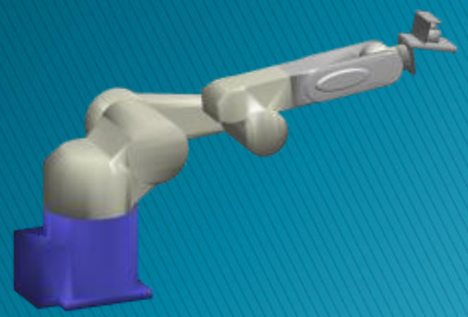
4.1 Forward Kinematics

Centre for Autonomous Systems
University of Technology Sydney



Roadmap

Rotation Matrices

$$\mathbf{R} = [\hat{x} \quad \hat{y} \quad \hat{z}]$$

→

Transformation Matrices

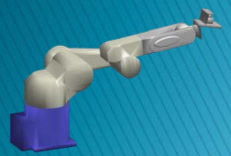
$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix}$$

→

Forward Kinematics

$$\mathbf{x} = \mathbf{f}(\mathbf{q})$$

Given a set of joint positions $\mathbf{q}$, what is the pose of the robot tool-tip $\mathbf{x}$?

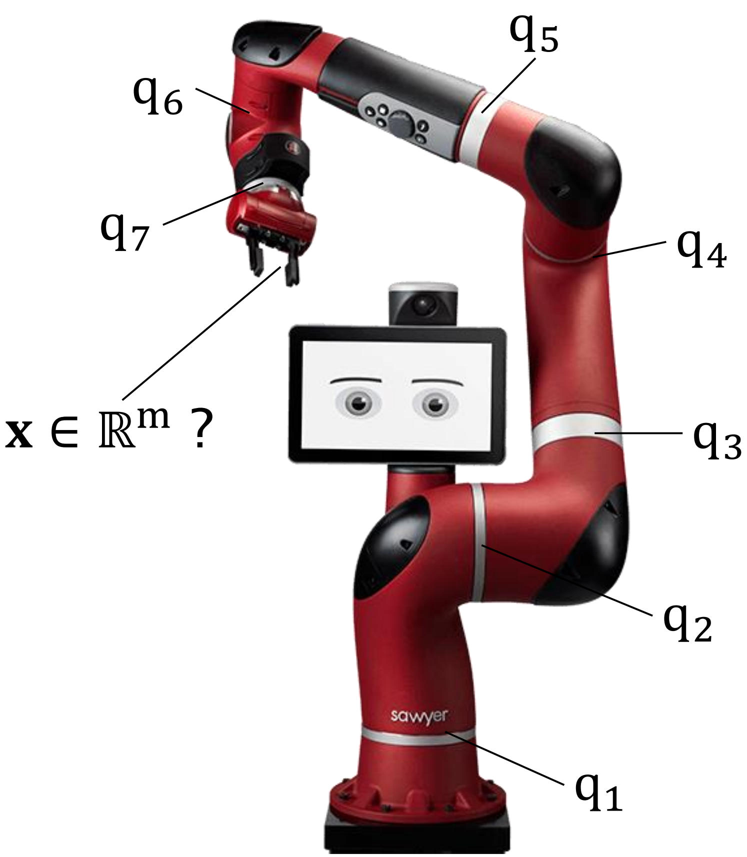


The Forward Kinematics Problem

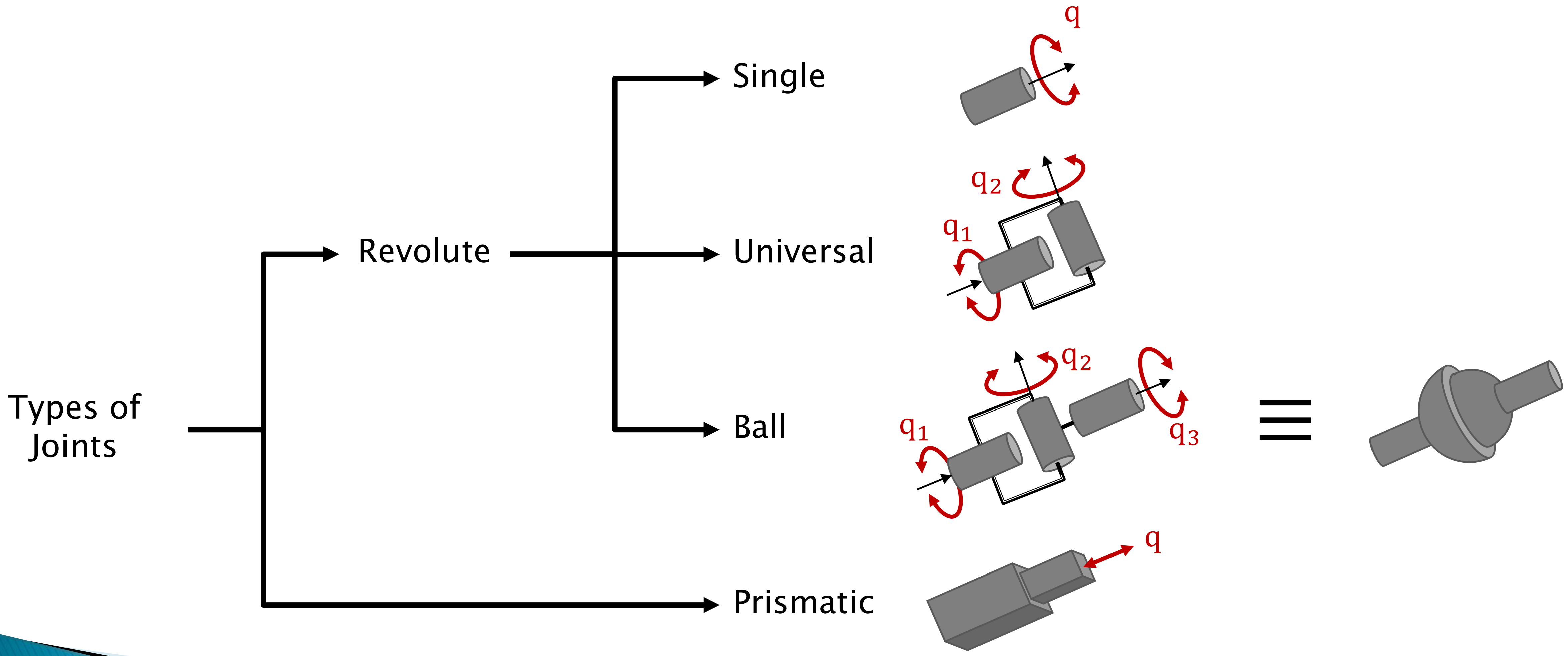
Given a set of joint angles/positions $\mathbf{q} \in \mathbb{R}^n$…

… determine the position and orientation (pose) of the robot tool-tip (end-effector) $\mathbf{x} \in \mathbb{R}^m$.

That is, solve the vector function $\mathbf{x} = \mathbf{f}(\mathbf{q})$, or some equivalent.



Types of Joints



Jon Woolfrey

Forward Kinematics of a 2DOF, Planar Manipulator

Task space:

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^2$$

Joint/control space:

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \in \mathbb{R}^2$$

Forward kinematics:

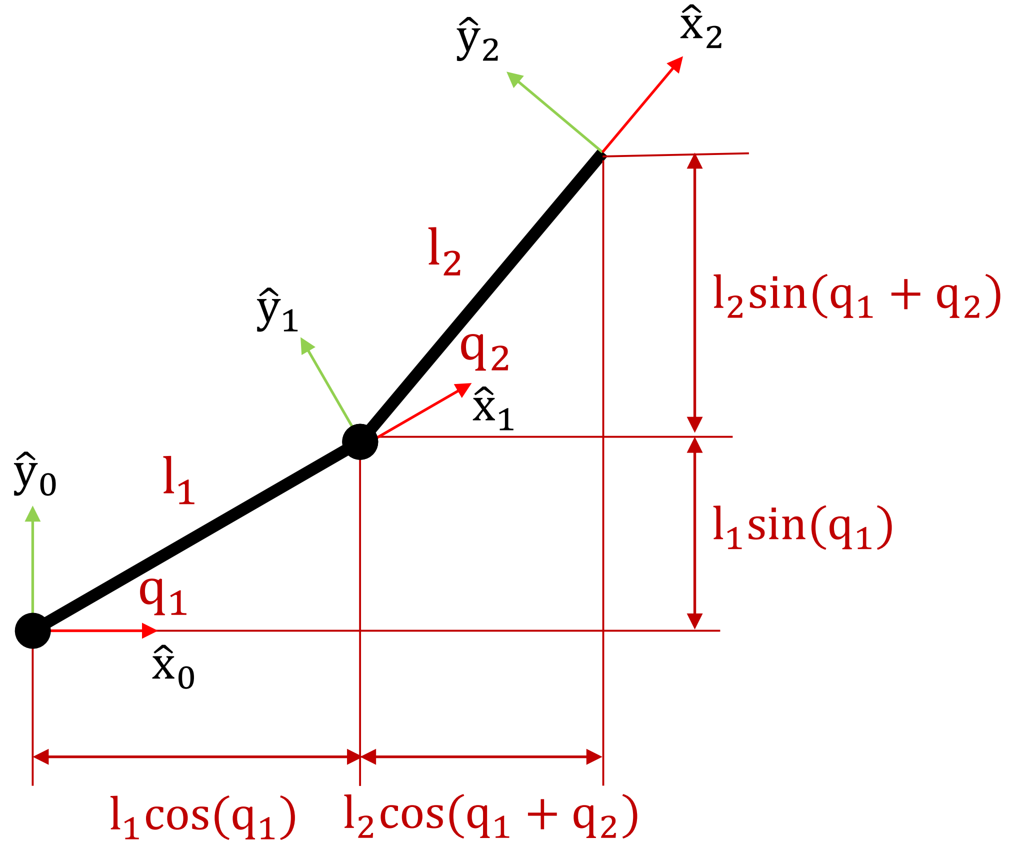
$$\mathbf{x} = \mathbf{f}(\mathbf{q})$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} l_1\cos(q_1) + l_2\cos(q_1+q_2) \\ l_1\sin(q_1) + l_2\sin(q_1+q_2) \end{bmatrix}$$

Simples!

What about orientation?

$$\begin{bmatrix} x \\ y \\ \psi \end{bmatrix} = \begin{bmatrix} l_1\cos(q_1) + l_2\cos(q_1+q_2) \\ l_1\sin(q_1) + l_2\sin(q_1+q_2) \\ q_1 + q_2 \end{bmatrix}$$



Jon Woolfrey

Forward Kinematics of a 3DOF Manipulator

Task Space:

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \mathbb{R}^3$$

Joint/control space:

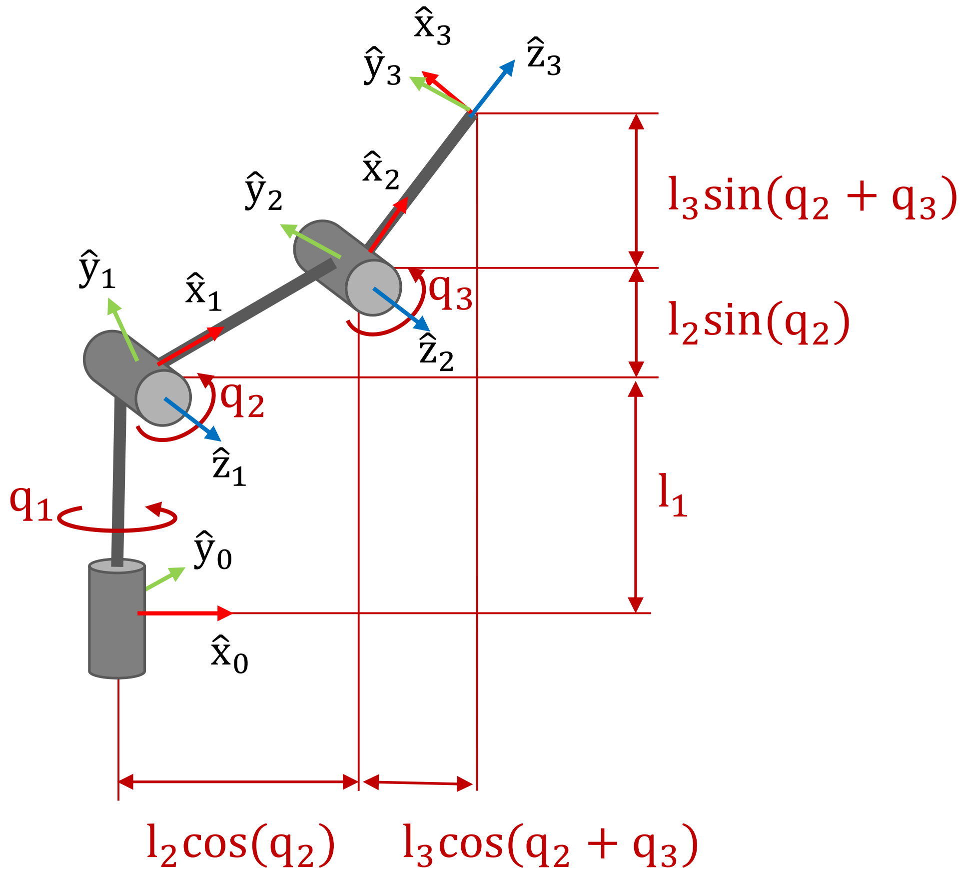
$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \in \mathbb{R}^3$$

Height of end-effector:

$$z = l_1 + l_2 \sin(q_2) + l_3 \sin(q_2 + q_3)$$

Also, distance of end-effector projected on x-y plane:

$$d = l_2 \cos(q_2) + l_3 \cos(q_2 + q_3)$$



Forward Kinematics of a 3DOF Manipulator

Distance of end-effector projected on x-y plane:

$$d = l_2\cos(q_2) + l_3\cos(q_2+q_3)$$

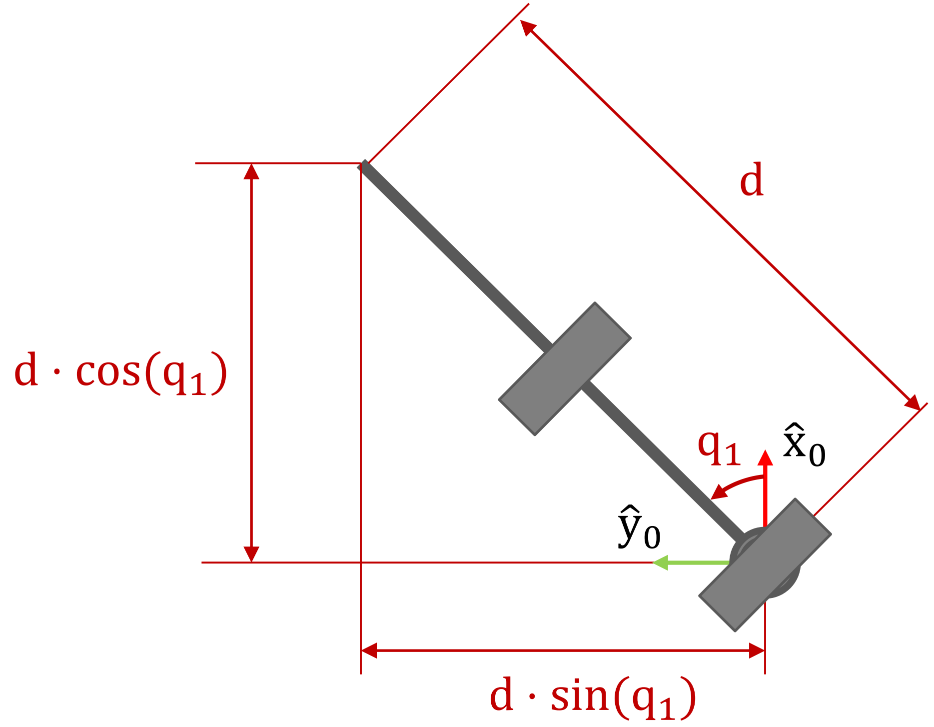
The x and y position of the end-effector is then:

$$\begin{aligned} x &= d\cdot\cos(q_1) \\ &= l_2\cos(q_1)\cos(q_2) + l_3\cos(q_1)\cos(q_2+q_3) \\ y &= d\cdot\sin(q_1) \\ &= l_2\sin(q_1)\cos(q_2) + l_3\sin(q_1)\cos(q_2+q_3) \end{aligned}$$

Forward kinematics:

$$\mathbf{x} = \mathbf{f}(\mathbf{q})$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} l_2\cos(q_1)\cos(q_2) + l_3\cos(q_1)\cos(q_2+q_3) \\ l_2\sin(q_1)\cos(q_2) + l_3\sin(q_1)\cos(q_2+q_3) \\ l_1 + l_2\sin(q_2) + l_3\sin(q_2+q_3) \end{bmatrix}$$



Alternative Approaches to Forward Kinematics

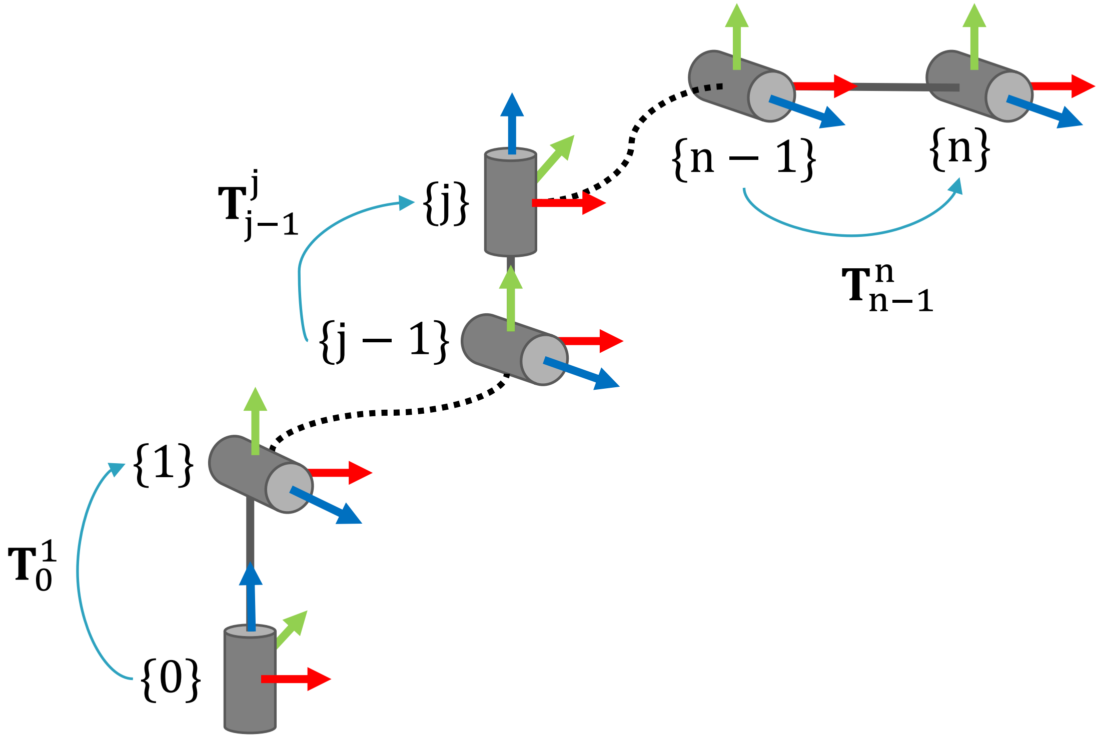
- Problem: Forward kinematics can get tricky
- Need to find a simpler way to derive forward kinematics
- Needs to be applied universally to *any* robot arm

Forward Kinematics Using Transformation Matrices

We can concatenate transformation matrices between joint frames to determine the end-effector pose.

$$\mathbf{T}_0^n = \mathbf{T}_0^1 \times \mathbf{T}_1^2 \times \mathbf{T}_2^3 \times \cdots \times \mathbf{T}_{n-1}^n$$

$$= \prod_{j=1}^{n} \mathbf{T}_{j-1}^{j}$$



Need to describe $\mathbf{T}_{j-1}^{j}$ as a function of simple geometry.

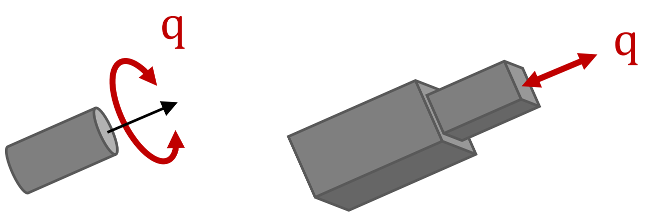
Summary of Forward Kinematics

The general Forward Kinematics (FK) problem expresses the end-effector pose $\mathbf{x} \in \mathbb{R}^m$ as a function of the joint positions $\mathbf{q} \in \mathbb{R}^n$:

$$\mathbf{x} = \mathbf{f}(\mathbf{q})$$

2 types of joints:

- Revolute (single, universal, ball)
- Prismatic



Alternatively, chain the homogeneous transforms from joint-to-joint to get the end-effector pose:

$$\mathbf{T}_0^n = \mathbf{T}_0^1 \times \mathbf{T}_1^2 \times \mathbf{T}_2^3 \times \cdots \times \mathbf{T}_{n-1}^n$$

$$= \prod_{j=1}^{n} \mathbf{T}_{j-1}^j$$