COMP0127: Robotic System Engineering Coursework 2 : Jacobian and Inverse Kinematics

Wenguang Lin

December 2019

1 question 1

The transformation matrix for a 2D 4R-planar manipulator can be expressed as:

$$ {}^{0}T_e = \begin{bmatrix} r_1 & r_2 & 0 & p_{ex} \\ r_{21} & r_{22} & 0 & p_{ey} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1} $$

$$ {}^{0}T_e(q) = \begin{bmatrix} cos(\theta_{1234} & -sin_{1234} & 0 & l_1cos(\theta_1) + l_2cos(\theta - 12) + l_3cos(\theta_{123}) + l_4cos(\theta_{1234}) \\ sin(\theta_{1234} & cos_{1234} & 0 & l_1sin(\theta_1) + l_2sin(\theta - 12) + l_3sin(\theta_{123}) + l_4sin(\theta_{1234}) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2} $$

Solving four equations for $q = (\theta_1, \theta_2, \theta_3)$:

$$ cos(\theta_{1234}) = r_1 \tag{3} $$

$$ sin(\theta_{1234}) = r_3 \tag{4} $$

$$ l_1cos(\theta_1) + l_2cos(\theta_{12}) + l_3cos(\theta_{123}) + l_4cos(\theta_{1234}) = p_{ex} \tag{5} $$

$$ l_1sin(\theta_1) + l_2sin(\theta_{12}) + l_3sin(\theta_{123}) + l_4sin(\theta_{1234}) = p_{ey} \tag{6} $$

Inserting the equation (3) and (4) into the equation (5) and (6). The new equations can be showns as:

$$ l_1cos(\theta_1) + l_2cos(\theta_{12}) + l_3cos(\theta_{123}) + l_4r_1 = p_{ex} \tag{7} $$

$$ l_1sin(\theta_1) + l_2sin(\theta_{12}) + l_3sin(\theta_{123}) + l_4r_3 = p_{ey} \tag{8} $$

As it shows in the equation (7) and (8), there are three unknowns which are θ_1, θ_2 and θ_3 but there are only two equations. Therefore, it has infinite inverse kinematic solutions exist for a 2D 4R-planar manipulator.

2 question 2

There are some criteria as choosing an optimal solution for the robot moving in a free space without any obstacle.

- joint Limit:
  It can represent the farthest distance which the robot can reach within the work space.
- Singularity:
  It represents a condition when two or more robotics joint axes are col-linear alignment, and the degree of freedom of the robot is less than the actual dimension that the robot requires, it would result in unpredictable robot motion and velocities. There are three different kinds of singularities:(1. Wrist singularities 2.Shoulder singularities 3.Elbow singularities). It is very essential to avoid any singularity conditions.
- Joint Torque: It should be designed as a small value when calculating the inverse-kinematic solutions
- Energy Consumption: Energy consumption should be as small as possible.
- Computation Time: This represents how much time it is required to compute to reach the desired goal. The computation time also means how efficiency the system is to reach the goal.

3 question 3

for atan(y/x), it finds that the corresponding angle based on the tangent value of y/x can be considered as the 2 quadrant arc-tangent.

- when $y/x > 0$, the range of atan(y/x) is from 0 to $\pi/2$
- when $y/x < 0$, the range of atan(y/x) is from $-\pi/2$ to 0

However, for atan2(y,x), it is the 4 quadrant arc-tangent. its value depends not only on the tangent value of (y/x), but also on which quadrant the point(x,y) falls into:

- when point(x,y) falls into the first quadrant, the range of atan2(y,x) is from 0 to $\pi/2$
- when point(x,y) falls into the second quadrant, the range of atan2(y,x) is from $\pi/2$ to π
- when point(x,y) falls into the third quadrant, the range of atan2(y,x) is from $-\pi$ to $-\pi/2$
- when point(x,y) falls into the fourth quadrant, the range of atan2(y,x) is from $-\pi/2$ to 0

4 question 4

4.1 b

Manipulators usually have several joints to define the position of the end effector and last three joins for the orientation(spherical wrist). The last two joints intersect at wrist centre. They can expressed as:

$$o_c^0 = [x_c, y_c, z_c]^T, o_0^c = o_0^3 = o_0^4 \tag{9}$$

The forward kinematic transformation ${}^0T_5$ and the the z and o c be expressed as:

$$\begin{cases} r_{13} = S_{234}C_1(4b1) \\ r_{23} = S_{234}S_1(4b2) \\ r_{33} = C_{234}(4b3) \\ p_x = -C_1(a_1 - a_3S_{23} - a_2S_2 + a_4C_{234} + d_5S_{234})(4b4) \\ p_y = -S_1(a_1 - a_3S_{23} - a_2S_2 + a_4C_{234} + d_5S_{234})(4b5) \\ p_z = d_1 + a_3C_23 + a_2C_2 + a_4S_{234} - d_5C_{234}(4b6) \\ r_{11} = S_1S_5 + C_1C_{234}C_5(4b7) \\ r_{12} = -S_1C_5 + C_1C_{234}S_5(4b8) \\ r_{21} = -C_1S_5 + S_1C_{234}C_5(4b9) \\ r_{22} = C_1C_5 + S_1C_{234}S_5(4b10) \\ r_{31} = -S_{234}C_5(4b11) \\ r_{32} = S_{234}S_5(4b12) \end{cases}$$

where, $S1 = sin(\theta_1), C_1 = cos(\theta_1), S_{234} = Sin(\theta_2 + \theta_3 + \theta_4), C_{234} = Cos(\theta_2 + \theta_3 + \theta_4)$

Inserting the equations 4b1, 4b2 into the equation (4b4). the new equation becomes:

$$\frac{p_x + d_5r_{13}}{C_1} + a_1 + a_4r_{33} = a_3S_{23} + a_2S_2 = N \tag{10}$$

Inserting the equation 4b2 4b3 into the equation (4b5), then the new equation becomes:

$$\frac{p_y + d_5r_{23}}{S_1} + a_1 + a_4r_{33} = a_3S_{23} + a_2S_2 \tag{11}$$

as you can see from the equation (4b4 and 4b5), the relationship can be obtained as:

$$\frac{p_x}{C_1} = \frac{p_y}{S_1} \tag{12}$$

$$\longrightarrow tan(\theta_1) = \frac{p_y}{p_x} \longrightarrow \theta_1 = atan2(p_y, p_x), or \theta_1 = atan2(p_y, p_x) \pm \pi$$

Rearranging the equation (4b6)

$$M = p_Z - d_1 - a_4S_{234} + d_5r_{33} = a_3C_{23} + a_2C_2 \tag{13}$$

Considering two equations 11 and 13 and inserting equation (4b1) and (4b2), the new equations can be expressed under two conditions:
when $S_1 = 0$:

$$M = p_Z - d_1 - a_4\frac{r_{13}}{C_1} + d_5 r_{33} \tag{14}$$

$$N = \frac{p_x + d_5 r_{13}}{C_1} + a_1 + a_4 r_{33} \tag{15}$$

when $S_1 \neq 0$:

$$M = p_Z - d_1 - a_4\frac{r_{23}}{C_1} + d_5 r_{33} \tag{16}$$

$$N = \frac{p_x + d_5 r_{23}}{C_1} + a_1 + a_4 r_{33} \tag{17}$$

now we need to calculate the summation between N square and M square:

$$N^2 = (a_3 S_{23} + a_2 S_2)^2 = a_3^2 S_{23}^2 + 2a_3 a_2 S_2 S_{23} + a_2^2 S_2^2 \tag{18}$$

$$M^2 = (a_3 C_{23} + a_2 C_2)^2 = a_3^2 C_{23}^2 + 2a_3 a_2 C_2 C_{23} + a_2^2 C_2^2 \tag{19}$$

$$M^2 + N^2 = a_2^2 + a_3^2 + 2C_3 a_2 a_3 \tag{20}$$

$$C_3 = \frac{N^2 + M^2 - a_2^2 - a_3^2}{2a_2 a_3}$$

Therefore, θ_3 can be calcuated as:

$$\theta_3 = \pm cos^{-1}(\frac{N^2 + M^2 - a_2^2 - a_3^2}{2a_2 a_3}) \tag{21}$$

expanding the equation 13:

$$M = a_3 C_2 3 + a_2 C_2 = a_2 C_2 + a_3 C_2 C_3 - a_3 S_2 S_3 \tag{22}$$

$$\longrightarrow M = (a_2 + a_3 C_3) C_2 - S_2(a_3 S_3)$$

$$M = AC_2 - BS_2 = Cos(\theta_2 + \lambda)$$

where, $A = a_2 + a_3 C_3$, $B = a_3 S_3$, $\lambda = atan2(B, A)$
Apply the similar method to calculate N:

$$N = a_3 S_{23} + a_2 S_2 = Sin(\theta_2 + \lambda) \tag{23}$$

Therefore, θ_2 can be calculated by:

$$\frac{N}{M} = \frac{sin(\theta_2 + \lambda)}{cos(\theta_2 + \lambda)} \tag{24}$$

$$\longrightarrow \theta_2 = atan2(N, M) - atan2(B, A)$$

The rotational matrix can be obatined by ${}^3R_5 = ({}^0R_3)^{-1} R$:

$${}^3R_5 = \begin{bmatrix} -C_5 S_4 & -S_4 S_5 & C_4 \\ C_4 C_5 & C_4 S_5 & S_4 \\ -S_5 & C_5 & 0 \end{bmatrix} = \begin{bmatrix} n_{11} & n_{12} & n_{13} \\ n_{21} & n_{22} & n_{23} \\ n_{31} & n_{32} & n_{33} \end{bmatrix} \tag{25}$$

According to the equation 25, $\theta_4 and \theta_5$ can be shown as:

$$\theta_4 = atan2(-n_{23}, n_{13}) \tag{26}$$

$$\theta_5 = atan2(n_{31}, n_{32}) \tag{27}$$

5 question 5

5.1 a

The trajectory function has the form of :

$$q(t) = a_0 + a_1 t + a_t^2 + a_3 t^3 \tag{28}$$

Four constraints on positions and velocities in the function can be shown as:

$$q(t_s) = q_s,\ \dot{q}(t_s) = v_s,\ q(t_f) = q_f,\ \dot{q}(t_f) = v_f \tag{29}$$

from the equation (28), the velocity parameters can be obtained:

$$\dot{q}(t) = a_1 + 2a_2 t + 3a_3 t^2 \tag{30}$$

substituting four constraints into the equations (28) and (30):

$$q(t_s) = a_0 + a_1 t_s + a_2 t_s^2 + a_3 t_s^3 \tag{31}$$

$$\dot{q}(t_s) = a_1 + 2a_2 t_s + 3a_3 t_s^2 \tag{32}$$

$$q(t_f) = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 \tag{33}$$

$$\dot{q}(t_f) = a_1 + 2a_2 t_f + 3a_3 t_f^2 \tag{34}$$

The system of linear equations can be shown in a matrix form:

$$\begin{bmatrix} 1 & t_s & t_s^2 & t_s^3 \\ 0 & 1 & 2t_s & 3t_s^2 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} q_s \\ v_s \\ q_f \\ v_f \end{bmatrix} \tag{35}$$

In the question, $t_s = 0$, and $t_f = 10$, thus the system of linear equations can be rewritten as:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 10 & 100 & 1000 \\ 0 & 1 & 20 & 300 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} q_s \\ v_s \\ q_f \\ v_f \end{bmatrix} \tag{36}$$

The heard starting position and the starting velocity are:

$$qs = \begin{bmatrix} 0.00122 \\ 0.39156 \\ -0.35444 \\ 0.32441.01005 \end{bmatrix} \ and \ \ v_s = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{37}$$

The final positions and velocity can be read from 'sensor msgs:: jointState()', thus matrix a can be calculated by:

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} q_s \\ v_s \\ q_f \\ v_f \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 10 & 100 & 1000 \\ 0 & 1 & 20 & 300 \end{bmatrix}^{-1} \tag{38}$$

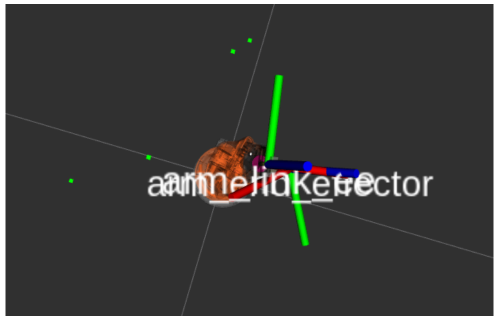

Figure 1: result of going through all check points

With the calculated matrix a, the trajectory function and its velocity can be calculated in the equation (28) and (30).

- First of all, it gets 10 data points(time, position and velocity)
- For every two check points, it takes 10 seconds to travel between two check points. For each check point, it is given the position and velocities, thus the matrix a can be calculated.
- the trajectory point would be read at each second, thus there are 50 points data in total.
- finally the 50 point messages will be published to the Youbot

The figure above shows the robot successfully goes through all the check points.

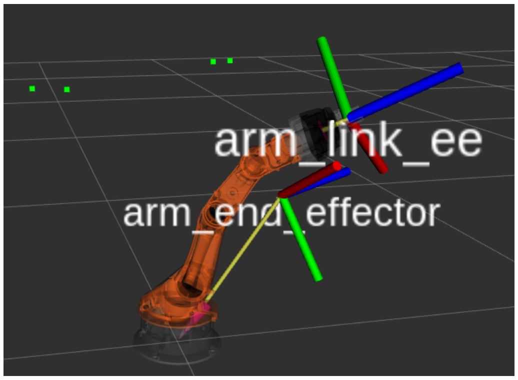

Figure 2: Shortest path through all check points

5.2 b

First of all, all path possibilities are calculated by using for loops, in order to calculate a best and shortest distance through all check points. from As what it has shown in the code, 'a', 'b', 'c', 'd', 'e' represent the sequence of traveling points. 'a' is the starting point. 'b' is the second point where the robot will pass. Similarly, 'e' represents the ending point. In one path possibility, it will sum up all distances between each check point. And then it will compare with other distances calculated from other path possibilities. In this way, the sequence of check points for a shortest distance can be found. Once 'a', 'b', 'c', 'd', 'e' are found , they would be updated to a sequence matrix for the later publishing use.

AS it shows in the figure above, the robot goes through all the check point in a shortest path.