



**TÉCNICO**  
LISBOA

# ROBOTICS OF MANIPULATION

## MASTERS IN BIOMEDICAL AND ELECTRICAL ENGINEERING

---

### Assignment #2 - Dynamics [EN]

---

#### Authors:

Afonso Araújo (96138)  
Tiago Freitas (94186)

[afonso.d.araujo@tecnico.ulisboa.pt](mailto:afonso.d.araujo@tecnico.ulisboa.pt)  
[tiagogoncalvesfreitas@tecnico.ulisboa.pt](mailto:tiagogoncalvesfreitas@tecnico.ulisboa.pt)

**Group 5**

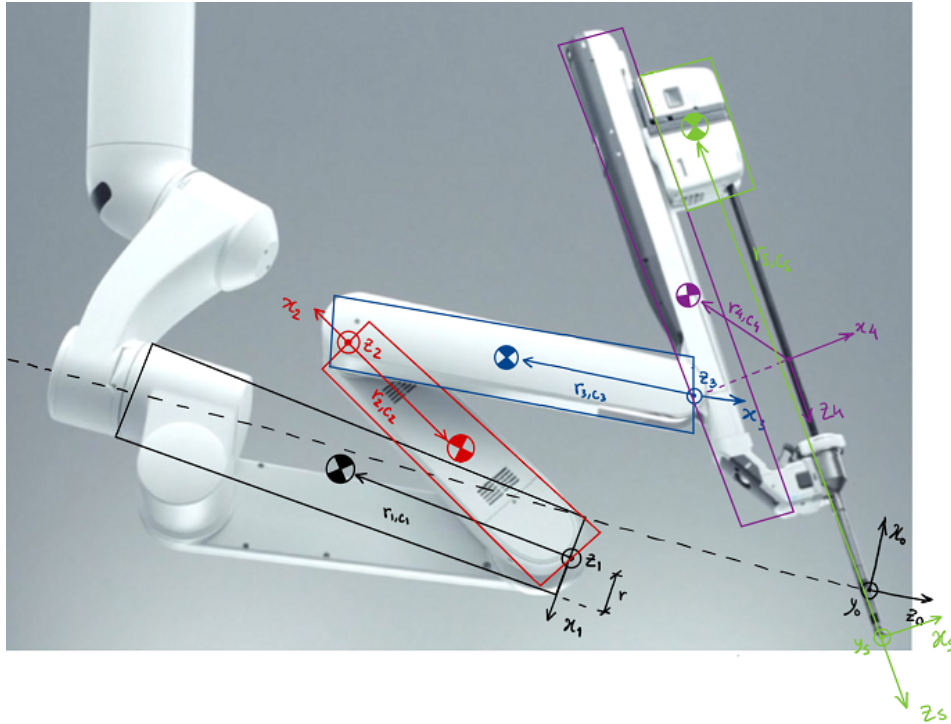
2022/2023 – 2<sup>nd</sup> Semester, P4

# Contents

<b>1</b>	<b>Question 1</b>	<b>2</b>
<b>2</b>	<b>Question 2</b>	<b>4</b>
<b>3</b>	<b>Question 3</b>	<b>8</b>
3.1	Simulation Pre-Validation . . . . .	11
<b>4</b>	<b>Question 4</b>	<b>13</b>
4.1	Simulation Pre-Validation . . . . .	14
<b>5</b>	<b>Question 5</b>	<b>16</b>
5.1	Decentralized and Centralized Controller Tests for Trajectory . . . . .	23
5.1.1	First Simulation . . . . .	23
5.1.2	Second Simulation . . . . .	26
5.1.3	Third Simulation . . . . .	28
5.2	Conclusion . . . . .	30
<b>6</b>	<b>References</b>	<b>30</b>

# 1 Question 1

As seen through the image below, the Dynamical approach of the DaVinci will take into consideration 5 links, as the end of the the end of the trocar can be neglected due to its relative lower mass,



**Figure 1:** DaVinci Xi Links and their Centers of Mass Representations

Links #1-#4 will be approximated as cylinders, while Link #5 will be approximated as a solid cuboid, for the latter inertia tensor calculations. Moreover, it is necessary to define both center of mass,  $c_m = [c_{mx} c_{my} c_{mz}]^T$  and approximate mass for each link. When calculating the centers of mass vectors for Links #1-#3, a simpler approach will be taken as the depth of the robot will be neglected, it will be assumed that every link before Link# 4, stands on the same 2D plane.

Below is a table for the display of this data.

Link	$c_m = r_{i,c_i}$	mass/kg	$\mathbf{I}_i^i$
1	$\begin{pmatrix} 0 \\ -0.2 \\ 0 \end{pmatrix}$	3	$\begin{pmatrix} -\frac{1}{12}m(3r^2 + h^2) & 0 & 0 \\ 0 & \frac{1}{2}mr^2 & 0 \\ 0 & 0 & \frac{1}{12}m(3r^2 + h^2) \end{pmatrix}$
2	$\begin{pmatrix} -0.125 \\ 0 \\ 0 \end{pmatrix}$	1.5	$\begin{pmatrix} \frac{1}{2}mr^2 & 0 & 0 \\ 0 & \frac{1}{12}m(3r^2 + h^2) & 0 \\ 0 & 0 & \frac{1}{12}m(3r^2 + h^2) \end{pmatrix}$
3	$\begin{pmatrix} -0.15 \\ 0 \\ 0 \end{pmatrix}$	1.5	$\begin{pmatrix} \frac{1}{2}mr^2 & 0 & 0 \\ 0 & \frac{1}{12}m(3r^2 + h^2) & 0 \\ 0 & 0 & \frac{1}{12}m(3r^2 + h^2) \end{pmatrix}$
4	$\begin{pmatrix} -0.05 \\ 0 \\ -0.05 \end{pmatrix}$	1	$\begin{pmatrix} \frac{1}{12}m(3r^2 + h^2) & 0 & 0 \\ 0 & \frac{1}{12}m(3r^2 + h^2) & 0 \\ 0 & 0 & \frac{1}{2}mr^2 \end{pmatrix}$
5	$\begin{pmatrix} -0.03 \\ 0 \\ -0.5 \end{pmatrix}$	3	$\begin{pmatrix} \frac{1}{12}m(h^2 + d^2) & 0 & 0 \\ 0 & \frac{1}{12}m(w^2 + d^2) & 0 \\ 0 & 0 & \frac{1}{12}m(w^2 + h^2) \end{pmatrix}$

**Table 1:** Dynamical Parameters of the DaVinci

The centers of mass, vectors  $r_{i,c_i}$ , represent the distance from referential  $i$  to the center of mass as represented by  $\bullet$  in 1. For example, for Link#4, the  $\bullet$  is 5cm across the negative  $x_4$  axis, and "up" 5cm across the negative  $z_4$  axis.

For the calculation of the Inertia Tensors the following equation was applied,

$$I_i^i = R_\alpha^i I_i^\alpha R_i^\alpha \quad (1)$$

This equation shows the relationship between the inertia tensor at the centers of mass of 1,  $I_i^\alpha$ , and the inertia tensor at the referential origin  $i$ ,  $I_i^i$ .

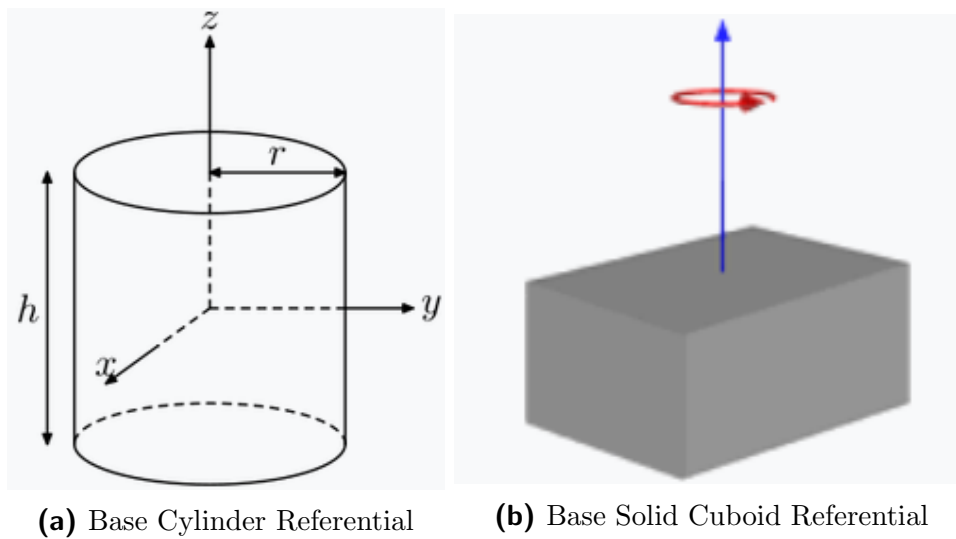
For cylindrical shapes the inertia tensor is,

$$I_i^\alpha = \begin{pmatrix} \frac{1}{12}m(3r^2 + h^2) & 0 & 0 \\ 0 & \frac{1}{12}m(3r^2 + h^2) & 0 \\ 0 & 0 & \frac{1}{2}mr^2 \end{pmatrix}, \text{ with } r = 0.05\text{m and } h = 0.2\text{m} \quad (2)$$

while for solid cuboids,

$$I_i^\alpha = \begin{pmatrix} \frac{1}{12}m(h^2 + d^2) & 0 & 0 \\ 0 & \frac{1}{12}m(w^2 + d^2) & 0 \\ 0 & 0 & \frac{1}{12}m(w^2 + h^2) \end{pmatrix}, \text{ with } w = 0.1\text{m}, d = 0.2\text{m and } h = 0.2\text{m} \quad (3)$$

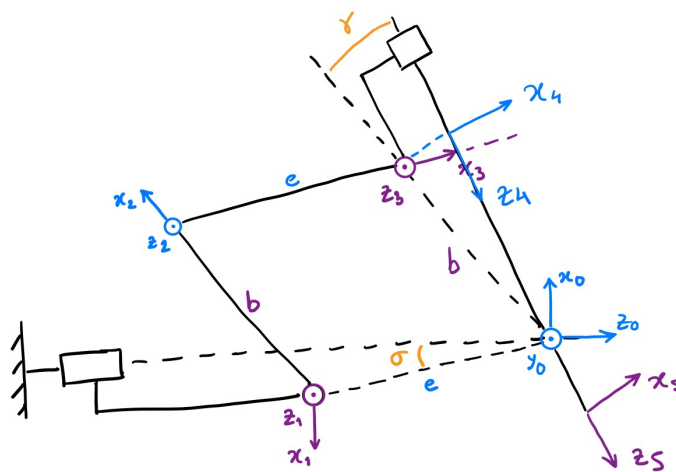
For the same example as before, Link #4, the chosen referential aligns both the  $x$ -axis, the  $y$ -axis and the  $z$ -axis with the  $x$ -axis,  $y$ -axis and  $z$ -axis respectively of the  $\bullet$ , 2a. Hence,  $I_4^4 = I_i^\alpha$ .



**Figure 2:** Solid Shape's Referentials

## 2 Question 2

For a Dynamical approach of the DaVinci Xi, a new DaVinci joint model is presented,



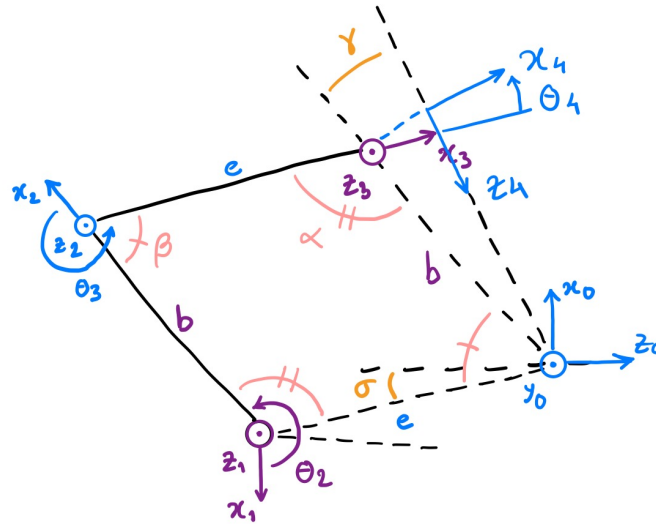
**Figure 3:** DaVinci Dynamical Configuration

this model yields the following Denavit-Hartenberg table,

$d_i$	$\theta_i$	$a_i$	$\alpha_i$
$-e \cos(\sigma)$	$\theta_1$	$e \sin(\sigma)$	$\frac{\pi}{2}$
0	$\theta_2$	$b$	0
0	$\theta_3$	$e$	0
0	$\theta_4$	$b \sin(\gamma)$	$\frac{\pi}{2}$
$d_5$	0	0	0

**Table 2:** Denavit Hartenberg Table

Due to the parallelogram structure in the middle of the DaVinci Xi, the motion of joints 2,3 and 4 is constrained producing a "remote center of motion". Moreover, the parallelogram's properties are used to solve the Kinematic constraints, represented below,



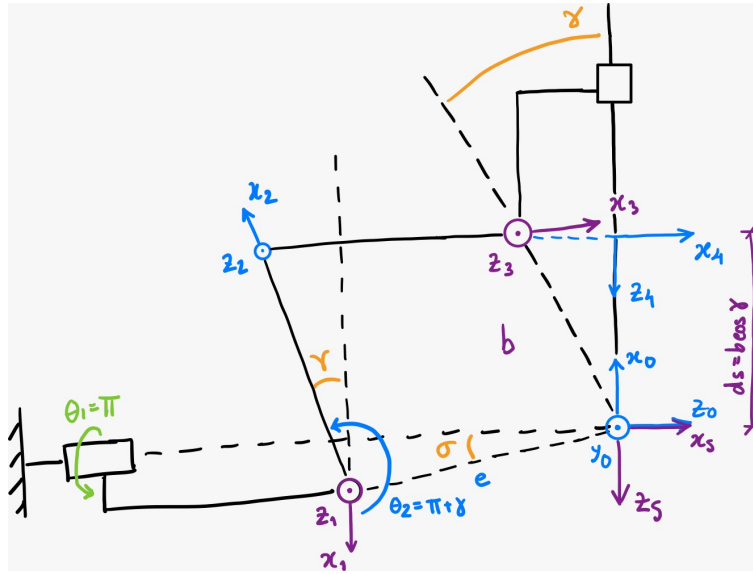
**Figure 4:** DaVinci Constraints

which yields a new DH table,

$d_i$	$\theta_i$	$a_i$	$\alpha_i$
$-e \cos(\sigma)$	$\theta_1$	$e \sin(\sigma)$	$\frac{\pi}{2}$
0	$\theta_2$	$b$	0
0	$-\theta_2 + \sigma + \frac{\pi}{2}$	$e$	0
0	$\theta_2 - \sigma - \gamma - \pi$	$b \sin \gamma$	$\frac{\pi}{2}$
$d_5$	0	0	0

**Table 3:** Denavit Hartenberg Table with Kinematic Constraints

With these new Kinematic Constraints added, it is practical to add a new set of offsets as to make the home configuration of the DaVinci easier to work with. This leads to the following Home configuration of the DaVinci Xi,



**Figure 5:** Offsetted DaVinci Dynamical Model

which in turn leads to the final DH table,

$d_i$	$\theta_i$	$a_i$	$\alpha_i$
$-e \cos(\sigma)$	$\theta_1 + \pi$	$e \sin(\sigma)$	$\frac{\pi}{2}$
0	$\theta_2 + \pi + \gamma$	$b$	0
0	$-(\theta_2 + \pi + \gamma) + \sigma + \frac{\pi}{2}$	$e$	0
0	$(\theta_2 + \pi + \gamma) - \sigma - \gamma - \pi$	$b \sin(\gamma)$	$\frac{\pi}{2}$
$d_5 + b \cos \gamma$	0	0	0

$q_{\text{offsetted}} = (\theta_1 + \pi, \theta_2 + \pi + \gamma, d_5 + b \cos \gamma)$

$d_i$	$\theta_i$	$a_i$	$\alpha_i$
$-e \cos(\sigma)$	$\theta_1 + \pi$	$e \sin(\sigma)$	$\frac{\pi}{2}$
0	$\theta_2 + \pi + \gamma$	$b$	0
0	$-\theta_2 - \gamma + \sigma - \frac{\pi}{2}$	$e$	0
0	$\theta_2 - \sigma$	$b \sin(\gamma)$	$\frac{\pi}{2}$
$d_5 + b \cos \gamma$	0	0	0

$q_{\text{offsetted}} = (\theta_1 + \pi, \theta_2 + \pi + \gamma, -\theta_2 - \gamma + \sigma - \frac{\pi}{2}, \theta_2 - \sigma, d_5 + b \cos(\gamma))$

**Table 4:** Denavit Hartenberg Table with Offsetted Kinematic Constraints

(4)

In order to create a direct dynamical model for the DaVinci Xi the Newton-Euler method was adopted as to solve for:

$$\ddot{q} = \underbrace{B^{-1}}_{\text{Acceleration Dynamics}} (n + \tau) \quad (5)$$

where  $n$ ,

$$n = \underbrace{\phi}_{\text{Velocity Dynamics}} + \underbrace{G}_{\text{Gravity Dynamics}} \quad (6)$$

The Newton-Euler method consists of a Forward Recursion,

$$\omega_i^i = \begin{cases} R_i^{i-1T} \omega_{i-1}^{i-1} & , q_i \text{ is prismatic} \\ R_i^{i-1T} (\omega_{i-1}^{i-1} + \dot{q}_i \hat{k}) & , q_i \text{ is revolute} \end{cases} \quad (7)$$

$$\dot{\omega}_i^i = \begin{cases} R_i^{i-1T} \dot{\omega}_{i-1}^{i-1} & , q_i \text{ is prismatic} \\ R_i^{i-1T} (\dot{\omega}_{i-1}^{i-1} + \ddot{q}_i \hat{k} + \dot{q}_i \omega_{i-1}^{i-1} \times \hat{k}) & , q_i \text{ is revolute} \end{cases} \quad (8)$$

$$\ddot{q}_i^i = \begin{cases} R_i^{i-1T} (\ddot{p}_{i-1}^{i-1} + \ddot{q}_i \hat{k}) + 2\dot{q}_i \omega_i^i \times (R_i^{i-1T} \hat{k}) + \omega_i^i \times (R_i^{i-1T} r_{i-1,i}^{i-1}) + \omega_i^i \times (\omega_i^i \times (R_i^{i-1T} r_{i-1,i}^{i-1})) & , q_i \text{ is prismatic} \\ R_i^{i-1T} \ddot{p}_{i-1}^{i-1} + \ddot{\omega}_i^i \times (R_i^{i-1T} r_{i-1,i}^{i-1}) + \omega_i^i \times (\omega_i^i \times (R_i^{i-1T} r_{i-1,i}^{i-1})) & , q_i \text{ is revolute} \end{cases} \quad (9)$$

$$\ddot{p}_{C_j}^i = \ddot{p}_i^i + \dot{\omega}_i^i \times r_{i,i}^i + \omega_i^i \times (\omega_i^i \times r_{i,i}^i) \quad (10)$$

and a Backward Recursion,

$$f_i^i = m_i \ddot{p}_{C_j}^i + R_{i+1}^i f_{i+1}^{i+1} \quad (11)$$

$$\mu_i^i = \bar{I}_i^i \ddot{\omega}_i^i + \omega_i^i \times (\bar{I}_i^i \omega_i^i) + R_{i+1}^i \mu_{i+1}^{i+1} - r_{i,C_j}^i \times (R_{i+1}^i f_{i+1}^{i+1}) + (R_i^{i-1T} r_{i-1,i}^{i-1} + r_{i,C_j}^i) \times f_i^i \quad (12)$$

yielding our joint axis projection with  $\tau$ ,

$$\tau_i = \begin{cases} \hat{k} R_i^{i-1} f_i^i & , q_i \text{ is prismatic} \\ \hat{k}^T R_i^{i-1} \mu_i^i & , q_i \text{ is revolute} \end{cases} \quad (13)$$

However, this method is used differently whether the  $B(q)$  or  $n(q, \dot{q})$  matrices are being taken in consideration as,

$$\text{Newton\_Euler}(\text{Denavit\_Hartenberg Parameters}, \text{Mass\_Vectors}, \dot{q}, \ddot{q}, \omega_0^0, \dot{\omega}_0^0, \ddot{p}_0^0(|g|)) \quad (14)$$

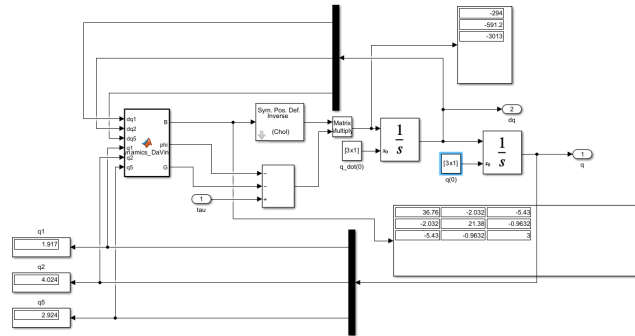
hence,

$$B(q) \dot{q} = \tau_a = \text{Newton\_Euler}(\text{DH}, M, 0, \ddot{q}, 0, \dot{\omega}_0^0, \ddot{p}_0^0(0)) \iff B(q) = \frac{\partial \tau_a}{\partial \dot{q}} \quad (15)$$

$$\phi(q, \dot{q}) = \text{Newton\_Euler}(\text{DH}, M, \dot{q}, 0, \omega_0^0, 0, 0) \quad (16)$$

$$g(q) = \text{Newton\_Euler}(\text{DH}, M, 0, 0, 0, 0, \ddot{p}_0^0(|g|) - \ddot{p}_0^0(0)) \quad (17)$$

The *Simulink* model of the Direct Dynamics is represented below,



**Figure 6:** Direct Dynamical Model

Since the simulated  $B(q)$  matrix is symmetric, the model should be validated.



### 3 Question 3

With the Dynamic simulation built, the next step is to build a decentralized and a centralized controller to compare the performance of the chosen trajectory on both controllers. Here the decentralized controller will be discussed.

In the decentralized controller three joints are controlled independently. This means, that the coupling between the motion of the joints is treated as a disturbance, along with the variations in the mass distribution.

It is then possible to approximate the DaVinci's dynamics to a linear system represented as,

$$B_{\max}\ddot{q} = \tau \quad (18)$$

where,

$$B_{\max} = \text{diag}(\max(B_{11})\dots\max(B_{nn})) \quad (19)$$

The chosen decentralized controller acts as a **P-D controller** defined as,

$$\tau = -K_d\dot{q} - K_p(q - q_d) \quad (20)$$

with

$$K_d = \begin{pmatrix} K_{d1} & \dots & 0 \\ \vdots & \ddots & 0 \\ 0 & \dots & K_{dn} \end{pmatrix} \quad (21)$$

$$K_p = \begin{pmatrix} K_{p1} & \dots & 0 \\ \vdots & \ddots & 0 \\ 0 & \dots & K_{pn} \end{pmatrix} \quad (22)$$

The closed loop then becomes,

$$B_{\max}\ddot{q} + K_d\dot{q} + K_pq \approx K_pq_d \quad (23)$$

or, in the frequency domain,

$$(s^2 + \frac{K_{d_i}}{B_{\max_{ii}}}s + \frac{K_{p_i}}{B_{\max_{ii}}})Q(s) \approx \frac{K_{p_i}}{B_{\max_{ii}}}Q_{d_i}(s) \quad (24)$$

The stiffness and damping matrices,  $K_p$  and  $K_d$ , are related to the frequency and damping ratio through,

$$K_{p_i} = B_{\max_{ii}}\omega_{n_i}^2, K_{d_i} = 2B_{\max_{ii}}\zeta_i\omega_{n_i} \quad (25)$$

Looking back at the  $B(q)$  matrix, it is described as,

$$B(q) = \begin{pmatrix} B_{11}(q_1, q_2, q_3) & \dots & \vdots \\ \vdots & B_{22}(q_1, q_2, q_3) & \vdots \\ \vdots & \dots & B_{33}(q_1, q_2, q_3) \end{pmatrix} \quad (26)$$

Since the worst inertia is verified for the maximum values for the diagonal of  $B(q)$ , 19, the following computation must be made,

$$\max(B_{11}) = B_{11}((\frac{\partial B_{11}}{\partial q_1} = 0), (\frac{\partial B_{11}}{\partial q_2} = 0), (\frac{\partial B_{11}}{\partial q_3} = 0)) \quad (27)$$

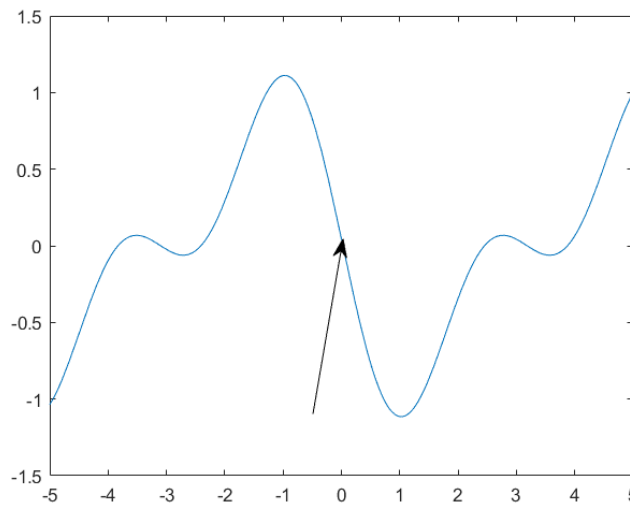
$$\max(B_{22}) = B_{22}((\frac{\partial B_{22}}{\partial q_1} = 0), (\frac{\partial B_{22}}{\partial q_2} = 0), (\frac{\partial B_{22}}{\partial q_3} = 0)) \quad (28)$$

$$\max(B_{33}) = B_{33}((\frac{\partial B_{33}}{\partial q_1} = 0), (\frac{\partial B_{33}}{\partial q_2} = 0), (\frac{\partial B_{33}}{\partial q_3} = 0)) \quad (29)$$

$$(30)$$

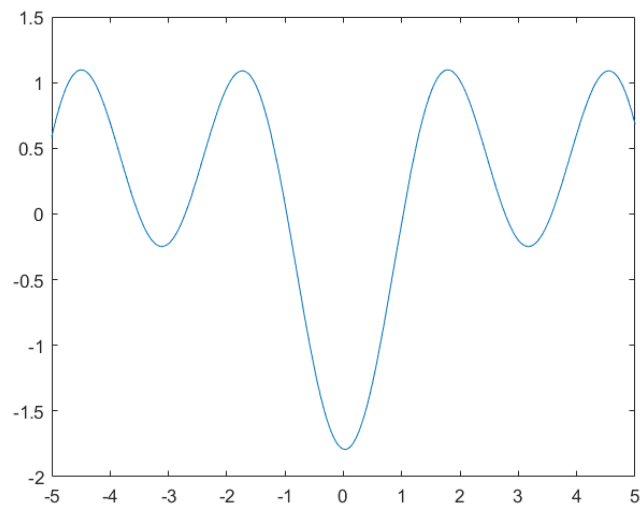
Due to the fact that the trocar, joint  $q_3$ , lives on an axis with no bounds a nominal stable value for it must be elected. The chosen value was  $q_{3,nominal} = 0.5$  representing the fact that the trocar cannot extend further than 50 cm at any time. Moreover, it was verified that  $B_{11}$  did not depend on  $q_1$ ,  $B_{22}$  on  $q_1$  and  $q_2$  and  $B_{33}$  was constant. This aligns with the theoretical model as the maximum inertia felt at joint  $q_1$  depends on how far the center of mass of joint  $q_2$  and the trocar are, the maximum inertia at joint  $q_2$  only depends on how far the trocar, joint  $q_3$ , can travel while the maximum inertia at joint  $q_3$  does not depend on the configuration of the joints of the rest of the DaVinci. Every non-dependency of  $B_{ii}$  from joint  $q_i$  leads to an unusable derivative value, which was ignored in the calculation of the maximum value for each diagonal element.

In accordance to the above assumptions, the only  $B_{ii}$  that requires deeper analysis on the choice of value of joint  $q_i$ , is  $B_{11}$ , as  $B_{22}$  only depends on the choice of nominal value  $q_3$ , and  $\max(B_{33}) = B_{33}$ . Below is a representation of  $\frac{\partial B_{11}}{\partial q_2}$ , with  $q_{3,nominal} = 0.5$ , obtained through the command "fplot(jacobian(simplify(subs(B(1,1),[q2 q5],[q2 0.5])),q2))" at the **Worst Inertia.m** function of MATLAB.



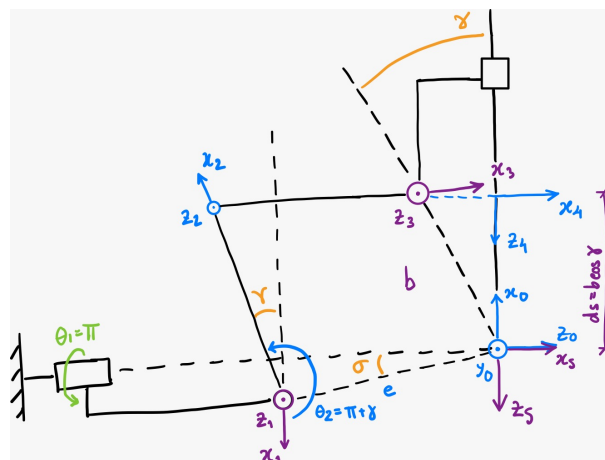
**Figure 7:** Derivative Values for  $q_2$

Even with the second derivative it can be confirmed that  $q_2 = 0$  is a maximum for  $B_{11}$ ,



**Figure 8:** Second Derivative in order of  $q_2$  for  $B_{11}$

From here it is extrapolated that for a value of  $q_2 = 0$  rad,  $B_{11}$  is at its maximum which creates the following configuration, same as Home,



**Figure 9:** Worst Inertia Configuration for  $B_{11}$

Hence,

$$B_{\max} = \begin{pmatrix} 21.93 \\ 0.68 \\ 3 \end{pmatrix} \quad (31)$$

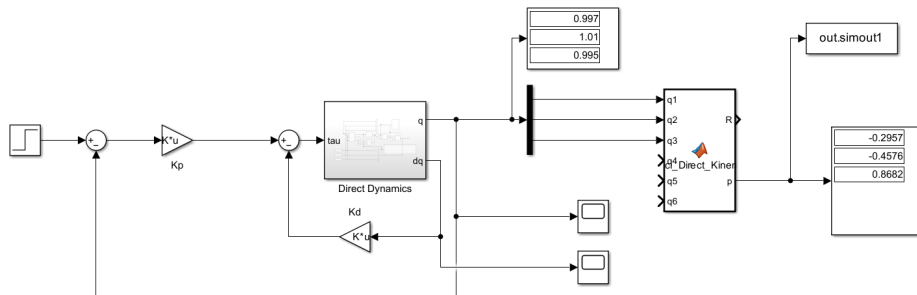
With the worst inertia values calculated, and before trying out the trajectory as defined in 5, some tests can be ran with the decentralized controller to check its overall performance. For this, the damping ratio were set to 1 (critically damped) and frequency was varied (40, 50 and 100Hz). In this case, the objective will be to critically dampen the system isolating it for a good frequency analysis ( $\zeta=1$ ), by picking which frequency has the smaller tracking error and overshoot.

$\omega_n/\text{Hz}$	$\zeta$
40	1
50	1
100	1

**Table 5:** Frequency of Ondulation and Dampening Parameters

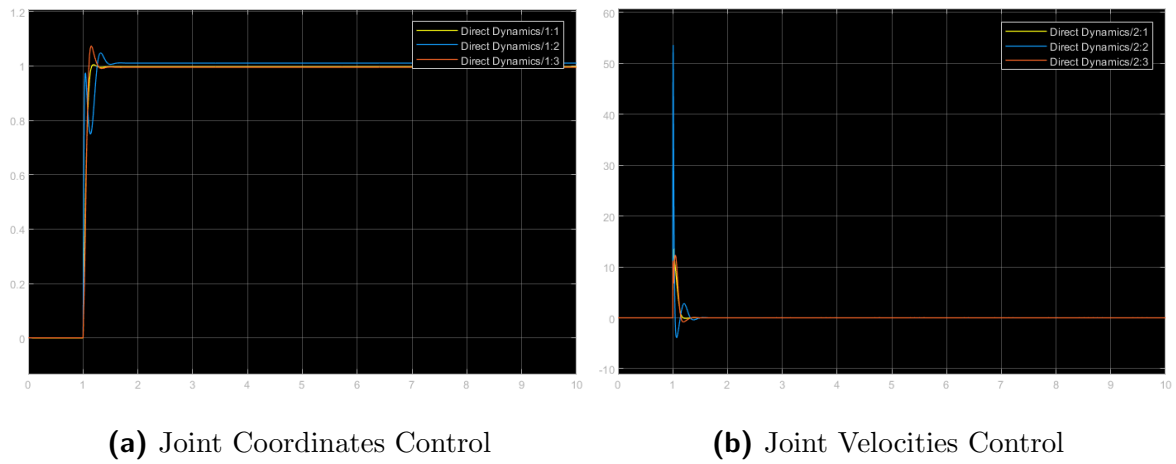
### 3.1 Simulation Pre-Validation

Starting off with the system architecture, the controller was desgined as such,

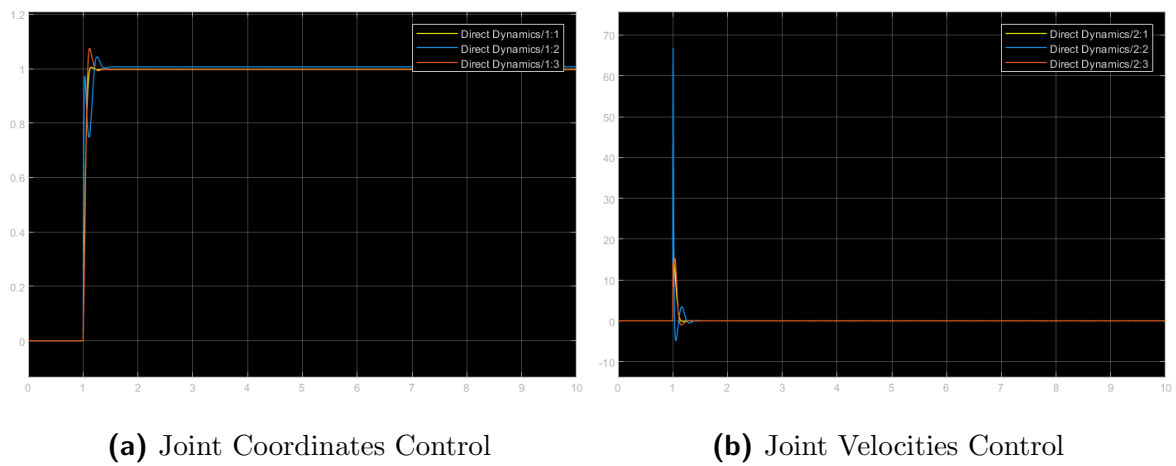


**Figure 10:** Decentralized Controller Architecture

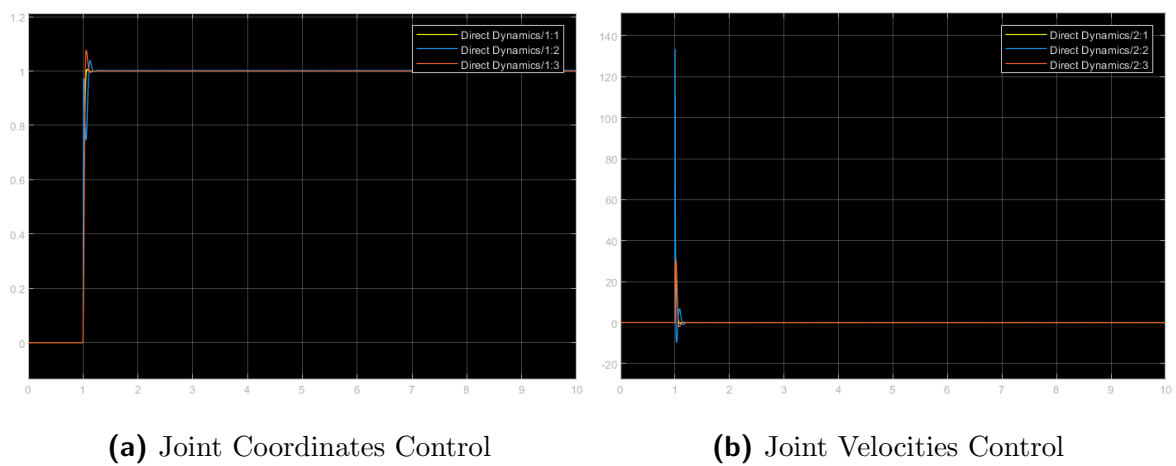
As for the desired joint coordinates,  $q_d$ , they were set as a step function of amplitude 1. As for the axis of the graphs below, the x-axis represent time. On the coordinate space, y-axis represent radians for joints  $q_1$  - yellow - and  $q_2$  - blue - and meters for joint  $q_3$  - orange. On the velocity space the y-axis represent rad/s for joints  $q_1$  and  $q_2$  and m/s for joints  $q_3$ .



**Figure 11:** Control for  $\omega_n = 40\text{Hz}$



**Figure 12:** Control for  $\omega_n = 50\text{Hz}$



**Figure 13:** Control for  $\omega_n = 100\text{Hz}$

Through analysis of the graphs above, the best frequency for the decentralized controller is

50Hz has it has smaller overshoot and tracking error both in the coordinate and velocity space.

## 4 Question 4

As for the centralized controller the joints are controlled in a different degree than before. When large operational speeds are needed for the handling of a manipulator, the nonlinear coupling terms strongly influence system performance. Contradicting the logic applied for the decentralized controller, considering the effects of the components of the DaVinci as a disturbance could originate large tracking errors. Hence when designing the centralized controller, the entire dynamical model of the manipulator should be taken into consideration.

This shifts our control problem to the context of nonlinear multivariable systems, as this control approach leads to a nonlinear centralized control architecture.

Ergo to control a manipulator whose dynamics are,

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau \longrightarrow B\ddot{q} + n = \tau \quad (32)$$

with a control action,

$$\tau = \hat{n} + \hat{B}u \quad (33)$$

then,

$$B\ddot{q} + n = \hat{n} + \hat{B}u \quad (34)$$

which for an exact estimate of  $\hat{B}$  and  $\hat{n}$  leads to a reduced dynamical equation,

$$\ddot{q} = u \quad (35)$$

With the input defined comes the definition for the rest of the linearized controller,

$$\ddot{e} + K_d\dot{e} + K_p e = 0, \text{ with, } e = q - q_d \quad (36)$$

yielding,

$$u = \ddot{q} = \ddot{q}_d - K_d(\dot{q} - \dot{q}_d) - K_p(q - q_d), q \in \mathbb{R}^n, K_d \in \mathbb{R}^{n \times n} \text{ and } K_p \in \mathbb{R}^{n \times n} \quad (37)$$

Defining  $e$  is to say that  $e$  is,

$$e = \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} = \begin{pmatrix} q_1 - q_{d1} \\ q_2 - q_{d2} \end{pmatrix} \quad (38)$$

which computing from 36 gives,

$$\ddot{e}_i + K_{d_i}\dot{e}_i + K_{p_i}e_i = 0 \quad (39)$$

Taking the Laplace transform of the linearization,

$$E_i(s)(s^2 + K_{d_i}s + K_{p_i}) = 0 \quad (40)$$

which written in regards to the frequency,  $\omega_n$ , and damping ratio,  $\zeta$  equates to,

$$E_i(s)(s^2 + \underbrace{2\omega_{n_i}\zeta_i}_{K_{d_i}}s + \underbrace{\omega_{n_i}^2}_{K_{p_i}}) = 0 \quad (41)$$

as this is a  $2^{nd}$  order system.

Concluding our controller, the dynamics of the DaVinci manipulator become,

$$\tau = \hat{n} + \hat{B}(\ddot{q}_d - K_d(\dot{q} - \dot{q}_d) - K_p(q - q_d)) \quad (42)$$

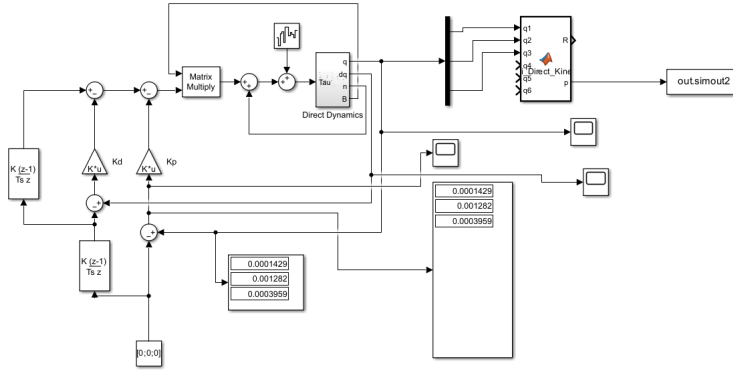
Analog to what was done previously, before jumping to the chosen trajectory, 5, the controller can be tested to check if it is performing to its standards. For the frequency and damping ratio, the combinations will be the same as before,

$\omega_n/\text{Hz}$	$\zeta$
40	1
50	1
100	1

**Table 6:** Frequency of Ondulation and Dampening Parameters

## 4.1 Simulation Pre-Validation

Starting off with the system architecture, the controller was designed as such,

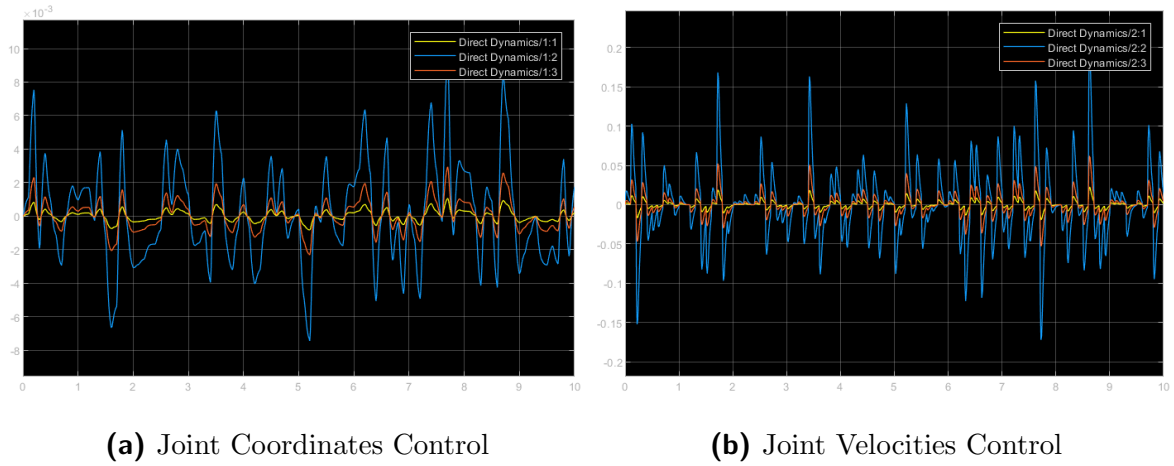


**Figure 14:** Centralized Controller Architecture

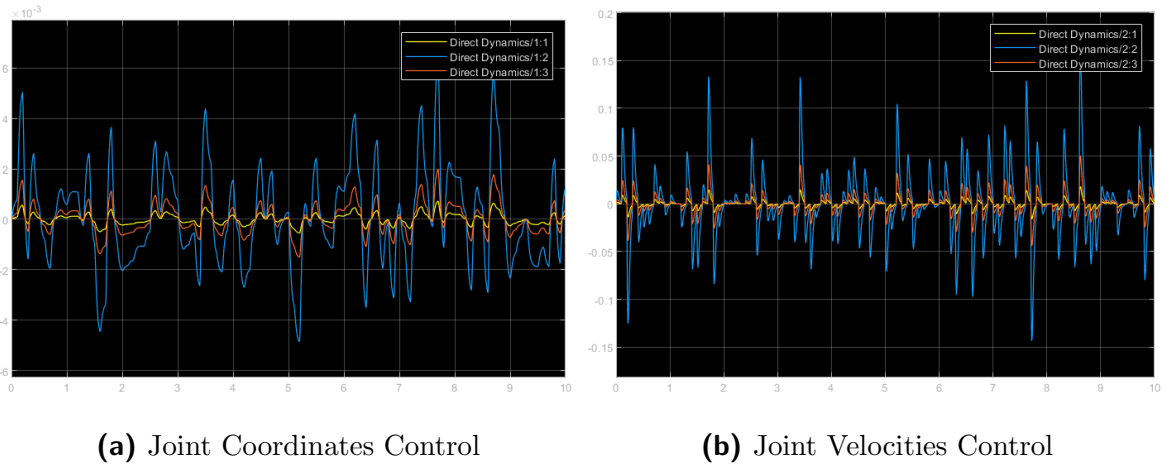
As for the desired joint coordinates,  $q_d$ , they were set as the origin of the referential,  $q_d = [0, 0, 0]^T$ . To verify the useful behavior of the controller,  $\hat{B} \neq B$  and  $\hat{n} \neq n$ . The difference between both equations can come from two different sources:

- Create a new Direct Dynamics model where the mass of the links are  $x\%$  (i.e 10%) than the ones from the standard Direct Dynamics model;
- Add a Band-limited White Noise, i.e of 0.1;

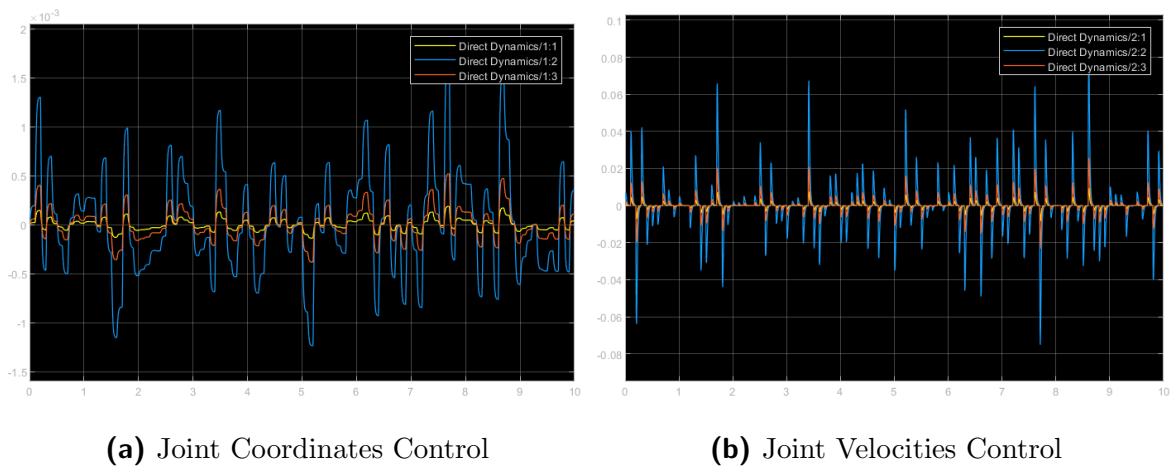
The group selected the second option with the white gaussian noise. Moreover, as it can be seen in the image above, 14, in order to obtain  $\ddot{q}$ , the group utilized the discrete-derivative block as to avoid computation errors of simulink. As for the axis of the graphs below, the x-axis represent time. On the coordinate space, y-axis represent radians for joints  $q_1$  - yellow - and  $q_2$  - blue - and meters for joint  $q_3$  - orange. On the velocity space the y-axis represent rad/s for joints  $q_1$  and  $q_2$  and m/s for joints  $q_3$ .



**Figure 15:** Control for  $\omega_n = 40\text{Hz}$  and  $\zeta = 1$



**Figure 16:** Control for  $\omega_n = 50\text{Hz}$  and  $\zeta = 1$



**Figure 17:** Control for  $\omega_n = 100\text{Hz}$  and  $\zeta = 1$

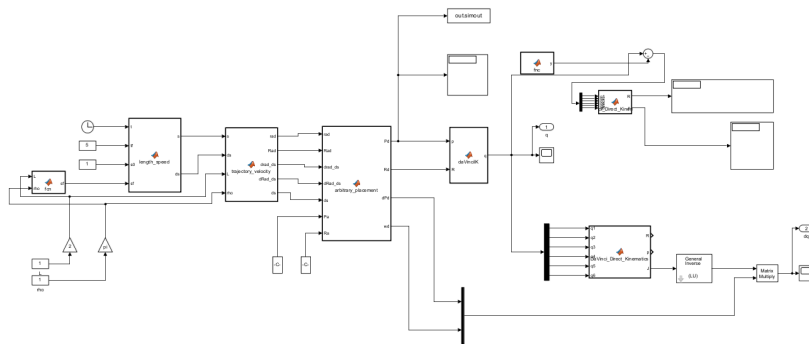
Through quick analysis of the graphs above, the best frequency for the centralized controller



is 50Hz, as there is smaller overshoot and tracking error both in the coordinate space and the velocity space.

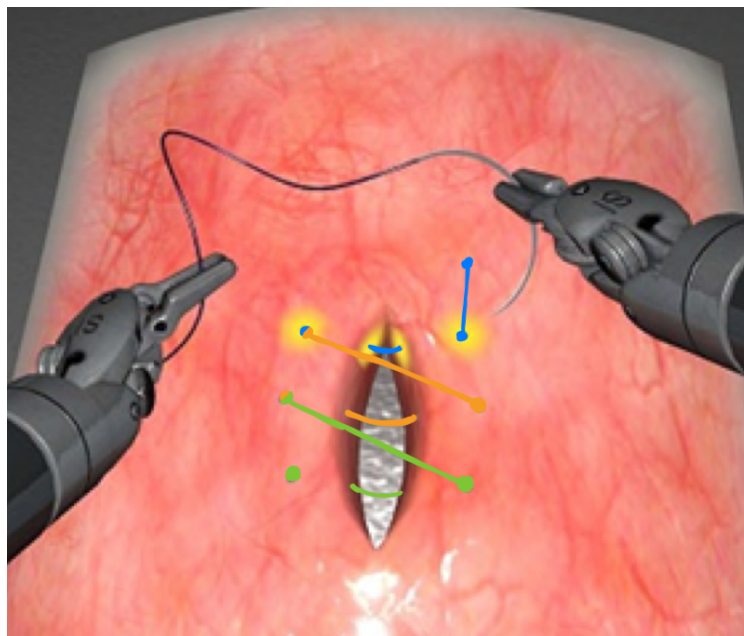
## 5 Question 5

For this section, a trajectory for the DaVinci Xi was chosen such that it was similar to a real trajectory the robot would follow in surgery. Below is the system architecture,



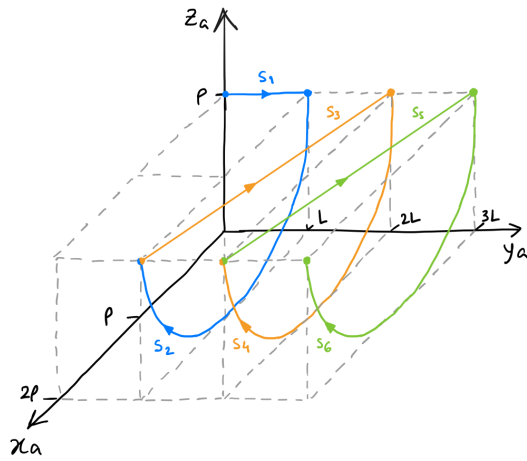
**Figure 18:** Trajectory Planning Architecture

The chosen trajectory was a suture trajectory as seen below,

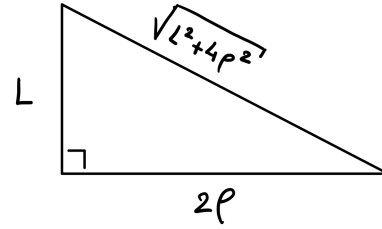


**Figure 19:** Suture Trajectory on Abdominal Tissue

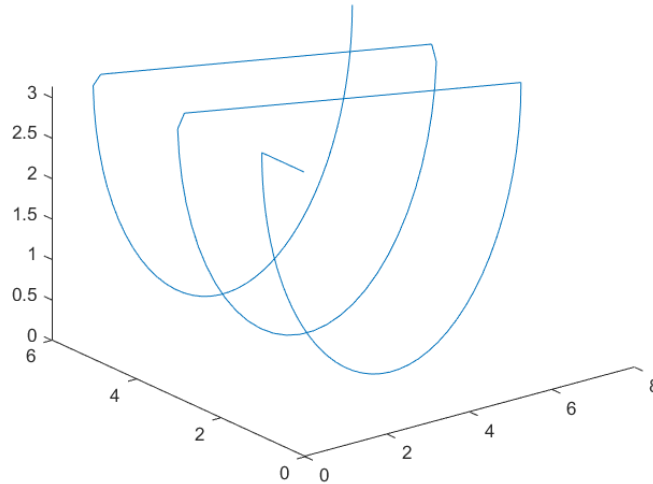
It is possible to map the suture trajectory onto a 3D cartesian referential,



(a) Translation through Sub-Paths

(b) Hypotenuse Calculation for  $s_3$  and  $s_5$ **Figure 20:** Suture Trajectory represented through Sub-Paths

and MATLAB,

**Figure 21:** Suture Trajectory in MATLAB

From here, the correct position  $r_{\alpha, d_i}^\alpha(s_i)$ , and orientation,  $R_{d_i}^\alpha(s_i)$ , matrices for each sub-path,  $P_{d_1}(s_1)$ ,  $P_{d_2}(s_2)$ ,  $P_{d_3}(s_3)$ ,  $P_{d_4}(s_4)$ ,  $P_{d_5}(s_5)$  and  $P_{d_6}(s_6)$  can be extracted in order to correctly parametrize the DaVinci Xi trajectory in space.

Starting off with the position matrices, they can be defined as follows,

$$\left\{ \begin{array}{ll} r_{\alpha,d_1}^\alpha(s_1) = \begin{pmatrix} 0 \\ s_1 \\ \rho \end{pmatrix} & 0 \leq s_1 \leq L \\ r_{\alpha,d_2}^\alpha(s_2) = \begin{pmatrix} \rho(1 - \cos(\frac{s_2}{\rho})) \\ L \\ \rho(1 - \sin(\frac{s_2}{\rho})) \end{pmatrix} & 0 \leq s_2 \leq \rho\pi \\ r_{\alpha,d_3}^\alpha(s_3) = \begin{pmatrix} 2\rho(1 - \frac{s_3}{\sqrt{(L^2+4\rho^2)}} \\ L(1 + \frac{s_3}{\sqrt{(L^2+4\rho^2)}}) \\ \rho \end{pmatrix} & 0 \leq s_3 \leq \sqrt{(L^2+4\rho^2)} \\ r_{\alpha,d_4}^\alpha(s_4) = \begin{pmatrix} \rho(1 - \cos(\frac{s_4}{\rho})) \\ 2L \\ \rho(1 - \sin(\frac{s_4}{\rho})) \end{pmatrix} & 0 \leq s_4 \leq \rho\pi \\ r_{\alpha,d_5}^\alpha(s_5) = \begin{pmatrix} 2\rho(1 - \frac{s_5}{\sqrt{(L^2+4\rho^2)}} \\ L(2 + \frac{s_5}{\sqrt{(L^2+4\rho^2)}}) \\ \rho \end{pmatrix} & 0 \leq s_5 \leq \sqrt{(L^2+4\rho^2)} \\ r_{\alpha,d_6}^\alpha(s_6) = \begin{pmatrix} \rho(1 - \cos(\frac{s_6}{\rho})) \\ 3L \\ \rho(1 - \sin(\frac{s_6}{\rho})) \end{pmatrix} & 0 \leq s_6 \leq \rho\pi \end{array} \right. \quad (43)$$

Due to the length,  $s$ , along the chosen trajectory being an amalgamation of every sub-path it is possible to obtain the following relationship between  $s$  and  $s_i$ ,

$$s_i = s - \sum_{k=1}^{i-1} l_k \quad (44)$$

where  $n$  represents the number of sub-paths, in this case 6, and  $l_k$  the length of sub-path  $k$ .

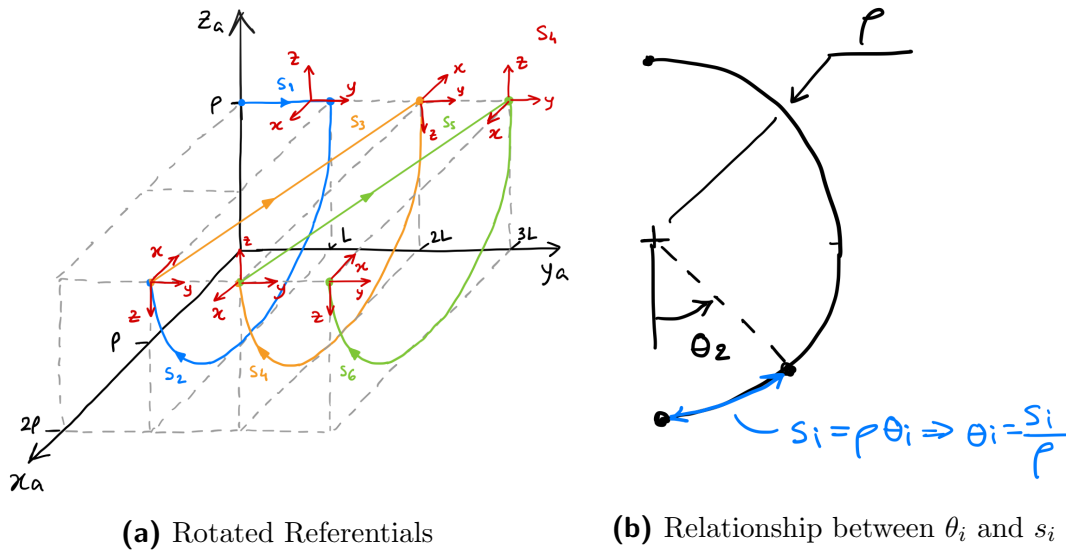
From direct observation of the image, [20a](#), it is possible to infer that,

$$\left\{ \begin{array}{l} s_1 = s \\ s_2 = s - L \\ s_3 = s - L - \rho\pi \\ s_4 = s - L - \rho\pi - \sqrt{(L^2+4\rho^2)} \\ s_5 = s - L - 2\rho\pi - \sqrt{(L^2+4\rho^2)} \\ s_6 = s - L - 2\rho\pi - 2\sqrt{(L^2+4\rho^2)} \end{array} \right. \quad (45)$$

hence,

$$\left\{ \begin{array}{ll} r_{\alpha,d_1}^\alpha(s_1) = \begin{pmatrix} 0 \\ s \\ \rho \end{pmatrix} & 0 \leq s \leq L \\ r_{\alpha,d_2}^\alpha(s_2) = \begin{pmatrix} \rho(1 - \cos(\frac{s-L}{\rho})) \\ L \\ \rho(1 - \sin(\frac{s-L}{\rho})) \end{pmatrix} & L \leq s \leq \rho\pi + L \\ r_{\alpha,d_3}^\alpha(s_3) = \begin{pmatrix} 2\rho(1 - \frac{s-L-\rho\pi}{\sqrt{(L^2+4\rho^2)}}) \\ L(1 + \frac{s-L-\rho\pi}{\sqrt{(L^2+4\rho^2)}}) \\ \rho \end{pmatrix} & L + \rho\pi \leq s \leq L + \rho\pi + \sqrt{(L^2+4\rho^2)} \\ r_{\alpha,d_4}^\alpha(s_4) = \begin{pmatrix} \rho(1 - \cos(\frac{s-L-\rho\pi-\sqrt{(L^2+4\rho^2)}}{\rho})) \\ 2L \\ \rho(1 - \sin(\frac{s-L-\rho\pi-\sqrt{(L^2+4\rho^2)}}{\rho})) \end{pmatrix} & L + \rho\pi + \sqrt{(L^2+4\rho^2)} \leq s \leq L + 2\rho\pi + \sqrt{(L^2+4\rho^2)} \\ r_{\alpha,d_5}^\alpha(s_5) = \begin{pmatrix} 2\rho(1 - \frac{s-L-2\rho\pi-\sqrt{(L^2+4\rho^2)}}{\sqrt{(L^2+4\rho^2)}}) \\ L(2 + \frac{s-L-2\rho\pi-\sqrt{(L^2+4\rho^2)}}{\sqrt{(L^2+4\rho^2)}}) \\ \rho \end{pmatrix} & L + 2\rho\pi + \sqrt{(L^2+4\rho^2)} \leq s \leq L + 2\rho\pi + 2\sqrt{(L^2+4\rho^2)} \\ r_{\alpha,d_6}^\alpha(s_6) = \begin{pmatrix} \rho(1 - \cos(\frac{s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)}}{\rho})) \\ 3L \\ \rho(1 - \sin(\frac{s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)}}{\rho})) \end{pmatrix} & L + 2\rho\pi + 2\sqrt{(L^2+4\rho^2)} \leq s \leq L + 3\rho\pi + 2\sqrt{(L^2+4\rho^2)} \end{array} \right. \quad (46)$$

From here, comes the calculation of the six rotation referentials,



**Figure 22:** Suture Trajectory represented through Rotated Referentials

where for paths  $s_1$ ,  $s_3$  and  $s_5$  the following rotation matrices are verified,

$$R_{d_1}^\alpha(s_1) = R_{d_5}^\alpha(s_5) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (47)$$

$$R_{d_3}^\alpha(s_3) = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad (48)$$

while for sub-paths  $s_2$ ,  $s_4$  and  $s_6$  three other rotation matrices, representing the semi-circumference trajectory, are defined,

$$R_{d_2}^\alpha(s_2) = \begin{pmatrix} \cos(\frac{s_2}{\rho}) & 0 & -\sin(\frac{s_2}{\rho}) \\ 0 & 1 & 0 \\ \sin(\frac{s_2}{\rho}) & 0 & \cos(\frac{s_2}{\rho}) \end{pmatrix} \quad (49)$$

$$R_{d_4}^\alpha(s_4) = \begin{pmatrix} -\cos(\frac{s_4}{\rho}) & 0 & -\sin(\frac{s_4}{\rho}) \\ 0 & 1 & 0 \\ -\sin(\frac{s_4}{\rho}) & 0 & -\cos(\frac{s_4}{\rho}) \end{pmatrix} \quad (50)$$

$$R_{d_6}^\alpha(s_6) = \begin{pmatrix} \cos(\frac{s_6}{\rho}) & 0 & -\sin(\frac{s_6}{\rho}) \\ 0 & 1 & 0 \\ \sin(\frac{s_6}{\rho}) & 0 & \cos(\frac{s_6}{\rho}) \end{pmatrix} \quad (51)$$

It is possible to verify the rotation matrices for sub-paths  $s_2$ ,  $s_4$  and  $s_6$  through the following computations,

$$R_{d_2}^\alpha(0) = R_{d_1}^\alpha(s_1) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, R_{d_2}^\alpha(\rho\pi) = R_{d_3}^\alpha(s_3) = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

$$R_{d_4}^\alpha(0) = R_{d_3}^\alpha(s_3) = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, R_{d_4}^\alpha(\rho\pi) = R_{d_5}^\alpha(s_5) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R_{d_6}^\alpha(0) = R_{d_5}^\alpha(s_5) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Writing the rotation matrices in function of the whole path length leads to,

$$\left\{ \begin{array}{ll} R_{d_1}^\alpha(s_1) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & 0 \leq s \leq L \\ R_{d_2}^\alpha(s_2) = \begin{pmatrix} \cos(\frac{s-L}{\rho}) & 0 & -\sin(\frac{s-L}{\rho}) \\ 0 & 1 & 0 \\ \sin(\frac{s-L}{\rho}) & 0 & \cos(\frac{s-L}{\rho}) \end{pmatrix} & L \leq s \leq \rho\pi + L \\ R_{d_3}^\alpha(s_3) = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} & L + \rho\pi \leq s \leq L + \rho\pi + \sqrt{(L^2 + 4\rho^2)} \\ R_{d_4}^\alpha(s_4) = \begin{pmatrix} -\cos(\frac{s-L-\rho\pi-\sqrt{(L^2+4\rho^2)}}{\rho}) & 0 & -\sin(\frac{s-L-\rho\pi-\sqrt{(L^2+4\rho^2)}}{\rho}) \\ 0 & 1 & 0 \\ -\sin(\frac{s-L-\rho\pi-\sqrt{(L^2+4\rho^2)}}{\rho}) & 0 & -\cos(\frac{s-L-\rho\pi-\sqrt{(L^2+4\rho^2)}}{\rho}) \end{pmatrix} & L + \rho\pi + \sqrt{(L^2 + 4\rho^2)} \leq s \leq L + 2\rho\pi + \sqrt{(L^2 + 4\rho^2)} \\ R_{d_5}^\alpha(s_5) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & L + 2\rho\pi + \sqrt{(L^2 + 4\rho^2)} \leq s \leq L + 2\rho\pi + 2\sqrt{(L^2 + 4\rho^2)} \\ R_{d_6}^\alpha(s_6) = \begin{pmatrix} \cos(\frac{s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)}}{\rho}) & 0 & -\sin(\frac{s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)}}{\rho}) \\ 0 & 1 & 0 \\ \sin(\frac{s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)}}{\rho}) & 0 & \cos(\frac{s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)}}{\rho}) \end{pmatrix} & L + 2\rho\pi + 2\sqrt{(L^2 + 4\rho^2)} \leq s \leq L + 3\rho\pi + 2\sqrt{(L^2 + 4\rho^2)} \end{array} \right. \quad (52)$$

With this, the mapping of the trajectory defined in reference frame  $\alpha$  is done. This allows for the following representation,

$$R_d^o(s) = R_\alpha^o R_d^\alpha(s) \quad (53)$$

$$p_d^o(s) = p_\alpha^o + R_\alpha^o r_{\alpha,d}^o(s) \quad (54)$$

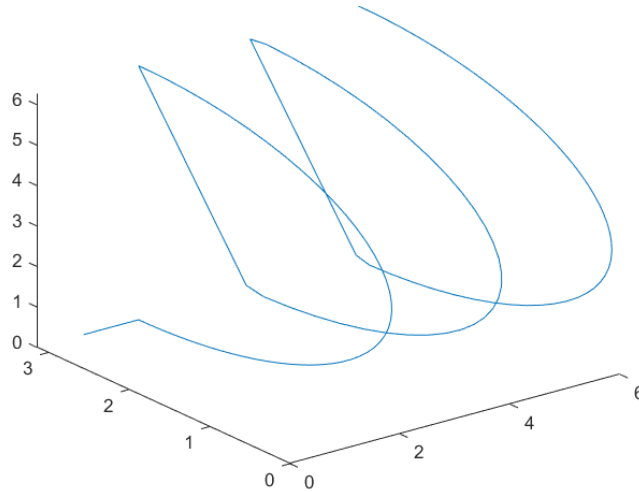
lending the ability to orient the frame freely, through  $R_\alpha$  and  $p_\alpha$ .

In order to calculate both the linear and angular velocities, the below equations, 55, 56, can be used

$$\dot{p}_d(s, \dot{s}) = R_\alpha \dot{r}_{\alpha,d}^\alpha(s) = R_\alpha \frac{\partial}{\partial s} [r_{\alpha,d}^\alpha(s)] \cdot \dot{s} \quad (55)$$

$$\omega_d(s, \dot{s}) = \text{vect}(\dot{R}_d(s) R_d^T(s)) = \text{vect}\left(R_\alpha \frac{\partial R_d^\alpha(s)}{\partial S} R_d^\alpha(s)^T R_\alpha^T\right) \dot{s} \quad (56)$$

Here  $\dot{R}_\alpha = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$  and  $\dot{p}_\alpha = \begin{pmatrix} -0,15 \\ 0 \\ 0 \end{pmatrix}$ . This is to represent the rotation and translation along the reference 0, at the base of the manipulator, to the beginning of our suture trajectory.



**Figure 23:** Trajectory as Rotated and Translated in regards to Reference 0

$$\frac{\partial}{\partial s}[r_{\alpha,d}^{\alpha}(s)] = \begin{cases} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} & 0 \leq s \leq L \\ \begin{pmatrix} \sin \frac{s-L}{\rho} \\ 0 \\ -\cos \frac{s-L}{\rho} \end{pmatrix} & L \leq s \leq L + \rho\pi \\ \begin{pmatrix} -\frac{2\rho}{\sqrt{(L^2+4\rho^2)}} \\ \frac{L}{\sqrt{(L^2+4\rho^2)}} \\ 0 \end{pmatrix} & L + \rho\pi \leq s \leq 2L + \rho\pi \\ \begin{pmatrix} \sin \left( \frac{s-L-\rho\pi-\sqrt{(L^2+4\rho^2)}}{\rho} \right) \\ 0 \\ -\cos \left( \frac{s-L-\rho\pi-\sqrt{(L^2+4\rho^2)}}{\rho} \right) \end{pmatrix} & L + \rho\pi + \sqrt{(L^2+4\rho^2)} \leq s \leq L + 2\rho\pi + \sqrt{(L^2+4\rho^2)} \\ \begin{pmatrix} -\frac{2\rho}{\sqrt{(L^2+4\rho^2)}} \\ \frac{L}{\sqrt{(L^2+4\rho^2)}} \\ 0 \end{pmatrix} & L + 2\rho\pi + \sqrt{(L^2+4\rho^2)} \leq s \leq L + 2\rho\pi + 2\sqrt{(L^2+4\rho^2)} \\ \begin{pmatrix} \sin \left( \frac{s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)}}{\rho} \right) \\ 0 \\ -\cos \left( \frac{s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)}}{\rho} \right) \end{pmatrix} & L + 2\rho\pi + 2\sqrt{(L^2+4\rho^2)} \leq s \leq L + 3\rho\pi + 2\sqrt{(L^2+4\rho^2)} \end{cases} \quad (57)$$

$$\frac{\partial}{\partial s}[R_d^{\alpha}(s)] = \begin{cases} \begin{pmatrix} -\frac{1}{\rho} \sin \frac{(s-L)}{\rho} & 0 & -\frac{1}{\rho} \cos \frac{(s-L)}{\rho} \\ 0 & 0 & 0 \\ \frac{1}{\rho} \cos \frac{(s-L)}{\rho} & 0 & -\frac{1}{\rho} \sin \frac{(s-L)}{\rho} \end{pmatrix} & 0 \leq s \leq L \\ \begin{pmatrix} \frac{1}{\rho} \sin \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & \frac{1}{\rho} \cos \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} \\ 0 & 0 & 0 \\ -\frac{1}{\rho} \cos \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & \frac{1}{\rho} \sin \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} \end{pmatrix} & L \leq s \leq L + \rho\pi \\ \begin{pmatrix} -\frac{1}{\rho} \sin \frac{(s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & -\frac{1}{\rho} \cos \frac{(s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)})}{\rho} \\ 0 & 0 & 0 \\ \frac{1}{\rho} \cos \frac{(s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & -\frac{1}{\rho} \sin \frac{(s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)})}{\rho} \end{pmatrix} & L + \rho\pi \leq s \leq L + \rho\pi + \sqrt{(L^2+4\rho^2)} \\ \begin{pmatrix} \frac{1}{\rho} \sin \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & \frac{1}{\rho} \cos \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} \\ 0 & 0 & 0 \\ -\frac{1}{\rho} \cos \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & \frac{1}{\rho} \sin \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} \end{pmatrix} & L + \rho\pi + \sqrt{(L^2+4\rho^2)} \leq s \leq L + 2\rho\pi + \sqrt{(L^2+4\rho^2)} \\ \begin{pmatrix} -\frac{1}{\rho} \sin \frac{(s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & -\frac{1}{\rho} \cos \frac{(s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)})}{\rho} \\ 0 & 0 & 0 \\ \frac{1}{\rho} \cos \frac{(s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & -\frac{1}{\rho} \sin \frac{(s-L-2\rho\pi-2\sqrt{(L^2+4\rho^2)})}{\rho} \end{pmatrix} & L + 2\rho\pi + \sqrt{(L^2+4\rho^2)} \leq s \leq L + 2\rho\pi + 2\sqrt{(L^2+4\rho^2)} \\ \begin{pmatrix} \frac{1}{\rho} \sin \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & \frac{1}{\rho} \cos \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} \\ 0 & 0 & 0 \\ -\frac{1}{\rho} \cos \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} & 0 & \frac{1}{\rho} \sin \frac{(s-L-\rho\pi-\sqrt{(L^2+4\rho^2)})}{\rho} \end{pmatrix} & L + 2\rho\pi + 2\sqrt{(L^2+4\rho^2)} \leq s \leq L + 3\rho\pi + 2\sqrt{(L^2+4\rho^2)} \end{cases} \quad (58)$$

To represent the the length of the path as a function,

$$s(t) = \begin{cases} s_0 + \frac{3}{t_f^2}(s_f - s_0)t^2 - \frac{2}{t_f^3}(s_f - s_0)t^3 & 0 \leq t \leq t_f \\ s_f & \text{otherwise} \end{cases} \quad (59)$$

$$\dot{s}(t) = \begin{cases} \frac{6}{t_f^2}(s_f - s_0)t - \frac{6}{t_f^3}(s_f - s_0)t^2 & 0 \leq t \leq t_f \\ 0 & \text{otherwise} \end{cases} \quad (60)$$

## 5.1 Decentralized and Centralized Controller Tests for Trajectory

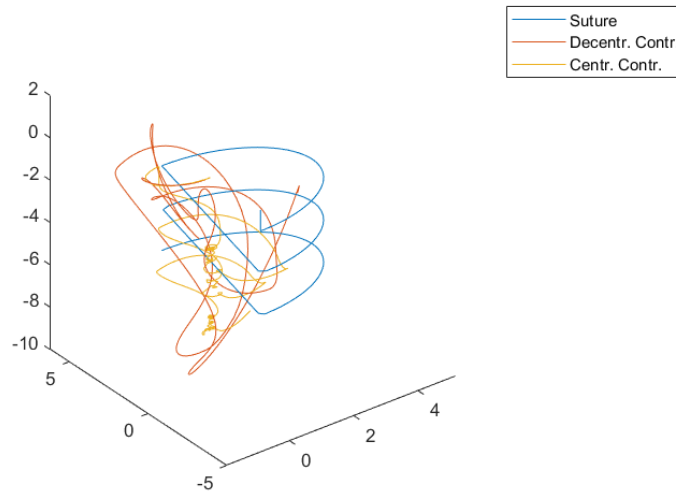
To test the performance for the centralized and decentralized controllers the speed at which the trajectory was computed through *Simulink* was altered. The main suture trajectory assumes  $L = 2\text{m}$  and  $\rho = \pi\text{m}$ . For the better interpretability of the overlap between the centralized/decentralized controller's trajectories, the suture trajectory was rotated. The axis for the trajectories as displayed represent meters. Furthermore, to compute the  $q_i$  values for the trajectory the Inverse Kinematics function block as developed in the previous assignment was utilized. On the other end, to compute  $\dot{q}_i$  the group opted to utilize the Jacobian, as

$$\dot{q}_d = J^{-1}(q_d) \cdot \begin{pmatrix} \dot{P}_d \\ \omega_d \end{pmatrix} \quad (61)$$

for the centralized controller

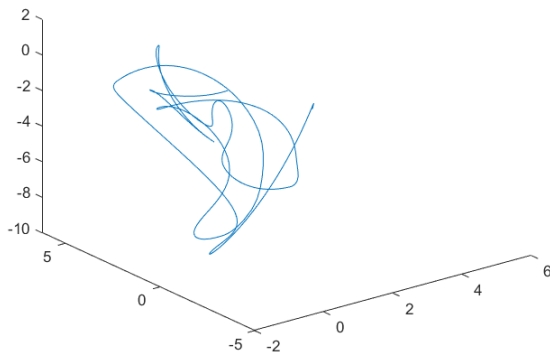
### 5.1.1 First Simulation

For the first simulation  $t_f$ , which represent the final time for our  $s$  trajectory, was set at 10 seconds, the same time frame of the simulation computation. The frequencies for the controllers were the ones determined as best, as computed previously (3.1, 4.1).

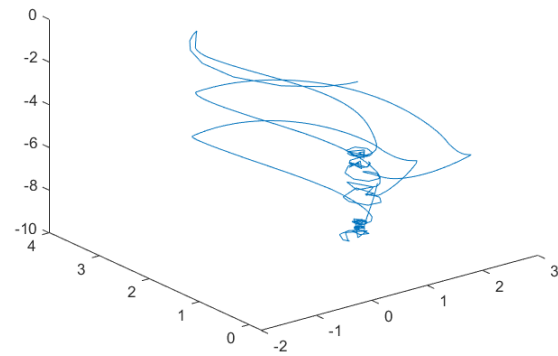


**Figure 24:** Overlapping Trajectories for  $t_f = 10$





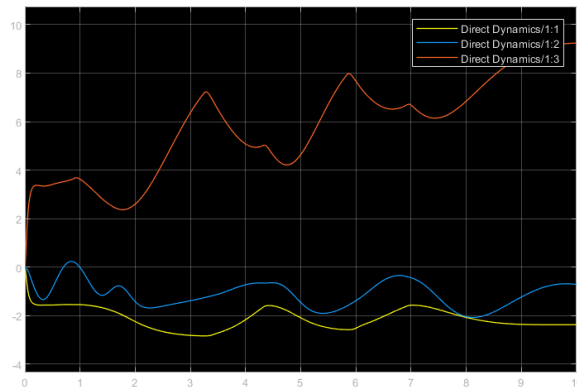
(a) Decentralized Controller



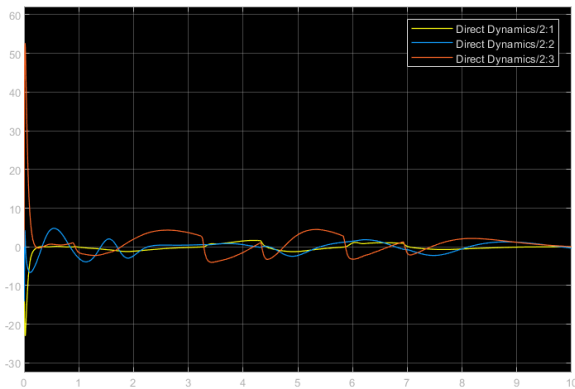
(b) Centralized Controller

**Figure 25:** Comparison of Suture Trajectories between Controllers for  $t_f = 10$ 

With the images above, it is clear that the centralized controller holds a better performance and a smaller tracking error. The image as plotted in 25b holds a better approximation of the trajectory while 25a has larger tracking error and more overshoot. Nevertheless, due to the characteristics of the suture trajectory in itself, with sudden changes in direction, the centralized controller is not perfect as it has a certain degree of tracking error and overshoot. Such can be thoroughly analyzed below, with the effect of the controllers on the joints.

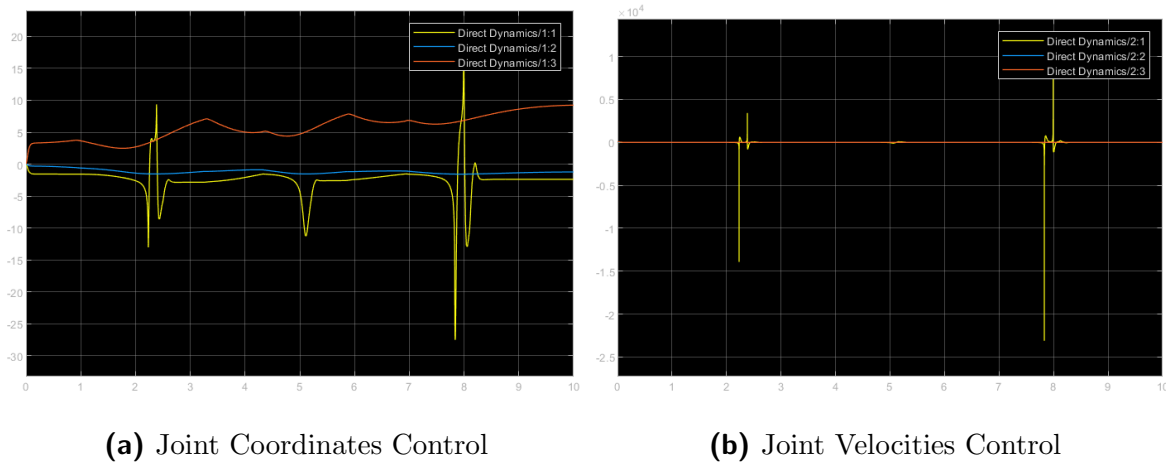


(a) Joint Coordinates Control



(b) Joint Velocities Control

**Figure 26:** Decentralized Control for  $t_f = 10$



**Figure 27:** Centralized Control for  $t_f = 10$

Analyzing now the control for both the joint coordinates and joint velocities, the controllers exhibit different behaviours. The axis for coordinate space and velocity space take the same meaning as expressed before, radians and meters for  $q_1/q_2$  and  $q_3$  respectively on the coordinate space and rad/s and m/s for  $q_1/q_2$  and  $q_3$  respectively on the velocity space.

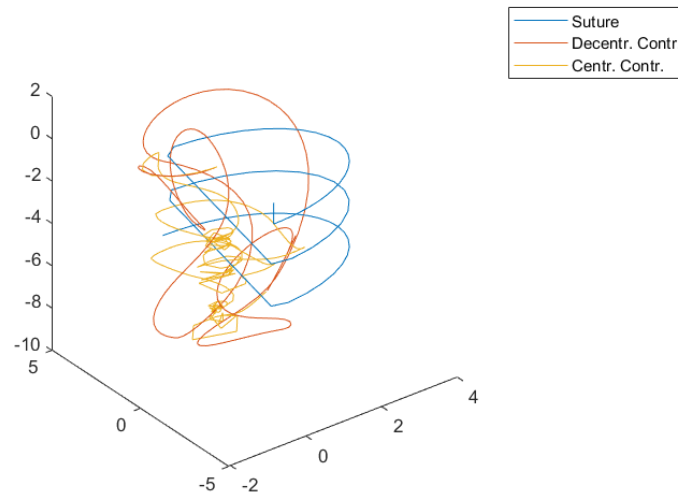
Starting off with the centralized controller, its error associated with joint  $q_2$  reaches the zero axis fast, the error associated with joint  $q_1$  has 3 large spikes stabilizing shortly after, and joint  $q_3$  never seems to reach the zero axis. As for joint  $q_1$ , its error spikes could be explained via the complexity of the proposed trajectory. Due to the nature of a suture-like trajectory, there are sudden changes in the orientation and position matrices. This could lead to an undesirable and complex configuration of the manipulator. These 3 error spikes can be seen in the velocity space as well, translating the fact that for an undesirable configuration the manipulator must rapidly increase the velocity on joint  $q_1$ , for example 10000 rad/s. As for joint  $q_3$ , its error never seems to reach zero in terms of distance traveled. However its velocity error is 0. This is due to the fact that the trocar lives on an unbounded axis, hence the choice of a nominal value,  $q_{3,nominal} = 0.5$ . Should this value be smaller, the error in coordinate space would follow suit.

Jumping to the decentralized controller, the problem with joint  $q_3$  remains the same as before. It is noted that for joint  $q_1$  there are no error spikes in the chosen time frame, however neither the velocity error nor coordinate error get close to the 0 value. As for joint  $q_2$  the control, even if useful, never reaches the accuracy of the centralized controller both in the velocity and coordinate space.

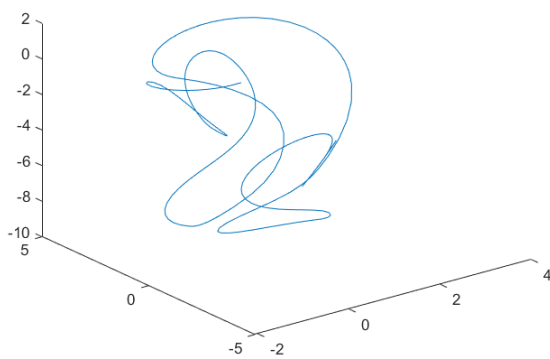
Were not for the rapid reorientation of some sub-paths, as going from  $s_1$  to  $s_2$ , it would be easier to see that the centralized controller has a higher performance. However when looking to the graphs above it is hard to conclude which controller is best, since while the centralized allows for a smaller error on joints  $q_2$  and  $q_3$  both on coordinate and velocity space, the biggest error in the decentralized controller, as the beginning of every joint velocity being very high but stabilizing shortly after, is smaller than the biggest error on the centralized controller, precisely on joint  $q_1$ .

### 5.1.2 Second Simulation

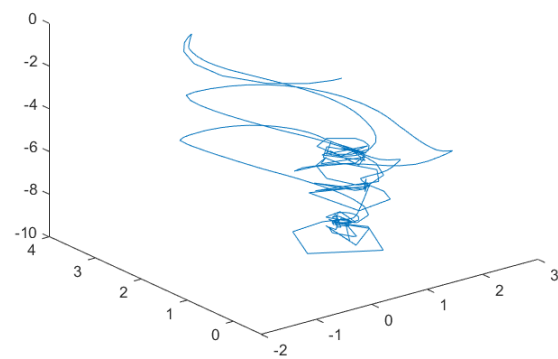
For the second simulation,  $t_f$  was set at 5 seconds, creating a direct comparison to the tests done before.



**Figure 28:** Overlapping Trajectories for  $t_f = 5$ ,  $\zeta = 0.5$



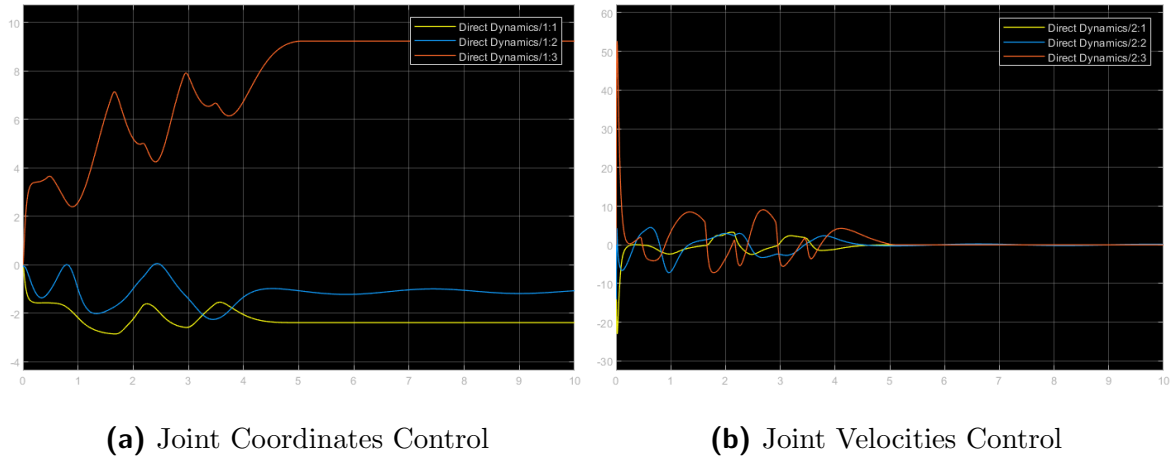
**(a)** Decentralized Controller



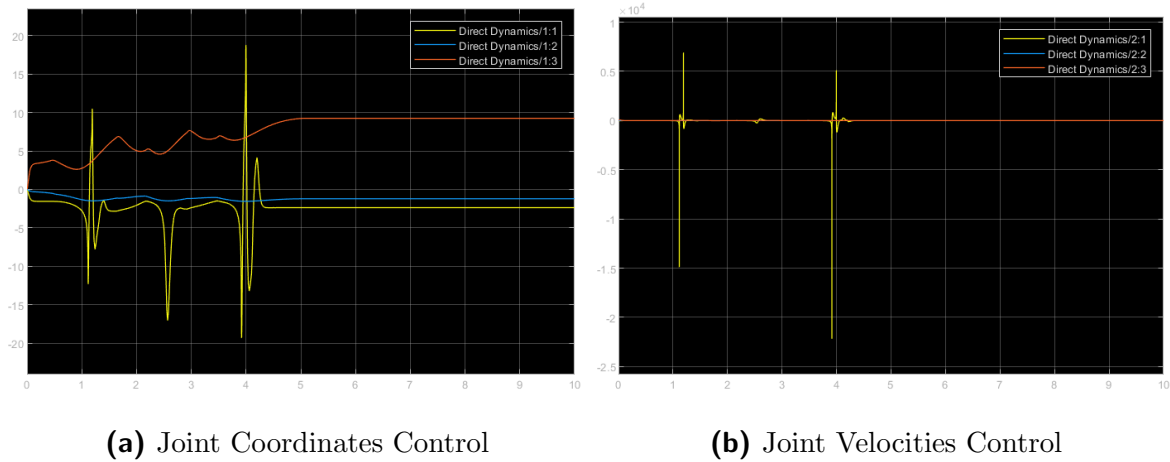
**(b)** Centralized Controller

**Figure 29:** Comparison of Suture Trajectories between Controllers for  $t_f = 5$

With this analysis, the goal was to check that if the velocity of the manipulator increased, the tracking error increased and the trajectory had smaller resolution for both the decentralized and centralized controllers. Such is verified, since, when comparing 25a and 29a, it is clear that for a smaller velocity even the decentralized controller approaches better the suture trajectory. When looking at 29b the trajectory is still clear, however the error and overshoot are higher when comparing with 25b.



**Figure 30:** Decentralized Control for  $t_f = 5$

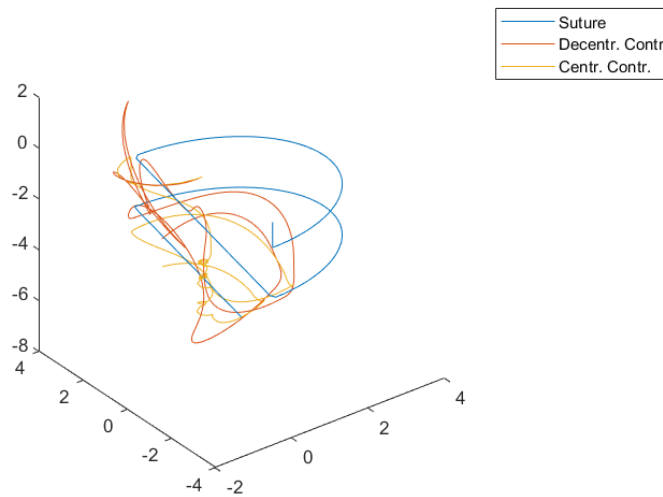


**Figure 31:** Centralized Control for  $t_f = 5$

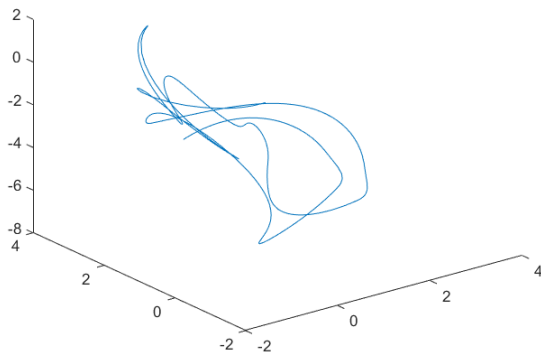
When looking at the effects of our controllers on the joints there is still a similar behaviour as described before. The  $q_3$  joint error never reaches 0 in the coordinate space, there are 3 error spikes on  $q_1$  in the centralized controller, and  $q_2$  is overall well controlled on the decentralized controller and very well on the centralized controller. In these graphs the effects of the increase in velocity show with the fact that the error spikes of joint  $q_1$  are reached faster and it takes longer to stabilize on the velocity space for the decentralized controller. This was expected as the increase in velocity creates a more unstable state for the manipulator and harder to control, while highlighting the larger errors sooner.

### 5.1.3 Third Simulation

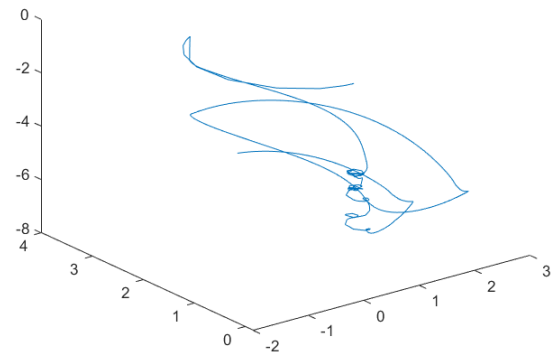
For the third simulation,  $t_f$  was set at 15 seconds as to show that for a decrease in velocity, even though the complete trajectory is not reached, there is a higher stability and path resolution throughout the whole process.



**Figure 32:** Overlapping Trajectories for  $t_f = 15$



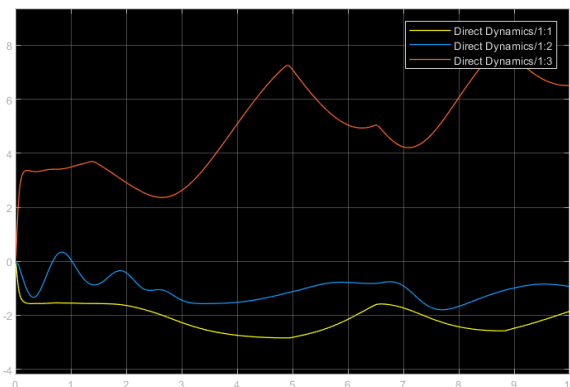
(a) Decentralized Controller



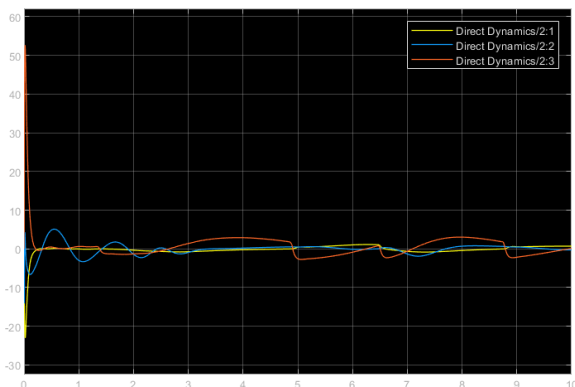
(b) Centralized Controller

**Figure 33:** Comparison of Suture Trajectories between Controllers for  $t_f = 15$ 

Such is verified as when comparing both 25 and 33 the processed trajectories are done much smoother for both the decentralized controller and the centralized controller. There is lesser tracking error and overshoot for both controllers.

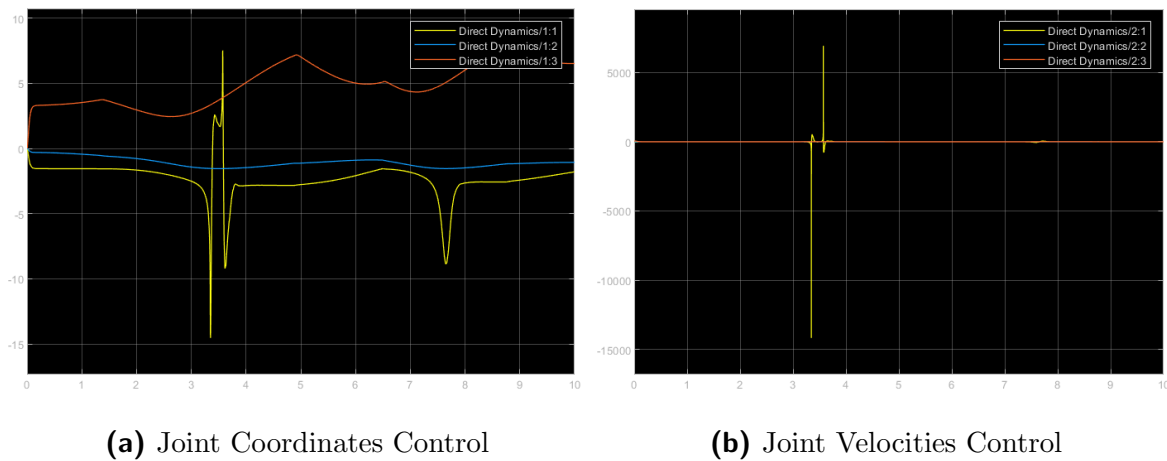


(a) Joint Coordinates Control



(b) Joint Velocities Control

**Figure 34:** Decentralized Control for  $t_f = 15$



**Figure 35:** Centralized Control for  $t_f = 15$

Lastly, looking at the joint spaces for both controllers one of the error spike of joint  $q_1$  disappears,  $q_3$  has a smaller error both on the decentralized and centralized controller, even though never stabilizing, and while the centralized controller manages to get a zero error on velocity space for  $q_3$  and  $q_2$ , the decentralized controller largest error is 50m/s while for the centralized controller is 5000 rad/s.

## 5.2 Conclusion

The hypothesis to check was, that the centralized controller had a higher performance than its counterpart. However, the weight of the chosen trajectory shows itself when looking at the graphs discussed. The sudden change in trajectories reflected on the orientation and position matrices, can sometimes not be perfectly controlled by the centralized controller leading to an erratic movement and very high rotation speeds at  $q_1$ . Nevertheless, the decentralized controller is not perfect either, as in the beginning of every velocity error space there are high errors on both joints  $q_1$  and  $q_3$ .

This leads to the conclusion that when defining the trajectory the rapid transitions of orientation and position should be programmed to be as smooth as possible, the trocar should have a lower nominal value, as to constrain it more for it to be safe on a medical environment, and there is always a trade-off between time to complete the trajectory and tracking error - the lesser the tracking error the more the time to complete the trajectory, and the smoother it will be.

## 6 References

- [1] Robotics of Manipulation - [RMan - Dynamics] Slides of Theoretical Classes 2022/2023, 2nd Semester (MEMec), by Professor Jorge M. M. Martins
- [2] Beer, F. P., & Johnston Jr., E. R. (1984). Vector Mechanics for Engineers (4th ed.). McGraw-Hill.