

Robot Dynamics HS19

Thomas Bucher

Based on Summary of Sean Bone
http://weblog.zumguy.com/

January 26, 2020

1 Parametrizations

1.1 Position and velocity

For every position parametrization, there is a linear mapping between linear velocities $\dot{\mathbf{r}}$ and derivatives of the representation $\dot{\chi}$.
 $\dot{\mathbf{r}} = \mathbf{E}_P(\chi_P) \dot{\chi}_P$, $\dot{\chi}_P = \mathbf{E}_P(\chi_P)^{-1} \dot{\mathbf{r}}$

Cartesian Coordinates:

$$\mathbf{E}_{P_c} = \mathbb{I}$$

$$\chi_{P_c} = [x \ y \ z]^T, \quad {}_A \mathbf{r} = [x \ y \ z]^T$$

Cylindrical coordinates:

$$\chi_{P_z} = [\rho \ \theta \ z]^T,$$

$${}_A \mathbf{r} = [\rho \cos \theta \ \rho \sin \theta \ z]^T$$

$$\mathbf{E}_{P_z} = \begin{bmatrix} \cos \theta & -\rho \sin \theta & 0 \\ \sin \theta & \rho \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{E}_{P_z}^{-1} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta / \rho & \cos \theta / \rho & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

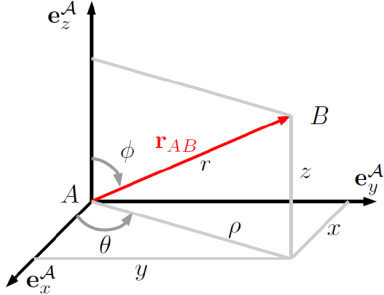
Spherical coordinates:

$$\chi_{P_s} = [r \ \theta \ \phi]^T,$$

$${}_A \mathbf{r} = [r \cos \theta \sin \phi \ r \sin \theta \sin \phi \ z]^T$$

$$\mathbf{E}_{P_s} = \begin{bmatrix} \cos \theta \sin \phi & -r \sin \phi \sin \theta & r \cos \phi \cos \theta \\ \sin \theta \sin \phi & r \cos \theta \sin \phi & r \cos \phi \sin \theta \\ \cos \phi & 0 & -r \sin \phi \end{bmatrix}$$

$$\mathbf{E}_{P_s}^{-1} = \begin{bmatrix} \cos \theta \sin \phi & \sin \theta \sin \phi & \cos \phi \\ -\sin \theta / (r \sin \phi) & \cos \theta / (r \sin \phi) & 0 \\ (\cos \theta \cos \phi) / r & (\cos \phi \sin \theta) / r & -\sin \phi / r \end{bmatrix}$$



1.2 Rotation

$${}_A \mathbf{u} = \mathbf{C}_{AC} \cdot {}_C \mathbf{u} = \mathbf{C}_{AB} \mathbf{C}_{BC} \cdot {}_C \mathbf{u}$$

$$\mathbf{C}_{BA} = \mathbf{C}_{AB}^{-1} = \mathbf{C}_{AB}^T$$

$$\mathbf{C}_{AB} \mathbf{C}_{AB}^T = \mathbf{I}_n \text{ (Orthogonality)}$$

Elementary rotations:

$$\mathbf{C}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi & -\sin \varphi \\ 0 & \sin \varphi & \cos \varphi \end{bmatrix}$$

$$\mathbf{C}_y = \begin{bmatrix} \cos \varphi & 0 & \sin \varphi \\ 0 & 1 & 0 \\ -\sin \varphi & 0 & \cos \varphi \end{bmatrix}$$

$$\mathbf{C}_z = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Euler ZYZ (proper) angles:

$$\chi_{R,ZYZ} = \begin{pmatrix} \text{atan2}(c_{23}, c_{13}) \\ \text{atan2}(\sqrt{c_{13}^2 + c_{23}^2}, c_{33}) \\ \text{atan2}(c_{32}, -c_{31}) \end{pmatrix}$$

Euler ZXZ (proper) angles:

$$\chi_{R,ZXZ} = \begin{pmatrix} \text{atan2}(c_{13}, -c_{23}) \\ \text{atan2}(\sqrt{c_{13}^2 + c_{23}^2}, c_{33}) \\ \text{atan2}(c_{31}, c_{32}) \end{pmatrix}$$

Euler ZYX (Tait-Bryan) angles:

$$\chi_{R,ZYX} = \begin{pmatrix} \text{atan2}(c_{21}, c_{11}) \\ \text{atan2}(-c_{31}, \sqrt{c_{11}^2 + c_{21}^2}) \\ \text{atan2}(c_{32}, c_{33}) \end{pmatrix}$$

Euler XYZ (Cardan) angles:

$$\chi_{R,XYZ} = \begin{pmatrix} \text{atan2}(-c_{23}, c_{33}) \\ \text{atan2}(c_{13}, \sqrt{c_{11}^2 + c_{12}^2}) \\ \text{atan2}(c_{12}, -c_{11}) \end{pmatrix}$$

Angle-axis/Rotation-vector (non-minimal):

$$\chi_{R,AA} = \begin{pmatrix} \theta \\ \mathbf{n} \end{pmatrix}, \quad \mathbf{n} = \frac{1}{2 \sin(\theta)} \cdot \begin{pmatrix} c_{32} - c_{23} \\ c_{31} - c_{13} \\ c_{21} - c_{12} \end{pmatrix},$$

$$\theta = \arccos\left(\frac{c_{11} + c_{22} + c_{33} - 1}{2}\right), \quad \varphi = \theta \cdot \mathbf{n}(\text{nunit})$$

Unit Quaternions (non-minimal):

$$\chi_{R,quat} = \xi = \begin{pmatrix} \xi \\ \xi \end{pmatrix}, \quad \xi^{-1} = \begin{pmatrix} \xi \\ -\xi \end{pmatrix}$$

$$\xi_0 = \cos(\theta/2), \quad \xi = \mathbf{n} \cdot \sin(\theta/2)$$

$$\chi_{R,quat} = \frac{1}{2} \begin{pmatrix} \sqrt{c_{11} + c_{22} + c_{33} + 1} \\ \text{sgn}(c_{32} - c_{23}) \sqrt{c_{11} - c_{22} - c_{33} + 1} \\ \text{sgn}(c_{13} - c_{31}) \sqrt{c_{22} - c_{11} - c_{33} + 1} \\ \text{sgn}(c_{21} - c_{12}) \sqrt{c_{33} - c_{11} - c_{22} + 1} \end{pmatrix}$$

$$\xi_{AB} \otimes \xi_{BC} = \begin{bmatrix} \xi_0 & -\xi_1 & -\xi_2 & -\xi_3 \\ \xi_1 & \xi_0 & -\xi_3 & \xi_2 \\ \xi_2 & \xi_3 & \xi_0 & -\xi_1 \\ \xi_3 & -\xi_2 & \xi_1 & \xi_0 \end{bmatrix}_{AB} \begin{bmatrix} \xi_0 \\ \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix}_{BC}$$

$$\begin{pmatrix} 0 \\ {}_A \mathbf{r} \end{pmatrix} = \xi_{AB} \otimes \begin{pmatrix} 0 \\ {}_B \mathbf{r} \end{pmatrix} \otimes \xi_{AB}^{-1}$$

1.3 Angular Velocity

$$[{}_A \omega_{AB}]_{\times} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} = \dot{\mathbf{C}}_{AB} \mathbf{C}_{AB}^T$$

$${}_A \omega_{AB} = \mathbf{E}_R(\chi_R) \dot{\chi}_R \text{ (see Script p.23-25)}$$

1.4 Transformations

$$\begin{pmatrix} {}_A \mathbf{r}_{AP} \\ 1 \end{pmatrix} = \underbrace{\begin{bmatrix} \mathbf{C}_{AB} & {}_A \mathbf{r}_{AB} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}}_{\mathbf{T}_{AB}} \begin{pmatrix} {}_B \mathbf{r}_{BP} \\ 1 \end{pmatrix}$$

$$\mathbf{T}_{AB}^{-1} = \begin{bmatrix} \overbrace{\mathbf{C}_{AB}^T}^{\mathbf{B}^T \mathbf{B}^A} & -\mathbf{C}_{AB}^T {}_A \mathbf{r}_{AB} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}$$

2 Kinematics

2.1 Velocity in rigid bodies

- \mathbf{v}_P : abs. velocity of P
- \mathbf{a}_P : abs. acceleration of P
- $\Omega_B = \mathcal{I} \omega_B$: angular vel. of frame B
- $\Psi_B = \dot{\Omega}_B$: angular accel. of frame B

$${}_A \mathbf{v}_{AP} = {}_A(\dot{\mathbf{r}}_{AP}) = {}_A \mathbf{v}_{AB} + {}_A \omega_{AB} \times {}_A \mathbf{r}_{BP}$$

In general, unless C is an inertial frame:

$${}_C \mathbf{v}_{AP} = {}_C(\dot{\mathbf{r}}_{AP}) \neq \frac{d}{dt}({}_C \mathbf{r}_{AP})$$

In rigid body formulation:

$$\mathbf{v}_P = \mathbf{v}_B + \Omega \times \mathbf{r}_{BP}$$

$$\mathbf{a}_P = \mathbf{a}_B + \Psi \times \mathbf{r}_{BP} + \Omega \times (\Omega \times \mathbf{r}_{BP})$$

In a kinematic chain:

$$\mathcal{I} \mathbf{v}_{IE} = \mathcal{I} \omega_{I1} \times \mathcal{I} \mathbf{r}_{12} + \dots + \mathcal{I} \omega_{In} \times \mathcal{I} \mathbf{r}_{nE}$$

$$\mathcal{I} \omega_{IE} = \mathcal{I} \omega_{I1} + \mathcal{I} \omega_{12} + \dots + \mathcal{I} \omega_{nE}$$

2.2 Forward kinematics

$$\mathbf{T}_{\mathcal{I}\mathcal{E}}(\mathbf{q}) = \mathbf{T}_{\mathcal{I}0} \left(\prod_{k=1}^{n_j} \mathbf{T}_{k-1,k}(q_k) \right) \mathbf{T}_{n_j \mathcal{E}}$$

2.3 Analytical Jacobian

$$\dot{\chi}(\mathbf{q}) = \frac{\partial \chi}{\partial \mathbf{q}} \dot{\mathbf{q}} = J_A(\mathbf{q}) \cdot \dot{\mathbf{q}} = \begin{bmatrix} \frac{\partial \chi_{pos}}{\partial \mathbf{q}} \\ \frac{\partial \chi_{rot}}{\partial \mathbf{q}} \end{bmatrix} \dot{\mathbf{q}}$$

2.4 Geometric / Basic Jacobian

$$\mathbf{w}_E = \begin{bmatrix} \mathbf{v}_E \\ \omega_E \end{bmatrix} = J_0(\mathbf{q}) \dot{\mathbf{q}}$$

$$J_{0re}(\mathbf{q}) = \begin{bmatrix} J_{0,P} \\ J_{0,R} \end{bmatrix} = \begin{bmatrix} \mathbf{n}_1 \times \mathbf{r}_{1,E} & \dots & \mathbf{n}_n \times \mathbf{r}_{n,E} \\ \mathbf{n}_1 & \dots & \mathbf{n}_n \end{bmatrix}$$

$$J_{0pr}(\mathbf{q}) = \begin{bmatrix} J_{0,P} \\ J_{0,R} \end{bmatrix} = \begin{bmatrix} \mathbf{n}_1 & \dots & \mathbf{n}_n \\ \mathbf{0} & \dots & \mathbf{0} \end{bmatrix}$$

$$\mathcal{I} \mathbf{n}_i = \mathbf{C}_{i-1 \ i-1} \mathbf{n}_i$$

$$\Rightarrow J_0(\mathbf{q}) = E_e(\chi) J_A(\mathbf{q})$$

For planar systems: $J_0(\mathbf{q}) = J_A(\mathbf{q})$

2.5 Inverse differential kinematics

$$\mathbf{w}_E = J \dot{\mathbf{q}} \Rightarrow \dot{\mathbf{q}} = J^+ \mathbf{w}_E$$

where $J^+ = J^T (J J^T)^{-1}$ (Moore-Penrose). However we risk encountering singular configurations \mathbf{q}_s where $\text{rank}(J(\mathbf{q}_s)) < m_0$, m_0 being the number of operational-space coordinates. Here J is badly conditioned. We can mitigate this by using a redundant robot to carefully avoid singularities, and/or by damping the pseudo-inverse:

$$\dot{\mathbf{q}} = J^T (J J^T + \lambda^2 \mathbb{I})^{-1} \mathbf{w}_E$$

Now the pseudo-inverse minimizes $\|\mathbf{w}_E^* - J \dot{\mathbf{q}}\|^2 + \lambda^2 \|\dot{\mathbf{q}}\|^2$ instead of just $\|\mathbf{w}_E^* - J \dot{\mathbf{q}}\|^2$, so convergence is slower but more stable for larger λ .

In a redundant configuration \mathbf{q}^* where $\text{rank}(J(\mathbf{q}^*)) < n$, the pseudoinverse minimizes $\|\dot{\mathbf{q}}\|^2$ while satisfying $\mathbf{w}_E^* = J \dot{\mathbf{q}}$ by using

$$\dot{\mathbf{q}} = J \mathbf{w}_E^* + N \dot{\mathbf{q}}_0$$

$$J(J^+ \mathbf{w}_E^* + N \dot{\mathbf{q}}_0) = \mathbf{w}_E^* \quad \forall \dot{\mathbf{q}}_0$$

where $N = \mathbb{I} - J^+ J \rightarrow JN = 0$.

2.6 Multi-task IDK

Equal Priority

Given n_t tasks $\{J_i, \mathbf{w}_i^*\}$, we have:

$$\dot{\mathbf{q}} = \begin{bmatrix} J_1 \\ \vdots \\ J_{n_t} \end{bmatrix}^+ \begin{pmatrix} \mathbf{w}_1^* \\ \vdots \\ \mathbf{w}_{n_t}^* \end{pmatrix}$$

In case the row-rank of the stacked Jacobian is greater than the column-rank, we are only minimizing $\|\bar{\mathbf{w}} - \bar{J} \dot{\mathbf{q}}\|^2$. We can weigh the tasks with

$$\bar{J}^W = (\bar{J}^T W \bar{J})^{-1} \bar{J}^T W$$

where $W = \text{diag}(w_1, \dots, w_m)$ and we minimize $\|W^{1/2}(\bar{\mathbf{w}} - \bar{J} \dot{\mathbf{q}})\|^2$.

Task Prioritization

$$\dot{\mathbf{q}} = J_1^+ \mathbf{w}_1^* + N_1 \dot{\mathbf{q}}_0$$

$$\mathbf{w}_2 = J_2 \dot{\mathbf{q}} = J_2 (J_1^+ \mathbf{w}_1^* + N_1 \dot{\mathbf{q}}_0)$$

$$\Rightarrow \dot{\mathbf{q}}_0 = (J_2 N_1)^+ (\mathbf{w}_2^* - J_2 J_1^+ \mathbf{w}_1^*)$$

$$\Rightarrow \dot{\mathbf{q}} = J_1^+ \mathbf{w}_1^* + N_1 (J_2 N_1)^+ (\mathbf{w}_2^* - J_2 J_1^+ \mathbf{w}_1^*)$$

In general:

$$\dot{\mathbf{q}} = \sum_{i=1}^{n_t} \bar{N}_i \dot{\mathbf{q}}_i$$

$$\dot{\mathbf{q}}_i = (J_i \bar{N}_i)^+ \left(\mathbf{w}_i^* - J_i \sum_{k=1}^{i-1} \bar{N}_k \dot{\mathbf{q}}_k \right)$$

whereby \bar{N}_i is the Nullspace of the stacked Jacobian $\bar{J}_i = [J_1^T \dots J_{i-1}^T]$. With 2 tasks we first $\min \|\dot{\mathbf{q}}\|^2$ and then $\min \|J_2 \dot{\mathbf{q}} - \mathbf{w}_2^*\|^2$ s.t. $J_1 \dot{\mathbf{q}} = \mathbf{w}_1^* = 0$

2.7 Inverse Kinematics

General goal: $\mathbf{q} = \mathbf{q}(\chi^*)$

1. $\mathbf{q} \leftarrow \mathbf{q}^0$
2. While $\|\chi_e^* \ominus \chi_e(\mathbf{q})\| > \text{tol}$ do
3. $J_A \leftarrow J_A(\mathbf{q}) = \frac{\partial \chi_e}{\partial \mathbf{q}}(\mathbf{q})$
4. $J_A^+ \leftarrow (J_A)^+$
5. $\Delta \chi_e \leftarrow \chi_e^* \ominus \chi_e(\mathbf{q})$
6. $\mathbf{q} \leftarrow \mathbf{q} + J_A^+ \Delta \chi_e$

One issue is that for very large errors $\Delta \chi_e$, we get too imprecise. We can avoid this by scaling the update with a factor $0 < k < 1$: $\mathbf{q} \leftarrow \mathbf{q} + k J_A^+ \Delta \chi_e$. But we still have issues inverting J_A in singular configurations. An alternative is $\mathbf{q} \leftarrow \mathbf{q} + \alpha J_A^T \Delta \chi_e$, which converges for small α . We must also appropriately compute the difference $\chi_e^* \ominus \chi_e(\mathbf{q})$ depending on the parametrization. For cartesian coordinates, this

is regular vector subtraction. Also note that with cartesian coordinates $J_{0,P} = J_{A,P}$. For rotational difference we can extract the rotation vector $\Delta\varphi$ from the "rotation difference", and use that for the update:

$$\begin{aligned} \mathbf{C}_{GS}(\Delta\varphi) &= \mathbf{C}_{GI}(\varphi^*)\mathbf{C}_{SI}(\varphi^T)^T \\ \mathbf{q} &\leftarrow \mathbf{q} + k_{PR} J_{0,R}^+ \Delta\varphi \end{aligned}$$

2.8 Trajectory control

Position: with $\Delta\mathbf{r}_e^t = \mathbf{r}_e^*(t) - \mathbf{r}_e(\mathbf{q}^t)$

$$\dot{\mathbf{q}}^* = J_{e0P}^+(\mathbf{q}^t)(\dot{\mathbf{r}}_e^*(t) + k_{PP}\Delta\mathbf{r}_e^t)$$

Orientation: with $\Delta\varphi$ as above,

$$\dot{\mathbf{q}}^* = J_{e0R}^+(\mathbf{q}^t)(\omega_e^*(t) + k_{PR}\Delta\varphi)$$

3 Dynamics

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

- $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n_q \times n_q}$ Mass matrix (\perp).
- $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^{n_q}$ Gen. pos., vel., accel.
- $\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n_q}$ Coriolis and centrifugal terms
- $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^{n_q}$ Gravity terms
- $\boldsymbol{\tau} \in \mathbb{R}^{n_q}$ External generalized forces
- $\mathbf{F}_c \in \mathbb{R}^{3 \times n_c}$ External cartesian forces
- $\mathbf{J}_c(\mathbf{q}) \in \mathbb{R}^{n_c \times n_q}$ Geometric Jacobian of location where external forces apply

$$\begin{aligned} \begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\Omega} \end{pmatrix} &= \begin{bmatrix} J_P \\ J_R \end{bmatrix} \dot{\mathbf{q}} \\ \begin{pmatrix} \mathbf{a}_s \\ \boldsymbol{\Psi} \end{pmatrix} &= \begin{pmatrix} \dot{\mathbf{v}}_s \\ \dot{\boldsymbol{\Omega}} \end{pmatrix} = \begin{bmatrix} J_P \\ J_R \end{bmatrix} \ddot{\mathbf{q}} + \begin{bmatrix} \dot{J}_P \\ \dot{J}_R \end{bmatrix} \dot{\mathbf{q}} \end{aligned}$$

3.1 Newton-Euler method

- m body mass
- $\boldsymbol{\Theta}_S$ inertia matrix around CoG
- $\mathbf{p}_S = m\mathbf{v}_S$ linear momentum
- $\dot{\mathbf{N}}_S = \boldsymbol{\Theta}_S \cdot \dot{\boldsymbol{\Omega}}$ angular momentum around CoG
- $\dot{\mathbf{p}} = m\mathbf{a}_S$ change in linear momentum
- $\dot{\mathbf{N}}_S = \boldsymbol{\Theta}_S \cdot \boldsymbol{\Psi} + \boldsymbol{\Omega} \times \boldsymbol{\Theta}_S \cdot \boldsymbol{\Omega}$ change in angular momentum

Cut each link free as a single rigid body, and introduce constraint forces \mathbf{F}_i acting on the body at the joint. Then apply conservation of linear and angular momentum in all DoFs subject to all external forces (*including* constraints \mathbf{F}_i):

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

For calculations all quantities must be in the same coordinate system. For the inertia matrix we have ${}_{\mathcal{B}}\boldsymbol{\Theta} = \mathbf{C}_{BA} \cdot {}_{\mathcal{A}}\boldsymbol{\Theta} \cdot \mathbf{C}_{BA}^T$.

3.2 Lagrange method

Define the *Lagrangian function*:

$$\mathcal{L} = \mathcal{T} - \mathcal{U}$$

Where \mathcal{T} is the kinetic energy and \mathcal{U} the potential energy. Then the *Euler-Lagrange equation of the second kind* holds for the total external generalized forces $\boldsymbol{\tau}$:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right) - \left(\frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right) = \boldsymbol{\tau}$$

The kinetic energy for a system of n_b bodies is defined as:

$$\begin{aligned} \mathcal{T} &:= \sum_{i=1}^{n_b} \left(\frac{1}{2} m_i {}_{\mathcal{A}}\dot{\mathbf{r}}_{S_i}^T {}_{\mathcal{A}}\dot{\mathbf{r}}_{S_i} + \frac{1}{2} {}_{\mathcal{B}}\dot{\boldsymbol{\Omega}}_{S_i}^T {}_{\mathcal{B}}\boldsymbol{\Theta}_{S_i} \cdot {}_{\mathcal{B}}\dot{\boldsymbol{\Omega}}_{S_i} \right) \\ &= \frac{1}{2} \dot{\mathbf{q}}^T \left(\underbrace{\sum_{i=1}^{n_b} (J_{S_i}^T m_i J_{S_i} + J_{R_i}^T \boldsymbol{\Theta}_{S_i} J_{R_i})}_{\mathbf{M}(\mathbf{q})} \right) \dot{\mathbf{q}} \\ &= \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} \end{aligned}$$

The potential energy is typically in the form of gravitational and elastic terms:

$$\mathcal{U} = - \underbrace{\sum_{i=1}^{n_b} \mathbf{r}_{S_i}^T (m_i \mathbf{g} \cdot \mathbf{e}_g)}_{\text{gravitational}} + \underbrace{\sum_{j=1}^{n_E} \frac{1}{2} k_j (d(\mathbf{q}) - d_{0,j})^2}_{\text{elastic}}$$

Here we have n_E elastic components with coefficients k_j and rest configuration $d_{0,j}$.

3.3 Proj. Newton-Euler Method

$$\begin{aligned} \mathbf{M} &= \sum_{i=1}^{n_b} ({}_{\mathcal{A}}J_{S_i}^T m_i {}_{\mathcal{A}}J_{S_i} + {}_{\mathcal{B}}J_{R_i}^T {}_{\mathcal{B}}\boldsymbol{\Theta}_{S_i} {}_{\mathcal{B}}J_{R_i}) \\ \mathbf{b} &= \sum_{i=1}^{n_b} ({}_{\mathcal{A}}J_{S_i}^T m_i {}_{\mathcal{A}}\dot{J}_{S_i} \dot{\mathbf{q}} + {}_{\mathcal{B}}J_{R_i}^T ({}_{\mathcal{B}}\boldsymbol{\Theta}_{S_i} {}_{\mathcal{B}}\dot{J}_{R_i} \dot{\mathbf{q}} \\ &\quad + {}_{\mathcal{B}}\boldsymbol{\Omega}_{S_i} \times {}_{\mathcal{B}}\boldsymbol{\Theta}_{S_i} {}_{\mathcal{B}}\boldsymbol{\Omega}_{S_i})) \\ \mathbf{g} &= \sum_{i=1}^{n_b} (-{}_{\mathcal{A}}J_{S_i}^T {}_{\mathcal{A}}\mathbf{F}_{g,i}) \\ \tau_{F,ext} &= \sum_{j=1}^{n_{f,ext}} J_{P,j}^T F_j \\ \tau_{T,ext} &= \sum_{k=1}^{n_{m,ext}} J_{R,k}^T T_{ext,k} \end{aligned}$$

4 Floating-base dynamics

Generalized coordinates are now $\mathbf{q} = [\mathbf{q}_b^T \mathbf{q}_j^T]^T$, where \mathbf{q}_b are the generalized coordinates of the base (position and orientation). The generalized velocities are therefore no longer $\dot{\mathbf{q}}$, but are denoted $\mathbf{u} = [{}_{\mathcal{T}}\mathbf{v}_B^T {}_{\mathcal{B}}\boldsymbol{\omega}_{IB}^T \dot{\mathbf{q}}_j^T]^T$.

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

- $\mathbf{u}, \dot{\mathbf{u}} \in \mathbb{R}^{n_u}$ Gen. vel., accel.
- \mathbf{S} selection matrix of actuated joints, $u_j = Su = [0_{6 \times 6} \mathbb{I}_{6 \times n_j}](u_b u_j)^T$
- $\mathbf{F}_{ext} \in \mathbb{R}^{3 \times n_c}$ External cartesian forces acting on robot
- $\mathbf{J}_{ext}(\mathbf{q}) \in \mathbb{R}^{n_c \times n_u}$ Geometric Jacobian of location where external forces apply

Position and velocity of a point Q on the robot:

$$\begin{aligned} {}_{\mathcal{T}}\mathbf{r}_{IQ}(\mathbf{q}) &= {}_{\mathcal{T}}\mathbf{r}_{IB}(\mathbf{q}) + \mathbf{C}_{IB}(\mathbf{q}) \cdot {}_{\mathcal{B}}\mathbf{r}_{BQ}(\mathbf{q}) \\ {}_{\mathcal{T}}\mathbf{v}_Q &= \underbrace{\left[{}_{\mathcal{I}3 \times 3} -\mathbf{C}_{IB} \cdot [{}_{\mathcal{B}}\mathbf{r}_{BQ}] \times \mathbf{C}_{IB} \cdot {}_{\mathcal{B}}J_{P_{q_j}}(\mathbf{q}_j) \right]}_{= {}_{\mathcal{T}}J_Q(\mathbf{q})} \cdot \mathbf{u} \end{aligned}$$

4.1 Contact kinematics

The point of contact C is not allowed to move: $\mathbf{r}_C = \text{const.}$ and $\dot{\mathbf{r}}_C = \ddot{\mathbf{r}}_C = \mathbf{0}$. Written in generalized coordinates these are:

$${}_{\mathcal{T}}J_{C_i} \mathbf{u} = \mathbf{0}, \quad {}_{\mathcal{T}}J_{C_i} \dot{\mathbf{u}} + {}_{\mathcal{T}}J_{C_i} \mathbf{u} = \mathbf{0}$$

We can therefore stack the constraint Jacobians:

$$\mathbf{J}_c = \begin{bmatrix} {}_{\mathcal{T}}J_{C_1} \\ \vdots \\ {}_{\mathcal{T}}J_{C_{n_c}} \end{bmatrix} \in \mathbb{R}^{3n_c \times (n_b + n_j)}$$

By using the nullspace projection \mathbf{N}_c of \mathbf{J}_c we can still move the system:

$$\begin{aligned} \mathbf{0} = \dot{\mathbf{r}} &= \mathbf{J}_c \dot{\mathbf{q}} \Rightarrow \dot{\mathbf{q}} = \mathbf{J}_c^+ \mathbf{0} + \mathbf{N}_c \dot{\mathbf{q}}_0 \\ \mathbf{0} = \ddot{\mathbf{r}} &= \mathbf{J}_c \ddot{\mathbf{q}} + \dot{\mathbf{J}}_c \dot{\mathbf{q}} \Rightarrow \ddot{\mathbf{q}} = \mathbf{J}_c^+ (-\dot{\mathbf{J}}_c \dot{\mathbf{q}}) + \mathbf{N}_c \ddot{\mathbf{q}}_0 \end{aligned}$$

The contact Jacobian tells us how the system can move. If we partition it into the part relating to the base and the part relating to the joints:

- $\mathbf{J}_c = [\mathbf{J}_{c,b} \mathbf{J}_{c,j}]$
- $\text{rank}(\mathbf{J}_{c,b})$ is the number of constraints on the base \rightarrow the number of controllable base DoFs.

- $\text{rank}(\mathbf{J}_c) - \text{rank}(\mathbf{J}_{c,b})$ is the number of constraints on the actuators.

Quadruped (18 DoF; 6 for base, 12 actuators):

| | | | | | |
|---|---|---|---|---|----|
| | | | | | |
| Total constraints $\text{rank}(\mathbf{J}_c)$ | 0 | 3 | 6 | 9 | 12 |
| Base constraints $\text{rank}(\mathbf{J}_{c,b})$ | 0 | 3 | 5 | 6 | 6 |
| Internal constraints $\text{rank}(\mathbf{J}_j) - \text{rank}(\mathbf{J}_{c,j})$ | 0 | | 1 | 3 | 6 |
| Uncontrollable DoFs $6 - \text{rank}(\mathbf{J}_{c,b})$ | 6 | 3 | 1 | 0 | 0 |

4.2 Support-consistent dynamics

If we use **soft contacts** to model the contact, we simply introduce an external force acting on the robot:

$$\mathbf{F}_c = k_p(\mathbf{r}_c - \mathbf{r}_{c0}) + k_d \dot{\mathbf{r}}_c$$

However such problems are hard to accurately solve numerically (slow system dynamics, fast contact dynamics).

Instead it works better to use **hard contacts**. We impose the kinematic constraint ${}_{\mathcal{T}}J_{C_i} \dot{\mathbf{u}} + {}_{\mathcal{T}}J_{C_i} \mathbf{u} = \mathbf{0}$ from above and calculate the resulting force and null-space matrix:

$$\begin{aligned} \mathbf{F}_c &= (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1} (\mathbf{J}_c \mathbf{M}^{-1} (\mathbf{S}^T \boldsymbol{\tau} - \mathbf{b} - \mathbf{g}) + \dot{\mathbf{J}}_c \mathbf{u}) \\ \mathbf{N}_c &= \mathbb{I} - \mathbf{M}^{-1} \mathbf{J}_c^T (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1} \mathbf{J}_c \\ \Rightarrow \boxed{\mathbf{N}_c^T (\mathbf{M} \dot{\mathbf{u}} + \mathbf{b} + \mathbf{g}) = \mathbf{N}_c^T \mathbf{S}^T \boldsymbol{\tau}, \quad \mathbf{J}_c \mathbf{N}_c = \mathbf{0}} \end{aligned}$$

By defining the *end-effector inertia* $\boldsymbol{\Lambda}_c = (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1}$ we can write the kinetic energy loss on impact:

$$\begin{aligned} \mathbf{u}^+ &= \mathbf{N}_c \mathbf{u}^- \\ E_{loss} = \Delta E_{kin} &= -\frac{1}{2} \Delta \mathbf{u}^T \mathbf{M} \Delta \mathbf{u} = -\frac{1}{2} \dot{\mathbf{r}}^{-T} \mathbf{M} \dot{\mathbf{r}}^- \end{aligned}$$

5 Dynamic control

5.1 Joint-space Dynamic Control

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

Torque as a function of position and velocity error:

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

Compensate for gravity by adding an estimated gravity term:

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

Compensate for **system dynamics**:

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

If the model is exact, we have $\mathbb{I}\ddot{\mathbf{q}} = \ddot{\mathbf{q}}^*$ (decoupled control), meaning we can perfectly control system dynamics. We could apply a PD-control law, making each joint behave like a mass-spring-damper with unitary mass:

$$\ddot{\mathbf{q}}^* = k_p(\mathbf{q}^* - \mathbf{q}) + k_d(\dot{\mathbf{q}}^* - \dot{\mathbf{q}})$$

$$\omega = \sqrt{k_p}, \quad D = \frac{k_d}{2\sqrt{k_p}}$$

5.2 Task-space Dynamic Control

$$\dot{\mathbf{w}}_e = J_e \ddot{\mathbf{q}} + \dot{J}_e \dot{\mathbf{q}} = J_e \mathbf{M}^{-1}(\boldsymbol{\tau} - \mathbf{b} - \mathbf{g}) + \dot{J}_e \dot{\mathbf{q}}$$

$$\boldsymbol{\tau} = J_e^T \mathbf{F}_e \quad , \quad \ddot{\mathbf{q}} = J_e^+ (\dot{\mathbf{w}}_e^* - \dot{J}_e \dot{\mathbf{q}})$$

End-Effector Motion Control

Generalized framework to control motion and force. **End-effector dynamics**:

$$\begin{aligned} \Lambda \dot{\mathbf{w}}_e + \boldsymbol{\mu} + \mathbf{p} &= \mathbf{F}_e \\ \Lambda &= (\mathbf{J}_e \mathbf{M}^{-1} \mathbf{J}_e^T)^{-1} \\ \boldsymbol{\mu} &= \Lambda \mathbf{J}_e \mathbf{M}^{-1} \mathbf{b} - \Lambda \dot{J}_e \dot{\mathbf{q}} \\ \mathbf{p} &= \Lambda \mathbf{J}_e \mathbf{M}^{-1} \mathbf{g} \end{aligned}$$

represent the end-effector inertia, centrifugal/coriolis and gravitational terms in task space. Following from the dynamics the **end-effector control** can be found:

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

$$\mathbf{w}_e^* = \mathbf{k}_p \begin{pmatrix} r_e^* - r_e \\ \Delta \phi_e \end{pmatrix} + \mathbf{k}_d (\mathbf{w}_e^* - \mathbf{w}_e)$$

$$\Rightarrow \begin{pmatrix} r_e^* - r_e \\ \Delta \phi_e \end{pmatrix} = \begin{bmatrix} \mathcal{K} & 0 \\ 0 & E_R \end{bmatrix}$$

$$\dot{\mathbf{w}}_e^* = k_p \mathbf{E} (\chi_e^* \boxminus \chi_e) + k_d (\mathbf{w}_e^* - \mathbf{w}_e) \quad \underbrace{\dot{\mathbf{w}}_e^*(t)}_{\text{trajectory control}}$$

Operational Space Control

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

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

with \mathbf{S}_M and \mathbf{S}_F being the selection matrices for Motion and Force. Let \mathbf{C} represent the rotation from the inertial frame to the contact force frame. The selection matrices can be calculated as (with $\sigma_i \in \{0, 1\}$):

$$\boldsymbol{\Sigma}_p = \begin{bmatrix} \sigma_{px} & 0 & 0 \\ 0 & \sigma_{py} & 0 \\ 0 & 0 & \sigma_{pz} \end{bmatrix}, \quad \boldsymbol{\Sigma}_r = \begin{bmatrix} \sigma_{rx} & 0 & 0 \\ 0 & \sigma_{ry} & 0 \\ 0 & 0 & \sigma_{rz} \end{bmatrix}$$

$$\mathbf{S}_M = \begin{bmatrix} \mathbf{C}^T \boldsymbol{\Sigma}_p \mathbf{C} & 0 \\ 0 & \mathbf{C}^T \boldsymbol{\Sigma}_r \mathbf{C} \end{bmatrix}$$

$$\mathbf{S}_F = \begin{bmatrix} \mathbf{C}^T (\mathbf{I} - \boldsymbol{\Sigma}_p) \mathbf{C} & 0 \\ 0 & \mathbf{C}^T (\mathbf{I} - \boldsymbol{\Sigma}_r) \mathbf{C} \end{bmatrix}$$

OSC with multiple objectives

Example: quadruped with three stationary legs and one in swing.

- Leg swing: $\ddot{\mathbf{r}}_{OF} = \mathbf{J}_F \ddot{\mathbf{q}}_F + \dot{\mathbf{J}}_F \dot{\mathbf{q}}_F = \ddot{\mathbf{r}}_{OF,des}(t) = k_p (\mathbf{q}^* - \mathbf{r}) + k_d (\dot{\mathbf{r}}^* - \dot{\mathbf{r}}) + \ddot{\mathbf{r}}^*$
- Body movement (translation and orientation): $\dot{\mathbf{w}}_B = \mathbf{J}_B \dot{\mathbf{q}}_B + \dot{\mathbf{J}}_B \dot{\mathbf{q}}_B = \dot{\mathbf{w}}_{OB,des}(t) = k_p \begin{pmatrix} \mathbf{r}^* - \mathbf{r} \\ \boldsymbol{\varphi}^* \boxminus \boldsymbol{\varphi} \end{pmatrix} + k_d (\mathbf{w}^* - \mathbf{w}) + \dot{\mathbf{w}}^*$
- Enforce contact constraints: $\ddot{\mathbf{r}}_c = \mathbf{J}_c \ddot{\mathbf{q}}_c + \dot{\mathbf{J}}_c \dot{\mathbf{q}}_c = 0$

Solve for generalized acceleration and torque giving each task **equal priority**:

$$\ddot{\mathbf{q}}^* = \begin{bmatrix} \mathbf{J}_F \\ \mathbf{J}_B \\ \mathbf{J}_c \end{bmatrix}^+ \left(\begin{pmatrix} \ddot{\mathbf{r}}_{OF,des}(t) \\ \dot{\mathbf{w}}_{B,des}(t) \\ 0 \end{pmatrix} - \begin{bmatrix} \dot{\mathbf{J}}_F \\ \dot{\mathbf{J}}_B \\ \dot{\mathbf{J}}_c \end{bmatrix} \dot{\mathbf{q}} \right)$$

Solve **with prioritization**:

$$\begin{aligned} \ddot{\mathbf{q}}^* &= \sum_{i=1}^{n_t} \mathbf{N}_i \ddot{\mathbf{q}}_i, \\ \ddot{\mathbf{q}}_i &:= (\mathbf{J}_i \mathbf{N}_i)^+ \left(\mathbf{w}_i^* - \dot{\mathbf{J}}_i \dot{\mathbf{q}} - \mathbf{J} \sum_{k=1}^{i-1} \mathbf{N}_k \ddot{\mathbf{q}}_k \right) \end{aligned}$$

Where \mathbf{N}_i is the nullspace projection of $\mathbf{J}_i := [\mathbf{J}_1^T \dots \mathbf{J}_i^T]^T$.

5.3 Inv. Dynamics Floating-Base

Given a desired acceleration \mathbf{u}^* from the support-consistent dynamics follows:

$$\boldsymbol{\tau}^* = (\mathbf{N}_c^T \mathbf{S}^T)^+ \mathbf{N}_c^T (\mathbf{M} \mathbf{u}^* + \mathbf{b} + \mathbf{g}) + \underbrace{\mathcal{N}(\mathbf{N}_c^T \mathbf{S}^T) \boldsymbol{\tau}_0^*}_{\text{multiple solutions}}$$

Task-Space Control as QP

The behaviour of a robotic system can be described as multi-task control problem with the optimization variable x as follows:

$$x_{fixedB} = \begin{pmatrix} \ddot{\mathbf{q}} \\ \mathbf{F}_c \\ \boldsymbol{\tau} \end{pmatrix} \quad \text{or} \quad x_{floatingB} = \begin{pmatrix} \dot{\mathbf{u}} \\ \mathbf{F}_c \\ \boldsymbol{\tau} \end{pmatrix}$$

Using the optimization variable x the EoM $\mathbf{M} \ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g} + \mathbf{J}_c^T \mathbf{F}_c = \mathbf{S}^T \boldsymbol{\tau}$ can be formulated as least square problem $\mathbf{A}x - \mathbf{b} = 0$:

$$\mathbf{A} = \begin{bmatrix} \hat{\mathbf{M}} & \hat{\mathbf{J}}_c^T & -\mathbf{S}^T \end{bmatrix} \quad \mathbf{b} = -\hat{\mathbf{b}} - \hat{\mathbf{g}}$$

To achieve a desired acceleration in the **joint space** $\ddot{\mathbf{q}}$ or at a point of interest in the **task space** $\mathbf{J} \ddot{\mathbf{q}} + \dot{\mathbf{J}} \dot{\mathbf{q}} = \dot{\mathbf{w}}_e$:

$$\mathbf{A} = [\mathbf{I} \text{ or } \hat{\mathbf{J}}_i \quad 0 \quad 0] \quad \mathbf{b} = \ddot{\mathbf{q}} \text{ or } \dot{\mathbf{w}}_e^* - \hat{\mathbf{J}}_i \dot{\mathbf{q}}$$

Pushing with a certain force $\mathbf{F}_i = \mathbf{F}_i^*$:

$$\mathbf{A} = [0 \quad \mathbf{I} \quad 0] \quad \mathbf{b} = \mathbf{F}_i^*$$

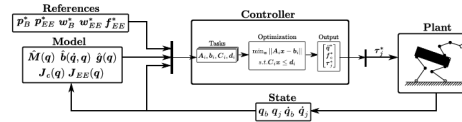
6 Legged Robots

6.1 Hierarchical Optimization

Formulating a Hierarchical Optimization (HO) problem as a QP:

$$\min ||\mathbf{A}_i x - \mathbf{b}_i|| \quad , \quad \mathbf{C}_i x \leq \mathbf{d}_i$$

Achieved by following control scheme for robots:



The HO variable x and EoM are defined as:

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

$$x = [\ddot{\mathbf{q}}^T \quad \mathbf{f}_c^T \quad \boldsymbol{\tau}_j^T]^T$$

Task 1: Fulfill equation of Motion:

$$\mathbf{A}_1 = [\mathbf{M}(q) - \mathbf{J}_c^T - \mathbf{S}^T] \quad , \quad \mathbf{b}_1 = -\mathbf{b} - \mathbf{g}$$

Task 2: Ensure feet stationary on ground:

$$\mathbf{A}_2 = [\mathbf{J}_{c,lin} \quad 0 \quad 0] \quad , \quad \mathbf{b}_2 = \dot{\mathbf{w}}_c^* - \mathbf{J}_{c,lin} \dot{\mathbf{q}}$$

Task 3: Move body accord ref. trajectory:

$$\mathbf{w}_B^* = k_p (p_B^* - p_B) + k_d (w_B^* - w_B)$$

$$\mathbf{A}_3 = [\mathbf{J}_B \quad 0 \quad 0] \quad , \quad \mathbf{b}_3 = \dot{w}_B^* - \mathbf{J}_B \dot{\mathbf{q}}$$

...To be continued...

7 Rotorcraft

Propeller thrust and drag proportional to squared rotational speed (b : thrust constant; d : drag constant):

$$T_i = b \omega_{p,i}^2, \quad Q_i = d \omega_{p,i}^2$$

7.1 Kinematics

Use Tait-Bryan angles, consisting of yaw ψ (Z-axis), pitch θ (Y-axis) and roll ϕ (X-axis).

$$\mathbf{C}_{EB} = \mathbf{C}_{E1}(\mathbf{z}, \psi) \cdot \mathbf{C}_{12}(\mathbf{y}, \theta) \cdot \mathbf{C}_{2B}(\mathbf{x}, \phi)$$

Angular velocity:

$$\begin{aligned} \mathcal{B} \boldsymbol{\omega} &= \mathcal{B} \boldsymbol{\omega}_{\text{roll}} + \mathcal{B} \boldsymbol{\omega}_{\text{pitch}} + \mathcal{B} \boldsymbol{\omega}_{\text{yaw}} \\ \mathcal{B} \boldsymbol{\omega}_{\text{roll}} &= (\dot{\psi}, 0, 0)^T \\ \mathcal{B} \boldsymbol{\omega}_{\text{pitch}} &= \mathbf{C}_{2B}^T(0, \dot{\theta}, 0)^T \\ \mathcal{B} \boldsymbol{\omega}_{\text{yaw}} &= [\mathbf{C}_{12} \cdot \mathbf{C}_{2E}]^T(0, 0, \dot{\phi})^T \\ \mathcal{B} \boldsymbol{\omega} &= J_r \dot{\chi}_r = J_r \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \end{aligned}$$

$$J_r = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \quad \theta = \phi = 0 \quad \mathbb{I}_{3 \times 3}$$

NB: singularity for $\theta = \pm 90^\circ$ (Gimbal lock).

7.2 Dynamics

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

Change of momentum and spin in the body frame (\mathbf{M} = total moment/torque):

$$\begin{bmatrix} m \mathbb{I} & 0 \\ 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathcal{B} \dot{\mathbf{v}} \\ \mathcal{B} \dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} \mathcal{B} \boldsymbol{\omega} \times m \mathcal{B} \mathbf{v} \\ \mathcal{B} \boldsymbol{\omega} \times \mathbf{I} \mathcal{B} \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \mathcal{B} \mathbf{F} \\ \mathcal{B} \mathbf{M} \end{bmatrix}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \mathbf{C}_{EB} \mathbf{v} = \mathbf{C}_{EB} \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

Forces and moments come from gravity and aerodynamics:

$$\mathcal{B} \mathbf{F} = \mathcal{B} \mathbf{F}_G + \mathcal{B} \mathbf{F}_{Aero}$$

$$\mathcal{B} \mathbf{M} = \mathcal{B} \mathbf{M}_{Aero}$$

$$\mathcal{B} \mathbf{F}_G = \mathbf{C}_{EB}^T \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}$$

$$\mathcal{B} \mathbf{F}_{Aero} = \sum_{i=1}^4 \begin{bmatrix} 0 \\ 0 \\ -T_i = -b \omega_{p,i}^2 \end{bmatrix}$$

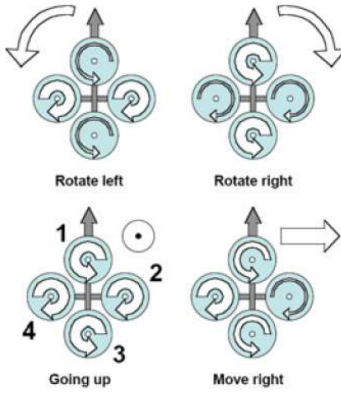
$$\mathcal{B} \mathbf{M}_{Aero} = \mathcal{B} \mathbf{M}_T + \mathcal{B} \mathbf{Q} =$$

$$\begin{bmatrix} l(T_4 - T_2) \\ l(T_1 - T_3) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^4 Q_i (-1)^{(i-1)} \end{bmatrix}$$

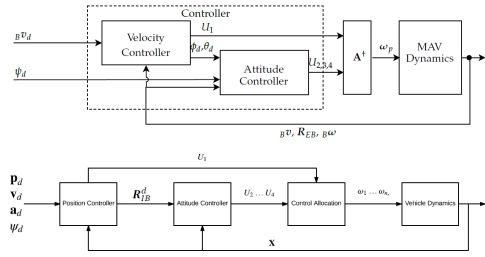
Full control over all rotational speeds, independently of the current position state. **Only directly control of vertical cartesian velocity - attitude control must be used for full position control.**

7.3 Control

Movement directions with four propellers:



Possible Control Structures:



To formulate the control architecture, a virtual control input U is used:

$$\begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{pmatrix} = A \begin{pmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{pmatrix}, \quad A^\dagger = A^T(AA^T)^{-1}$$

Hence the translational and rotational dynamics are stated as follows:

$$\begin{aligned} \dot{p} &= R_{EB} \mathcal{B}v \\ \mathcal{B}\dot{v} &= -\omega \times \mathcal{B}v + \begin{pmatrix} 0 \\ 0 \\ \frac{U_1}{m} \end{pmatrix} + R_{EB}^T g \\ \dot{R}_{EB} &= R_{EB}\omega \\ \dot{\omega} &= J^{-1}(-\omega \times J\omega + \begin{pmatrix} U_2 \\ U_3 \\ U_4 \end{pmatrix}) \end{aligned}$$

Equilibrium Point:

$$\phi = \theta = p = q = r = 0; U_2 = U_3 = U_4 = 0$$

$$U_1 = mg \sin(x) \approx x, \cos(x) \approx 1$$

This results in following Control Inputs:

$$\begin{aligned} U_1 &= T_{des} \\ U_2 &= (\phi_{des} - \phi)k_p \text{Roll} - \dot{\phi}k_d \text{Roll} \\ U_3 &= (\theta_{des} - \theta)k_p \text{Pitch} - \dot{\theta}k_d \text{Pitch} \\ U_4 &= (\psi_{des} - \psi)k_p \text{Yaw} - \dot{\psi}k_d \text{Yaw} \end{aligned}$$

...Velocity or Position Control...

7.4 Propeller aerodynamics

Propeller in hover:

- Thrust force T normal to prop. plane, $|T| = \frac{\rho}{2} A_P C_T (\omega_p R_p)^2$
- Drag torque Q , around rotor plane $|Q| = \frac{\rho}{2} A_P C_Q (\omega_p R_p)^2 R_p$
- C_T and C_Q depend on blade pitch angle (prop geometry), Reynolds number (prop speed, velocity, rotational speed).

Propeller in forward flight: additional forces due to force unbalance between forward- and backward-moving props.

- Hub force H (orthogonal to T , opposite to horizontal flight direction V_H), $|H| = \frac{\rho}{2} A_P C_H (\omega_p R_p)^2 R_p$
- Rolling torque R around flight direction $|R| = \frac{\rho}{2} A_P C_R (\omega_p R_p)^2 R_p$
- C_R and C_H depend on advance ratio $\mu = \frac{V}{\omega_p R_p}$

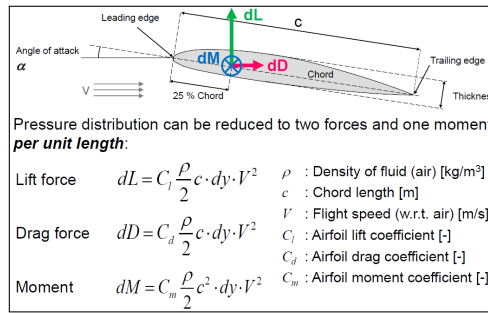
Ideal power consumption at hover: $P = \frac{F_{Thrust}^{3/2}}{\sqrt{2\rho A_R}} = \frac{(mg)^{3/2}}{\sqrt{2\rho A_R}}$. The prop efficiency is measured with the Figure of Merit FM:

$$FM = \frac{\text{Ideal power to hover}}{\text{Actual power to hover}} < 1$$

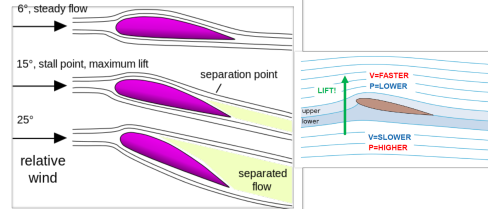
Blade Elemental and Momentum Theory (BEMT): blade shape determines drag and lift coefficients c_D , c_L .

8 Fixed-Wing

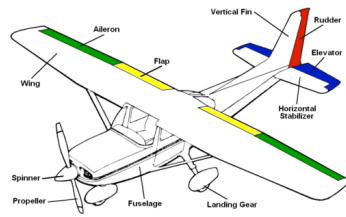
8.1 Aerodynamic Basics



Stall does highly depend on fluid, foil and Reynolds number:



Small FW provide following control surfaces:



8.2 Kinematics

Body-axis \mathcal{B}

Body velocity: $\mathcal{B}v_a = (u, v, w)^T$

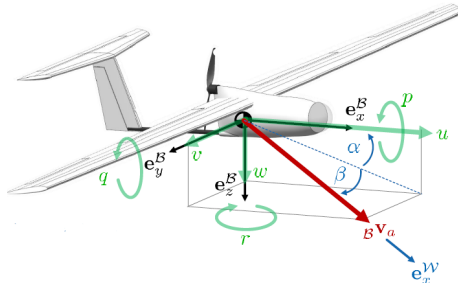
Body rates: $\mathcal{B}\omega = (p, q, r)^T$

Air-mass relative speed (airspeed): $V = \sqrt{u^2 + v^2 + w^2}$

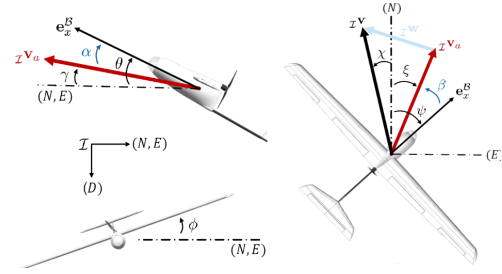
Wind-axis \mathcal{W}

Angle of attack: $\alpha = \tan^{-1}(w/u)$

Sideslip angle: $\beta = \sin^{-1}(v/V)$



Polar Coordinates



γ : Flight path angle from horizon

θ : Pitch angle from horizon to x

ϕ : Roll angle, rotation about x

ξ : Heading angle, from North

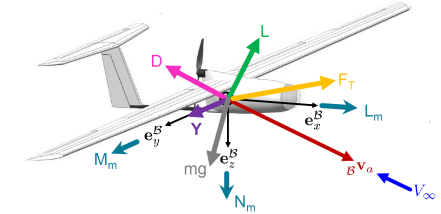
ψ : Yaw angle, from North

χ : Course angle from North

$\mathcal{I}v$: Ground based internal velocity / ground speed)

$$\begin{aligned} \mathcal{I}v_a &= C_{IB} \mathcal{B}v_a \quad \mathcal{I}v = \mathcal{I}v_a + \mathcal{I}w = \mathcal{I}\dot{r} = \\ &= \begin{bmatrix} V \cos \gamma \cos \xi + \omega_N \\ V \cos \gamma \sin \xi + \omega_E \\ -V \sin \gamma + \omega_D \end{bmatrix} \end{aligned}$$

8.3 Dynamics



Lift $L = \frac{1}{2} \rho V^2 S c_L$

Drag $D = \frac{1}{2} \rho V^2 S c_D$

Rolling Moment $L_m = \frac{1}{2} \rho V^2 S b c_l$

Pitching Moment $M_m = \frac{1}{2} \rho V^2 S \bar{c} c_m$

Yawing Moment $N_m = \frac{1}{2} \rho V^2 S b c_n$

EoM Translation

$$\begin{aligned} \dot{u} &= rv - qw + \frac{1}{2} (F_T \cos \epsilon - D \cos \alpha + L \sin \alpha) - g \\ \dot{v} &= pw - ru + \frac{1}{m} Y + g \sin \phi \cos \theta \\ \dot{w} &= qu - pv + \frac{1}{m} (F_T \sin \epsilon - D \sin \alpha - L \cos \alpha) + g \end{aligned}$$

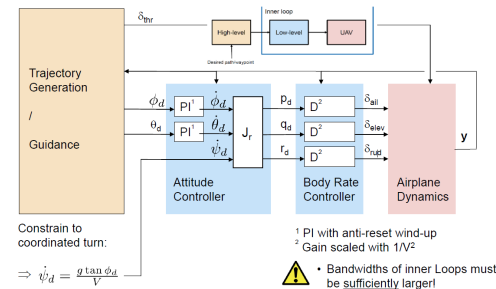
$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = C_{IB} \begin{bmatrix} u \\ v \\ w \end{bmatrix} + \mathcal{I}w$$

EoM Rotation (Assumed $I_{xz} \approx 0$)

$$\begin{aligned} \dot{p} &= \frac{1}{I_{xx}} (L_m + L_{m_T} - qr(I_{zz} - I_{yy})) \\ \dot{q} &= \frac{1}{I_{yy}} (M_n + M_{m_T} - pr(I_{xx} - I_{zz})) \\ \dot{r} &= \frac{1}{I_{zz}} (N_m + N_{m_T} - pq(I_{yy} - I_{xx})) \end{aligned}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = J_r^{-1} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} p + q \tanh \theta \sin \phi + r \tan \theta \cos \phi \\ q \cos \phi - r \sin \phi \\ q \frac{\sin \phi}{\cos \theta} + r \frac{\cos \phi}{\cos \theta} \end{bmatrix}$$

8.4 Control



Steady level turning flight $\mathcal{B}v_a = \mathcal{B}\dot{w} = 0$
Steady (unaccelerated)

$\theta = \alpha \rightarrow \gamma = 0$ Level

$\phi = \text{const.} \neq 0$ Turning

$\xi = \psi$ No Sideslip

$Y = 0$ Coordinated turn

L increases with $\frac{1}{\cos \phi}$

V_{min} increases with $\sqrt{\frac{1}{\cos \phi}}$

From Force balance and assumption $\dot{\psi} \approx \dot{\xi}$