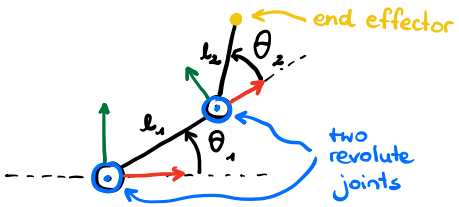


DH matrix:

	d	θ	r	α
J_1	0	θ_1	l_1	0
J_2	0	θ_2	l_2	0

We consider 3D coordinates to make the computations match the general case discussed during the lecture, but obviously everything happens in 2D for this robot

Transformation from end effector frame to base frame:

$$T(d_1)\cdot R(\theta_1)\cdot T(r_1)\cdot R(\alpha_1)\cdot T(d_2)\cdot R(\theta_2)\cdot T(r_2)\cdot R(\alpha_2) =$$

$$I \quad \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I & \begin{matrix} l_1 \\ 0 \\ 0 \end{matrix} \\ \begin{matrix} 0 & 0 & 0 \end{matrix} & 1 \end{bmatrix} \quad I \quad I \cdot \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I & \begin{matrix} l_2 \\ 0 \\ 0 \end{matrix} \\ \begin{matrix} 0 & 0 & 0 \end{matrix} & 1 \end{bmatrix} \quad I$$

$$= \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I & \begin{matrix} l_1 \\ 0 \\ 0 \end{matrix} \\ \begin{matrix} 0 & 0 & 0 \end{matrix} & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I & \begin{matrix} l_2 \\ 0 \\ 0 \end{matrix} \\ \begin{matrix} 0 & 0 & 0 \end{matrix} & 1 \end{bmatrix} =$$

$$= \left[\begin{array}{cc|c} \cos\theta_1 & -\sin\theta_1 & l_1\cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & l_1\sin\theta_1 \\ 0 & 0 & 1 \\ \hline 0 & 0 & 0 \end{array}\right] \left[\begin{array}{cc|c} \cos\theta_2 & -\sin\theta_2 & l_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & l_2\sin\theta_2 \\ 0 & 0 & 1 \\ \hline 0 & 0 & 0 \end{array}\right] =$$

$$= \left[\begin{array}{cc|c|c} \cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2 & -\cos\theta_1\sin\theta_2 - \sin\theta_1\cos\theta_2 & 0 & l_2\cos\theta_1\cos\theta_2 - l_2\sin\theta_1\sin\theta_2 + l_1\cos\theta_1 \\ \sin\theta_1\cos\theta_2 + \cos\theta_1\sin\theta_2 & -\sin\theta_1\sin\theta_2 + \cos\theta_1\cos\theta_2 & 0 & l_2\sin\theta_1\cos\theta_2 + l_2\cos\theta_1\sin\theta_2 + l_1\sin\theta_1 \\ \hline 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

$$= \begin{bmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) & 0 & l_2\cos(\theta_1+\theta_2) + l_1\cos\theta_1 \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) & 0 & l_2\sin(\theta_1+\theta_2) + l_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =: M(\theta_1, \theta_2)$$

If we have a point with coordinates: $[x_e, y_e, z_e, 1]^T$ in the end effector frame, we can get the coordinates in the base frame using $M(\theta_1, \theta_2)$:

$$\begin{bmatrix} x_{des} \\ y_{des} \\ z_{des} \\ 1 \end{bmatrix} = M(\theta_1, \theta_2) \begin{bmatrix} x_e \\ y_e \\ z_e \\ 1 \end{bmatrix}$$

Forward Kinematics

So... if we want to know where the end effector is, we just need to input the $\vec{0}$ vector (because the end effector is at the origin of its frame):

$$\begin{bmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) & 0 & l_2\cos(\theta_1+\theta_2)+l_1\cos\theta_1 \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) & 0 & l_2\sin(\theta_1+\theta_2)+l_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} =$$

$$= \begin{bmatrix} l_2\cos(\theta_1+\theta_2)+l_1\cos\theta_1 \\ l_2\sin(\theta_1+\theta_2)+l_1\sin\theta_1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} FK(\theta_1,\theta_2) \\ 1 \end{bmatrix} =: \begin{bmatrix} x_{des} \\ y_{des} \\ z_{des} \\ 1 \end{bmatrix}$$

← this is not interesting obviously

Inverse kinematics

Now we would like to know an inverse relation:

$$FK^{-1}: (x_{des}, y_{des}, z_{des}) \longrightarrow (\theta_1, \theta_2)$$

Sometimes it might be well defined, but often isn't. In our case:

$$x_d^2 + y_d^2 = l_2^2 + l_1^2 + 2l_1l_2\cos(\theta_1+\theta_2)\cos\theta_1 + 2l_1l_2\sin(\theta_1+\theta_2)\sin\theta_1 =$$

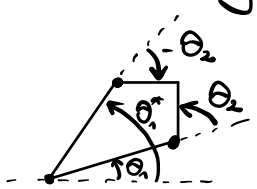
$$= l_2^2 + l_1^2 + 2l_1l_2\cos\theta_2$$

$$\rightsquigarrow \quad \cos\theta_2 = \frac{x_d^2 + y_d^2 - l_1^2 - l_2^2}{2l_1l_2}$$

→ $|\ | > 1$: no solutions for θ_2

→ $|\ | \leq 1$: 2 solutions

↳ they correspond to two ways we can reach a point:



Now, it's possible to find the formula for θ_1 as well, but instead...

Approximate Inverse Kinematics

... let's check how one of the approximation methods works. So, let's assume that we are in some known position $\vec{x}$ at which we know the values of joint parameters $q := [\theta_1, \theta_2]^T$ and want to move by some chosen, small $\Delta\vec{x}$:

$$\vec{x} + \Delta\vec{x} = FK(q + \Delta q) \approx FK(q) + DFK(q)\cdot\Delta q$$

($\vec{x}$ = known, $\Delta\vec{x}$ = known, q = known, Δq = unknown, $FK(q) = \vec{x}$)

And so: $\Delta q \approx DFK^{-1}(q)\cdot\Delta\vec{x}$

Note that $DFK(q)$ is known to us, because we evaluate the derivative at the current value of joint parameters q. The only unknown is Δq. In our case:

$$DFK(\theta_1, \theta_2) = \begin{bmatrix} -l_2 \sin(\theta_1+\theta_2) - l_1 \sin\theta_1 & -l_2 \sin(\theta_1+\theta_2) \\ l_2 \cos(\theta_1+\theta_2) + l_1 \cos\theta_1 & l_2 \cos(\theta_1+\theta_2) \\ 0 & 0 \end{bmatrix}$$

This is obviously not invertable, but if we make a (reasonable) assumption that we do not care about z coordinate, then we should be fine in most places.

Let's say that we are currently at $\vec{x} = [l_1, l_2]^T$ and want to move slightly away from the origin, i.e. $\Delta x = \left[\frac{l_1}{100}, \frac{l_2}{100}\right]^T$. Since our robot is currently at $\vec{x}$, we can check on our controls what is the current state of the parameters: $\vec{q} = [\theta_1, \theta_2]^T = [0, \frac{\pi}{2}]$. Hence (remember that we are ommitting z dimension now):

$$DFK(0, \tfrac{\pi}{2}) = \begin{bmatrix} -l_2 \sin\frac{\pi}{2} - l_1 \sin 0 & -l_2 \sin\frac{\pi}{2} \\ l_2 \cos\frac{\pi}{2} + l_1 \cos 0 & l_2 \cos\frac{\pi}{2} \end{bmatrix} = \begin{bmatrix} -l_2 & -l_2 \\ l_1 & 0 \end{bmatrix}$$

$$\rightsquigarrow DFK^{-1}(0, \tfrac{\pi}{2}) = \frac{1}{l_1 l_2} \begin{bmatrix} 0 & -l_1 \\ l_2 & -l_2 \end{bmatrix}^T = \frac{1}{l_1 l_2} \begin{bmatrix} 0 & l_2 \\ -l_1 & -l_2 \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{l_1} \\ -\frac{1}{l_2} & \frac{1}{l_1} \end{bmatrix}$$

And so we have to change the parameters by approximately:

$$\Delta q \approx DFK^{-1}(0, \tfrac{\pi}{2}) \cdot \Delta x = \begin{bmatrix} 0 & \frac{1}{l_1} \\ -\frac{1}{l_2} & \frac{1}{l_1} \end{bmatrix} \begin{bmatrix} \frac{l_1}{100} \\ \frac{l_2}{100} \end{bmatrix} = \frac{1}{100} \begin{bmatrix} \frac{l_2}{l_1} \\ -\frac{l_1}{l_2} + \frac{l_2}{l_1} \end{bmatrix}$$

Let's make a simplifying assumption that $l_1 = l_2 =: l$, so that we can easily compare this approximate result with the exact solution. In such case $\Delta q = \frac{1}{100}[1, 0]^T$ and so:

- the desired (exact) state was:

  $$\vec{x} + \Delta x = [l, l]^T + \frac{1}{100}[l, l]^T = \frac{101}{100}[l, l]^T$$

- the state reached using approximation is:

$$FK(\theta_1+\Delta\theta_1, \theta_2+\Delta\theta_2) = FK\left(\frac{1}{100}, \frac{\pi}{2}\right) = \begin{bmatrix} l\cos\left(\frac{1}{100}+\frac{\pi}{2}\right) + l\cos\frac{1}{100} \\ l\sin\left(\frac{1}{100}+\frac{\pi}{2}\right) + l\sin\frac{1}{100} \end{bmatrix} \approx$$

$$\approx l \cdot \begin{bmatrix} 0.98995 \\ 1.0099498 \end{bmatrix}$$

So, we have moved by approximately proper distance along both axes, but on the x axis we have moved in the incorrect direction. This shouldn't surprise us, since in the approximate case we change only θ_1 and so our robot moves like this: