (*ingressi*, *parametri*)