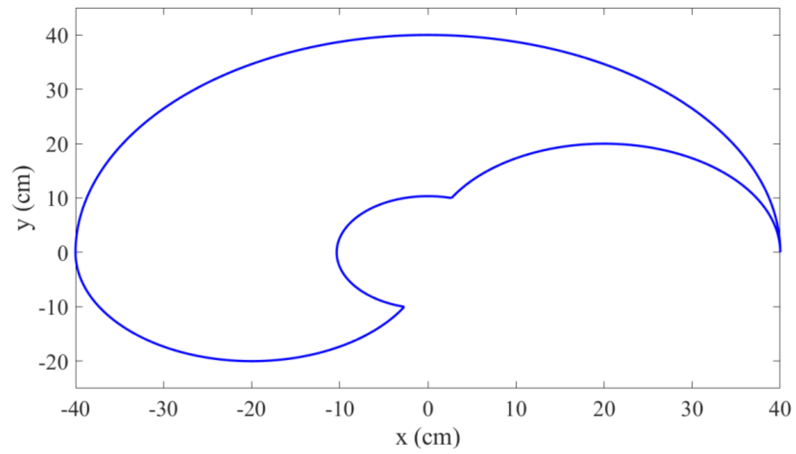
TD 5 & 6 – Cobots & HMI

Groupe 7 :

- MARÇAL Thomas

- KOSKAS Axel

1.



- What do you observe from the plot obtained ?

We can see a blue region that corresponds to the limit of the robot workspace. This workspace of a robot arm is the set of all positions it can reach .

There are 2 possible configurations: Elbow up and Elbow down. In our case here, we are in the Elbow up case.

- What are the joint limits for this plot?

For this plot, the joint limits are according to the code of the robot function :

$$\theta_1 = 0° \; to \; 180°$$

$$\theta_2 = 0° \; to \; 150°$$

We have a singularity with $\theta_2 = 0° \; to \; 180°$.

2.

For this problem we have the following derivatives of r(t) :

$$r(t) = 10 * \left(\frac{t}{t_f}\right)^3 - 15 * \left(\frac{t}{t_f}\right)^4 + 6 * \left(\frac{t}{t_f}\right)^5$$

$$\dot{r}(t) = 30 * \frac{t^2}{t_f^3} - 60 * \frac{t^3}{t_f^4} + 30 * \frac{t^4}{t_f^5}$$

$$\ddot{r}(t) = 60 * \frac{t}{t_f^3} - 180 * \frac{t^2}{t_f^4} + 120 * \frac{t^3}{t_f^5}$$

- Determine the step size required for the trajectory.

We have :

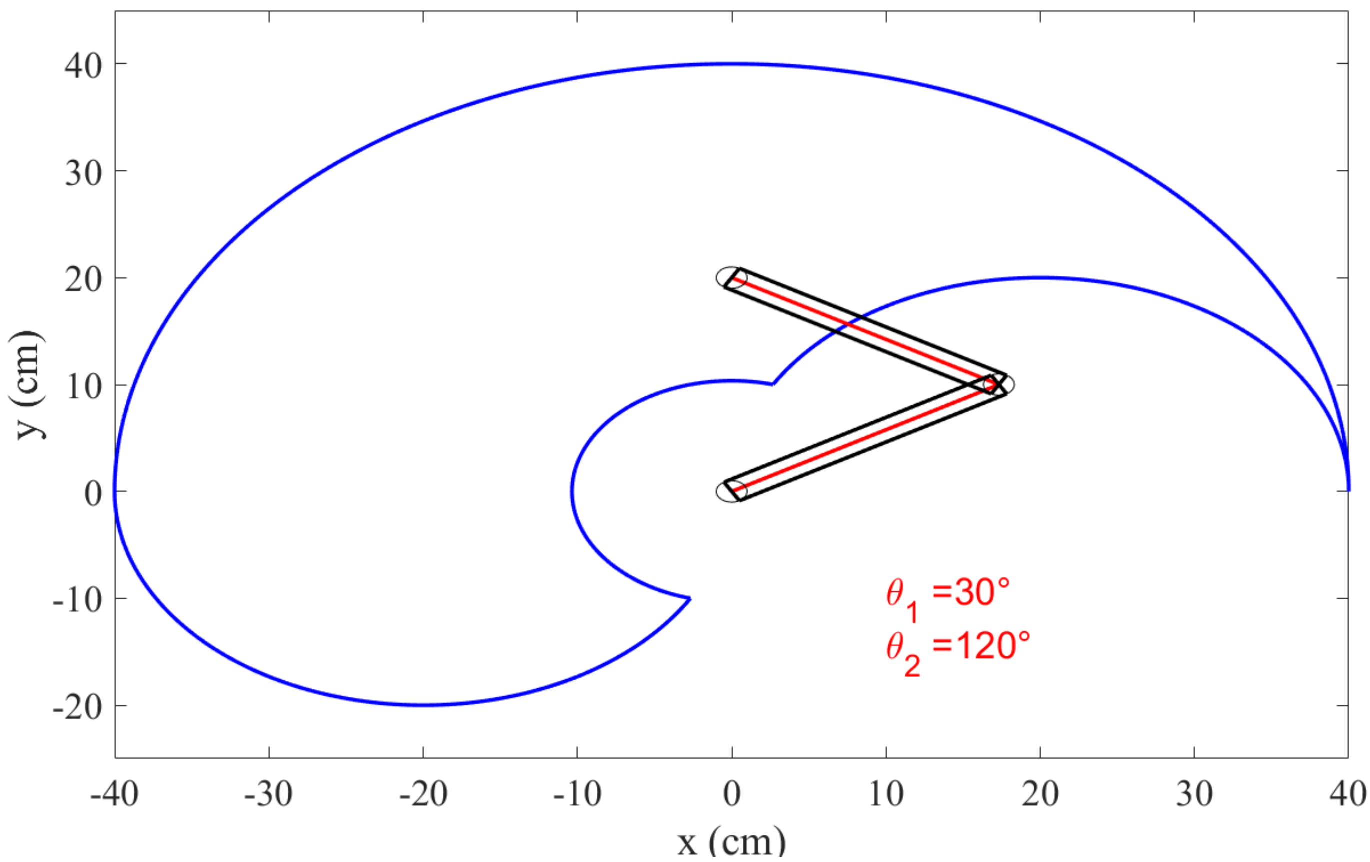
$$T_f = 20 \; s$$

$$f = 2.85 \; Hz$$

So :

$$stepsize = T_f * f = 57$$

3. See the code.

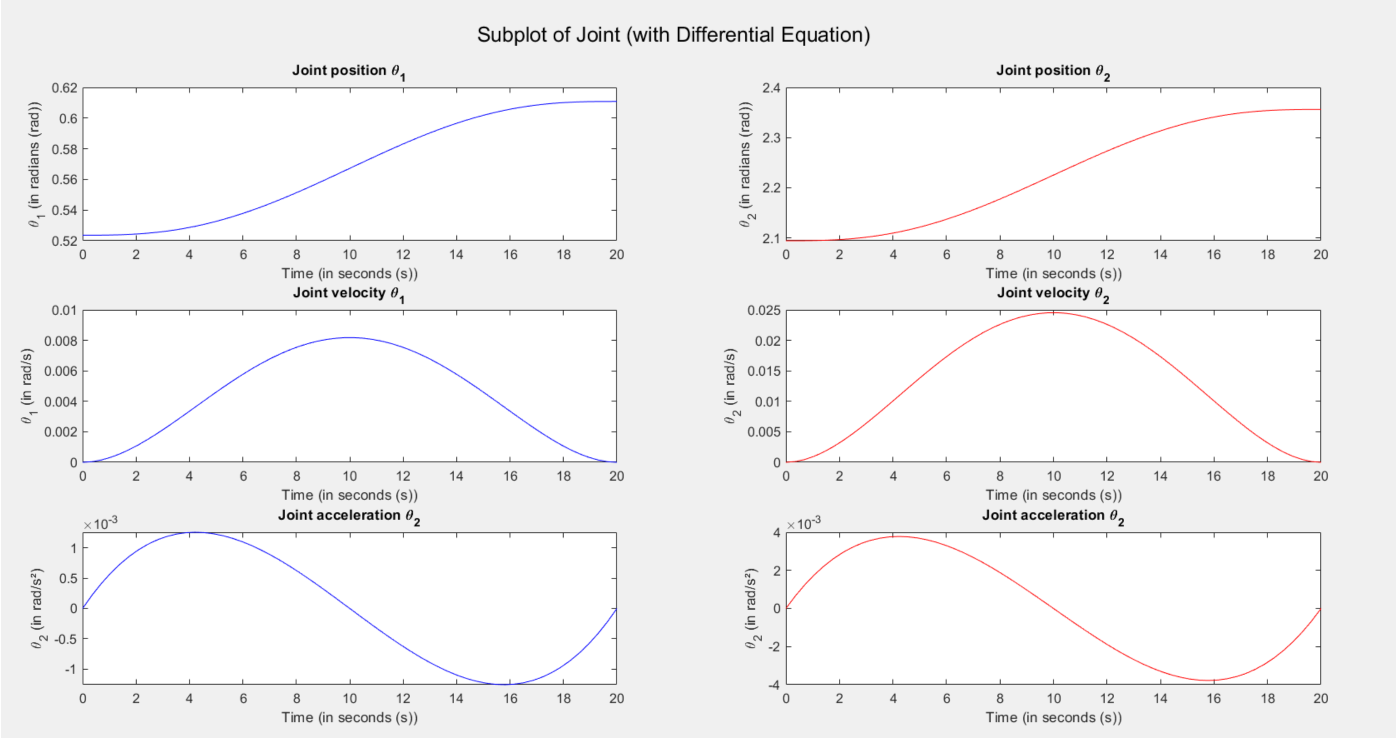


4.

Joint space = Control and manipulation of each and every actuated joint. We are not concerned about the position of the end-effector.

Work space (Task Space) = We estimate joint variables for each position of end-effector do ensure the correct configuration & limits (singularities). We are concerned about the position of end-effector.

5. See the code.

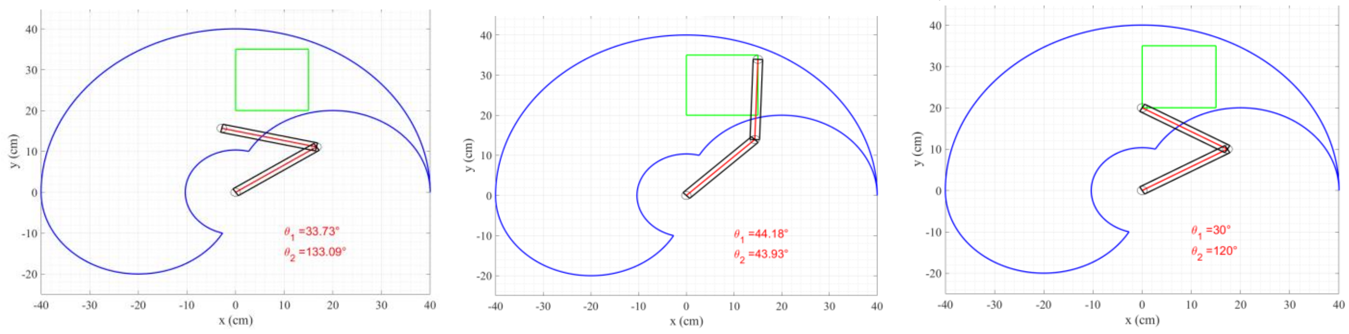


6.

To obtain this result, we have created the function to obtain the inverse kinematics of our system *InverseKinematics(X,Y)* of the R2 robot.

```
function [Theta1,Theta2] = InverseKinematics(X,Y,l1,l2)
    Theta2 = acos((-l1^2 - l2^2 + (X^2+Y^2))/(2*l1*l2));
    Theta1 = atan2(Y,X) - atan2(l2*sin(Theta2),l1+l2*cos(Theta2));
end
```

We thus obtain the desired animation where the robot follows the line of the square.



Then, in order to obtain the joint position, velocity and acceleration, we use the formulas provided by the statement.

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix}$$

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} - \mathbf{J}^{-1}\dot{\mathbf{J}} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

And knowing that in the TD 3&4, we had realized a code allowing to obtain the jacobian of the robot R2, we have :

```
J =

[- L2*sin(q1 + q2) - L1*sin(q1), -L2*sin(q1 + q2)]
[  L2*cos(q1 + q2) + L1*cos(q1),  L2*cos(q1 + q2)]
[                             0,                0]
[                             0,                0]
[                             0,                0]
[                             1,                1]
```

- Thanks to Matlab, we obtain the inverse Jacobian :

Jinv =

[-cos(q1 + q2)/(L1*cos(q1 + q2)*sin(q1) - L1*sin(q1 + q2)*cos(q1)), -sin(q1 + q2)/(L1*cos(q1 + q2)*sin(q1) - L1*sin(q1 + q2)*cos(q1))]

[(L2*cos(q1 + q2) + L1*cos(q1))/(L1*L2*cos(q1 + q2)*sin(q1) - L1*L2*sin(q1 + q2)*cos(q1)), (L2*sin(q1 + q2) + L1*sin(q1))/(L1*L2*cos(q1 + q2)*sin(q1) - L1*L2*sin(q1 + q2)*cos(q1))]

So, we create the function JacobianInverseR2(L1,L2,q1,q2) which allows us to obtain the inverse of the Jacobian.

- For the derivative of the Jacobian, we have that :

$$\frac{dJ}{dt} = \frac{dJ}{d\theta}\frac{d\theta}{dt} \Leftrightarrow \frac{dJ}{dt} = \frac{d(J*\dot{\theta})}{d\theta}$$

We have therefore created the function JacobianDerivateR2(L1,L2,q1,q2,q1der,q2der) which corresponds to the derivative of the Jacobian.

We can thus obtain these 3 points requested: