This report is part of Dynamics of Non linear Robotic Systems [DNlRS] course for first year master students at Innopolis University. In this report yo can find the full Derivation of the dynamic model for Mitsubishi robot RV-1A while the implementation is available on GitHub.

1 Introduction

Derivation of the dynamic model of a manipulator plays an important role for simulation of motion, analysis of manipulator structures, and design of control algorithms. In this report we will use Euler-lagrane method in order ro derive the dynamic model of the mitsubishi robot RV-1A. As it is shown in (Figure 1) we have a 6 DOF arm manipulator. I am assuming all

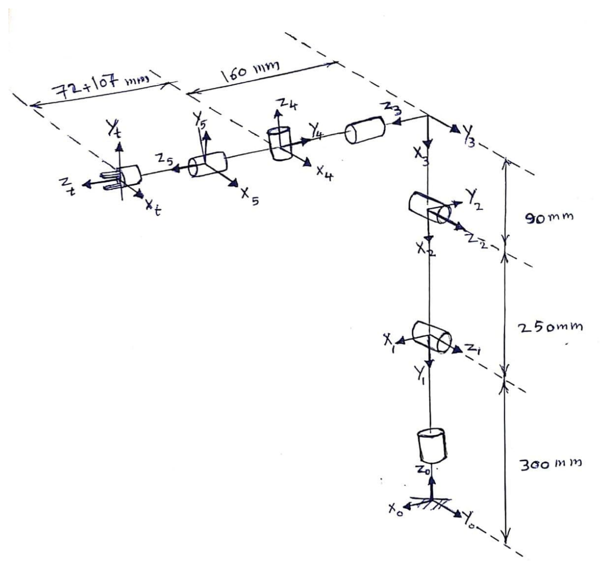


Figure 1: Mitsubishi robot RV-1A scheme.

of masses (M1, M2, M3, M4, M5m M6) exist at the end of links for simplicity. six actuators exist at each joint, and directly actuate the torques (u1, u2, u3, u4, u5, u6). The manipulator kinematics is governed by six joint angles (q1, q2, q3, q4, q5, q6).

2 Theoretical Background

The dynamic model of a manipulator provides a description of the relationship between the joint actuator torques and the motion of the structure. With lagrange formulation, the equations of motion can be derived in a systematic way independently of the reference coordinate frame. lagrangian of the mechanical system can be defined as a function of the generalized coordinates. The general dynamic equation is obtained by:

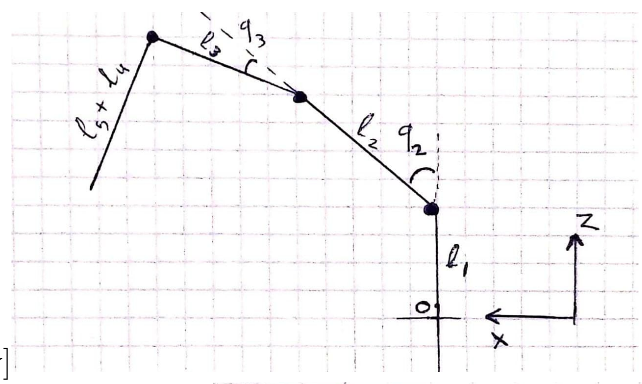
$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\right) - \frac{\partial T}{\partial q_j} + \frac{\partial V}{\partial q_j} = P_j \tag{1}$$

Where, T is the total kinetic energy, V is the total potential energy of the system.P_j is the generalized force, t is time, q_j is the generalized coordinates and $\dot{q}_j$ is the generalized velocity.

3 Dynamic model usg the Euler-lagrange approach

3.1 Math Matrix $M(q)$:

First of all, we need to determine the center of mass - which in our case in the end of each link - and write the equations of each link projected on each axes.

[Side view] 

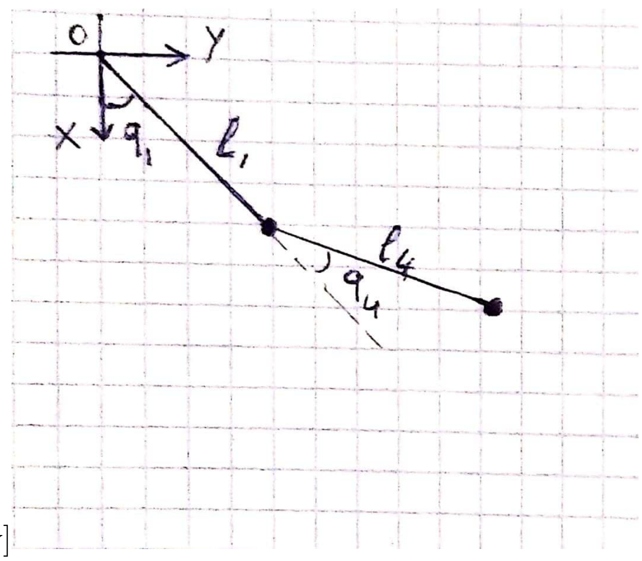
[Top view] 

Figure 2: Mitsubishi robot RV-1A Top and Side views

$x_1 = l1.c1$
$x_2 = l1.c1 + l2.c2$
$x_3 = l1.c1 + l2.c2 + l3.c(q_2 + q_3)$
$x_4 = l1.c1 + l2.c2 + l3.c(q_2 + q_3) + l4.c(q_1 + q_4)$
$x_5 = l1.c1 + l2.c2 + l3.c(q_2 + q_3) + l4.c(q_1 + q_4)$
$x_6 = l1.c1 + l2.c2 + l3.c(q_2 + q_3) + l4.c(q_1 + q_4) + l6.c(q_1 + q_4 + q_6)$

$y_1 = l1.s1$

$y_2 = l1.s1$
$y_3 = l1.s1$
$y_4 = l1.s1 + l4.s(q_1 + q_4)$
$y_5 = l1.s1 + l4.s(q_1 + q_4) + l5.c5$
$y_6 = l1.s1 + l4.s(q_1 + q_4) + l5.c5 + l6.s(q_1 + q_4 + q_6)$

$z_1 = 0$
$z_2 = l2.s2$
$z_3 = l2.s2 + l3.c(q_2 + q_3)$
$z_4 = l2.s2 + l3.c(q_2 + q_3)$
$z_5 = l2.s2 + l3.c(q_2 + q_3) + l5.s5$
$z_6 = l2.s2 + l3.c(q_2 + q_3) + l5.s5$

Now we have to compute translation and rotation jacobian matrices in order to compute $M_(q)$.
let us compute the translation jacobians:

$$\mathbb{J}_v^i = \begin{bmatrix} \frac{\partial x_i}{q_1} & \cdots & \frac{\partial x_i}{q_6} \\ \frac{\partial y_i}{q_1} & \cdots & \frac{\partial y_i}{q_6} \\ \frac{\partial z_i}{q_1} & \cdots & \frac{\partial z_i}{q_6} \end{bmatrix} \tag{2}$$

$$\mathbb{J}_v^1 = \begin{bmatrix} -l1.s1 & 0 & 0 & 0 & 0 & 0 \\ l1.c1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbb{J}_v^2 = \begin{bmatrix} -l1.s1 & -l2.s2 & 0 & 0 & 0 & 0 \\ l1.c1 & 0 & 0 & 0 & 0 & 0 \\ 0 & l2.c2 & 0 & 0 & 0 & 0] \end{bmatrix}$$

$$\mathbb{J}_v^3 = \begin{bmatrix} -l1.s1 & -l3.s23 - l2.s2 & -l3.s23 & 0 & 0 & 0 \\ l1.c1 & 0 & 0 & 0 & 0 & 0 \\ 0 & l2.c2 - l3.s23 & -l3.s23 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbb{J}_v^4 = \begin{bmatrix} -l4.s14 - l1.s1 & -l3.s23 - l2.s2 & -l3.s23 & -l4.s14 & 0 & 0 \\ l4.c14 + l1.c1 & 0 & 0 & l_4.c14 & 0 & 0 \\ 0 & l2.c2 - l3.s23 & -l3.s23 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbb{J}_v^5 = \begin{bmatrix} -l4.s14 - l1.s1 & -l3.s23 - l2.s2 & -l3.s23 & -l4.s14 & 0 & 0 \\ l4.c14 + l1.c1 & 0 & 0 & l4.c14 & -l5.s5 & 0 \\ 0 & l2.c2 - l3.s23 & -l3.s23 & 0 & l5.c5 & 0 \end{bmatrix}$$

$$\mathbb{J}_v^6 = \left[\begin{matrix} -l4.s14 - l1.s1 - l6.s146, -l3.s23 - l2.s2 & -l3.s23 & -l4.s14 - l6.s146 & 0 \\ l4.c14 + l1.c1 + l6.c146 & 0 & 0 & l4.c14 + l6.c146 \\ -l4.s14 - l1.s1 - l6.s146 & -l3.s23 - l2.s2 & -l3.s23 & -l4.s14 - l6.s146 \end{matrix} \right.$$

and now we will move to the rotation jacobian:

$$\mathbb{J}_w^1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbb{J}_w^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbb{J}_w^3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbb{J}_w^4 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \mathbb{J}_w^5 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbb{J}_w^6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

now we can compute $M(q)$ matrix, by using the following formula:

$$M(q) = \sum_{i=1}^{6} m_i \mathbb{J}_v^{i\,T} \mathbb{J}_v^i + \mathbb{J}_w^{i\,T} R_i I R_i^T \mathbb{J}_w^i \tag{3}$$

where m_i is the mass of the i^{th} joint, and I is the inertia matrix in which we will consider it as a unit matrix for simplicity (but we can change it whenever we want regarding our model)

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

so, our $M(q)$ matrix will be in the following form:

$$M(q) = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} \\ m_{31} & m_{32} & m_{33} & m_{34} & m_{35} & m_{36} \\ m_{41} & m_{42} & m_{43} & m_{44} & m_{44} & m_{46} \\ m_{51} & m_{52} & m_{53} & m_{54} & m_{55} & m_{56} \\ m_{61} & m_{62} & m_{63} & m_{64} & m_{65} & m_{66} \end{bmatrix} \tag{5}$$

Then we calculate $C(q, \dot{q})$matrix, by using the following formula:

$$C_{ij} = \sum_{k=1}^{6} c_{ijk} \dot{q}_k \tag{6}$$

and we will also get a matrix as following:

$$C(q, \dot{q}) = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} & c_{16} \\ c_{21} & c_{22} & c_{23} & c_{24} & c_{25} & c_{26} \\ c_{31} & c_{32} & c_{33} & c_{34} & c_{35} & c_{36} \\ c_{41} & c_{42} & c_{43} & c_{44} & c_{44} & c_{46} \\ c_{51} & c_{52} & c_{53} & c_{54} & c_{55} & c_{56} \\ c_{61} & c_{62} & c_{63} & c_{64} & c_{65} & c_{66} \end{bmatrix} \tag{7}$$

and the $g(q)$ matrix is:

$$g = \sum_{k=1}^{6} (\mathbb{J}_{wi}^{k})^T g_0 \tag{8}$$

finally we can derive the dynamic model by using Euler-Lagrange approach by substituting in the next formula:

$$M(q) + C(q, \dot{q})\dot{q} + g(q) = \tau \tag{9}$$

you can find the code and implementation on GitHub.