INNOPOLIS UNIVERSITY

“Project Task2”

DYNAMICS OF NON-LINEAR ROBOTIC SYSTEMS

By:

Ahmed Mohsen Mohamed Abdelkhalek Elsayed Ali

Date

12-sep-2021

Jacobian Matrix Derivation (General Form)

Spherical robot is a 3 DOF robot with configuration of RRP. In this assignment, a spherical wrist is attached with YXY configurations for better position and orientation control. In the figure below, hand sketch model was drawn to define robot joints and their rotation axes. The overall configuration of the robot is RRPRRR.

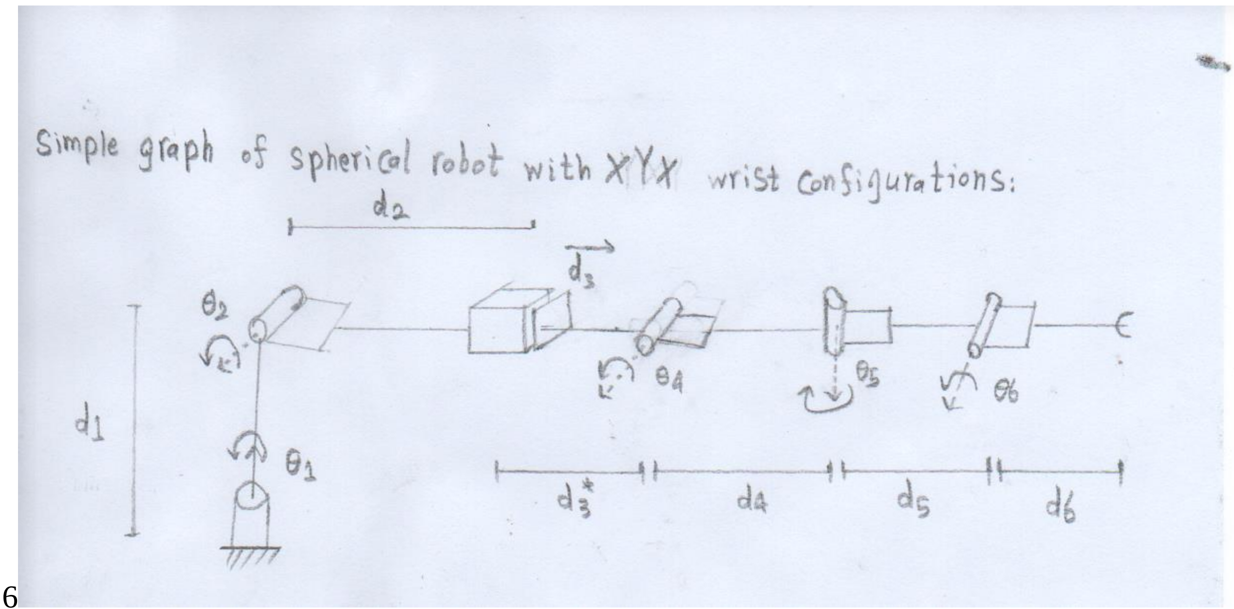
6 

Figure 1; Simple graph of spherical robot

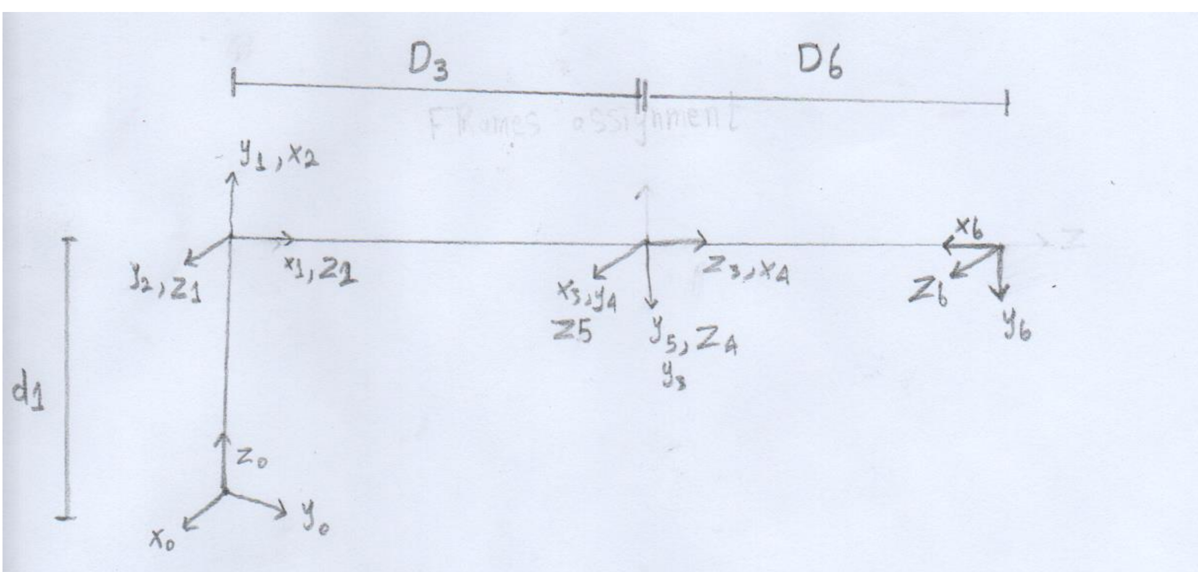


Figure 2; Frames Assignment

The jacobian depends on the number of robot joints. It it is a m×n matrix where m represents space dimension of end effector and n represents the number of joints. The general formula for the robot can be calculated as follows:

$$j = [j_1\ j_2\ \ldots \ldots.\ j_n]$$

]Knowing the type of each joint, the jacobian matrix of each joint can be calculated using the following formulas:

if i^{th}joint is revolute:

$$\begin{pmatrix} Z_{i-1} \times (O_n - O_{n-}) \\ Z_{i-1} \end{pmatrix}$$

if i^{th}joint is prismatic:

$$\begin{pmatrix} Z_{i-1} \\ 0 \end{pmatrix}$$

Jacobian Matrix Derivation (Model)

Using the same procedures, the final jacobian matrix of this model will be:

$$j = [j_1\ j_2\ j_3\ j_4\ j_5\ \ j_6]$$

Where:

$$j_1 \ = \begin{pmatrix} Z_0 \times (O_6 - O_0) \\ Z_i \end{pmatrix}$$

$$j_2 \ = \begin{pmatrix} Z_1 \times (O_6 - O_1) \\ Z_1 \end{pmatrix}$$

$$j_3 \ = \begin{pmatrix} Z_2 \\ 0 \end{pmatrix}$$

$$j_4 \ = \begin{pmatrix} Z_3 \times (O_6 - O_3) \\ Z_3 \end{pmatrix}$$

$$j_5 = \begin{pmatrix} Z_4 \times (O_6 - O_4) \\ Z_4 \end{pmatrix}$$

$$j_6 = \begin{pmatrix} Z_5 \times (O_6 - O_5) \\ Z_5 \end{pmatrix}$$

The jacobian matrix for the third joint is different from others since it is the only prismatic joint in the model configuration.

Jacobian Matrix Singularities

According to the lecture singularity can occur when the determinant of the Jacobian is equal to zero. In order to check for this case, the following condition has been added to the code:

```
newj=j;
newj(1:3,4:6)=zeros(3);
%This condition checks if there is simgularity on a given configuration
if det(newj(1:3,1:3))*det(newj(4:6,4:6))==0
    fprintf('This is a signularity!')
```

This condition follows the formula below based on the idea of making the last three joint along with the end effector on the same origin. The formula is as follow:

$$J = \begin{pmatrix} J_{11} & 0 \\ J_{21} & J_{22} \end{pmatrix}$$

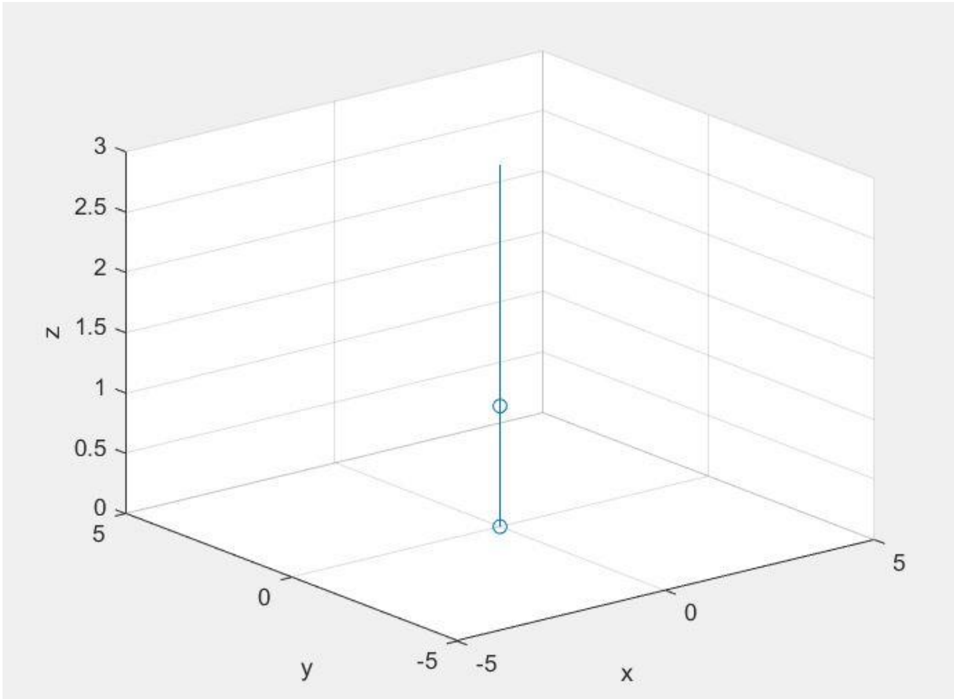
As a result:

$$\text{Det}(J) = \text{Det}(J_{11}) \times \text{Det}(J_{21})$$

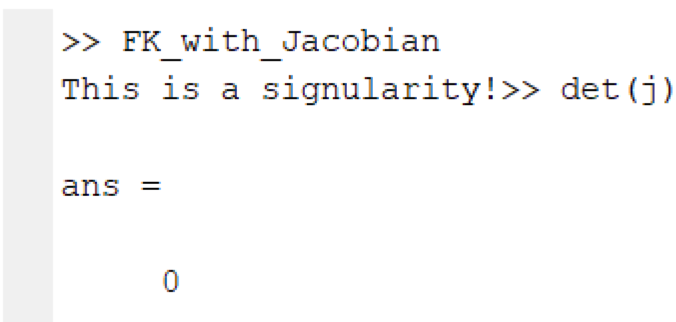
The determinant can be zero in the following conditions:

Case1: The robot is fully extended

This can occure when theta 2 = 90 degrees as shown in the model below



The determinant for this configuration was:



```
>> FK_with_Jacobian
This is a signularity!>> det(j)

ans =

     0
```

Case2: end effector is on first joint axis:

There are multiple configuration that can result into this singularity. This can be overcome using controller limits

Case3: Axis 4 and 6 are collinear:

This occure when theta 5 is equal to 0 or 180. This can be overcome using mechanical limits.