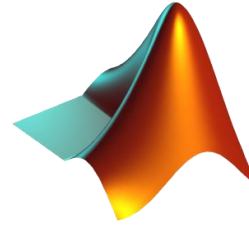


FONDAMENTI DI AUTOMATICA 2023/24

LABORATORIO 1 Identificazione dei parametri del sistema



POLITECNICO
MILANO 1863



LABORATORIO MATLAB/SIMULINK





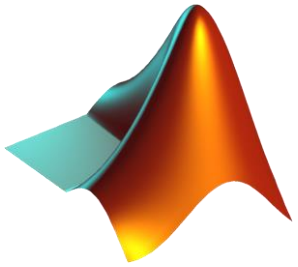
❑ Scopo:

- Ricavare la funzione di trasferimento di una sospensione per autoveicoli a partire dai dati sperimentali



❑ Contenuto del laboratorio:

- Introduzione a Matlab e Simulink
- Descrizione del sistema fisico
- Modello dinamico in spazio di stato
- Prima approssimazione del sistema osservando la risposta reale a scalino
- Funzione di trasferimento finale del sistema osservando le risposte in frequenza

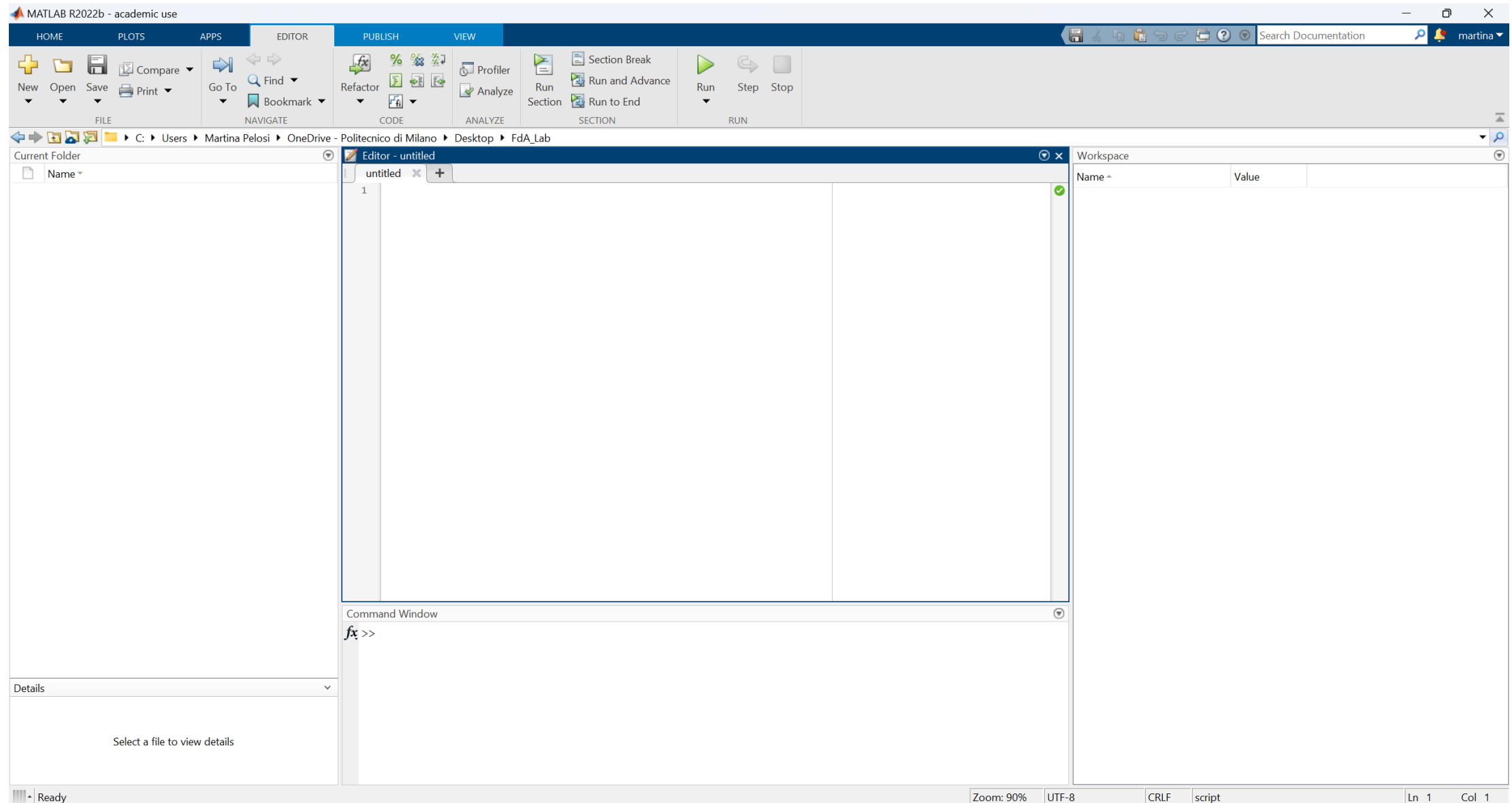


MATLAB = MATrix LABoratory

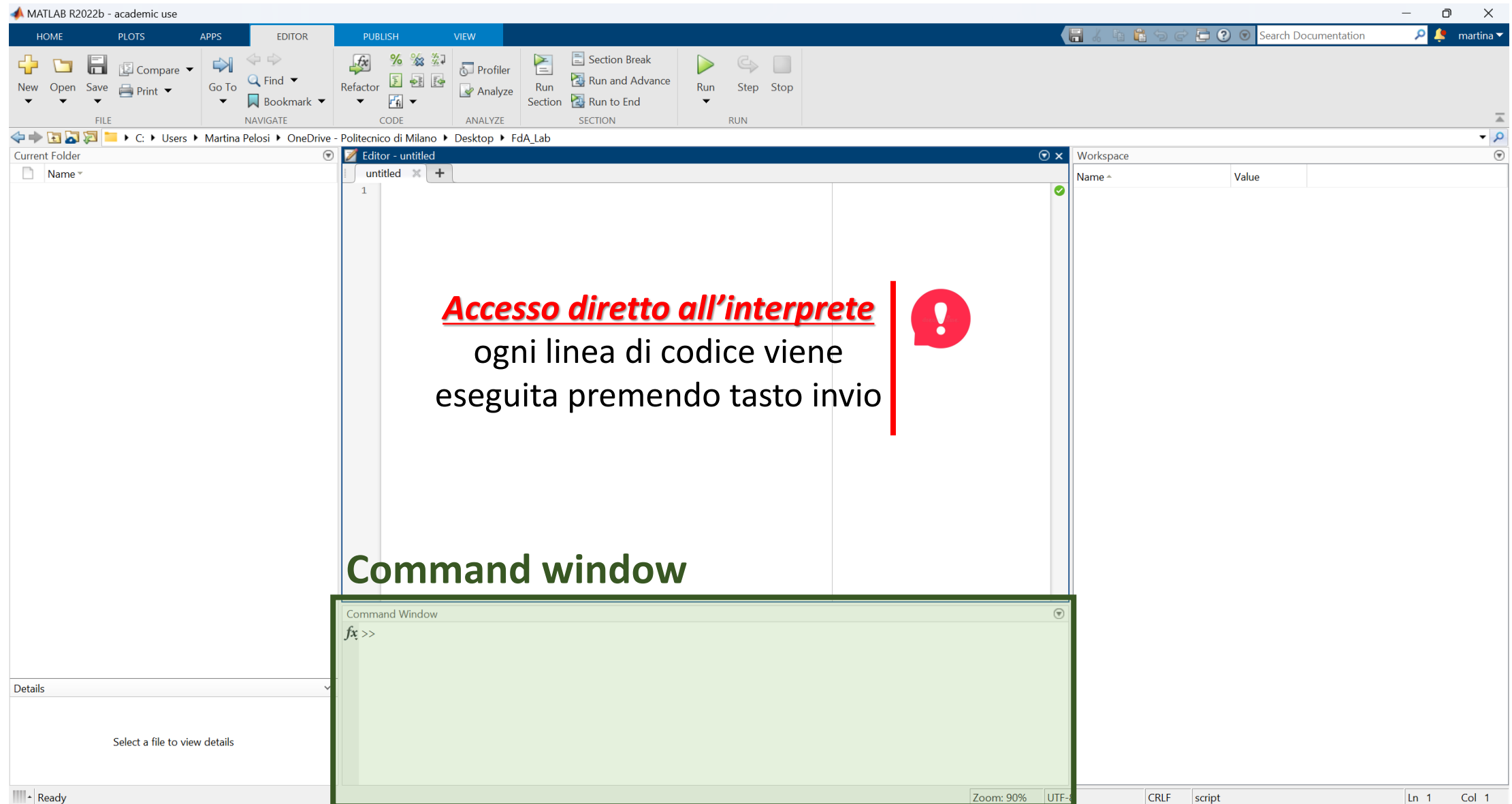
Linguaggio di programmazione utilizzato per:

- Calcoli matematici e matriciali
- Modellistica e simulazione
- Analisi dati e visualizzazione grafica
- Sviluppo di applicazioni e interfacce utente

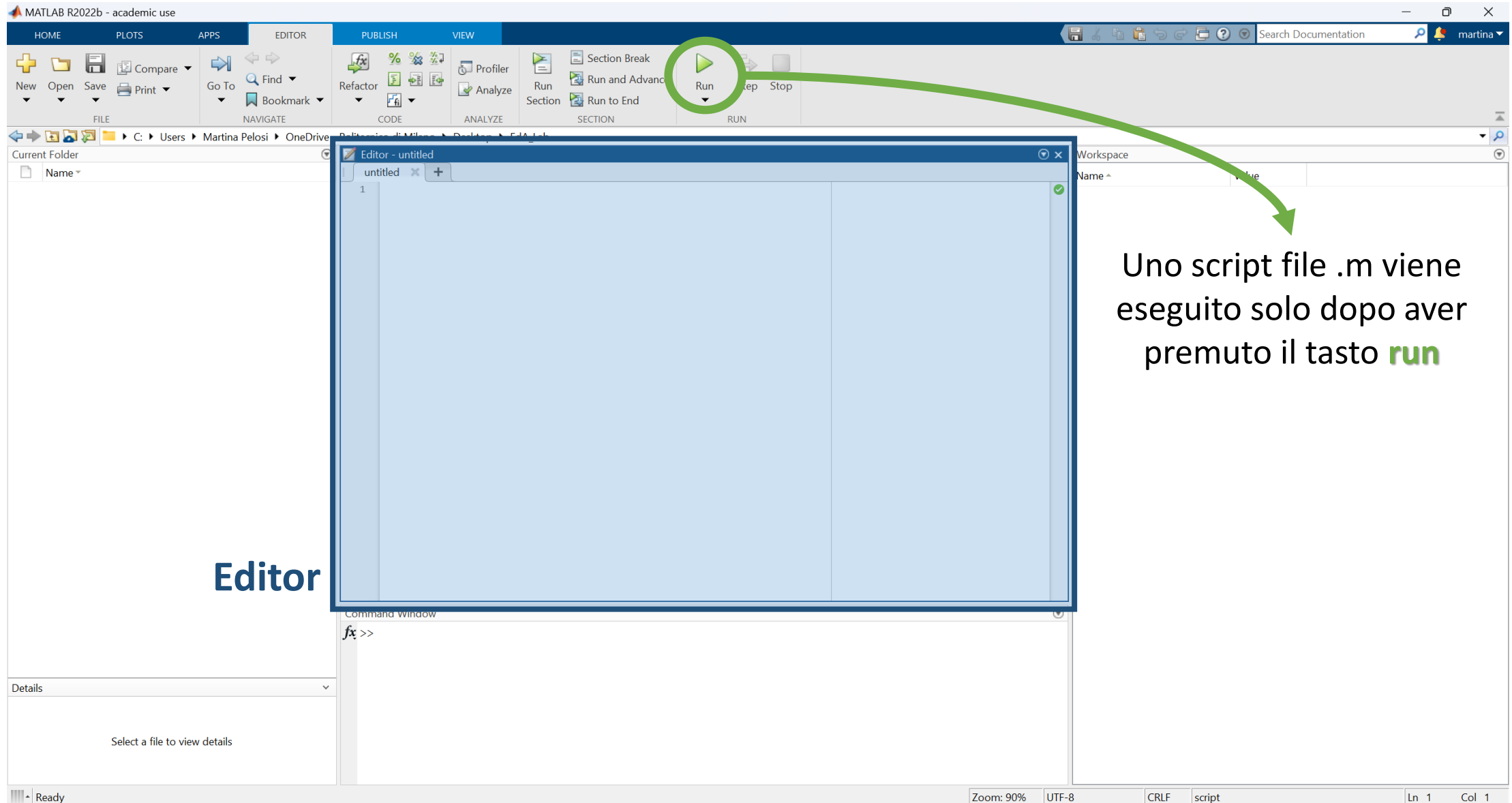
Introduzione a Matlab: visuale di default



Introduzione a Matlab: command window



Introduzione a Matlab: editor



Introduzione a Matlab: current folder

The image shows the MATLAB R2022b interface. The 'Current Folder' browser on the left is highlighted with an orange border. The 'Editor' window in the center contains the following text:

Restrizioni a nomi file

Non possono iniziare con un numero e
non possono contenere caratteri
speciali (solo '_' consentito)

A red exclamation mark icon is positioned to the right of this text. The 'Command Window' at the bottom shows the prompt 'x >>'. The 'Workspace' window on the right is empty. The status bar at the bottom indicates 'Zoom: 90%', 'UTF-8', 'CRLF', 'script', 'Ln 1', and 'Col 1'.

Introduzione a Matlab: variabili nel workspace

The screenshot shows the MATLAB R2022b interface. The top menu bar includes HOME, PLOTS, APPS, EDITOR, PUBLISH, and VIEW. The current folder is C:\Users\Martina Pelosi\OneDrive - Politecnico di Milano\Desktop\FdA_Lab. The editor window is titled 'untitled' and shows a single line of code: '1 |'. The Workspace window is open on the right, displaying a table of variables:

Name	Value
a	1.2300
b	1x2 cell
lettera	'c'
scatola	1x1 struct
stringa	"Hello world!"

Workspace
Dove visualizzare le variabili generate

Introduzione a Matlab: variabili nel workspace

The screenshot shows the MATLAB environment. The top toolbar includes a 'VIEW' button and a search bar. Below the toolbar, the 'Workspace' window is open, displaying a list of variables: 'a' (1.2300), 'b' (1x2 cell), 'lettera' ('c'), 'scatola' (1x1 struct), and 'stringa' ('Hello world!'). The 'scatola' variable is highlighted with a red box. A red arrow points from the 'scatola' variable in the workspace to the 'Variables - scatola' window. This window shows the details of the 'scatola' variable, which is a 1x1 struct with three fields: 'x' (1), 'y' (2), and 'z' (3). The 'Command Window' at the bottom shows the prompt 'fx >>'.

Doppio clic sulla variabile per visualizzarla nella finestra variables

- ❑ Comando `clear` per cancellare variabili dal workspace:
 - `clear all` pulisce workspace da tutte le variabili
 - `clear <var>` cancella solo la variabile `<var>`
- ❑ Comando `clc` per pulire command window
- ❑ Comando `help` per avere informazioni su una funzione/operazione (funzionamento, input e output)
 - `help elfun` elenco funzioni elementari
 - `help <fun>` informazioni sulla funzione `<fun>`
- ❑ Comando `close` per chiudere figure
 - `close all` chiude tutte le figure
 - `clear(fig)` chiude solo la figura `fig`
- ❑ Comando `load` per caricare file `.m` specificato
- ❑ Comando `save` per salvare variabili specificate in una directory



Non è necessario dichiarare le variabili per utilizzarle!

Per assegnare un valore ad una variabile si usa '=':

Nome





Non è necessario dichiarare le variabili per utilizzarle!

Per assegnare un valore ad una variabile si usa '=':

Nome

In entrambi i casi la
variabile compare nel
workspace

```
Command Window
>> a = 1.2;
fx >>

Command Window
>> a = 1.2
a =
    1.2000
fx >>
```

Il ';' non è obbligatorio dopo un comando!

Aggiungerlo serve a nascondere l'output del comando nel command window



SCALARI

Operazioni

Somma	$a + b$
Sottrazione	$a - b$
Moltiplicazione	$a * b$
Divisione	a / b
Elevamento a potenza	$a ^ b$



Le variabili scalari si indicano con la lettera minuscola

Utilizzare il comando `help` per sapere come usare una funzione

Funzioni

Trigonometriche	<code>sin, cos, acos, asin, atan</code>
Esponenziali/logaritmiche	<code>exp, log, log10</code>
Numeri complessi	<code>abs, angle, real, imag, conj</code>
Altre	<code>sqrt, sign, floor, ceil</code>

MATRICI

Operazioni

Somma

$A + B$

Sottrazione

$A - B$

Moltiplicazione riga per colonna

$A * B$

Moltiplicazione per scalare

$A * a$

Prodotto elemento per elemento

$A .* B$

Esponenziale di matrice

$\text{expm}(A)$

Funzioni

Creazione di matrici

`eye, zeros, ones`

Analisi matrice

`det, inv, diag, eig`



Le variabili matriciali si indicano con la lettera maiuscola

Se le dimensioni delle matrici non sono consistenti si genera un **errore**



Estrazione di sotto-matrici

$A(i, :) =$ tutti gli elementi della i-esima riga

$A(:, j) =$ tutti gli elementi della j-esima colonna

$A(i:h, j:k) =$ elementi dalla posizione (i,j) alla (h,k)

$A(i, \text{end}) =$ ultimo elemento della i-esima riga

Comandi per creazione di figure/grafici

`figure()` per aprire una figura

`close()` per chiudere una figura

`plot(x, y)` per creare un grafico di $y = f(x)$

`semilogx(x, y)` per creare un grafico di $y = f(x)$ in scala logaritmica su x

`subplot(m, n, p)` per creare una finestra con $m \times n$ grafici e crea gli assi del grafico corrente in posizione p

Comandi per gestione di grafici

`grid on` per aggiungere la griglia al grafico

`hold on` per sovrapporre più grafici nella stessa figura

`xlabel(...)`, `ylabel(...)` per aggiungere l'etichetta agli assi x e y

`title(...)` per aggiungere titolo al grafico

`axis([xmin, xmax, ymin, ymax])` per riscalarare l'asse x tra `xmin` e `xmax` e y tra `ymin` e `ymax`

Sistema dinamico LTI in spazio di stato

$$S = \begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}$$

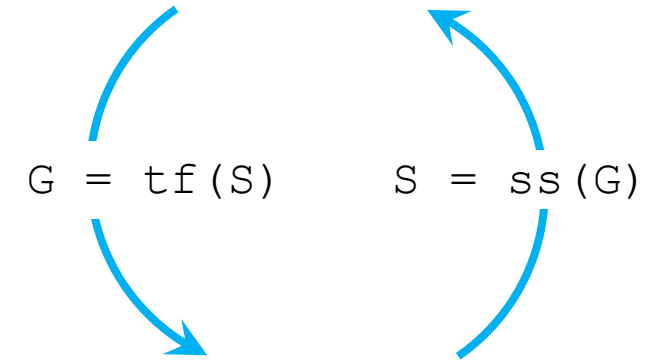
→ In Matlab: `>> S = ss(A, B, C, D);`

Calcolo autovalori

`eig(A), eig(S)`

Calcolo di poli, zeri, guadagno statico

`pole(S), zero(S), dcgain(S)`



Funzione di trasferimento $G(s)$

$$G(s) = \frac{s}{s^2 + 3s + 2}$$

→ In Matlab:

Metodo 1

```
>> s = tf('s');  
>> G = s/(s^2 + 3*s + 2);
```

Metodo 2

```
>> numeratore = [1 0];  
>> denominatore = [1 3 2];  
>> G = tf(numeratore, denominatore);
```

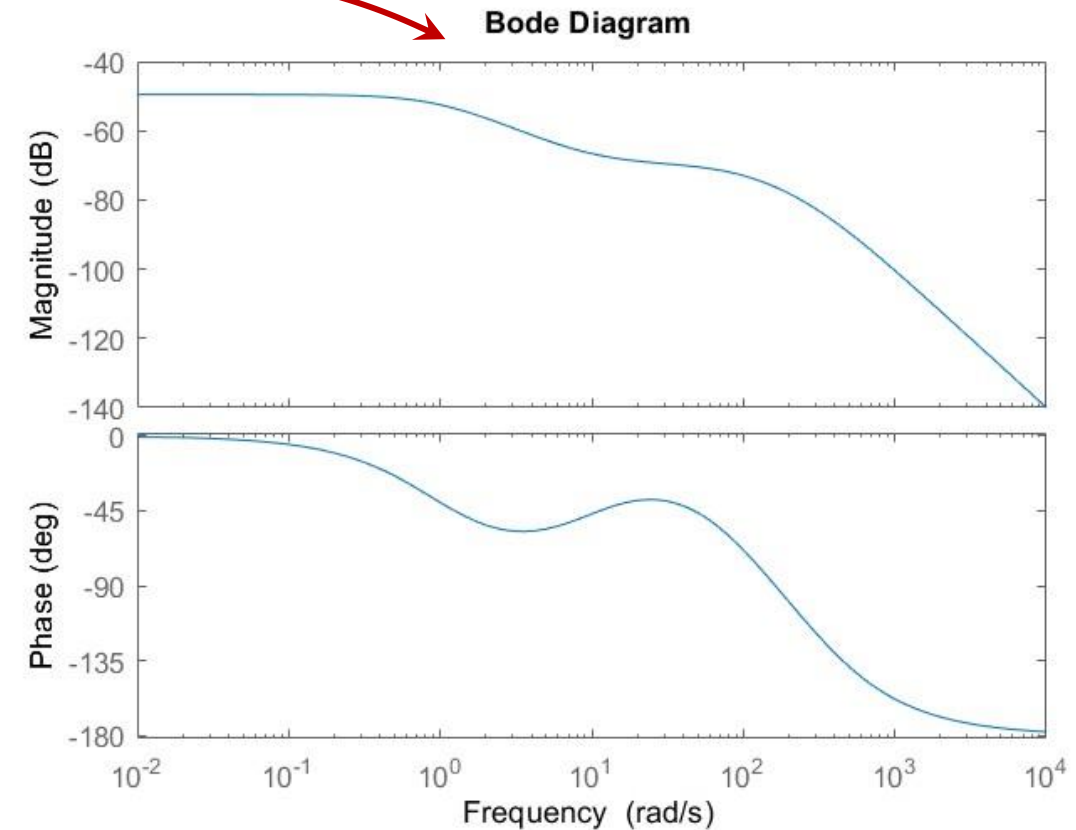
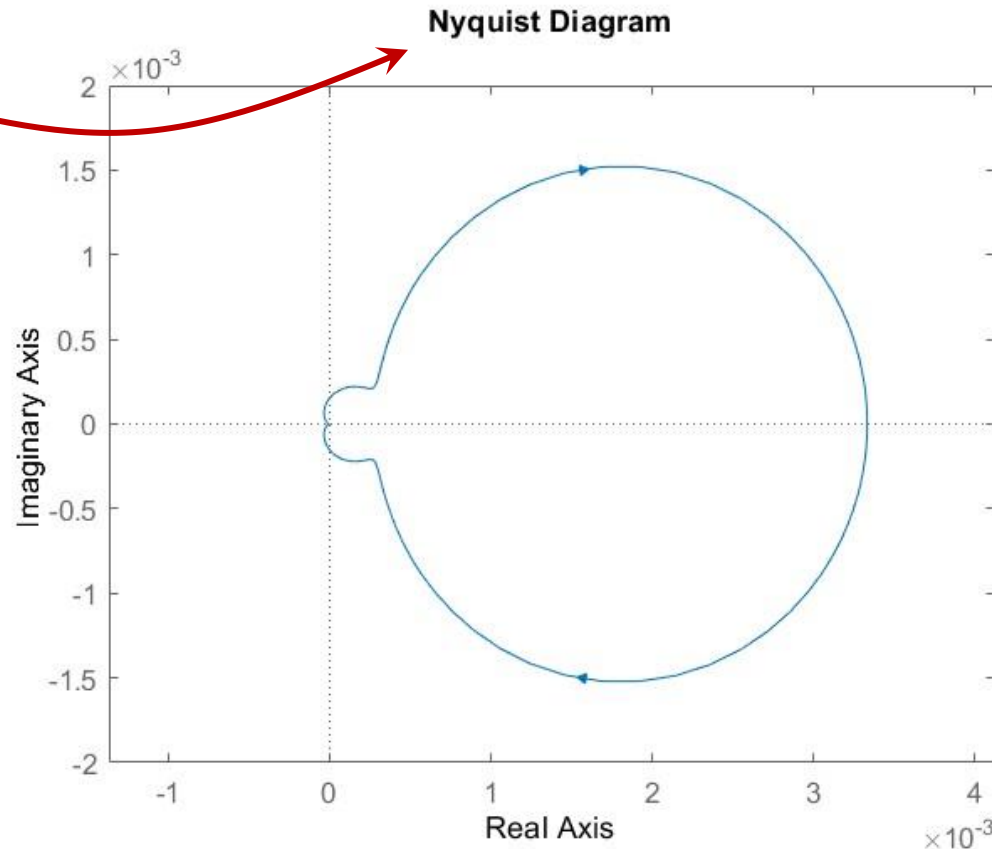

Introduzione a Matlab: diagrammi di Bode e Nyquist

```
>> s = tf('s');  
G = 10*(s+10)/((s+1)*(s+100)*(s+300));  
figure(1)  
bode(G)  
figure(2)  
nyquist(G)
```

Calcolo valori puntuali

```
[module, phase] = bode(G,w)
```

```
[Re,Im] = nyquist(G,w)
```





S in forma ss

Si possono confrontare più sistemi con

`step(S1,S2,...,t)` e `impulse(S1,S2,...,t)`

Comandi per il calcolo della risposta di S

Nel tempo

Movimento libero

```
[y,t,x] = initial(S,x0,t)
```

Risposta all'impulso

```
[y,t] = impulse(S,t)
```

```
[y,t,x] = impulse(S,t)
```

Risposta allo scalino unitario

```
[y,t] = step(S,t)
```

```
[y,t,x] = step(S,t)
```

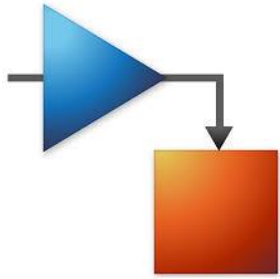
Risposta allo scalino di ampiezza A

```
opt = stepDataOptions('StepAmplitude',A)
```

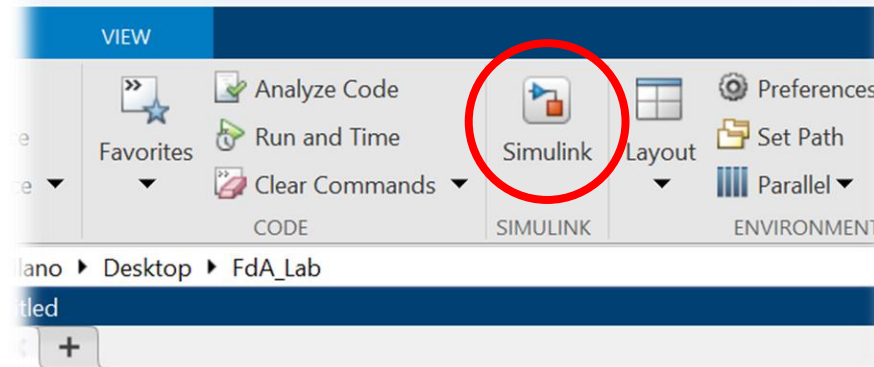
Movimento generico (libero + forzato)

```
[y,t,x] = lsim(S,u,t,x0)
```

Introduzione a Simulink

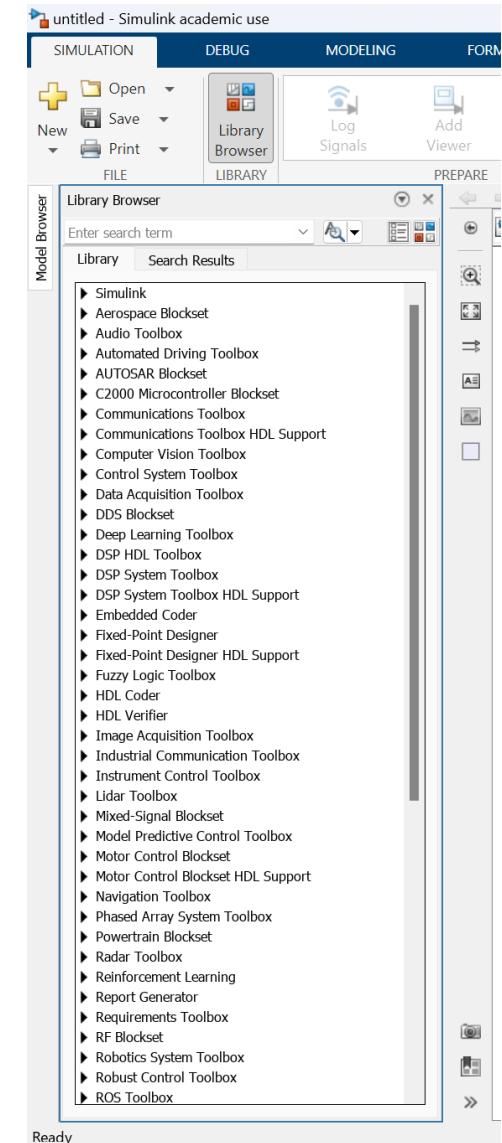


Simulink

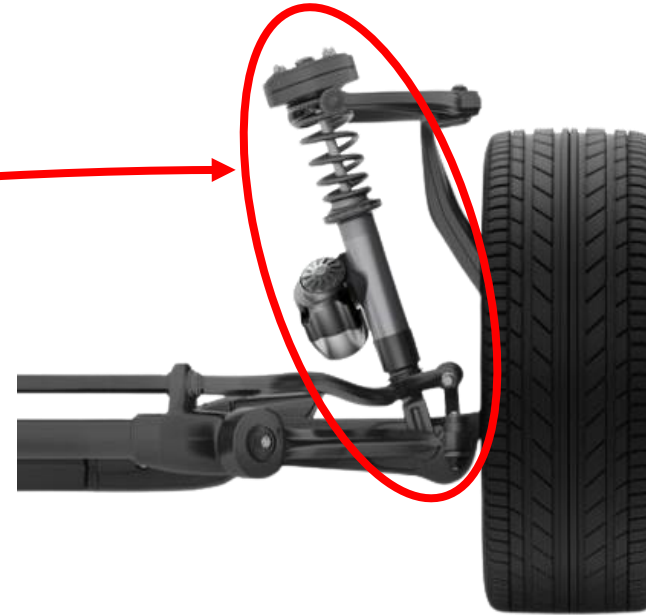
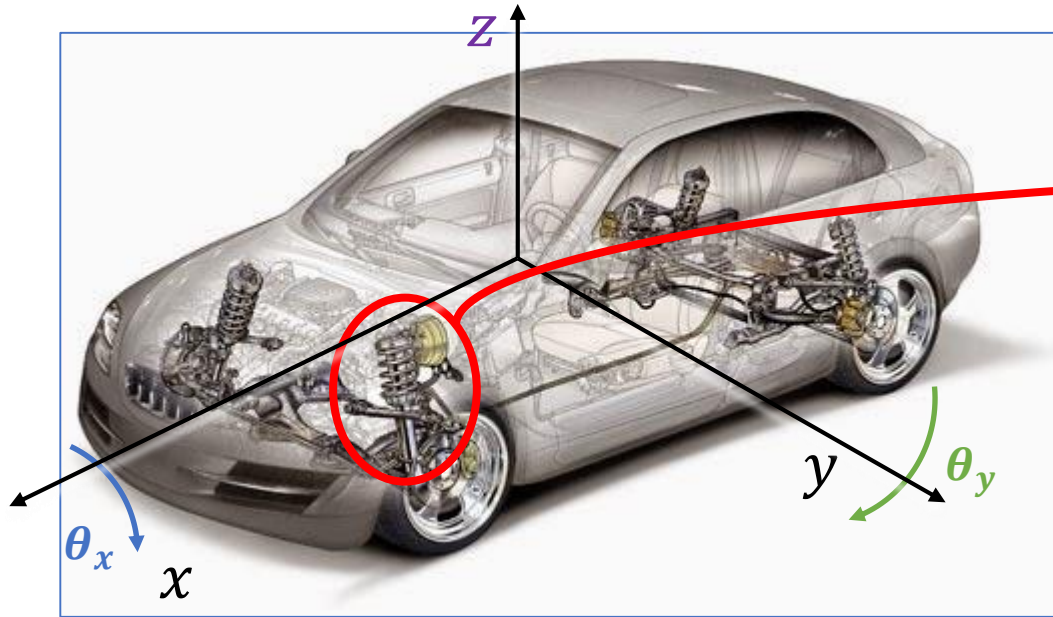


Software:

- Integrato in Matlab
- Ambiente di diagrammi a blocchi
- Modellazione, simulazione e analisi di sistemi dinamici



Introduzione al sistema fisico: sospensione attiva per autoveicoli



La **sospensione** influisce:

- Principalmente sulla dinamica **verticale** (Z)
- Su **rollio** (θ_x)
- Su **beccheggio** (θ_y)

Elemento elastico di rigidezza k

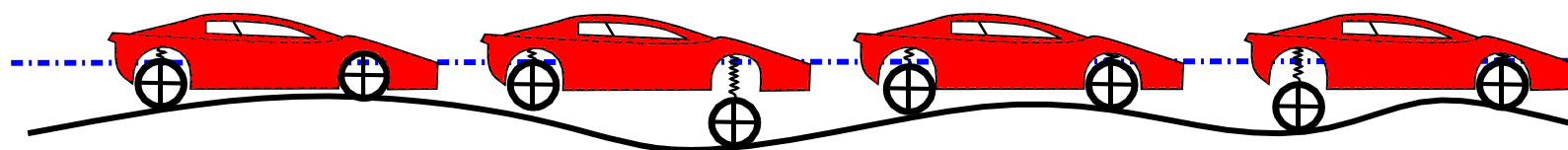
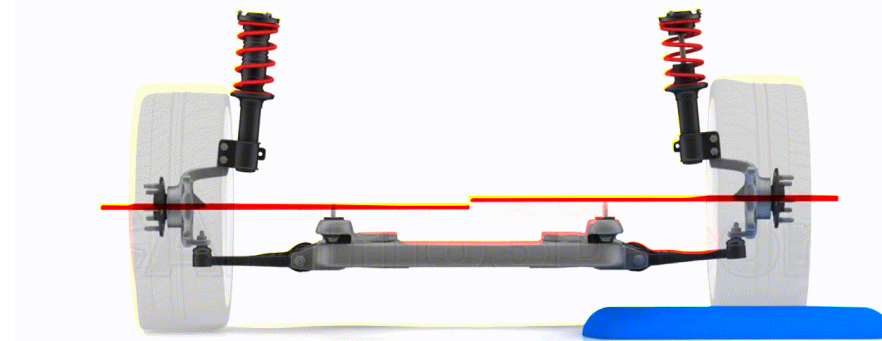
+

Elemento smorzante di smorzamento c

Introduzione al sistema fisico: sospensione attiva per autoveicoli

OBIETTIVI DEL CONTROLLO

Mantenere contatto costante tra pneumatico e manto stradale limitando variazioni sull'assetto del veicolo



*Comportamento
ideale*

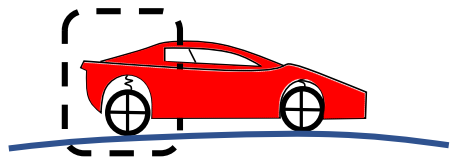
☐ **Tenuta di strada:**

Ottimizzare performance di guida limitando variazioni di forza di contatto tra pneumatico e strada

☐ **Comfort dei passeggeri:**

Ottimizzare comfort a bordo filtrando le irregolarità stradali limitando le variazioni sull'accelerazione verticale

Sospensione Ideale



Identificazione modello

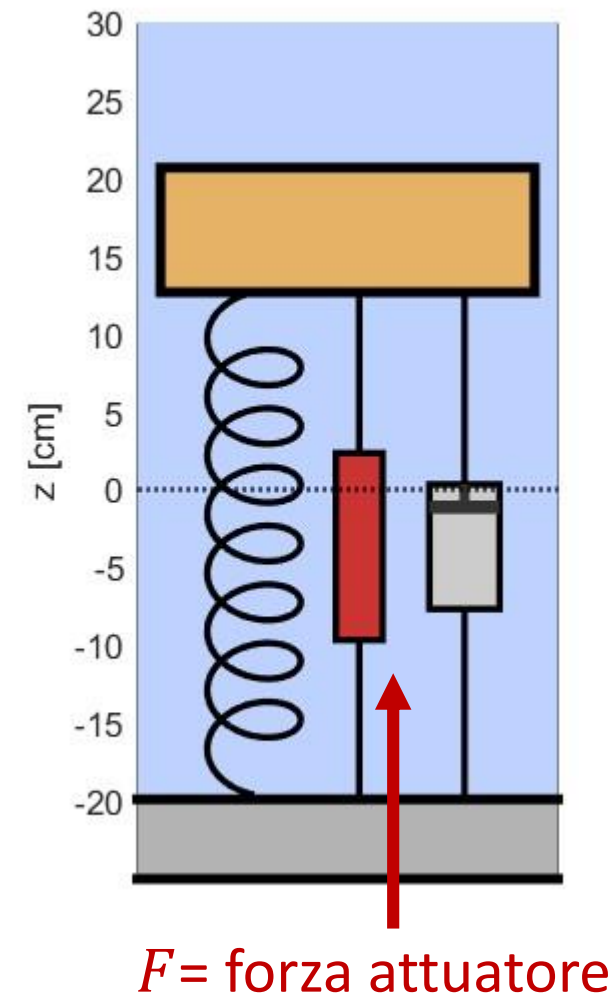
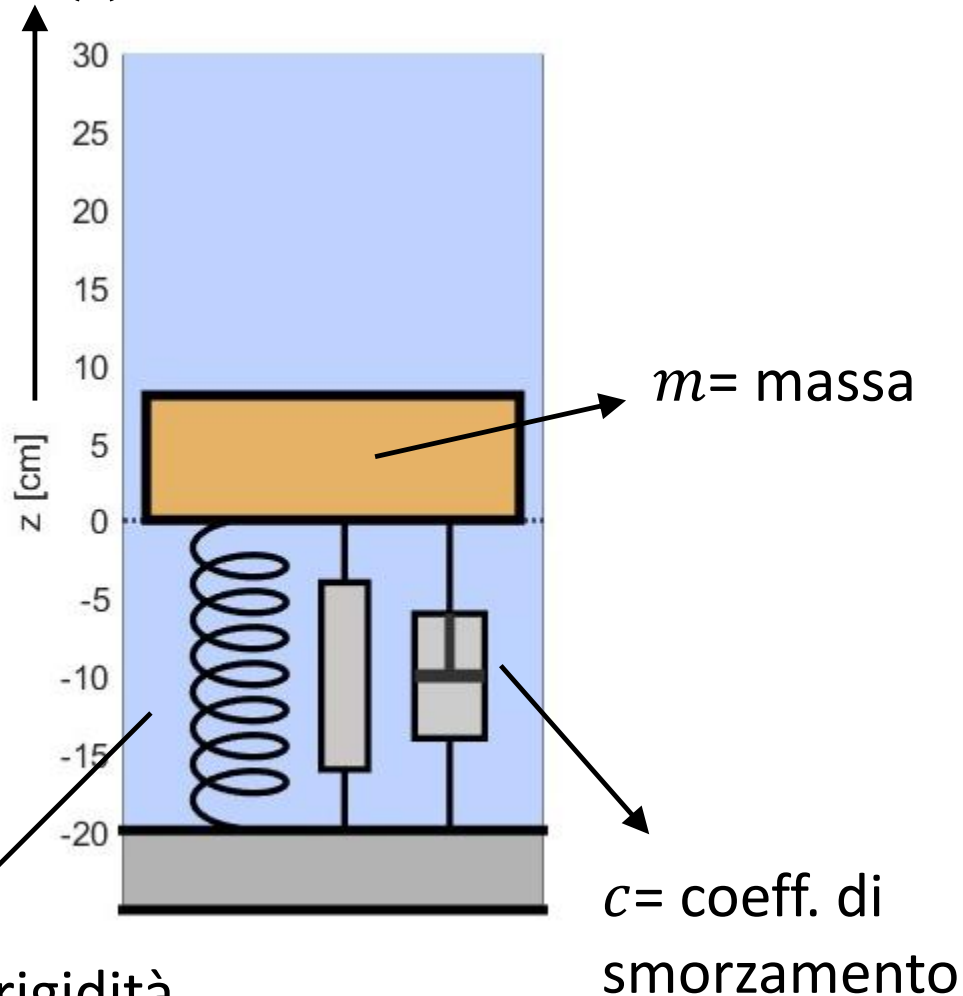
HP: veicolo fermo, considero solo effetto dell'attuatore

Modellazione dinamica verticale: sistema massa molla smorzatore

Trascuro l'effetto del terreno, sarà considerato come un disturbo in fase di controllo

k = costante di rigidità

$z(t)$ = spostamento verticale



Funzione di trasferimento

Equazione differenziale che descrive la dinamica verticale del sistema (bilancio delle forze)

$$m\ddot{z}(t) + kz(t) + c\dot{z}(t) - F(t) = 0$$

Rappresentazione di stato

$$x(t) = \begin{bmatrix} x_1(t) = z(t) \\ x_2(t) = \dot{z}(t) \end{bmatrix} \quad \begin{array}{ll} \text{posizione} & \text{forza attuatore} \\ \text{velocità} & u(t) = F(t) \end{array}$$

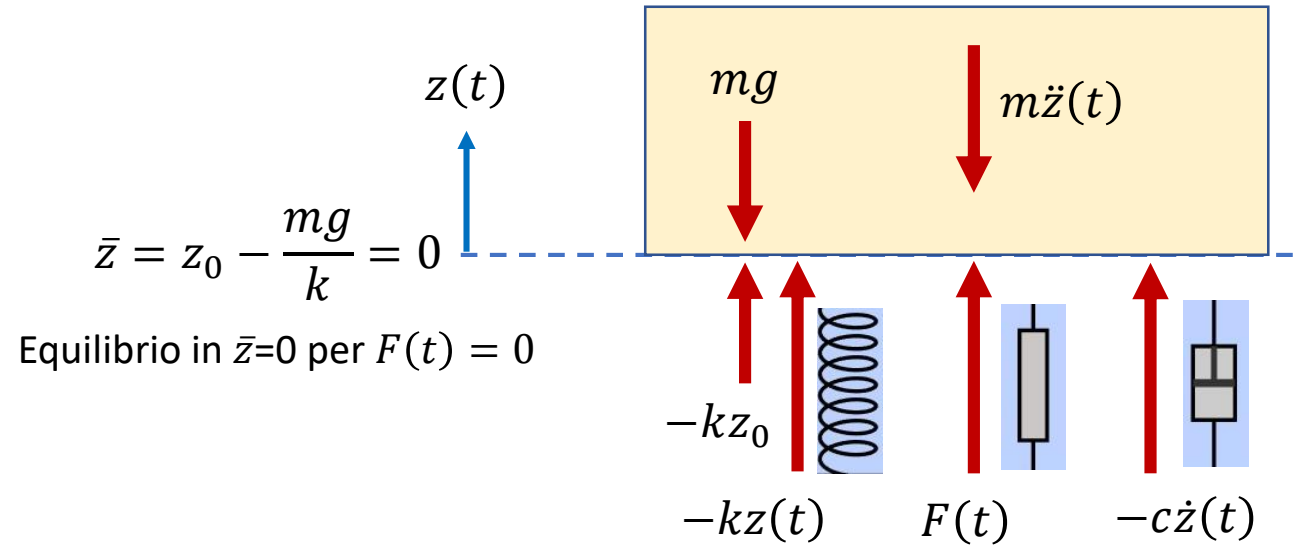
$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= -\frac{k}{m}x_1(t) - \frac{c}{m}x_2(t) + \frac{1}{m}u(t) \\ y(t) &= x_1(t) \end{aligned}$$

Funzione di trasferimento

$$\frac{Y(s)}{U(s)} = G(s) = \frac{1/m}{s^2 + \frac{c}{m}s + \frac{k}{m}}$$

Poli $s_{1,2} = \frac{-c}{2m} \pm \frac{1}{2m} \sqrt{c^2 - 4km}$

Se $c^2 < 4mk$ ho una coppia di poli complessi coniugati con $\text{Re}(s_{1,2}) < 0$

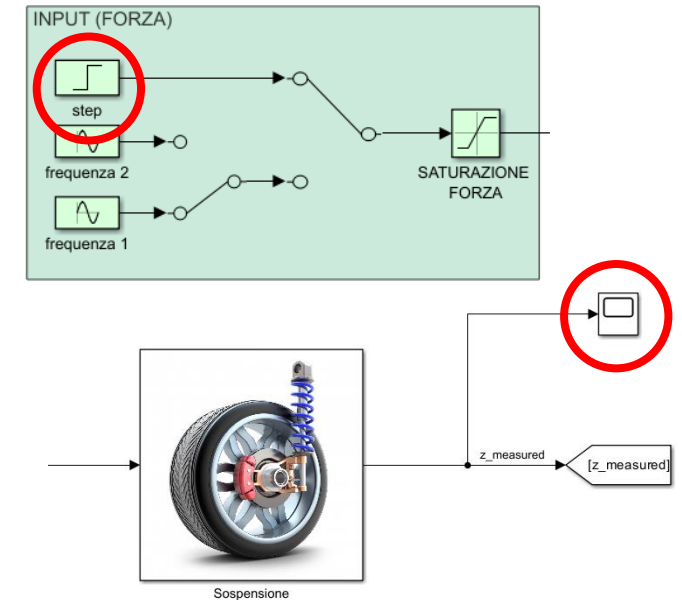
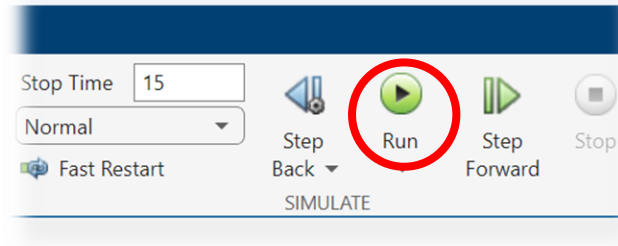


Esperimento 1: risposta allo scalino

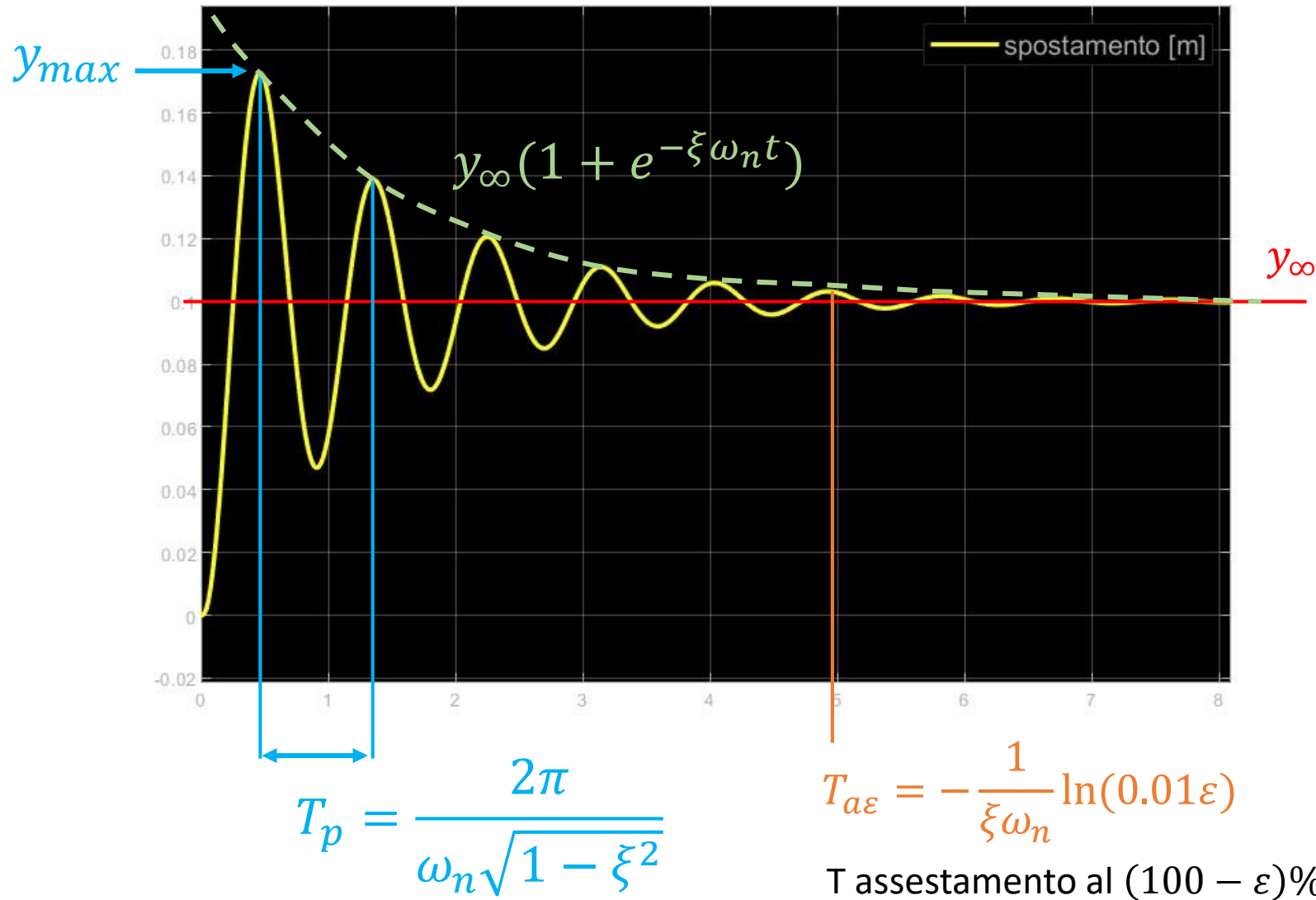
SCOPO

Ottenere i parametri che descrivono la funzione di trasferimento $G(s)$ del modello del sistema a partire dalla risposta a scalino del sistema stesso.

- Aprire modello Simulink del sistema fornito
- Selezionare lo scalino come ingresso al sistema con doppio clic su Manual Switch
- Fare doppio clic sul blocco in ingresso per visualizzare i parametri dello scalino
(es: $A = 100$)
- Simulare la risposta premendo il tasto Run
(Stop Time = 15)
- Al termine della simulazione aprire lo Scope per visualizzare la risposta $y(t)$ allo scalino



Esperimento 1: risposta allo scalino



Coppia poli c.c.

$$G(s) = \frac{\mu\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

Sovraelongazione massima percentuale

$$S_{\%} = 100 \frac{y_{max} - y_\infty}{y_\infty}$$

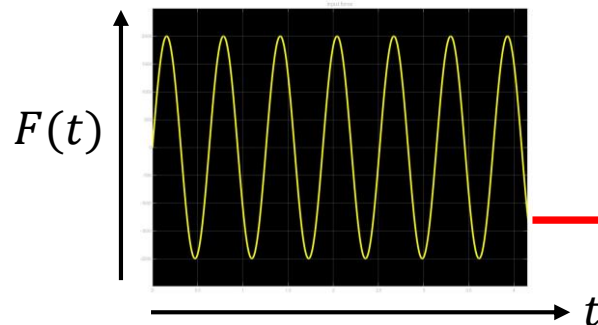
$$S_{\%} = 100 e^{\frac{-\xi\pi}{\sqrt{1-\xi^2}}}$$

Esperimento 2: risposta in frequenza

SCOPO

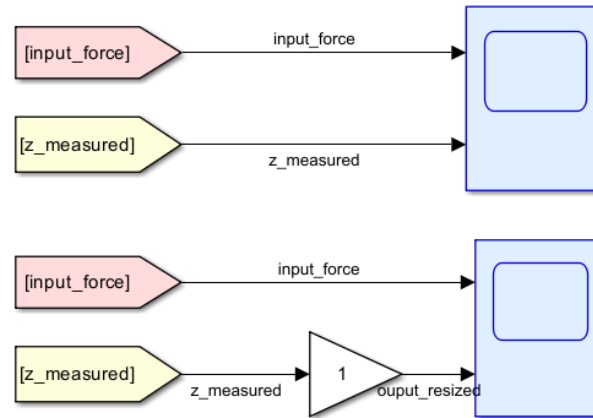
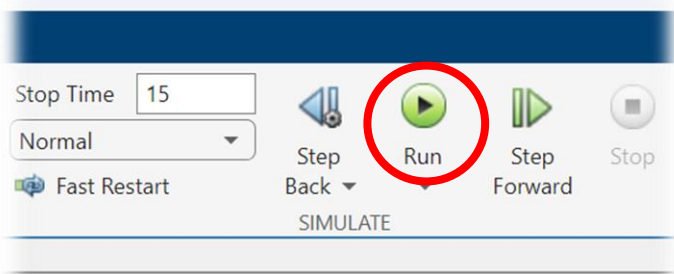
Ottenere la funzione di trasferimento finale osservando le discrepanze tra le risposte in frequenza del modello iniziale e le risposte del sistema reale

- Aprire lo script Matlab fornito
- Inserire i parametri ottenuti dal primo esperimento: μ , $S\%$, Ta
- Ricavare ω_n e ξ in funzione degli altri parametri
- Scrivere la funzione di trasferimento $G(s)$
- Fare clic su Run per ottenere le variabili nel workspace



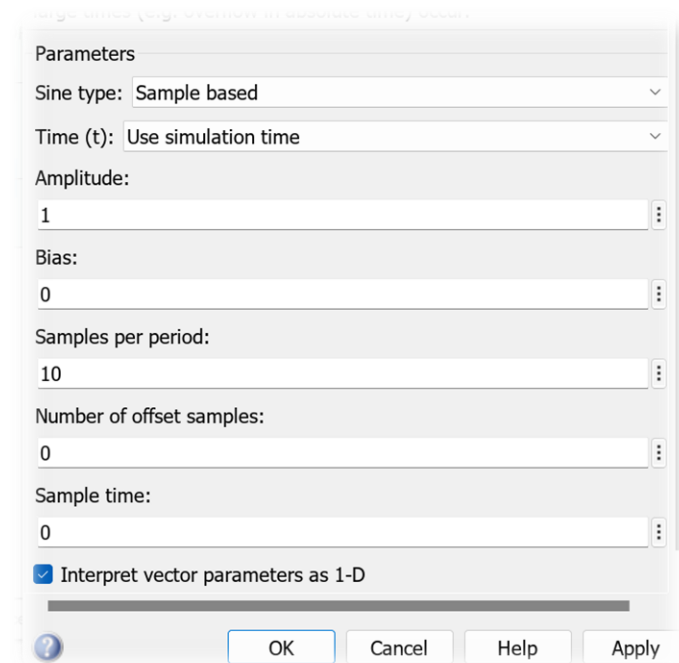
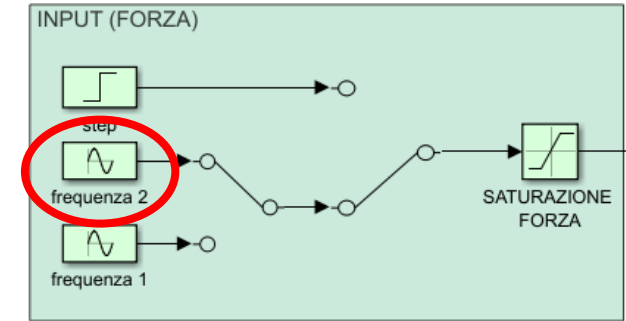
Esperimento 2: risposta in frequenza

- Aprire modello Simulink del sistema fornito
- Selezionare il segnale sinusoidale come ingresso al sistema con doppio clic su Manual Switch
- Fare doppio clic sul blocco in ingresso per visualizzare e modificare i parametri della sinusoide (pulsazione w)
- Simulare la risposta premendo il tasto Run (Stop Time = 15)
- Al termine della simulazione aprire lo Scope per confrontare la risposta del sistema con il modello ottenuto dalla risposta a scalino

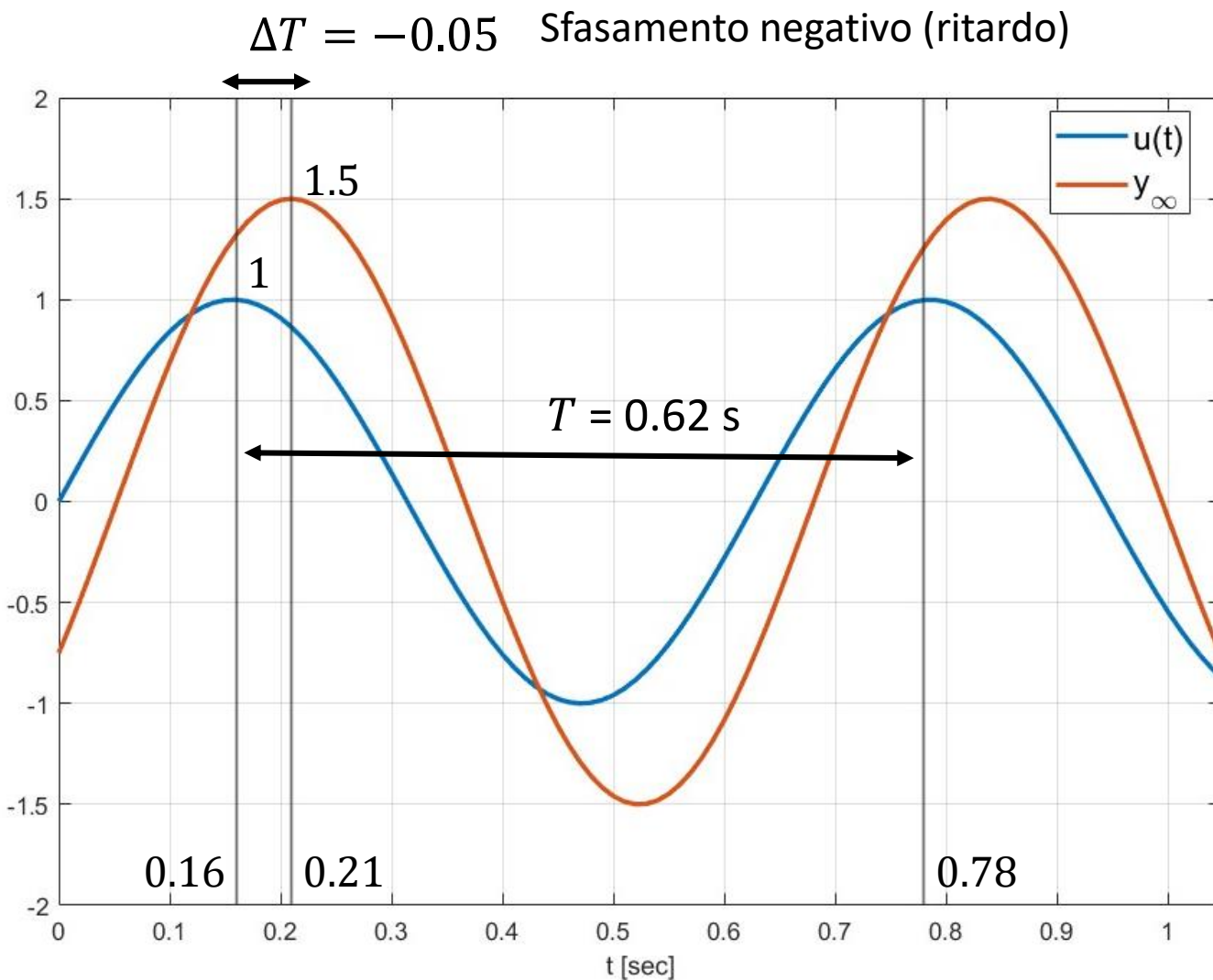


Scope
segnali
separati

Scope segnali
sovrapposti
(modificare il guadagno)



Esperimento 2: risposta in frequenza



Teorema fondamentale della risposta in frequenza

$$u(t) = U \sin(\omega_0 t) \quad u(t) \rightarrow \boxed{G(s)} \rightarrow y(t)$$
$$y_\infty(t) = |G(j\omega_0)| U \sin(\omega_0 t + \arg G(j\omega_0))$$

Esempio

$$u(t) = \sin(10t) \quad y_\infty(t) = 1.5 \sin(10t - \pi/6)$$

A partire dall'analisi dei segnali

Frequenza $\omega = \frac{2\pi}{T} \cong \frac{2\pi}{0.62} \cong 10 \text{ rad/s}$

modulo

$$|G(j10)| = \frac{1.5}{1} = 1.5$$

fase

$$\arg G(j10) = 360 \frac{\Delta T}{T} \cong 360 \frac{-0.05}{0.62} \cong -30^\circ \cong 0.52 \text{ rad}$$