ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

Department of Electrical and Computer Engineering

INTRODUCTION TO ROBOTICS

Control of a Robotic Surgery Tool

Supervisor: Bechlioulis Charalampos

Students:

Kalakonis Georgios-1083846

Theodoropoulos Christos-1089860

Dandakis Theophilos-1083718

Contents

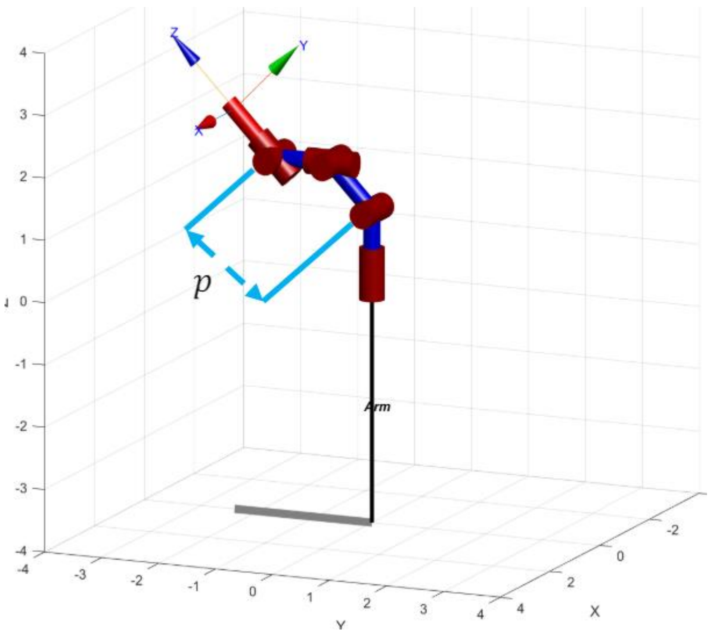
Abstract

Limiting the possible movements of a manipulator link, which is restricted to move through an entry, is a common problem in robot-assisted surgery. During these operations, long and thin instruments are attached to the robot limb. These tools are inserted inside the patient's body through tiny incisions in the patient's skin via a trocar. Each incision, fixed or moving due to breathing or heartbeat, must act as a remote center of motion (RCM). If a general-purpose operator is used, the trocar is generally not rigidly connected to the environment. Thus, in practical operations in which a human is exerting forces on the end of the robot to manipulate the tool, it is difficult to know which generalized forces will not stress the wall of the incision. Therefore, a control approach is needed to limit the tool movement so that the incision wall is not stressed by lateral forces during a physical interaction between the human and the robot. This problem is addressed by using a target dynamic model, which will be used in a control system that decouples (decouples) the robot's joint space appropriately, so that the RCM constraint is satisfied during the tool manipulation. Various experiments and simulations have taken place with different robotic manipulators, which confirm the effectiveness and efficiency of the system mentioned above for both fixed and moving intersections.

Modelling the problem

In the context of the problem, an anthropomorphic robotic arm with 6 degrees of freedom is used, to which a tool (cylindrical shape, 12 cm long) is appropriately adapted at the last joint. The tip of the tool is considered as the end-effector and the following parameters were calculated based on the Denavit-Hartenberg convention:

Link	r_i	a_i	d_i
1	0	π/2	13
2	0	0	0
3	0	π/2	0
4	0	-π/2	8
5	0	π/2	0
6	0	0	12



6 dof anthropomorphic robotic arm

In addition, the incision point will be simulated by a ring with an outer radius of 4.5 cm of negligible thickness and its position and orientation will be taken for granted.

The problem is divided into two parts. Firstly, the tool must move from its random position to the centre of the ring representing the patient's incision, with the same orientation under the constraints of not colliding with the incision walls and keeping the joint angles within the permissible limits. The tool should then enter the incision and implement translational movements satisfying the RCM (Remote Center of Motion) constraint, i.e. the tool entry point should remain stationary.

Forward kinematics

Forward kinematics analysis involves the calculation of the position and orientation of the end-effector given the joint angles. The calculation is made possible by utilizing transformation matrices as follows:

$$T^b_{end-effector} = \begin{bmatrix} n^b_{end-effector}(q) & s^b_{end-effector}(q) & a^b_{end-effector}(q) & p^b_{end-effector}(q) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T^b_{end-effector} = A^0_1(q_1)A^1_2(q_2)A^2_3(q_3)A^3_4(q_4)A^4_5(q_5)A^5_6(q_6),$$

Where according to the parameters

$$A^{i-1}_i(q_i) = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}c_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}c_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Constraints

The movement of the end-effector in the task-space is subject to various constraints. First, the angles of the joints must be within the desired limits and at the same time the nature of the problem requires the introduction of a constraint that prevents the end-effector from getting too close to the incision to avoid possible collision with it. In the case of a redundant operator, these constraints could concern secondary objectives, which would be implemented in the null space. However, because the robot-arm used has 6 degrees of freedom this is not possible to implement. Thus, the constraints will mainly be used for the correct operation of the program, as they will stop the execution by immobilizing the arm when a constraint is not satisfied.

Joint limits

$L_i = \left(\frac{q_i - \tilde{q}_i}{q_{iM} - q_{im}}\right)^2$, where

- $\tilde{q}_i$ the average value of the angle of the i joint
- q_{iM} the maximum value of the angle of joint i
- q_{im} the minimum value of the angle of joint i

Distance from the incision

The incision distance problem is equivalent to a problem of determining the distance between a line and a ring in three-dimensional space. Initially, it will be assumed that the robot arm has negligible dimensions and each line segment between successive joints will be defined using the corresponding frames.

Lets consider joint i:

Το διάνυσμα από την i άρθρωση στην i +1

The vector from the joint i to joint $i+1$

$p(s) = p_i + s\frac{(p_{i+1} - p_i)}{\|p_{i+1} - p_i\|}$ where,

- p_i, p_{i+1}, the position vectors of the i and i +1 joints respectively
- Arc length s, $0 \leq s \leq \|p_{i+1} - p_i\|$

Consider a ring of negligible thickness, with a given radius r, center C and orientation with a rotation matrix $R = [n_c \quad s_c \quad a_c]$. Furthermore, the z-axis of the circle is assumed to be perpendicular to the plane defined by the circle. In this way the points of the circle are defined parametrically as follows:

$c(t) = C + r(n_c cos(t) + s_c\, sin(t)), 0 \leq t \leq 2\pi$

Finally, we search for s, t which minimize the following distance. In the context of the problem we are mainly interested in the value of this minimum distance

$$d(s,t) = \|p(s) - c(t)\|$$

Thus, a minimization problem is defined under the following constraints

- $0 \leq s \leq \|p_{i+1} - p_i\|$

- $0 \leq t \leq 2\pi$

The minimize function of the scipy.optimize library of python is used to solve the problem, which uses the 'L-BFGS-B' algorithm.

This procedure is repeated for successive pairs of joints and finally the minimum distance is selected from these calculated distances. The code of the implementation follows:

```
def mindist(self, angle_pos, obstacle, radius, R):
    obstacle = obstacle.reshape((3,1))
    distances=[]

    positions=[self.tf_A01(angle_pos)[:3, 3], self.tf_A02(angle_pos)[:3, 3],
            self.tf_A03(angle_pos)[:3, 3], self.tf_A04(angle_pos)[:3, 3],
            self.tf_A05(angle_pos)[:3, 3], self.tf_A06(angle_pos)[:3, 3]]

    initial_guess = np.array([0, np.pi])
    for i in range(len(positions)-1):
        pi=positions[i]
        pnext=positions[i+1]
        p=pnext-pi
        if np.linalg.norm(p)>1e-3:
            args=(pi,pnext,obstacle,radius,R)
            upper_limit = np.linalg.norm(pnext - pi)
            bounds = [(0, upper_limit), (0, 2*np.pi)]
            result = minimize(self.objective_function, initial_guess, args=args,  bounds=bounds)
            distances.append(result.fun)

    if min(distances)<0.1:
        print("warning dist")
    return min(distances)


def objective_function(self,x,pi,pf,circle_center, radius,R):
    s=x[0]
    t=x[1]
    pi=pi.reshape((3,1))
    pf=pf.reshape((3,1))
    d=pi+s*(pf-pi)/np.linalg.norm(pf-pi) - circle_center -radius*(np.cos(t)*R[:3,0]+np.sin(t)*R[:3,1]).reshape((3,1))
```

It should also be considered that both arm and ring have non-negligible dimensions.

Inverse kinematics

Knowing the position and orientation of the circle, the final action element should be moved to this configuration to then perform the desired movements. The inverse kinematics algorithm will be used for this purpose:

$$\dot{q} = J^{\dagger}\, k \begin{bmatrix} p_d - p_{ee}(q) \\ \eta_{ee}(q)\epsilon_d - \eta_d\, \epsilon_{ee}(q) - S(\epsilon_d)\epsilon_{ee}(q) \end{bmatrix}, \text{ where}$$

$$k = diag([1,1,1,1,0.1,0.1])$$

Remote Center of Motion

In the case of robotic-assisted surgery, it is important that the tool has the ability to create movements at its end without moving the incision entry point, as this would cause complications and undesirable results in the procedure. This limitation is called RCM (Remote Center of Motion)

Let's consider, $p_c \epsilon \mathbb{R}^3$,the point of incision and $p_t \epsilon \mathbb{R}^3$with orientantion

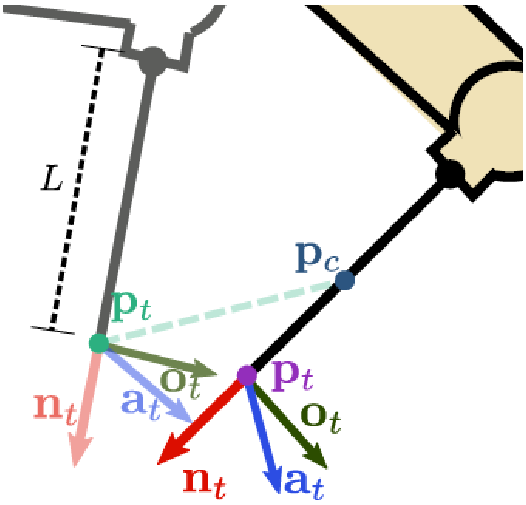
$R_t = [a_t \quad o_t \quad n_t]$. The goal is to keep the projection $p_t - p_c$ in the plane perpendicular to the unit vector n_t as close to zero as possible.

The basis of this level can be defined as follows,

$$B_c = [a_t \quad o_t] \epsilon \mathbb{R}^{3x2}.$$

$$x_c = {B_c}^T (p_t - p_c) \epsilon \mathbb{R}^2$$

The norm $\|x_c\|$ represents the minimum distance between the point p_c and the axis $\mathrm{n_t}$



Shared control with haptic feedback for robotic-assisted minimally invasive surgery, Kastritsi Theodora [1]

The time derivative $\dot{x_c} = {B_c}^T \dot{p_t} + \dot{B_c}^T (p_t - p_c)$

However, $R_t = [B_c \quad n_t]$ and $\dot{R_t} = \widehat{\omega_t} R_t$,where $\widehat{\omega_t} = \dot{R_t} {R_t}^T$, so finally $\dot{B_t} = \widehat{\omega_t} B_t$ and therefore

$$\dot{x_c} = {B_c}^T \dot{p_t} + {B_t}^T \widehat{\omega_t}^T (p_t - p_c)$$

The above expression captures the RCM constraint in the task space and can be written as follows:

$$\dot{x_c} = A_x v_t \text{ ,where}$$

$$A_x = {B_c}^T [I_{3x3} \quad \widehat{(p_t - p_c)}] \epsilon \mathbb{R}^{2x6}$$

$$v_t = [\dot{p_t}^T \quad {\omega_t}^T]^T \epsilon \mathbb{R}^6$$

Having calculated A_x, its null space will be calculated so that the movements of the end-effector will take place there and will not affect the initial RCM constraint. Thus the basis of the null space of A_x:

$$Z_x = \begin{bmatrix} {n_t}^T & 0_{1x3} \\ {R_t}^T \widehat{(p_t - p_c)} & {R_t}^T \end{bmatrix} \epsilon \mathbb{R}^{6x4}$$

The restrictions should be expressed in the joint space

$$\dot{x_c} = A\dot{q_d}$$

$$A = A_x \mathrm{J_t}(q_d) \epsilon \mathbb{R}^{2x6}$$

Finally, the joint velocity is expressed by the sum of two perpendicular components one related to the constrained space and the other to the free space.

$$\boldsymbol{\dot{q_d} = A^{\dagger}\dot{x_c} + \mathrm{J_t}^{\dagger}{Z_x}^T v_u},$$

where $v_u \epsilon \mathbb{R}^4$ is the unconstrained velocity which contains the translational velocity in the direction n_t and the angular velocities about the point of the incision expressed in terms of the axes of the end-effector

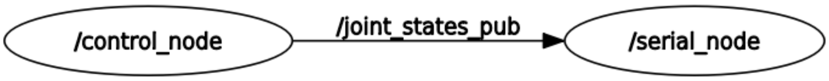
The main advantage of the above method is the decoupling between free and constrained velocity

Implementation

For the implementation, the laboratory's 6-degree-of-freedom anthropomorphic robotic arm with joint limits was used:

Link	**Limits (°)**
1	(-90, 90)
2	(0, 125)
3	(0, 125)
4	(-90, 90)
5	(-90, 30)
6	(0, 180)

Specifically, using ROS, it was possible to do the higher-level calculations through Python and control the servo motors through the Arduino of the robot-arm. For this purpose, the following nodes and topics were defined

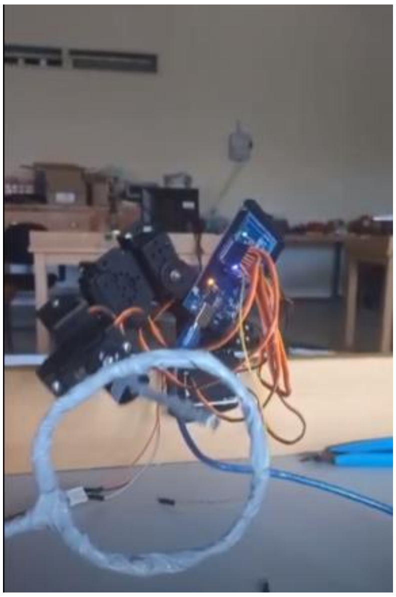


The procedure followed involves first achieving the desired position and orientation and then implementing the desired movements

Experiments

Achieving a certain pose

We assume that the arm starts from the configuration [5,75,20,45, -30,45] and we wish to move to the position (14.44,-10.653,15.187), with orientation R= $\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$ where the ring simulating the incision is located.



[.https://drive.google.com/file/d/1zO3hW3HonFWf6hmYEBG72_4PjSIlntgi/view?usp=drive_link](https://drive.google.com/file/d/1zO3hW3HonFWf6hmYEBG72_4PjSIlntgi/view?usp=drive_link)

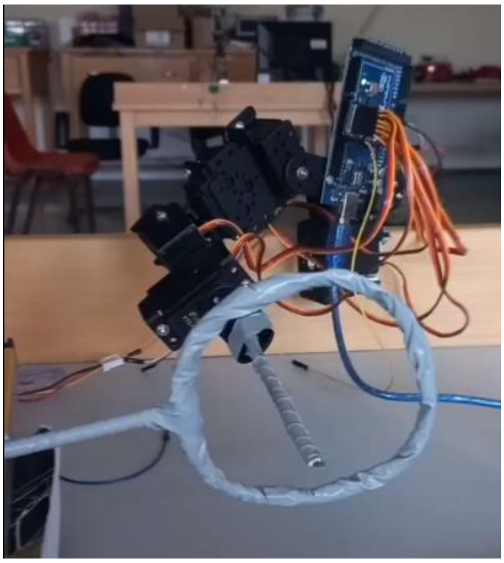
Translational movements

According to what was descirebed earlier the unconstrained velocity $v_u \epsilon \mathbb{R}^4$ which contains the translational velocity in the direction n_t and the angular velocities about the point of the incision expressed in terms of the axes of the end-effector.

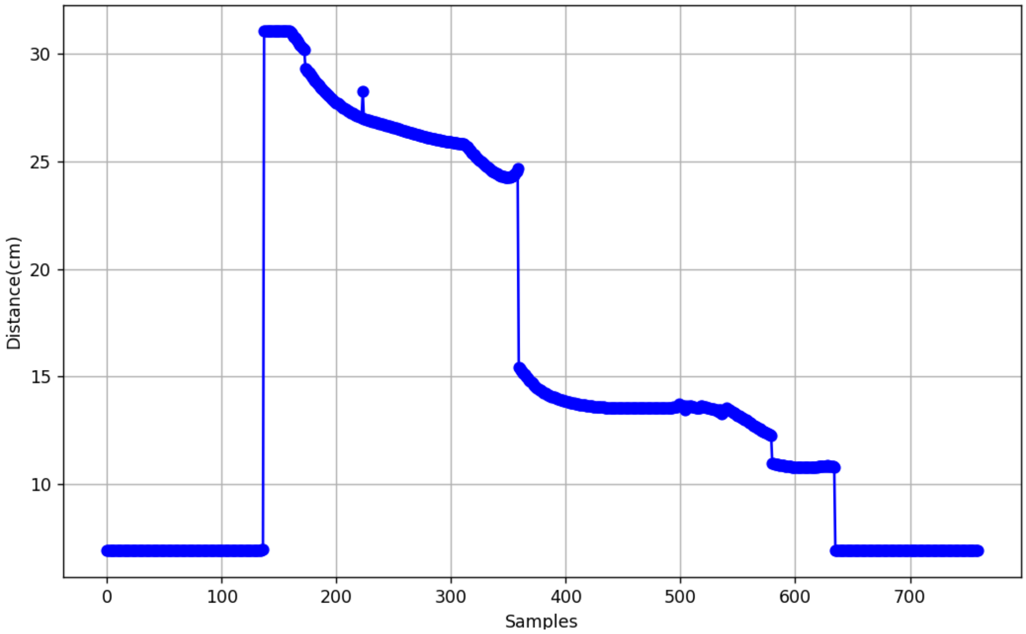
After experimentation and testing it was decided that for the smoothest translational motion of the arm, the velocity should contain a component along the nt axis and a smaller component in the corresponding axis of the desired translational motion.

The following sections include videos of the implementation, followed by graphs of the minimum distance and RCM constraint.

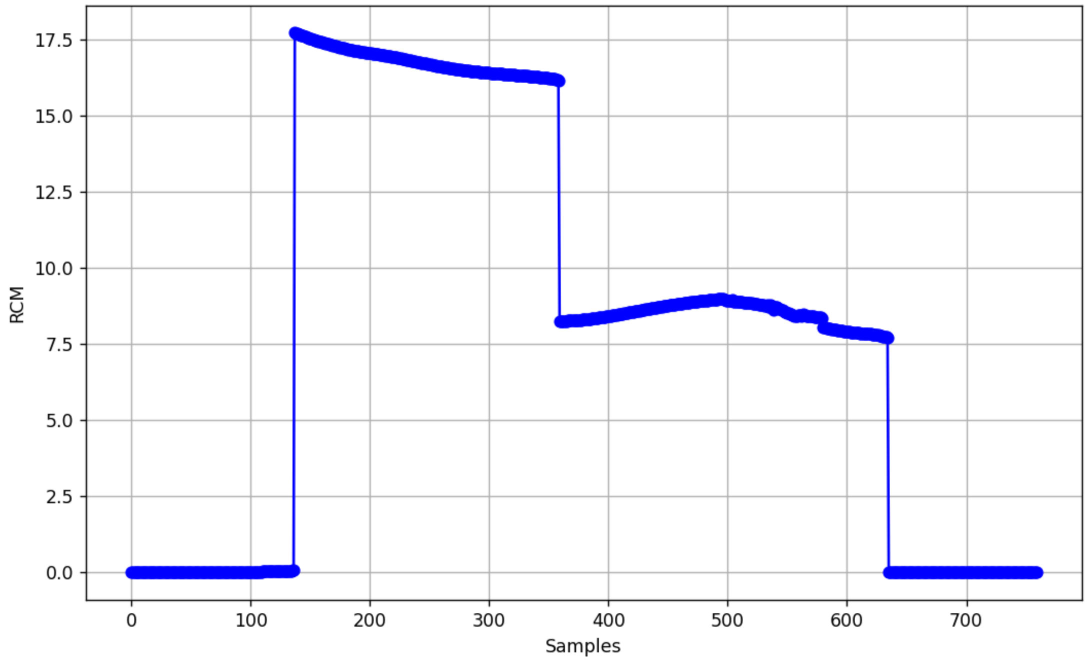
- $v_u = [1 \quad 0 \quad 0 \quad 0]$



https://drive.google.com/file/d/1gSmNu2s8JeWGH_sY6quhTm_oqL48JOQZ/view?usp=drive_link

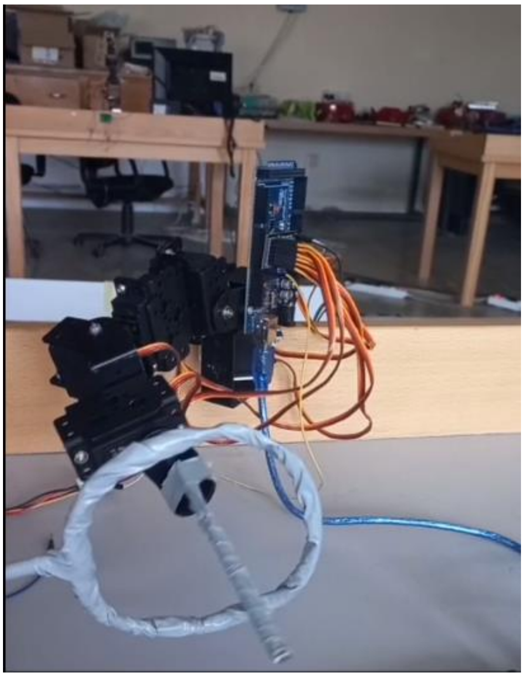


Graph of the distance for movement along the z-axis of the end-effector

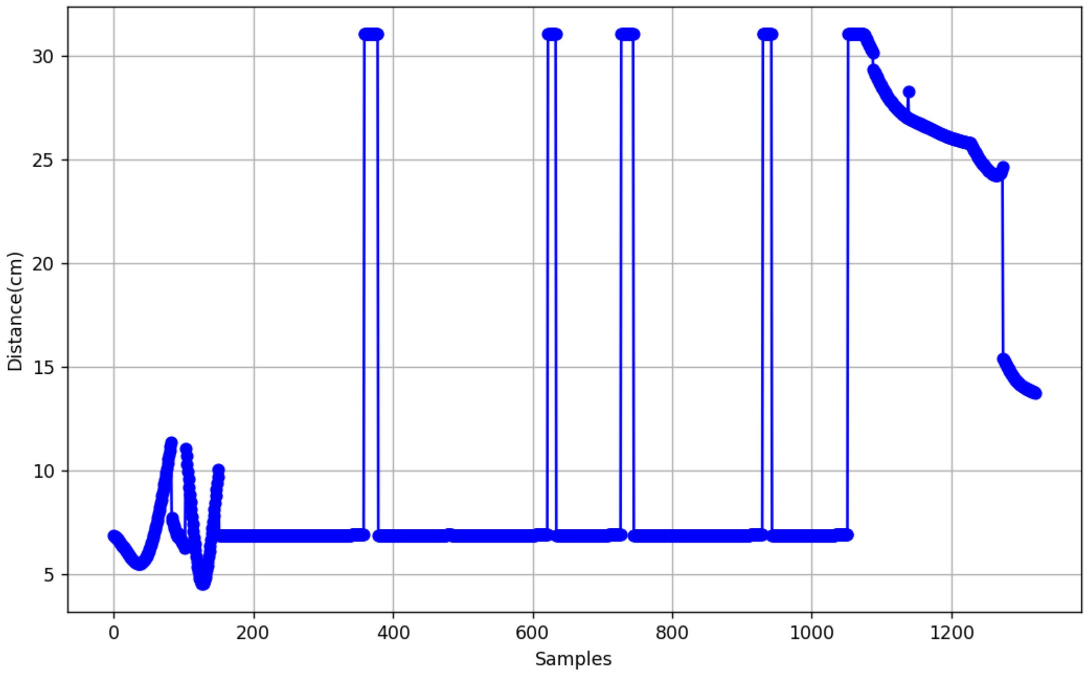


Graph of the RCM for motion along the z-axis of the end-effector

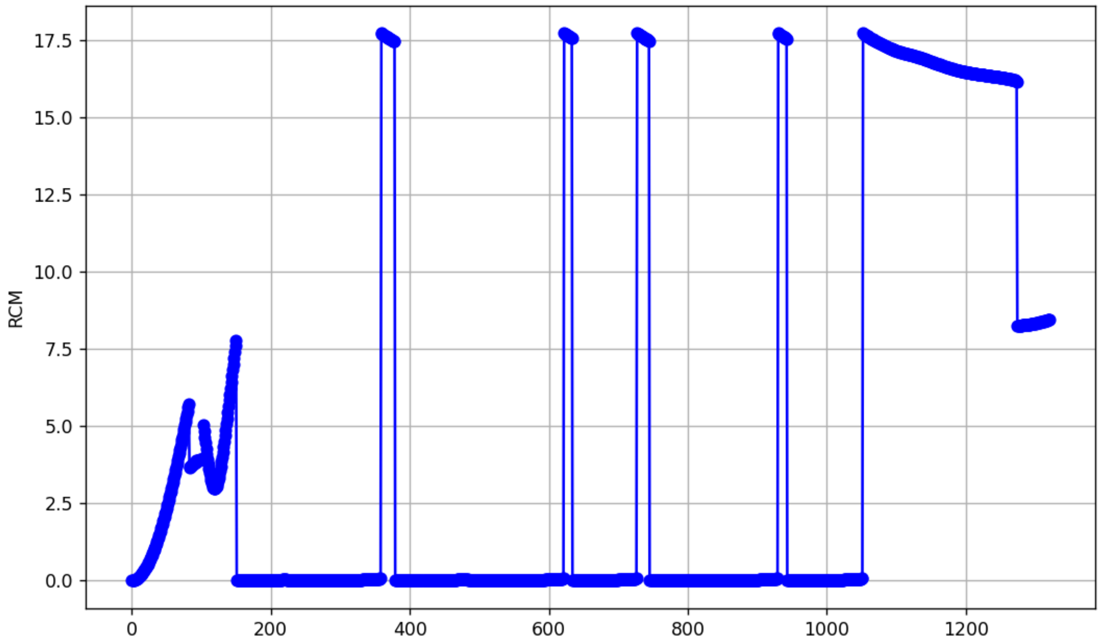
- $v_u = [1 \quad 0.1 \quad 0 \quad 0]$



https://drive.google.com/file/d/1JX92oNhtCIa9Q7IkoL6yoh1vbaD5aNxp/view?usp=drive_link

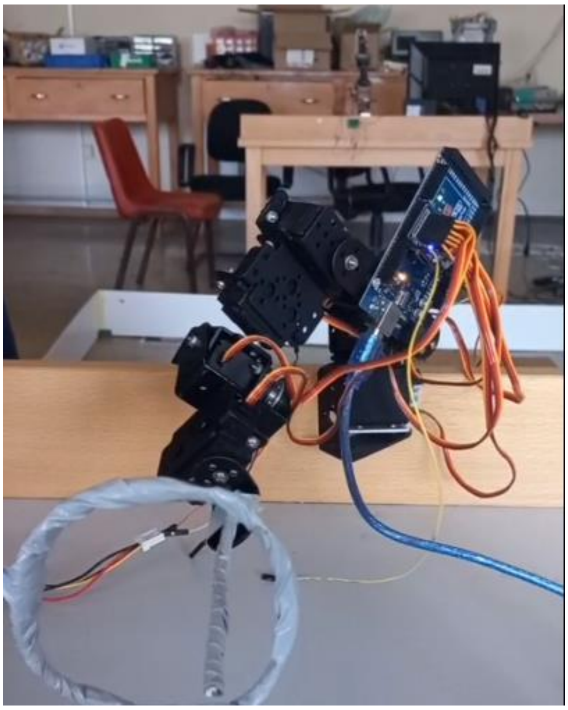


Graph of the distance for movement along the x-axis of the end-effector

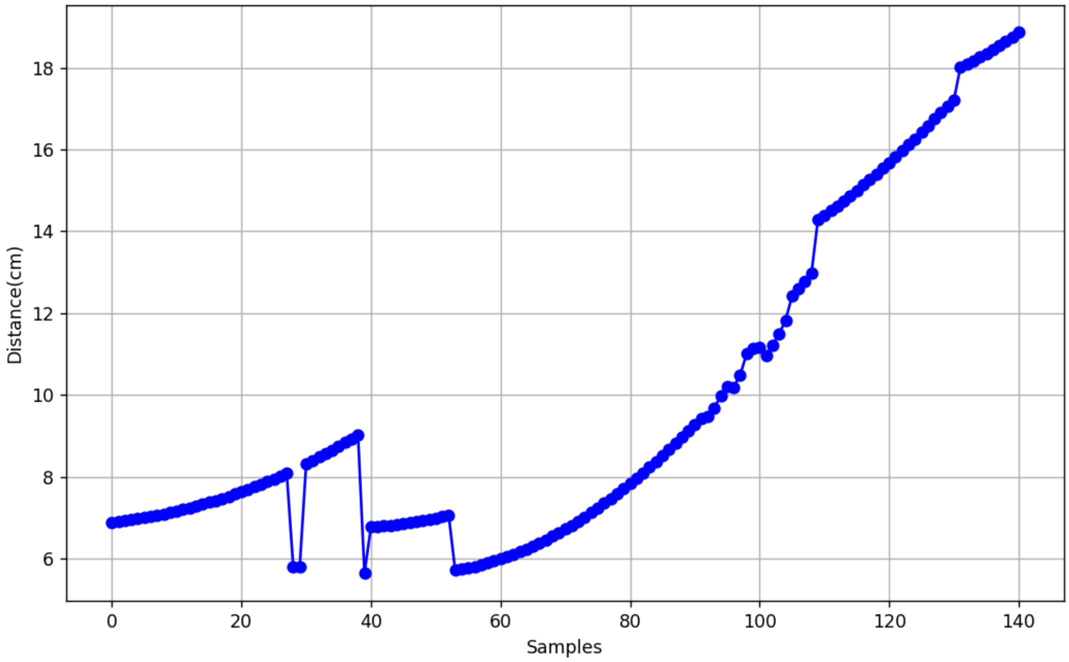


Graph of the RCM for motion along the x-axis of the end-effector

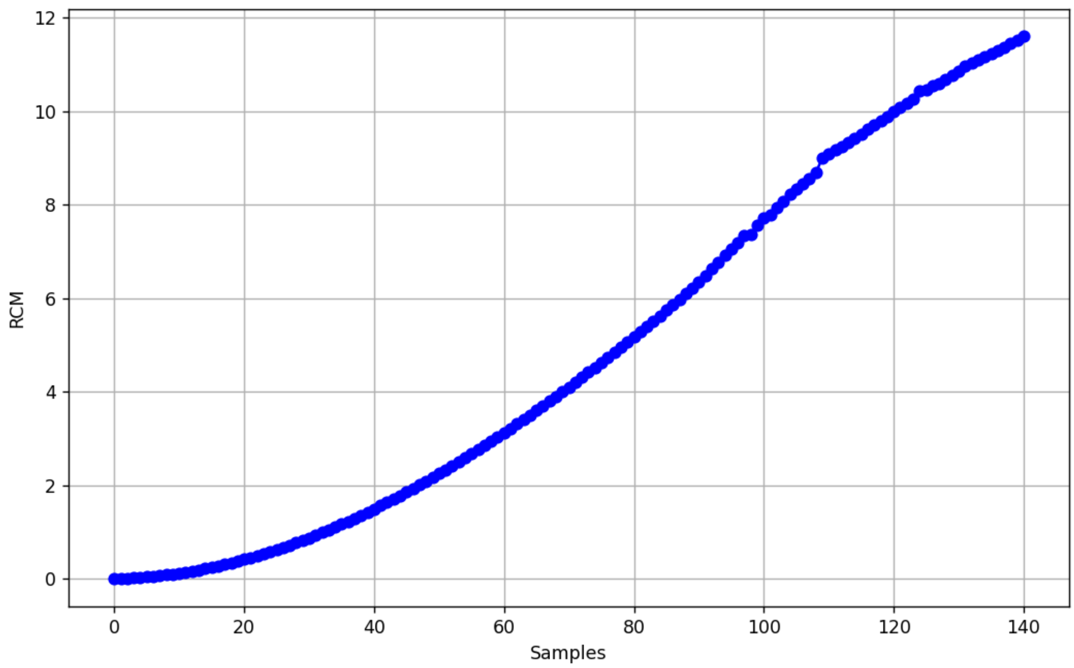
- $v_u = [1 \quad 0 \quad 0.1 \quad 0]$



https://drive.google.com/file/d/1jXcaAgvUnLZAA3kiqUbJZM-8d5N84kM2/view?usp=drive_link

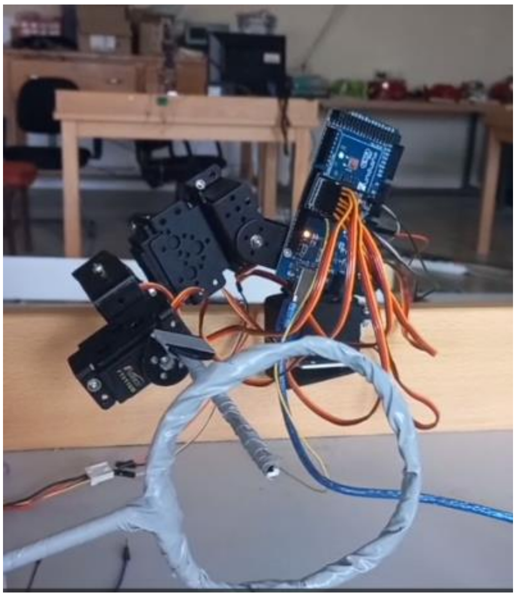


Graph of the distance for movement along the y-axis of the end-effector

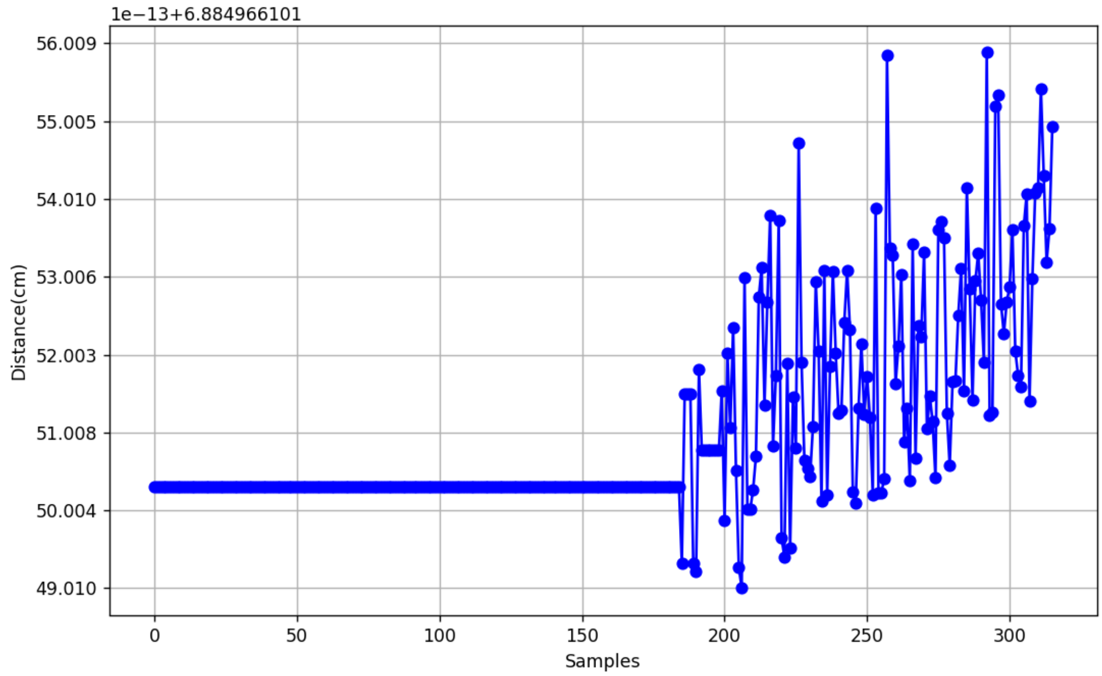


Graphical representation of the RCM for motion along the y-axis of the end-effector

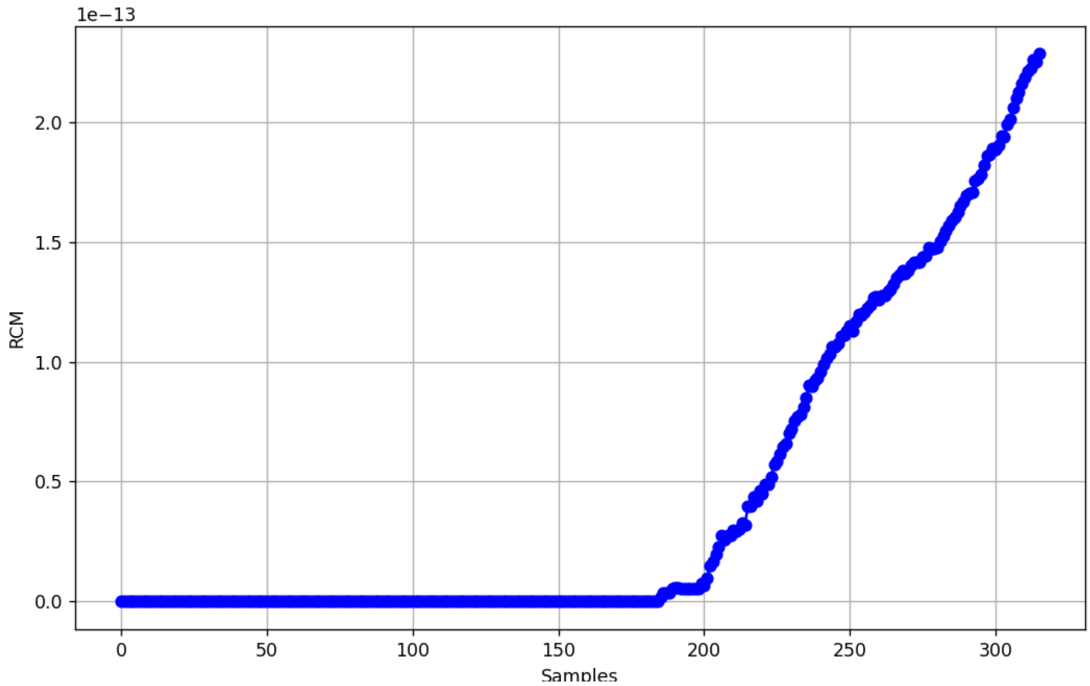
- $v_u = [1 \quad 0 \quad 0 \quad 0.1]$



https://drive.google.com/file/d/1vCVQbL3FVVxK6vZqA2vmXiKenmQoqqRs/view?usp=drive_link



Graph of distance for rotational movement along the z-axis of the end-effector



Graph of RCM for rotational motion along the z-axis of the end-effector

Evaluation of results

As expected, the results have a deviation from reality, especially concerning the minimum distance, since there is a great uncertainty in the exact dimensions of each joint which was not taken into account. Even though there were cases where the arm collided with the incision this is not reflected in the graphs. However, they do correctly capture the tendency of the result.

On the other hand, the RCM constraint graphs are more precise and clearly show the tendency of the graph to return to zero even when it deviates for a while as this is the primary objective.

Conclusions

The results confirm the theoretical basis of the implementation, although they do not fully agree due to phenomena that were not taken into account such as the dynamics of the arm as well as its imperfections. To improve the performance of the operation, it is considered useful to implement the mechanism in an robot-arm with more degrees of freedom, which on the one hand would provide more alternatives to achieve the primary objective and on the other hand would allow the achievement of secondary objectives (avoiding singularities) that would contribute to a more accurate and smoother movement of the robotic arm.

Bibliography

[1]. Shared control with haptic feedback for robotic-assisted minimally invasive surgery, Kastritsi Theodora

[2].Robotics Modelling, Planning and Control, Bruno Siciliano , Lorenzo Sciavicco , Luigi Villani , Giuseppe Oriolo