Rigid-Body Velocity and Robot Kinematics

Yuzhe Qin

Review: Homogenous Transformation

- General Rigid-body Motion:
  - $T^{4\times4} = \begin{bmatrix} R & p \\ 0_{1\times3} & 1 \end{bmatrix}$, where $R \in SO(3), p \in R^3$
  - T represent position and orientation in a single matrix
  - $\tilde{p} \triangleq \begin{bmatrix} p \\ 1 \end{bmatrix} \in R^4$
  - $T\tilde{p}$ change reference frame of p

Review: Physical Interpretation

- Rotation (**axis-angle**):
  - Any rotation in R^3 is equivalent to rotation about a fixed axis $\hat{\omega} \in R^3$ through an positive angle θ
  - Rotation: $\{\hat{\omega}, \theta\}$
- General Rigid-Body Motion (**screw motion**):
  - Any rigid body motion $SE(3)$ in R^3 is equivalent to rotating about a fixed axis $\hat{\omega} \in R^3$ through an positive angle θ while also translating along axis for d. This axis pass through point $q \in R^3$
  - Homogenous Transformation: $\{\hat{\omega}, \theta, q\}$

Topics

- **Exponential Coordinate of SE(3)**
- Rigid-Body Velocity
- Robot Kinematics
- Case study: Hand-Eye Calibration

Recall: the Lie Algebra of SO(3)

- Exponential coordinate:
  - $\widehat{\omega}\theta \in R^3$
- Skew-symmetric Matrix:
  - $[\widehat{\omega}]\theta \in so(3)$
- Rotation matrix:
  - $R = e^{\widehat{\omega}\theta} \in SO(3)$
- Interpretation:
  - Axis-angle

Goal: The Lie Algebra of SE(3)

- Exponential coordinate:
  - $\widehat{\omega}\theta \in R^3$
- Skew-symmetric Matrix:
  - $[\widehat{\omega}]\theta \in so(3)$
- Rotation matrix:
  - $R = e^{\widehat{\omega}\theta} \in SO(3)$
- Interpretation:
  - Axis-angle

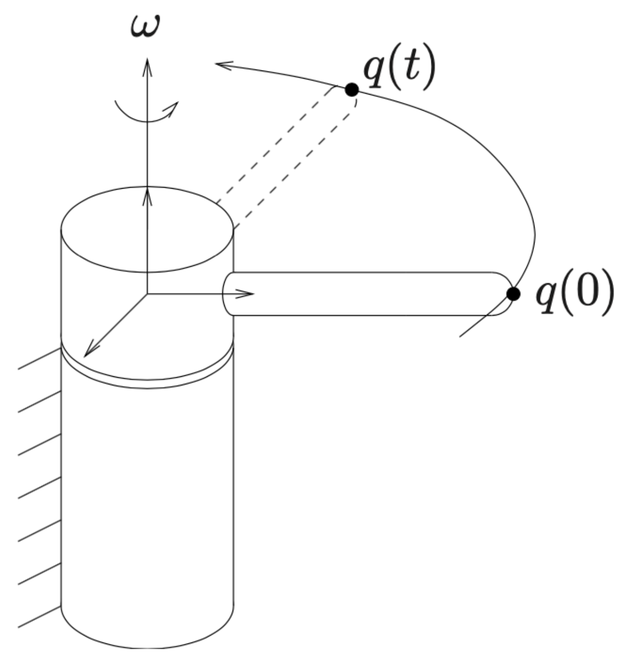
- Exponential coordinate:
  - $\hat{\xi}\theta \in R^6$
- $se(3)$ matrix:
  - $[\hat{\xi}]\theta \in se(3)$
- Homogenous transformation matrix:
  - T$= e^{\hat{\xi}\theta} \in SE(3)$
- Interpretation:
  - Screw motion

Recall: Find so(3) via ODE

- Consider a point q in body frame. At time $t = 0$, the position is q_0. Rotate q with **unit angular velocity** around axis $\widehat{\omega}$:
  - $v = \widehat{\omega} \times r$
  - $\dot{q}(t) = \widehat{\omega} \times q(t) = [\widehat{\omega}]q(t)$
  - $\mathrm{q(t)} = e^{[\widehat{\omega}]t} q_0$
  - Since $\theta(t) = t$

$$p(\theta) = e^{[\widehat{\omega}]\theta} p_0$$

  - $[\widehat{\omega}] \in so(3)$



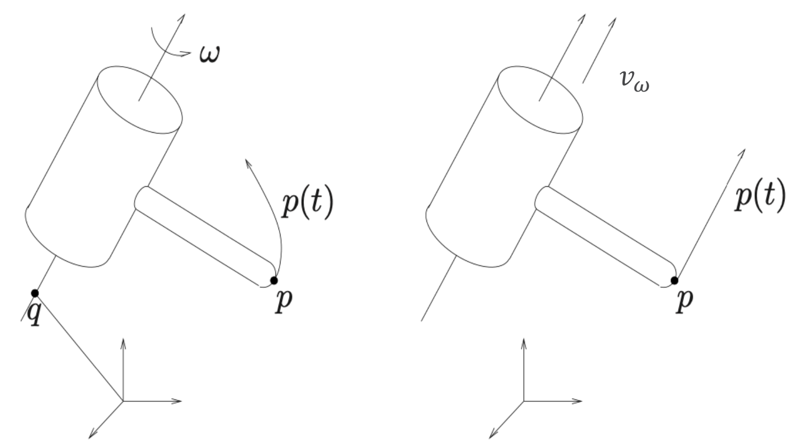
Find se(3) via ODE of Screw Motion

- Consider a point p in body frame. Rotate p with **unit angular velocity** around fixed axis $\widehat{\omega}$, q is any point on this axis, the linear velocity along axis $\widehat{\omega}$ is v_ω:
  - $\dot{p}(t) = \widehat{\omega}\times(p(t) - q) + v_\omega = [\widehat{\omega}]p(t) - \widehat{\omega}\times q + v_\omega$
  - $A \triangleq \begin{bmatrix} [\widehat{\omega}] & -[\widehat{\omega}]q + v_\omega \\ 0 & 0 \end{bmatrix} \triangleq \begin{bmatrix} [\widehat{\omega}] & v \\ 0 & 0 \end{bmatrix}$, $\tilde{p}(t) = e^{At}\tilde{p}_0$
  - Since $\theta(t) = t$

$$\tilde{p}\,(\theta) = e^{A\theta}\tilde{p}_0$$

  - For matrix $\mathrm{A}, e^{A\theta} \in SE(3)$

$$A \in se(3)$$



Exponential Coordinate of SE(3)

- For rotation, define $[\widehat{\omega}]\theta \in so(3)$

- For homogenous transformation, $\hat{\xi}\theta = \begin{bmatrix} \widehat{\omega} \\ v \end{bmatrix}\theta \in R^6$:

$$se(3) \triangleq \{[\hat{\xi}]\theta = \begin{bmatrix} [\widehat{\omega}] & v \\ 0 & 0 \end{bmatrix}\theta : [\omega]\theta \in so(3)\}$$

- Recall: $\widehat{\omega}\theta$ is the exponential coordinate of 3D rotation

- Similarly, $\hat{\xi}\theta \in R^6$ is the **exponential coordinate** of $SE(3)$

- $\hat{\xi}$ is the **direction of motion**, which is also called **Unit Twist**

Exponential Mapping of se(3)

- $\hat{\xi} = \begin{bmatrix} \widehat{\omega} \\ v \end{bmatrix}$, $e^{[\hat{\xi}]\theta} \in SE(3)$

$$e^{[\hat{\xi}]\theta} = I + [\hat{\xi}] + \frac{1 - cos\theta}{\theta^2}[\hat{\xi}]^2 + \frac{\theta - sin\theta}{\theta^3}[\hat{\xi}]^3$$

- This formula has **different form** compared to $e^{[\widehat{\omega}]\theta}$ for rotation
- Similarly, a log function exists uniquely: $SE(3) \rightarrow se(3)$
- Note that $\hat{\xi}$ means that the **first three value** has norm one, in another word $\widehat{\omega}$ is a unit vector, **no guarantee for v**

Example: $\hat{\xi}\theta$ to SE(3)

- Given $\hat{\xi}_{sb}^{s}\theta = [\pi, 0,0,0, \pi, 0]^T$, superscript: $\{s\}$ frame

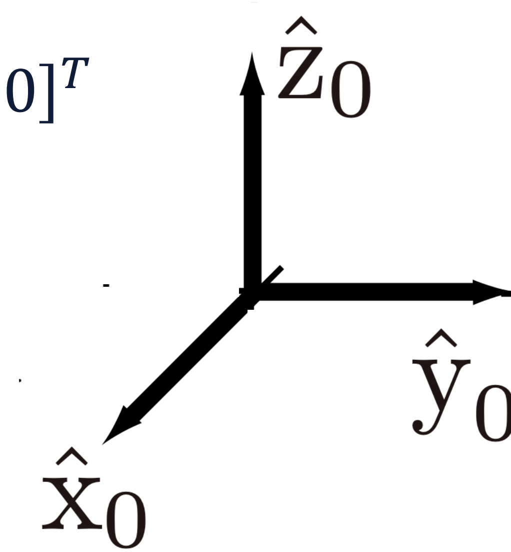
1. Find $\hat{\xi}_{sb}^{s} = [1,0,0,0,1,0]^T$, $\theta = \pi$

2. Find rotation matrix by $\widehat{\omega} = [1,0,0]^T$

3. Find screw axis $(\widehat{\omega}, q)$ by $\mathrm{v} = -\widehat{\omega}\times q + v_\omega$

$$v_\omega = [0,0,0]^T, -\widehat{\omega}\times q = [0,1,0]^T$$
$$\mathrm{q} = [0,0,1]^T$$

4. Find origin after transformation



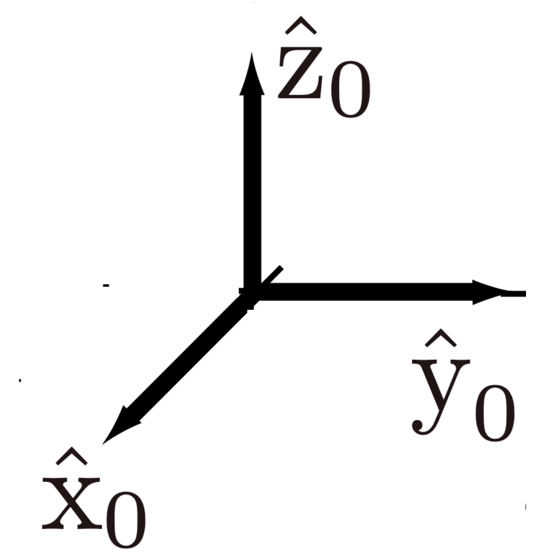
Example: $\hat{\xi}\theta$ to SE(3)

- Given $\hat{\xi}_{sb}^{s}\theta = [\pi, 0,0,0, \pi, 0]^T$, superscript: $\{s\}$ frame

1. $\mathrm{q} = [0,0,1]^T, \hat{\omega} = [1,0,0]^T, v_\omega = [0,0,0]^T$

2. Recall screw motion

$$T_{sb} = \begin{bmatrix} 1 & 0 & 0 & ? \\ 0 & -1 & 0 & ? \\ 0 & 0 & -1 & ? \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Example: $\hat{\xi}\theta$ to SE(3)

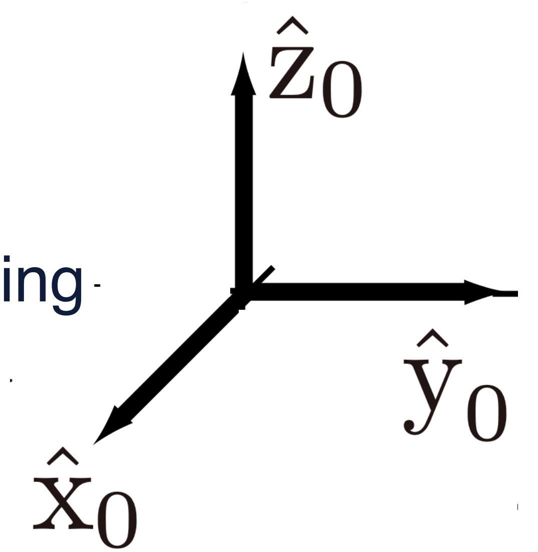
- Given $\hat{\xi}^{s}_{sb}\theta = [\pi, 0,0,0, \pi, 0]^T$, superscript: $\{s\}$ frame

1. $\text{q} = [0,0,1]^T, \hat{\omega} = [1,0,0]^T, v_\omega = [0,0,0]^T$

2. Recall screw motion

$$T_{sb} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

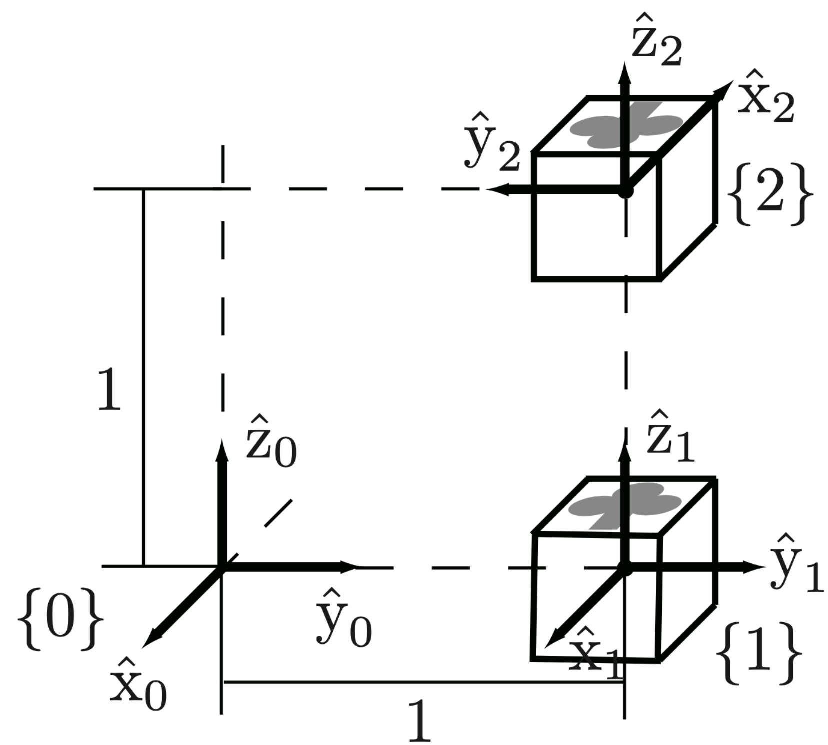
Same result using exponential mapping



Example: SE(3) to $\hat{\xi}\theta$

Given $SE(3)$, find screw motion and twist

- $T_{02} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

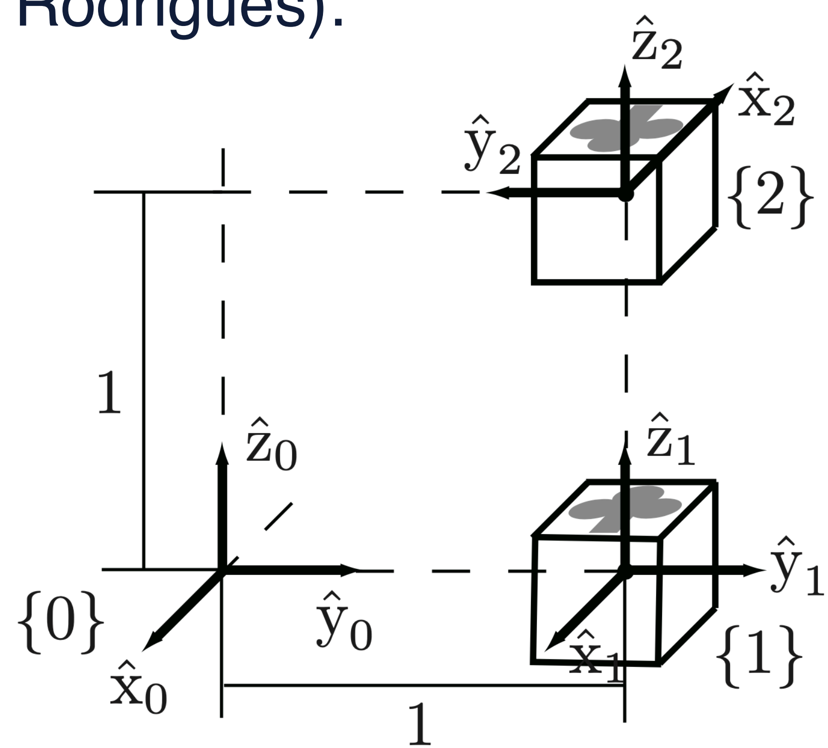


Example: SE(3) to $\hat{\xi}\theta$

Given $SE(3)$, find screw motion and twist

- $T_{02} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

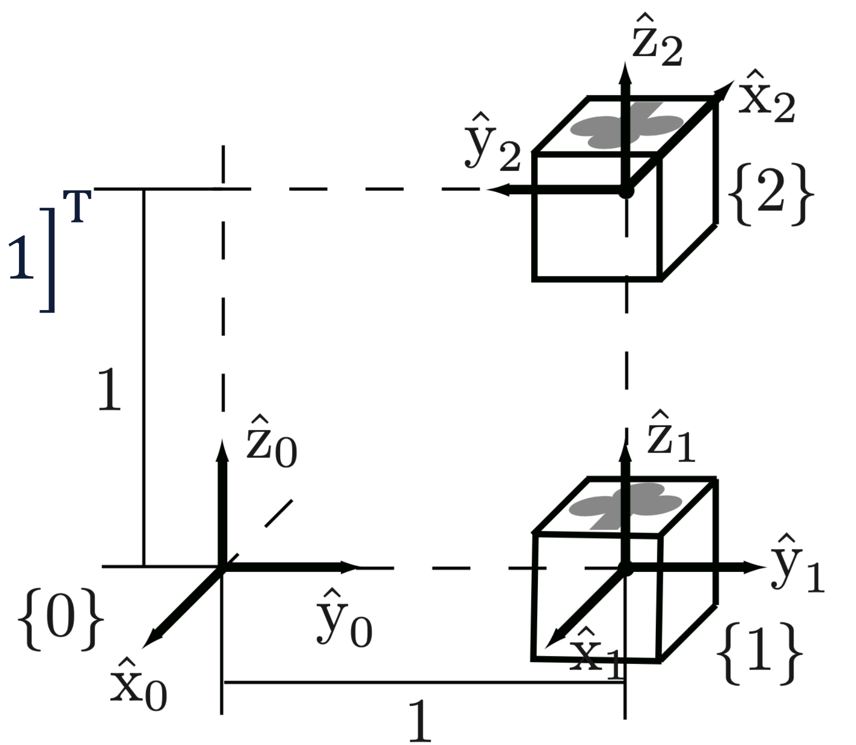
- Consider **rotation only** (Inverse Rodrigues):
  - $tr(R) = -1 \Rightarrow \theta = \pi$
  - $\widehat{\omega} = \frac{1}{\sqrt{2(1+r_{33})}} \begin{bmatrix} r_{13} \\ r_{23} \\ 1 + r_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$



Example: SE(3) to $\hat{\xi}\theta$

Given $SE(3)$, find screw motion and twist

- $T_{02} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \widehat{\omega} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$
- Find screw axis $q = \left[0, \frac{1}{2}, 0\right]^T$
- Since $\mathrm{v}\theta = -\omega \times q + v_\omega \theta = \left[\frac{\pi}{2}, 0, 1\right]^{\mathrm{T}}$
- Find $\hat{\xi}\theta = \left[0, 0, \pi, \frac{\pi}{2}, 0, 1\right]^T$



Comparison: SO(3) and SE(3)

- Exponential coordinate:
  - $\widehat{\omega}\theta \in R^3$
- Skew-symmetric Matrix:
  - $[\widehat{\omega}]\theta \in so(3)$
- Rotation matrix:
  - $R = e^{\widehat{\omega}\theta} \in SO(3)$
- Interpretation:
  - Axis-angle

- Exponential coordinate:
  - $\hat{\xi}\theta \in R^6$
- $se(3)$ matrix:
  - $[\hat{\xi}]\theta \in se(3)$
- Homogenous transformation matrix:
  - T$= e^{\hat{\xi}\theta} \in SE(3)$
- Interpretation:
  - Screw motion

Topics

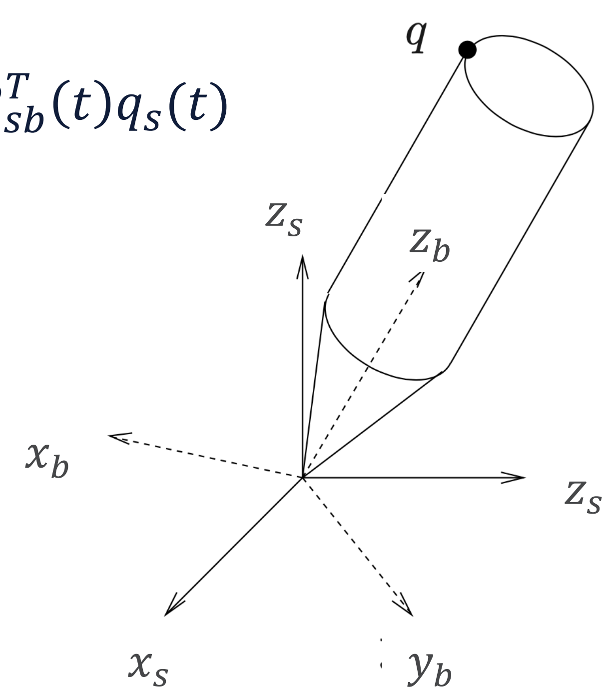
- Exponential Coordinate of SE(3)
- Rigid-Body Velocity
- Robot Kinematics
- Case study: Hand-Eye Calibration

Angular Velocity from SO(3)

Question: for moving frame $R(t)$, find angular velocity ω at any time t

Consider a point q fixed on a moving frame $\{b\}$:

- Fact 1, change reference frame:
  - $q_s(t) = R_{sb}(t)q_b \rightarrow \dot{q}_s(t) = \dot{R}_{sb}(t)q_b$
  - $\dot{q}_s(t) = \dot{R}_{sb}(t)R_{sb}^T(t)R_{sb}(t)q_b = \dot{R}_{sb}(t)R_{sb}^T(t)q_s(t)$
- Fact 2, physical interpretation:
  - $\dot{q}_s(t) = [\omega(t)]q_s(t) = [\omega(t)]q_s(t)$



Angular Velocity from SO(3)

Question: for moving frame $R(t)$, find angular velocity ω at any time t

- Fact 1:
  - $q_s(t) = R_{sb}(t)q_b \rightarrow \dot{q}_s(t) = \dot{R}_{sb}(t)q_b$
  - $\dot{q}_s(t) = \dot{R}_{sb}(t)R_{sb}^{-1}(t)R_{sb}(t)q_b = \dot{R}_{sb}(t)R_{sb}^{-1}(t)q_s(t)$
- Fact 2, physical interpretation:
  - $\dot{q}_s(t) = [\omega(t)]q_s(t)$
- For any q and ω $[\omega(t)]q_s(t) = \dot{R}_{sb}(t)R_{sb}^{-1}(t)q_s(t)$:

$$[\omega(t)] = \dot{R}_{sb}(t)R_{sb}^{-1}(t)$$

Angular Velocity of Rigid-Body

$$[\omega(t)] = \dot{R}_{sb}(t)R_{sb}^{-1}(t)$$

- Why $\dot{R}_{sb}(t)R_{sb}(t)$ can represent angular velocity?

- $\dot{R}_{sb}(t)R_{sb}(t)$ is a skew-symmetric matrix:

  - $R(t)R^T(t) = I \xrightarrow{derivative} \dot{R}(t)R^T(t) + R(t)\dot{R}^T(t) = 0$

  - $\dot{R}(t)R^T(t) = -R(t)\dot{R}^T(t) = -\left(\dot{R}(t)R^T(t)\right)^T$

  - $R(t)\dot{R}^T(t) \in so(3)$, which is a skew-symmetric matrix

General Velocity of Rigid-Body

Velocity of general motion can be represented as **twist**:

Similarly, $[\xi] = \dot{T}(t)\, T^{-1}(t), T \in SE(3)$

Question: the velocity is represented in which frame?

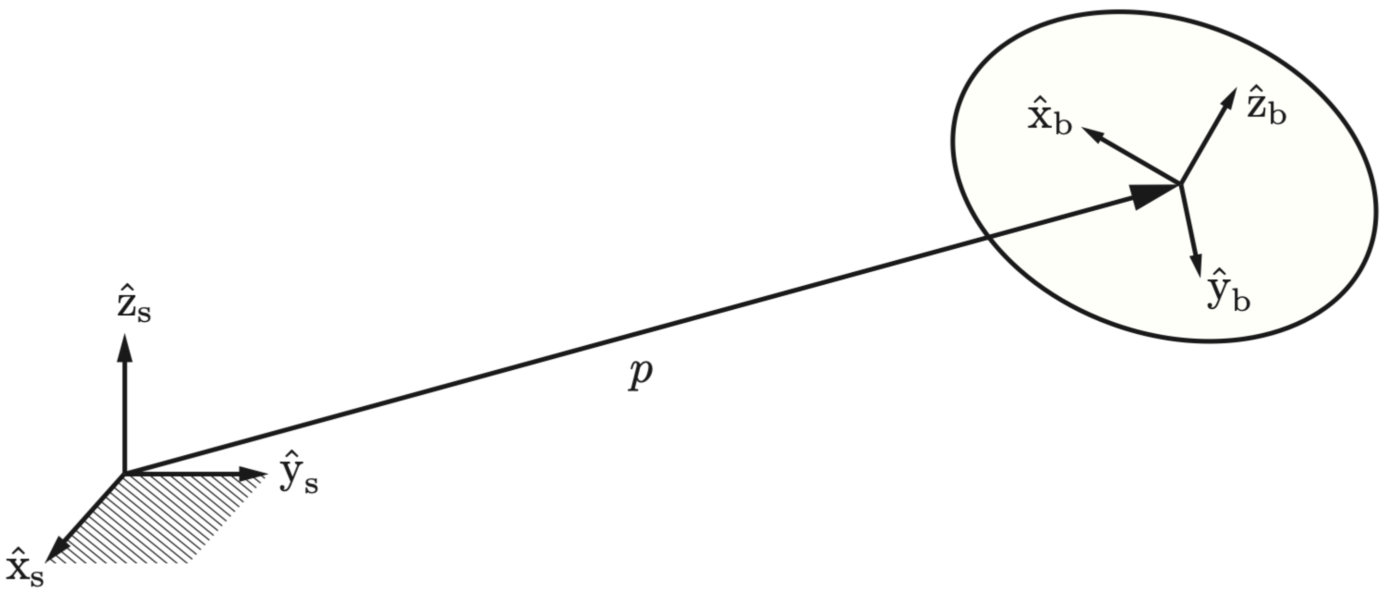
- Consider $\xi_{sb}^{s} = \dot{T}_{sb} T_{sb}^{-1}$, which is called **Spatial Twist**
  - Velocity of frame {b} observed in frame {s}, using the coordinate system of {s} (superscript) to represent velocity

What is a Twist?

- **Unit twist** is the **direction of motion**:
  - $\hat{\xi}\theta$ can representation $SE(3)$ motion
- **Twist** is the **velocity** of general rigid-body motion**:**
  - ξ contains angular velocity and “linear velocity”
  - ξt can representation $SE(3)$ motion

Change Frame of Twist

- How to change the reference frame of a twist
- $[\xi_{12}^{s}] = T_{12}[\xi_{12}^{b}]T_{12}^{-1}$
- physical interpretation?



Adjoint Matrix

- Adjoint matrix is used to change the reference frame of twist
- Adjoint matrix: $[Ad_T] \triangleq \begin{bmatrix} R & 0 \\ [p]R & R \end{bmatrix} \in R^{6\times 6}$
- $\xi_{12}^{s} = [Ad_{T_{sb}}]\xi_{12}^{b} = \begin{bmatrix} R_{sb}\omega_{12}^{b} \\ [p_{sb}]R_{sb}\omega_{12}^{b} + R_{sb}v_{12}^{b} \end{bmatrix}$
  - Where $T_{sb} = \begin{bmatrix} R_{sb} & p_{sb} \\ 0 & 1 \end{bmatrix}$, $\xi_{12}^{b} = \begin{bmatrix} \omega_{12}^{b} \\ v_{12}^{b} \end{bmatrix}$
- Equivalently, $[\xi_{12}^{s}] = T_{sb}[\xi_{12}^{b}]T_{sb}^{-1}$

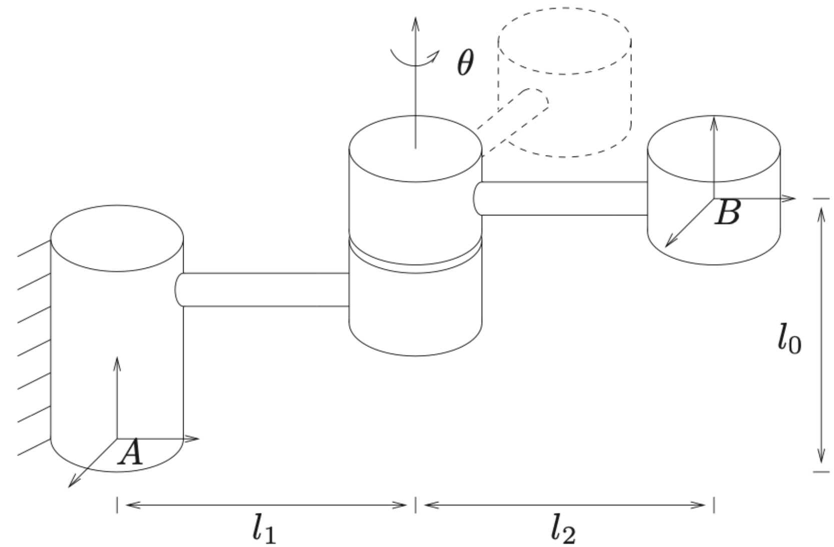
Example: General Rigid-Body Velocity

- Given the motion of rigid-body

  - $T_{BA}(t) = \begin{bmatrix} cos\theta(t) & -sin\theta(t) & 0 & -l_2 sin\theta(t) \\ sin\theta(t) & cos\theta(t) & 0 & l_1 + l_2 cos\theta(t) \\ 0 & 0 & 1 & l_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

- What is the spatial twist?

- What is the body twist?

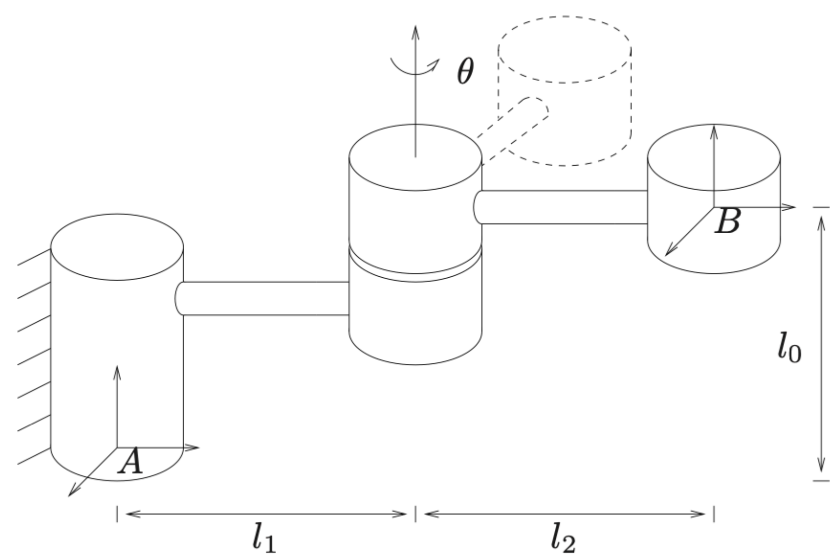


Example: General Rigid-Body Velocity

$$T_{BA}(t) = \begin{bmatrix} cos\theta(t) & -sin\theta(t) & 0 & -l_2 sin\theta(t) \\ sin\theta(t) & cos\theta(t) & 0 & l_1 + l_2 cos\theta(t) \\ 0 & 0 & 1 & l_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- $[\xi_{AB}^A] = \dot{T}T^{-1}, \xi_{AB}^A = []$

- $\xi_{AB}^B = T^{-1}\dot{T} =$



Topics

- Exponential Coordinate of SE(3)
- Rigid-Body Velocity
- **Robot Kinematics**
- Case study: Hand-Eye Calibration

Robot Kinematics

Kinematics:

- Motion of bodies including spatial relationship of different objects and their velocity. Kinematics **does not consider** how to achieve motion via force

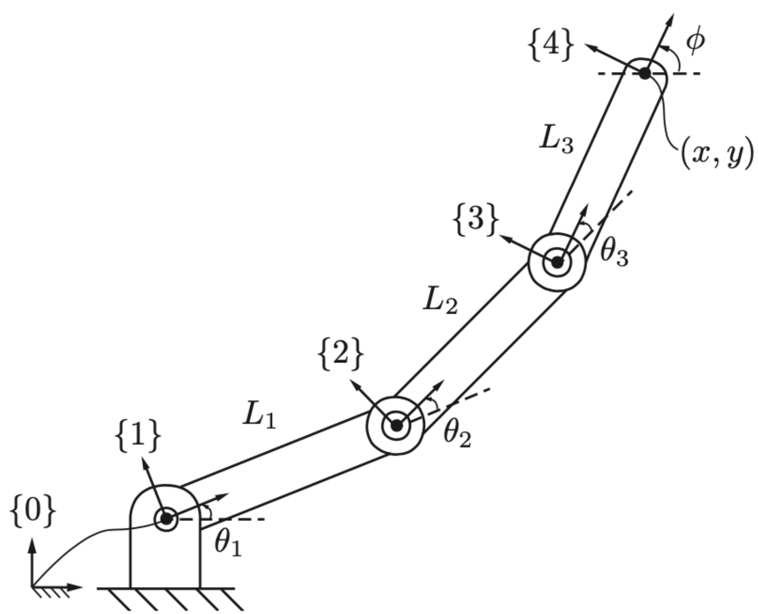
Link and Joint

Link:

- **Links** are the rigid-body connected in sequence

Joint:

- **Joints** are the movable components of a **robot/object** that cause relative motion between adjacent links

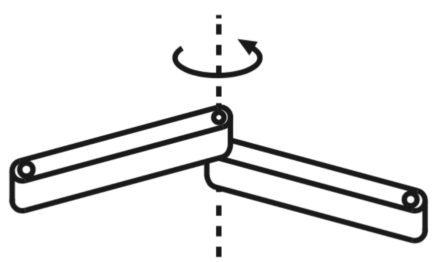


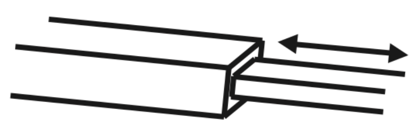
Two Common Joint Type

Joint:

- Revolute/Hinge/Rotational joint

- Prismatic/Translational joint



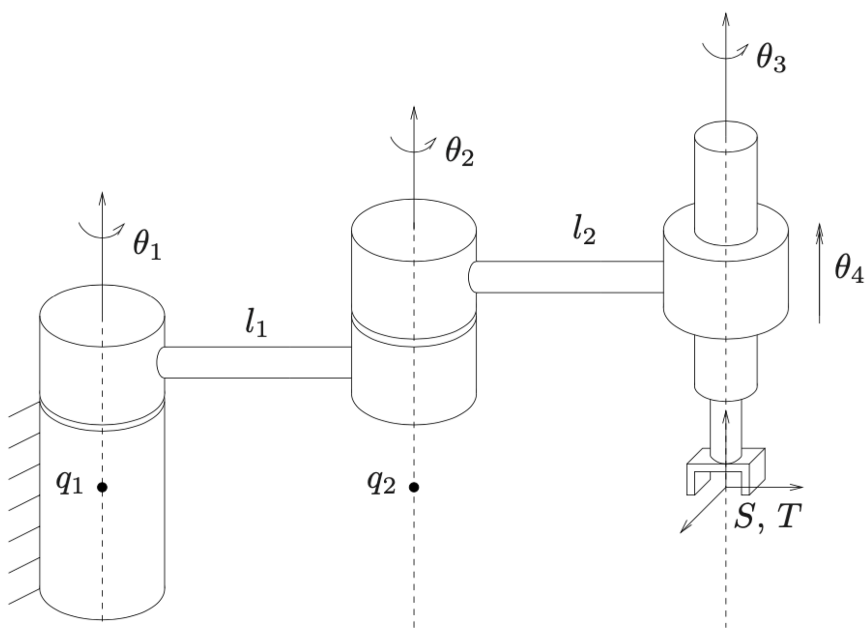


Forward Kinematics

Robot Forward Kinematics:

- Calculate the position and orientation of a robot link (often end-effector) given its joint variables $\theta = (\theta_1, \theta_2, \ldots\ldots, \theta_n)$

- Before calculation, we need to **assign a frame** at each robot link



Example: Forward Kinematics

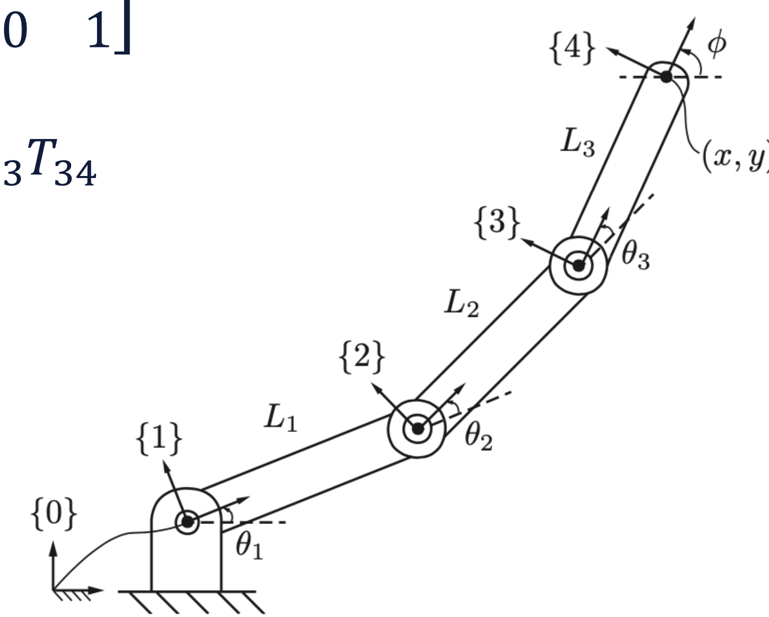
Robot forward kinematics calculation:

- Represent the motion of each joint in $SE(3)$

- Simply multiply each matrix

$$T_{01} = Rot(\hat{z}, \theta_1) \begin{bmatrix} cos\theta_1 & sin\theta_1 & 0 & 0 \\ sin\theta_1 & cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$, similarly T_{12}, T_{23}, T_{34}

$$T_{04} = T_{01} T_{12} T_{23} T_{34}$$

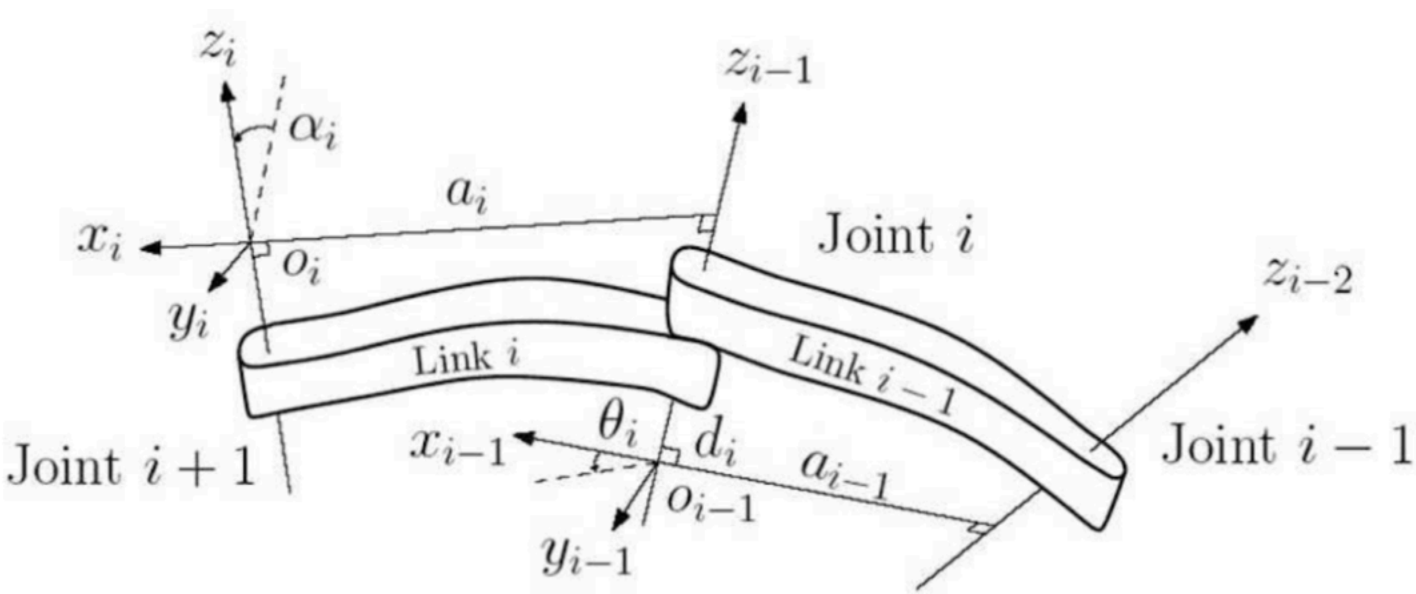


D-H Parameters

Frame assignment in 3D space is not trivial

Can we find a unique way to assign frames?

- **Denavit-Hartenberg (D-H) Parameters**:
  - Applying a set of rules which specify the position and orientation of frames attached to each link of the robot



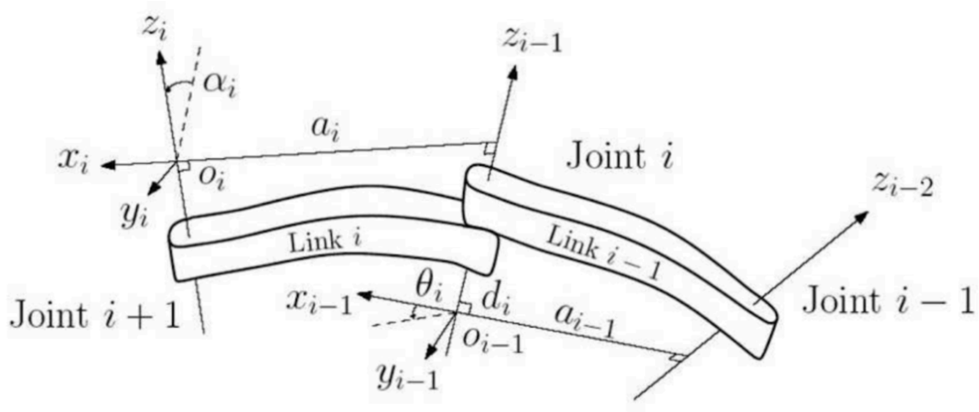
D-H Parameters

- **Denavit-Hartenberg (D-H) Parameters**:
  - Redundancy of link relative pose, no need to use 6-DoF $SE(3)$ representation

  - Link offset d_i: translate along z_{i-1}
  - Link length a_i: translate along x_i
  - Twist angle θ_i: rotate along x_i

- No need to understand these parameters in this course

- D-H parameters can **represent robot kinematics model**



Example: D-H Table

i	α_i	a_i	d_i	θ_i
1	$\pi / 2$	0	-D1	q1
2	$\pi / 2$	0	0	q2
3	$\pi / 2$	0	-(D2 + D3)	q3
4	$\pi / 2$	0	-e2	q4
5	$\pi / 2$	0	-(D4 + D5)	q5
6	$\pi / 2$	0	0	q6
7	π	0	-(D6 + D7)	q7

Jacobian

- Recall: In algebra course, Jacobian of a function $x = f(\theta)$ can be defined as, where $x \in R^m, \theta \in R^n$:

$$\mathrm{J}(\theta) \triangleq \left[\frac{\partial f}{\partial \theta}(\theta)\right] = \left[\frac{\partial f_i}{\partial \theta_j}\right] \in R^{m \times n}$$

- Jacobian can relate the derivative of two variable together, if both x and θ is the function of another variable t

$$\dot{x} = \left[\frac{\partial f}{\partial \theta}(\theta)\right]\frac{d\theta}{dt} = J(\theta)\dot{\theta}$$

Geometric Jacobian

Question: where will robot end-effector move given velocity of each joint?

- Velocity for a $SE(3)$ pose can be represented as twist ξ
- Geometric Jacobian $J(\theta)$:

$$\xi = \begin{bmatrix} \omega \\ v \end{bmatrix} = J(\theta)\dot{\theta}, \text{ where } J(\theta) \in R^{6\times n}, \text{ n is robot DoF}$$

- The i-th column of $J(\theta)$ is the twist when the robot is moving about the i-th joint at unit speed $\dot{\theta}_i = 1$ while all other joints stay static

Geometric Jacobian Calculation

Question: how to compute Geometric Jacobian?

- Using forward kinematics: $T_{sb}(\theta) = f(\theta)$
- General velocity: $[\xi_{sb}^{s}] = \dot{T}_{sb} T_{sb}^{-1}$, where $\dot{T}_{sb} = \frac{\partial T_{sb}}{\partial t} = g(\dot{\theta})$
- Compute each column of geometric Jacobian $J(\theta)$:

Let $\dot{\theta}_i = 1$ and $\dot{\theta}_j = 0$ for $j \neq i$

i-th column of $J(\theta)$ will be $\frac{\partial T_{sb}}{\partial \dot{\theta}} T_{sb}^{-1}$

Inverse Kinematics

Inverse Kinematics (IK):

- Given the forward kinematics $T(\theta)$ and the target pose $\mathrm{T}_{\mathrm{target}} \in SE(3)$, find solutions θ that satisfy $T(\theta) = T_{target}$

- **Analytical solution** of IK for robot with more than 3-DoF is very complex:
  - For a 6-DoF robot, you will need a several pages to write down the formula

- If you need analytical solution, just use libraries:
  - IKFast, IKBT

Numerical Solution of IK

Numerical IK, a root finding problem

- Inverse kinematics can be viewed as finding roots of a nonlinear equation with $SE(3)$ constrain

- Standard root-finding algorithm can be adapted for $T(\theta) = \xi^b$, e.g. Newton-Raphason method

- TBD

$$J(\theta^i) = \frac{\partial f}{\partial \theta}|_{\theta^i}$$

$$\Delta\theta = J^\dagger(\theta^i)\Delta\xi$$

Kinematic Singularity

Question: is it possible to move end-effector to any direction $\hat{\xi}$ for a robot with $DoF \geq 6$

- The pseudo inverse $J^{+}(\theta)$ map link twist back to joint velocity
- Kinematic singularity:
  - A **robot configuration** where the robot's end-effector loses the ability to move in one direction instantaneously
- Mathematically, $J(\theta)$ rank deficiency leads to kinematic singularity
- Kinematic singularity does not mean that there exists a configuration that is not accessible

Topics

- Exponential Coordinate of SE(3)
- Rigid-Body Velocity
- Robot Kinematics
- **Case study: Hand-Eye Calibration**

Vision-based Robotic Planning

- Visual perception comes from camera/lidar, the position and orientation is captured in sensor frame
  - E.g., point cloud in camera frame

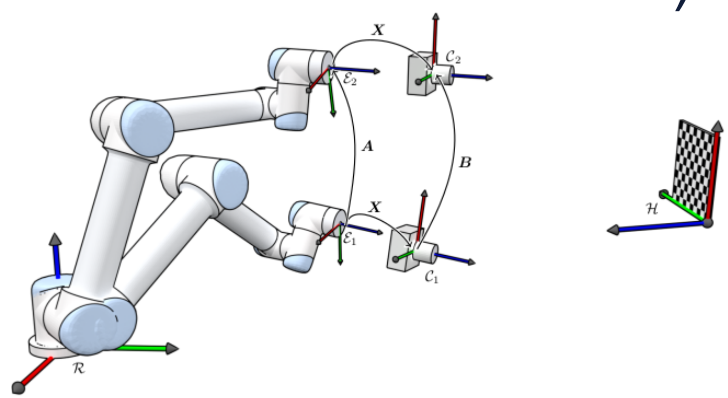
- Moving signal is command in robot frame
  - E.g., move the robot hand left in robot base frame

- Hand-eye Calibration computes the transformation from camera to robot

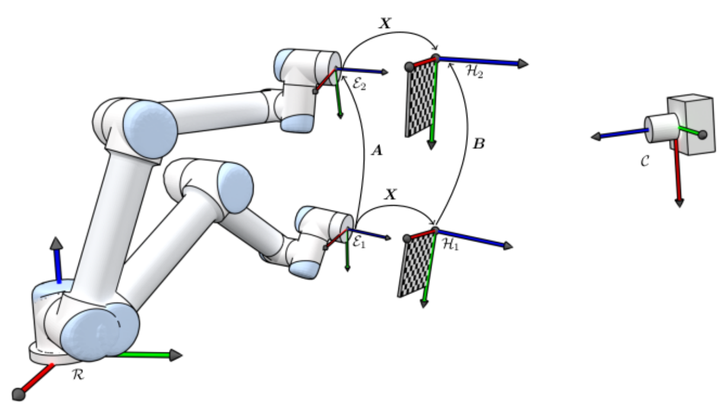
Settings of Hand-eye Calibration

- There are two kinds of problem for hand-eye calibration

- Eye-in-hand (camera mounted on hand):



- Eye-to-hand (camera not fixed with hand):



https://www.torsteinmyhre.name/snippets/robcam_calibration.html

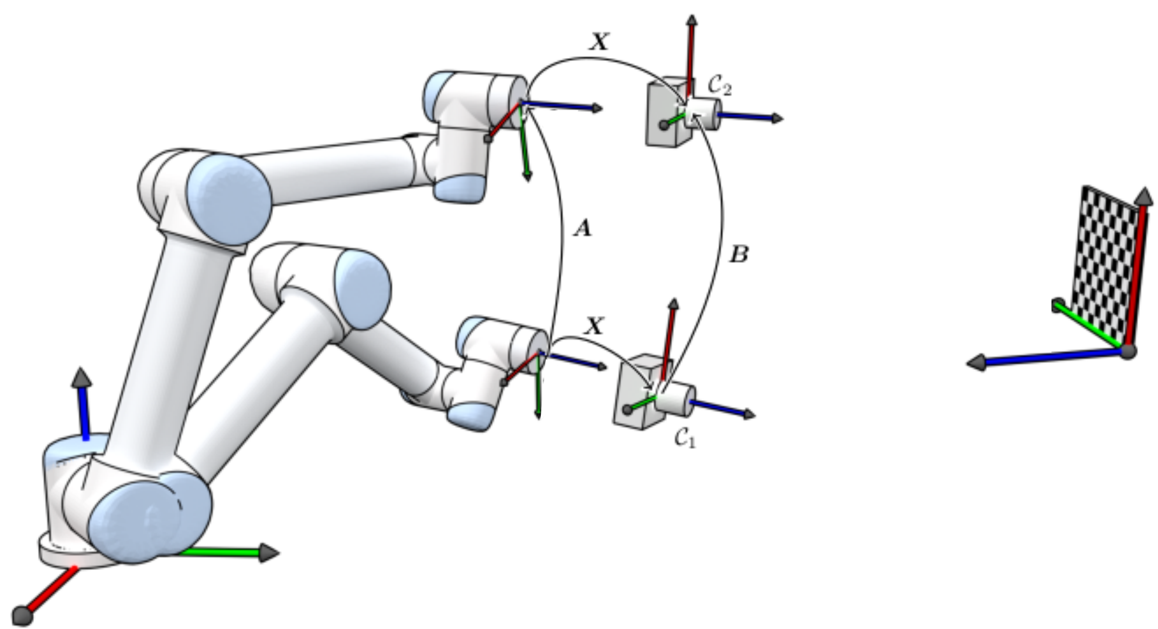
Hand-Eye Transformation Equation

Take *eye-in-hand* (e.g. camera fixed to hand) as example

- Goal: transformation from camera to hand T_{hc}

- Denote: space frame (robot base) $\{s\}$, camera $\{c\}$,hand $\{h\}$, and marker (auxiliary object) $\{m\}$

$$T_{sh}T_{hc}T_{cm} = \mathrm{T}_{sm}$$



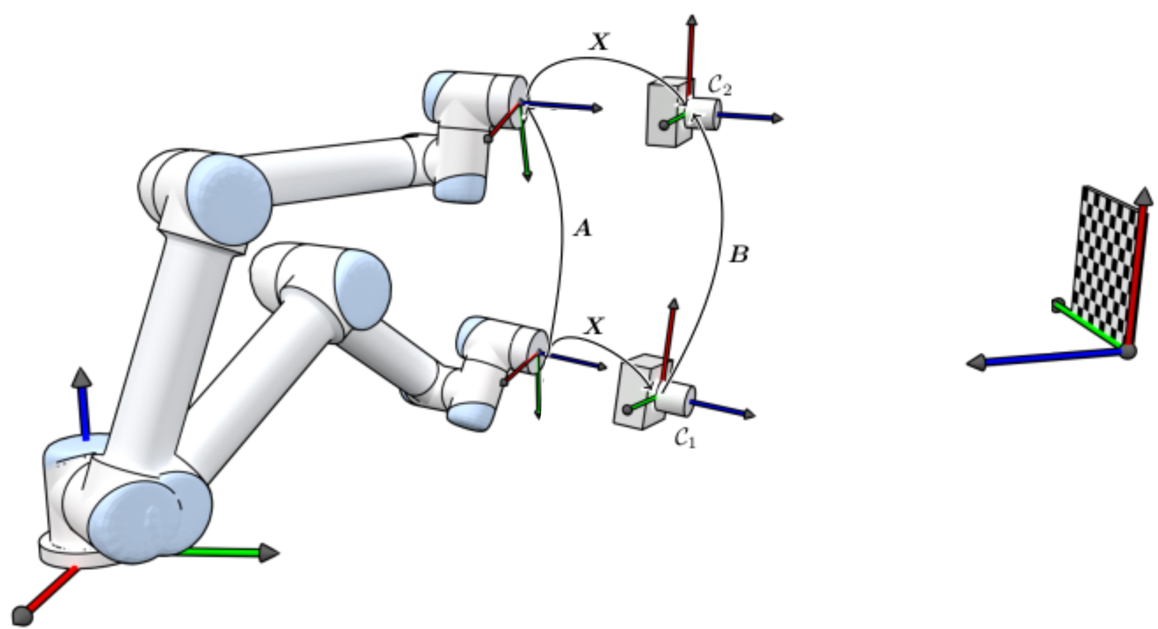
Hand-Eye Transformation Equation

Take *eye-in-hand* (e.g. camera fixed to hand) as example

- Assume we can get the pose of marker in camera frame

$$T_{sh}T_{hc}T_{cm} = \mathrm{T}_{sm}$$

T_{sh}	T_{hc}	T_{cm}	T_{sm}
Known Variant	What we want Constant	Assume known Variant	Unknown Constant



Capture Calibration Data

To solve the hand-eye transformation equation, we need to prepare multiple pairs of T_{sh} and T_{cm}

- Repeat the following steps for n-times:
  1. Move the robot hand to a target pose, where camera can see the marker

  2. Capture the T_{sh}^{i} for i-th pose of hand to base, often calculated by forward kinematics

  3. Capture the T_{cm}^{i} for i-th pose of marker to camera, calculated by a marker-specific algorithm

AX=XB for Hand-Eye Calibration

- The pose from marker to spatial frame T_{sm} is fixed

$$T_{sh}^{i}\, T_{hc}\, T_{cm}^{i} \; = \mathrm{T}_{sm} = T_{sh}^{i+1} \;\; T_{hc}\, T_{cm}^{i+1}$$

$$\left(T_{sh}^{i+1}\right)^{-1} T_{sh}^{i} T_{hc} = T_{hc}\; T_{cm}^{i+1} \left(T_{cm}^{i}\right)^{-1}$$

- Now we get a $AX = XB$ type function with constraints
  - $\mathrm{A} = \left(T_{sh}^{i+1}\right)^{-1} T_{sh}^{i}$ and $B = T_{cm}^{i+1} \left(T_{cm}^{i}\right)^{-1}$ are all known

- It is common to use multiple pairs of data for the equation. Actually, **at least three pairs** are necessary in order for a unique solution.

Horaud et al., "**Hand-eye Calibration**", *IJRR 1995*

Solving AX=XB

Note that the $X \in SE(3)$, which is a constrain to this equation

- Two mainstream to solve this equation

  1. Determine first rotation and then translation[1]

  2. Determine rotation and translation simultaneously[2]

- To solve the equation more precisely:
  - Poses of hand are chosen follow some **solver-specific guidelines**

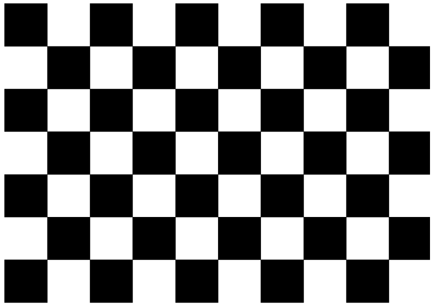
  - More data

Tsai et al., "**A new technique for fully autonomous and efficient 3 D robotics hand/eye calibration**", *TRO 1989*

Daniilidis, "**Hand-Eye Calibration Using Dual Quaternions**", *IJRR 1999*

Markers for Hand-eye Calibration

- Checkerboard is a most common visual marker in robotics:
  - Checkerboard pose can be easily solved using standard method like PnP



- As long as we have method to estimate its pose, anything can be a marker