



Lecture «Robot Dynamics»: Summary

151-0851-00 V

| | | |
|-----------|-------|---|
| lecture: | HG F3 | Tuesday 10:15 – 12:00, every week |
| exercise: | HG F3 | Wednesday 8:15 – 10:00, according to schedule |

Marco Hutter, Roland Siegwart, and Thomas Stastny

| | | | | | |
|------------|---------------------|--|------------|-------------|--|
| 17.09.2019 | Intro and Outline | Course Introduction; Recapitulation Position, Linear Velocity | | | |
| 24.09.2019 | Kinematics 1 | Rotation and Angular Velocity; Rigid Body Formulation, Transformation | 25.09.2019 | Exercise 1a | Kinematics Modeling the ABB arm |
| 01.10.2019 | Kinematics 2 | Kinematics of Systems of Bodies; Jacobians | 02.10.2019 | Exercise 1a | Differential Kinematics of the ABB arm |
| 08.10.2019 | Kinematics 3 | Kinematic Control Methods: Inverse Differential Kinematics, Inverse Kinematics; Rotation Error; Multi-task Control | 09.10.2019 | Exercise 1b | Kinematic Control of the ABB Arm |
| 15.10.2019 | Dynamics L1 | Multi-body Dynamics | 16.10.2019 | Midterm 1 | Programming kinematics with matlab |
| 22.10.2019 | Dynamics L2 | Floating Base Dynamics | 23.10.2019 | Exercise 2a | Dynamic Modeling of the ABB Arm |
| 29.10.2019 | Dynamics L3 | Dynamic Model Based Control Methods | 30.10.2019 | Exercise 2b | Dynamic Control Methods Applied to the ABB arm |
| 05.11.2019 | Legged Robot | Dynamic Modeling of Legged Robots & Control | 06.11.2019 | Midterm 2 | Programming dynamics with matlab |
| 12.11.2019 | Case Studies 1 | Legged Robotics Case Study | 13.11.2019 | Exercise 3 | Legged robot |
| 19.11.2019 | Rotorcraft | Dynamic Modeling of Rotorcraft & Control | 20.11.2019 | | |
| 26.11.2019 | Case Studies 2 | Rotor Craft Case Study | 27.11.2019 | Exercise 4 | Modeling and Control of Multicopter |
| 03.12.2019 | Fixed-wing | Dynamic Modeling of Fixed-wing & Control | 04.12.2019 | | |
| 10.12.2019 | Case Studies 3 | Fixed-wing Case Study (Solar-powered UAVs - AtlantikSolar, Vertical Take-off and Landing UAVs – Wingtra) | 11.12.2019 | Exercise 5 | Fixed-wing Control and Simulation |
| 17.12.2019 | Summery and Outlook | Summery; Wrap-up; Exam | | | Robot Dynamics - Summary 17.12.2019 2 |

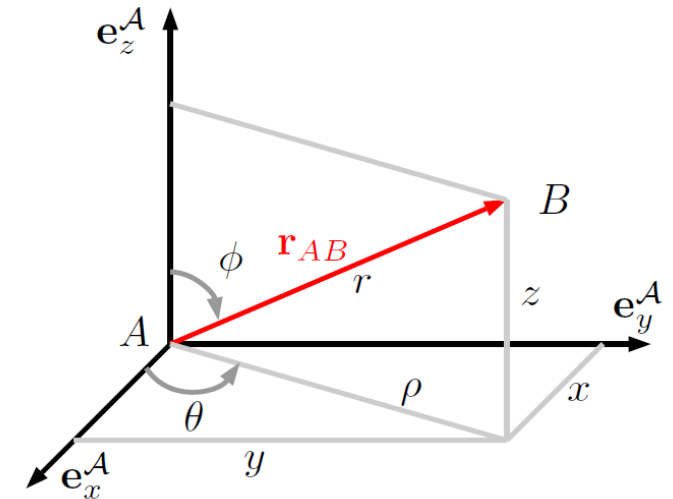
Position

- Position ${}^A\mathbf{r}_{AB} \in \mathbb{R}^3$, reference frames A

$${}^A\mathbf{r}_{AP} = {}^A\mathbf{r}_{AB} + {}^A\mathbf{r}_{BP}$$

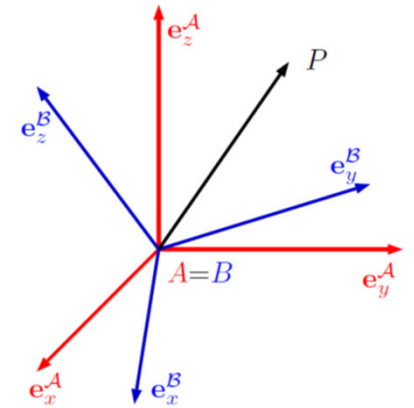
- Different parameterizations, e.g.

- Cartesian coordinates $\chi_{Pc} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$
 - Position vector ${}^A\mathbf{r} = x\mathbf{e}_x^A + y\mathbf{e}_y^A + z\mathbf{e}_z^A = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$
- Cylindrical coordinates $\chi_{Pz} = \begin{pmatrix} \rho \\ \theta \\ z \end{pmatrix}$
 - Position vector ${}^A\mathbf{r} = \begin{pmatrix} \rho \cos \theta \\ \rho \sin \theta \\ z \end{pmatrix}$



Rotation

- Rotation $\phi_{AB} \in SO(3)$
- Rotation matrix ${}^A\mathbf{r}_{AP} = [{}^A\mathbf{e}_x^B \quad {}^A\mathbf{e}_y^B \quad {}^A\mathbf{e}_z^B] \cdot {}^B\mathbf{r}_{AP} = \mathbf{C}_{AB} \cdot {}^B\mathbf{r}_{AP}$
- Different parameterizations, e.g.



- Euler angles $\chi_{R,eulerZYZ} = \begin{pmatrix} z_1 \\ y \\ z_2 \end{pmatrix}$
- Quaternions $\chi_{R,quat} = \xi = \begin{pmatrix} \xi_0 \\ \check{\xi} \end{pmatrix} \in \mathbb{H} \quad \check{\xi} = \begin{pmatrix} \cos(\theta/2) \\ \sin(\theta/2)\mathbf{n} \end{pmatrix}$

Introduction to algebra with quaternions, e.g.

$$\xi_{AB} \otimes \xi_{BC} \longleftrightarrow \mathbf{C}_{AB} \mathbf{C}_{BC}$$

...

- Relation to rotation matrix:

$$\begin{aligned} \mathbf{C}_{AB} &= \mathbb{I}_{3 \times 3} + 2\xi_0 [\check{\xi}]_{\times} + 2[\check{\xi}]_{\times}^2 = (2\xi_0^2 - 1) \mathbb{I}_{3 \times 3} + 2\xi_0 [\check{\xi}]_{\times} + 2\check{\xi}\check{\xi}^T \\ &= \begin{bmatrix} \xi_0^2 + \xi_1^2 - \xi_2^2 - \xi_3^2 & 2\xi_1\xi_2 - 2\xi_0\xi_3 & 2\xi_0\xi_2 + 2\xi_1\xi_3 \\ 2\xi_0\xi_3 + 2\xi_1\xi_2 & \xi_0^2 - \xi_1^2 + \xi_2^2 - \xi_3^2 & 2\xi_2\xi_3 - 2\xi_0\xi_1 \\ 2\xi_1\xi_3 - 2\xi_0\xi_2 & 2\xi_0\xi_1 + 2\xi_2\xi_3 & \xi_0^2 - \xi_1^2 - \xi_2^2 + \xi_3^2 \end{bmatrix}. \end{aligned}$$

Velocity

- Linear velocity $\dot{\mathbf{r}}_{AB}$
 - Representation $\dot{\chi}_P$

$$\begin{aligned} \dot{\mathbf{r}} &= \mathbf{E}_P(\chi_P) \dot{\chi}_P \\ \dot{\chi}_P &= \mathbf{E}_P^{-1}(\chi_P) \dot{\mathbf{r}} \end{aligned}$$

e.g. cylindrical coordinates $\mathbf{E}_{Pz}(\chi_{Pz}) = \begin{bmatrix} \cos \theta & -\rho \sin \theta & 0 \\ \sin \theta & \rho \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$
- Angular velocity $[_{\mathcal{A}}\omega_{\mathcal{AB}}]_{\times} = \dot{\mathbf{C}}_{\mathcal{AB}} \cdot \mathbf{C}_{\mathcal{AB}}^T$ $[_{\mathcal{A}}\omega_{\mathcal{AB}}]_{\times} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}, \quad {}_{\mathcal{A}}\omega_{\mathcal{AB}} = \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}$
 - Representation ${}_{\mathcal{A}}\omega_{\mathcal{AB}} = \mathbf{E}_R(\chi_R) \cdot \dot{\chi}_R$
 - e.g. quaternions
$$\begin{aligned} \mathbf{E}_{R,quat} &= 2\mathbf{H}(\xi), \\ \mathbf{E}_{R,quat}^{-1} &= \frac{1}{2}\mathbf{H}(\xi)^T \end{aligned}$$

$$\begin{aligned} \mathbf{H}(\xi) &= \begin{bmatrix} -\check{\xi} & [\check{\xi}]_{\times} + \xi_0 \mathbb{I}_{3 \times 3} \end{bmatrix} \in \mathbb{R}^{3 \times 4} \\ &= \begin{bmatrix} -\xi_1 & \xi_0 & -\xi_3 & \xi_2 \\ -\xi_2 & \xi_3 & \xi_0 & -\xi_1 \\ -\xi_3 & -\xi_2 & \xi_1 & \xi_0 \end{bmatrix}. \end{aligned}$$

Kinematics of Systems of Bodies

- Generalized coordinates $\mathbf{q} = \begin{pmatrix} q_1 \\ \vdots \\ q_n \end{pmatrix}$
- End-effector position and orientation $\mathbf{x}_e = \begin{pmatrix} \mathbf{r}_e \\ \phi_e \end{pmatrix} \in SE(3)$ parameterized by $\chi_e = \begin{pmatrix} \chi_{eP} \\ \chi_{eR} \end{pmatrix} = \begin{pmatrix} \chi_1 \\ \vdots \\ \chi_m \end{pmatrix} \in \mathbb{R}^m$
- Forward kinematics $\chi_e = \chi_e(\mathbf{q})$
- Forward differential kinematics

Analytic $\delta \chi_e \approx \frac{\partial \chi_e(\mathbf{q})}{\partial \mathbf{q}} \delta \mathbf{q} = \mathbf{J}_{eA}(\mathbf{q}) \delta \mathbf{q}$ with $\mathbf{J}_{eA} = \frac{\partial \chi_e}{\partial \mathbf{q}} = \begin{bmatrix} \frac{\partial \chi_1}{\partial q_1} & \dots & \frac{\partial \chi_1}{\partial q_{n_j}} \\ \vdots & \ddots & \vdots \\ \frac{\partial \chi_m}{\partial q_1} & \dots & \frac{\partial \chi_m}{\partial q_{n_j}} \end{bmatrix}$

$$\dot{\chi}_e = \mathbf{J}_{eA}(\mathbf{q}) \dot{\mathbf{q}} \quad \text{with } \mathbf{J}_{eA}(\mathbf{q}) \in \mathbb{R}^{m_e \times n_j}$$

Depending on
parameterization!!

- Geometric

$$\mathbf{w}_e = \begin{pmatrix} \mathbf{v}_e \\ \boldsymbol{\omega}_e \end{pmatrix} = \mathbf{J}_{e0}(\mathbf{q}) \dot{\mathbf{q}} \quad \text{with } \mathbf{J}_{e0}(\mathbf{q}) \in \mathbb{R}^{6 \times n_j}$$

Independent of
parameterization

$$\mathbf{w}_e = \mathbf{E}_e(\chi_e) \dot{\chi}_e$$



$$\mathbf{J}_{e0}(\mathbf{q}) = \mathbf{E}_e(\chi) \mathbf{J}_{eA}(\mathbf{q})$$

Geometric Jacobian Derivation

n : rotation axis

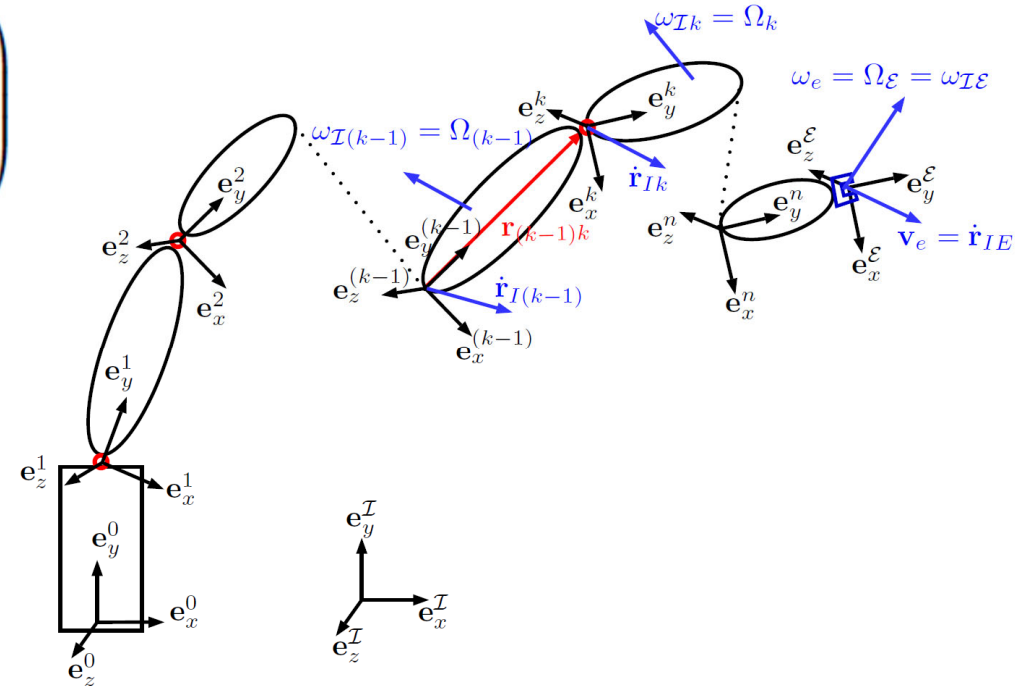
r : translational vector between two adjacent coordinate

Linear velocity

$$\dot{\mathbf{r}}_{IE} = \underbrace{\begin{bmatrix} \mathbf{n}_1 \times \mathbf{r}_{1(n+1)} & \mathbf{n}_2 \times \mathbf{r}_{2(n+1)} & \dots & \mathbf{n}_n \times \mathbf{r}_{n(n+1)} \end{bmatrix}}_{\mathbf{J}_{e0P}} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{pmatrix}$$

Angular velocity

$$\omega_{IE} = \sum_{i=1}^n \mathbf{n}_i \dot{q}_i = \underbrace{\begin{bmatrix} \mathbf{n}_1 & \mathbf{n}_2 & \dots & \mathbf{n}_n \end{bmatrix}}_{\mathbf{J}_{e0R}} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{pmatrix}$$



Analytical and Kinematic Jacobian

- Analytical Jacobian

$$\dot{\chi}_e = \mathbf{J}_{eA}(\mathbf{q}) \dot{\mathbf{q}}$$

$$\mathbf{J}_{e0}(\mathbf{q}) = \mathbf{E}_e(\chi) \mathbf{J}_{eA}(\mathbf{q})$$

- Relates **time-derivatives of config. parameters** to generalized velocities
- Depending on selected parameterization (mainly rotation) in 3D $\Delta\chi \Leftrightarrow \Delta\mathbf{q}$
Note: there exist no "rotation angle"
- Mainly used for numeric algorithms

- Geometric (or basic) Jacobian

$$\mathbf{w}_e = \begin{pmatrix} \mathbf{v}_e \\ \boldsymbol{\omega}_e \end{pmatrix} = \mathbf{J}_{e0}(\mathbf{q}) \dot{\mathbf{q}}$$

- Relates **end-effector velocity** to generalized velocities
- Unique** for every robot
- Used in **most** cases

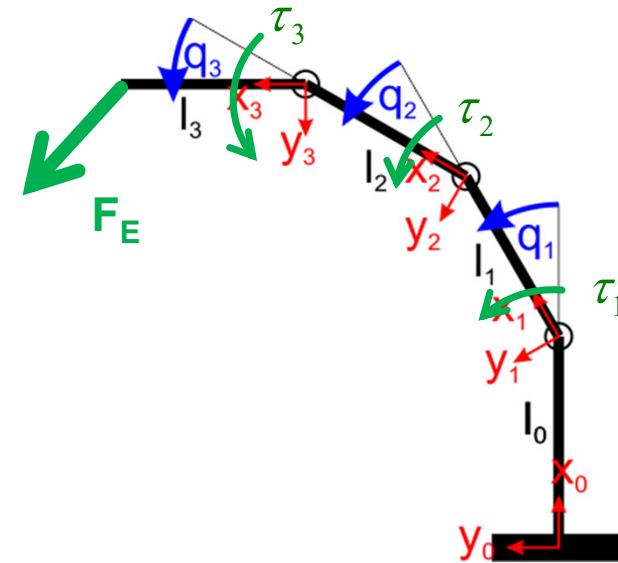
Importance of Jacobian

- Kinematics (mapping of changes from joint to task space)
 - Inverse kinematics control
 - Resolve redundancy problems
 - Express contact constraints
- Statics (and later also dynamics)
 - Principle of virtual work
 - Variations in work must cancel for all virtual displacement
 - Internal forces of ideal joint don't contribute

$$\begin{aligned}\delta W &= \sum_i \mathbf{f}_i \mathbf{x}_i = \boldsymbol{\tau}^T \delta \mathbf{q} + (-\mathbf{F}_E)^T \delta \mathbf{x}_E \\ &= \boldsymbol{\tau}^T \delta \mathbf{q} + (-\mathbf{F}_E)^T \mathbf{J} \delta \mathbf{q} = 0 \quad \forall \delta \mathbf{q}\end{aligned}$$

➤ Dual problem from principle of virtual work

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{J} \dot{\mathbf{q}} \\ \boldsymbol{\tau} &= \mathbf{J}^T \mathbf{F}\end{aligned}$$



Inverse Differential Kinematics

- Differential kinematics $\mathbf{w}_e = \mathbf{J}_{e0} \dot{\mathbf{q}}$
- Inverse differential kinematics $\dot{\mathbf{q}} = \mathbf{J}_{e0}^+ \mathbf{w}_e^*$.
 - Singularity: minimizing $\|\mathbf{w}_e^* - \mathbf{J}_{e0} \dot{\mathbf{q}}\|^2$
 - Redundancy: $\dot{\mathbf{q}} = \mathbf{J}_{e0}^+ \mathbf{w}_e^* + \mathbf{N} \dot{\mathbf{q}}_0$ null-space projection matrix $\mathbf{N} = \mathcal{N}(\mathbf{J}_{e0})$
- Multi-task control: $task_i := \{\mathbf{J}_i, \mathbf{w}_i^*\}$
 - Equal priority

$$\dot{\mathbf{q}} = \underbrace{\begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_{n_t} \end{bmatrix}}_{\bar{\mathbf{J}}}^+ \underbrace{\begin{pmatrix} \mathbf{w}_1^* \\ \vdots \\ \mathbf{w}_{n_t}^* \end{pmatrix}}_{\bar{\mathbf{w}}}$$

Multi-task with prioritization

$$\dot{\mathbf{q}} = \sum_{i=1}^{n_t} \mathbf{N}_i \dot{\mathbf{q}}_i, \quad \text{with} \quad \dot{\mathbf{q}}_i = (\mathbf{J}_i \mathbf{N}_i)^+ \left(\mathbf{w}_i^* - \mathbf{J} \sum_{k=1}^{i-1} \mathbf{N}_k \dot{\mathbf{q}}_k \right)$$

Inverse Kinematics

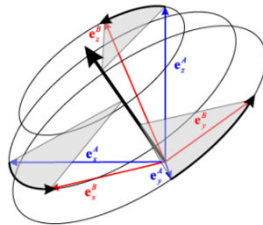
- Numerical approach $\Delta \chi_e = \mathbf{J}_{eA} \Delta \mathbf{q}$

Algorithm 1 Numerical Inverse Kinematics

| | |
|---|--|
| 1: $\mathbf{q} \leftarrow \mathbf{q}^0$ | ▷ Start configuration |
| 2: while $\ \chi_e^* - \chi_e(\mathbf{q})\ > tol$ do | ▷ While the solution is not reached |
| 3: $\mathbf{J}_{eA} \leftarrow \mathbf{J}_{eA}(\mathbf{q}) = \frac{\partial \chi_e}{\partial \mathbf{q}}(\mathbf{q})$ | ▷ Evaluate Jacobian for \mathbf{q} |
| 4: $\mathbf{J}_{eA}^+ \leftarrow (\mathbf{J}_{eA})^+$ | ▷ Calculate the pseudo inverse |
| 5: $\Delta \chi_e \leftarrow \chi_e^* - \chi_e(\mathbf{q})$ | ▷ Find the end-effector configuration error vector |
| 6: $\mathbf{q} \leftarrow \mathbf{q} + \mathbf{J}_{eA}^+ \Delta \chi_e$ | ▷ Update the generalized coordinates |
| 7: end while | |

Position/Rotation Errors and Trajectory Control

- Position error $\Delta \mathbf{r}_e^t = \mathbf{r}_e^*(t) - \mathbf{r}_e(\mathbf{q}^t)$
 - Trajectory control with position-error feedback $\dot{\mathbf{q}} = \mathbf{J}_{e0_P}^+ (\dot{\mathbf{r}}^* + k_{PP} \Delta \mathbf{r}_e^t)$
- Rotation error $\Delta \varphi$ is not $\varphi^* - \varphi^t \implies \mathbf{C}_{\mathcal{GS}}(\Delta \varphi) = \mathbf{C}_{\mathcal{GI}}(\varphi^*) \mathbf{C}_{\mathcal{SI}}^T(\varphi^t)$
 - Trajectory control with rotation-error feedback $\dot{\mathbf{q}} = \mathbf{J}_{e0_R}^+ (\omega(t)_e^* + k_{PR} \Delta \varphi)$



Floating Base Kinematics

- Describe system by **base** and **joint** coordinates $\mathbf{q} = \begin{pmatrix} \mathbf{q}_b \\ \mathbf{q}_j \end{pmatrix}$
- Base coordinates: rotation and position of base $\mathbf{q}_b = \begin{pmatrix} \mathbf{q}_{b_P} \\ \mathbf{q}_{b_R} \end{pmatrix} \in \mathbb{R}^3 \times SO(3)$
- Contact constraints: $\mathcal{I}\mathbf{r}_{IC_i} = \text{const}, \quad \mathcal{I}\dot{\mathbf{r}}_{IC_i} = \mathcal{I}\ddot{\mathbf{r}}_{IC_i} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$

Multi-body Dynamics

- We learned how to get the equation of motion in joint space
 - Newton-Euler
 - Projected Newton-Euler
 - Lagrange II

- Started from the principle for virtual work

$$\delta W = \int_{\mathcal{B}} \delta \mathbf{r}^T \cdot (\ddot{\mathbf{r}} dm - d\mathbf{F}_{ext}) = 0.$$

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + \mathbf{J}_c^T \mathbf{F}_c = \boldsymbol{\tau}$$

| | |
|--|---------------------------------|
| $\ddot{\mathbf{q}}$ | Generalized accelerations |
| $\mathbf{M}(\mathbf{q})$ | Mass matrix |
| $\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})$ | Centrifugal and Coriolis forces |
| $\mathbf{g}(\mathbf{q})$ | Gravity forces |
| $\boldsymbol{\tau}$ | Generalized forces |
| \mathbf{F}_c | External forces |
| \mathbf{J}_c | Contact Jacobian |

| | |
|---------------------|---------------------------------------|
| $d\mathbf{F}_{ext}$ | external forces acting on element i |
| $\ddot{\mathbf{r}}$ | acceleration of element i |
| dm | mass of element i |
| $\delta \mathbf{r}$ | virtual displacement of element i |

Impulse and angular momentum

- Use the following definitions

$$\mathbf{p}_S = m\mathbf{v}_S$$

linear momentum

$$\mathbf{N}_S = \mathbf{\Theta}_S \mathbf{\Omega}_S$$

angular momentum

$$\dot{\mathbf{p}}_S = m\mathbf{a}_S$$

change in linear momentum

$$\dot{\mathbf{N}}_S = \mathbf{\Theta}_S \mathbf{\Psi} + \mathbf{\Omega} \times \mathbf{\Theta}_S \mathbf{\Omega}$$

change in angular momentum

- Conservation of impulse and angular momentum

$$0 = \delta W = \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \mathbf{\Phi} \end{pmatrix}^T \left(\begin{array}{c|c} \boxed{\dot{\mathbf{p}}_S} & \boxed{\mathbf{F}_{ext}} \\ \hline \boxed{\dot{\mathbf{N}}_S} & \boxed{\mathbf{T}_{ext}} \end{array} \right) \quad \forall \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \mathbf{\Phi} \end{pmatrix}$$

Newton A free body can move
In all directions

Euler External forces and moments

Change in impulse and angular momentum

$$\begin{aligned} \dot{\mathbf{p}}_S &= \mathbf{F}_{ext} \\ \dot{\mathbf{N}}_S &= \mathbf{T}_{ext} \end{aligned}$$

Projected Newton Euler

- Consider only directions the system can move (c.f. generalized coordinates)

$$0 = \delta W = \delta \mathbf{q}^T \sum_{i=1}^{n_b} \underbrace{\begin{pmatrix} \mathbf{J}_{S_i} \\ \mathbf{J}_{R_i} \end{pmatrix}^T \begin{pmatrix} m \mathbf{J}_{S_i} \\ \boldsymbol{\Theta}_{S_i} \mathbf{J}_{R_i} \end{pmatrix}}_{\mathbf{M}(\mathbf{q})} \ddot{\mathbf{q}} + \underbrace{\begin{pmatrix} \mathbf{J}_{S_i} \\ \mathbf{J}_{R_i} \end{pmatrix}^T \begin{pmatrix} m \dot{\mathbf{J}}_{S_i} \dot{\mathbf{q}} \\ \boldsymbol{\Theta}_{S_i} \dot{\mathbf{J}}_{R_i} \dot{\mathbf{q}} + \mathbf{J}_{R_i} \dot{\mathbf{q}} \times \boldsymbol{\Theta}_{S_i} \mathbf{J}_{R_i} \dot{\mathbf{q}} \end{pmatrix}}_{\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})} - \underbrace{\begin{pmatrix} \mathbf{J}_{P_i} \\ \mathbf{J}_{R_i} \end{pmatrix}^T \begin{pmatrix} \mathbf{F}_{ext,i} \\ \mathbf{T}_{ext,i} \end{pmatrix}}_{\mathbf{g}(\mathbf{q})} \quad \boxed{\forall \delta \mathbf{q}}$$

- Resulting in

$$\begin{aligned} \mathbf{M} &= \sum_{i=1}^{n_b} \left(\mathcal{A} \mathbf{J}_{S_i}^T \cdot m \cdot \mathcal{A} \mathbf{J}_{S_i} + \mathcal{B} \mathbf{J}_{R_i}^T \cdot \mathcal{B} \boldsymbol{\Theta}_{S_i} \cdot \mathcal{B} \mathbf{J}_{R_i} \right) \\ \mathbf{b} &= \sum_{i=1}^{n_b} \left(\mathcal{A} \mathbf{J}_{S_i}^T \cdot m \cdot \mathcal{A} \dot{\mathbf{J}}_{S_i} \cdot \dot{\mathbf{q}} + \mathcal{B} \mathbf{J}_{R_i}^T \cdot \left(\mathcal{B} \boldsymbol{\Theta}_{S_i} \cdot \mathcal{B} \dot{\mathbf{J}}_{R_i} \cdot \dot{\mathbf{q}} + \mathcal{B} \boldsymbol{\Omega}_{S_i} \times \mathcal{B} \boldsymbol{\Theta}_{S_i} \cdot \mathcal{B} \boldsymbol{\Omega}_{S_i} \right) \right) \\ \mathbf{g} &= \sum_{i=1}^{n_b} \left(-\mathcal{A} \mathbf{J}_{S_i}^T \mathcal{A} \mathbf{F}_{g,i} \right) \end{aligned}$$

Lagrange II

- Get equation of motion from $\frac{d}{dt} \left(\frac{\partial \mathcal{T}}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial \mathcal{T}}{\partial \mathbf{q}} + \frac{\partial \mathcal{U}}{\partial \dot{\mathbf{q}}} = \boldsymbol{\tau}$

- Kinetic energy
$$\mathcal{T} = \sum_{i=1}^{n_b} \left(\frac{1}{2} m_i \mathbf{A} \dot{\mathbf{r}}_{S_i}^T \mathbf{A} \dot{\mathbf{r}}_{S_i} + \frac{1}{2} \mathbf{B} \boldsymbol{\Omega}_{S_i}^T \cdot \mathbf{B} \boldsymbol{\Theta}_{S_i} \cdot \mathbf{B} \boldsymbol{\Omega}_{S_i} \right)$$

- Potential energy
$$\mathbf{F}_{g_i} = m_i g I \mathbf{e}_g$$

$$\mathcal{U}_g = - \sum_{i=1}^{n_b} \mathbf{r}_{S_i}^T \mathbf{F}_{g_i}$$

$$\mathcal{U}_{E_j} = \frac{1}{2} k_j (d(\mathbf{q}) - d_0)^2$$

External Forces

Cart pendulum example

- Equation of motion without actuation

$$\underbrace{\begin{bmatrix} m_c + m_p & lm_p \cos(\varphi) \\ lm_p \cos(\varphi) & m_p l^2 + \theta_p \end{bmatrix}}_{\mathbf{M}} \ddot{\mathbf{q}} + \underbrace{\begin{pmatrix} -\dot{\varphi}^2 lm_p \sin(\varphi) \\ 0 \end{pmatrix}}_{\mathbf{b}} + \underbrace{\begin{pmatrix} 0 \\ m_p gl \sin(\varphi) \end{pmatrix}}_{\mathbf{g}} = \mathbf{0}$$

- Add actuator for the pendulum

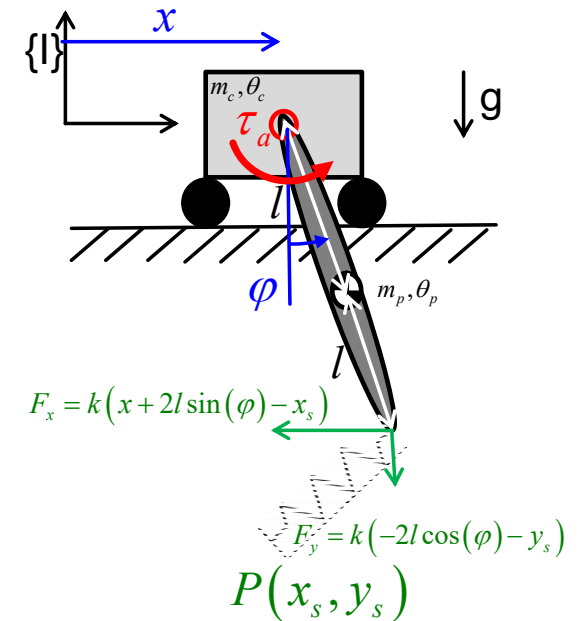
- Action on pendulum $T_p = \tau_a$
- Reaction on cart $T_c = -\tau_a$

$$\left. \begin{array}{l} \mathbf{J}_{Rp} = \begin{bmatrix} 0 & 1 \end{bmatrix} \\ \mathbf{J}_{Rc} = \begin{bmatrix} 0 & 0 \end{bmatrix} \end{array} \right\} \boldsymbol{\tau} = \sum \mathbf{J}_{R,i}^T \mathbf{T}_i = \mathbf{J}_{Rc}^T \mathbf{T}_c + \mathbf{J}_{Rp}^T \mathbf{T}_p = \begin{pmatrix} 0 \\ \tau_a \end{pmatrix}$$

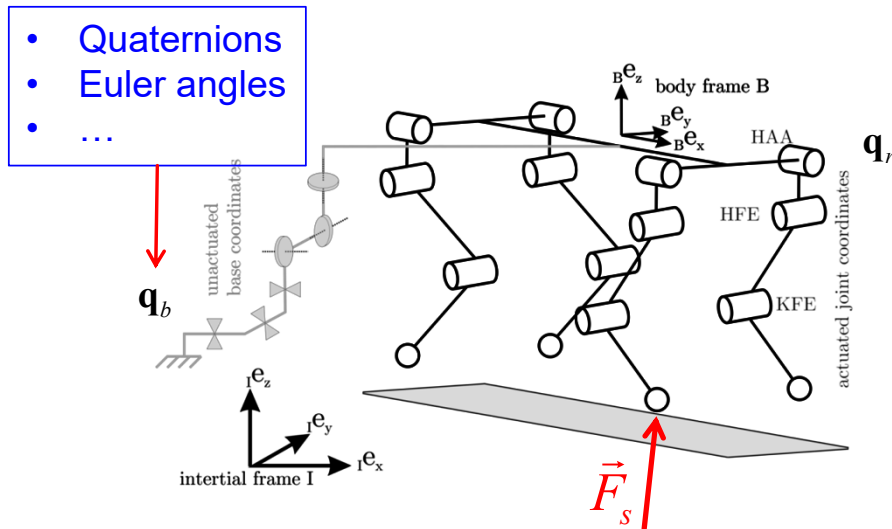
- Add spring to the pendulum

- (world attachment point P, zero length 0, stiffness k)
- Action on pendulum

$$\mathbf{F}_s = \begin{pmatrix} -F_x \\ -F_y \end{pmatrix} \quad \left. \begin{array}{l} \mathbf{r} = \begin{pmatrix} x + 2l \sin(\varphi) \\ -2l \cos(\varphi) \end{pmatrix} \\ \mathbf{J}_s = \frac{\partial \mathbf{r}_s}{\partial \mathbf{q}} = \begin{bmatrix} 1 & 2l \cos(\varphi) \\ 0 & 2l \sin(\varphi) \end{bmatrix} \end{array} \right\} \boldsymbol{\tau} = \mathbf{J}^T \mathbf{F}_s = \begin{pmatrix} -F_x \\ -2l(F_x \cos(\varphi) + F_y \sin(\varphi)) \end{pmatrix}$$



Dynamics of Floating Base Systems



$$\mathbf{q} = \begin{pmatrix} \mathbf{q}_b \\ \mathbf{q}_j \end{pmatrix} \quad \begin{array}{l} \text{Un-actuated base} \\ \text{Actuated joints} \end{array}$$

- EoM from last time

$$\mathbf{M}\ddot{\mathbf{q}}_j + \mathbf{b} + \mathbf{g} = \boldsymbol{\tau}$$

- Not all joint are actuated

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g} = \mathbf{S}^T \boldsymbol{\tau}$$

- Selection matrix of actuated joints

$$\mathbf{S} = \begin{bmatrix} \mathbf{0}_{n \times 6} & \mathbf{I}_{n \times n} \end{bmatrix} \quad \mathbf{q}_j = \mathbf{S}\mathbf{q}$$

- Contact force acting on system

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g} = \mathbf{S}^T \boldsymbol{\tau} + \mathbf{J}_s^T \mathbf{F}_s, \text{ acting on system}$$

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g} + \mathbf{J}_s^T \mathbf{F}_s, \text{ exerted by robot} = \mathbf{S}^T \boldsymbol{\tau}$$

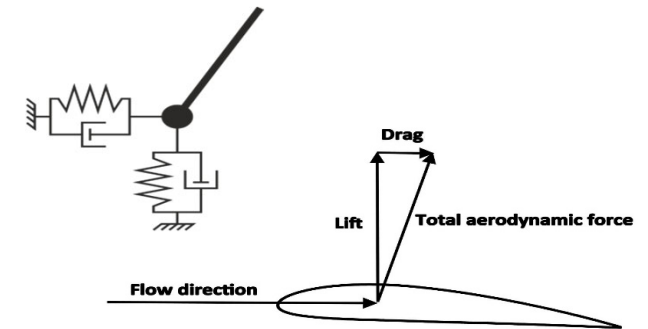
| | |
|---------------|------------------------------------|
| Manipulator: | interaction forces at end-effector |
| Legged robot: | ground contact forces |
| UAV: | lift force |

Note: for simplicity we don't use here \mathbf{u} but only time derivatives of \mathbf{q}

External Forces

Some notes

- External forces from force elements or actuator
 - E.g. soft contact $\mathbf{F}_s = k_p (\mathbf{r}_{des} - \mathbf{r}) + k_d (-\dot{\mathbf{r}})$
 - Aerodynamics $F_s = \frac{1}{2} \rho c_v A c_L$
- External forces from constraints...



Support Consistent Dynamics

- Equation of motion $\mathbf{M}\ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g} + \mathbf{J}_s^T \mathbf{F}_s = \mathbf{S}^T \boldsymbol{\tau} \quad (1)$
 - Cannot directly be used for control due to the occurrence of contact forces

- Contact constraint $\mathbf{r}_s = \mathbf{J}_s \ddot{\mathbf{q}} + \dot{\mathbf{J}}_s \dot{\mathbf{q}} = \mathbf{0}$

- Contact force $\mathbf{F}_s = (\mathbf{J}_s \mathbf{M}^{-1} \mathbf{J}_s^T)^{-1} (\mathbf{J}_s \mathbf{M}^{-1} (\mathbf{S}^T \boldsymbol{\tau} - (\mathbf{b} + \mathbf{g})) + \dot{\mathbf{J}}_s \dot{\mathbf{q}})$
 - Back-substitute in (1),
replace $\dot{\mathbf{J}}_s \dot{\mathbf{q}} = -\mathbf{J}_s \ddot{\mathbf{q}}$ and use
support null-space projection

$$\mathbf{N}_s = \mathbf{I} - \mathbf{M}^{-1} \mathbf{J}_s^T (\mathbf{J}_s \mathbf{M}^{-1} \mathbf{J}_s^T)^{-1} \mathbf{J}_s \quad \boxed{\mathbf{J}_s \mathbf{N}_s = \mathbf{0}}$$

- Support consistent dynamics

$$\boxed{\mathbf{N}_s^T \mathbf{M} \ddot{\mathbf{q}} + \mathbf{N}_s^T (\mathbf{b} + \mathbf{g}) = \mathbf{N}_s^T \mathbf{S}^T \boldsymbol{\tau}}$$

Dynamic Control Methods

- Joint impedance control (w/o and w/ gravity compensation)
- Inverse dynamics control
- Generalized motion and force control

Joint Impedance Control

$$M(q) \ddot{q} + b(q, \dot{q}) + g(q) = \tau$$

- Torque as function of position and velocity error $\tau^* = k_p (q^* - q) + k_d (\dot{q}^* - \dot{q})$

- Closed loop behavior

~~$$M(q) \ddot{q} + b(q, \dot{q}) + g(q) = \tau = k_p (q^* - q) + k_d (\dot{q}^* - \dot{q})$$~~

- Static offset due to gravity

- Impedance control and gravity compensation

$$\tau^* = k_p (q^* - q) + k_d (\dot{q}^* - \dot{q}) + \hat{g}(q)$$

Estimated gravity term

Simple setup...
but configuration dependent load



Inverse Dynamics Control

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}$$

- Compensate for system dynamics $\boldsymbol{\tau} = \hat{\mathbf{M}}(\mathbf{q}) \ddot{\mathbf{q}}^* + \hat{\mathbf{b}}(\mathbf{q}, \dot{\mathbf{q}}) + \hat{\mathbf{g}}(\mathbf{q})$
- In case of no modeling errors,
 - the desired dynamics can be perfectly prescribed $\mathbb{I} \ddot{\mathbf{q}} = \ddot{\mathbf{q}}^*$
- PD-control law $\mathbb{I} \ddot{\mathbf{q}} = \ddot{\mathbf{q}}^* = \mathbf{k}_p (\mathbf{q}^* - \mathbf{q}) + \mathbf{k}_d (\dot{\mathbf{q}}^* - \dot{\mathbf{q}})$

Can achieve great performance...
but requires accurate modeling

Operational Space Control

Generalized framework to control motion and force

- Joint-space dynamics

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}$$

- End-effector dynamics

$$\Lambda \dot{\mathbf{w}}_e + \boldsymbol{\mu} + \mathbf{p} = \mathbf{F}_e$$

$$\boldsymbol{\tau} = \mathbf{J}_e^T \mathbf{F}_e$$

$$\Lambda = (\mathbf{J}_e \mathbf{M}^{-1} \mathbf{J}_e^T)^{-1}$$

$$\boldsymbol{\mu} = \Lambda \mathbf{J}_e \mathbf{M}^{-1} \mathbf{b} - \Lambda \dot{\mathbf{J}}_e \dot{\mathbf{q}}$$

$$\mathbf{p} = \Lambda \mathbf{J}_e \mathbf{M}^{-1} \mathbf{g}$$

- Determine the corresponding joint torque

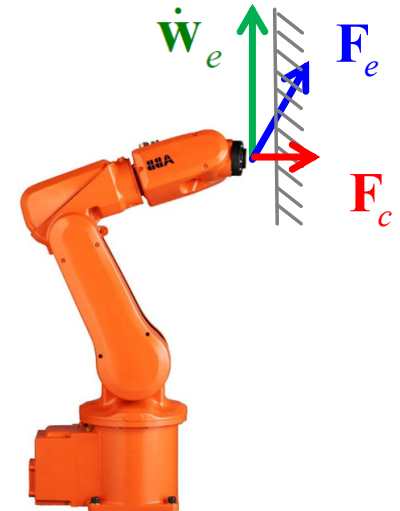
$$\boldsymbol{\tau}^* = \hat{\mathbf{J}}^T \left(\hat{\Lambda}_e \dot{\mathbf{w}}_e^* + \hat{\boldsymbol{\mu}} + \hat{\mathbf{p}} \right)$$

- Extend end-effector dynamics in contact with contact force

$$\mathbf{F}_c + \Lambda \dot{\mathbf{w}}_e + \boldsymbol{\mu} + \mathbf{p} = \mathbf{F}_e$$

- Introduce selection matrices to separate motion force directions

$$\boldsymbol{\tau}^* = \hat{\mathbf{J}}^T \left(\hat{\Lambda} \mathbf{S}_M \dot{\mathbf{w}}_e + \mathbf{S}_F \mathbf{F}_c + \hat{\boldsymbol{\mu}} + \hat{\mathbf{p}} \right)$$



Inverse Dynamics of Floating Base Systems

- Equation of motion of floating base systems

$$\mathbf{M}(\mathbf{q}) \dot{\mathbf{u}} + \mathbf{b}(\mathbf{q}, \mathbf{u}) + \mathbf{g}(\mathbf{q}) + \mathbf{J}_c^T \mathbf{F}_c = \mathbf{S}^T \boldsymbol{\tau}$$

- Support-consistent

$$\mathbf{N}_c^T (\mathbf{M} \ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g}) = \mathbf{N}_c^T \mathbf{S}^T \boldsymbol{\tau}$$

- Inverse-dynamics

$$\boldsymbol{\tau}^* = (\mathbf{N}_c^T \mathbf{S}^T)^+ \mathbf{N}_c^T (\mathbf{M} \ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g})$$

- Multiple solutions

$$\boldsymbol{\tau}^* = (\mathbf{N}_c^T \mathbf{S}^T)^+ \mathbf{N}_c^T (\mathbf{M} \ddot{\mathbf{q}}^* + \mathbf{b} + \mathbf{g}) + \mathcal{N}(\mathbf{N}_c^T \mathbf{S}^T) \boldsymbol{\tau}_0^*$$

Operational Space Control as Quadratic Program

A general problem

$$\min_{\mathbf{x}} \quad \|\mathbf{A}_i \mathbf{x} - \mathbf{b}_i\|_2 \quad \mathbf{x} = \begin{pmatrix} \dot{\mathbf{u}} \\ \mathbf{F}_c \\ \boldsymbol{\tau} \end{pmatrix}$$

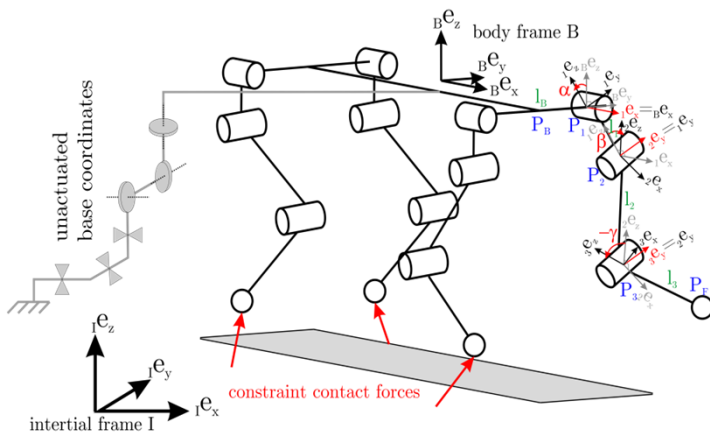
- We search for a solution that fulfills the equation of motion

$$\mathbf{M}(\mathbf{q}) \dot{\mathbf{u}} + \mathbf{b}(\mathbf{q}, \mathbf{u}) + \mathbf{g}(\mathbf{q}) + \mathbf{J}_c^T \mathbf{F}_c = \mathbf{S}^T \boldsymbol{\tau} \quad \Rightarrow \quad \mathbf{A} = \begin{bmatrix} \hat{\mathbf{M}} & \hat{\mathbf{J}}_c^T & -\mathbf{S}^T \end{bmatrix} \quad \mathbf{b} = -\hat{\mathbf{b}} - \hat{\mathbf{g}}$$

- Motion tasks: $\mathbf{J} \dot{\mathbf{u}} + \dot{\mathbf{J}} \mathbf{u} = \dot{\mathbf{w}}^*$ $\Rightarrow \mathbf{A} = \begin{bmatrix} \hat{\mathbf{J}}_i & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad \mathbf{b} = \dot{\mathbf{w}}^* - \dot{\mathbf{J}}_i \mathbf{u}$
- Force tasks: $\mathbf{F}_i = \mathbf{F}_i^*$ $\Rightarrow \mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbb{I} & \mathbf{0} \end{bmatrix} \quad \mathbf{b} = \mathbf{F}_i^*$
- Torque min: $\min \|\boldsymbol{\tau}\|_2$ $\Rightarrow \mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbb{I} \end{bmatrix} \quad \mathbf{b} = \mathbf{0}$

Kinematics of Floating Base / Mobile Systems

example



- Quadrupedal robot
- Static walking
- 3 legs in stance [NR 1,2,3]
- 1 in swing [NR 4]

1. How many generalized coordinates?
2. How many base coordinates?
3. How many actuated joint coordinates?
4. How many contact constraints?
5. Write down the contact constraint
6. How many DoFs remain adjustable?
7. Which DoFs remain adjustable?
8. Given a desired swing velocity, what is the generalized velocity?
9. Is it unique?
10. Is it possible to follow the desired swing trajectory without moving the joints of leg 4? How?

Modeling of Rotorcrafts

- Representing Altitude
- Rotational Velocity
- Gimbal lock

1 Yaw

$$C_{E1}(z, \psi) = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2 Pitch

$$C_{12}(y, \theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}$$

3 Roll

$$C_{2B}(x, \phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}$$

$\phi) - s(\psi)c(\phi) \quad c(\psi)s(\theta)c(\phi) + s(\psi)s(\phi)$
 $\phi) + c(\psi)c(\phi) \quad s(\psi)s(\theta)c(\phi) - c(\psi)s(\phi)$
 $\phi)s(\phi) \quad c(\theta)c(\phi)$

Roll $(-\pi < \phi < \pi)$
Pitch $(-\pi/2 < \theta < \pi/2)$
Yaw $(-\pi < \psi < \pi)$

${}^B\omega_{roll} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix}$

${}^B\omega_{pitch} = C_{2B}^T(x, \phi) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \dot{\theta} \cos \phi \\ \dot{\theta} (-\sin \phi) \end{bmatrix}$

${}^B\omega_{yaw} = [C_{12}(y, \theta) C_{2B}(x, \phi)]^T \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} = [C_{12}(y, \theta) \cdot C_{2B}(x, \phi)]^T \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix}$

$E_r \dot{\chi}_r = {}^B\omega \quad E_r = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix}; \quad \dot{\chi}_r = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$

\Rightarrow Singularity for $\theta = \pm 90^\circ$

Linearized relation at hover

$E_r|_{\phi=0, \theta=0} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = {}^B\omega$

Modeling of Rotorcrafts

Body dynamics

$$\mathbf{M}(\vec{\varphi})\ddot{\vec{\varphi}} + \vec{b}(\vec{\varphi}, \dot{\vec{\varphi}}) + \vec{g}(\vec{\varphi}) + \mathbf{J}_{ex}^T \vec{F}_{ex} = \mathbf{S}^T \vec{\tau}_{act}$$

- Change of momentum and spin in the body frame

$$\begin{bmatrix} mE_{3 \times 3} & 0 \\ 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} {}_B \dot{\mathbf{v}} \\ {}_B \dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} {}_B \boldsymbol{\omega} \times m {}_B \mathbf{v} \\ {}_B \boldsymbol{\omega} \times \mathbf{I}_B \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} {}_B \mathbf{F} \\ {}_B \mathbf{M} \end{bmatrix}$$

$E_{3 \times 3}$: Identity matrix
Torque!!

- Position in the inertial frame and the attitude

$${}_E \dot{\mathbf{x}} = \mathbf{C}_{EB} {}_B \mathbf{v} \quad {}_E \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = {}_B \boldsymbol{\omega}$$

- Forces and moments

- Aerodynamics and gravity

$$\begin{aligned} {}_B \mathbf{F} &= {}_B \mathbf{F}_G + {}_B \mathbf{F}_{Aero} \\ {}_B \mathbf{M} &= {}_B \mathbf{M}_{Aero} \end{aligned} \quad {}_B \mathbf{F}_G = \mathbf{C}_{EB}^T \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}$$

- Hover forces

- Thrust forces in the shaft direction

$${}_B \mathbf{F}_{Aero} = \sum_{i=1}^4 \begin{bmatrix} 0 \\ 0 \\ -T_i \end{bmatrix}$$

$$T_i = b_i \omega_{p,i}^2$$

- Hover moments

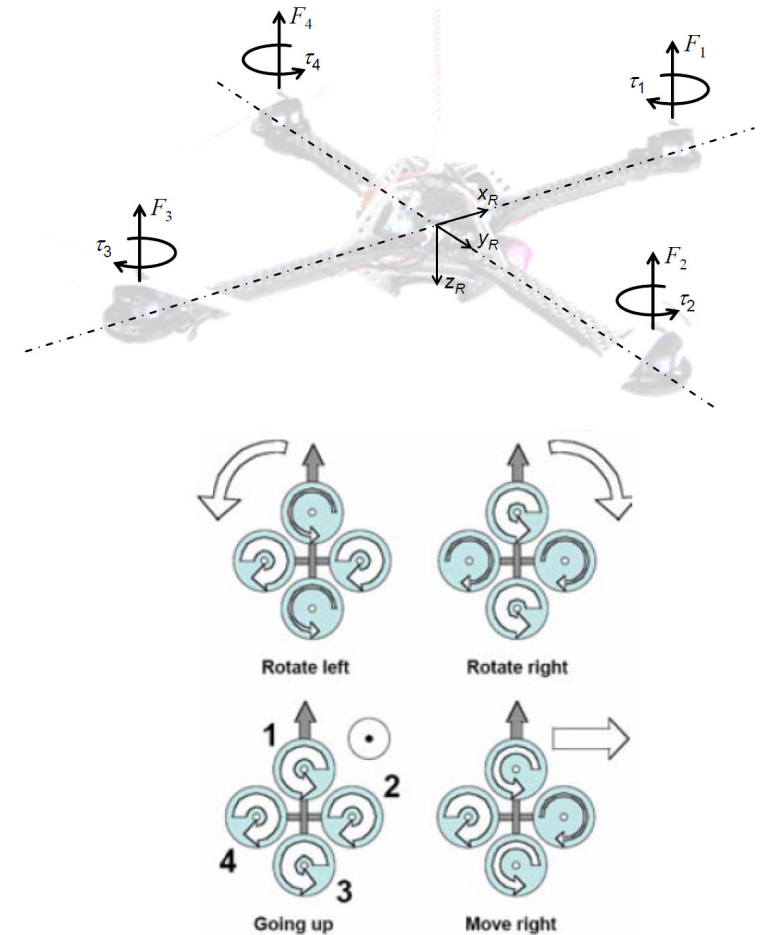
- Thrust induced moment
- Drag torques

$${}_B \mathbf{M}_{Aero} = {}_B \mathbf{M}_T + {}_B \mathbf{Q} = \begin{bmatrix} l(T_4 - T_2) \\ l(T_1 - T_3) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^4 Q_i (-1)^{(i-1)} \end{bmatrix}$$

$$T_i = b_i \omega_{p,i}^2 \quad Q_i = d_i \omega_{p,i}^2$$

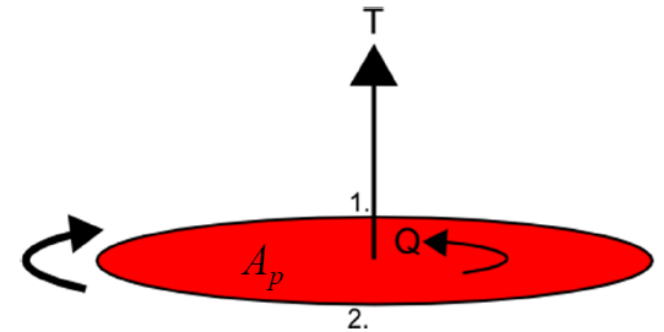
Modeling of a Rotorcraft

- Singularity issues (Tait-bryan angles)
- Analysis of Dynamics
 - Linearization of dynamics at hover
- Control
 - Virtual control input for decoupled attitude control



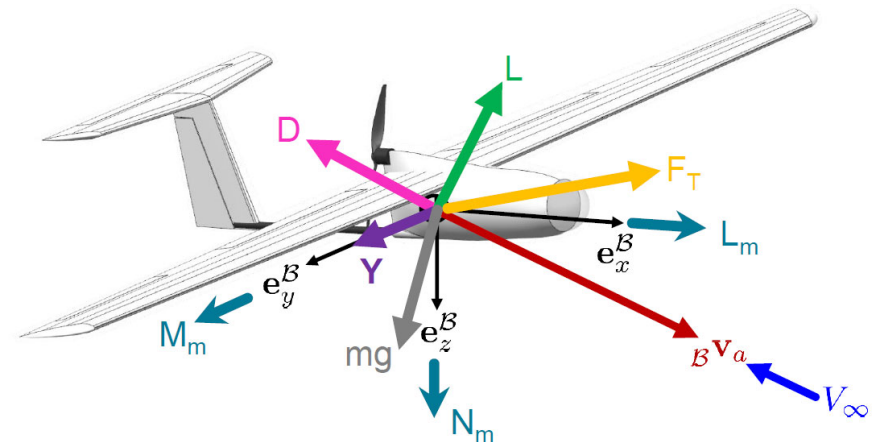
Introduction to propeller aerodynamics

- Thrust force
- Drag torque
- Momentum theory
 - Conservation of fluid mass, fluid momentum and energy
- BEMT (blade element momentum theory)



Fixed wing aerodynamics

- Lift force and drag force
 - Polars of airfoils
- Kinematics and dynamics of an aircraft
 - Forces and moments acting on airplane
 - Lift $L = \frac{1}{2}\rho V^2 S c_L$
 - Drag $D = \frac{1}{2}\rho V^2 S c_D$
 - Thrust
 - Gravity
- Cascaded control of FWs
 - Attitude controller, body rate controller



Preparation for exam

- Online example exam
- Quiz in Script and Lectures
- Exercises in Book
 - Robotics – Modelling, Planning and Control (Siciliano, Sciavicco, Villani, Oriolo)
<http://link.springer.com/book/10.1007%2F978-1-84628-642-1>
- Piazza
 - 140** total posts
 - 452** total contributions
 - 137** instructors' responses
 - 44** students' responses
 - 117 min** avg. response time

Exam

- Date: January 29th 2020, 9-11, HIL
4 A4-pages personal summary + calculator, no PC
- Q&A session TUE January 21st 1200-1330
- You can still post questions on piazza until January 21st



Lecture «Robot Dynamics»: Exam

151-0851-00 V

| | | |
|-----------|-------|---|
| lecture: | HG F3 | Tuesday 10:15 – 12:00, every week |
| exercise: | HG F3 | Wednesday 8:15 – 10:00, according to schedule |

Marco Hutter, Roland Siegwart, and Thomas Stastny

Multiple Choice

A. Multiple Choice

24 pts

Decide whether the following statements are true or false. Cross the checkbox on the corresponding answer. You will be credited 1 point for a correct answer, while 1 pt will be subtracted from the total, if your answer is wrong.

- | | | | |
|--|---------------------------------------|--|--------|
| (1) Parameterizing the orientation of a free floating body with three parameters always results in a singularity problem. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (2) There exists a parameterization χ_{AB} for rotations for which it holds that $\dot{\chi}_{AB} = {}^A \omega_{AB}$. | <input type="radio"/> True | <input checked="" type="radio"/> False | [1 pt] |
| (3) The quaternion corresponding to the elementary rotation of 60° around the z axis is $\xi = \frac{1}{2} (\sqrt{3} \ 0 \ 0 \ 1)^T$. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (4) A six-legged walking robot with three joints per leg has a total of 18 generalized coordinates. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (5) The definition of the generalized coordinates for a six-legged walking robot is unique. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (6) A point contact imposes three constraints on the generalized velocities of a legged robot. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (7) The generalized velocity of a free floating base can be uniquely defined with 6 parameters. | <input checked="" type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (8) Given the linear velocity at all point feet of a six-legged walking robot, it is possible to determine the generalized velocity. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (9) The geometric Jacobian is independent of the choice of generalized coordinates. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (10) The analytical Jacobian has always different dimension than the geometric Jacobian. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (11) Gain-scheduled linear controllers can be used as a computationally efficient means of controlling nonlinear effects of aircraft when further away from the nominal operating point. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |
| (12) Static aerodynamic reaction torques acting on an aircraft may be measured in a wind tunnel. | <input type="radio"/> True | <input type="radio"/> False | [1 pt] |

Multiple Choice

A. Multiple Choice

24 pts

Decide whether the following statements are true or false. Cross the checkbox on the corresponding answer. You will be credited 1 point for a correct answer, while 1 pt will be subtracted from the total, if your answer is wrong.

- | | | | |
|---|--|---|--------|
| (1) Parameterizing the orientation of a free floating body with three parameters always results in a singularity problem. | <input checked="" type="checkbox"/> True | <input type="checkbox"/> False | [1 pt] |
| (2) There exists a parameterization χ_{AB} for rotations for which it holds that $\dot{\chi}_{AB} = {}^A \omega_{AB}$. | <input type="checkbox"/> True | <input checked="" type="checkbox"/> False | [1 pt] |
| (3) The quaternion corresponding to the elementary rotation of 60° around the z axis is $\xi = \frac{1}{2} (\sqrt{3} \ 0 \ 0 \ 1)^T$. | <input checked="" type="checkbox"/> True | <input type="checkbox"/> False | [1 pt] |
| (4) A six-legged walking robot with three joints per leg has a total of 18 generalized coordinates. | <input type="checkbox"/> True | <input checked="" type="checkbox"/> False | [1 pt] |
| (5) The definition of the generalized coordinates for a six-legged walking robot is unique. | <input type="checkbox"/> True | <input checked="" type="checkbox"/> False | [1 pt] |
| (6) A point contact imposes three constraints on the generalized velocities of a legged robot. | <input checked="" type="checkbox"/> True | <input type="checkbox"/> False | [1 pt] |
| (7) The generalized velocity of a free floating base can be uniquely defined with 6 parameters. | <input checked="" type="checkbox"/> True | <input type="checkbox"/> False | [1 pt] |
| (8) Given the linear velocity at all point feet of a six-legged walking robot, it is possible to determine the generalized velocity. | <input type="checkbox"/> True | <input checked="" type="checkbox"/> False | [1 pt] |
| (9) The geometric Jacobian is independent of the choice of generalized coordinates. | <input type="checkbox"/> True | <input checked="" type="checkbox"/> False | [1 pt] |
| (10) The analytical Jacobian has always different dimension than the geometric Jacobian. | <input type="checkbox"/> True | <input checked="" type="checkbox"/> False | [1 pt] |
| (11) Gain-scheduled linear controllers can be used as a computationally efficient means of controlling nonlinear effects of aircraft when further away from the nominal operating point. | <input checked="" type="checkbox"/> True | <input type="checkbox"/> False | [1 pt] |
| (12) Static aerodynamic reaction torques acting on an aircraft may be measured in a wind tunnel. | <input checked="" type="checkbox"/> True | <input type="checkbox"/> False | [1 pt] |

B. Kinematics of an excavator

13 pts

The excavator depicted in Fig. 1 has a hydraulic arm with a bucket at the end-effector. The arm has a total of four degrees of freedom (DOF), namely three rotational DOFs ($\varphi_1, \varphi_2, \varphi_3$) actuated through parallel linkages as well as a linear DOF (x).

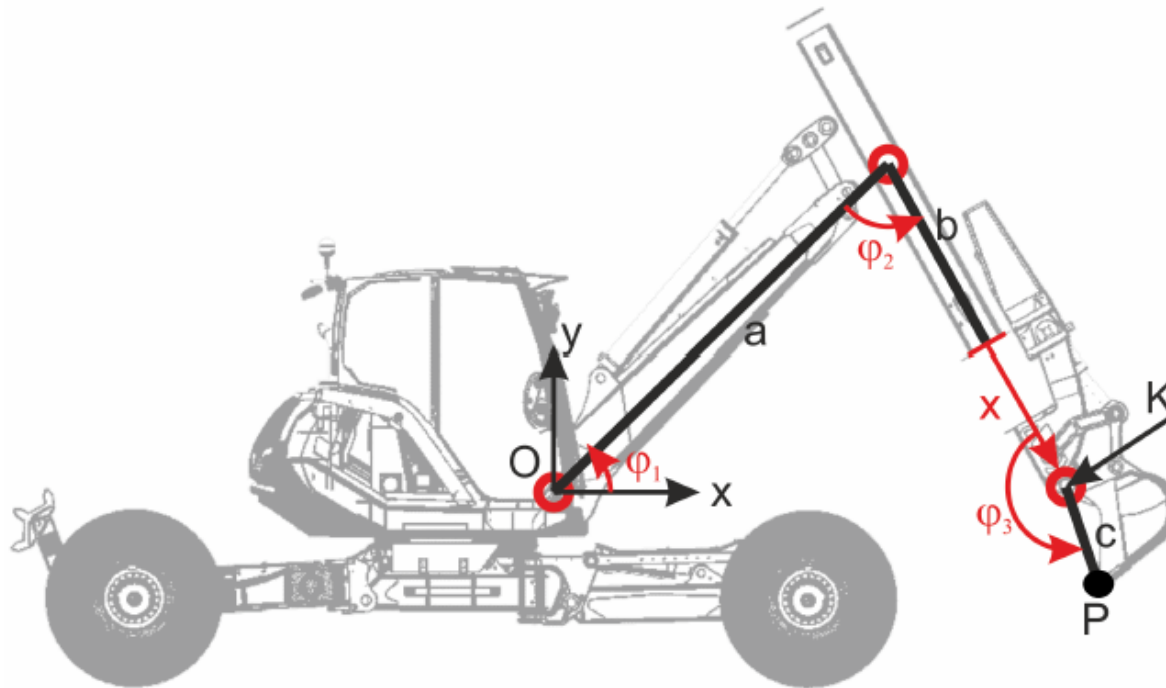


Figure 1: Excavator with four DOF arm and bucket.

(1) What is the generalized coordinate vector of the excavator arm?

$\mathbf{q} =$

(2) Please express the bucket position $\mathbf{r}_P = \mathbf{r}_{OP} \in \mathbb{R}^2$ (point P) and orientation $\varphi_e \in \mathbb{R}$ w.r.t the cabin frame as a function of the generalized coordinates.

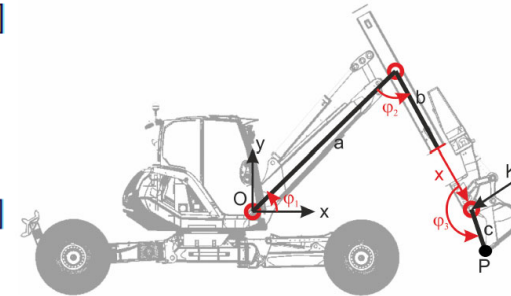
$$\mathbf{r}_P = \begin{pmatrix} \\ \end{pmatrix}$$

$\varphi_e =$

(3) What is the rotation Jacobian of the end-effector?

[1 pt]

[3 pts]



[1 pt]

(1) What is the generalized coordinate vector of the excavator arm?

$$\mathbf{q} = (\varphi_1 \quad \varphi_2 \quad x \quad \varphi_3)^T$$

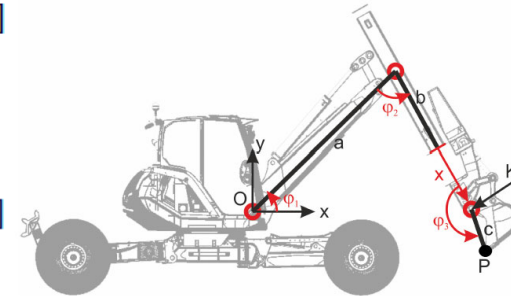
[1 pt]

(2) Please express the bucket position $\mathbf{r}_P = \mathbf{r}_{OP} \in \mathbb{R}^2$ (point P) and orientation $\varphi_e \in \mathbb{R}$ w.r.t the cabin frame as a function of the generalized coordinates.

$$\begin{aligned} \mathbf{r}_P &= \begin{pmatrix} a \cos(\varphi_1) + (b+x) \cos(\varphi_1 + \varphi_2 - \pi) + c \cos(\varphi_1 + \varphi_2 + \varphi_3) \\ a \sin(\varphi_1) + (b+x) \sin(\varphi_1 + \varphi_2 - \pi) + c \sin(\varphi_1 + \varphi_2 + \varphi_3) \end{pmatrix} \\ &= \begin{pmatrix} a \cos(\varphi_1) - (b+x) \cos(\varphi_1 + \varphi_2) + c \cos(\varphi_1 + \varphi_2 + \varphi_3) \\ a \sin(\varphi_1) - (b+x) \sin(\varphi_1 + \varphi_2) + c \sin(\varphi_1 + \varphi_2 + \varphi_3) \end{pmatrix} \end{aligned}$$

$$\varphi_e = \varphi_1 + \varphi_2 + \varphi_3 + (-2\pi)$$

[3 pts]



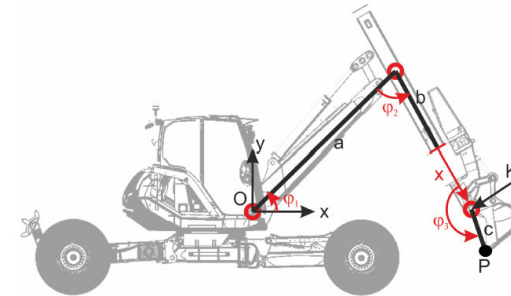
(3) What is the rotation Jacobian of the end-effector?

[1 pt]

$$\mathbf{J}_R = \begin{bmatrix} 1 & 1 & 0 & 1 \end{bmatrix}$$

- (4) Assume that we provide you with the position Jacobian $\mathbf{J}_K = \frac{\partial \mathbf{r}(\mathbf{q})}{\partial \mathbf{q}}$ of point K (mounting point of bucket). What is the position Jacobian \mathbf{J}_P of point P?

$$\mathbf{J}_P = \mathbf{J}_K +$$



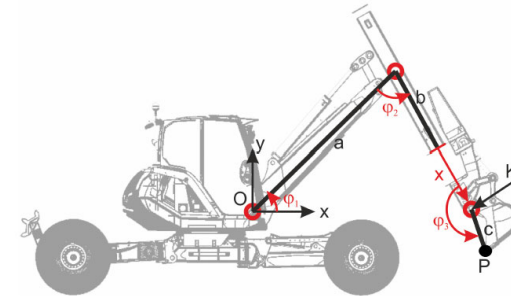
- (5) Assume you want to move the bucket (point P) along a horizontal trajectory with the velocity $\dot{\mathbf{r}}^* = (-v_x, 0)^T$. What least squares minimal generalized velocity produces such motion? (you can use all previously introduced variables)

$$\dot{\mathbf{q}} =$$

- (4) Assume that we provide you with the position Jacobian $\mathbf{J}_K = \frac{\partial \mathbf{r}(\mathbf{q})}{\partial \mathbf{q}}$ of point K (mounting point of bucket). What is the position Jacobian \mathbf{J}_P of point P?

$\mathbf{J}_P = \mathbf{J}_K + c$

$$\mathbf{J}_P = \mathbf{J}_K + c \begin{bmatrix} -\sin(\varphi_1 + \varphi_2 + \varphi_3) & -\sin(\varphi_1 + \varphi_2 + \varphi_3) & 0 & -\sin(\varphi_1 + \varphi_2 + \varphi_3) \\ \cos(\varphi_1 + \varphi_2 + \varphi_3) & \cos(\varphi_1 + \varphi_2 + \varphi_3) & 0 & \cos(\varphi_1 + \varphi_2 + \varphi_3) \end{bmatrix}$$

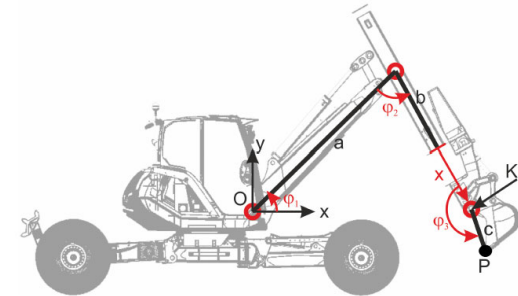


- (5) Assume you want to move the bucket (point P) along a horizontal trajectory with the velocity $\dot{\mathbf{r}}^* = (-v_x, 0)^T$. What least squares minimal generalized velocity produces such motion? (you can use all previously introduced variables)

$$\dot{\mathbf{q}} = \mathbf{J}_P^+ \dot{\mathbf{r}}^*$$

- (6) What is the generalized velocity if you want to move horizontally and keep the bucket orientation constant? Please provide a solution where keeping the orientation has higher priority than moving horizontally. (you can use all previously introduced variables)

$\dot{\mathbf{q}} = \dots$



- (7) Is the previously determined generalized velocity the unique answer to Question 6?

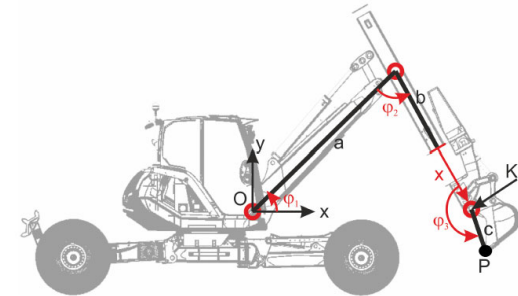
- (6) What is the generalized velocity if you want to move horizontally and keep the bucket orientation constant? Please provide a solution where keeping the orientation has higher priority than moving horizontally. (you can use all previously introduced variables)

$$\dot{\mathbf{q}} = \mathbf{J}_1^+ \dot{\mathbf{x}}_1 + \mathbf{N}_1 (\mathbf{J}_2 \mathbf{N}_1)^+ (\dot{\mathbf{x}}_2 - \mathbf{J}_2 \mathbf{J}_1^+ \dot{\mathbf{x}}_1)$$

$$\dot{\mathbf{q}} = \mathbf{J}_R^+ \mathbf{0} + \mathbf{N}_R \dot{\mathbf{q}}_0$$

$$\dot{\mathbf{r}} = \mathbf{J}_P \mathbf{N}_R \dot{\mathbf{q}}_0$$

$$\Rightarrow \dot{\mathbf{q}} = \mathbf{N}_R (\mathbf{J}_P \mathbf{N}_R)^+ \dot{\mathbf{r}}^*$$

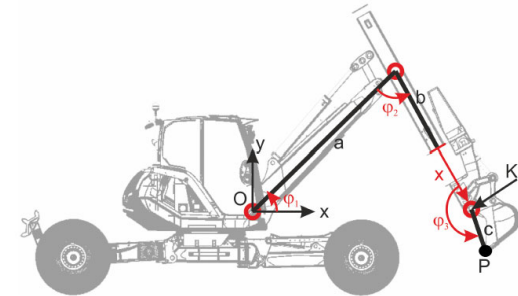


- (7) Is the previously determined generalized velocity the unique answer to Question 6?

No

- (8) What is the formula to determine the generalized velocity for problem 6 if keeping the orientation has the same priority as moving horizontally? (you can use all previously introduced variables)

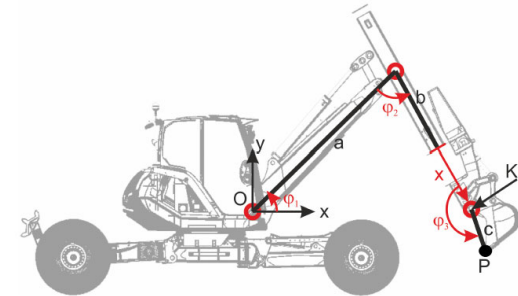
$\dot{\mathbf{q}} = \dots$



- (8) What is the formula to determine the generalized velocity for problem 6 if keeping the orientation has the same priority as moving horizontally? (you can use all previously introduced variables)

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}_K \\ \mathbf{J}_R \end{pmatrix}$$

$$\dot{\mathbf{q}} = \mathbf{J}^+ \begin{bmatrix} \dot{\mathbf{r}}^* \\ 0 \end{bmatrix}$$



C. Statics and Dynamics

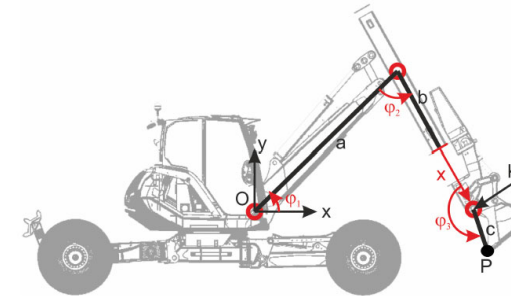
7 pts

In the following exercise, the excavator bucket is in contact with the ground at point P. In vertical direction, the bucket does not move and produces a contact force F_y . Moving in horizontal direction through earth imposes a drag force F_{drag} . Furthermore, assume that we have already determined the equations of motion $M\ddot{q} + b + g = \tau$ for the excavator arm in free motion.

- (1) Given all previously introduced variables, please express the equations of motion in end-effector coordinates $\chi_E = (r_P^T \ \varphi_e)^T \in \mathbb{R}^3$ for the arm in contact with the ground and as a function of a generalized end-effector force $F_E \in \mathbb{R}^3$. Hint: remember that the generalized end-effector force F_E is directly related to the joint torque τ .

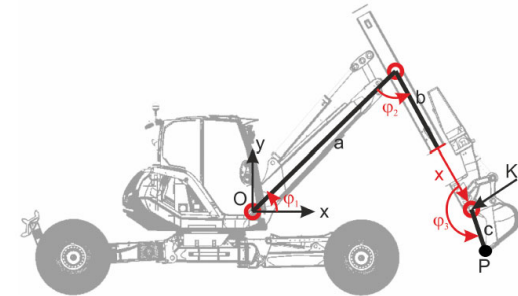
See exercises...

$$\ddot{\chi}_E = \dots$$



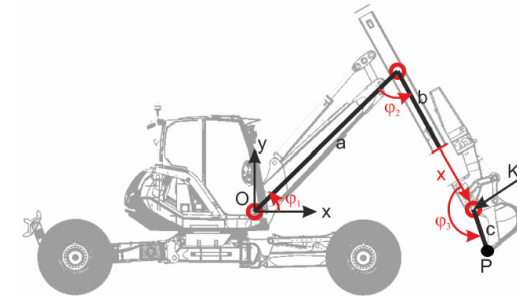
- (2) You now want to develop a controller that calculates the necessary generalized force τ that must be applied in order to move the arm with constant velocity $\dot{\mathbf{r}}^* = (-v_x \ 0)^T$. Thereby, the bucket orientation should not be changed (i.e. $\dot{\varphi}_e^* = 0$). The contact force is defined as in C.1. *Hint:* Use the equation derived in the previous question and introduce control gains where necessary.

See exercises...



- (3) Assume that the excavator has filled the bucket with 150 kg of dirt which must be lifted. What is the necessary joint torque for a given arm configuration \mathbf{q}_t if the excavator moves very slowly $\ddot{\mathbf{q}} \approx \dot{\mathbf{q}} \approx 0$.

See exercises...



THANK YOU

- Thank you for the active participation
- Enjoy the holidays: Make sure you take some days off
- Select your thesis early enough (if you want)
 - <https://rsl.ethz.ch/education-students/student-projects0.html>
- Applications for RA positions
 - <https://rsl.ethz.ch/education-students/Intern.html>