Robot Dynamics

Silvan Stadelmann - 29. Oktober 2025 - v0.0.1

github.com/silvasta/summary-rodyn

Contents

1 Kinematics

1.1 Vectors and Positions

Parametrizations

1.2 Linear Velocity

$$
\begin{aligned}
r &= r(\chi) \\
\dot{r} &= \frac{\partial r}{\partial \chi}\dot{\chi} = E_p(\chi)\dot{\chi}
\end{aligned}
$$

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

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

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

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

Equations of Motion

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

$M(q)$	Mass matrix
$\ddot{q}$	Generalized coordinates
$b(q,\dot{q})$	Centrifugal and Coriolis forces
$g(q)$	Gravity forces
τ	Generalized forces
F_c	External forces
J_c	Contact Jacobian

3 Principle of Virtual Work

- Principle of virtual work (D'Alembert's Principle)

- Dynamic equilibrium imposes zero virtual work (for all virtual displacements)

- Newton's law for every particle in direction it can move

4 Single Rigid Body

5 Newton-Euler Method

6 Projected Newton-Euler

Principle of virtual work for multi-body systems

7 Lagrange II

- Kinetic energy

- Potential energy

8 External Forces and Torques

Forces $\tau_{F_{ext}} = \sum_{j=1}^{n_{f,ext}} J_{P,j}^T T_{ext,k}$

Torques $\tau_{T_{ext}} = \sum_{k=1}^{n_{m,ext}} J_{R,k}^T T_{ext,k}$

Actuators $\tau_{a,k} = (J_{S_k} - J_{S_{k-1}})^T F_{a,k} + (J_{R_k} - J_{R_{k-1}})^T$

9 Velocity in Moving Bodies

- Definitions

- Moving Frame

10 Prismatic Joints

TODO: Jacobians

- Position Jacobian

- Rotation Jacobian

- Example

11 Dynamic Control

Joint Impedance Control

Inverse Dynamics Control

Compensate for system dynamics + PD law on acceleration

- every joint behaves like decoupled mass-spring damper

- Eigenfrequency

- Damping

12 Task-space Dynamics control

- single task: just use pseudo-inverse

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

12.1 End-effector dynamics

- end-effector motion control

- trajectory control (feedforwad term)

13 Interaction Control

13.1 Operational Space Control

Generalized framework to control motion and force

- F_c as contact force

13.2 Selection Matrix

- seperate motion and force directions

13.3 Inverse Dynamics as QP

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

- quadratic optimization

- Solving set of QPs

14 Floating Base Dynamics

14.1 Generalized coordinates

$$q = \begin{pmatrix} q_b \\ q_j \end{pmatrix} \quad \text{with} \quad q_b = \begin{pmatrix} q_{b_P} \\ q_{b_R} \end{pmatrix} \in \mathbb{R}^3 \times SO(3)$$

14.2 Generalized velocities and accelerations

$$u = \begin{pmatrix} {}_I v_B \\ {}_B \omega_I B \\ \dot{\varphi}_1 \\ \vdots \\ \dot{\varphi}_{n_j} \end{pmatrix} \quad \dot{u} = \begin{pmatrix} {}_I a_B \\ {}_B \psi_I B \\ \ddot{\varphi}_1 \\ \vdots \\ \ddot{\varphi}_{n_j} \end{pmatrix} \in \mathbb{R}^{6+n_j} = \mathbb{R}^{n_u}$$

Very often, people write $\dot{q}$ but they mean u

$$u = E_{fb} \cdot \dot{q} \quad \text{with} \quad E_{fb} \cdot \dot{q} = \begin{bmatrix} 1_{3\times3} & 0 & 0 \\ 0 & E_{\chi R} & 0 \\ 0 & 0 & 1_{n_j \times n_j} \end{bmatrix}$$

14.3 Differential kinematics

15 Contacts and Constraints

15.1 Properties of Contact Jacobian

16 Dynamics of Floating Base Systems

- External Forces
- Soft Contact
- Hard Contact

17 Constraint consistent dynamics

17.1 Contact Dynamics

18 Dynamic Control Methods

- Behavior as Multiple Tasks
- Internal Forces

18.1 Control using inverse dynamics

19 Task Space Control as Quadratic Program

20 Legged Robot

21 Rotorcraft

22 Fixed-Wing