Robot Dynamics

Silvan Stadelmann - 28. Oktober 2025 - v0.0.1

github.com/silvasta/summary-rodyn

Contents

1 Kinematics

1.1 Vectors and Positions

Parametrizations

1.2 Linear Velocity

$$r = r(\chi)$$
$$\dot{r} = \frac{\partial r}{\partial \chi}\dot{\chi} = E_p(\chi)\dot{\chi}$$

1.3 Rotations

$${}_{\mathcal{A}}r_{AP} = C_{\mathcal{AB}} \cdot {}_{\mathcal{B}}\, r_{AP}$$

$$C_{\mathcal{AB}} =$$

$$C_{\mathcal{AC}} = C_{\mathcal{AB}} \cdot C_{\mathcal{BC}}$$

- elementary rotatons

- homogenous transfomations

$$T_{\mathcal{AB}} = \begin{bmatrix} C_{\mathcal{AB}} & {}_{\mathcal{A}}r_{AP} \\ 0_{1\times 3} & 1 \end{bmatrix}$$

- passive and active rotation

1.4 Angular Velocity

1.5 Parametrization of 3d Rotations

- Rotation matrix
  - $3 \times 3 = 9$ parameters
  - Orthonormality $= 6$ constraints
- Euler angles
- Angle axis
- Rotation vector
- Unit quaternions
  - 4 parameters
  - no singularity, unitary constraints

1.6 Euler angles

- Consecutive elementary rotations

1.7 angle axis and rotation vector

1.8 Unit Quaternions

1.9 Algebra with Quaternions

- product

- rotating vector

1.10 Time Derivatives and Rotational Velocity

- problem with singularities, use Quaternions

1.11 Multi Body Kinematics

Generalized coordinates

$$q = \begin{pmatrix} q1 \\ \vdots \\ q_n \end{pmatrix}$$

End-effector configuration parameters

$$\chi_e = \begin{pmatrix} \chi_{eP} \\ \chi_{eR} \end{pmatrix}$$

Operational space coordinates

$$\chi_o = \begin{pmatrix} \chi_{oP} \\ \chi_{oR} \end{pmatrix}$$

1.12 Kinematics of Systems of Bodies

- Joint space
- Task space

1.13 Forward Kinematics

$$\chi_e = \chi_e(q)$$

use multiplied transfomation matrix T_{IE}

1.14 Jacobians

$$\boxed{\dot{\chi_e} = J_{eA}(q)\dot{q}}$$

$$\boxed{J_{eA} = \frac{\partial \chi_e}{\partial q_1} = \begin{bmatrix} \frac{\partial \chi_1}{\partial q_1} & \cdots & \frac{\partial \chi_1}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial \chi_m}{\partial q_1} & \cdots & \frac{\partial \chi_m}{\partial q_n} \end{bmatrix}}$$

TODO: geometric, algebraic L3.21

1.15 Velocity in Moving Bodies

- Rigid body formulation

ArAP = ArAB + [AwAB]*CAB*BrBP

vP = vB + Om x rBP

1.16 Jacobians for Prismatic Joint

- Position
- Rotation

1.17 Inverse Kinematics

$$w_e^\star = J_{e0}\dot{q}$$

Things to take into account:

- Singularities
- Redundancy
- Nullspace of matrix

1.18 Multi-task control

- singel Task
- Stacked Task
- Priorization

Error analysis

Mapping associated with the Jacobian

Joint space <=> Task space

1.19 Trajectory control

1.20 Velocity in Moving Bodies

2 Dynamics

3 Lecture 6 - Introduction Dynamic Control

3.1 Position vs. Torque Controlled Robot Arms

3.2 Joint Impedance and Inverse Dynamics Control

Joint Impedance Control

- Formula
- Formula with Gravity compensation

Inverse Dynamics Control

Compensate for system dynamics + PD law on acceleration

- every joint behaves like decoupled mass-spring damper

- Eigenfrequency

- Damping

Task-space dynamics control

- single task: just use pseudo-inverse

- multiple task: stack J_i, w_i, pseudo-inverse, done (equal priority)

3.3 Task-space Dynamics Control

- end-effector dynamics

- end-effector motion control

- trajectory control (feedforwad term)

3.4 Interaction Control

Operational Space Control

- F_c as contact force

Selection Matrix

- seperate motion and force directions

3.5 Inverse Dynamics as QP

- use: some sort of "mass-matrix weighted pseudo-inverse"

- quadratic optimization

- Solving set of QPs

4 Legged Robot

5 Rotorcraft

6 Fixed-Wing