



DeXpl.LAB Manual: Free Response

Summary

1. Object:.....	2
2. Purpose:.....	2
3. Apparatus:	2
4. Model and parameters estimation	3
5. Procedure	4
6. Analysis	8
6.1. Analisiacc.m	9
6.2. Visual comparison between the two accelerometer signals and coordinates display	10
6.1. Estimation of the damping ratio ζ	11
6.2. Estimation of the natural frequency of the system ω_n	13



1. Object:

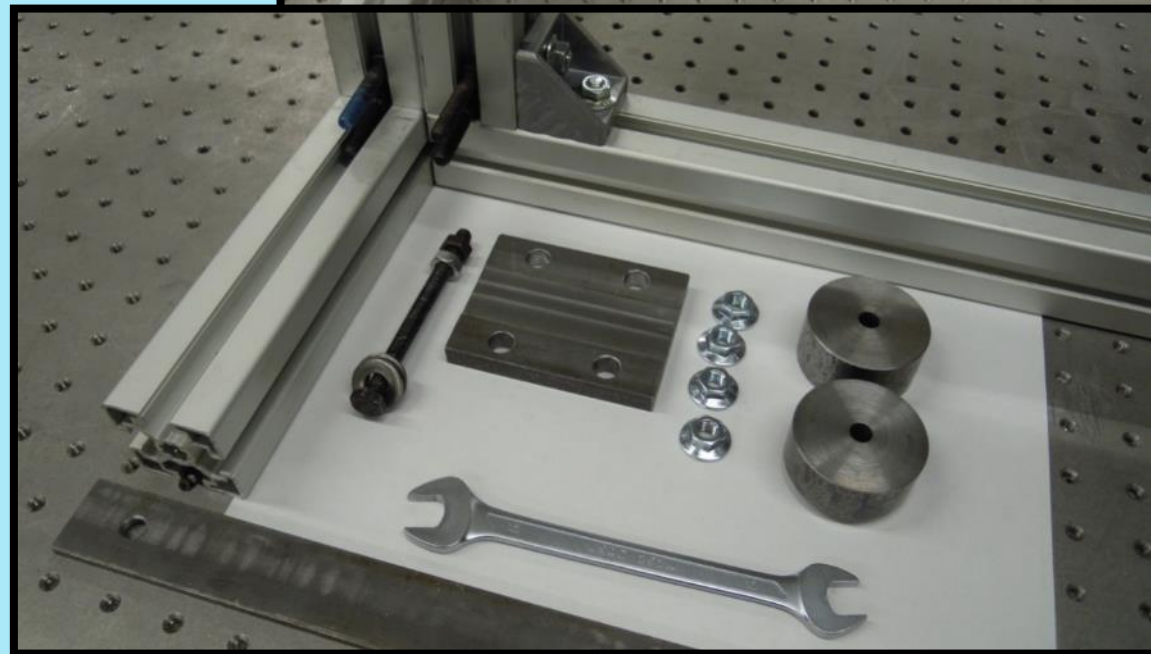
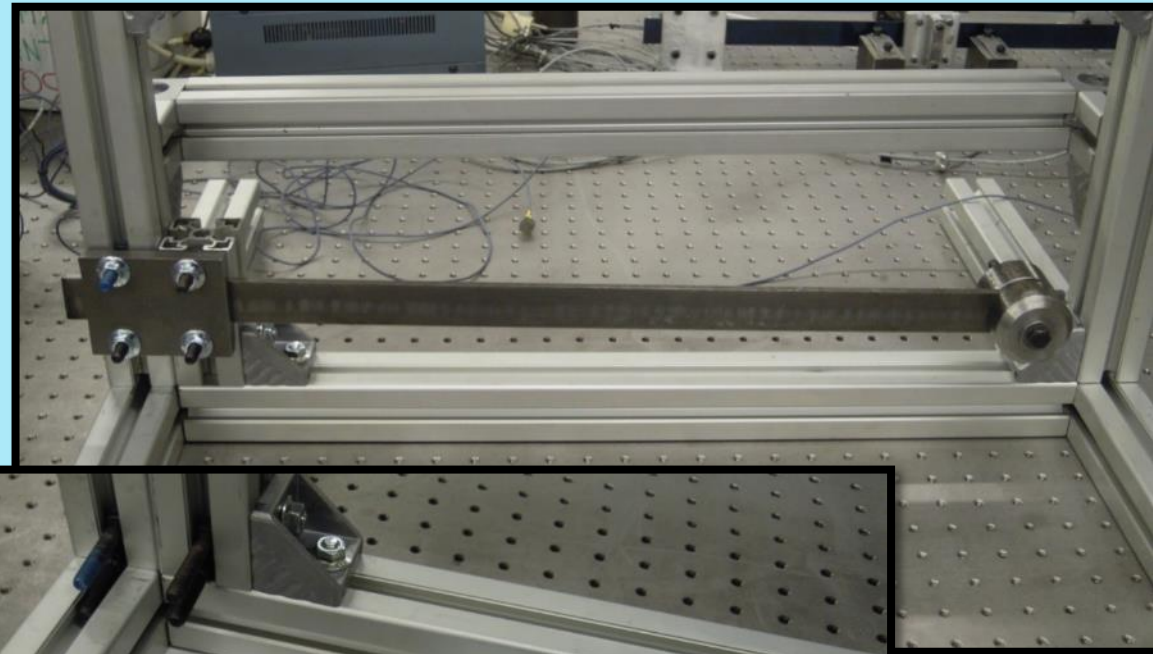
Study of a single degree of freedom vibrating system through experimental acceleration measurements

2. Purpose:

Estimate of the natural frequency of the system (ω_n) and the damping factor (ζ) varying the characteristic parameters (length of the beam and concentrated mass)

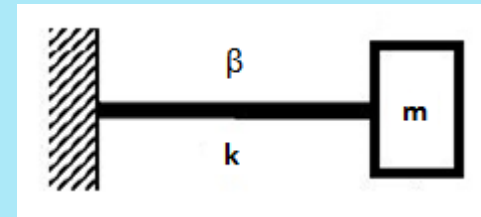
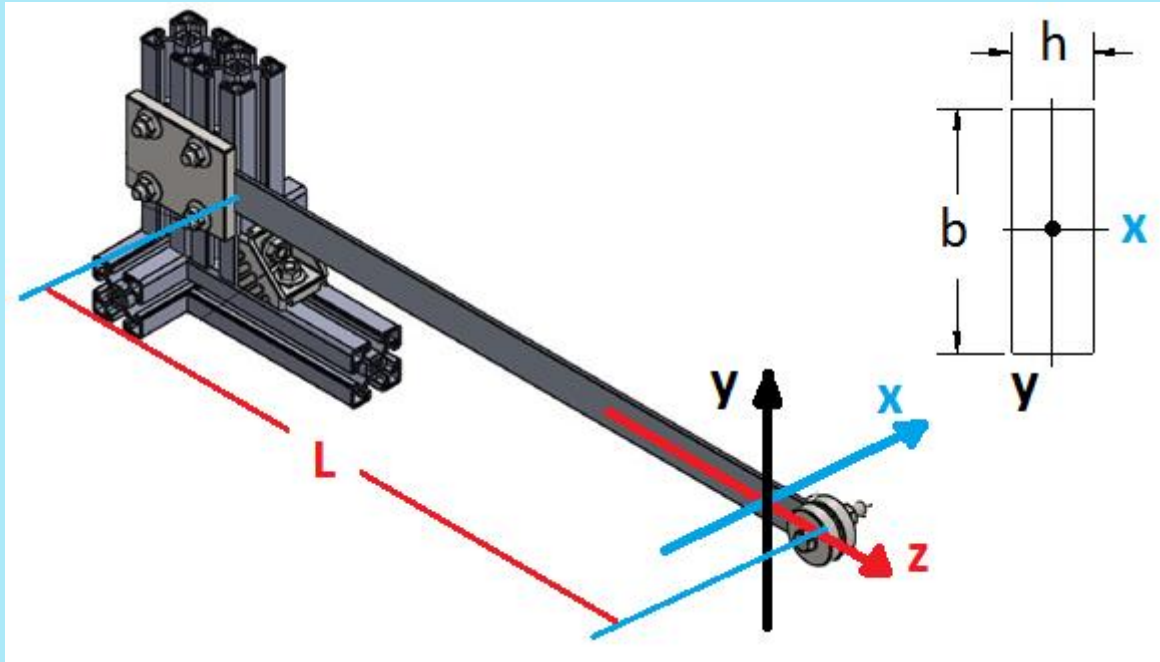
3. Apparatus:

- Steel and Aluminum beams
- Slotted plate
- 4 nuts for M8 hammer head screws
- 13 mm wrench
- Sample masses ($M_{Al}=0.1\text{kg}$
 $M_{Fe}=0.23\text{kg}$)
- 1 long screw, 2 nuts and 2 washers to fix the sample masses
- 2 single-axis accelerometers





4. Model and parameters estimation

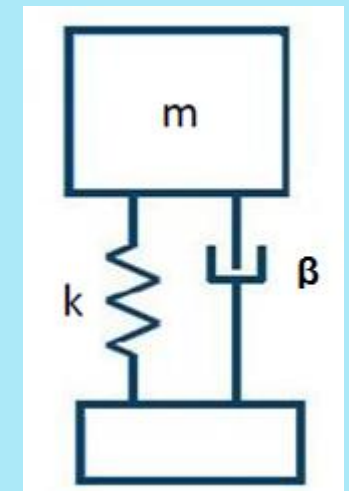


$$I = I_y = \frac{b h^3}{12}$$

$$k = \frac{3 E I}{L^3}$$

$$E_{Fe} = 210000 \text{ MPa}$$

$$E_{Al} = 64000 \text{ MPa}$$



m : exactly corresponds to the used sample mass

β : viscous damping coefficient (unknown)

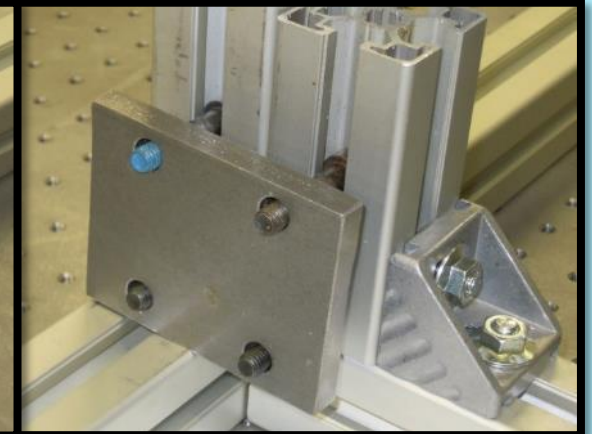
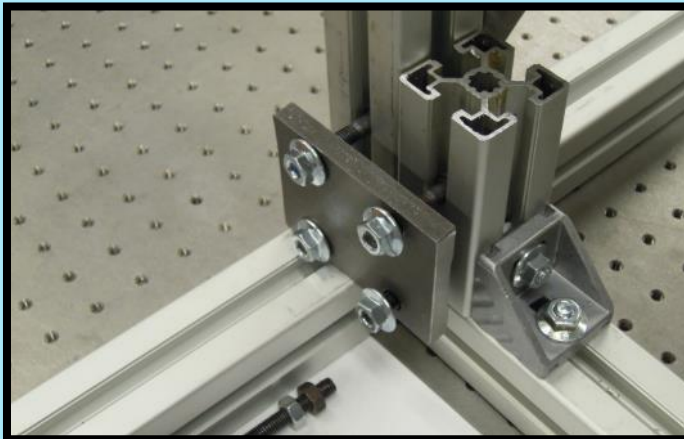
k : beam flexural stiffness

$$m \ddot{x} + \beta \dot{x} + k x = 0$$



5. Procedure

- 1) Place the plate inserting the screws protruding from the structure into the holes; remember to turn the hammer head so as to secure the screws.



- 2) Tighten the nuts of the screws to secure the plate; DO NOT TIGHTEN completely, leave sufficient space to allow the passage of the beam.

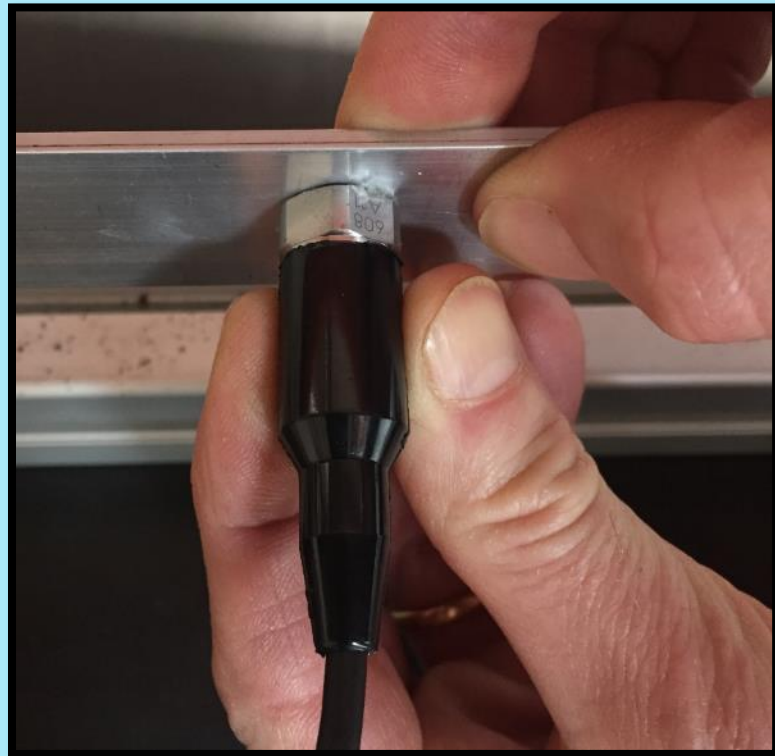
- 3) Insert the beam between the plate and the structure.
- 4) Tighten the nuts using the wrench, checking that the beam is horizontal.



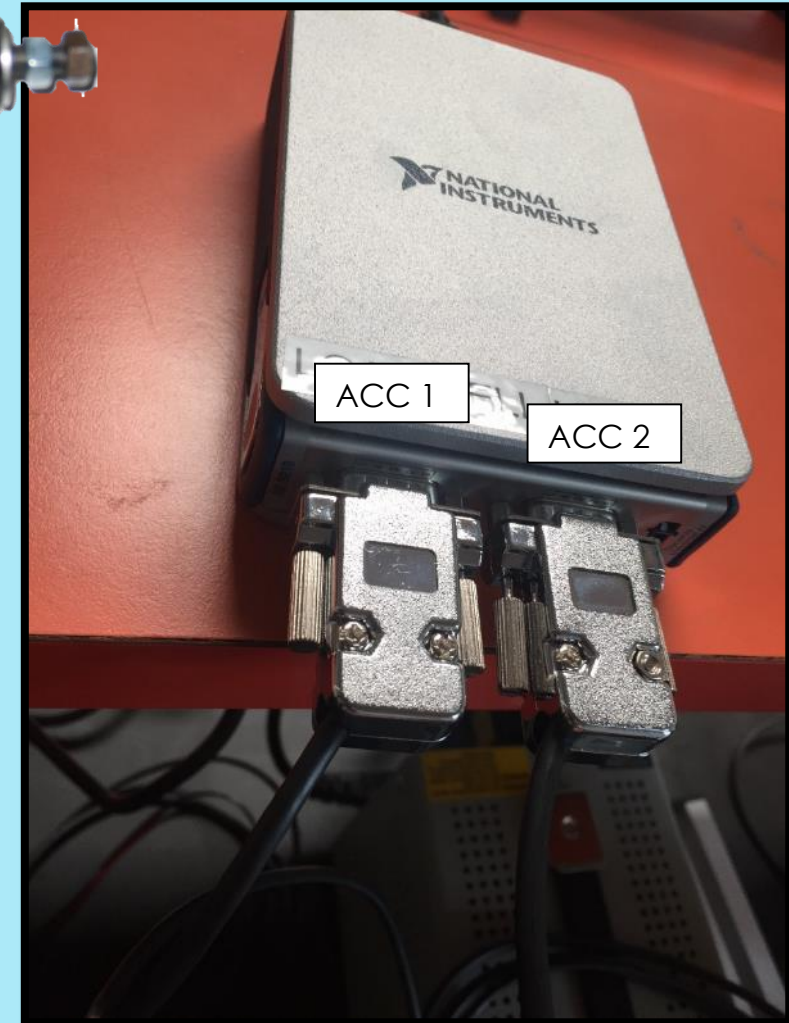


5) Assemble and mount the sample mass.

6) Attach the accelerometers to the beam using the appropriate wax as shown in the figure below.

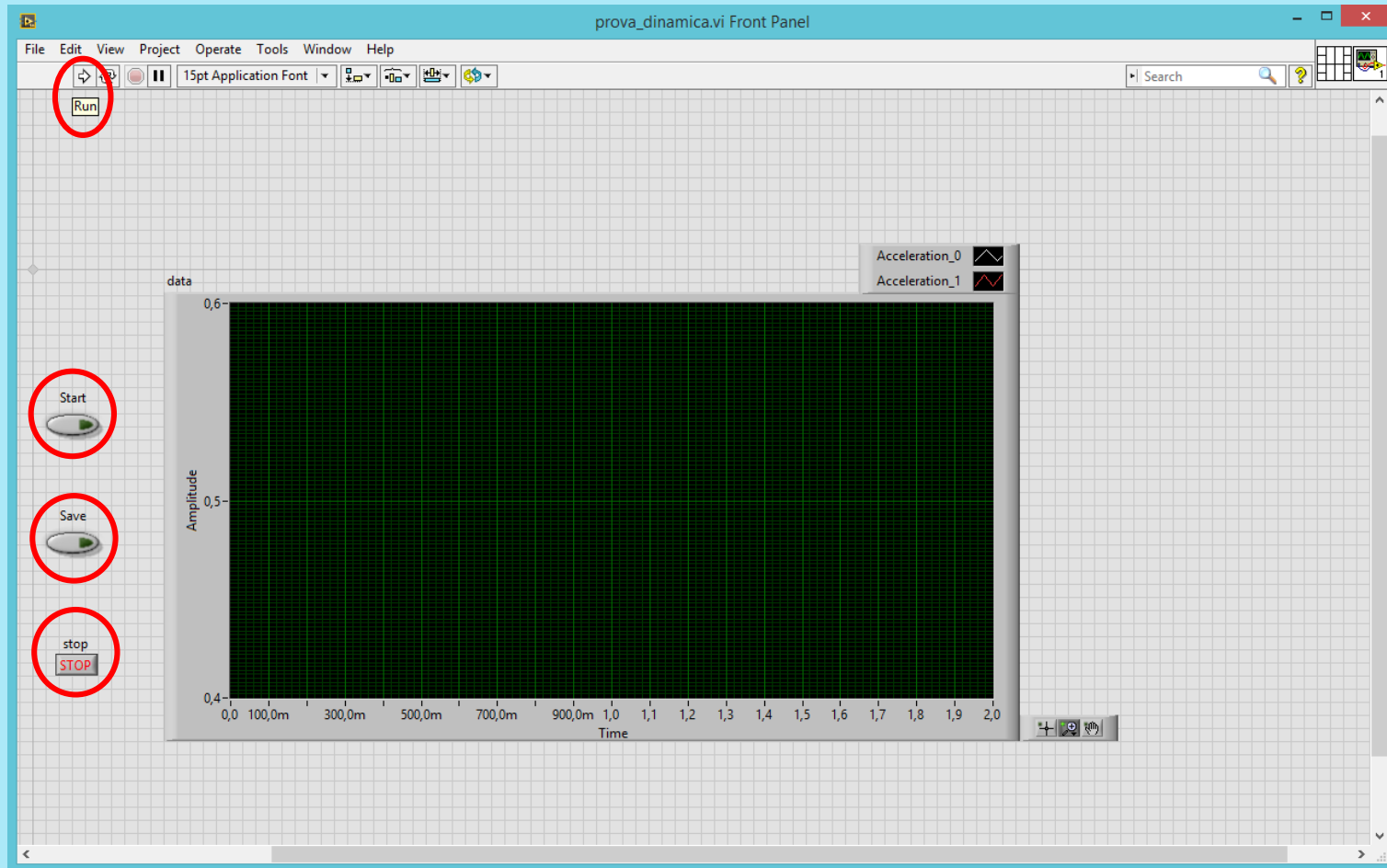


7) Connect the accelerometers to the digital acquisition card.





8) Start the LabVIEW acquisition interface: **prova_dinamica.vi**, located on the desktop or in the folder C: \ temp

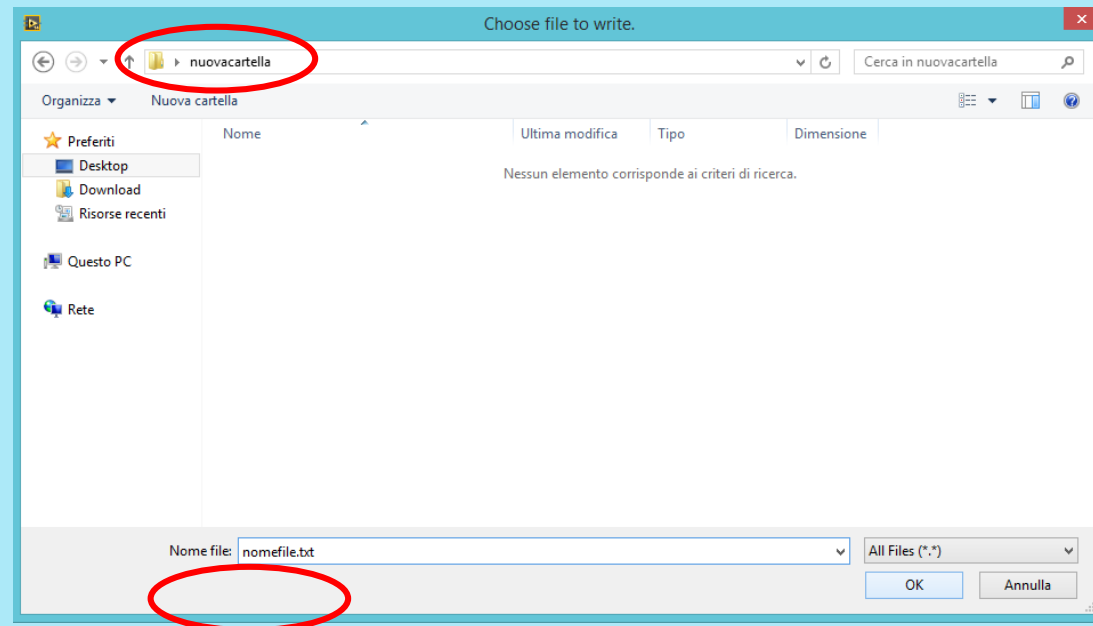
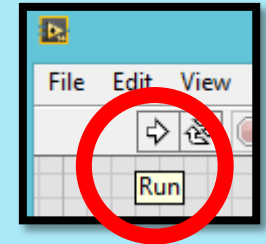




POLITECNICO DI TORINO:

BANCO PROVA DeXpl.LAB

- To start the acquisition program, press **RUN** →
- To start the acquisition, press **Start**; the second operator will then move the mass a few centimeters (approximately 2 cm) to excite the system oscillation. It is safe to wait a few moments between the Start button is pressed and the excitement of the system; **press the button once and wait 20 s for the displaying!**
- To save the acquisition, click **Save**:
Both the destination folder and the file name must be chosen.
It is advisable to generate a new folder e.g. **nuovacartella** and name the file according to the format **filename.txt**.
(The "Prepara Prova" file on the desktop automatically creates a new folder, with a matlab file inside, use that one!).
- To end the program, press **Stop** and close the LabVIEW window.

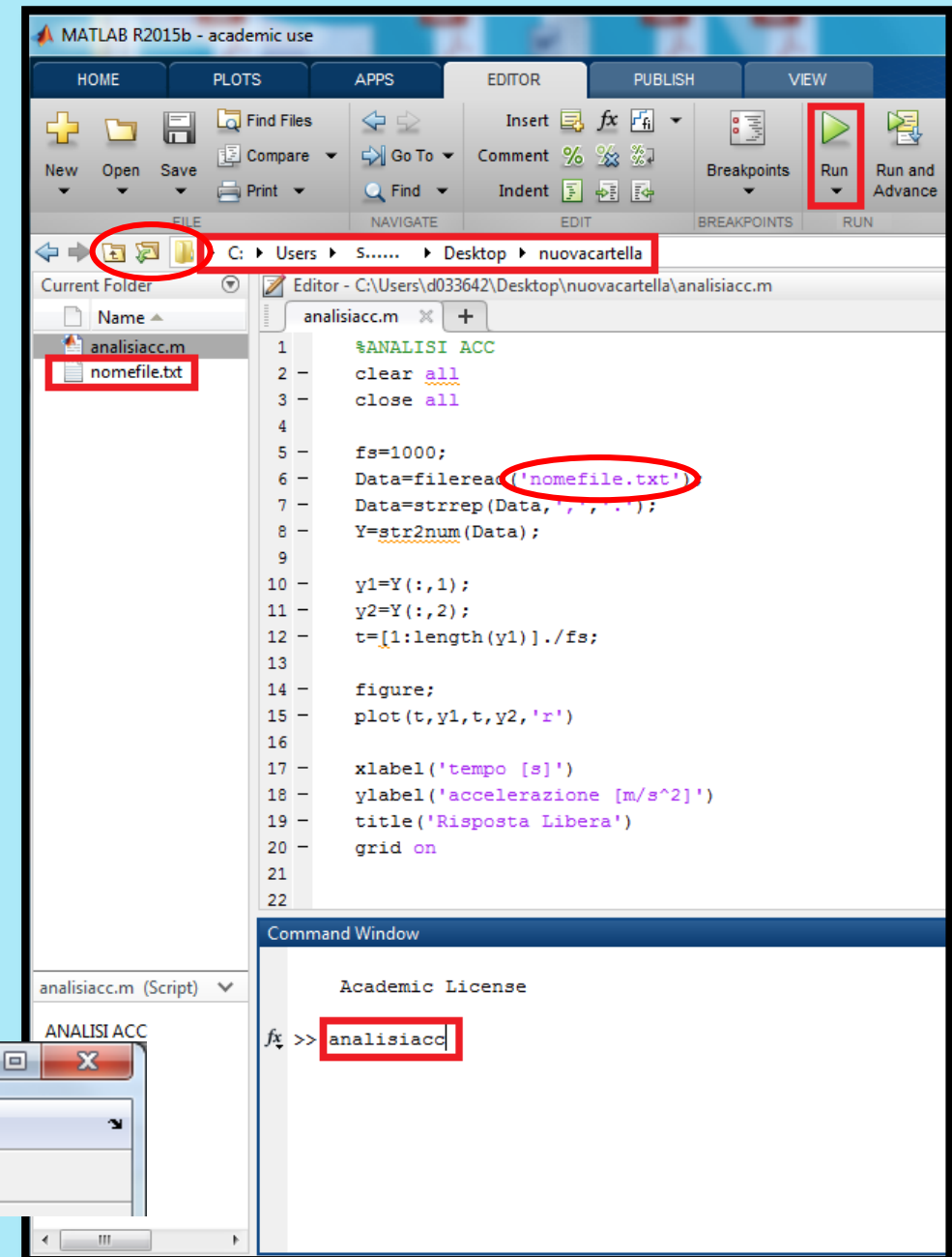
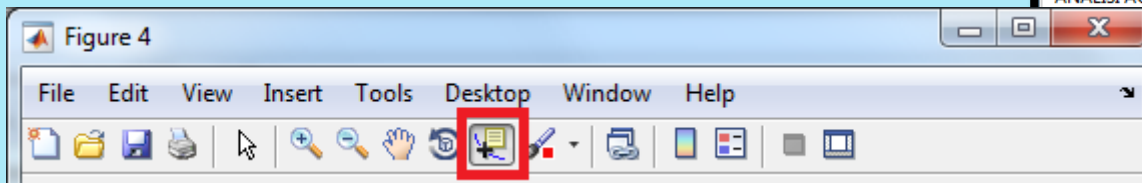




6. Analysis

Using the software for numerical computation **MATLAB**, you can graph the saved data, and perform further analysis.

1. Start Matlab
2. Select the work folder (e.g. **nuovacartella**)
3. If not present, copy the **analisiacc.m** file (C:\temp) in the work folder and run it.
4. Correct the name of the acquired signals file (e.g. **nomefile.txt**)
5. Run **analisiacc.m** typing its name in the Command Window or by pressing
6. To view the acceleration values on the graph, use the command *Data Cursor* shown in the figure below and select a point on the graph; the x and y coordinates will appear in a box (*datatip*).





6.1. *Analisiacc.m*

Below is reported the Matlab program to plot the acquired accelerations.

```
%ANALISI ACC

Clear all    % clears all the variables in the Workspace
Close all    % closes all the graphics previously created
fs=2048;     % sampling frequency: Number of measurements acquired in 1 second
Data=fileread('nomefile.txt'); % loads the txt file in the Data string variable
Data=strrep(Data,',','.'); % replaces the decimal comma with the point
Y=str2num(Data); % converts the string to numbers: a matrix composed of 2 columns
                % in which the acceleration values over will appear
y1=Y(:,1)-mean(Y(:,1)); % all the values (:) of the first column (acc1) are copied to the
                % Y1 variable (column vector)
y2=Y(:,2)-mean(Y(:,2)); % all the values of the second column (acc2) are copied to the vector y2
t=[1:length(y1)]/fs; % generates the time axis given the sampling frequency:
                % the vector 1 to the size of the vector y1
                % is multiplied by the dt among two following acquisitions: dt = 1 / fs
figure; % creates a new empty figure
plot(t,y1,t,y2,'r') % plots the acceleration values:
                % "r" changes the color of the second curve (red)
xlabel('time [s]') % x axis label
ylabel('acceleration [m/s^2]') % y axis label
title('Free Response') % title
legend('acc1','acc2') % legend
grid on % creates a dotted grid
```



6.2. Visual comparison between the two accelerometer signals and coordinates display

It is now possible to proceed to a graphical comparison of the acceleration trends in two measuring points, respectively, highlighted in blue and red.

It is easy to note that the two accelerations are perfectly in phase, and share the same period; only the amplitude is different because of the different observation point of the phenomenon.

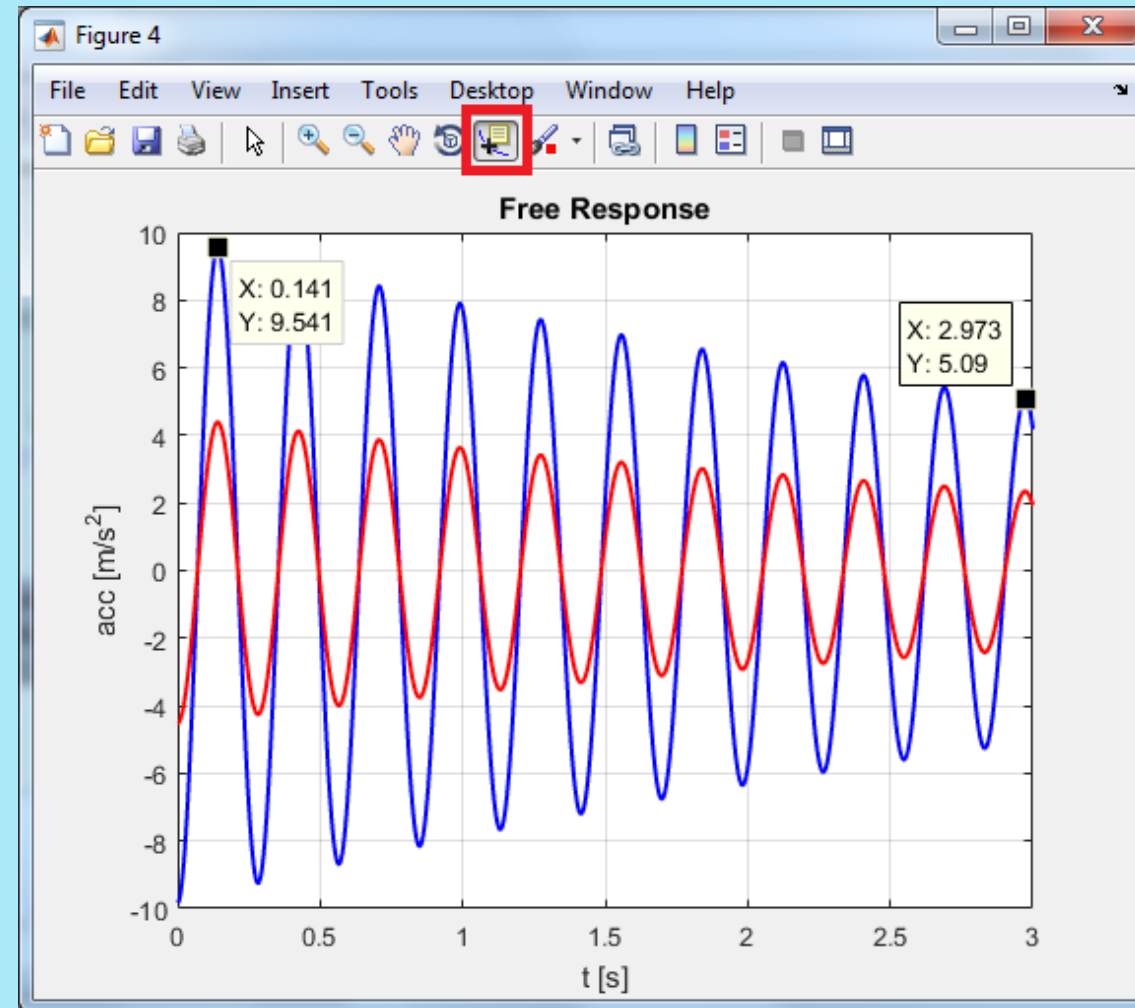
To view the acceleration values on the graph, use the command *Data Cursor* highlighted in the figure.

It is needed to read the values for at least two peaks distant a number n of periods (e.g., 10):

t_1, a_1, t_n, a_n will be then found.

When the damping is small in fact, the difference between two successive peaks ($n = 1$) could be small, and the calculations proposed later would be affected by a considerable error; it is therefore advisable to use a sufficiently large n , so that the difference among a_1 and a_n will be appreciable.

The same applies to the ω_d estimation through t_1 e t_n .





6.1. Estimation of the damping ratio ζ

For underdamped systems, it's possible to get an estimate of the damping ratio ζ by calculating the logarithmic decrement δ (ratio of the logarithm of 2 peaks).

$$a_1 = X_{acc} e^{-\zeta \omega_n t}$$

$$a_n = X_{acc} e^{-\zeta \omega_n (t+nT)}$$

$$\frac{a_1}{a_n} = \frac{e^{-\zeta \omega_n t}}{e^{-\zeta \omega_n (t+nT)}} = e^{\zeta \omega_n nT}$$

the logarithmic decrement δ is therefore:

$$\delta = \ln \frac{a_1}{a_n} = \zeta \omega_n nT$$

Remembering that

$$T = \frac{1}{f} = \frac{2\pi}{\omega_d}$$

and $\omega_d = \omega_n \sqrt{1 - \zeta^2}$ it is possible to get:

$$\delta = 2\pi n \frac{\zeta}{\sqrt{1 - \zeta^2}}$$

$$\zeta = \frac{\delta}{\sqrt{(2\pi n)^2 + \delta^2}}$$



In this example, for **$n=10$** (the second peak is 10 periods far from the first):

$$\delta = \ln \frac{a_1}{a_n} = \ln \frac{9,54}{5,09} = 0,628$$

In Matlab environment:

```
delta=log(9.541/5.09)
```

```
n=10
```

$$\zeta = \frac{\delta}{\sqrt{(2\pi n)^2 + \delta^2}} = \frac{0,628}{\sqrt{(20\pi)^2 + 0,628^2}} = 0,01$$

In Matlab environment:

```
zeta=delta/(((2*n*pi)^2+delta^2)^0.5)
```

NB. `delta=...` generates a new variable containing the result; this can be recalled later on, as shown in the second computation. It would be even possible to use `ans` variable, that recalls the last computed value, or to copy the approximated value of 0,628 in the second formula, but this increases the error.



6.2. *Estimation of the natural frequency of the system ω_n*

The natural frequency is calculated by estimating the damped period $= \frac{1}{f} = \frac{2\pi}{\omega_d}$:

$$\omega_d = \frac{2\pi}{T}$$

$$\omega_n = \frac{\omega_d}{\sqrt{1 - \zeta^2}} = \frac{2\pi}{T\sqrt{1 - \zeta^2}}$$

In the example, for **$n=10$**

$$T = \frac{t_n - t_1}{n} = \frac{2,973 - 0,141}{10} = 0,2832 \text{ s}$$

NOTE: with $n = 10$, T is the mean value over 10 periods; This increases the accuracy of the estimate.

$$\omega_d = \frac{2\pi}{T} = \frac{2\pi}{0,2832} = 22,1864 \text{ rad/s}$$

$$\omega_n = \frac{\omega_d}{\sqrt{1 - \zeta^2}} = \frac{22,186}{\sqrt{1 - 0,01^2}} = 22,1875 \text{ rad/s}$$

In Matlab environment:

$$T = (2.973 - 0.141) / n$$

$$Wd = 2 * \pi / T$$

$$Wn = Wd / (1 - \text{zeta}^2)^{0.5}$$