Topics Covered:

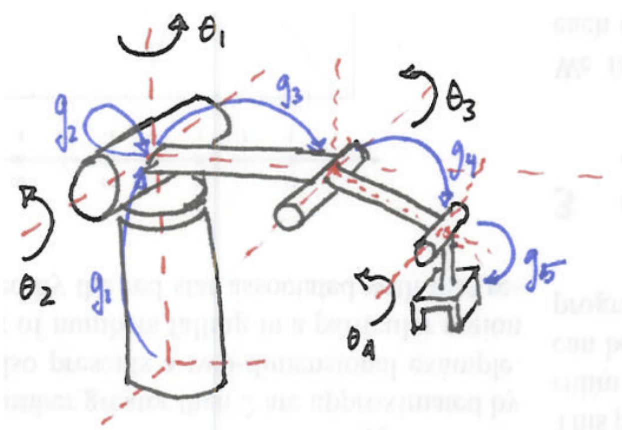
- Product of Exponentials
- Denavit Hartenberg Parameters

Additional Reading:

- LP 4.1; MLS Chapter 3, Section 2.2;

Review

Last class, we considered the following manipulator:



and used the product of homogeneous transformation matrices to derive the forward kinematics $g_E = g_{01}g_{12}g_{23}g_{34}g_{4E}$. Specifically, we obtained the displacements:

$$d_{01} = \begin{Bmatrix} 0 \\ 0 \\ l_0 \end{Bmatrix}, \quad d_{12} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}, \quad d_{23} = \begin{Bmatrix} 0 \\ l_1 \\ 0 \end{Bmatrix}, \quad d_{34} = \begin{Bmatrix} 0 \\ l_2 \\ 0 \end{Bmatrix}, \quad d_{4E} = \begin{Bmatrix} 0 \\ 0 \\ -l_3 \end{Bmatrix}$$

and the rotations:

$$\begin{array}{cc} R_{01} \rightarrow \text{rotation about z-axis} & R_{12} \rightarrow \text{rotation about x-axis} \\ R_{01} = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} & R_{12} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_2) & -\sin(\theta_2) \\ 0 & \sin(\theta_2) & \cos(\theta_2) \end{bmatrix} \\ R_{23} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_3) & -\sin(\theta_3) \\ 0 & \sin(\theta_3) & \cos(\theta_3) \end{bmatrix} & R_{34} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_4) & -\sin(\theta_4) \\ 0 & \sin(\theta_4) & \cos(\theta_4) \end{bmatrix} \\ R_{4E} = I & \end{array}$$

This class, we will discuss two alternative methods: the product of exponentials and Denavit-Hartenberg parameters. These methods are more commonly used in practice.

Product of Exponentials

The product of exponentials uses the formula:

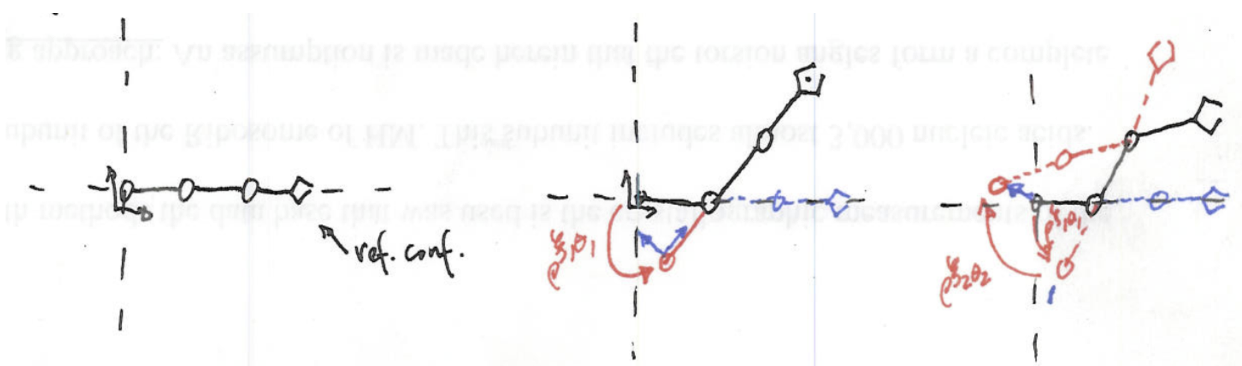
$$g_e = e^{\xi_1 \theta_1} \cdots e^{\xi_n \theta_n} g_0.$$

Note here that if $\vec{\theta} = 0$, then $e^0 = I$, and thus $g_e(0) = g_0$. Therefore, g_0 encodes the displacement that occurs at the "zero configuration", and is known as the reference configuration for the manipulator.

A given ξ_i will then say how the reference configuration changes for a given θ_i,

$$g_e(0, \cdots, \theta_i, \cdots, 0) = e^{\xi_i \theta_i} g_0$$

This can



What are the ξ_i? There are three basic types corresponding to the 3 single degree of freedom lower-pair joints:

1) revolute:	$\xi_i = \begin{Bmatrix} -\omega_i \times q_i \\ \omega_i \end{Bmatrix}$	ω_i - unit vector aligned w/ rotation axis q_i - point on the axis
2) prismatic:	$\xi_i = \begin{Bmatrix} v_i \\ 0 \end{Bmatrix}$	v_i - unit vector aligned w/ translation axis
3) helical/screw:	$\xi_i = \begin{Bmatrix} h\omega_i + [q_i]_\times \omega_i \\ \omega_i \end{Bmatrix}$	h - pitch of helical motion

These $\xi_i \in \mathfrak{se}(3)$ represent twists in spatial coordinates. Note here that $-\omega_i \times q_i = q_i \times \omega_i = [q_i]_\times \omega_i$.

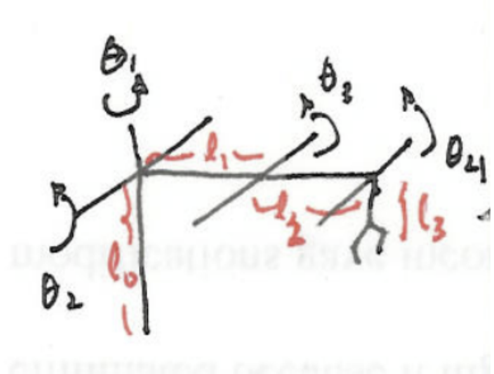
Based on seeing how end-effector changes for each joint from base to end-effector frame, can combine for all i to get total effect:

$$g_e = \underbrace{e^{\xi_1 \theta_1} \cdots e^{\xi_n \theta_n}}_{\text{order matters!}} g_0$$

Order here matters! The (ξ_i, θ_i) must enumerate from base to tool frame.

Example

Let's consider the example from before. For clarity, let's redraw the zero configuration:



With this zero configuration, the reference configuration is:

$$g_e(0) = g_0 = \begin{bmatrix} I & \begin{Bmatrix} 0 \\ l_1 + l_2 \\ l_0 - l_3 \end{Bmatrix} \\ 0 & 1 \end{bmatrix}$$

Next, we need to find each twist ξ_i that we will plug into the expression:

$$g_e(\theta) = e^{\hat{\xi}_1 \theta_1} \cdots e^{\hat{\xi}_4 \theta_4} g_0$$

What are the ξ_i?

ξ_1:	(rotation axis) $\omega_1 = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix}$, (point on axis) $q_1 = \begin{Bmatrix} 0 \\ 0 \\ l_0 \end{Bmatrix}$	$\xi_1 = \begin{Bmatrix} q_1 \times \omega_1 \\ \omega_1 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{Bmatrix}$
ξ_2:	$\omega_2 = \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix}$, $q_2 = \begin{Bmatrix} 0 \\ 0 \\ l_0 \end{Bmatrix}$, $q_2 \times \omega_2 = \begin{Bmatrix} 0 \\ l_0 \\ 0 \end{Bmatrix}$	$\xi_2 = \begin{Bmatrix} 0 \\ l_0 \\ 0 \\ 1 \\ 0 \\ 0 \end{Bmatrix}$
ξ_3:	$\omega_3 = \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix}$, $q_3 = \begin{Bmatrix} 0 \\ l_1 \\ l_0 \end{Bmatrix}$, $q_3 \times \omega_3 = \begin{Bmatrix} 0 \\ l_0 \\ -l_1 \end{Bmatrix}$	$\xi_3 = \begin{Bmatrix} 0 \\ l_0 \\ -l_1 \\ 1 \\ 0 \\ 0 \end{Bmatrix}$
ξ_4:	$\omega_4 = \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix}$, $q_4 = \begin{Bmatrix} 0 \\ l_1 + l_2 \\ l_0 \end{Bmatrix}$, $q_4 \times \omega_4 = \begin{Bmatrix} 0 \\ l_0 \\ -l_1 - l_2 \end{Bmatrix}$	$\xi_4 = \begin{Bmatrix} 0 \\ l_0 \\ -l_1 - l_2 \\ 1 \\ 0 \\ 0 \end{Bmatrix}$

Then, to actually compute each matrix exponential, we use our expressions for $e^{\hat{\xi}_i \theta_i}$:

$$e^{\hat{\xi}_i \theta_i} = \begin{bmatrix} \exp([\omega]_\times \tau) & (I - \exp([\omega]_\times \tau)) \frac{[\omega]_\times v}{\|\omega\|^2} + \frac{\omega \omega^\top}{\|\omega\|^2} v \tau \\ 0 & 1 \end{bmatrix}$$

and $\exp([\omega]_\times \tau)$ obtained using Rodrigues' formula:

$$\exp([\omega]_\times \tau) = I + \frac{[\omega]_\times}{\|\omega\|} \sin(\|\omega\| \tau) + \frac{[\omega]_\times^2}{\|\omega\|^2} (1 - \cos(\|\omega\| \tau))$$

Notice here that the upper-left block of the matrix exponential is still the rotation matrix, so in scenarios where ω aligns with a coordinate axis, we can use our rotation matrices as before. However, this method is more general and can handle any axis of rotation.

We can verify this by comparing to the product of Lie groups:

$$
\begin{aligned}
g_e(\theta) &= e^{\hat{\xi}_1\theta_1}\cdots e^{\hat{\xi}_4\theta_4} g_0 \\
e^{\hat{\xi}_1\theta_1} &= \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
e^{\hat{\xi}_2\theta_2} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta_2) & -\sin(\theta_2) & l_0\sin(\theta_2) \\ 0 & \sin(\theta_2) & \cos(\theta_2) & l_0(1-\cos(\theta_1)) \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
e^{\hat{\xi}_3\theta_3} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta_3) & -\sin(\theta_3) & l_0\sin(\theta_3) + l_1(1-\cos(\theta_3)) \\ 0 & \sin(\theta_3) & \cos(\theta_3) & l_0(1-\cos(\theta_3)) - l_1(\sin(\theta_3)) \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
e^{\hat{\xi}_4\theta_4} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta_4) & -\sin(\theta_4) & l_0\sin(\theta_4) + (l_1+l_2)(1-\cos(\theta_4)) \\ 0 & \sin(\theta_4) & \cos(\theta_4) & l_0(1-\cos(\theta_4)) - (l_1+l_2)\sin(\theta_4) \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
$$

Multiplying this all out should give us the same solution as before. Note: your homework will be to implement this example.

Denavit-Hartenberg Parameters

Lastly, the Denavit-Hartenberg parameters are a systematic way to describe the geometry of a manipulator. This method begins by assigning a specific frame of reference to each joint:

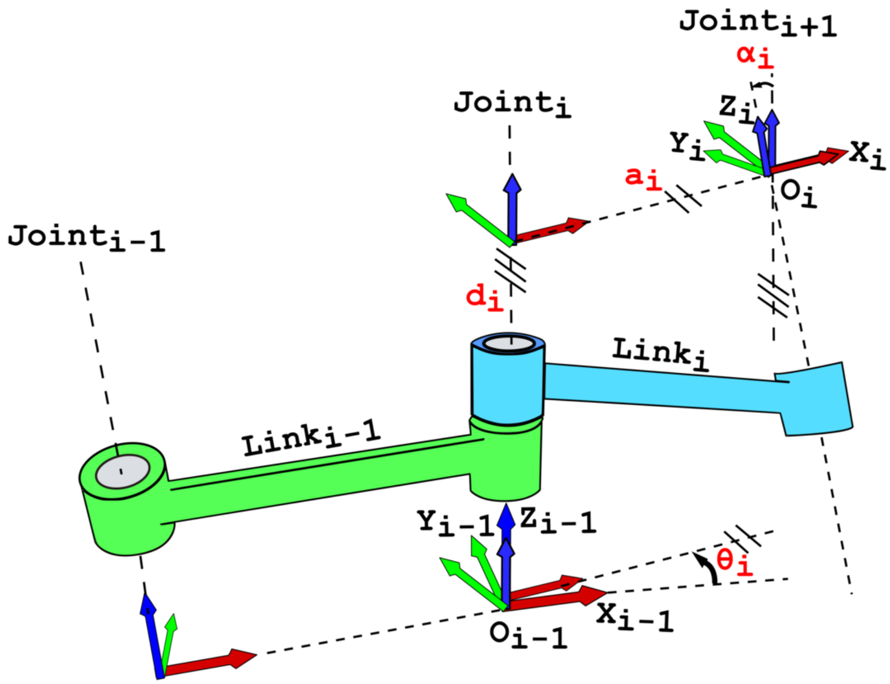
- the x-axis is parallel to the common normal (the line perpendicular to two non-intersecting joint axes)
- the z-axis is in the direction of the joint axis
- the y-axis follows from the x- and z-axes by choosing it to be a right-handed coordinate system

Using these reference frames, the method then assigns the following four parameters to each joint:

- a_i - distance from z_{i-1} to z_i along x_i axis
- α_i - angle from z_{i-1} to z_i about x_i axis

- d_i - distance from x_{i-1} to x_i along z_{i-1} axis
- θ_i - angle from x_{i-1} to x_i about z_{i-1} axis

For example, the diagram given by wikipedia is:



For our example, the parameters are as follows:

Link	a_i	α_i	d_i	θ_i
1	0	0	l_0	θ_1
2	l_1	$\pi/2$	0	θ_2
3	l_2	0	0	θ_3
4	l_3	0	0	$\theta_4 - \pi/2$

Once these parameters are assigned, the transformation matrix from frame $i-1$ to frame i can be written as:

$$A_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\cos(\alpha_i) & a_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\cos(\alpha_i) & a_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The forward kinematics can then be found by multiplying these matrices together:

$$g_E = A_1A_2A_3A_4$$

Note: your homework will also included implementing this method for the same manipulator.

When I was a student, I actually did my final project surrounding Denavit Hartenberg parameters. If you want to see the details, they're available here. Replicating a similar procedure could be an interesting final project for this class.