Robot Dynamics Exercise Session 03

previous weeks: Ex. 1 a – forward kinematics $\{\vec{r}_{IE},\, C_{\mathcal{IE}}\} = \mathrm{FK}(\vec{q})$ unique mapping

Ex. 1 b – differential forward kinematics $\{\vec{v}_{IE},\, \vec{\omega}_{\mathcal{IE}}\} = \mathrm{DFK}(\vec{q},\, \dot{\vec{q}}) = J(\vec{q})\,\dot{\vec{q}}$ unique mapping

today: Ex. 1 c – inverse kinematics $\vec{q} = \mathrm{IK}\left(\vec{r}_{IE},\, C_{\mathcal{IE}}\right)$ non-unique mapping

issue: Difficult to solve analytically

solution: Resort to numerical solutions based on solving the differential inverse kinematics

1) IK & DIK

Let $\vec{q} \in \mathbb{R}^n$ be the joint variables, e.g., $q_1,\, q_2,\, \dots$

Let $\vec{\chi} \in \mathbb{R}^m$ be the task variables, e.g., $\vec{r}_{IE},\, C_{\mathcal{IE}},\, \dots$

IK problem: find $\vec{q}$ s.t. $\vec{\chi}_d = \vec{f}(\vec{q})$

DIK problem: find $\dot{\vec{q}}$ s.t. $\dot{\vec{\chi}}_d = J(\vec{q})\,\dot{\vec{q}} \longrightarrow \dot{\vec{q}} = J^{+}(\vec{q})\,\dot{\vec{\chi}}_d$

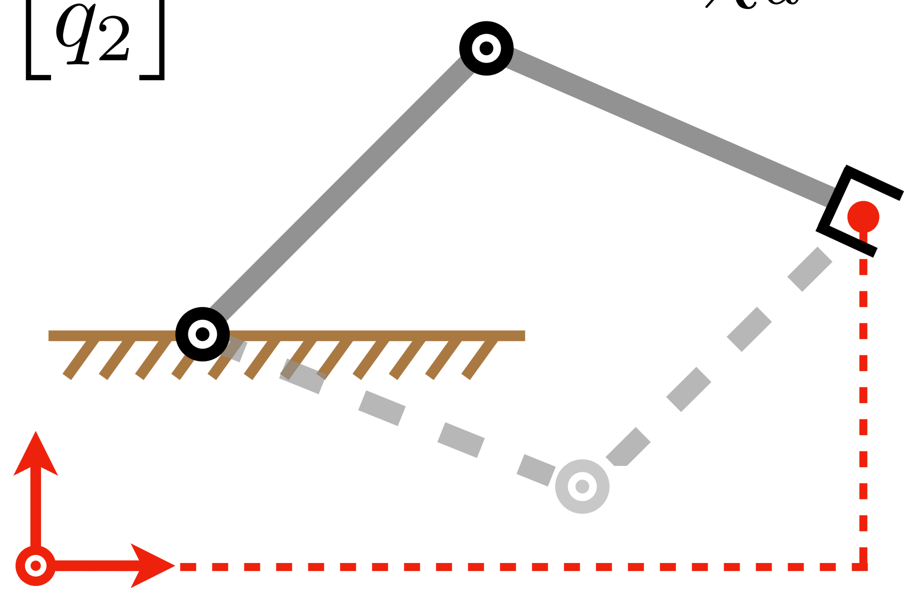
Robotic Systems Lab

Robot Dynamics Exercise Session 03

Assuming no kinematic singularities, we define the following three cases:

<u>Case 1:</u> $m = n$

$$\vec{q} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \qquad \vec{\chi}_d = \begin{bmatrix} x_d \\ y_d \end{bmatrix}$$



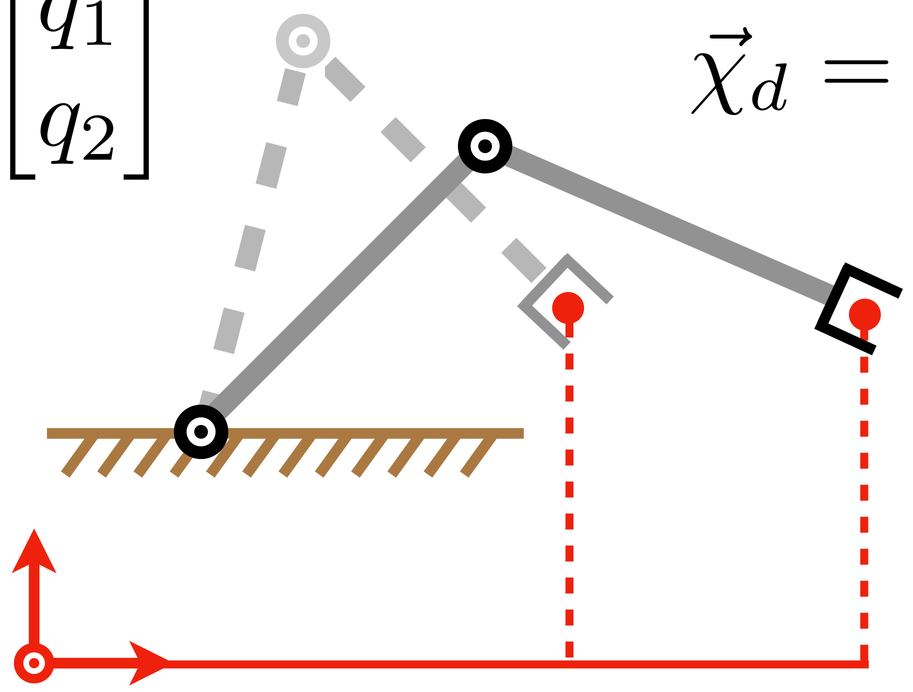
non unique but finitely many solutions for IK

$$\boxed{J^+(\vec{q}) = J^{-1}(\vec{q})}$$

J^+ is just J^{-1} as it is square

<u>Case 2:</u> $m < n$ (redundant)

$$\vec{q} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \qquad \vec{\chi}_d = \begin{bmatrix} y_d \end{bmatrix}$$



infinitely many solutions for IK

$$\boxed{J^+ = J^\top \left(J J^\top\right)^{-1}}$$

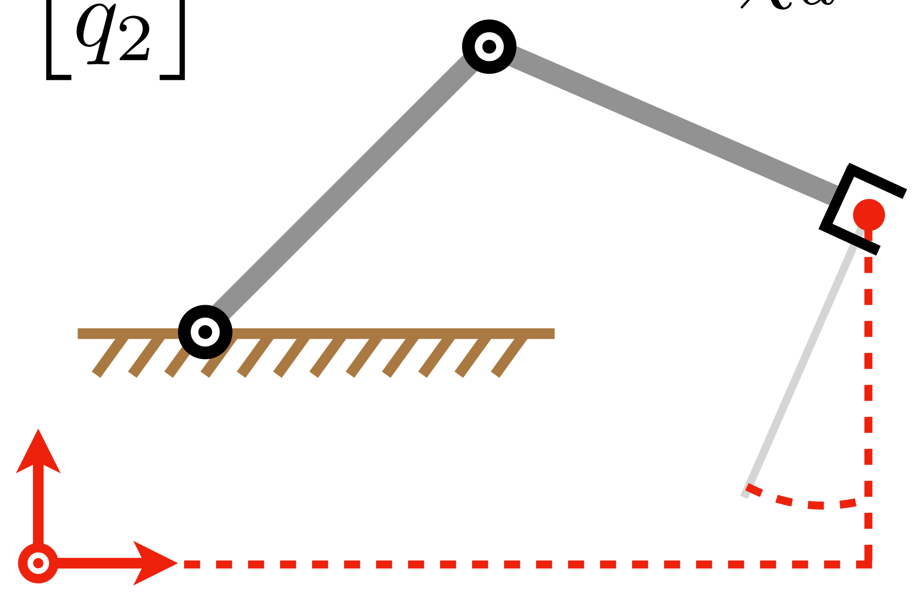
right pseudo-inverse

is constructed s.t.

$$\dot{\vec{q}}^* = \begin{cases} \underset{\dot{\vec{q}}}{\operatorname{argmin}} \ \frac{1}{2} \left\|\dot{\vec{q}}\right\|_2^2 \\ \text{s.t.} \quad \dot{\vec{\chi}}_d = J(\vec{q})\,\dot{\vec{q}} \end{cases}$$

<u>Case 3:</u> $m > n$

$$\vec{q} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \qquad \vec{\chi}_d = \begin{bmatrix} x_d \\ y_d \\ \alpha_d \end{bmatrix}$$



no exact solution for IK

$$\boxed{J^+ = \left(J^\top J\right)^{-1} J^\top}$$

left pseudo-inverse

is constructed s.t.

$$\dot{\vec{q}}^* = \underset{\dot{\vec{q}}}{\operatorname{argmin}} \ \frac{1}{2} \left\|\dot{\vec{\chi}}_d - J\,\dot{\vec{q}}\right\|_2^2$$

For kinematics singularities: Add $\lambda^2\,\mathbb{I}$ before taking the inverse (i.e. damping)

(Jacobian will have linearly dependent columns in case of singularity)

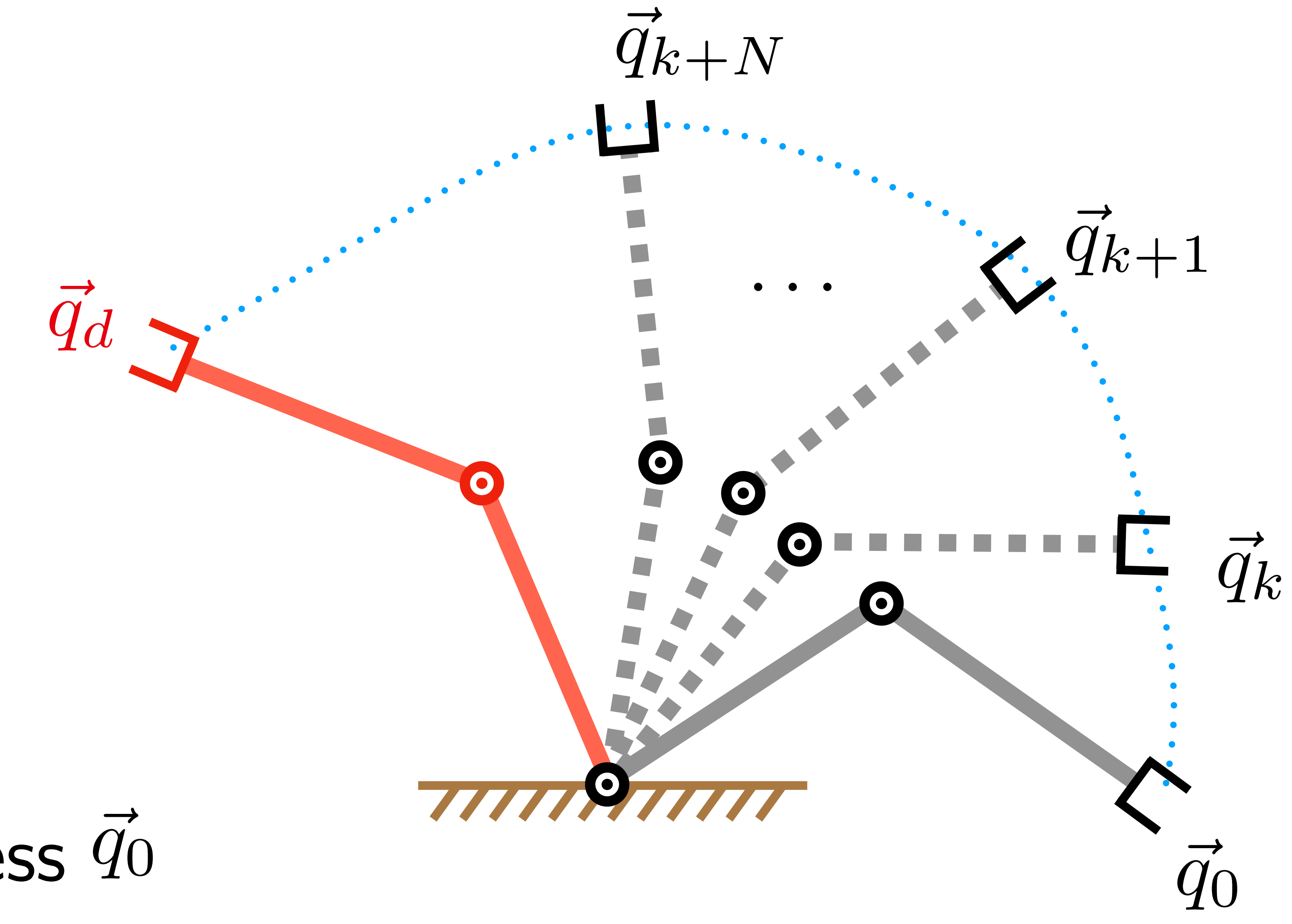
Robotic Systems Lab

Robot Dynamics Exercise Session 03

2) Iterative Inverse Kinematics

Solving IK is equivalent to finding the roots of $\vec{g}(\vec{q}) = \vec{\chi}_d - \vec{f}(\vec{q}) = \vec{0}$

Use of numerical technique analogous to Newton's method:



$$\vec{q}_{k+1} = \vec{q}_k - \left(\frac{\partial \vec{g}}{\partial \vec{q}}\Big|_{\vec{q}_k}\right)^+ \vec{g}(\vec{q}_k)$$

$$\vec{q}_{k+1} = \vec{q}_k + \left(\frac{\partial \vec{f}}{\partial \vec{q}}\Big|_{\vec{q}_k}\right)^+ \left(\vec{\chi}_d - \vec{f}(\vec{q}_k)\right)$$

for which we need an initial guess $\vec{q}_0$
and a stopping criterion $\|\vec{g}(\vec{q}_d)\|_2 \leq \varepsilon$

$$\vec{q}_{k+1} = \vec{q}_k + J_A^+(\vec{q}_k) \underbrace{\left(\vec{\chi}_d - \vec{f}(\vec{q}_k)\right)}_{\Delta\vec{\chi}_{\mathcal{I}\mathcal{E}}}$$

pose error ⚠

⚠ How to properly compute the pose error: $\Delta\vec{\chi}_{\mathcal{I}\mathcal{E}} = \begin{bmatrix} {}_I\vec{r}_{IE^*} - {}_I\vec{r}_{IE}(\vec{q}) \\ \underbrace{\Phi_{\mathcal{I}\mathcal{E}^*} \boxminus \Phi_{\mathcal{I}\mathcal{E}}(\vec{q})}_{{}_\mathcal{I}\varphi_{\mathcal{E}\mathcal{E}^*}} \end{bmatrix} \propto \begin{bmatrix} {}_\mathcal{I}\vec{v}_{IE} \\ {}_\mathcal{I}\vec{\omega}_{\mathcal{I}\mathcal{E}} \end{bmatrix}$ => use J_0 instead of J_A

$\boxminus$: Boxminus: shortest "path" between orientations (minus does not work in orientation space)

Given $C_{\mathcal{I}\mathcal{E}}, C_{\mathcal{I}\mathcal{E}^*}$:

$$\begin{aligned} {}_\mathcal{E}\vec{\varphi}_{\mathcal{E}\mathcal{E}^*} &= \mathrm{RotMat2RotVec}(C_{\mathcal{E}\mathcal{E}^*}) \\ {}_\mathcal{I}\vec{\varphi}_{\mathcal{E}\mathcal{E}^*} &= C_{\mathcal{I}\mathcal{E}}\, {}_\mathcal{E}\vec{\varphi}_{\mathcal{E}\mathcal{E}^*} = \mathrm{RotMat2RotVec}(C_{\mathcal{I}\mathcal{E}^*}\, C_{\mathcal{I}\mathcal{E}}^\top(\vec{q})) \end{aligned}$$

Robotic Systems Lab

Robot Dynamics Exercise Session 03

3) Kinematics Motion Control

Given a reference trajectory defined by $\vec{\chi}_d(t) = \begin{bmatrix} {}_{\mathcal{I}}\vec{r}_{IE^*} \\ \Phi_{\mathcal{I}\mathcal{E}^*} \end{bmatrix}$, $\vec{w}_d(t) = \begin{bmatrix} {}_{\mathcal{I}}\vec{v}_{IE^*} \\ {}_{\mathcal{I}}\vec{\omega}_{\mathcal{I}\mathcal{E}^*} \end{bmatrix}$

Control law $\dot{\vec{q}} = J^+ \underbrace{\begin{bmatrix} {}_{\mathcal{I}}\vec{v}_{IE} \\ {}_{\mathcal{I}}\vec{\omega}_{\mathcal{I}\mathcal{E}} \end{bmatrix}}_{\vec{w}(t)}$, where $\vec{w}(t) = \vec{w}_d(t) + \underbrace{K_p}_{>0} \Delta\vec{\chi}_{\mathcal{I}\mathcal{E}}$

we require feed forward reference if the desired pose is not constant

relates "changes": $\dot{(\,)} \sim \Delta$

Reason why this works: error dynamics $\Delta\dot{\vec{\chi}}_{\mathcal{I}\mathcal{E}} + K_p \Delta\vec{\chi}_{\mathcal{I}\mathcal{E}} = \vec{0}$

$\Delta\vec{\chi}_{\mathcal{I}\mathcal{E}} \rightarrow \vec{0}$

$\vec{\chi}_{\mathcal{I}\mathcal{E}}(t) \rightarrow \vec{\chi}_d(t)$

as $t \rightarrow \infty$