

Assignment 1

ELEC 442 - Introduction to Robotics

Sondre Myrberg

September 25, 2017



**THE UNIVERSITY
OF BRITISH COLUMBIA**

1.

Given the homogenous transformation

$$\begin{bmatrix} \mathbf{y} \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} Q & \mathbf{d} \\ \mathbf{0}^\top & 1 \end{bmatrix}}_T \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}$$

where Q and \mathbf{d} accounts for rotation and translation, respectively. We have that the inverse is on the form

$$T^{-1} = \begin{bmatrix} \tilde{Q} & \tilde{\mathbf{d}} \\ \mathbf{0}^\top & 1 \end{bmatrix}$$

where we know that $T^{-1}T$ is equal to the 4×4 identity matrix. This yields

$$\begin{aligned} T^{-1}T &= \begin{bmatrix} \tilde{Q} & \tilde{\mathbf{d}} \\ \mathbf{0}^\top & 1 \end{bmatrix} \begin{bmatrix} Q & \mathbf{d} \\ \mathbf{0}^\top & 1 \end{bmatrix} \\ &= \begin{bmatrix} \tilde{Q}Q & \tilde{Q}\mathbf{d} + \tilde{\mathbf{d}} \\ \mathbf{0}^\top & 1 \end{bmatrix} = \mathbf{I}_{4 \times 4} \\ &\Rightarrow \begin{cases} \tilde{Q}Q &= \mathbf{I}_{3 \times 3} \\ \tilde{Q}\mathbf{d} + \tilde{\mathbf{d}} &= \mathbf{0} \end{cases} \\ &\Rightarrow \begin{cases} \tilde{Q} &= Q^{-1} = Q^\top \\ \tilde{\mathbf{d}} &= -\tilde{Q}\mathbf{d} = -Q^\top \mathbf{d} \end{cases} \\ &\Rightarrow T^{-1} = \underline{\begin{bmatrix} Q^\top & -Q^\top \mathbf{d} \\ \mathbf{0}^\top & 1 \end{bmatrix}} \end{aligned} \tag{1}$$

2.

Considering the homogenous transformation matrix

$${}^0T_1 = \begin{bmatrix} Q & \mathbf{d} \\ \mathbf{0}^\top & 1 \end{bmatrix}$$

with

$$Q = \underbrace{\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{Q_1} \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{1}{2} \end{bmatrix}}_{Q_2}$$

and

$$\mathbf{d} = \begin{bmatrix} -\frac{5}{\sqrt{2}} \\ \frac{5}{\sqrt{2}} \\ 4 \end{bmatrix} \text{ cm}$$

2a).

By observing the rotation matrices Q_1 and Q_2 we see that Q_1 is a simple rotation around the \mathbf{k} -axis. The angle of this rotation is given by $\theta = \arccos\left(\frac{1}{\sqrt{2}}\right) = \frac{\pi}{4}$. Q_2 is a simple rotation around the \mathbf{i} -axis and the rotation angle is given by $\alpha = \arccos\left(-\frac{1}{2}\right) = \frac{2\pi}{3}$. To determine d_1 and a_1 we recognize that a homogenous transformation matrix can be written as the product of four transformation matrices; angle, offset, length and twist. This gives us

$$\begin{aligned}
{}^0T_1 &= \begin{bmatrix} Q & \mathbf{d} \\ \mathbf{0}^\top & 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \exp(\theta \mathbf{k} \times) & \mathbf{0} \\ \mathbf{0}^\top & 1 \end{bmatrix}}_{\text{angle}} \underbrace{\begin{bmatrix} \mathbf{I} & d\mathbf{k} \\ \mathbf{0}^\top & 1 \end{bmatrix}}_{\text{offset}} \underbrace{\begin{bmatrix} \mathbf{I} & a\mathbf{i} \\ \mathbf{0}^\top & 1 \end{bmatrix}}_{\text{length}} \underbrace{\begin{bmatrix} \exp(\alpha \mathbf{i} \times) & \mathbf{0} \\ \mathbf{0}^\top & 1 \end{bmatrix}}_{\text{twist}} \\
&= \begin{bmatrix} \exp(\theta \mathbf{k} \times + \alpha \mathbf{i} \times) & \exp(\theta \mathbf{k} \times)(a\mathbf{i} + d\mathbf{k}) \\ \mathbf{0}^\top & 1 \end{bmatrix} = \begin{bmatrix} Q & \mathbf{d} \\ \mathbf{0}^\top & 1 \end{bmatrix} \\
\Rightarrow \exp(\theta \mathbf{k} \times)(a\mathbf{i} + d\mathbf{k}) &= \begin{bmatrix} -\frac{2}{\sqrt{2}} \\ \frac{5}{\sqrt{2}} \\ 4 \end{bmatrix} \\
\Rightarrow \exp(\theta \mathbf{k} \times) \begin{bmatrix} a \\ 0 \\ d \end{bmatrix} &= \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ 0 \\ d \end{bmatrix} = \begin{bmatrix} \frac{a}{\sqrt{2}} \\ -\frac{a}{\sqrt{2}} \\ d \end{bmatrix} = \begin{bmatrix} -\frac{5}{\sqrt{2}} \\ \frac{5}{\sqrt{2}} \\ 4 \end{bmatrix} \\
\Rightarrow \begin{cases} a = -5 \\ d = 4 \end{cases}
\end{aligned}$$

And we have numeric values for all our four DH parameters.

2b).

For the point represented in coordinate system 1 by $\underline{x} = [1 \ 0 \ 0]^\top$ cm we get the representation in system 0 given by

$$\begin{aligned}
\begin{bmatrix} {}^0\mathbf{x} \\ 1 \end{bmatrix} &= {}^0T_1 \begin{bmatrix} {}^1\mathbf{x} \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{2\sqrt{2}} & -\frac{\sqrt{3}}{2\sqrt{2}} & -\frac{5}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{2\sqrt{2}} & -\frac{\sqrt{3}}{2\sqrt{2}} & \frac{5}{\sqrt{2}} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{1}{2} & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} \frac{6}{\sqrt{2}} \\ -\frac{6}{\sqrt{2}} \\ 4 \\ 1 \end{bmatrix} \\
\Rightarrow {}^0\mathbf{x} &= \begin{bmatrix} \frac{6}{\sqrt{2}} \\ -\frac{6}{\sqrt{2}} \\ 4 \end{bmatrix} \text{ cm}
\end{aligned}$$

2c).

For the opposite case, that we have a point represented in coordinate system 0 by $\underline{x} = [1 \ 0 \ 0]^\top$ cm we apply the inverse transformation matrix that is on the form we found in (1). This gives

$$\begin{aligned}
\begin{bmatrix} {}^1\mathbf{x} \\ 1 \end{bmatrix} &= {}^0T_1^{-1} \begin{bmatrix} {}^0\mathbf{x} \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 5 \\ -\frac{1}{2\sqrt{2}} & -\frac{1}{2\sqrt{2}} & \frac{\sqrt{3}}{2} & -2\sqrt{3} \\ -\frac{\sqrt{3}}{2\sqrt{2}} & -\frac{\sqrt{3}}{2\sqrt{2}} & -\frac{1}{2} & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} \frac{10+\sqrt{2}}{2} \\ -\frac{8\sqrt{3}+\sqrt{2}}{4} \\ \frac{16-\sqrt{6}}{4} \\ 1 \end{bmatrix} \\
\Rightarrow {}^1\mathbf{x} &= \begin{bmatrix} \frac{10+\sqrt{2}}{2} \\ -\frac{8\sqrt{3}+\sqrt{2}}{4} \\ \frac{16-\sqrt{6}}{4} \end{bmatrix} \text{ cm}
\end{aligned}$$

2d).

Task 2d)

3.

A MATLAB function named `DH_homog` is implemented with the code shown in Listing 1.

```
1 function T = DH_homog(theta, d, a, alpha)
2     i=[1;0;0];
3     k=[0;0;1];
4     angle = [expm(theta*skew(k)) zeros(3,1); zeros(1,3) 1];
5     offset = [eye(3) d*k; zeros(1,3) 1];
6     length = [eye(3) a*i; zeros(1,3) 1];
7     twist = [expm(alpha*skew(i)) zeros(3,1); zeros(1,3) 1];
8     T = angle*offset*length*twist;
9 end
```

Listing 1: MATLAB code to generate homogenous transformation matrix based on the Denavit-Hartenberg convention