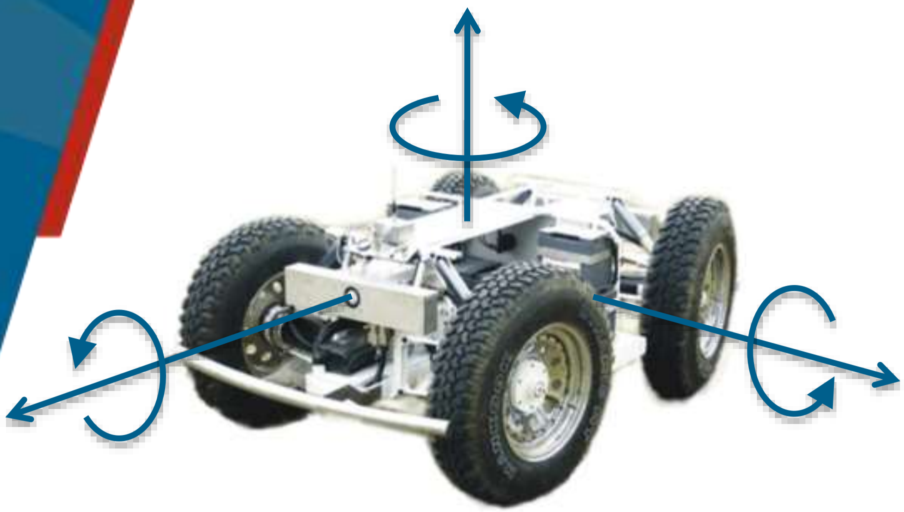
Spatial Kinematics – Foundations I



Prof. Dr. Karsten Berns
Robotics Research Lab
Department of Computer Science
University of Kaiserslautern, Germany

Content

- Description object poses in 3D Euclidean space (E_3)
- Description of orientation with 3x3 matrices
- 6-dim. description vector
- Homogenous coordinates and transformation matrix
- Sequences of pose descriptors and their relation to each other

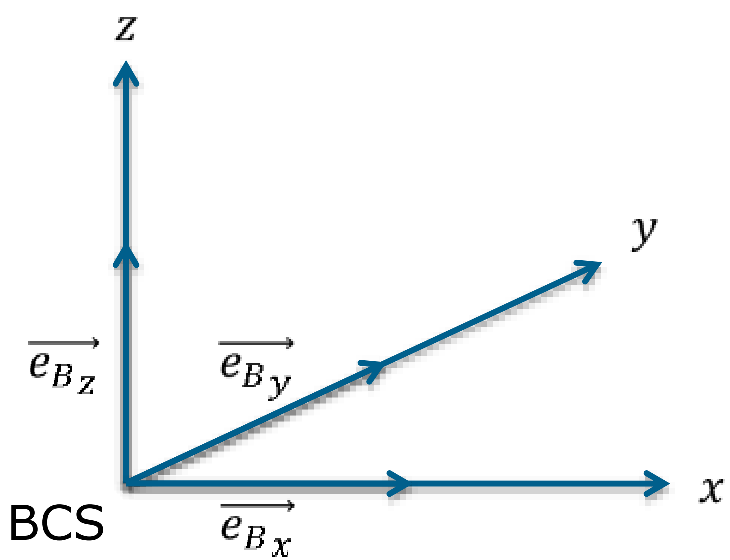
Notation

- Scalars: small letters, e.g. s
- Vectors: with an arrow, e.g. $\vec{u}$
- Matrices: upper case letters, e.g. A
- Identifier of scalars, vectors and points: indices at bottom right, e.g. $\overrightarrow{u_1}$
- Abbreviation of $\sin$ and $\cos$: sine and cosine:
  - $\cos(\theta_1) = C\theta_1 = C_1 , \sin(\theta_1) = S\theta_1 = S_1$
- Coordinate systems (frames): upper case letters, e.g. B
- Vectors referenced due to a certain frame: Frame upper left, e.g. ${}^{B}\vec{u}$
- Matrix transforming from frame B to frame A: Frames lower and upper left, e.g. ${}^{A}_{B}R$

Description of Objects and Poses in E_3

Base coordinate system (BCS)

- 3-dim. coordinate system defined by orthogonal unit vectors

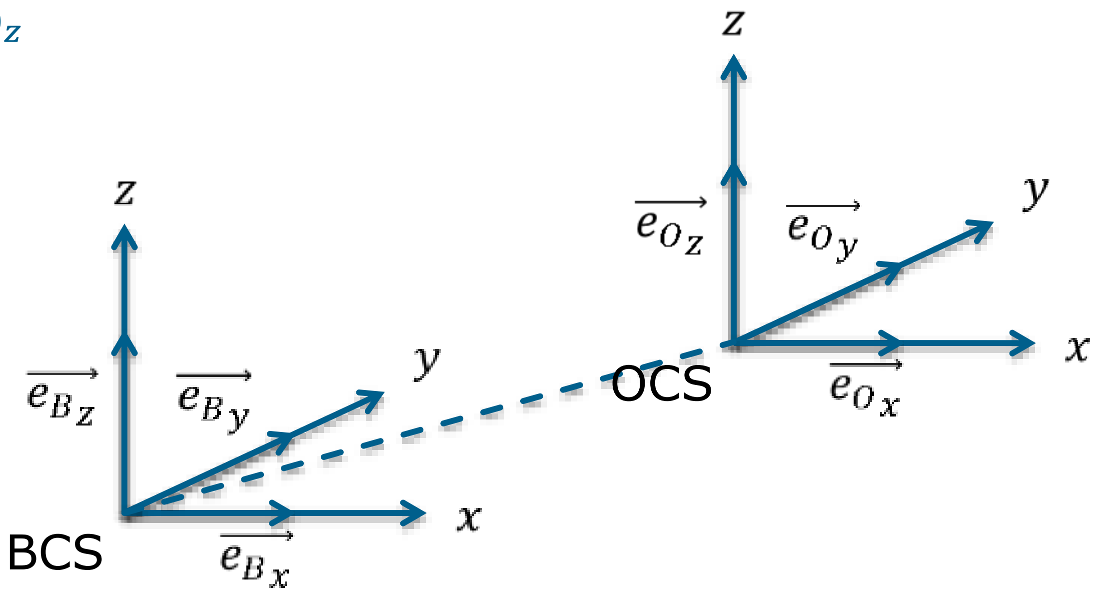


Description of Object Poses in E_3

Object coordinate system (OCS)

- Any rigid body can be related to a local coordinate system
- Local coordinate system is defined by orthogonal unit vectors

$\overrightarrow{e_{0_x}}, \overrightarrow{e_{0_y}}, \overrightarrow{e_{0_z}}$



Reminder: Scalar Product

$$\vec{a} \cdot \vec{b} = \sum_{i=1}^{n} a_i b_i = a_1 \mathrm{b}_1 + \mathrm{a}_2 \mathrm{b}_2 + \cdots + \mathrm{a}_\mathrm{n} \mathrm{b}_\mathrm{n}$$

$$\vec{a} \cdot \vec{b} = a \cdot b \cdot \cos\gamma$$

- γ: smallest angle between a and b
- 0, if the vectors are orthogonal
- Commutative and distributive property hold
- Associative does not hold
- With respect to scalars it is: $\mathrm{n}(\vec{a} \cdot \vec{b}) = (n \cdot \vec{a}) \cdot \vec{b} = \vec{a} \cdot (n \cdot \vec{b})$
- It holds:
  - $\vec{a} \cdot \vec{a} = |\vec{a}|^2$
  - $\overrightarrow{e_x} \cdot \overrightarrow{e_x} = \overrightarrow{e_y} \cdot \overrightarrow{e_y} = \overrightarrow{e_z} \cdot \overrightarrow{e_z} = 1$
  - $\overrightarrow{e_x} \cdot \overrightarrow{e_y} = \overrightarrow{e_y} \cdot \overrightarrow{e_z} = \overrightarrow{e_z} \cdot \overrightarrow{e_x} = 0$

Reminder: Cross Product/Vector Product

- Cross product $\vec{a} \times \vec{b}$ in spaces: Vector, which is perpendicular to $\vec{a}, \vec{b}$ and therefore normal to the plane containing them

- Definition for $\mathbb{R}^3$: $\vec{a} \times \vec{b} = |\vec{a}| \cdot |\vec{b}| \cdot \sin\theta \cdot \vec{e}$
  - θ: angle between the vectors
  - $\vec{e}$: perpendicular unit vector

- Cross product can be computed component wise for $\mathbb{R}^3$

$$\vec{a} \times \vec{b} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \times \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix}$$

Reminder: Cross Product/Vector Product

- Magnitude of the cross product is equal to the area of the parallelogram $A_P = |\vec{a} \times \vec{b}| = |\vec{a}| \cdot |\vec{b}| \cdot \sin\theta$
- For parallel vectors the cross product is 0
- It holds: $\vec{a} \times \vec{a} = \vec{0}$
- Distributive und anticommutative property hold
- $|\vec{a} \times \vec{b}| = \begin{vmatrix} \overrightarrow{e_1} & a_1 & b_1 \\ \overrightarrow{e_2} & a_2 & b_2 \\ \overrightarrow{e_3} & a_3 & b_3 \end{vmatrix}$

Reminder: Triple Product

$$V_{\vec{a},\vec{b},\vec{c}} = (\vec{a} \times \vec{b}) \cdot \vec{c} = (\vec{b} \times \vec{c}) \cdot \vec{a} = (\vec{c} \times \vec{a}) \cdot \vec{b} = \det \begin{bmatrix} a_x & b_x & c_x \\ a_y & b_y & c_y \\ a_z & b_z & c_z \end{bmatrix}$$

- Combination of cross and scalar product
- Magnitude: Signed volume (V) of the prism defined by the three vectors
  - $V > 0$ for right handed coordinate systems
  - $V < 0$ for left handed coordinate systems
- It holds:
  - for linear dependent vectors it is 0
  - anticommutative property holds

Reminder: Determinant

- Determinant of a $n \times n$-Matrix (Laplace‘s formula for i-th row)

$$\det(A) = \sum_{j=1}^{n} (-1)^{i+j}\, a_{ij} \det\left(A_{ij}\right)$$

- Rule of thump for 2×2-Matrices: Rule of Sarrus

$$\det \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = a_{11} \cdot a_{22} - a_{12} \cdot a_{21}$$

  - Example: Expanding the determinant along row 1:

$$\det \begin{bmatrix} 0 & 3 & 2 \\ 1 & 2 & 3 \\ 2 & 1 & 1 \end{bmatrix} = 0 \cdot \det \begin{bmatrix} 2 & 3 \\ 1 & 1 \end{bmatrix} - 3 \cdot \det \begin{bmatrix} 1 & 3 \\ 2 & 1 \end{bmatrix} + 2 \cdot \det \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}$$
$$= -3 \cdot -5 + 2 \cdot -3 = 15 - 6 = 9$$

Properties of a Determinant

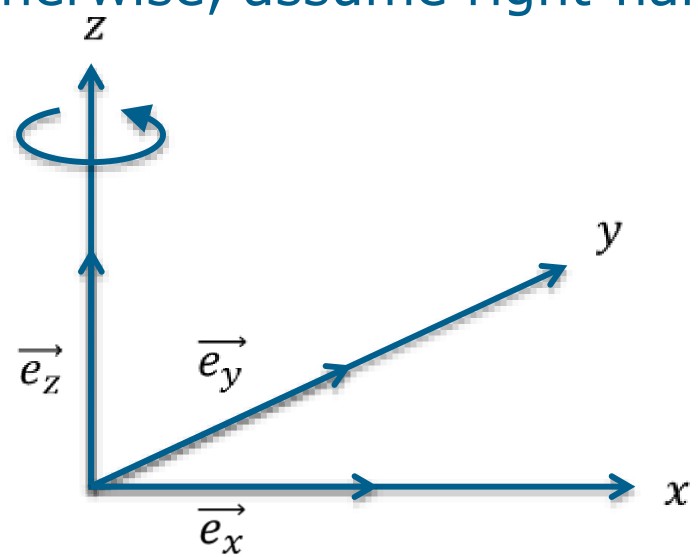
- $\det A = \det A^T$
- $\det \lambda A = \lambda^n \det A$
- $\det(A^{-1}) = \frac{1}{\det A}$ for $\det A \neq 0$
- Determinant is 0, if
  - all elements of a row/column are 0
  - two rows are linearly dependent

- Exchanging two rows changes the sign of the determinant

Orthogonal, Cartesian Coordinate Systems

Counterclockwise rotating coordinate system

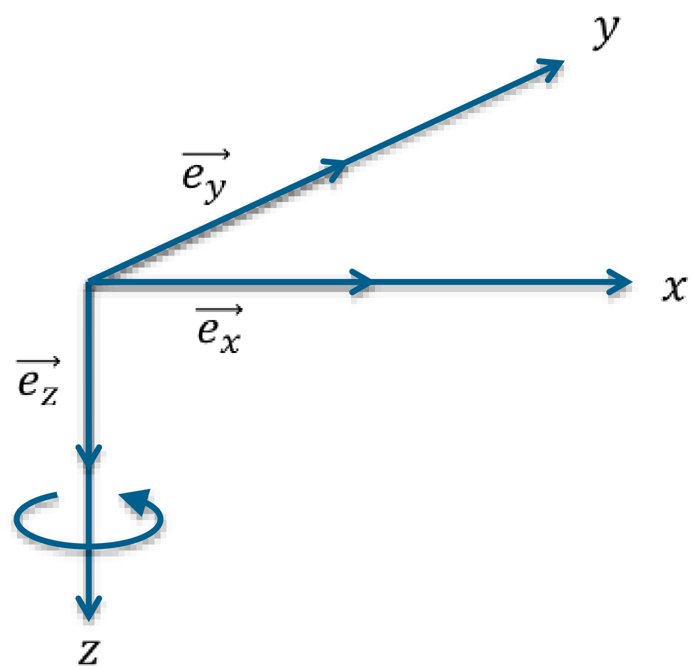
- Right-hand-rule: Thumb x, index finger y, middle finger z
- $\overrightarrow{e_x} \times \overrightarrow{e_y} = \overrightarrow{e_z}$ with cross product $\times$
- If not specified otherwise, assume right-handed CS



Orthogonal, Cartesian Coordinate Systems

Clockwise rotating coordinate system

- Left-hand-rule: Thumb x , index finger y , middle finger z
- $\overrightarrow{e_x} \times \overrightarrow{e_y} = -\overrightarrow{e_z}$



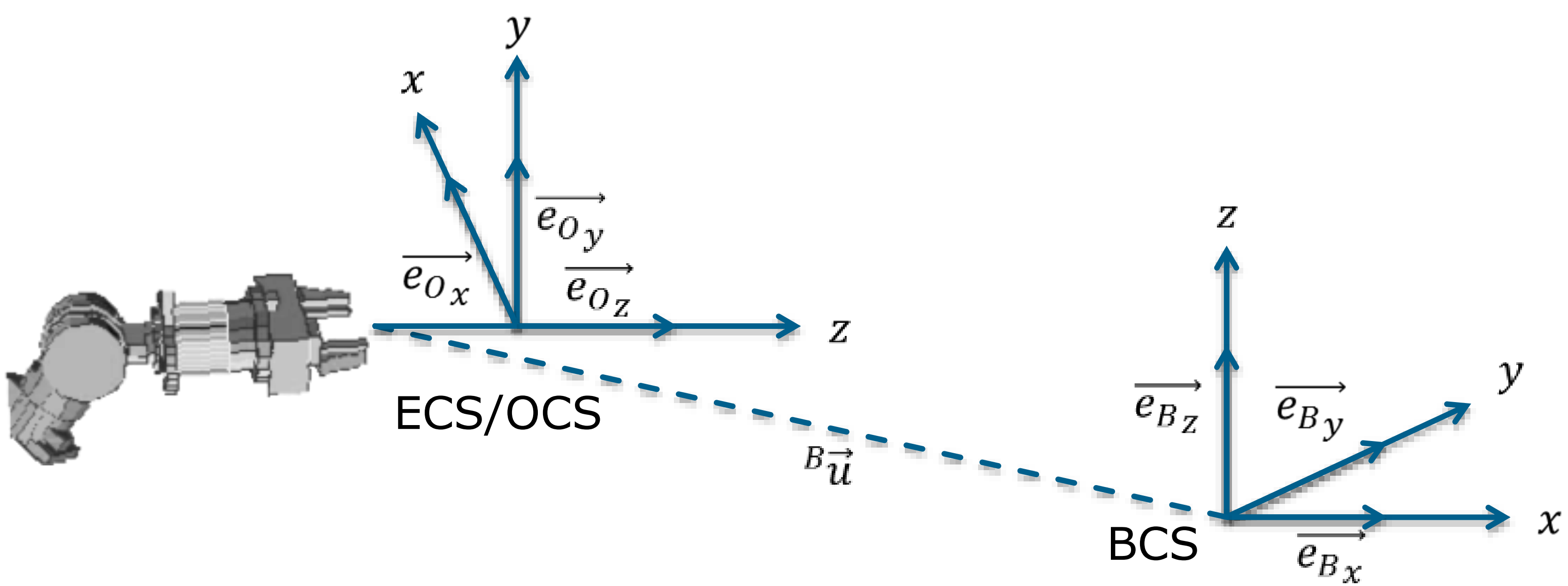
Object Poses in Space

- Location in BCS: Position vector from origin of BCS to origin of OCS
- Orientation in BCS: Mapping of unit vectors of OCS to the unit vectors of BCS using rotation matrix
- Pose: Position vector and rotation matrix of the OCS related to the BCS

Transformation

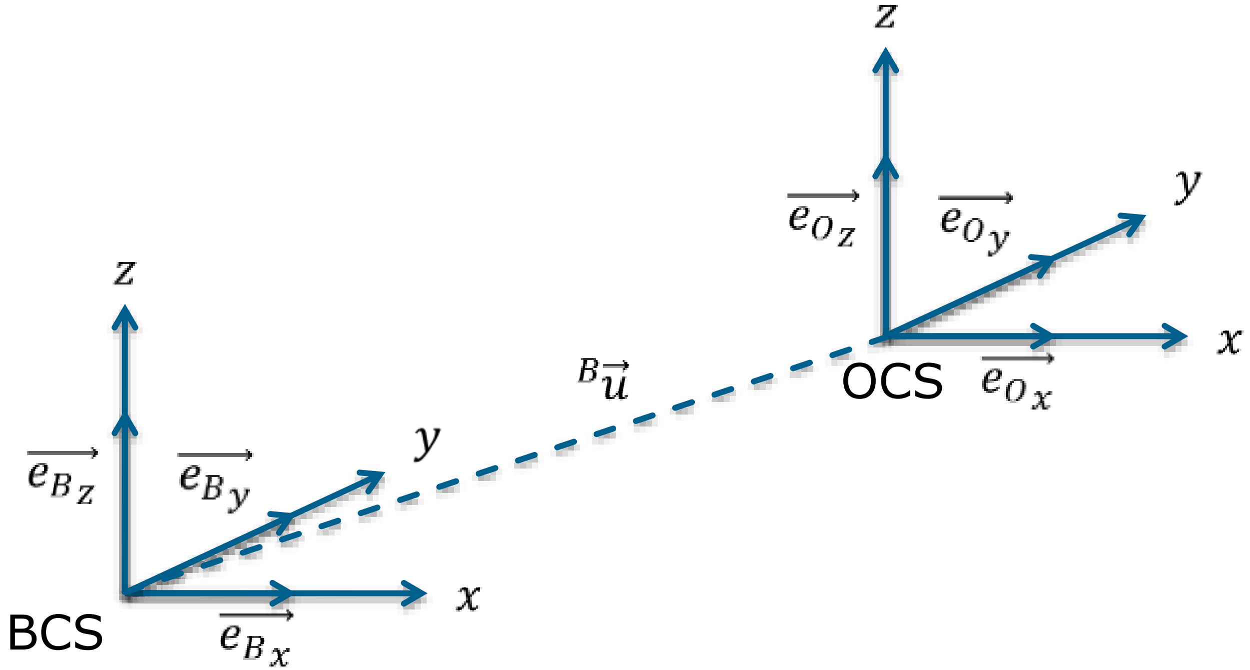
In addition to the BCS, various other local coordinate systems are used for describing robotic applications, e.g. …

- OCS: Object Coordinate System
- ECS: Effector Coordinate System (TCP – Tool Center Point)
- SCS: Sensor Coordinate System



Possible Transformations

- Translation vector: $\overrightarrow{{}^{B}u} = {}^{B}a \cdot \overrightarrow{e_{B_x}} + {}^{B}b \cdot \overrightarrow{e_{B_y}} + {}^{B}c \cdot \overrightarrow{e_{B_z}}$
- Rotation matrix: $R = R_\alpha \cdot R_\beta \cdot R_\gamma$
- Rotation angle around coordinate axes: x, y, z: $\alpha_x, \beta_y, \gamma_z$

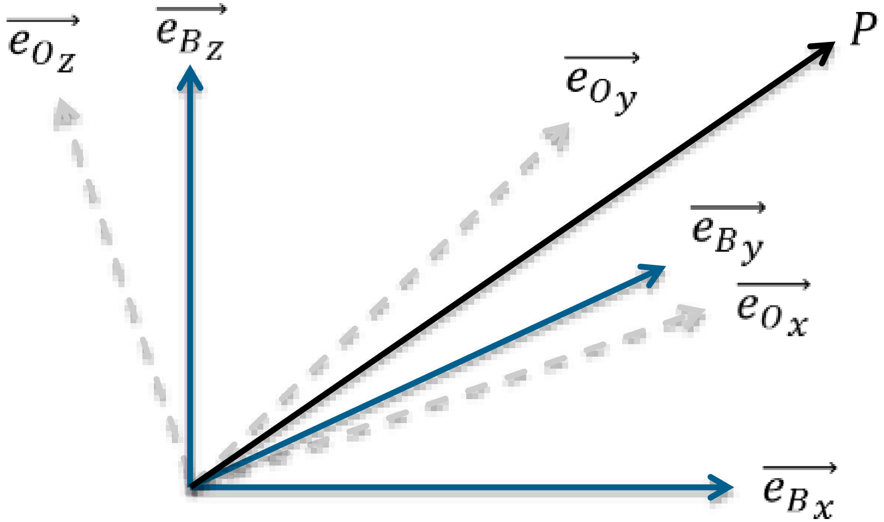


Rotation of a Coordinate System

- Let BCS and OCS be rotated against each other with unit vectors $\overrightarrow{e_{B_x}}, \overrightarrow{e_{B_y}}, \overrightarrow{e_{B_z}}$ and $\overrightarrow{e_{0_x}}, \overrightarrow{e_{0_y}}, \overrightarrow{e_{0_z}}$

- Given a position vector of a point P, either defined relative to the OCS ${}^{O}\overrightarrow{p}$ or the BCS ${}^{B}\overrightarrow{p}$

  -> find position vector relative to the other coordinate system



Rotation of a Coordinate System

- $^{B}p = {}^{B}p_x \cdot \overrightarrow{e_{B_x}} + {}^{B}p_y \cdot \overrightarrow{e_{B_y}} + {}^{B}p_z \cdot \overrightarrow{e_{B_z}}$ with $^{B}\vec{p} = \begin{bmatrix} {}^{B}p_x \\ {}^{B}p_y \\ {}^{B}p_z \end{bmatrix}$
- $^{O}p = {}^{O}p_x \cdot \overrightarrow{e_{B_x}} + {}^{O}p_y \cdot \overrightarrow{e_{B_y}} + {}^{O}p_z \cdot \overrightarrow{e_{B_z}}$ with $^{O}\vec{p} = \begin{bmatrix} {}^{O}p_x \\ {}^{O}p_y \\ {}^{O}p_z \end{bmatrix}$
- ^{O}p projection to base vectors of BCS yields to BCS coordinates:
  - $^{B}p_x = \overrightarrow{e_{B_x}} \cdot {}^{O}p = \overrightarrow{e_{B_x}} \cdot \overrightarrow{e_{O_x}} \cdot {}^{O}p_x + \overrightarrow{e_{B_x}} \cdot \overrightarrow{e_{O_y}} \cdot {}^{O}p_y + \overrightarrow{e_{B_x}} \cdot \overrightarrow{e_{O_z}} \cdot {}^{O}p_z$
  - $^{B}p_y = \overrightarrow{e_{B_y}} \cdot {}^{O}p = \overrightarrow{e_{B_y}} \cdot \overrightarrow{e_{O_x}} \cdot {}^{O}p_x + \overrightarrow{e_{B_y}} \cdot \overrightarrow{e_{O_y}} \cdot {}^{O}p_y + \overrightarrow{e_{B_y}} \cdot \overrightarrow{e_{O_z}} \cdot {}^{O}p_z$
  - $^{B}p_z = \overrightarrow{e_{B_z}} \cdot {}^{O}p = \overrightarrow{e_{B_z}} \cdot \overrightarrow{e_{O_x}} \cdot {}^{O}p_x + \overrightarrow{e_{B_z}} \cdot \overrightarrow{e_{O_y}} \cdot {}^{O}p_y + \overrightarrow{e_{B_z}} \cdot \overrightarrow{e_{O_z}} \cdot {}^{O}p_z$

Rotation of a Coordinate System

- Transformation from BCS to OCS coordinates:

  - ${}^{O}p_x = \overrightarrow{e_{O_x}} \cdot {}^{B}p = \overrightarrow{e_{O_x}} \cdot \overrightarrow{e_{B_x}} \cdot {}^{B}p_x + \overrightarrow{e_{O_x}} \cdot \overrightarrow{e_{B_y}} \cdot {}^{B}p_y + \overrightarrow{e_{O_x}} \cdot \overrightarrow{e_{B_z}} \cdot {}^{B}p_z$
  - ${}^{O}p_y = \overrightarrow{e_{O_y}} \cdot {}^{B}p = \overrightarrow{e_{O_y}} \cdot \overrightarrow{e_{B_x}} \cdot {}^{B}p_x + \overrightarrow{e_{O_y}} \cdot \overrightarrow{e_{B_y}} \cdot {}^{B}p_y + \overrightarrow{e_{O_y}} \cdot \overrightarrow{e_{B_z}} \cdot {}^{B}p_z$
  - ${}^{O}p_z = \overrightarrow{e_{O_z}} \cdot {}^{B}p = \overrightarrow{e_{O_z}} \cdot \overrightarrow{e_{B_x}} \cdot {}^{B}p_x + \overrightarrow{e_{O_z}} \cdot \overrightarrow{e_{B_y}} \cdot {}^{B}p_y + \overrightarrow{e_{O_z}} \cdot \overrightarrow{e_{B_z}} \cdot {}^{B}p_z$

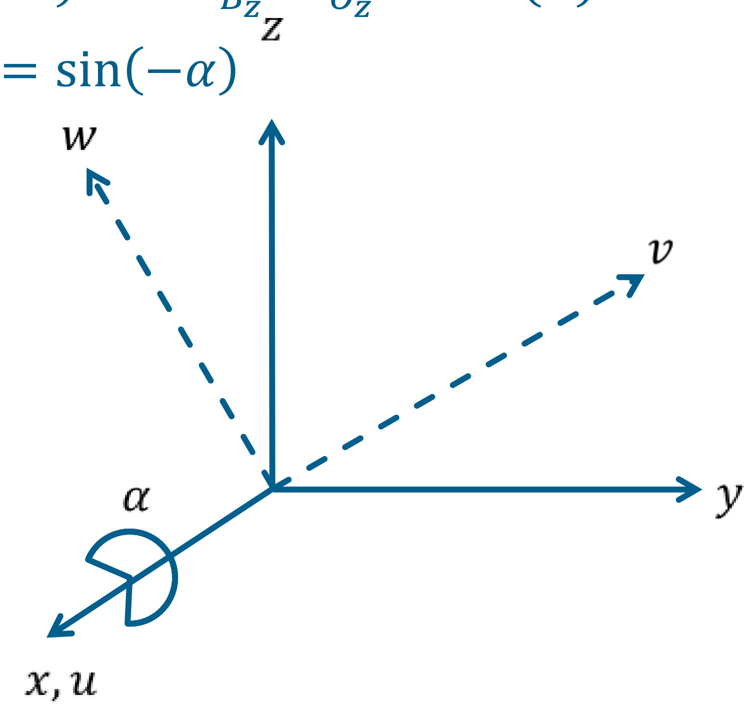
Using Matrix Notation

- $${}^{B}_{O}R_1 = \begin{bmatrix} \overrightarrow{e_{B_x}} \cdot \overrightarrow{e_{O_x}} & \overrightarrow{e_{B_x}} \cdot \overrightarrow{e_{O_y}} & \overrightarrow{e_{B_x}} \cdot \overrightarrow{e_{O_z}} \\ \overrightarrow{e_{B_y}} \cdot \overrightarrow{e_{O_x}} & \overrightarrow{e_{B_y}} \cdot \overrightarrow{e_{O_y}} & \overrightarrow{e_{B_y}} \cdot \overrightarrow{e_{O_z}} \\ \overrightarrow{e_{B_z}} \cdot \overrightarrow{e_{O_x}} & \overrightarrow{e_{B_z}} \cdot \overrightarrow{e_{O_y}} & \overrightarrow{e_{B_z}} \cdot \overrightarrow{e_{O_z}} \end{bmatrix} \text{ and } {}^{O}\vec{p} = \begin{bmatrix} {}^{O}p_x \\ {}^{O}p_y \\ {}^{O}p_z \end{bmatrix}$$
- $${}^{O}_{B}R_2 = \begin{bmatrix} \overrightarrow{e_{O_x}} \cdot \overrightarrow{e_{B_x}} & \overrightarrow{e_{O_x}} \cdot \overrightarrow{e_{B_y}} & \overrightarrow{e_{O_x}} \cdot \overrightarrow{e_{B_z}} \\ \overrightarrow{e_{O_y}} \cdot \overrightarrow{e_{B_x}} & \overrightarrow{e_{O_y}} \cdot \overrightarrow{e_{B_y}} & \overrightarrow{e_{O_y}} \cdot \overrightarrow{e_{B_z}} \\ \overrightarrow{e_{O_z}} \cdot \overrightarrow{e_{B_x}} & \overrightarrow{e_{O_z}} \cdot \overrightarrow{e_{B_y}} & \overrightarrow{e_{O_z}} \cdot \overrightarrow{e_{B_z}} \end{bmatrix} \text{ and } {}^{B}\vec{p} = \begin{bmatrix} {}^{B}p_x \\ {}^{B}p_y \\ {}^{B}p_z \end{bmatrix}$$
- $${}^{B}\vec{p} = {}^{B}_{O}R_1 \cdot {}^{O}\vec{p} = {}^{O}_{B}R_2{}^{-1} \cdot {}^{O}\vec{p}$$
- $${}^{O}\vec{p} = {}^{O}_{B}R_2 \cdot {}^{B}\vec{p} = {}^{B}_{O}R_1{}^{-1} \cdot {}^{B}\vec{p}$$
- Therefore: $R_1 = R_2^{-1}$, $R_2 = R_1^{-1}$ and $R_2 = R_1^T$ (orthogonal matrix)

Rotation around x-Axis with Angle α

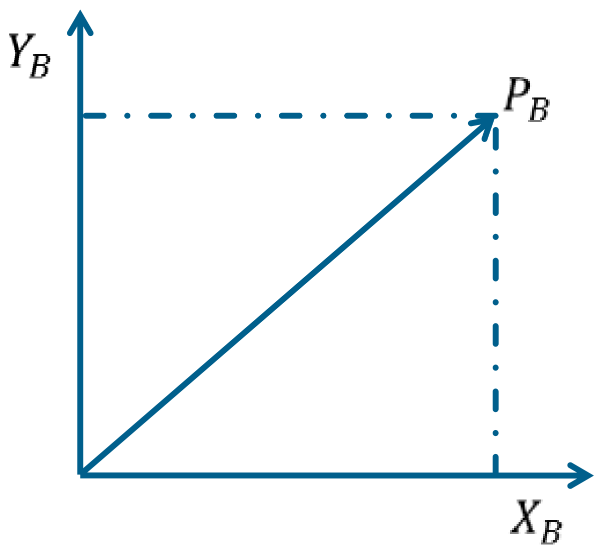
Using scalar product: $\vec{a}\cdot\vec{b} = |\vec{a}|\cdot|\vec{b}|\cos\alpha$

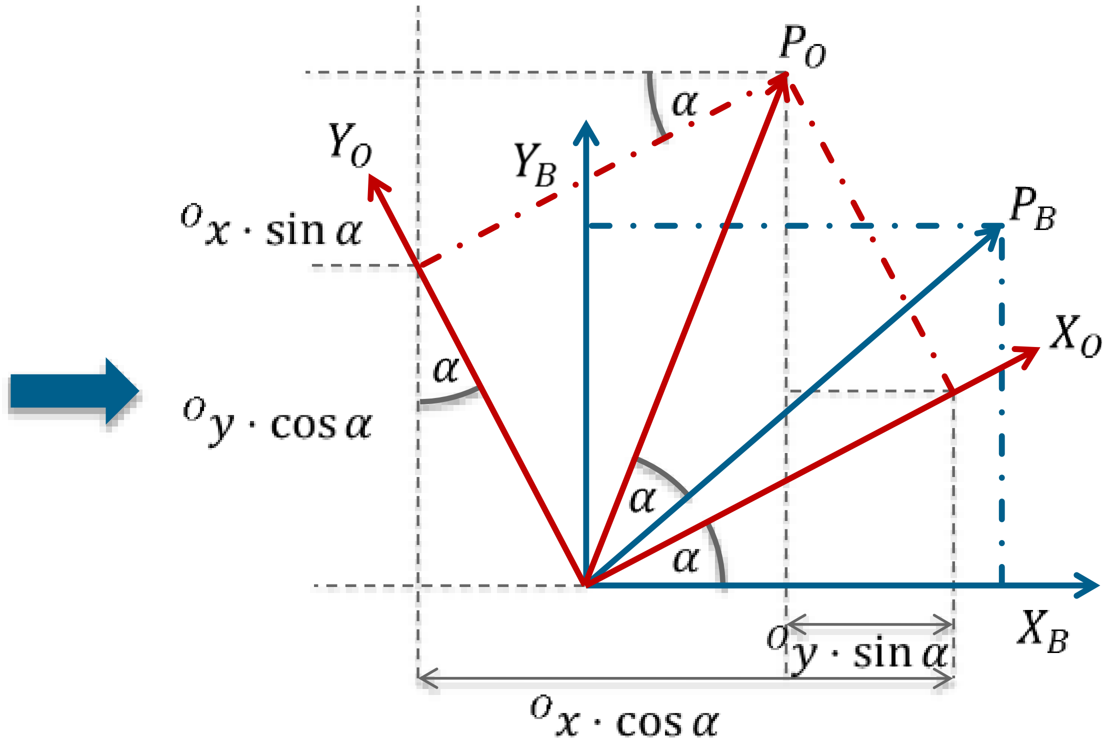
- $\overrightarrow{e_{B_x}}\cdot\overrightarrow{e_{O_x}} = 1$ $\quad\overrightarrow{e_{B_x}}\cdot\overrightarrow{e_{O_y}} = 0$ $\quad\overrightarrow{e_{B_x}}\cdot\overrightarrow{e_{O_z}} = 0$
- $\overrightarrow{e_{B_y}}\cdot\overrightarrow{e_{O_x}} = 0$ $\quad\overrightarrow{e_{B_y}}\cdot\overrightarrow{e_{O_y}} = \cos(\alpha)$ $\quad\overrightarrow{e_{B_y}}\cdot\overrightarrow{e_{O_z}} = c(\alpha)$
- $\overrightarrow{e_{B_z}}\cdot\overrightarrow{e_{o_x}} = 0$ $\quad\overrightarrow{e_{B_z}}\cdot\overrightarrow{e_{O_z}} = c(-\alpha)$ $\quad\overrightarrow{e_{B_z}}\cdot\overrightarrow{e_{O_z}} = \cos(\alpha)$
- $c(\alpha) = \cos(90^\circ + \alpha) = -\sin\alpha = \sin(-\alpha)$
- $R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\alpha & -S\alpha \\ 0 & S\alpha & C\alpha \end{bmatrix}$
- $C\alpha = \cos(\alpha), S\alpha = \sin(\alpha)$



Rotation Matrix: Geometric Derivation

- Frame $OX_0Y_0Z_0$ resulted from frame $BX_BY_BZ_B$ through rotation around axis z with angle α.
- Calculation of coordinates of point $P_O = ({}^O x,\ {}^O y,\ {}^O z)^T$ in coordinate system B





Rotation around the z-Axis

- Rotation around $z-$ axis with angle α
  - Point P_O with the coordinates $({}^{O}x, {}^{O}y, {}^{O}z)^T$ in OCS receives the coordinates in BCS …
    - ${}^{B}x = {}^{O}x \cdot \cos\alpha - {}^{O}y \cdot \sin\alpha$
    - ${}^{B}y = {}^{O}x \cdot \sin\alpha + {}^{O}y \cdot \cos\alpha$
    - ${}^{B}z = {}^{O}z$
  - $z-$ coordinate fixed, $z-$ axis is axis of rotation
- Matrix form: ${}^{B}\vec{p} = \begin{bmatrix} {}^{B}x \\ {}^{B}y \\ {}^{B}z \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} {}^{O}x \\ {}^{O}y \\ {}^{O}z \end{bmatrix} = {}^{B}_{O}R_z(\alpha) \cdot {}^{O}\vec{p}$

Rotation Matrix

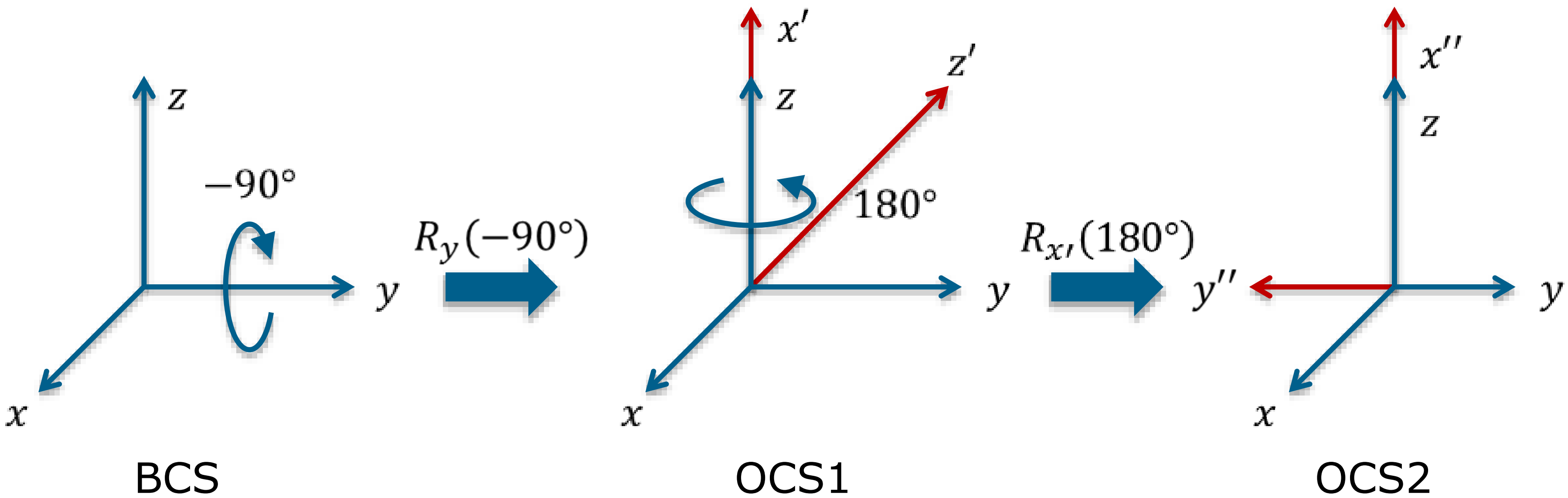
- Rotation matrix $R_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$
- Rotation around x and y - axes
  - $R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$
  - $R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$

Rotation Matrix - Properties

- Affine mapping $\mathbb{R}_3 \rightarrow \mathbb{R}_3$
- Real matrix
- Quadratic
- Invertible
- Orthogonal
  - Row or column vectors are orthogonal to each other
- Let R be a rotation matrix:
  - Rank $R_g(R) = 3$
  - $R^T = R^{-1}$
  - $R \cdot R^{-1} = R^{-1} \cdot R = I$
  - $\det R = 1$

Basic Rotations

- Let OCS result based on 2 rotations from BCS



$$R_y(-90^\circ) = \begin{bmatrix} \cos -\frac{\pi}{2} & 0 & \sin -\frac{\pi}{2} \\ 0 & 1 & 0 \\ -\sin -\frac{\pi}{2} & 0 & \cos -\frac{\pi}{2} \end{bmatrix}, \qquad R_{x'}(180^\circ) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\pi & -\sin\pi \\ 0 & \sin\pi & \cos\pi \end{bmatrix}$$

Vector Coordinates due to a new Frame

- Calculation of ${}^{B}\vec{u}$ from ${}^{O2}\vec{u}$
- ${}^{B}\vec{u} = {}^{B}_{O1}R_y(-90°) \cdot {}^{O1}\vec{u} = {}^{B}_{O1}R_y(-90°) \cdot {}^{O1}_{O2}R_{x'}(180°)\,{}^{O2}\vec{u} =$

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} {}^{O2}\overrightarrow{u_1} \\ {}^{O2}\overrightarrow{u_2} \\ {}^{O2}\overrightarrow{u_3} \end{bmatrix} = \begin{bmatrix} {}^{O2}\overrightarrow{u_3} \\ -{}^{O2}\overrightarrow{u_2} \\ {}^{O2}\overrightarrow{u_1} \end{bmatrix}$$

- ${}^{B}\overrightarrow{e_{O2_x}} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$
- ${}^{B}\overrightarrow{e_{O2_y}} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$
- ${}^{B}\overrightarrow{e_{O2_z}} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

Interpretation of several, elementary Rotations

- Pre-multiplication $R = \left(R_n(R_{n-1} \dots (R_2 R_1) \dots)\right)$:
  - Interpretation - rotation around a fixed axis of the original coordinate system

- Post-multiplication $\left((\cdots(R_1 R_2) \cdots R_{n-1})R_n\right)$:
  - Interpretation - rotation around an axis of the rotated CS

Different Notations for Rotations

- Many different notations for rotations exist
- All equivalent, but different benefits
  - Rotation around 1 axis
    - Trade-off between others
  - Quaternions
    - Computationally fast
  - Exponential coordinates
    - More similar to its kinematic
  - Euler angels
    - Follows chained Joint-Setup
  - Roll-Pitch-Yaw
    - Easy to interpret by humans
  - …

Rotation around 1 Axis

- Instead of rotation with BCS-axis, rotate around 1 Axis:
- Goal:

Find $\vec{g} \in \mathbb{R}^3,\ \|\vec{g}\| = 1,\ \theta \in [0, 2\pi)$
such that:
For BCS $x, y, z \in \mathbb{R}^3$ and arbitrary $\alpha, \beta, \gamma \in [0, 2\pi)$
the following holds:

$$R_{\vec{g}}(\theta) = R_z(\gamma) R_y(\beta) R_x(\alpha)$$

Rodrigues Formula:

- For $\mathrm{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$

  Choose $\theta = \cos^{-1}\left(\frac{tr(R)-1}{2}\right) \in [0, \pi]$, and $\vec{g} = \frac{1}{2S\theta}\begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$, then:

- $R = R_{\vec{g}}(\theta) = C\Theta I_3 + S\theta\vec{g} + \big(1 - cos(\theta)\big)\vec{g}\vec{g}^T$

$$= \begin{bmatrix} g_1^2\eta\theta + C\theta & g_1g_2\eta\theta - S\theta & g_1g_3\eta\theta + g_2S\theta \\ g_1g_2\eta\theta + g_3S\theta & g_2^2\eta\theta + C\theta & g_2g_3\eta\theta - g_1S\theta \\ g_1g_3\eta\theta - g_2S\theta & g_2g_3\eta\theta + g_1S\theta & g_3^2\eta\theta + C\theta \end{bmatrix}$$

With:

$\mathrm{S}\theta = \sin\theta,\ C\theta = \cos\theta,\ \eta\theta = 1 - \cos\theta\ , \vec{g} = (g_1, g_2, g_3)^T$

Theorem (Euler):

Every rotation matrix R_3 is equivalent to a rotation around a fixed axis

$$\vec{g} \in \mathbb{R}^3, ||\vec{g}|| = 1,$$

And a rotation angle

$$\theta \in [0,2\pi).$$

Proof:

- $$\mathrm{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \overset{!}{=} R_{\vec{g}}(\theta)$$

$$= \begin{bmatrix} g_1^2\eta\theta + C\theta & g_1 g_2\eta\theta - S\theta & g_1 g_3\eta\theta + g_2 S\theta \\ g_1 g_2\eta\theta + g_3 S\theta & g_2^2\eta\theta + C\theta & g_2 g_3\eta\theta - g_1 S\theta \\ g_1 g_3\eta\theta - g_2 S\theta & g_2 g_3\eta\theta + g_1 S\theta & g_3^2\eta\theta + C\theta \end{bmatrix}$$

The following applies to the trace of the matrices:

$$trR = r_{11} + r_{22} + r_{33} \overset{!}{=} 3\cos\theta + (1-\cos\theta)\sum g_i^2 = 1 + 2\cos\theta$$

$$\Longrightarrow \theta = cos^{-1}\left(\frac{tr\,R - 1}{2}\right) \in [0,\pi]$$

This equation can be solved for θ, because the eigenvalues λ_i of R have amount 1 and therefore:

$$-1 \leq tr\,R = \sum \lambda_i \leq 3$$

Proof:

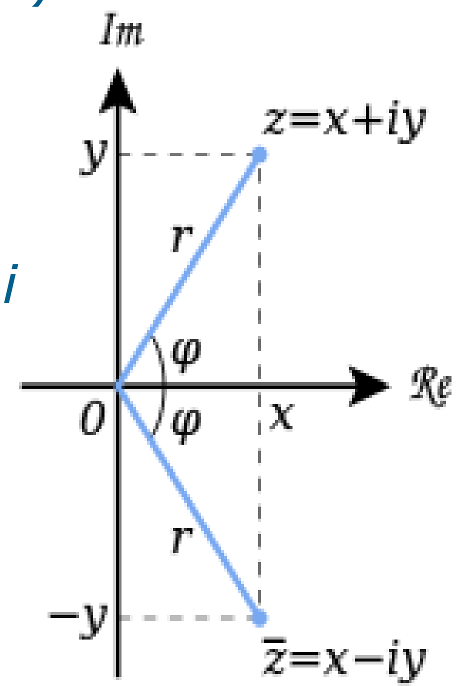
- To determine the axis of rotation, we use the remaining matrix entries:

$$\left.\begin{aligned} r_{32} - r_{23} &\overset{!}{=} 2g_1 S\theta \\ r_{13} - r_{31} &\overset{!}{=} 2g_2 S\theta \\ r_{21} - r_{12} &\overset{!}{=} 2g_3 S\theta \end{aligned}\right\} \overset{\theta \neq 0}{\Longrightarrow} \vec{g} = \frac{1}{2S\theta}\begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$$

If $\mathrm{R} = I_3$, then $trR = 3$, and therefore $\theta = 0$. In this case, $\vec{g}$ could be any vector, then $R_{\vec{g}}(0) = I_3$.

Complex Numbers

- **Form of Complex number**: $a + bi$
  - a,b are real numbers,
  - i is the **imaginary unit** with $i^2 = -1$, $i = \sqrt{-1}$, $(\pm i)^2 = -1$
- **Adding:** $(a + bi) + (c + di) = (a + c) + (b + d)i$
- **Subtracting:** $(a + bi) - (c + di) = (a - c) + (b - d)i$
- **Multiplying:** $(a + bi) \times (c + di) = (ac - bd) + (ad + bc)i$
- **Complex conjugate**: $a - bi$.
- **Dividing:** $1 / (a + bi) = (a - bi) / (a^2+b^2)$

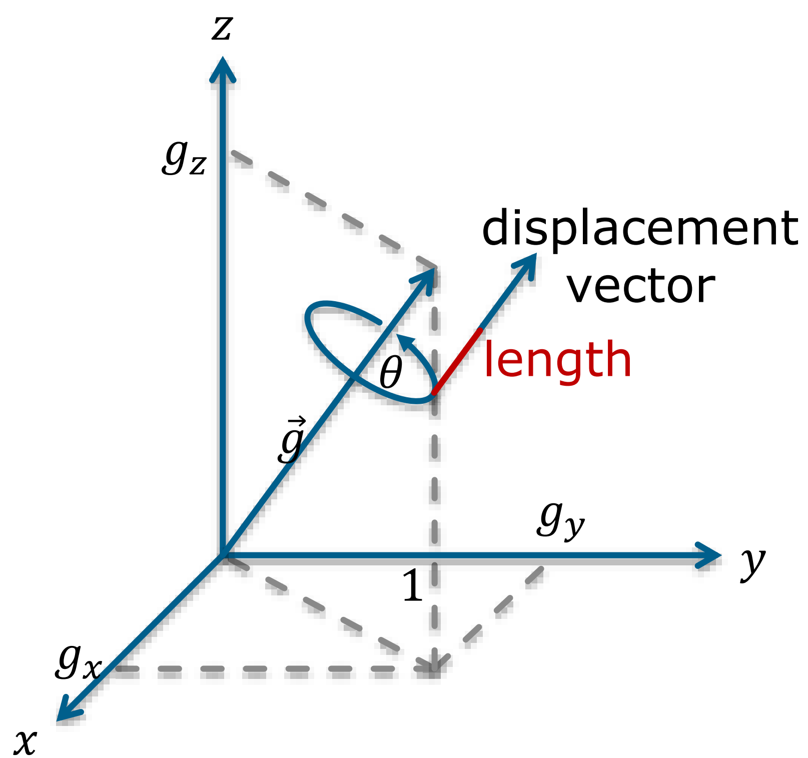


Historical Aspects

- Complex number *a* + *bi* viewable as point in a plane
- Transfer to space leads to Quaternions
- Quaternion from lat. Word *quaternio, -ionis f.* „Foursome“)
- 1843, Hamilton representation of vector in space through complex numbers
- Multiplication not possiblw with triples but with quadruples
- Basic rules of multiplication $i^2 = j^2 = k^2 = ijk = -1$
- $Q = r_1 + i \cdot r_2 + j \cdot r_3 + k \cdot r_4$

Quaternions

- Problems of (homogeneous) rotation matrix
  - High redundancy
  - Many arithmetic operations at concatenation
  - Singularities
- Orientation of a rigid body
  - Quaternion: Rotation axis (3 dim. Vector $\vec{g}$) and angle θ sufficient
  - Reduction of the required computational effort



Quaternions

- Quaternions $Q = (r_1, r_2, r_3, r_4)$ with $r_1, r_2, r_3, r_4 \in \mathbb{R}$
- Often represented as a linear vector space over $\mathbb{R}$
  - $Q = r_1 + i \cdot r_2 + j \cdot r_3 + k \cdot r_4$ (extension of $\mathbb{C}$)
- For basic elements $1, i, j, k$ the following multiplicative linkage table (not commutative) applies:

$\cdot$	1	i	j	k
1	1	i	j	k
i	i	-1	k	$-j$
j	j	$-k$	-1	i
k	k	j	$-i$	-1

Real Quaternions

- Scalar part: r_1 (angle of rotation)
- Vector part: $i \cdot r_2 + j \cdot r_3 + k \cdot r_4$ (axis of rotation)
- Quaternions can be used to display all rotations in which the axis of rotation passes through the origin of the reference system
  - Conjugated: $\bar{Q} = r_1 - i \cdot r_2 - j \cdot r_3 - k \cdot r_4$
  - Magnitude: $|Q| = \sqrt{Q \cdot \bar{Q}}$
  - Inverse: $Q^{-1} = \frac{\bar{Q}}{|Q|^2} \rightarrow Q \cdot Q^{-1} = Q^{-1} \cdot Q = 1$

Real Quaternions - Example

- Let us consider: $Q_1 = (3,2,-4,1)$ and $Q_2 = (4,-3,1,-5)$
- It holds:
  - $Q_1 + Q_2 = (7,-1,-3,-4)$
  - $Q_1 \cdot Q_2 = (3 + 2i - 4j + k)(4 - 3i + j - 5k) =$
    $12 - 9i + 3j - 15k + 8i - 6i^2 + 2ij - 10ik - 16j + 12ji - 4j^2 +$
    $20jk + 4k - 3ki + kj - 5k^2 =$
    $12 - 9i + 3j - 15k + 8i + 6 + 2k + 10j - 16j - 12k + 4 + 20i +$
    $4k - 3j - i + 5 = (12 + 6 + 4 + 5) + i(-9 + 8 + 20 - 1) +$
    $j(3 + 10 - 16 - 3) + k(-15 + 2 - 12 + 4) = (27,18,-6,-21)$
  - $Q_2 \cdot Q_1 = (27,-20,-20,-1)$
  - $Q_1^{-1} = \frac{(3-2i+4j-k)}{30}$

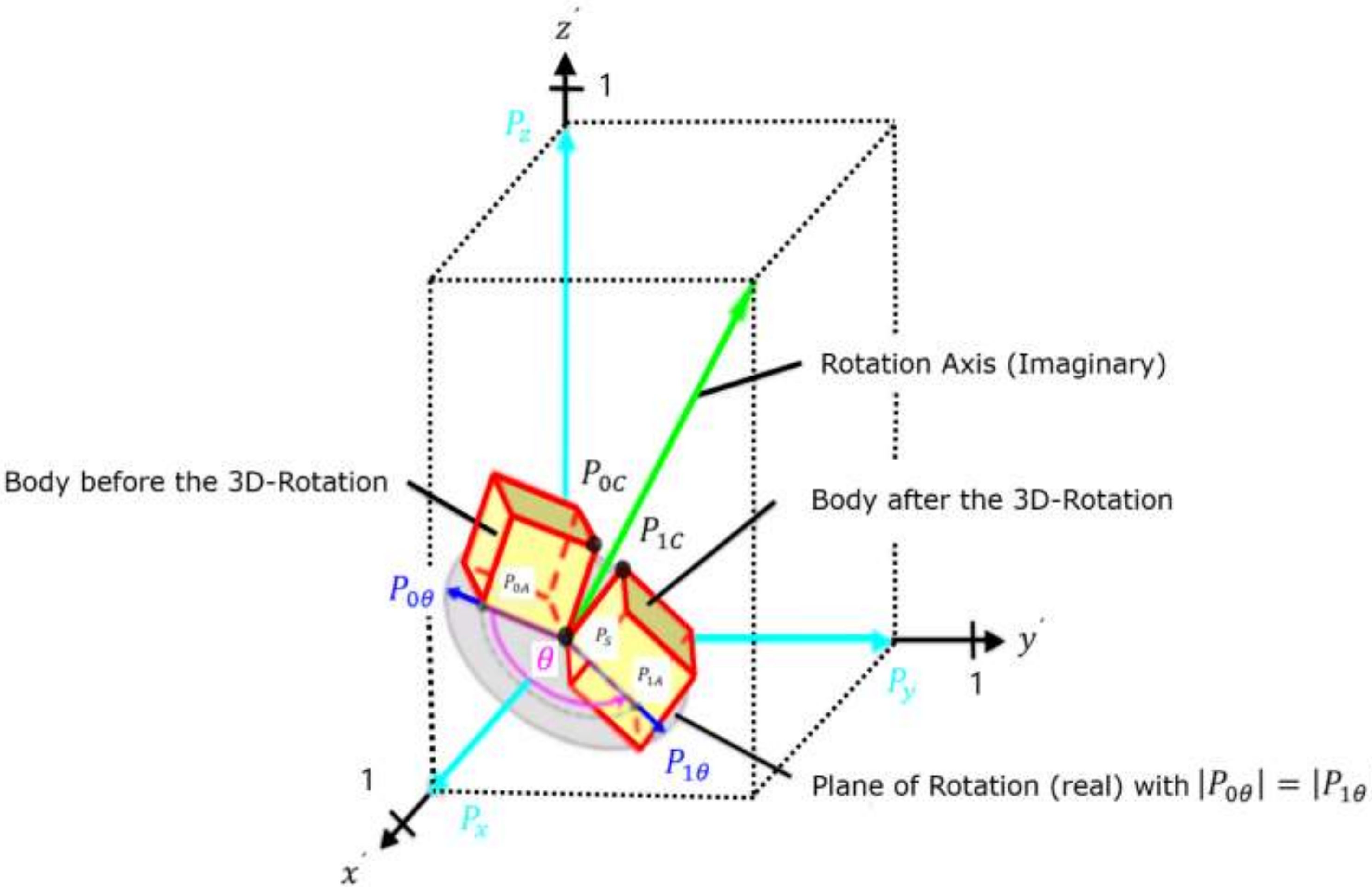
Rotation of Points by Means of Quaternions

Unit quaternion : $|Q| = 1 \Rightarrow Q^{-1} = \bar{Q}$,
since $|Q|^2 = 1 \rightarrow$ Simple forward / backward calculation

Rotation of point $\vec{p}$ at axis $\vec{g}$ with angle θ

1. Create a unique quaternion from $\vec{g}$ and θ
   (1) Standardization of $\vec{g}$ to 1
   (2) $Q = \left[\cos\frac{\theta}{2}, \sin\left(\frac{\theta}{2}\right)\vec{g}\right]$, since $\cos^2(\theta) + \sin^2(\theta) = 1$
2. Represent point $\vec{p}$ as quaternion
   $P = [0, \vec{p}]$
1. Final rotation:
   $P' = Q \cdot P \cdot Q^{-1} = Q \cdot P \cdot \bar{Q}$

Rotation of Points by Means of Quaternions



Converting Quaternion/Rotation matrix

- Rotation quaternion $Q = \left(s, (x, y, z)\right)$

- From rotation by means of unit quaternion $|Q| = 1 \Rightarrow Q^{-1} = \bar{Q}$ follows the rotation matrix R:

-

$$R = \begin{bmatrix} 1 - 2(y^2 + z^2) & 2xy - 2sz & 2sy + 2xz \\ 2xy + 2sz & 1 - 2(x^2 + z^2) & -2sx + 2yz \\ -2sy + 2xz & 2sx - 2yz & 1 - 2(x^2 + y^2) \end{bmatrix}$$

Converting Quaternion/Rotation matrix

- From R with the entries $r_{ij}, i, j \in \{1,2,3\}$, the corresponding rotation quaternion is calculated $Q = \big(s, (x, y, z)\big)$ as follows:
  - $s = \frac{1}{2}\sqrt{1 + r_{11} + r_{22} + r_{33}}$
  - $x = \frac{r_{32} - r_{23}}{4s}$
  - $y = \frac{r_{13} - r_{31}}{4s}$
  - $z = \frac{r_{21} - r_{12}}{4s}$

Exponential coordinates Motivation

- Rotation axis $\vec{g}$, rotation angle θ

- Motivation:
  Rotate a point $\vec{q}$ with a velocity of 1 around an axis $\vec{g}$

$$\Rightarrow \dot{\vec{q}}(t) = \vec{g} \times \vec{q}(t) =: \tilde{\vec{g}}\vec{q}(t)$$

With:

$$\tilde{\vec{g}} = \begin{bmatrix} 0 & -g_3 & g_2 \\ g_3 & 0 & -g_1 \\ -g_2 & g_1 & 0 \end{bmatrix}$$

Exponential coordinates

- Rotate by angle θ with integral equation:

$$\int_0^{\theta} g\,\vec{q}(t)dt = e^{g\theta}\vec{q}(0)$$

Where $e^{g\theta}$ is exponential of a matrix.

$$\begin{aligned} e^{g\theta} &:= 1 + g\theta + \frac{(g\theta)^2}{2!} + \frac{(g\theta)^3}{3!} + \ldots \\ &= 1 + g\sin\theta + g^2(1-\cos\theta) \end{aligned}$$

Axes of Rotation in Robotics

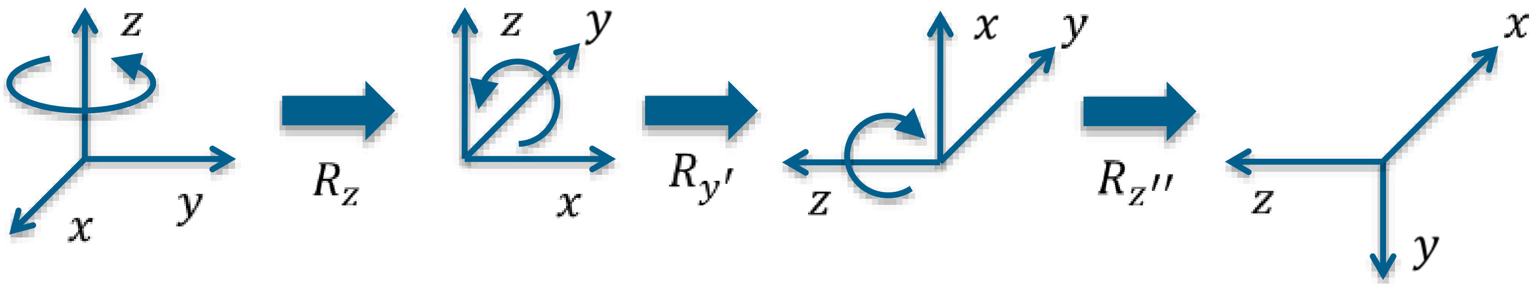
- Rotation axes usually BCS
- Convention of rotation axes and their order usually in …
  - Euler-angles
  - Roll, Pitch, Yaw

Euler-Angles (zxz)

- Rotation α around the z - axis of BCS: $R_z(\alpha)$
- Rotation β around the new x – axis x': $R_{x'}(\beta)$
- Rotation γ around the new z – axis z'': $R_{z''}(\gamma)$
- $R_S = R_z(\alpha) \cdot R_{x'}(\beta) \cdot R_{z''}(\gamma)$
- $R_S(\alpha, \beta, \gamma) = \begin{bmatrix} C\alpha\, C\gamma - C\beta\, S\gamma\, S\alpha & -C\alpha\, S\gamma - C\beta\, C\gamma\, S\alpha & S\alpha\, S\beta \\ S\alpha\, C\gamma + C\beta\, S\gamma\, C\alpha & C\alpha\, C\beta\, C\gamma - S\alpha\, S\gamma & -C\alpha\, S\beta \\ S\gamma\, S\beta & C\gamma\, S\beta & C\beta \end{bmatrix}$

Euler-Angles (zyz)

- Rotation α around the z - axis of BCS: $R_z(\alpha)$
- Rotation β around the new y – axis y': $R_{y'}(\beta)$
- Rotation γ around the new z – axis z'': $R_{z''}(\gamma)$
- $R_s(\alpha,\beta,\gamma) = R_z(\alpha) \cdot R_{y'}(\beta) \cdot R_{z''}(\gamma)$
- $R_s(\alpha,\beta,\gamma) = \begin{bmatrix} C\alpha\, C\beta\, C\gamma - S\alpha\, S\gamma & -C\alpha\, C\beta\, S\gamma - S\alpha\, C\gamma & C\alpha\, S\beta \\ S\alpha\, C\beta\, C\gamma + C\alpha\, S\gamma & -S\alpha\, C\beta\, S\gamma - C\alpha\, C\gamma & S\alpha\, S\beta \\ -S\beta\, C\gamma & S\beta\, S\gamma & C\beta \end{bmatrix}$
- Rotation around changed axes $R_{z,\alpha}, R_{y',\beta}, R_{z'',\gamma}$

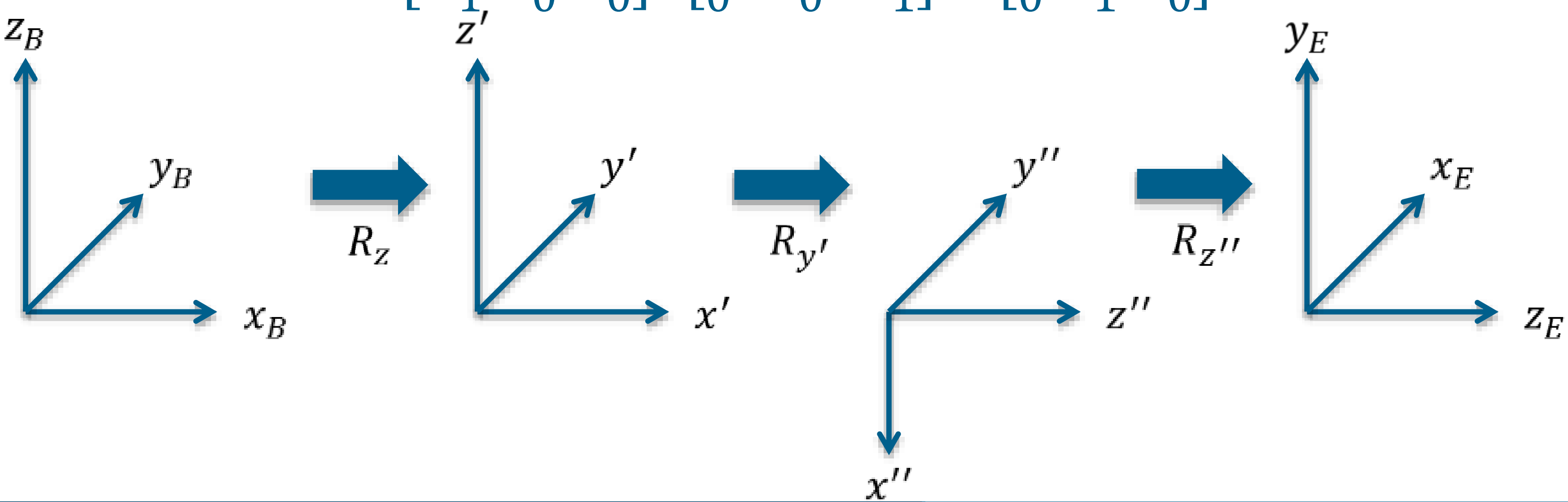


Euler-Angles - Example

- $R_s = R_z(0^\circ) \cdot R_{y'}(90^\circ) \cdot R_{z''}(90^\circ)$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

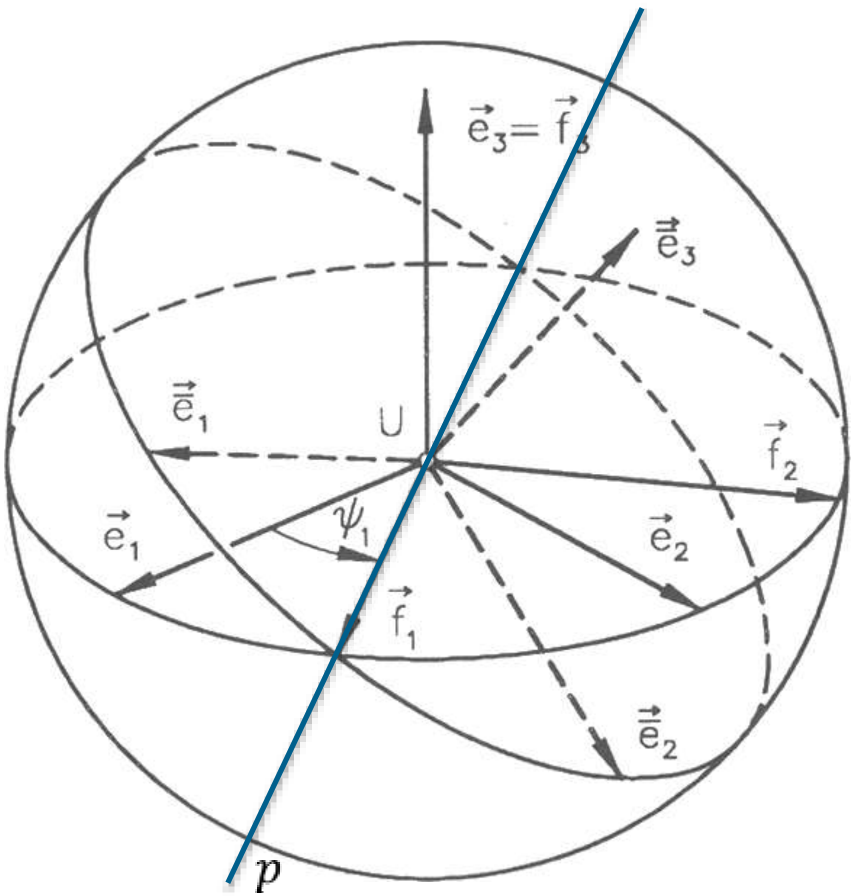


Euler-Angles: Derivation

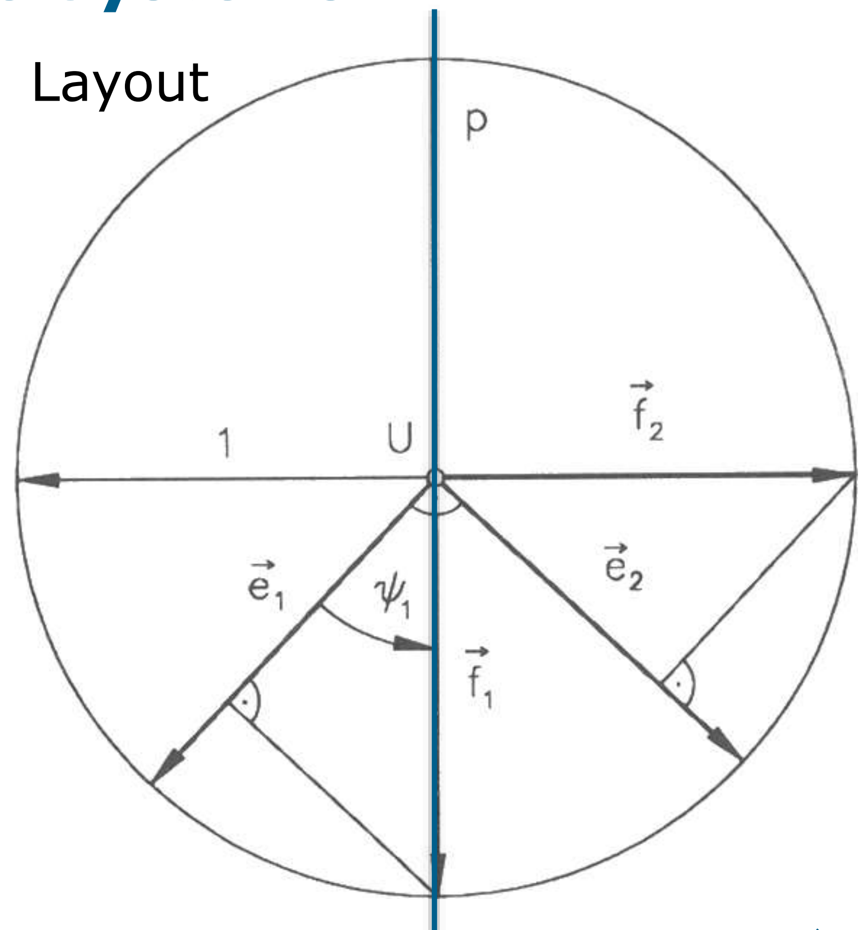
- Proof: All orientations can be described using Euler-angles

Theorem: If two right-handed Cartesian coordinate systems $R = \{U, \overrightarrow{e_1}, \overrightarrow{e_2}, \overrightarrow{e_3}\}$ and $\{U, \overrightarrow{\overline{e}_1}, \overrightarrow{\overline{e}_2}, \overrightarrow{\overline{e}_3}\}$ with a common origin exist, then there exists an orthogonal matrix A that maps R to $\bar{R}$

Euler-Angles - Coordinate Systems



Layout

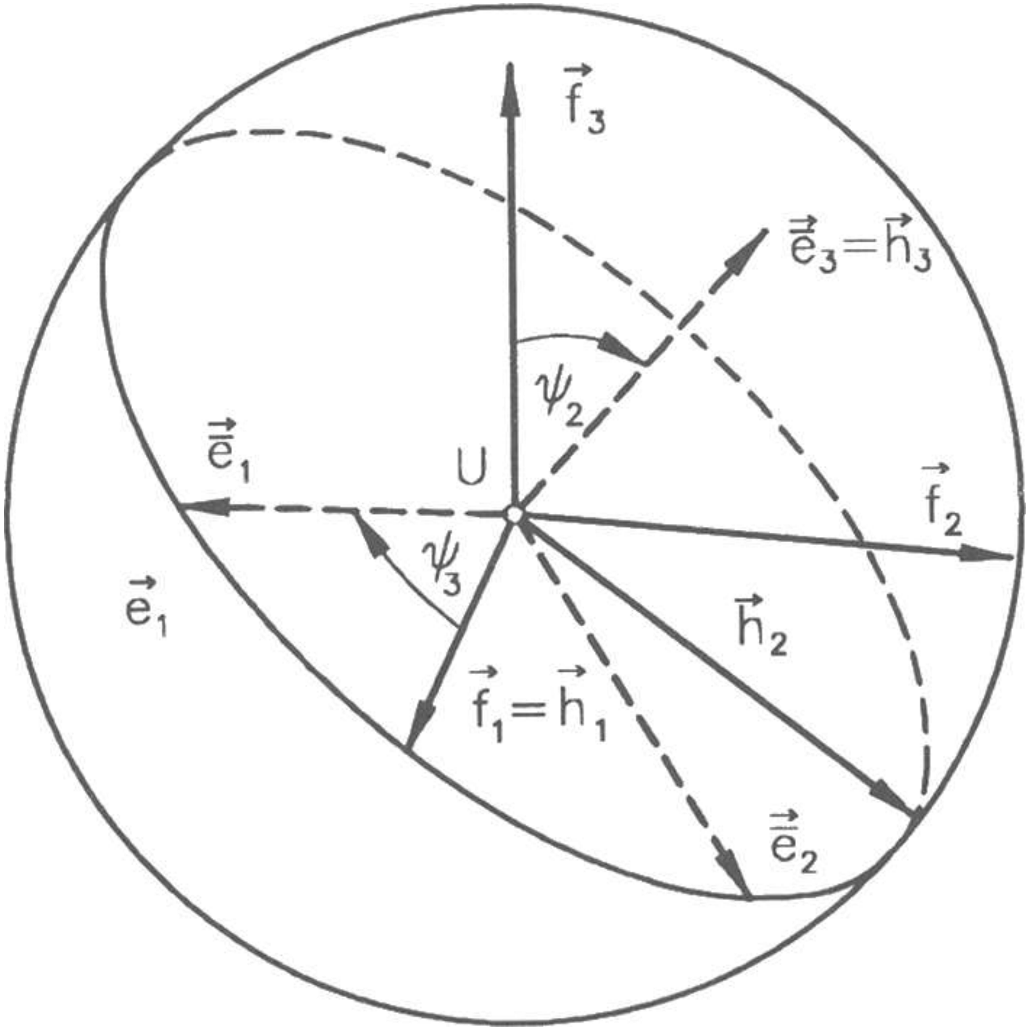


- Plane E_1 (spanned by $\overrightarrow{e_1}$ and $\overrightarrow{e_2}$) intersects E_2 (spanned by $\overrightarrow{\overline{e_1}}$ and $\overrightarrow{\overline{e_2}}$) in line p.

Euler-Angles: Derivation

1. Rotation around $\overrightarrow{e_3}$ with the positive angle ψ_1 so that $\overrightarrow{e_1}$ is mapped onto $\overrightarrow{f_1}$
   - $\overrightarrow{f_1}$, constructed by positive rotation with ψ_1 with $0 \leq \psi_1 \leq \pi$, lies on p
   - R transforms into $R = \{U, \overrightarrow{f_n}, \overrightarrow{f_2}, \overrightarrow{f_3} = \overrightarrow{e_3}\}$
   - $A_1 = \begin{bmatrix} \cos\psi_1 & -\sin\psi_1 & 0 \\ \sin\psi_1 & \cos\psi_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} R_1 = RA_1$
   - $\overrightarrow{f_1} \perp \overrightarrow{e_3}$ and $\overrightarrow{f_1} \perp \overrightarrow{e_3}$

Euler-Angles - Coordinate Systems



Euler-Angles: Derivation

2. Rotate R_1 around axis $\overrightarrow{f_1}$ with angle ψ_2 so that $\overrightarrow{e_3} = \overrightarrow{f_3}$ falls together with $\overrightarrow{\overline{e_3}}$

- R transforms to $R_2 = \left\{U, \overrightarrow{f_1} = \overrightarrow{h_1}, \overrightarrow{h_2}, \overrightarrow{h_3} = \overrightarrow{\overline{e_3}}\right\}$

- $A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Psi_2 & -\sin\Psi_2 \\ 0 & \sin\Psi_2 & \cos\Psi_2 \end{bmatrix} R_2 = R_1 A_2$

- $\overrightarrow{f_2}$ is mapped onto $\overrightarrow{h_2}$

- $\overrightarrow{h_2}$ lies in the plane spanned by $\overrightarrow{\overline{e_1}}$ and $\overrightarrow{\overline{e_2}}$

Euler-Angles: Derivation

3. Rotate R_3 with the angle ψ_3, so that R_2 falls together with $\bar{R}$

- $$A_3 = \begin{bmatrix} \cos\psi_3 & -\sin\psi_3 & 0 \\ \sin\psi_3 & \cos\psi_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} R_3 = R_2 A_3$$

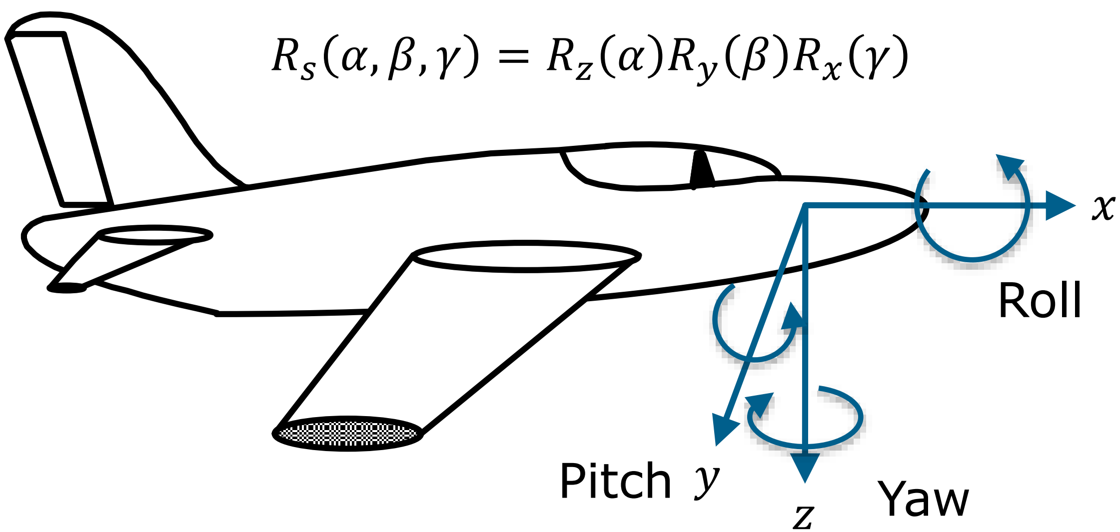
Euler-Angles: Derivation

- $\bar{R} = (R_1 A_2) A_3 = (R A_1)(A_2 A_3)$
- Let $A = A_1 A_2 A_3$, then $\bar{R} = RA$ with
- $$A = \begin{bmatrix} C\psi_1\, C\psi_3 - S\psi_1 C\psi_2 S\psi_3 & -C\psi_1 S\psi_3 - S\psi_1 C\psi_2 C\psi_3 & S\psi_1 S\psi_2 \\ S\psi_1 C\psi_3 - C\psi_1\, C\psi_2 S\psi_3 & -S\psi_1 S\psi_3 + C\psi_1 C\psi_2 C\psi_3 & -C\psi_1 S\psi_2 \\ S\psi_2 S\psi_3 & S\psi_2 C\psi_3 & C\psi_2 \end{bmatrix}$$
- Through equating coefficients it is possible to uniquely identify ψ_1, ψ_2, ψ_3 with $0 \leq \psi_1 \leq \pi$
  - $\cos\psi_2 = a_{33}$ $\quad$ $\sin\psi_1 \sin\psi_2 = a_{13}$ $\quad$ $-\sin\psi_2 \cos\psi_1 = a_{23}$
  - $\sin\psi_2 \sin\psi_3 = a_{31}$ $\quad$ $\sin\psi_2 \cos\psi_3 = a_{32}$

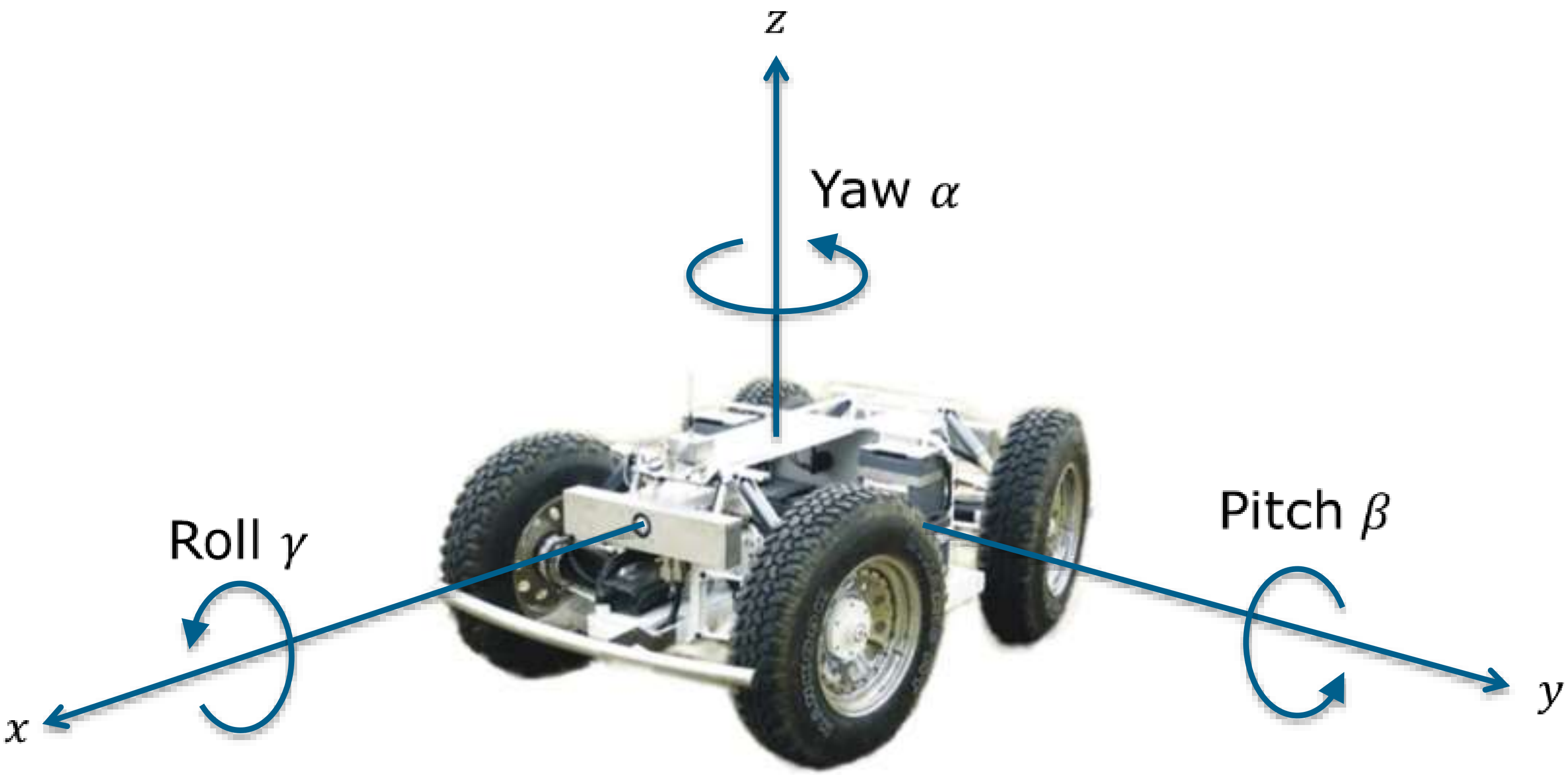
Roll-Pitch-Yaw

- Roll γ around x-axis of BCS: $R_x(\gamma)$
- Pitch β around y-axis of BCS: $R_y(\beta)$
- Yaw α around z-axis of BCS: $R_z(\alpha)$

$$R_s(\alpha, \beta, \gamma) = R_z(\alpha) R_y(\beta) R_x(\gamma)$$



Roll-Pitch-Yaw in Robotics

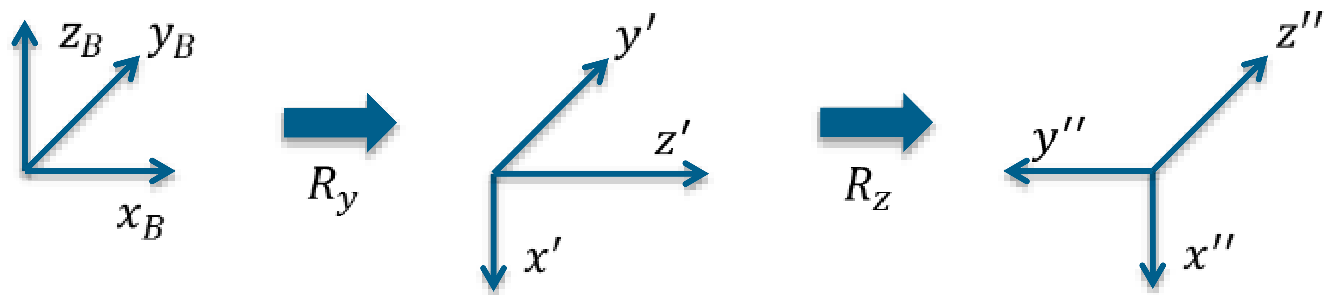


Roll-Pitch-Yaw - Rotation Matrix

- $R_S = \begin{bmatrix} C\alpha\, C\beta & C\alpha\, S\beta\, S\gamma - S\alpha\, C\gamma & C\alpha\, S\beta\, C\gamma + S\alpha\, S\gamma \\ S\alpha\, C\beta & S\alpha\, S\beta\, S\gamma + C\alpha\, C\gamma & S\alpha\, S\beta\, C\gamma - C\alpha\, S\gamma \\ -S\beta & C\beta\, S\gamma & C\beta\, C\gamma \end{bmatrix}$
- Rotation matrix R_S relative to BCS
- Rotation around unchanged axes

Roll-Pitch-Yaw - Example

- $$R_s = R_z(90^\circ) \cdot R_y(90^\circ) \cdot R_x(0^\circ)$$
  $$= \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
  $$= \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$
  $$= \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{bmatrix}$$



Representation of Orientation

- Roll-Pitch-Yaw:
  - xyz-system
  - Used in aerospace, in mobile robotics
- Euler-angles:
  - $zx'z''$-system: usual mathematical definition
  - $zy'x''$-system: programming of numerically controlled manipulators
  - $zy'z''$-system: programming language VAL, PUMA-robot

Next Lecture

- Basics of spatial kinematics
  - Homogenous coordinates
  - Transformation matrices
  - Pose calculation after several transformations