Robot manipulator: project-parts 1 and 2

Instructor: Prof. Patel

Peyman Yadmellat

1. Forward kinematic transformations for the Stanford manipulator

Considering the Stanford manipulator shown in Figure 1, following D-H parameters can be obtained regarding the definitions given in section 3.4 in book:

Table 1. D-H parameters of the Stanford manipulator

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0°	0	h = 0.4 m	θ_1
2	-90°	0	r = 0.1 m	θ_2
3	90°	0	f	0°

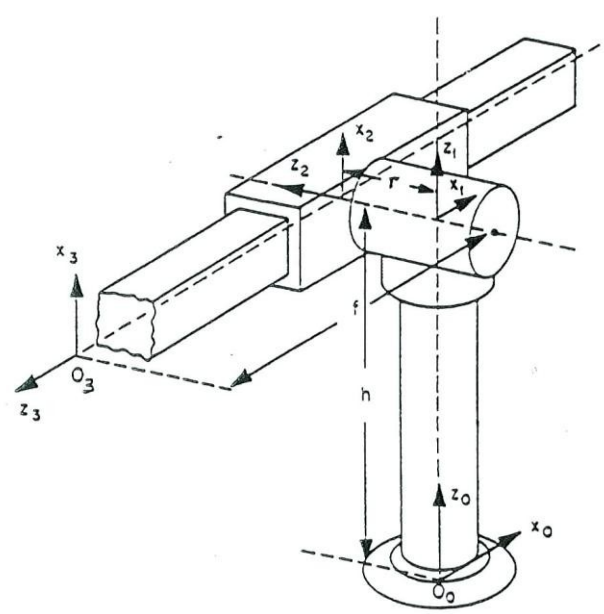

Figure 1. Stanford manipulator

Using Eq. (3.6) in book, each link transformation can be obtained as follow

$$ {}^{0}_{1}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0.4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1} $$

$$ {}^{1}_{2}T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0.1 \\ -s\theta_2 & -c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2} $$

$$ {}^{2}_{3}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -f \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3} $$

where c and s stand for *cos* and *sin*, respectively.

Then, the all possible link transformations can be obtained by following relation to find the single transformation which relates frame $\{Q\}$ to $\{P\}$:

$$ {}^{P}_{N}T = {}^{P}_{L}T\,{}^{L}_{L+1}T \cdots {}^{Q-1}_{Q}T \tag{4} $$

for $0 \leq P < Q \leq N$, where $\{N\}$ is the end-effector frame.

Thus, other possible single transformations can be computed for the Stanford manipulator as given below:

$$ {}^{0}_{2}T = {}^{0}_{1}T\,{}^{1}_{2}T = \begin{bmatrix} c\theta_1 c\theta_2 & -c\theta_1 s\theta_2 & -s\theta_1 & -0.1s\theta_1 \\ s\theta_1 c\theta_2 & -s\theta_1 s\theta_2 & c\theta_1 & -0.1c\theta_1 \\ -s\theta_2 & -c\theta_2 & 0 & 0.4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5} $$

$$ {}^{0}_{3}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T = \begin{bmatrix} c\theta_1 c\theta_2 & -s\theta_1 & c\theta_1 s\theta_2 & fc\theta_1 s\theta_2 - 0.1s\theta_1 \\ s\theta_1 c\theta_2 & c\theta_1 & s\theta_1 s\theta_2 & fs\theta_1 s\theta_2 - 0.1c\theta_1 \\ -s\theta_2 & 0 & c\theta_2 & fc\theta_2 + 0.4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6} $$

$$ {}^{1}_{3}T = {}^{1}_{2}T\,{}^{2}_{3}T = \begin{bmatrix} c\theta_2 & 0 & s\theta_2 & fs\theta_2 \\ 0 & 1 & 0 & 0 \\ -s\theta_2 & 0 & c\theta_2 & fc\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7} $$

Now by substituting the particular joint variables' values given in the project, i.e. $\theta = [9.22^\circ \quad 115.1^\circ \quad 0.3536\ m]^T$ into Eq. (6), we have

$$ {}_3^0T = \begin{bmatrix} -0.4187 & -0.1602 & 0.8939 & 0.3 \\ -0.0680 & 0.9871 & 0.1451 & 0.15 \\ -9.056 & 0 & -0.4242 & 0.25 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8} $$

Therefore, the position of end-efector relative to frame $\{0\}$ is

$$ {}^0P_{3ORG} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} 0.3 \\ 0.15 \\ 0.25 \end{bmatrix} $$

2. Inverse kinematic expression for the Stanford manipulator

Considering Eq. (6), following expression should be satisfied in order to find corresponding joint variables, i.e. $[\theta_1 \quad \theta_2 \quad f]^T$:

$$ \begin{cases} fc\theta_1 s\theta_2 - 0.1s\theta_1 = P_x \\ fs\theta_1 s\theta_2 + 0.1c\theta_1 = P_y \\ \quad fc\theta_2 + 0.4 = P_z \end{cases} \tag{9} $$

Now by multiplying first and second equation respectively by $-s\theta_1$ and $c\theta_1$ in (9), after simple manipulations we have

$$ -P_x s\theta_1 + P_y c\theta_1 = 0.1 \tag{10} $$

To solve the equation given in (10), we can make the trigonometric substitutions

$$ \begin{cases} P_x = \rho\cos\varphi \\ P_y = \rho\sin\varphi \end{cases} \tag{11} $$

where

$$ \begin{cases} \rho = \sqrt{P_x^2 + P_y^2} \\ \varphi = A\tan 2(P_x, P_y) \end{cases} $$

Substituting (11) into (10), we get

$$ \sin(\varphi - \theta_1) = \frac{0.1}{\rho} $$

Consequently,

$$\cos(\varphi - \theta_1) = \pm\sqrt{1 - \left(\frac{0.1}{\rho}\right)^2}$$

Thus,

$$(\varphi - \theta_1) = A\tan 2\left(\frac{0.1}{\rho}, \pm\sqrt{1 - \left(\frac{0.1}{\rho}\right)^2}\right)$$

Finally, the solution for θ_1 can be yield as follows

$$\theta_1 = A\left[\tan 2(P_x, P_y) - \tan 2\left(\frac{0.1}{\rho}, \pm\sqrt{1 - \left(\frac{0.1}{\rho}\right)^2}\right)\right] \quad (12)$$

which gives two possible solutions for θ_1 corresponding to the plus-or-minus sign in (12).

In order to find the solution for f, by multiplying first and second equation respectively by $c\theta_1$ and $s\theta_1$ in (9), after simple manipulations we have

$$P_x c\theta_1 + P_y s\theta_1 = f s\theta_2 \quad (13)$$

Now by squaring (13) and also third equation in (9), we have

$$f^2 = (P_x c\theta_1 + P_y s\theta_1)^2 + (P_z - 0.4)^2$$

So

$$f = \pm\sqrt{(P_x c\theta_1 + P_y s\theta_1)^2 + (P_z - 0.4)^2} \quad (14)$$

which yields two possible solutions for$f$ corresponding to the plus-or-minus sign in (14) for each value of θ_1. Thus, we have four possible solutions for f.

Now that θ_1 and f are known, we can obtain a solution forθ_2 according to (13) and third equation in (9) as given below:

$$\theta_2 = A\tan 2\left(\frac{P_x c\theta_1 + P_y s\theta_1}{f}, \frac{P_z - 0.4}{f}\right) \quad (15)$$

Notice that we have four different solutions for θ_2 corresponding to different values of θ_1 and f.

For instance, following set of possible solutions can be obtained for the position of end-efector given in the project ($[0.3 \quad 0.15 \quad 0.25]$):

$$\begin{bmatrix}\theta_1\\ \theta_2\\ f\end{bmatrix}=\begin{bmatrix}9.2190^\circ\\ 115.1041^\circ\\ 0.3536^\circ\end{bmatrix},\qquad \begin{bmatrix}\theta_1\\ \theta_2\\ f\end{bmatrix}=\begin{bmatrix}9.2190^\circ\\ -64.8959^\circ\\ -0.3536^\circ\end{bmatrix}$$

$$\begin{bmatrix}\theta_1\\ \theta_2\\ f\end{bmatrix}=\begin{bmatrix}-136.0889^\circ\\ -115.1041^\circ\\ 0.3536^\circ\end{bmatrix},\qquad \begin{bmatrix}\theta_1\\ \theta_2\\ f\end{bmatrix}=\begin{bmatrix}-136.0889^\circ\\ 64.8959^\circ\\ -0.3536^\circ\end{bmatrix}$$

3. Matlab numerical results

Two Matlab function have been written to solve forward kinematic and inverse kinematic problems.

First, "Forward_kin" is the function to find the transformation ${}^{\mathrm{frame1}}_{\mathrm{frame2}}T$ which transforms vector defined in {frame2} to their description in {frame1} corresponding to D-H parameters, i.e. $\alpha, a, d,$ and θ. Syntax of this function is considered as

```
T=Forward_kin(alpha,a,d,theta,frame1,frame2)
```

For example, by considering following D-H parameters according table 1 and numerical values of

```
>> alpha=[0;-90;90];
>> a=[0;0;0];
>> d=[0.4;0.1;0.3536];
>> theta=[9.22;115.1;0];
```

Following numerical results are obtained using this function

```
>> T_0_1=Forward_kin(alpha,a,d,theta,0,1)

T_0_1 =

    0.9871   -0.1602         0         0
    0.1602    0.9871         0         0
         0         0    1.0000    0.4000
         0         0         0    1.0000
```

```
>> T_0_2=Forward_kin(alpha,a,d,theta,0,2)

T_0_2 =

   -0.4187   -0.8939   -0.1602   -0.0160
   -0.0680   -0.1451    0.9871    0.0987
   -0.9056    0.4242         0    0.4000
         0         0         0    1.0000
```

```
>> T_0_3=Forward_kin(alpha,a,d,theta,0,3)

T_0_3 =

   -0.4187   -0.1602    0.8939    0.3000
   -0.0680    0.9871    0.1451    0.1500
   -0.9056         0   -0.4242    0.2500
         0         0         0    1.0000
```

```
>> T_1_2=Forward_kin(alpha,a,d,theta,1,2)

T_1_2 =

   -0.4242   -0.9056         0         0
         0         0    1.0000    0.1000
   -0.9056    0.4242         0         0
         0         0         0    1.0000
```

```
>> T_1_3=Forward_kin(alpha,a,d,theta,1,3)

T_1_3 =

   -0.4242         0    0.9056    0.3202
         0    1.0000         0    0.1000
   -0.9056         0   -0.4242   -0.1500
         0         0         0    1.0000
```

```
>> T_2_3=Forward_kin(alpha,a,d,theta,2,3)

T_2_3 =

    1.0000         0         0         0
         0         0   -1.0000   -0.3536
         0    1.0000         0         0
         0         0         0    1.0000
```

Matlab code of this function is given in Appendix 1.

Second, "Inverse_kin" returns the possible solutions for f, θ_1 and θ_2 corresponding to the numerical values of position relevant to end-efector frame for Stanford manipulator. Following syntax is used to

```
[f,theta1,theta2]=Inverse_kin(position)
```

Then, following numerical results are obtained according to the position of end-efector given in the project

```
f =

 -.35355339059327376220042218105242
 -.35355339059327376220042218105242
  .35355339059327376220042218105242
  .35355339059327376220042218105242



theta1 =

  .16090165737780173492850695307865
 -2.3751990929659827409626378734357
 -2.3751990929659827409626378734357
  .16090165737780173492850695307865



theta2 =

 -1.1326472962107263035350605033458
  1.1326472962107263035350605033458
 -2.0089453573790669349275828799337
  2.0089453573790669349275828799337
```