Robotics I

02. Coordinate Transformation (Introduction)

Review

What is a robot and what types are there?

- Definition
- Classification of Robot Types
  - Industrial Robots
  - Service Robots
  - Entertainment and Edutainment Robots
  - Biomimetic Robots
- Current Research Questions

KUKA KR 1300 titan

DLR Rollin' Justin

imes 3(P)RRR

Sony Qrio

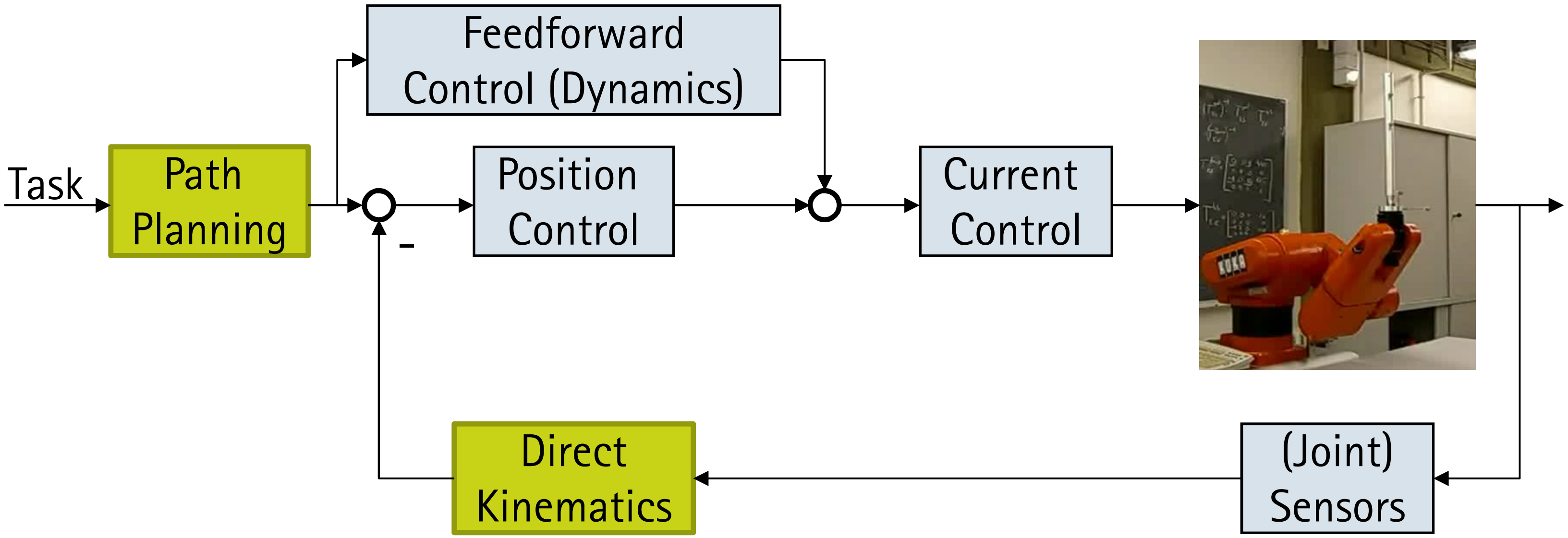
Total Content

Control
- Sensors
- Advanced Control Methods
- Multi-Axis Control
- Single-Axis Control
- Dynamics: Newton-Euler and Lagrange
- Path Planning
- Kinematically Redundant Robots
- Jacobian Matrix – Velocities and Forces
- Direct and Inverse Kinematics
- Coordinate Transformations
- Introduction

Kinematics and Dynamics

Total Content

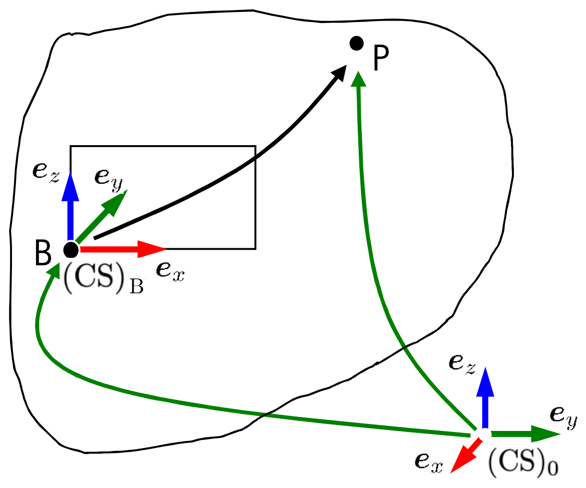
Connection of essential lecture contents in the form of a control loop:
Coordinate transformations form the basis for the (differential) kinematic modeling of a robot and, accordingly, also for path planning.



Coordinate Transformation

How do I know where the robot tip (end effector) is located?

- Joint- and Taskspace
- Coordinate Systems
- Rotations I
  - Rotation Matrices
  - Elementary Rotations
  - Composition of Rotations
- Describing Spatial Position and Orientation
- Homogeneous Transformation
- Rotations II
  - Rotation around an Axis
  - Quaternions



Literatur
W. Khalil & E. Dombre – Modeling, Identification & Control of Robots, S. 13–25, 53 ff.
J. J. Craig – Introduction to Robotics Mechanics and Control, S. 19–59
V. E. Kremer – Quaternions and SLERP
J. Diebel - Representing Attitude: Euler Angles, Unit Quaternions, and Rotation

Joint and Task Space

Joint coordinates

Correspond to minimal coordinates of a serial robot

$\boldsymbol{q} = (q_1,\ q_2,\ \ldots,\ q_n)^{\mathrm{T}}$

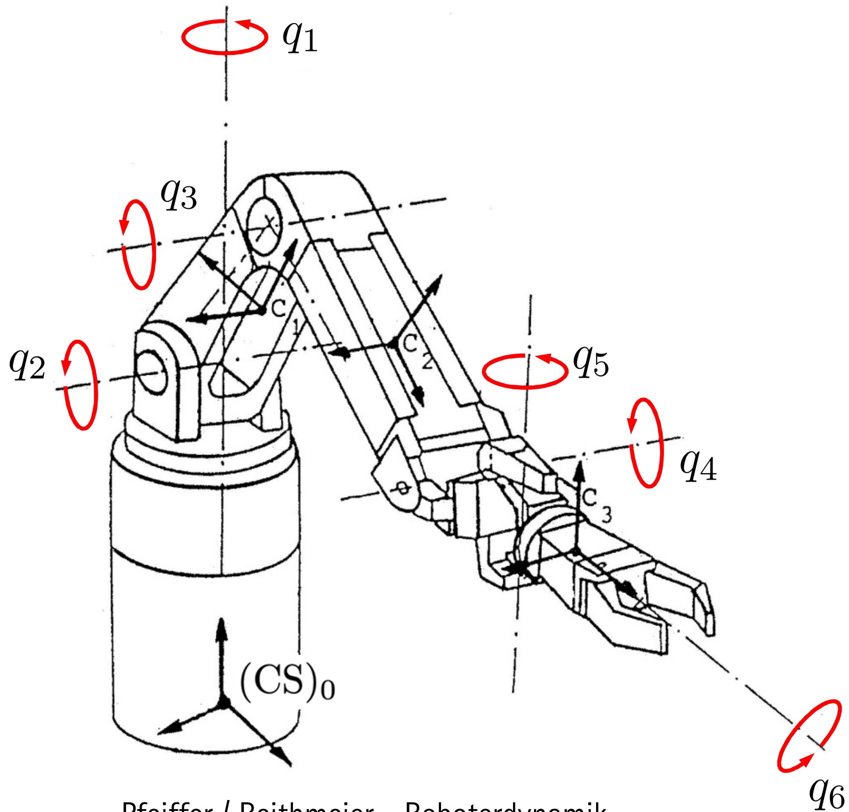
- n number of degrees of freedom
- q_i Angle (rotational actuators) or lengths (prismatic actuators)

Minimal coordinates

Independently defined parameters that uniquely describe a system

$\mathcal{Q} = \{\boldsymbol{q} \mid \boldsymbol{q}_{\min} \leq \boldsymbol{q} \leq \boldsymbol{q}_{\max}\}$

Joint space (Configuration space)



Pfeiffer / Reithmeier - Roboterdynamik

Joint and Task Space

Types of joints

Rotational Joint (R): 1 DOF

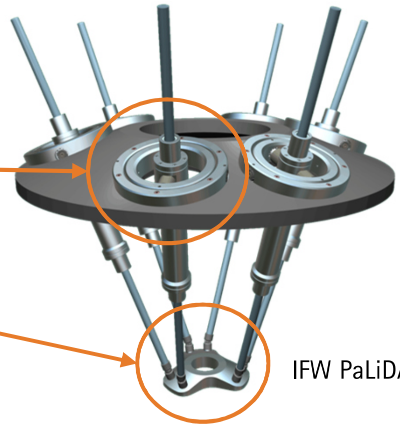
Prismatic Joints (P): 1 DOF

KUKA KR 10 scara R850

Special cases (Combinations)

Universal Joints (U): 2 DOF

Spherical Joints (S): 3 DOF



IFW PaLiDA

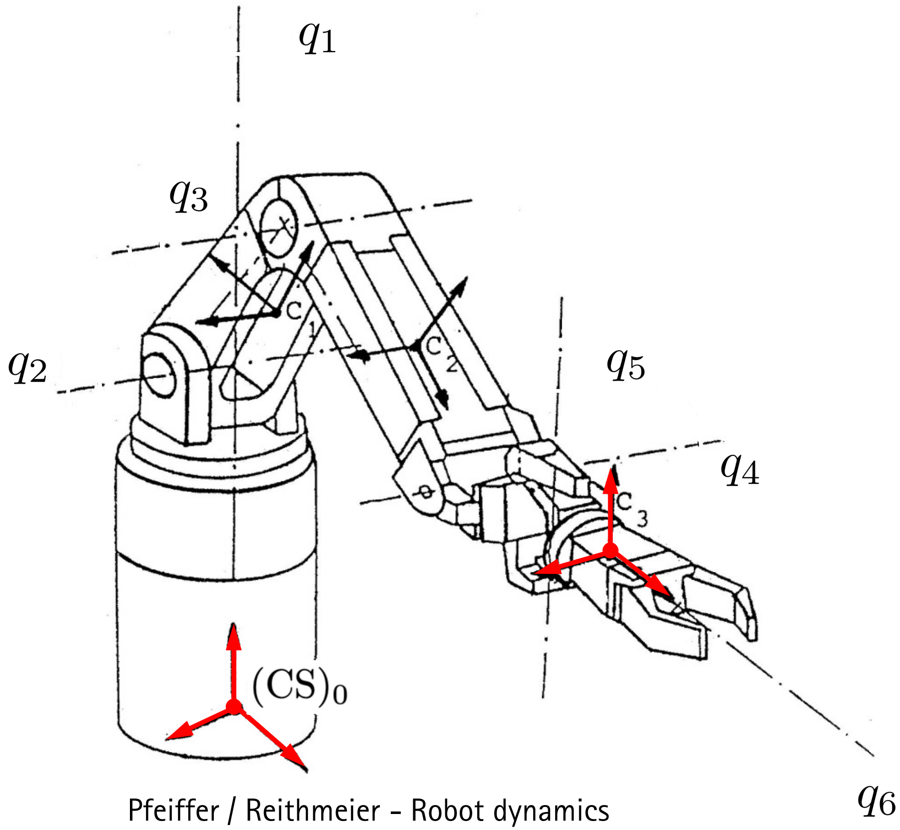
Joint and Task Space

Environmental coordinates

End-effector pose
(position and orientation)

$$\boldsymbol{x}_\mathrm{E} = (x_{\mathrm{E}_1},\ x_{\mathrm{E}_2},\ \ldots,\ x_{\mathrm{E}_m})^\mathrm{T}$$

- m: Number of independent parameters



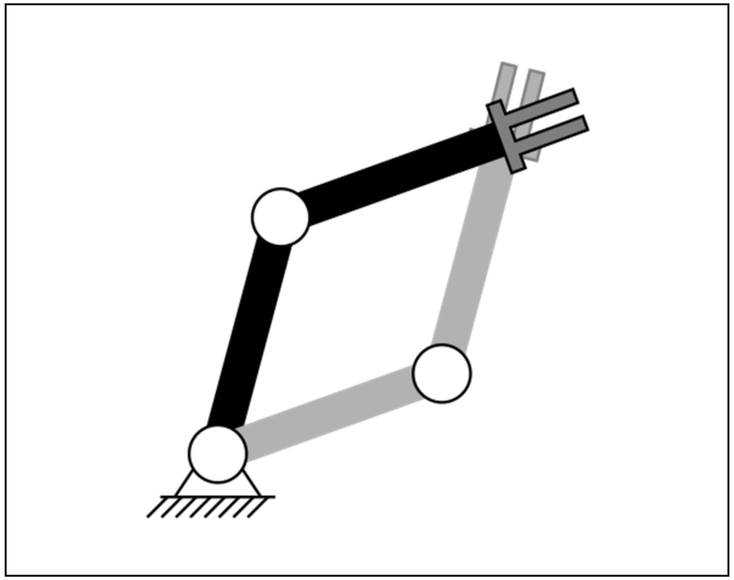
Pfeiffer / Reithmeier - Robot dynamics

Taskspace (Workspace)

End-effector pose $\boldsymbol{x}_\mathrm{E}$ clearly defined by joint positions $\boldsymbol{q}$

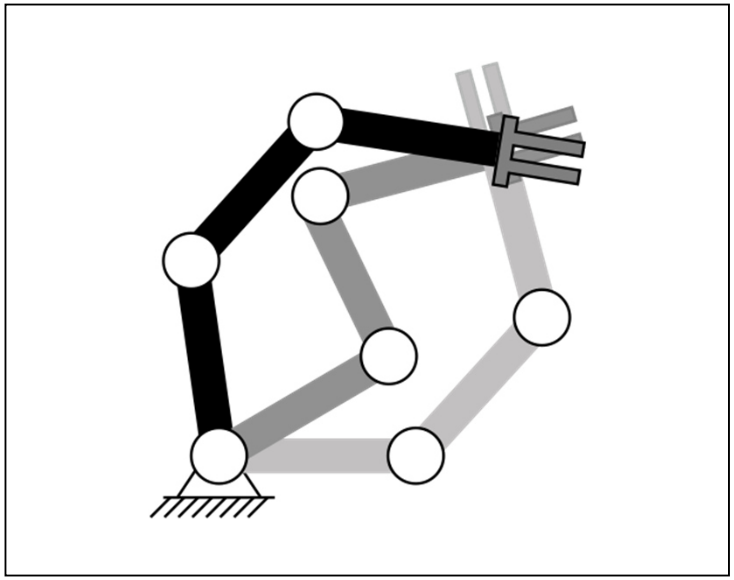
$$\mathcal{X} = \{\boldsymbol{x}_\mathrm{E} \mid \boldsymbol{x}_\mathrm{E} = \boldsymbol{f}(\boldsymbol{q}) \ \wedge\ \boldsymbol{q} \in \mathcal{Q}\}$$

Joint and Task Space - Example



Flat robot with two degrees of freedom

$$\dim(\boldsymbol{x}_\mathrm{E}) = 2,\ \dim(\boldsymbol{q}) = 2$$

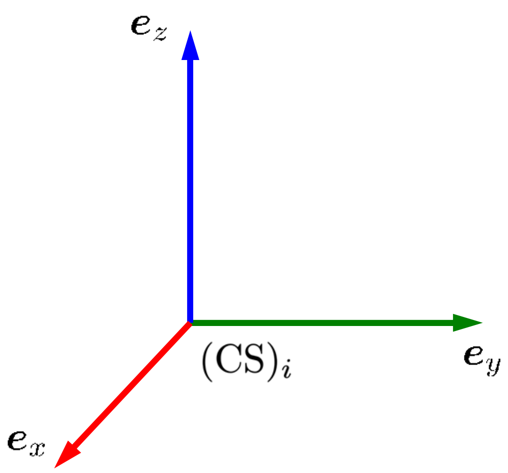


Flat robot with three degrees of freedom

$$\dim(\boldsymbol{x}_\mathrm{E}) = 2,\ \dim(\boldsymbol{q}) = 3$$

Coordinate Systems

- Coordinate system spans 3-dimensional geometric space
- Usually right-hand Cartesian coordinate system



Properties

$$\boldsymbol{e}_x = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \qquad \boldsymbol{e}_y = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \qquad \boldsymbol{e}_z = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

$$\|\boldsymbol{e}_x\|_2 = \| \boldsymbol{e}_y\|_2 = \| \boldsymbol{e}_z\|_2 = 1$$

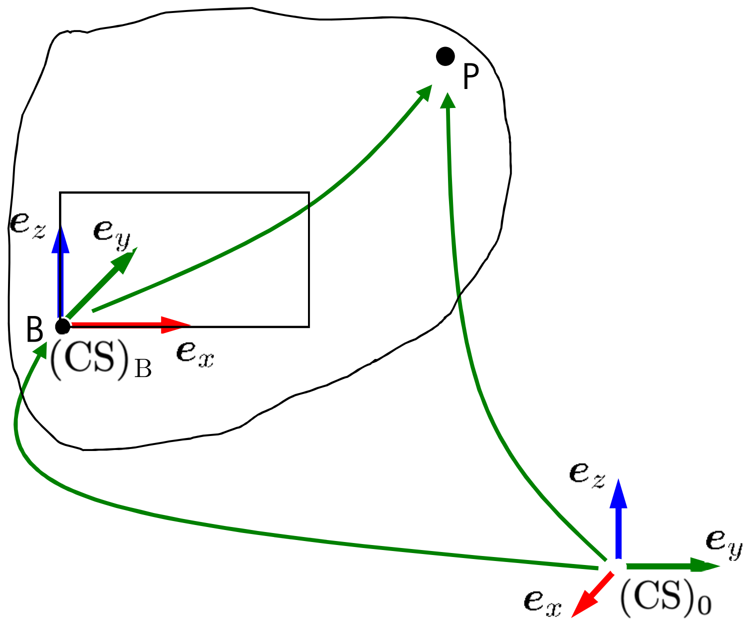
$$\boldsymbol{e}_x^{\mathrm{T}} \boldsymbol{e}_y = \boldsymbol{e}_x^{\mathrm{T}} \boldsymbol{e}_z = \boldsymbol{e}_y^{\mathrm{T}} \boldsymbol{e}_z = 0$$

$$\boldsymbol{e}_z = \boldsymbol{e}_x \times \boldsymbol{e}_y \quad (\det(\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z) = +1)$$

Coordinate Systems

Board notes

Classic coordinate systems in robotics



Inertial/world coordinate system $(\mathrm{CS})_0$
Fixed coordinate system
(typically placed in the robot base)

Body coordinate system $(\mathrm{CS})_\mathrm{B}$
Firmly attached to body, moves with it
(e.g. at the end effector or fixed to the joint)

Location (position and orientation) of an object in the room is usually described by a coordinate system.

Institute of Mechatronic Systems
imes Leibniz Universität Hannover

Leibniz Universität Hannover

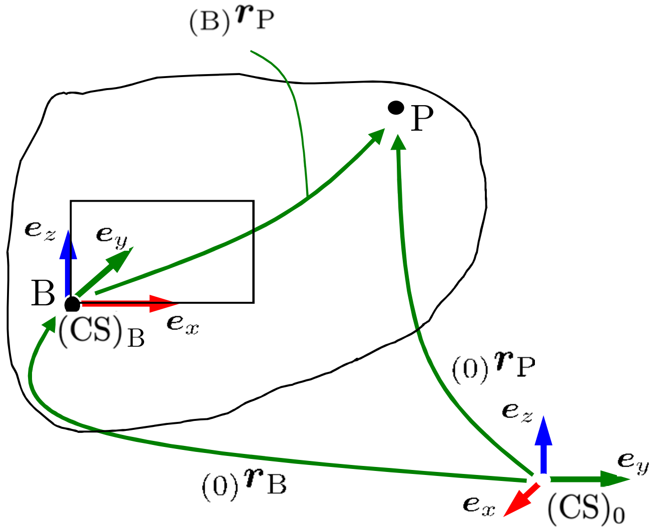
Robotics I

02. Coordinate Transformations (Rotations/Translations)

Position vectors

Position $(\mathrm{CS})_\mathrm{B}$ in $(\mathrm{CS})_0$ represented by

$${}_{(0)}\boldsymbol{r}_\mathrm{B} = \left({}_{(0)}x_\mathrm{B},\ {}_{(0)}y_\mathrm{B},\ {}_{(0)}z_\mathrm{B}\right)^\mathrm{T}$$



Position vector from $(\mathrm{CS})_\mathrm{B}$ to point P in $(\mathrm{CS})_\mathrm{B}$

$${}_{(\mathrm{B})}\boldsymbol{r}_\mathrm{P} = \left({}_{(\mathrm{B})}x_\mathrm{P},\ {}_{(\mathrm{B})}y_\mathrm{P},\ {}_{(\mathrm{B})}z_\mathrm{P}\right)^\mathrm{T}$$

Position vector from $(\mathrm{CS})_0$ to point P in $(\mathrm{CS})_0$

$${}_{(0)}\boldsymbol{r}_\mathrm{P} = {}_{(0)}\boldsymbol{e}_x^{(0)}\ {}_{(0)}x_\mathrm{P} + {}_{(0)}\boldsymbol{e}_y^{(0)}\ {}_{(0)}y_\mathrm{P} + {}_{(0)}\boldsymbol{e}_z^{(0)}\ {}_{(0)}z_\mathrm{P} = \underbrace{\left({}_{(0)}\boldsymbol{e}_x^{(0)},\ {}_{(0)}\boldsymbol{e}_y^{(0)},\ {}_{(0)}\boldsymbol{e}_z^{(0)}\right)}_{\boldsymbol{E}} \begin{pmatrix} {}_{(0)}x_\mathrm{P} \\ {}_{(0)}y_\mathrm{P} \\ {}_{(0)}z_\mathrm{P} \end{pmatrix}$$

Position vector goes from the origin of a $(\mathrm{CS})_i$ to a point P and is expressed in the same $(\mathrm{CS})_i$.

Coordinate Systems

Direction vectors

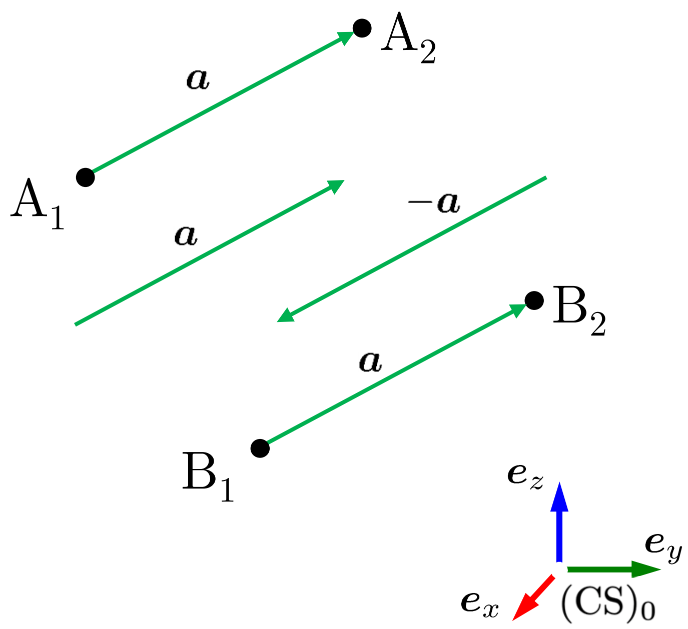
Direction vector from point A_1 to point A_2

$$\boldsymbol{a} = \overrightarrow{A_1\,A_2}$$

Points with the same displacement have the same direction vector

$$\boldsymbol{a} = \overrightarrow{A_1\,A_2} = \overrightarrow{B_1\,B_2}$$

Direction vector describes the displacement of two points in space (can be moved parallel in space)



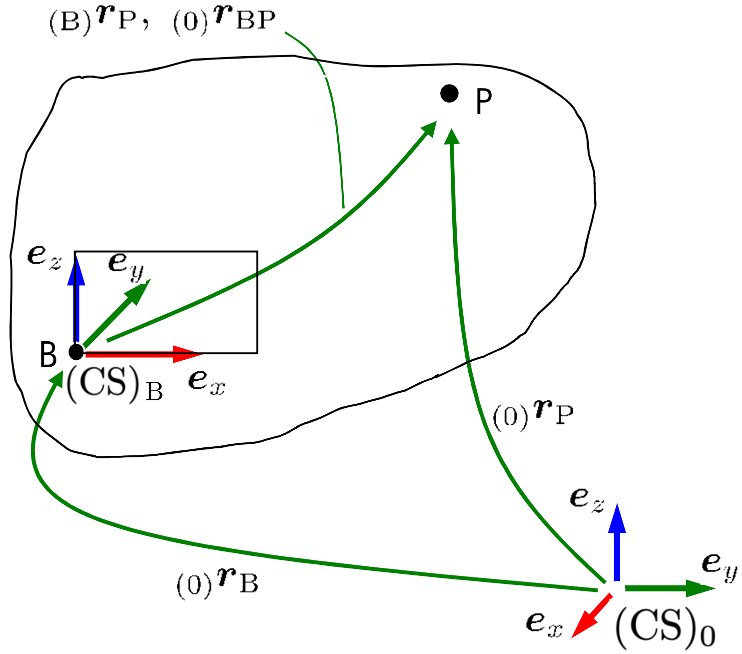
Coordinate Systems

Base transformation

Vector from the origin of $(\mathrm{CS})_\mathrm{B}$ to point P in $(\mathrm{CS})_0$

$$
{}_{(0)}\boldsymbol{r}_{\mathrm{BP}} = {}_{(0)}\boldsymbol{e}_x^{(\mathrm{B})}\,{}_{(\mathrm{B})}x_\mathrm{P} + {}_{(0)}\boldsymbol{e}_y^{(\mathrm{B})}\,{}_{(\mathrm{B})}y_\mathrm{P} + {}_{(0)}\boldsymbol{e}_z^{(\mathrm{B})}\,{}_{(\mathrm{B})}z_\mathrm{P}
$$

$$
= \underbrace{\left({}_{(0)}\boldsymbol{e}_x^{(\mathrm{B})},\ {}_{(0)}\boldsymbol{e}_y^{(\mathrm{B})},\ {}_{(0)}\boldsymbol{e}_z^{(\mathrm{B})}\right)}_{{}^{0}\boldsymbol{R}_\mathrm{B}} \begin{pmatrix} {}_{(\mathrm{B})}x_\mathrm{P} \\ {}_{(\mathrm{B})}y_\mathrm{P} \\ {}_{(\mathrm{B})}z_\mathrm{P} \end{pmatrix}
$$



Rotation matrix ${}^{0}\boldsymbol{R}_\mathrm{B}$ describes the mapping of the base coordinates of $(\mathrm{CS})_\mathrm{B}$ in $(\mathrm{CS})_0$ (explained in the following).

Robotics I

02. Coordinate Transformations (Rotations)

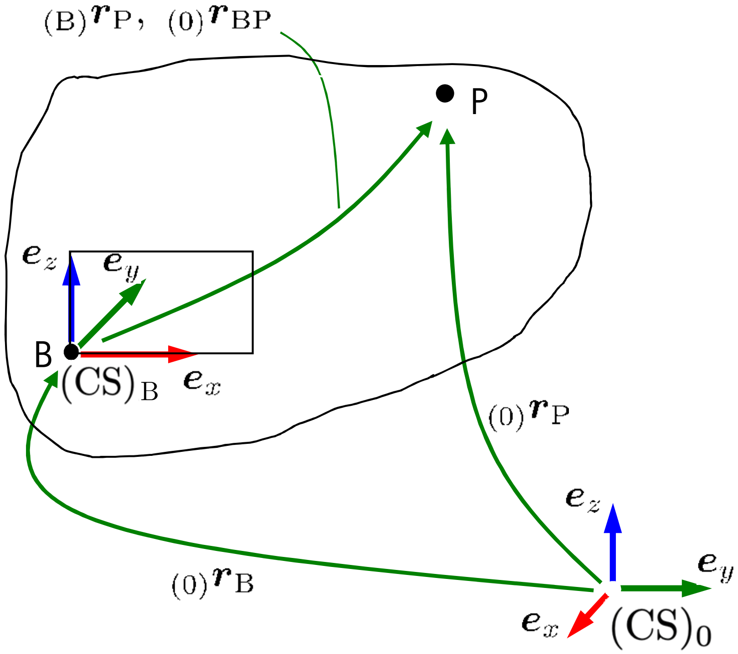
Coordinate Systems

Base transformation

Vector from the origin of $(\mathrm{CS})_\mathrm{B}$ to point P in $(\mathrm{CS})_0$

$$
\begin{aligned}
{}_{(0)}\boldsymbol{r}_\mathrm{BP} &= {}_{(0)}\boldsymbol{e}_x^{(\mathrm{B})}\,{}_{(\mathrm{B})}x_\mathrm{P} + {}_{(0)}\boldsymbol{e}_y^{(\mathrm{B})}\,{}_{(\mathrm{B})}y_\mathrm{P} + {}_{(0)}\boldsymbol{e}_z^{(\mathrm{B})}\,{}_{(\mathrm{B})}z_\mathrm{P} \\
&= \underbrace{\left({}_{(0)}\boldsymbol{e}_x^{(\mathrm{B})},\ {}_{(0)}\boldsymbol{e}_y^{(\mathrm{B})},\ {}_{(0)}\boldsymbol{e}_z^{(\mathrm{B})}\right)}_{{}^0\boldsymbol{R}_\mathrm{B}} \begin{pmatrix} {}_{(\mathrm{B})}x_\mathrm{P} \\ {}_{(\mathrm{B})}y_\mathrm{P} \\ {}_{(\mathrm{B})}z_\mathrm{P} \end{pmatrix}
\end{aligned}
$$

Rotation matrix ${}^0\boldsymbol{R}_\mathrm{B}$ describes the mapping of the base coordinates of $(\mathrm{CS})_\mathrm{B}$ in $(\mathrm{CS})_0$ (explained in the following).



Rotation Matrices

Board notes

Rotation matrix as basic transformation

$$ {}^{0}\boldsymbol{R}_{\mathrm{B}} = \left({}_{(0)}\boldsymbol{e}_x^{(\mathrm{B})},\ {}_{(0)}\boldsymbol{e}_y^{(\mathrm{B})},\ {}_{(0)}\boldsymbol{e}_z^{(\mathrm{B})} \right) \in \mathrm{SO}(3) $$

- Rotation matrices are orthogonal

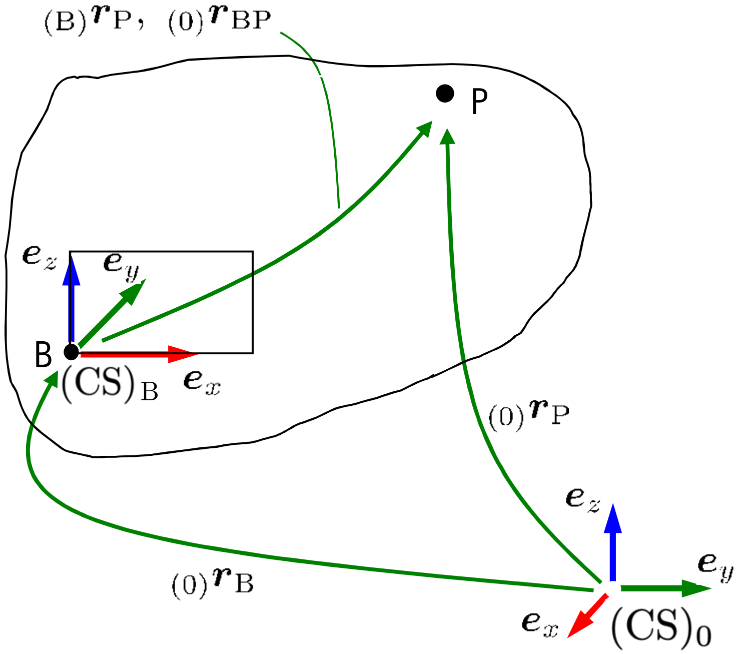
  $$ \left({}^{0}\boldsymbol{R}_{\mathrm{B}}\right)^{-1} = \left({}^{0}\boldsymbol{R}_{\mathrm{B}}\right)^{\mathrm{T}} = {}^{\mathrm{B}}\boldsymbol{R}_{0} $$

- Base vectors of a rotation matrix are orthonormal

  $$ \|{}_{(0)}\boldsymbol{e}_x^{(\mathrm{B})}\|_2 = \|{}_{(0)}\boldsymbol{e}_y^{(\mathrm{B})}\|_2 = \|{}_{(0)}\boldsymbol{e}_z^{(\mathrm{B})}\|_2 = 1 $$

  $$ \left({}_{(0)}\boldsymbol{e}_x^{(\mathrm{B})}\right)^{\mathrm{T}} {}_{(0)}\boldsymbol{e}_y^{(\mathrm{B})} = \left({}_{(0)}\boldsymbol{e}_x^{(\mathrm{B})}\right)^{\mathrm{T}} {}_{(0)}\boldsymbol{e}_z^{(\mathrm{B})} = \left({}_{(0)}\boldsymbol{e}_y^{(\mathrm{B})}\right)^{\mathrm{T}} {}_{(0)}\boldsymbol{e}_z^{(\mathrm{B})} = 0 $$

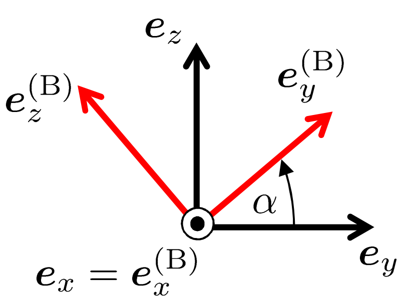
  $$ \det {}^{0}\boldsymbol{R}_{\mathrm{B}} = +1 $$

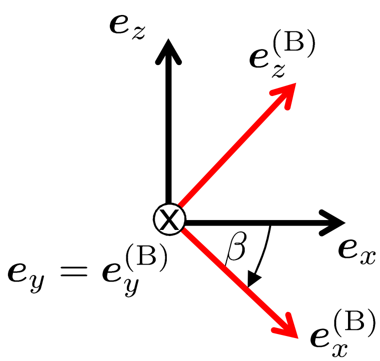


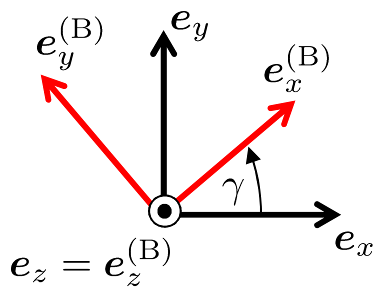
Based on these six constraints, three parameters are sufficient to unambiguously describe a general rotation

Elementary Rotations

Rotations around a single axis







Rotation around the x-axis $\boldsymbol{R}_x(\alpha)$

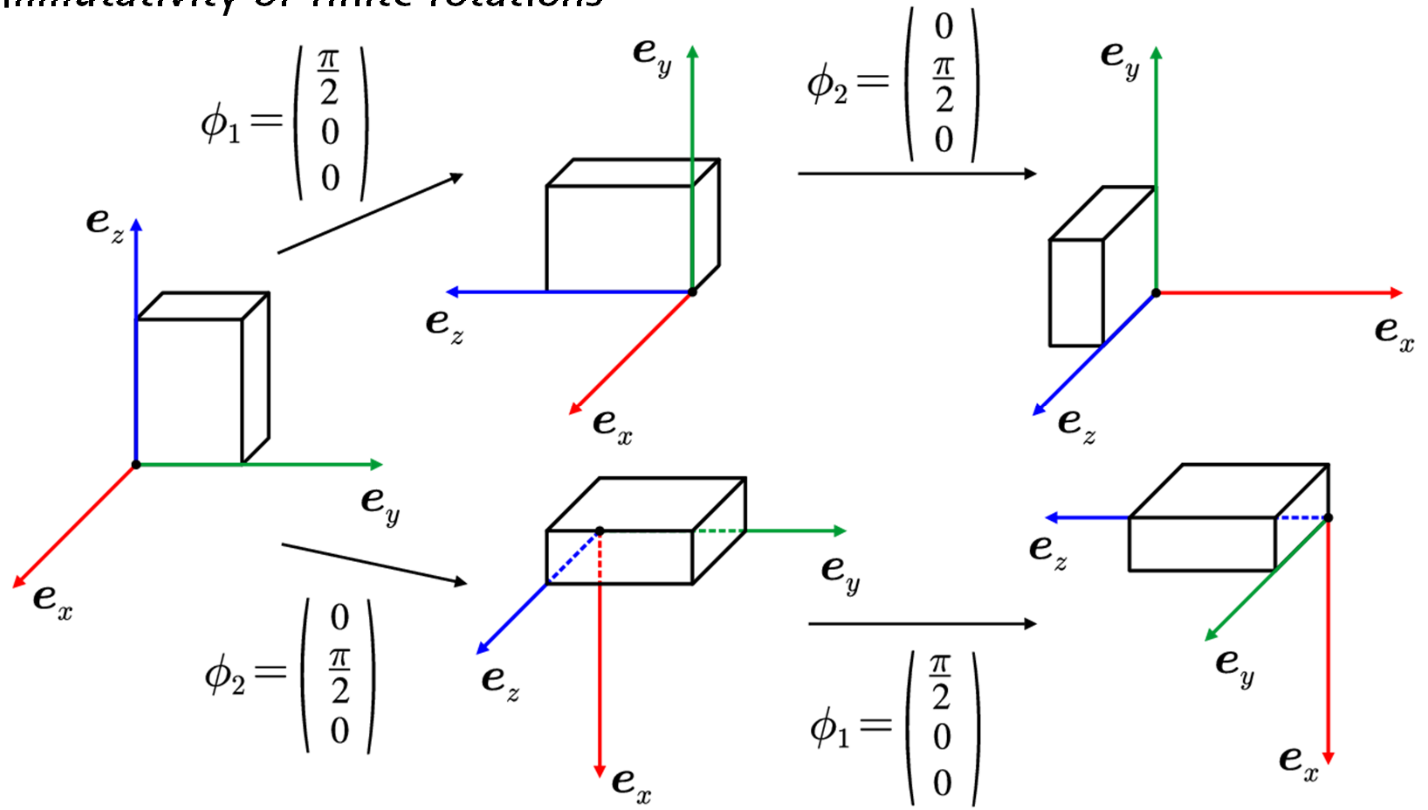
Rotation around the y-axis $\boldsymbol{R}_y(\beta)$

Rotation around the z-axis $\boldsymbol{R}_z(\gamma)$

$$\boldsymbol{R}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix} \quad \boldsymbol{R}_y(\beta) = \begin{pmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix} \quad \boldsymbol{R}_z(\gamma) = \begin{pmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Compound Rotations

Non-commutativity of finite rotations



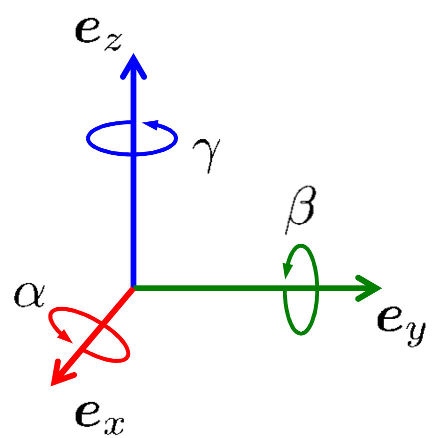
Compound Rotations

Board notes

Cardan angle

Rotations around the axes of the rotated system:

Results by right multiplication of the individual rotations.



1. Rotation around x-axis
2. Rotation around new y-axis
3. Rotation around latest z-axis

$$\boldsymbol{R}_{\mathrm{KARD}}(\alpha,\beta,\gamma) = \overbrace{\boldsymbol{R}_x(\alpha)}^{1.}\,\overbrace{\boldsymbol{R}_y(\beta)}^{2.}\,\overbrace{\boldsymbol{R}_z(\gamma)}^{3.} = \begin{pmatrix} \mathrm{c}_\beta\,\mathrm{c}_\gamma & -\mathrm{c}_\beta\,\mathrm{s}_\gamma & \mathrm{s}_\beta \\ \mathrm{c}_\alpha\,\mathrm{s}_\gamma + \mathrm{s}_\alpha\,\mathrm{s}_\beta\,\mathrm{c}_\gamma & \mathrm{c}_\alpha\,\mathrm{c}_\gamma - \mathrm{s}_\alpha\,\mathrm{s}_\beta\,\mathrm{s}_\gamma & -\mathrm{s}_\alpha\,\mathrm{c}_\beta \\ \mathrm{s}_\alpha\,\mathrm{s}_\gamma - \mathrm{c}_\alpha\,\mathrm{s}_\beta\,\mathrm{c}_\gamma & \mathrm{s}_\alpha\,\mathrm{c}_\gamma + \mathrm{c}_\alpha\,\mathrm{s}_\beta\,\mathrm{s}_\gamma & \mathrm{c}_\alpha\,\mathrm{c}_\beta \end{pmatrix}$$

Right multiplication

Institute of
Mechatronic Systems
imes Leibniz Universität Hannover

Leibniz
Universität
Hannover

Robotics I

02. Coordinate Transformations

Compound Rotations

Euler angle

Rotations around the axes of the rotated system:

Results by right multiplication of the individual rotations.

Of a total of twelve possible descriptions, three have been established:

- x convention: $\boldsymbol{R}_{\mathrm{EUL,x}}(\gamma, \alpha, \gamma') = \overbrace{\boldsymbol{R}_z(\gamma)}^{1.}\ \overbrace{\boldsymbol{R}_x(\alpha)}^{2.}\ \overbrace{\boldsymbol{R}_z(\gamma')}^{3.}$
- y convention: $\boldsymbol{R}_{\mathrm{EUL,y}}(\gamma, \beta, \gamma') = \boldsymbol{R}_z(\gamma)\, \boldsymbol{R}_y(\beta)\, \boldsymbol{R}_z(\gamma')$
- Aviation standard: $\boldsymbol{R}_{\mathrm{EUL,LF}}(\gamma, \beta, \alpha) = \boldsymbol{R}_z(\gamma)\, \boldsymbol{R}_y(\beta)\, \boldsymbol{R}_x(\alpha)$

Right multiplication

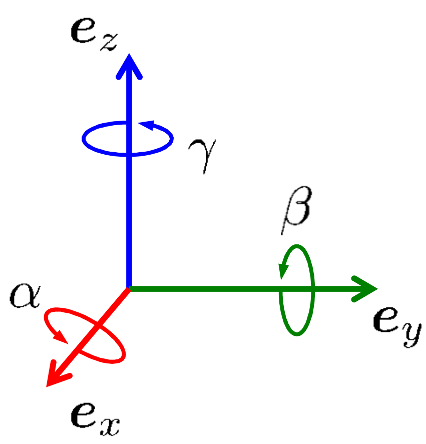
Compound Rotations

RPY angle

Rotations around the axes of the base system:

Results by <u>left multiplication</u> of the individual elementary rotations.

1. Rotation around x-axis (yaw)
2. Rotation around y-axis (pitch)
3. Rotation around z-axis (roll)



$$\boldsymbol{R}_{\mathrm{RPY}}(\alpha,\beta,\gamma) = \underbrace{\overbrace{\boldsymbol{R}_z(\gamma)}^{3.}\;\overbrace{\boldsymbol{R}_y(\beta)}^{2.}\;\overbrace{\boldsymbol{R}_x(\alpha)}^{1.}}_{\xleftarrow{\text{Left multiplication}}} = \begin{pmatrix} \mathrm{c}_\gamma\,\mathrm{c}_\beta & \mathrm{c}_\gamma\,\mathrm{s}_\beta\,\mathrm{s}_\alpha - \mathrm{s}_\gamma\,\mathrm{c}_\alpha & \mathrm{c}_\gamma\,\mathrm{s}_\beta\,\mathrm{c}_\alpha + \mathrm{s}_\gamma\,\mathrm{s}_\alpha \\ \mathrm{s}_\gamma\,\mathrm{c}_\beta & \mathrm{s}_\gamma\,\mathrm{s}_\beta\,\mathrm{s}_\alpha + \mathrm{c}_\gamma\,\mathrm{c}_\alpha & \mathrm{s}_\gamma\,\mathrm{s}_\beta\,\mathrm{c}_\alpha - \mathrm{c}_\gamma\,\mathrm{s}_\alpha \\ -\mathrm{s}_\beta & \mathrm{c}_\beta\,\mathrm{s}_\alpha & \mathrm{c}_\beta\,\mathrm{c}_\alpha \end{pmatrix}$$

Compound Rotations

Direct correlation

From given rotation angles to the rotation matrix, example RPY angle

Rotation angle $(\alpha, \beta, \gamma)^{\mathrm{T}}$ → Direct correlation $\boldsymbol{R}_z(\gamma)\,\boldsymbol{R}_y(\beta)\,\boldsymbol{R}_x(\alpha)$ → Rotation matrix $\boldsymbol{R}_{\mathrm{RPY}}(\alpha, \beta, \gamma)$

Inverse correlation

From a given rotation matrix to the rotation angles, example RPY angle

Rotation matrix $\boldsymbol{R}_{\mathrm{RPY}}(\alpha, \beta, \gamma)$ → Inverse correlation (non-linear GS) → Rotation angle $(\alpha, \beta, \gamma)^{\mathrm{T}}$

Compound Rotations

$$\mathrm{c}_i = \cos(i)$$
$$\mathrm{s}_i = \sin(i)$$
$$\mathrm{c}_{ij} = \cos(i+j)$$
$$\mathrm{s}_{ij} = \sin(i+j)$$

$$\boldsymbol{R}_{\mathrm{RPY}}(\alpha,\beta,\gamma) = \overbrace{\boldsymbol{R}_z(\gamma)}^{3.}\,\overbrace{\boldsymbol{R}_y(\beta)}^{2.}\,\overbrace{\boldsymbol{R}_x(\alpha)}^{1.} = \begin{pmatrix} \mathrm{c}_\gamma\,\mathrm{c}_\beta & \mathrm{c}_\gamma\,\mathrm{s}_\beta\,\mathrm{s}_\alpha - \mathrm{s}_\gamma\,\mathrm{c}_\alpha & \mathrm{c}_\gamma\,\mathrm{s}_\beta\,\mathrm{c}_\alpha + \mathrm{s}_\gamma\,\mathrm{s}_\alpha \\ \mathrm{s}_\gamma\,\mathrm{c}_\beta & \mathrm{s}_\gamma\,\mathrm{s}_\beta\,\mathrm{s}_\alpha + \mathrm{c}_\gamma\,\mathrm{c}_\alpha & \mathrm{s}_\gamma\,\mathrm{s}_\beta\,\mathrm{c}_\alpha - \mathrm{c}_\gamma\,\mathrm{s}_\alpha \\ -\mathrm{s}_\beta & \mathrm{c}_\beta\,\mathrm{s}_\alpha & \mathrm{c}_\beta\,\mathrm{c}_\alpha \end{pmatrix}$$

$$\Rightarrow\ \alpha = \operatorname{arctan2}\left(r_{\mathrm{RPY},(3,2)}, r_{\mathrm{RPY},(3,3)}\right),\ \beta = \operatorname{arctan2}\left(-r_{\mathrm{RPY},(3,1)}, r_{\mathrm{RPY},(3,2)}\,\mathrm{s}_\alpha + r_{\mathrm{RPY},(3,3)}\,\mathrm{c}_\alpha\right),$$
$$\gamma = \operatorname{arctan2}\left(r_{\mathrm{RPY},(2,1)}, r_{\mathrm{RPY},(1,1)}\right)$$

$r_{\mathrm{RPY},(\mathrm{i},\mathrm{j})}$: i-th element of j-th column of $\boldsymbol{R}_{\mathrm{RPY}}$

Compound Rotations

Summary

Advantages

- simple notation
- intuitively comprehensible
- few parameters

Disadvantages

- different conventions under the same name (e.g. Euler angle: twelve possibilities)
- order of multiplication important
- interpolation leads to unforeseen paths
- Gimbal-Lock problem

Robotics I

02. Coordinate Transformations (Rotations/Translations)

Description of an Object in the Room

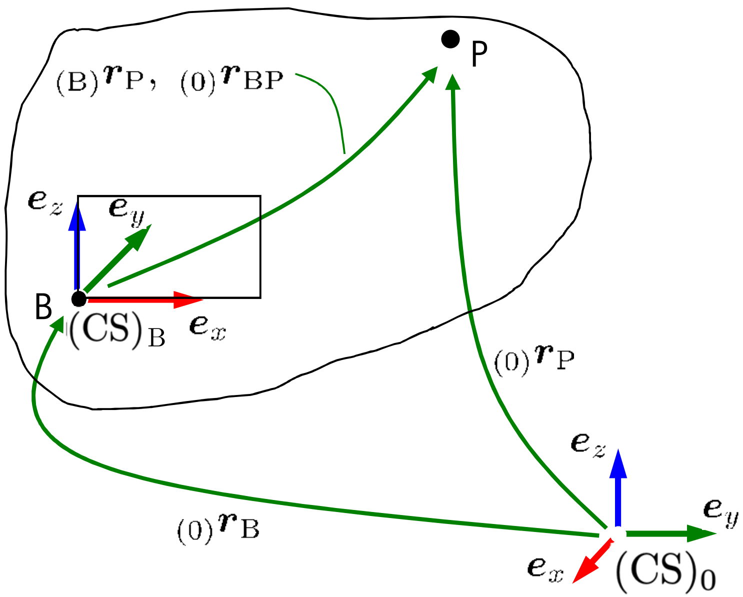
Merging orientation and position

Mapping $(\mathrm{CS})_\mathrm{B}$ → $(\mathrm{CS})_0$

$$ {}_{(0)}\boldsymbol{r}_\mathrm{P} = {}_{(0)}\boldsymbol{r}_\mathrm{B} + {}^{0}\boldsymbol{R}_\mathrm{B}\, {}_{(\mathrm{B})}\boldsymbol{r}_\mathrm{P} $$

Rotation

Translation



Mapping $(\mathrm{CS})_0$ → $(\mathrm{CS})_\mathrm{B}$

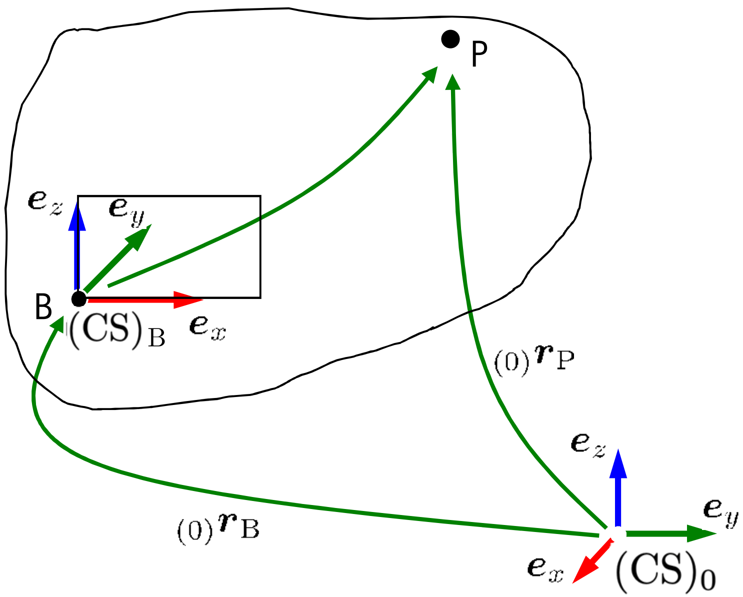
$$ \begin{aligned} {}_{(\mathrm{B})}\boldsymbol{r}_\mathrm{P} &= -\left({}^{0}\boldsymbol{R}_\mathrm{B}\right)^\mathrm{T} {}_{(0)}\boldsymbol{r}_\mathrm{B} + \left({}^{0}\boldsymbol{R}_\mathrm{B}\right)^\mathrm{T} {}_{(0)}\boldsymbol{r}_\mathrm{P} \\ &= -\,{}^{\mathrm{B}}\boldsymbol{R}_0\, {}_{(0)}\boldsymbol{r}_\mathrm{B} + {}^{\mathrm{B}}\boldsymbol{R}_0\, {}_{(0)}\boldsymbol{r}_\mathrm{P} \end{aligned} $$

Homogeneous Transformation

Compact representation using (4x4) matrices
and (4x1) homogeneous position vectors:
Uniform treatment of translation
and rotation through homogeneous expansion

$$ {}^{0}\boldsymbol{T}_{\mathrm{B}} = \left(\begin{array}{ccc|c} & {}^{0}\boldsymbol{R}_{\mathrm{B}} & & {}_{(0)}\boldsymbol{r}_{\mathrm{B}} \\ \hline 0 & 0 & 0 & 1 \end{array} \right) $$

$$ {}^{0}\boldsymbol{T}_{\mathrm{B}} \in \mathrm{SE}(3) = \{\mathrm{SO}(3) \times \mathbb{R}^3\} $$



Properties/special features:

- $\det {}^{0}\boldsymbol{T}_{\mathrm{B}} = 1$
- $\left({}^{0}\boldsymbol{T}_{\mathrm{B}}\right)^{-1} = {}^{\mathrm{B}}\boldsymbol{T}_{0} = \left(\begin{array}{ccc|c} & \left({}^{0}\boldsymbol{R}_{\mathrm{B}}\right)^{\mathrm{T}} & & -\left({}^{0}\boldsymbol{R}_{\mathrm{B}}\right)^{\mathrm{T}} {}_{(0)}\boldsymbol{r}_{\mathrm{B}} \\ \hline 0 & 0 & 0 & 1 \end{array} \right)$

Homogeneous Transformation

Transformation of points from a coordinate system into another (transformation of homogeneous position vectors)

Point P given in $(\mathrm{CS})_0$:

$$_{(0)}\boldsymbol{x}_\mathrm{P} = \left({}_{(0)}\boldsymbol{r}_\mathrm{P}^\mathrm{T},\ 1\right)^\mathrm{T} = \left({}_{(0)}x_\mathrm{P},\ {}_{(0)}y_\mathrm{P},\ {}_{(0)}z_\mathrm{P},\ 1\right)^\mathrm{T}$$

Point P given in $(\mathrm{CS})_\mathrm{B}$:

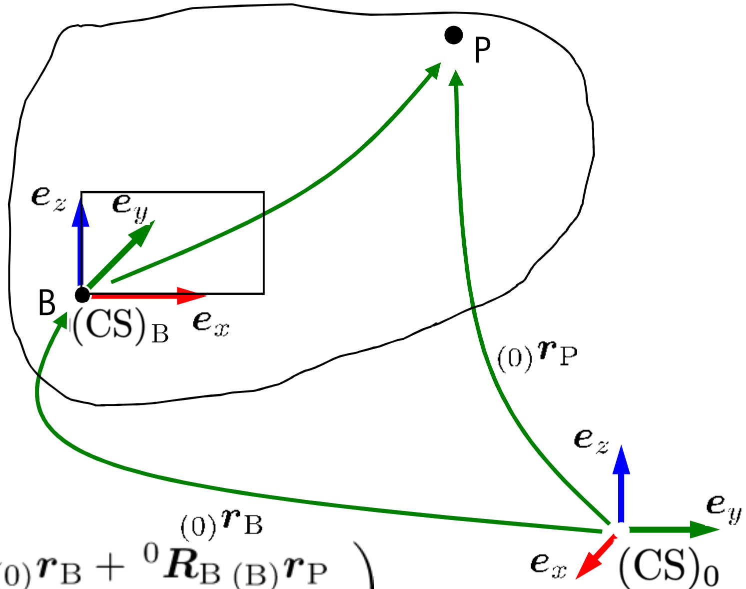
$$_{(\mathrm{B})}\boldsymbol{x}_\mathrm{P} = \left({}_{(\mathrm{B})}\boldsymbol{r}_\mathrm{P}^\mathrm{T},\ 1\right)^\mathrm{T} = \left({}_{(\mathrm{B})}x_\mathrm{P},\ {}_{(\mathrm{B})}y_\mathrm{P},\ {}_{(\mathrm{B})}z_\mathrm{P},\ 1\right)^\mathrm{T}$$

$$_{(0)}\boldsymbol{x}_\mathrm{P} = {}^0\boldsymbol{T}_\mathrm{B}\ {}_{(\mathrm{B})}\boldsymbol{x}_\mathrm{P} = \left(\begin{array}{ccc|c} & {}^0\boldsymbol{R}_\mathrm{B} & & {}_{(0)}\boldsymbol{r}_\mathrm{B} \\ \hline 0 & 0 & 0 & 1 \end{array}\right) \begin{pmatrix} {}_{(\mathrm{B})}\boldsymbol{r}_\mathrm{P} \\ 1 \end{pmatrix} = \begin{pmatrix} {}_{(0)}\boldsymbol{r}_\mathrm{B} + {}^0\boldsymbol{R}_\mathrm{B}\ {}_{(\mathrm{B})}\boldsymbol{r}_\mathrm{P} \\ 1 \end{pmatrix}$$

$$_{(0)}\boldsymbol{r}_\mathrm{P} = {}_{(0)}\boldsymbol{r}_\mathrm{B} + {}^0\boldsymbol{R}_\mathrm{B}\ {}_{(\mathrm{B})}\boldsymbol{r}_\mathrm{P}$$

$$_{(\mathrm{B})}\boldsymbol{x}_\mathrm{P} = {}^\mathrm{B}\boldsymbol{T}_0\ {}_{(0)}\boldsymbol{x}_\mathrm{P} = \left({}^0\boldsymbol{T}_\mathrm{B}\right)^{-1}\ {}_{(0)}\boldsymbol{x}_\mathrm{P}$$

$$_{(\mathrm{B})}\boldsymbol{r}_\mathrm{P} = -\left({}^0\boldsymbol{R}_\mathrm{B}\right)^\mathrm{T}{}_{(0)}\boldsymbol{r}_\mathrm{B} + \left({}^0\boldsymbol{R}_\mathrm{B}\right)^\mathrm{T}{}_{(0)}\boldsymbol{r}_\mathrm{P}$$



Homogeneous Transformation

Transformation of direction vectors from a coordinate system to another

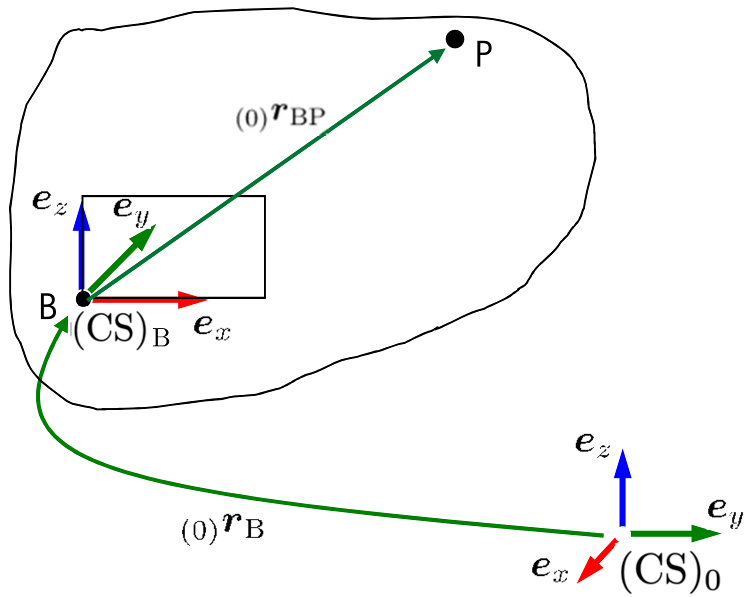
Direction vector ${}_{(0)}\boldsymbol{r}_{\mathrm{BP}}$ given in $(\mathrm{CS})_0$

$${}_{(0)}\boldsymbol{x}_{\mathrm{BP}} = \left({}_{(0)}\boldsymbol{r}_{\mathrm{BP}}^{\mathrm{T}},\ 0\right)^{\mathrm{T}}$$

Transformation to $(\mathrm{CS})_{\mathrm{B}}$ using ${}^{\mathrm{B}}\boldsymbol{T}_0$

$${}_{(\mathrm{B})}\boldsymbol{x}_{\mathrm{BP}} = {}^{\mathrm{B}}\boldsymbol{T}_0\, {}_{(0)}\boldsymbol{x}_{\mathrm{BP}} = \left({}^{0}\boldsymbol{T}_{\mathrm{B}}\right)^{-1} {}_{(0)}\boldsymbol{x}_{\mathrm{BP}}$$
$$= \begin{pmatrix} {}^{\mathrm{B}}\boldsymbol{R}_0\, {}_{(0)}\boldsymbol{r}_{\mathrm{BP}} \\ 0 \end{pmatrix}$$

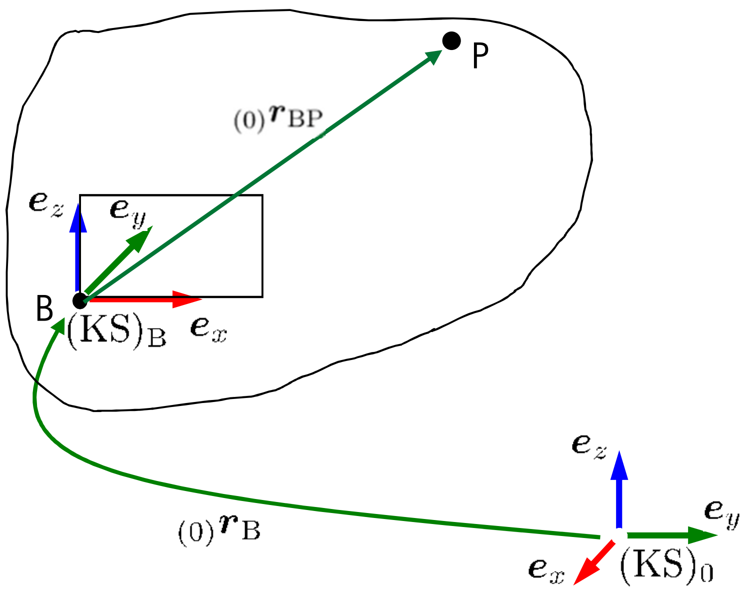
Relative shift ${}_{(0)}\boldsymbol{r}_{\mathrm{B}}$ of $(\mathrm{CS})_0$ and $(\mathrm{CS})_{\mathrm{B}}$ does not matter



Homogeneous Transformation – Code Example

```matlab
% Parameters for the gimbal angle representation
alpha = 30;    % Rotation around x [°]
beta  = 45;    % Rotation around new [°]
gamma = 60;    % Rotation around latest z [°]
t = [-1; -4; 3]; % Translation KS_0 -> KS_B
% Origin point in KS_0 (homogeneous coordinates)
p0 = [-2; -1; 7; 1];

%% Calculate rotation
a = deg2rad(alpha);
b = deg2rad(beta);
g = deg2rad(gamma);

Rx = [1 0       0;
      0 cos(a) -sin(a);
      0 sin(a)  cos(a)]

Ry = [ cos(b) 0 sin(b);
         0    1   0;
      -sin(b) 0 cos(b)]

Rz = [cos(g) -sin(g) 0;
      sin(g)  cos(g) 0;
        0       0    1]
R = Rx * Ry * Rz
%Homogeneous transformation matrix
T_0_B = [R    ,t;
         0 0 0 1]
%% Transform point from KS_0 to KS_B
pB = inv(T_0_B) * p0       % Point in KS_B
```

```
Rx = 3x3
    1.0000         0         0
         0    0.8660   -0.5000
         0    0.5000    0.8660

Ry = 3x3
    0.7071         0    0.7071
         0    1.0000         0
   -0.7071         0    0.7071

Rz = 3x3
    0.5000   -0.8660         0
    0.8660    0.5000         0
         0         0    1.0000

R = 3x3
    0.3536   -0.6124    0.7071
    0.9268    0.1268   -0.3536
    0.1268    0.7803    0.6124

T_0_A = 4x4
    0.3536   -0.6124    0.7071   -1.0000
    0.9268    0.1268   -0.3536   -4.0000
    0.1268    0.7803    0.6124    3.0000
         0         0         0    1.0000

pA = 4x1
    2.9341
    4.1142
    0.6817
    1.0000
```

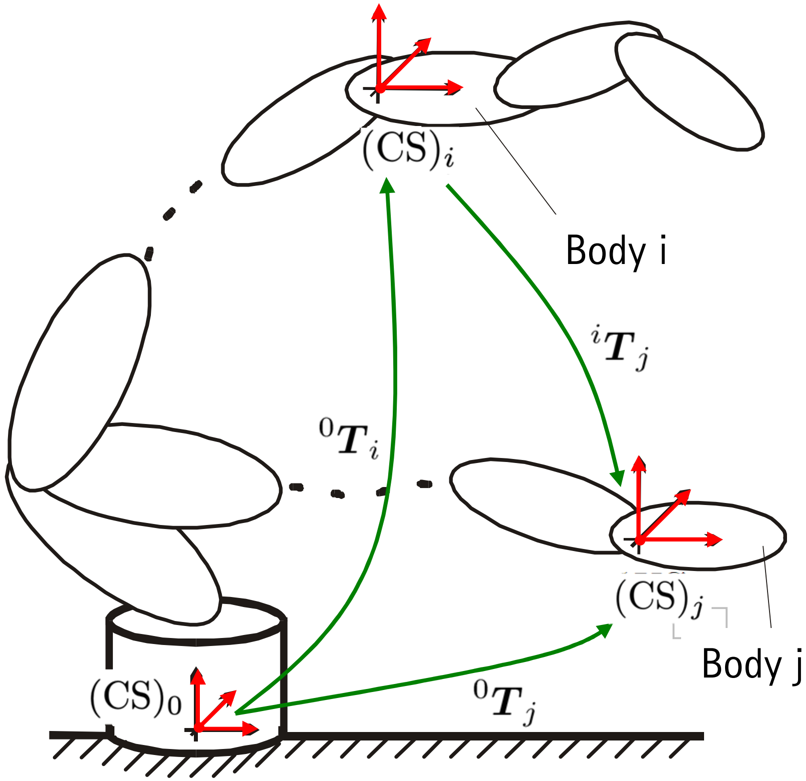


$$\boldsymbol{R}_{\mathrm{KARD}}(\alpha,\beta,\gamma) = \overbrace{\boldsymbol{R}_x(\alpha)}^{1.}\ \overbrace{\boldsymbol{R}_y(\beta)}^{2.}\ \overbrace{\boldsymbol{R}_z(\gamma)}^{3.}$$

Multiplication to the right

Homogeneous Transformation

Concatenation of transformations



Concatenation of homogeneous transformations

$$ {}_{(0)}\boldsymbol{x} = \underbrace{{}^0\boldsymbol{T}_1\,{}^1\boldsymbol{T}_2 \cdots {}^{n-2}\boldsymbol{T}_{n-1}\,{}^{n-1}\boldsymbol{T}_n}_{{}^0\boldsymbol{T}_n}\,{}_{(n)}\boldsymbol{x} $$

Transformation from $(\mathrm{CS})_j$ to $(\mathrm{CS})_i$

$$ {}^0\boldsymbol{T}_j = {}^0\boldsymbol{T}_i\,{}^i\boldsymbol{T}_j $$

$$ \Rightarrow\ {}^i\boldsymbol{T}_j = \left({}^0\boldsymbol{T}_i\right)^{-1}\,{}^0\boldsymbol{T}_j = {}^i\boldsymbol{T}_0\,{}^0\boldsymbol{T}_j $$

Recommendation
Retain notation during programming

Rotation around an Axis

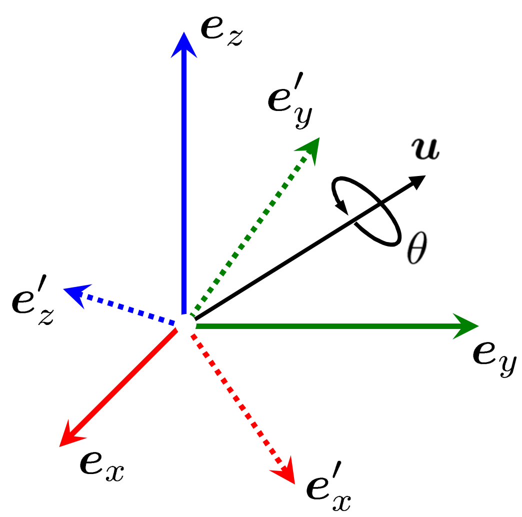
Axis-angle representation

Rotation with three angles can be equivalently represented by

Rotation around an axis $\boldsymbol{u}$ can be represented by the angle θ

The following applies: $||\boldsymbol{u}||_2 = 1$

Three parameters for the axis display $\boldsymbol{u} = (u_x,\, u_y,\, u_z)^{\mathrm{T}}$



Conversion from axis-angle representation to equivalent

Rotation matrix ($\mathrm{c}_\theta = \cos(\theta),\ \mathrm{s}_\theta = \sin(\theta)$):

$$\boldsymbol{R}_{\mathrm{u}} = \begin{pmatrix} u_x^2\,(1-\mathrm{c}_\theta)+\mathrm{c}_\theta & u_x\,u_y\,(1-\mathrm{c}_\theta)-u_z\,\mathrm{s}_\theta & u_x\,u_z\,(1-\mathrm{c}_\theta)+u_y\,\mathrm{s}_\theta \\ u_x\,u_y\,(1-\mathrm{c}_\theta)+u_z\,\mathrm{s}_\theta & u_y^2\,(1-\mathrm{c}_\theta)+\mathrm{c}_\theta & u_y\,u_z\,(1-\mathrm{c}_\theta)-u_x\,\mathrm{s}_\theta \\ u_x\,u_z\,(1-\mathrm{c}_\theta)-u_y\,\mathrm{s}_\theta & u_y\,u_z\,(1-\mathrm{c}_\theta)+u_x\,\mathrm{s}_\theta & u_z^2\,(1-\mathrm{c}_\theta)+\mathrm{c}_\theta \end{pmatrix}$$

Determination of $\boldsymbol{u}$ and θ from a given rotation matrix possible (see script).

Rotation around an Axis

Summary

Advantages

- few parameters
- calculations/conversions possible
- well suited for the interpolation of rotations
- comparatively intuitive display
- no singularities

Disadvantages

- concatenation of rotations not possible

Quaternions

Mathematical construct introduced by Hamilton in 1843

Extension of complex numbers

$$\boldsymbol{q} = q_0 + q_1\,\mathrm{i} + q_2\,\mathrm{j} + q_3\,\mathrm{k}$$

$$q_i \in \mathbb{R}, \quad \mathrm{i}^2 = \mathrm{j}^2 = \mathrm{k}^2 = \mathrm{i}\,\mathrm{j}\,\mathrm{k} = -1$$

Composed of scalar s and vector part $\boldsymbol{v}$

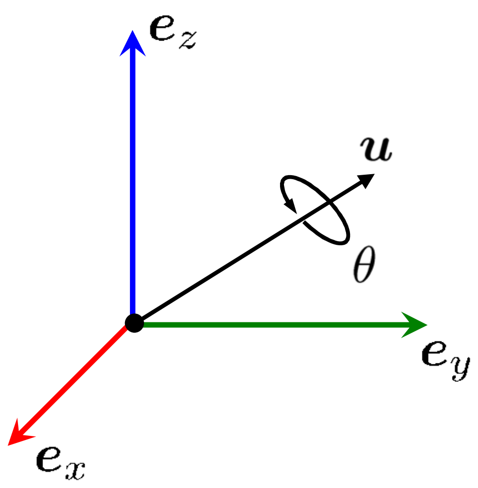
$$\boldsymbol{q} = (s,\, \boldsymbol{v}^{\mathrm{T}})^{\mathrm{T}}, \quad \text{mit } s = q_0,\, \boldsymbol{v} = (q_1,\, q_2,\, q_3)^{\mathrm{T}}$$

Definitions

- Adjuncts $\overline{\boldsymbol{q}} = (s,\, -\boldsymbol{v}^{\mathrm{T}})^{\mathrm{T}}$
- Inverse $\boldsymbol{q}^{-1} = \dfrac{\overline{\boldsymbol{q}}}{||\boldsymbol{q}||_2}$
- neutral quaternion $\boldsymbol{q}_{\mathrm{E}} = (1,\, (0,\, 0,\, 0))^{\mathrm{T}}$

Quaternions

Orientation is expressed using four parameters: Describe a rotation with the angle θ $(0 \leq \theta \leq \pi)$
around an axis with the unit vector $\boldsymbol{u}$



Based on the axis-angle representation:
Use of the unit quaternion with $||\boldsymbol{q}||_2 = 1$
to describe a rotation using

$$\boldsymbol{q} = (s,\, \boldsymbol{v}^{\mathrm{T}})^{\mathrm{T}} \qquad s = q_0 = \cos\left(\frac{\theta}{2}\right)$$

$$\boldsymbol{v} = \boldsymbol{u}\,\sin\left(\frac{\theta}{2}\right)$$

Quaternions

Conversion from quaternion to rotation matrix

$$\boldsymbol{R} = \begin{pmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2q_1q_2-2q_0q_3 & 2q_1q_3+2q_0q_2 \\ 2q_1q_2+2q_0q_3 & q_0^2-q_1^2+q_2^2-q_3^2 & 2q_2q_3-2q_0q_1 \\ 2q_1q_3-2q_0q_2 & 2q_2q_3+2q_0q_1 & q_0^2-q_1^2-q_2^2+q_3^2 \end{pmatrix}$$

Multiplication of two rotation matrices
(corresponds to the product of the corresponding quaternions)

$${}^{i}\boldsymbol{R}_j \mathrel{\hat{=}} \boldsymbol{q}_{ij}$$

$${}^{0}\boldsymbol{R}_1 \, {}^{1}\boldsymbol{R}_2 \, \cdots \, {}^{n-1}\boldsymbol{R}_n \mathrel{\hat{=}} \boldsymbol{q}_{01} \, \boldsymbol{q}_{12} \, \cdots \, \boldsymbol{q}_{n-1,n}$$

Quaternion multiplication

$$\begin{aligned} \boldsymbol{q}_3 = \boldsymbol{q}_1 \, \boldsymbol{q}_2 &= (s_1 \, s_2 - \boldsymbol{v}_1^{\mathrm{T}} \boldsymbol{v}_2, \; s_1 \boldsymbol{v}_2 + s_2 \boldsymbol{v}_1 + \boldsymbol{v}_1 \times \boldsymbol{v}_2)^{\mathrm{T}} \\ &= (s_3, \; \boldsymbol{v}_3^{\mathrm{T}})^{\mathrm{T}} \end{aligned}$$

Concept of dual quaternions (analogous to homogeneous transformation)

Quaternions

Summary

Advantages

- few parameters
- calculations/conversions possible
- compounding several rotations etc.
- complex interpolations also possible (SLERP)
- no singularities
- calculation more effective than rotation matrices

Disadvantages

- No intuitive description

Questions for self-monitoring

Coordinate transformation

How do I know where the robot tip (end effector) is located?

1. What options are you aware of for describing rotations? Explain the differences and name the advantages and disadvantages of each approach!
2. Name three types of compound rotations and state the rotation rules for each as a function of elementary rotations!
3. From the given total rotation matrix $\boldsymbol{R}$, determine the rotation angles around the individual axes of rotation using Kardan angles!

$$\boldsymbol{R}_{\mathrm{KARD}}(\alpha, \beta, \gamma) = \begin{pmatrix} \mathrm{c}_\beta \, \mathrm{c}_\gamma & -\mathrm{c}_\beta \, \mathrm{s}_\gamma & \mathrm{s}_\beta \\ \mathrm{c}_\alpha \, \mathrm{s}_\gamma + \mathrm{s}_\alpha \, \mathrm{s}_\beta \, \mathrm{c}_\gamma & \mathrm{c}_\alpha \, \mathrm{c}_\gamma - \mathrm{s}_\alpha \, \mathrm{s}_\beta \, \mathrm{s}_\gamma & -\mathrm{s}_\alpha \, \mathrm{c}_\beta \\ \mathrm{s}_\alpha \, \mathrm{s}_\gamma - \mathrm{c}_\alpha \, \mathrm{s}_\beta \, \mathrm{c}_\gamma & \mathrm{s}_\alpha \, \mathrm{c}_\gamma + \mathrm{c}_\alpha \, \mathrm{s}_\beta \, \mathrm{s}_\gamma & \mathrm{c}_\alpha \, \mathrm{c}_\beta \end{pmatrix}$$