UNIVERSITÀ DEGLI STUDI DI GENOVA

Robotics Engineering

AY. 2021/2022

Robot dynamics and control

first assignment’s report:

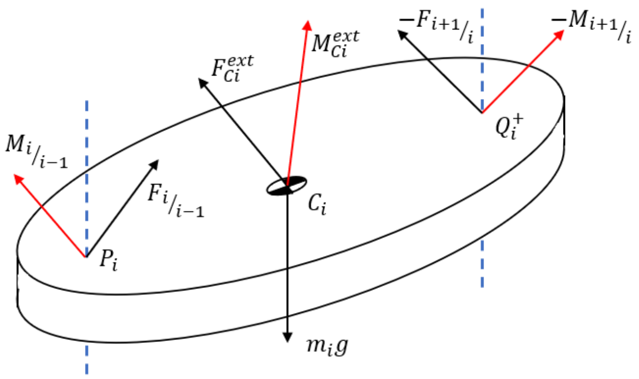
Static equilibrium of serial manipulators

Introduction:

This report's purpose is to explain the basics theory concepts used in the static equilibrium analysis of a mechanical chain, how it has been implemented in a MATLAB program, using also the data obtained from the Cad model of the manipulator, and a brief analysis of the results.

Theory concepts:

Let we consider a generic body i in the space that belongs to a mechanical chain, so which is connected with other bodies by a revolute joint or a prismatic joint:



We can divide all the forces and moment acting on the body in four main groups:

- The gravity's force of the body $m_i g$, acting on the center of mass of the body.
- The forces and moments that the body exchange with the previous one through the joint $F_{i/_{i-1}}, M_{i/_{i-1}}$ thanks to the mechanical constraint of the joint and the actuation action, acting on the connection point P_i.
- The forces and moments that the body exchange with the following one through the joint $F_{i+1/_i}, M_{i+1/_i}$ thanks to the mechanical constraint of the joint and the actuation action, acting on the connection point Q_i^+.
- The external forces and moments acting on the body reconducted to the center of mass $F_{Ci}^{ext}, M_{Ci}^{ext}$, acting on the center of mass C_i.

To obtain the static equilibrium of each the generic bodies and so of the manipulator we can introduce the recursive Newton-Euler algorithm: since we are looking for the equilibrium of the body the following condition must be satisfied:

$$\begin{cases} F_{Ci}^{tot} = 0 \\ M_{Ci}^{tot} = 0 \end{cases} \begin{cases} F_{Ci}^{ext} + m_i g + F_{i/_{i-1}} - F_{i+1/_i} = 0 \\ M_{Ci}^{ext} + M_{i/_{i-1}} - M_{i+1/_i} + r_{Pi/_{Ci}} \times F_{i/_{i-1}} - r_{Q_i^+/_{Ci}} \times F_{i+1/_i} = 0 \end{cases}$$

Where:

$$Q_i^+ = \begin{cases} Q_i & if\ i \in RJ \\ Q_i + K_{i+1} q_{i+1} & if\ i \in TJ \end{cases}$$

Starting from these equations that describes the equilibrium of the generic body we can compute the actuation forces and moments acting on the generic link, reconducting the rotational equilibrium for the moments at the point P_i instead of the center of mass C_i, and starting from the last body (n) with the set of equations:

$$\begin{cases} F n_{/_{n-1}} = -m_n g - F_{Cn}^{ext} \\ M n_{/_{n-1}} = -M_{Cn}^{ext} - r_{Pn_{/_{Cn}}} \times F n_{/_{n-1}} \end{cases}$$

Then we can pass to the following link where we need to take in account also the reaction forces and moments exchanged between the link n and the link $n-1$ which are the ones we've just evaluated:

$$\begin{cases} F_{n-1_{/_{n-2}}} = -m_{n-1} g - F_{Cn-1}^{ext} + F n_{/_{n-1}} \\ M_{n-1_{/_{n-2}}} = -M_{Cn-1}^{ext} - r_{Pn-1_{/_{Cn-1}}} \times (-F_{Cn-1}^{ext} - m_{n-1} g) + r_{Q_{n-1}^{+}{}_{/_{Pn-1}}} \times F n_{/_{n-1}} + M n_{/_{n-1}} \end{cases}$$

Then we can evaluate the general actuation forces and moments on each link with the equations:

$$\tau_i = \begin{cases} F_{i_{/_{i-1}}} \cdot K_i \quad if\ i \in TJ \\ M_{i_{/_{i-1}}} \cdot K_i \quad if\ i \in RJ \end{cases}$$

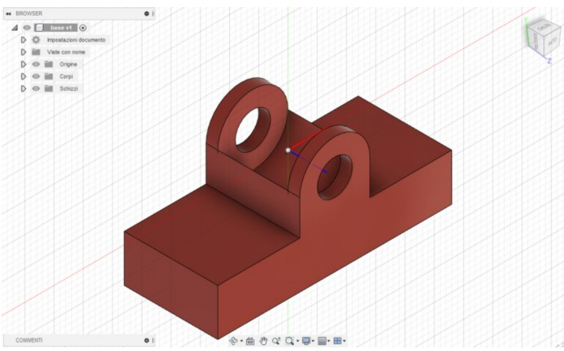
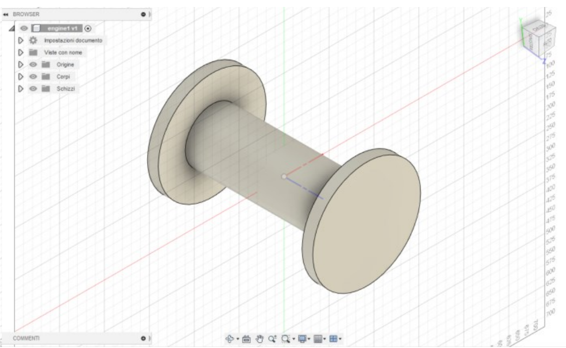
Pay attention: all the quantities in the formulas are geometric vectors and so to use these equations we've to project them on a reference frame, in our case the best option is to choose the local reference frame of each link.

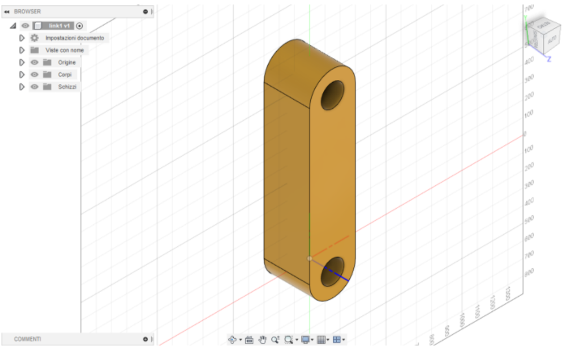
Some notes about the cad model:

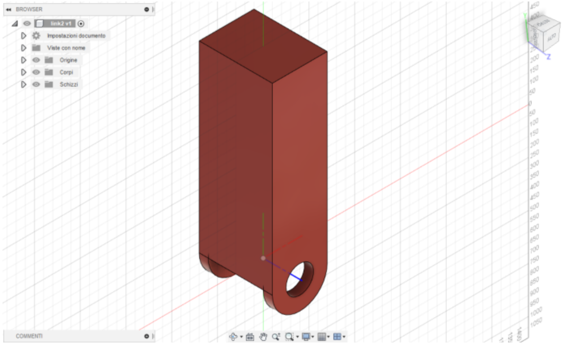
To correctly use the program there must be taken some attention during the link modelling, it's very important to set the local reference frame of the drawing space for each link in the joint point defined above as P_i, except for the base in which it should be in the joint point defined above as Q_i^+; with the y axis oriented along the link.

In this way when the program fulfills the center of mass's vector from the data file the actual position will be directly the vector $r_{Pi_{/_{Ci}}}$.

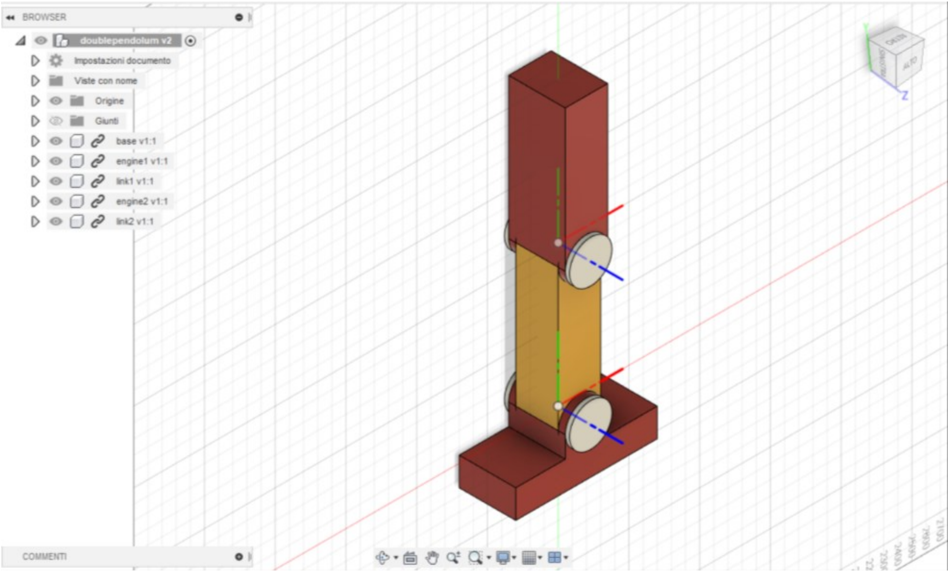
Let we see some examples:





With this convention the actual zero configuration of the joint, which means the configuration in which all the joint angles are zero, is the following one:

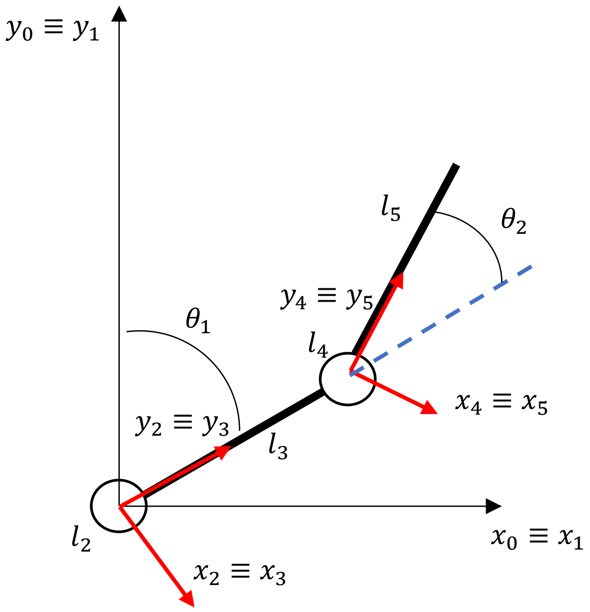


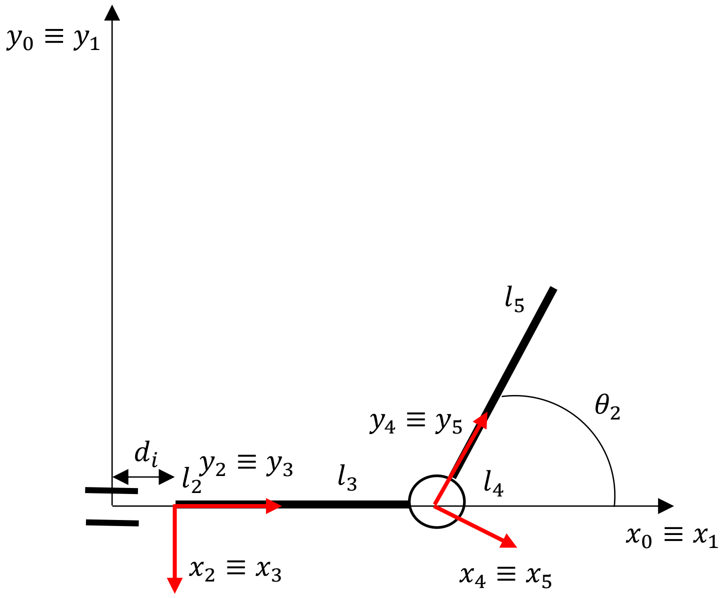
This must be taken in account during the configuration setting for each exercise.

Another consideration should be done on the number of model, to obtain a correct fulfillment of the vector of the center of mass and of the masses all the parts must be modeled (for example even if the engines in this exercise are actually the same they must be modeled two times) moreover, the model has also the material attribute (steel) for each body, which will affect the results of the exercises.

Exercises settings and results:

Before starting to analyze the exercises and the relative results some word about the program modeling of the problem must be spend:





The program is designed to be as general as possible, in it any object along the mechanical chain of the manipulator is consider as a link (even the engines) with his own joint type and configuration, and each link is enumerated starting with one for the base, till 5 for the second link.

Then some attention must be taken on the configuration vector q, in it the i line refer to the i joint starting from the one between base and first link. So, for example in our case:

- Line 1 refers to the joint between the base and the first engine.
- Line 2 refers to the joint between the first engine and the first link.
- Line 3 refers to the joint between the first link and the second engine.
- Line 4 refers to the joint between the second engine and the second link.

In each line we've three columns:

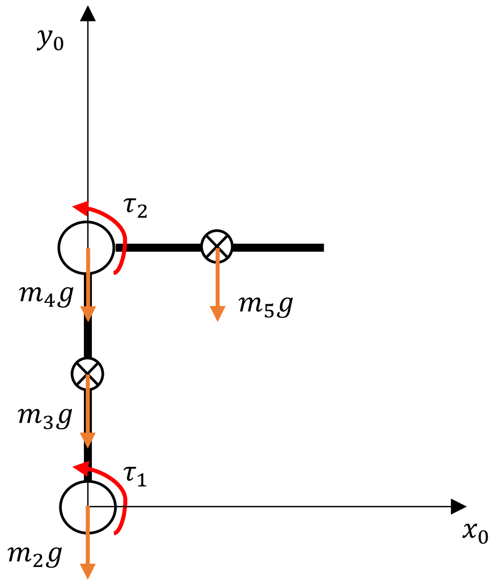
- The first column contains the configuration angle in radiant θ_i (if the joint is a revolute joint) or distance in meters d_i (if the joint is a prismatic joint) or the fixed angle between the frames θ_i in case of a rigid joint.
- The second column contains the joint's type in particular the following numbers are used to specify the joint's type:
  - 0 = revolute joint
  - 1 = prismatic joint
  - 2 = rigid joint
- The third column is used in case of a prismatic joint to specify his translational direction with an angle equal to the one between the previous link's reference frame and actual link's reference frame in radiant.

Then it's also necessary to consider the external forces, moments and gravity and how they are implemented:

- The external forces array F_ext: in it each line contains all the information of an external force needed for the following evaluation, in each line we've seven columns:
  - The first three columns contain the components of the force along the x, y, z axis of the absolute reference frame in Newton.
  - The fourth column contain the number of the link on which the force act:
    - 1 for the base.
    - 2 for the first engine.
    - 3 for the first link.
    - 4 for the second engine.
    - 5 for the second link.
  - The last three column contain the position along the x, y, z axis of the local reference frame of the link on which the force act in meters.
- The external moments array M_ext: in it each line contains all the information of an external moment needed for the following evaluation, in each line we've four columns:
  - The first three columns contain the components of the moment along the x, y, z axis of the absolute reference frame in Newton meters.
  - The fourth column contain the number of the link on which the force act:
    - 1 for the base.
    - 2 for the first engine.
    - 3 for the first link.
    - 4 for the second engine.
    - 5 for the second link.
- The gravity flag g_flag is used to turn on/off the gravity: by assigning a value of 1 the gravity is turned on, by assigning a value of 0 the gravity is turned off.

Exercise 1:

1)



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [0, 0, 0;
     0, 2, 0;
     -pi/2, 0, 0;
     0, 2, 0];
```

the gravity flag g_flag should be turned on:

```
g_flag = 1;
```

then there aren't any external force or moment so the external forces array and the external moments array must be set as:

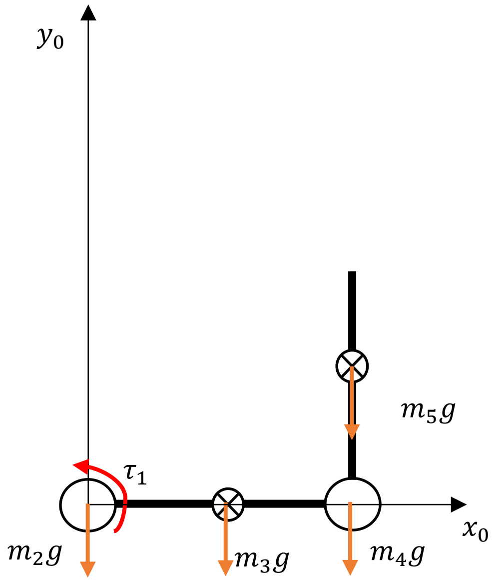
```
F_ext = [0 ,0 ,0 , 0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} \tau_1 \\ \tau_2 \end{cases} \begin{cases} 4.5553e+03\ Nm \\ 4.5553e+03\ Nm \end{cases}$$

In which the second link's weight produce a negative moment on the second joint which is balanced by a positive actuation torque.
Then the weight force of the second engine, the first link and the first engine don't produce any moment on the first joint since the link is in a vertical position and the forces pass through the first rotational joint, so the only moment experienced by the first revolute joint is the same of the second one.

2)



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [-pi/2, 0, 0;
     0, 2, 0;
     pi/2, 0, 0;
     0, 2, 0];
```

the gravity flag g_flag should be turned on:

```
g_flag = 1;
```

then there aren't any external force or moment so the external forces array and the external moments array must be set as:

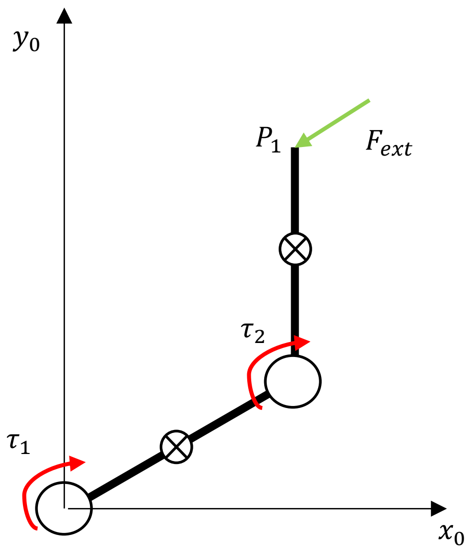
```
F_ext = [0 ,0 ,0 , 0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} \tau_1 \\ \tau_2 \end{cases} \begin{cases} 1.3317e+04\ Nm \\ 0\ Nm \end{cases}$$

In which the weight force of the second link and the second engine pass through the second rotational joint so don't produce any moment respect it, then the weight force of the second link and second engine produces a moment respect the first joint which is added to the moment produced by the weight of the first link, both negative; all these moments then are balanced by a positive actuation torque.

3) In the first case:



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [-pi/3, 0, 0;
     0, 2, 0;
     pi/3, 0, 0;
     0, 2, 0];
```

the gravity flag g_flag should be turned off:

```
g_flag = 0;
```

then there is an external force acting on the second link in the position P1 which is at the end of the link, and there isn't any moment so the external forces array and the external moments array must be set as:

```
F_ext = [-0.7 ,-0.5 ,0 , 5 ,0 ,1 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

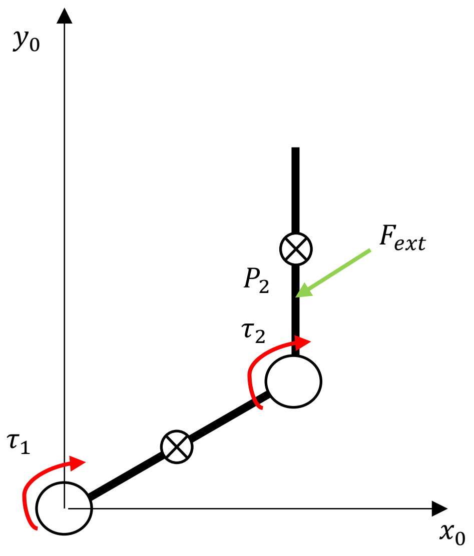
In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} \tau_1 \\ \tau_2 \end{cases} \begin{cases} -0.6170\ Nm \\ -0.7000\ Nm \end{cases}$$

In which the y-component of the force F_{ext} don't produce any moment on the second joint since it passes through it, while the x-component produce a positive moment, that is balanced by a negative actuation torque.

While on the first joint we have both the contribute to the moment of the y-component which produce a negative moment, and of the x-component which produce a positive moment, then the sum of the two is balanced by a negative actuation torque.

In the second case:



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [-pi/3, 0, 0;
     0, 2, 0;
     pi/3, 0, 0;
    0, 2, 0];
```

the gravity flag g_flag should be turned off:

```
g_flag = 0;
```

then there is an external force acting on the second link in the position P2 which is positioned 20 cm before the center of mass of the second link, and there isn't any moment so the external forces array and the external moments array must be set as:

```
F_ext = [-0.7 ,-0.5 ,0 , 5 ,0 ,0.3316 ,0;
          0 ,0 ,0 ,0 ,0 ,0 ,0;
          0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

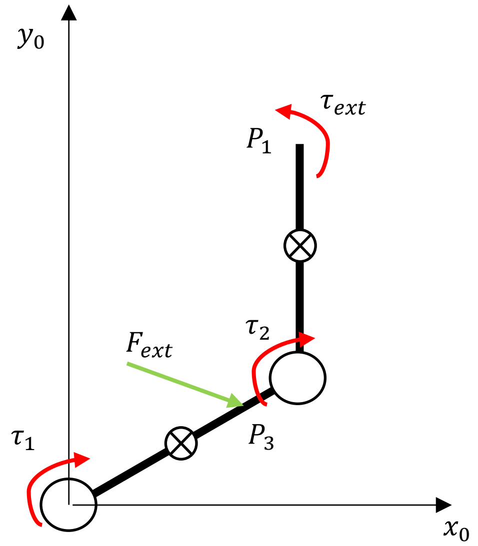
In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} \tau_1 \\ \tau_2 \end{cases} \begin{cases} -0.1491\ Nm \\ -0.2321\ Nm \end{cases}$$

In which the y-component of the force F_{ext} don't produce any moment on the second joint since the force passes through the joint itself, the x-component produce a positive moment which is balanced by a negative actuation torque.

While on the first joint we have both the contribute to the moment of the y-component, which produce a negative moment, and of the x-component, which produce a positive moment. The sum of the two contribute is then balanced by a negative actuation torque.

4)



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [-pi/3, 0, 0;
     0, 2, 0;
     pi/3, 0, 0;
    0, 2, 0];
```

the gravity flag g_flag should be turned off:

```
g_flag = 0;
```

then there is an external force acting on the first link in the position P3 which is 40 cm ahead the center of mass, and there is a moment acting around an axis passing through P1 so the external forces array and the external moments array must be set as:

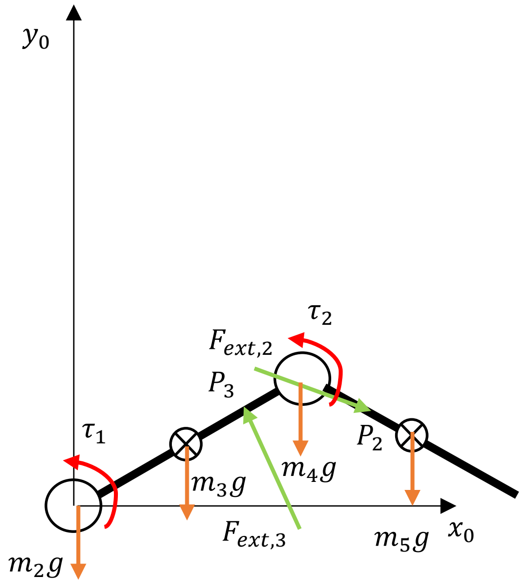
```
F_ext = [1.5 ,-0.3 ,0 , 3 ,0 ,0.9 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 1.2, 5;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} \tau_1 \\ \tau_2 \end{cases} \begin{cases} -0.2912\ Nm \\ -1.2000\ Nm \end{cases}$$

Where the torque τ_{ext} is positive, so it will be balanced by a negative moment on the second joint; then on the first joint we have the same torque τ_{ext} which translate in the position of the second joint summed with the contribute of the x-component which generate a negative moment as the y-component, and in the end the total resulting moment is balanced by a negative actuation torque.

5)



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [-pi/3, 0, 0;
     0, 2, 0;
     -pi/3, 0, 0;
     0, 2, 0];
```

the gravity flag g_flag should be turned on:

```
g_flag = 1;
```

then there are two external forces acting on the first and second link in the position P3 and P2 which are the first 40 cm ahead the center of mass of the first link and the second 20 cm before the center of mass of the second link, and there isn't any moment so the external forces array and the external moments array must be set as:

```
F_ext = [1.2 ,-0.2 ,0 , 5 ,0 ,0.3316 ,0;
         -0.4 ,1.2 ,0 ,3 ,0 ,0.9 ,0;
          0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

In conclusion we obtain the generalized actuation force vector composed as:

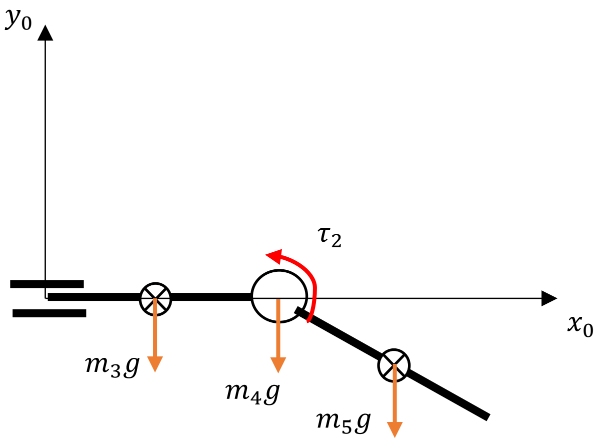
$$\tau_{eq} = \begin{cases} \tau_1 \\ \tau_2 \end{cases} \begin{cases} 1.5477e + 03\ Nm \\ 3.9449e + 03\ Nm \end{cases}$$

Here on the second joint, we've a negative moment generated by the weight of the second link, a negative moment generated by the y-component of the force $F_{ext,2}$ and a positive moment generated by the x-component of the force $F_{ext,2}$; the sum of all this contributes are balanced in the second joint by a positive actuation torque.
Then on the first joint, we've a negative moment generated by the weight of the second link, a negative moment generated by the y-component of the force $F_{ext,2}$, a negative moment

generated by the x-component of the force $F_{ext,2}$, a negative moment generated by the weight of the second engine, a positive component of both the y-component and x-component of the force $F_{ext,3}$ acting on P3 and a negative moment generated by the weight of the first link.
All this contribute are summed together to obtain a positive actuation moment.

Exercise 2:

1)



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [0, 1, -pi/2;
     0, 2, 0;
     -pi/4, 0, 0;
     0, 2, 0];
```

the gravity flag g_flag should be turned on:

```
g_flag = 1;
```

then there aren't any external force or moment so the external forces array and the external moments array must be set as:

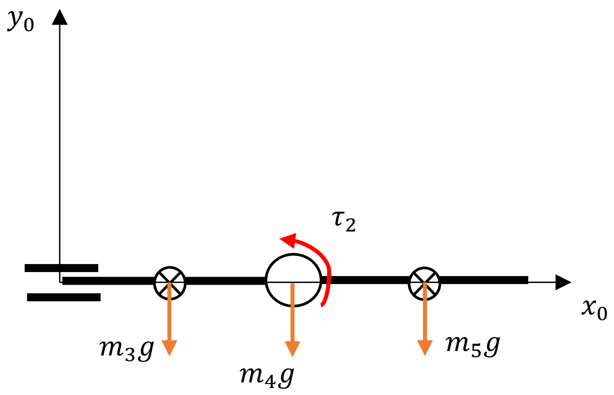
```
F_ext = [0 ,0 ,0 , 0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} f_1 \\ \tau_2 \end{cases} \begin{cases} 1.4377e-12 \cong 0\ N \\ 3.2211e+03\ Nm \end{cases}$$

Where the gravity force of the second link produces a negative moment around the rotational joint which is balanced by a positive actuation torque τ_2.
Then, since there aren't any horizontal force acting on the manipulator the actuator force on the prismatic joint will be 0.

2)



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [0, 1, -pi/2;
     0, 2, 0;
       0, 0, 0;
     0, 2, 0];
```

the gravity flag g_flag should be turned on:

```
g_flag = 1;
```

then there aren't any external force or moment so the external forces array and the external moments array must be set as:

```
F_ext = [0 ,0 ,0 , 0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

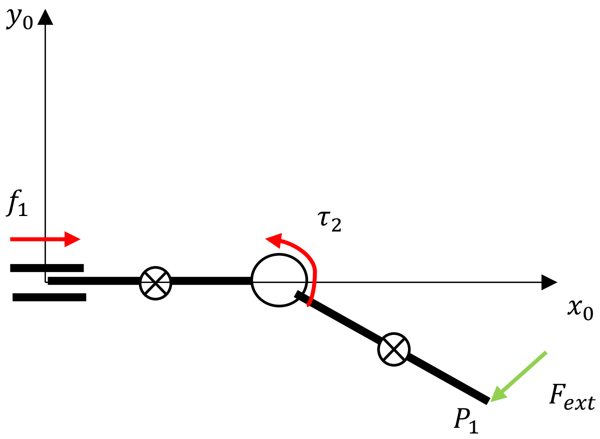
In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} f_1 \\ \tau_2 \end{cases} \begin{cases} 1.1062e - 12 \cong 0\ N \\ 4.5553e + 03\ Nm \end{cases}$$

Where the gravity force of the second link produces a negative moment around the rotational joint, bigger respect the previous case since the pitch of the moment is bigger, which is balanced by a positive actuation torque τ_2.

Then, since there aren't any horizontal force acting on the manipulator the actuator force on the prismatic joint will be 0.

3)



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [0, 1, -pi/2;
     0, 2, 0;
     -pi/4, 0, 0;
     0, 2, 0];
```

the gravity flag g_flag should be turned off:

```
g_flag = 0;
```

then there is one external force acting in P1 and there isn't any moment so the external forces array and the external moments array must be set as:

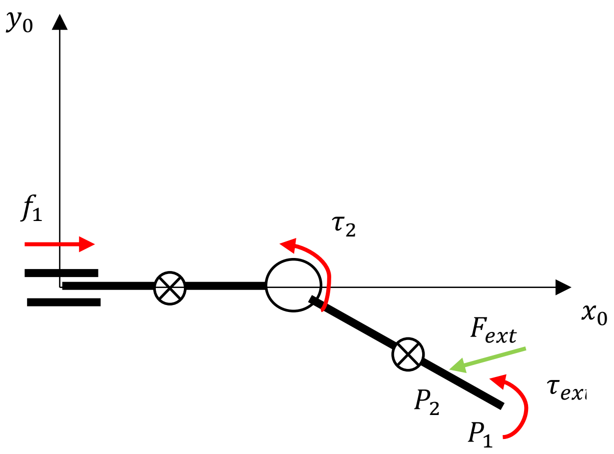
```
F_ext = [-0.8 ,-0.8 ,0 , 5 ,0 ,1 ,0;
          0 ,0 ,0 ,0 ,0 ,0 ,0;
          0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} f_1 \\ \tau_2 \end{cases} \begin{cases} 0.8000\ N \\ 1.1314\ Nm \end{cases}$$

Where the x-component of the force produce a negative moment on the revolute joint, as the y-component, and both are balanced by a positive actuation torque τ_2.
While the x-component of the force is negative and affect the actuation force on the prismatic joint which balances it with the positive actuation force f_1.

4)



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [0, 1, -pi/2;
     0, 2, 0;
     -pi/4, 0, 0;
     0, 2, 0];
```

the gravity flag g_flag should be turned off:

```
g_flag = 0;
```

then there is one external force acting in P2 and one moment acting around an axis passing through the point P1 so the external forces array and the external moments array must be set as:

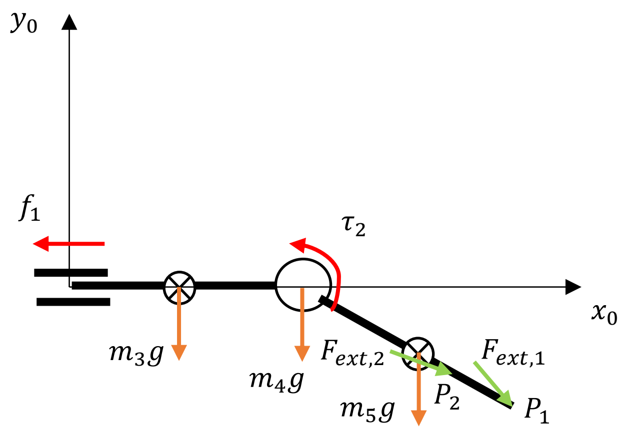
```
F_ext = [-0.8 ,-0.2 ,0 , 5 ,0 ,0.6816 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0;
         0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0.5, 5;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} f_1 \\ \tau_2 \end{cases} \begin{cases} 0.8000\ N \\ -0.0180\ Nm \end{cases}$$

Where the x-component of the force produce a negative moment on the revolute joint, as the y-component, moreover the moment applied in P1 is positive, and all are balanced by a positive actuation torque τ_2.
While the x-component of the force is negative and affect the actuation force on the prismatic joint which balances it with the positive actuation force f_1.

5)



In this case since the zero setting for the program is with all the link in a vertical position the actual configuration vector should be:

```
q = [0, 1, -pi/2;
     0, 2, 0;
     -pi/4, 0, 0;
     0, 2, 0];
```

the gravity flag g_flag should be turned on:

```
g_flag = 1;
```

then there are two external forces $F_{ext,1}, F_{ext,2}$ acting respectively on P1 and P2and no moment so the external forces array and the external moments array must be set as:

```
F_ext = [0.5 ,-0.6 ,0 , 5 ,0 ,1 ,0;
         1 ,-0.4 ,0 ,5 ,0 ,0.6816 ,0;
            0 ,0 ,0 ,0 ,0 ,0 ,0];

M_ext = [0, 0, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 0];
```

In conclusion we obtain the generalized actuation force vector composed as:

$$\tau_{eq} = \begin{cases} f_1 \\ \tau_2 \end{cases} \begin{cases} -1.5000\ N \\ 3.2209e+03\ Nm \end{cases}$$

Where the gravity force of the second link produces a negative moment around the rotational joint, the vertical component of both $F_{ext,1}, F_{ext,2}$ produce a negative moment and their horizontal component produce a positive moment, then all the forces system is balanced by a positive actuation torque τ_2.

Then both the horizontal component of the forces $F_{ext,1}, F_{ext,2}$ are positive and so are balanced by a negative actuation force f_1.

Conclusion:

The program work properly under the assumption of a 2D mechanism for any possible once composed of n links joined together with revolute joint and prismatic joint (by following the convention about the reference frame in the 3D modeling part), but it can be easily upgraded to consider also other types of joint, as spherical joint, and 3D mechanism (by this I mean mechanism with joints oriented in such a way that the mechanism exits from the plane).
In the exercises solved in this assignment we can find out how in all the cases where the weight of the links is evaluated together with the additional external forces and moments the contribute of these last one is very small compared to the first one since the ratio between the two is in the order of 10^4.
Moreover, in order to maintain a general structure of the program useful for a generic case I've changed the zero configuration of the system with respect to the one of the assignment, but it can easily be restored by redefine the configuration vector in such a way that the configuration angle of the first joint typed by the user is subtracted by -pi/2.
We can also note that the results on the second exercise are not affected by the weight of the first engine, of the first link and on the configuration of the prismatic joint if not for the vincular reaction which aren't computed in this program, but can be easily implemented as the remaining component of the forces and moments exchanged between the links.