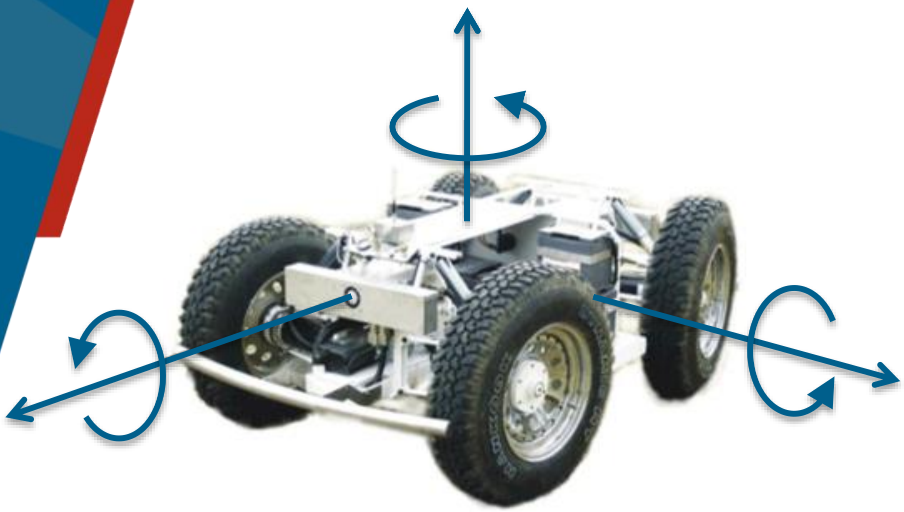
Spatial Kinematics – Foundations I



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

Contents

Notation

- Scalars: small letters, e.g. s
- Matrices: upper case letters, e.g. A
- Vectors: with an arrow, e.g. $\vec{u}$
- Identifier of scalars, vectors and points: indices at bottom right, e.g. $\overrightarrow{u_1}$
- Abbreviation of $\sin$e and $\cos$ine:
  - $\cos(\theta_1) = C\theta_1 = C_1\,, \sin(\theta_1) = S\theta_1 = S_1,$
  - $\cos(\theta_1 + \theta_2 + \cdots + \theta_n) = C_{12\ldots n},\ \sin(\theta_1 + \theta_2 + \cdots + \theta_n) = S_{12\ldots n}$
- 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. ${}^{A}_{B}R$

Reminder: Scalar Product

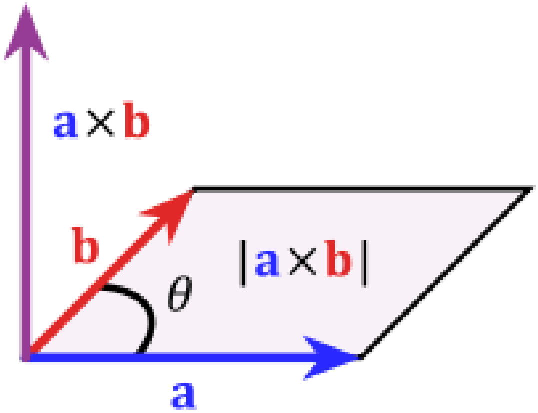
$$\vec{a} \cdot \vec{b} = \sum_{i=1}^{n} a_i b_i = a_1 \mathrm{b}_1 + \mathrm{a}_2 \mathrm{b}_2 + \cdots + \mathrm{a}_\mathrm{n} \mathrm{b}_\mathrm{n}$$

$$\vec{a} \cdot \vec{b} = |a| \cdot |b| \cdot \cos\theta$$

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

Reminder: Triple Product

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

- 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)

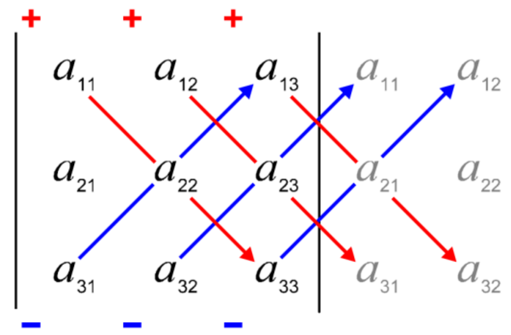
$$\det(A) = \sum_{j=1}^{n} (-1)^{i+j}\, a_{ij} \det(A_{ij})$$

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

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

- Rule of thump for 3×3-Matrices: Rule of Sarrus

$$\det \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{31}a_{22}a_{13} - a_{32}a_{23}a_{11} - a_{33}a_{21}a_{12}$$



Properties of a Determinant

- Example: Expanding the determinant along row 1:

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

$$= -3\cdot -5 + 2\cdot -3 = 15 - 6 = 9$$

- $\det A = \det A^T$
- $\det AB = \det\mathrm{A} * \det B$
- $\det \lambda A = \lambda^n \det A \quad A\ is\ a\ n \times n\ matrix$
- Determinant is 0, if
  - all elements of a row/column are 0
  - two rows are linearly dependent
- Similarity of A and B: $\mathrm{A}=\mathrm{X}^{-1}\mathrm{BX}$, $\det \mathrm{A} = \det \mathrm{B}$
- Exchanging two rows changes the sign of the determinant

Properties of Eigenvalues

- Trace of a matrix is the sum of all eigenvalues:

$$tr(A) = \sum_{i=1}^{n} \lambda_i$$

- Determinant of a matrix is the product of all eigenvalues:

$$\det(A) = \prod_{i=1}^{n} \lambda_i$$

- The eigenvectors belonging to different eigenvalues are linearly independent

Pseudo-Inverse of Matrices

- For each $m \times n$ matrix A, Pseudo-Inverse of A is defined as a $n \times m$ matrix A^+ satisfying all of the following four criteria, (Moore–Penrose conditions):
  - $AA^+A = A$

    A^+ does not need to be the general identity matrix, but it maps all column vectors of A to themselves.
  - $A^+AA^+ = A^+$

    A^+ acts like a weak inverse.
  - $(AA^+)^T = AA^+$

    AA^+ is Hermitian.
  - $(A^+A)^T = A^+A$

    A^+A is also Hermitian

Basic Properties

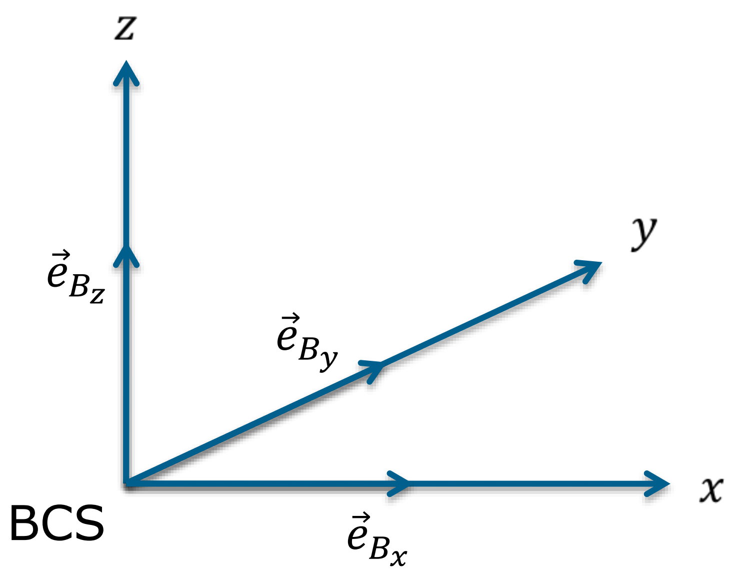
- The pseudo-inverse exists and is unique.
- The pseudo-inverse of a zero matrix is its transpose.
- If A is invertible, then its pseudoinverse is its inverse:
$$A^+ = A^{-1}$$
- The pseudo-inverse of the pseudo-inverse is the original matrix:
$$(A^+)^+ = A$$
- The pseudo-inverse of a scalar multiple of A is the reciprocal multiple of A^+:
$$(\lambda A)^+ = \frac{1}{\lambda} A^+ \quad \text{for } \lambda \neq 0$$

Description of Objects and Object Poses in E_3

Coordinate Systems

Base coordinate system (BCS)

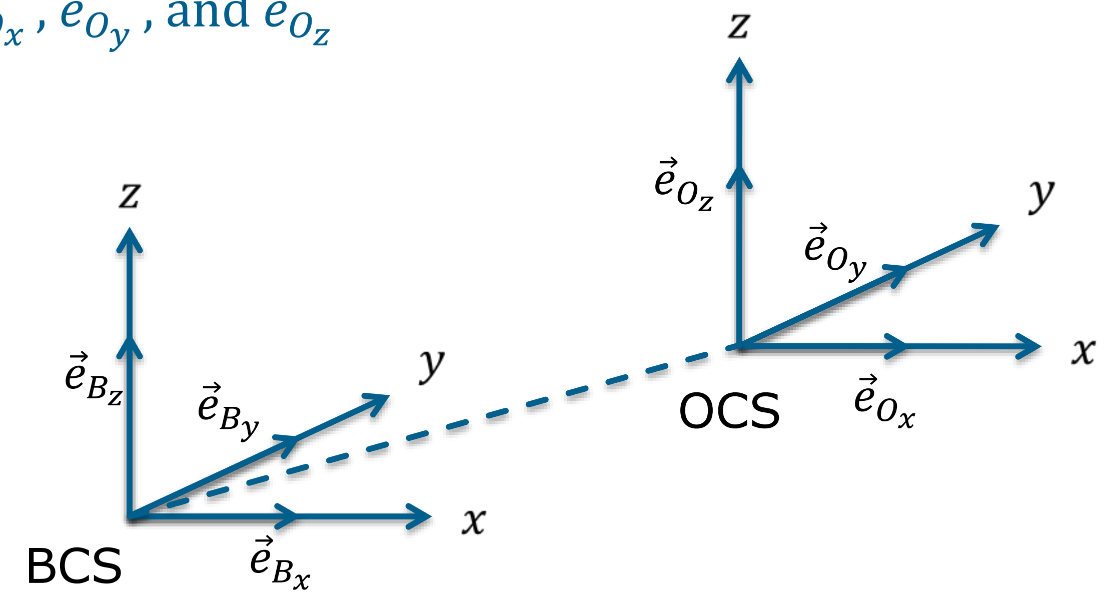
- 3-dim. coordinate system defined by orthogonal unit vectors



Coordinate Systems

Object coordinate system (OCS)

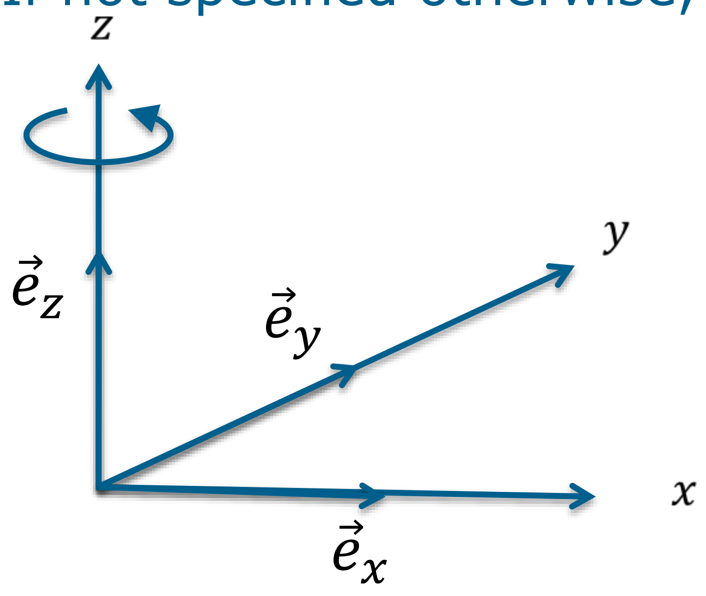
- Any rigid body can be related to a local coordinate system
- Local coordinate system is defined by orthogonal unit vectors $\vec{e}_{O_x}$, $\vec{e}_{O_y}$, and $\vec{e}_{O_z}$

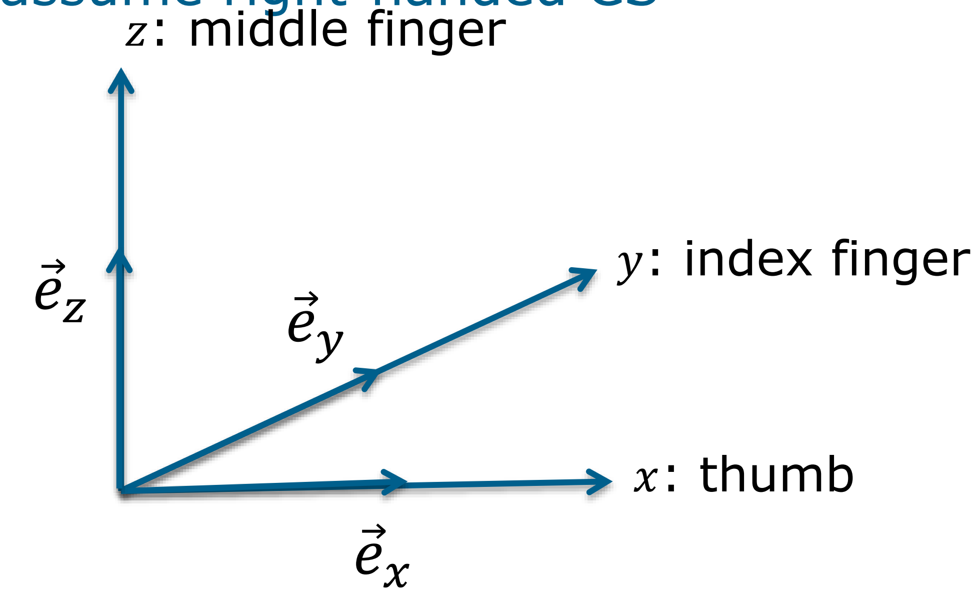


Orthogonal, Cartesian Coordinate Systems

Counterclockwise rotating coordinate system

- Right-hand-rule: Thumb x , index finger y , middle finger z
- $\vec{e}_x \times \vec{e}_y = \vec{e}_z$ with cross product $\times$
- If not specified otherwise, assume right-handed CS

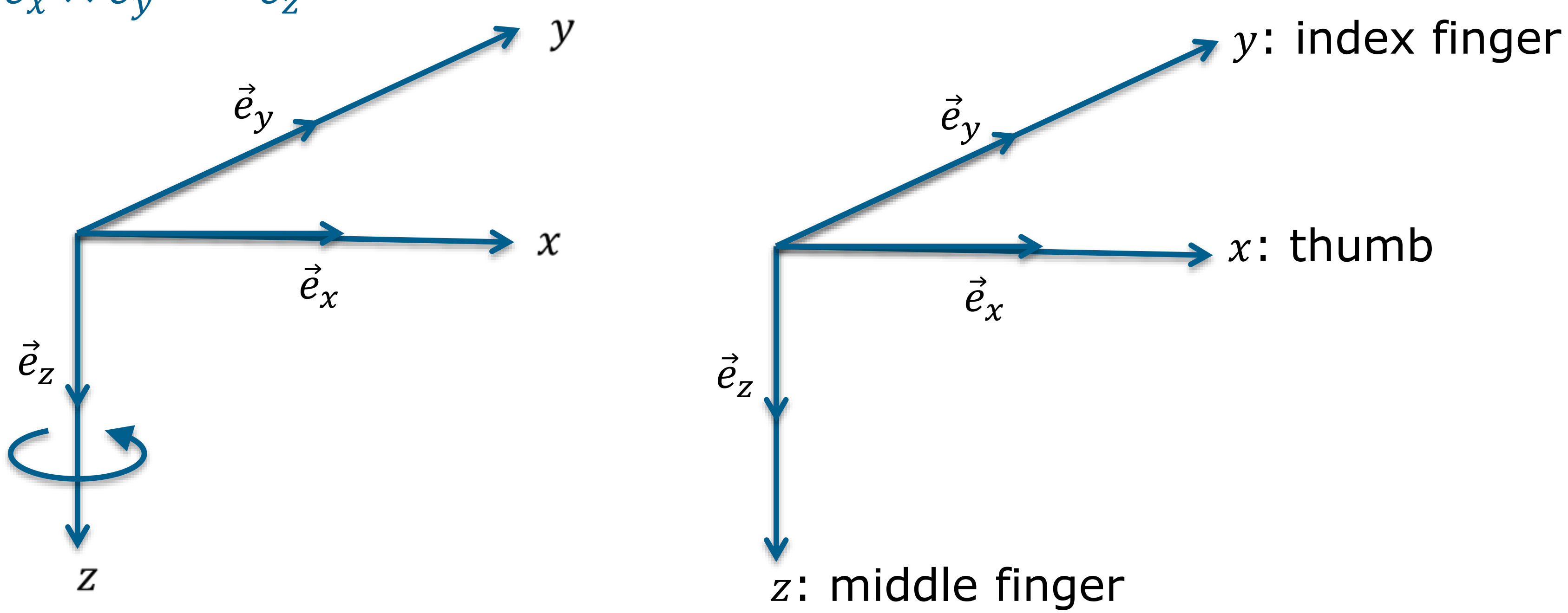




Orthogonal, Cartesian Coordinate Systems

Clockwise rotating coordinate system

- Left-hand-rule: Thumb x , index finger y , middle finger z
- $\vec{e}_x \times \vec{e}_y = -\vec{e}_z$



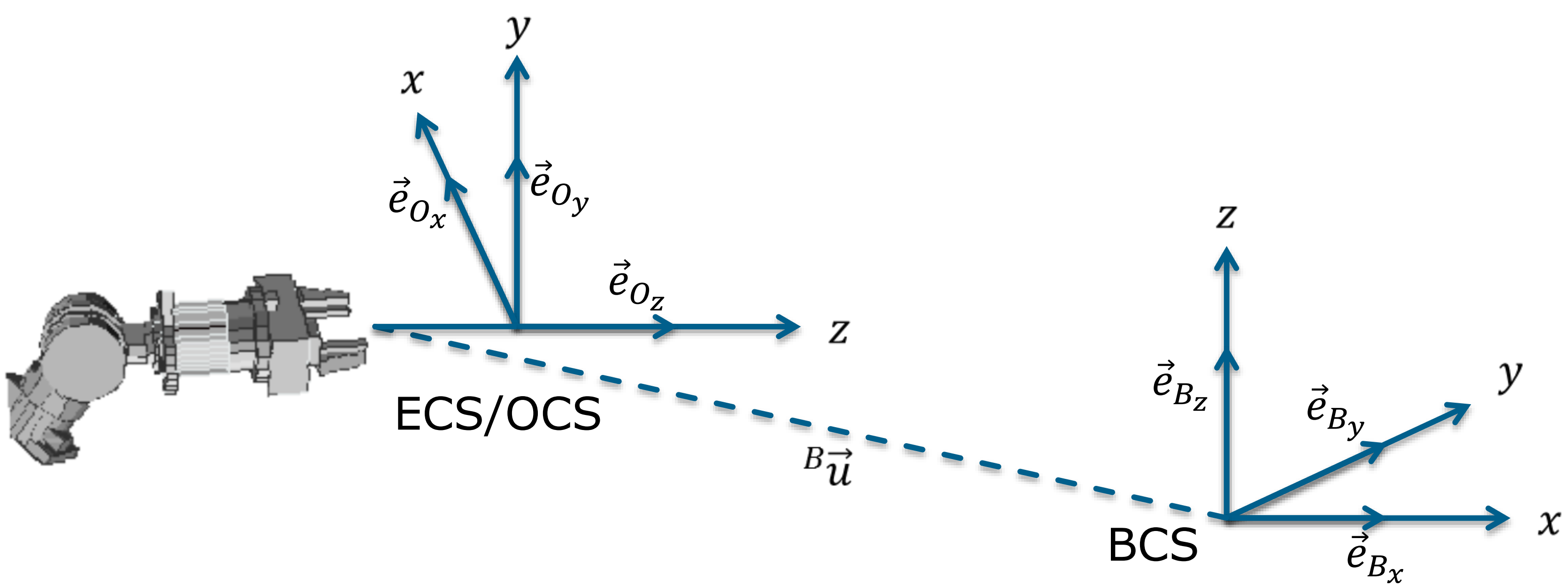
Object Poses in Space

- 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

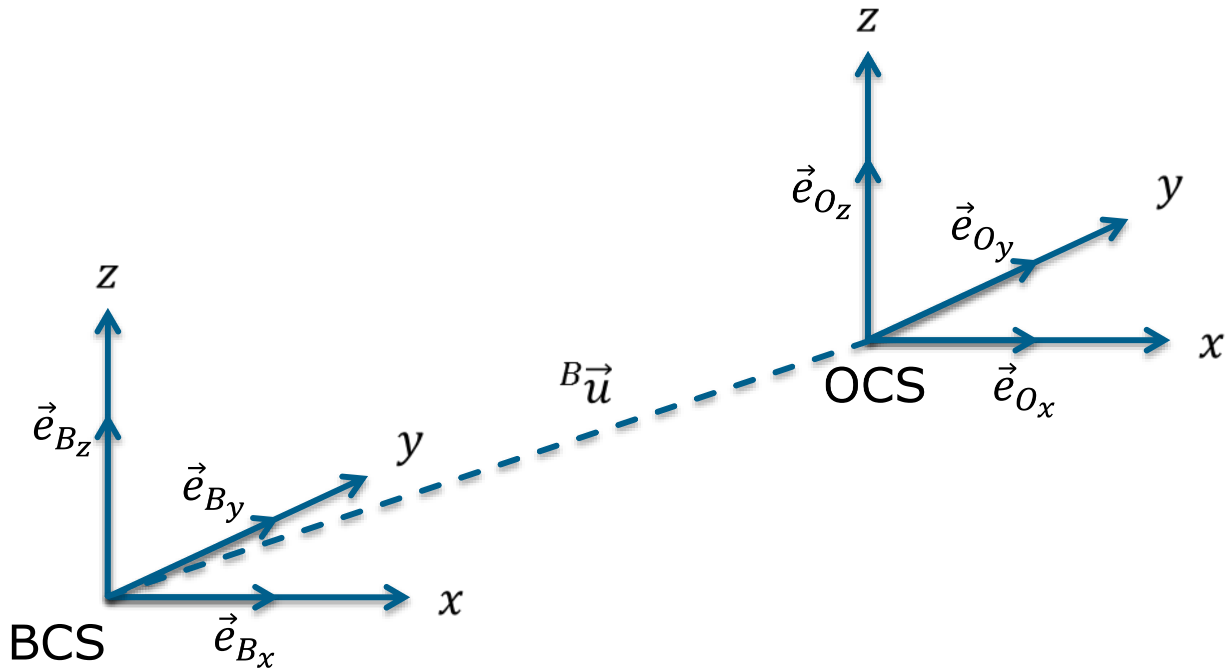
In addition to the BCS, various other local coordinate systems are used for describing robotic applications, e.g. …

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



Possible Transformations

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

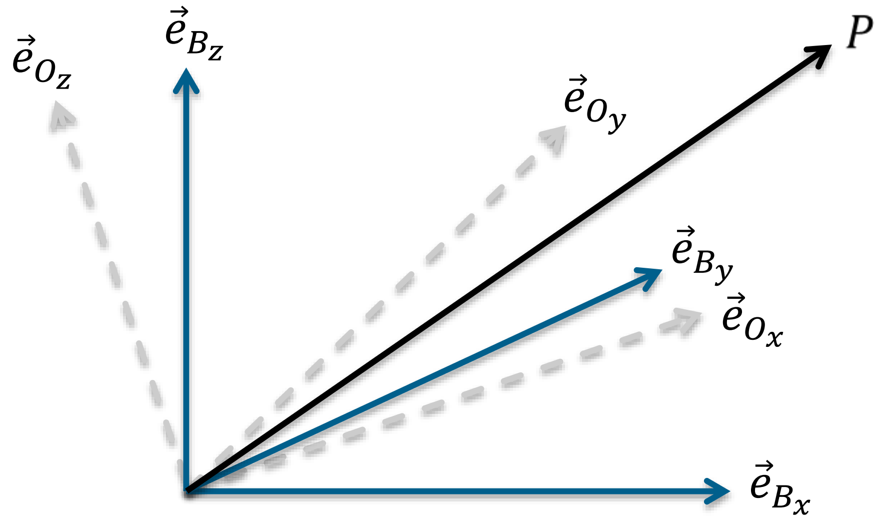


Rotation of a Coordinate System

- Let BCS and OCS be rotated against each other with unit vectors $\vec{e}_{B_x}$, $\vec{e}_{B_y}$, $\vec{e}_{B_z}$ and $\vec{e}_{O_x}$, $\vec{e}_{O_y}$, $\vec{e}_{O_z}$

- 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

- $^{B}\vec{p} = {}^{B}p_x \cdot \vec{e}_{B_x} + {}^{B}p_y \cdot \vec{e}_{B_y} + {}^{B}p_z \cdot \vec{e}_{B_z}$ with $^{B}\vec{p} = \begin{bmatrix} ^{B}p_x \\ ^{B}p_y \\ ^{B}p_z \end{bmatrix}$
- $^{O}\vec{p} = {}^{O}p_x \cdot \vec{e}_{B_x} + {}^{O}p_y \cdot \vec{e}_{B_y} + {}^{O}p_z \cdot \vec{e}_{B_z}$ 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:
  - $^{B}p_x = \vec{e}_{B_x} \cdot {}^{O}p = \vec{e}_{B_x} \cdot \vec{e}_{O_x} \cdot {}^{O}p_x + \vec{e}_{B_x} \cdot \vec{e}_{O_y} \cdot {}^{O}p_y + \vec{e}_{B_x} \cdot \vec{e}_{O_z} \cdot {}^{O}p_z$
  - $^{B}p_y = \vec{e}_{B_y} \cdot {}^{O}p = \vec{e}_{B_y} \cdot \vec{e}_{O_x} \cdot {}^{O}p_x + \vec{e}_{B_y} \cdot \vec{e}_{O_y} \cdot {}^{O}p_y + \vec{e}_{B_y} \cdot \vec{e}_{O_z} \cdot {}^{O}p_z$
  - $^{B}p_z = \vec{e}_{B_z} \cdot {}^{O}p = \vec{e}_{B_z} \cdot \vec{e}_{O_x} \cdot {}^{O}p_x + \vec{e}_{B_z} \cdot \vec{e}_{O_y} \cdot {}^{O}p_y + \vec{e}_{B_z} \cdot \vec{e}_{O_z} \cdot {}^{O}p_z$

Rotation of a Coordinate System

- Transformation from BCS to OCS coordinates:
  - $^{O}p_x = \vec{e}_{O_x} \cdot {}^{B}p = \vec{e}_{O_x} \cdot \vec{e}_{B_x} \cdot {}^{B}p_x + \vec{e}_{O_x} \cdot \vec{e}_{B_y} \cdot {}^{B}p_y + \vec{e}_{O_x} \cdot \vec{e}_{B_z} \cdot {}^{B}p_z$
  - $^{O}p_y = \vec{e}_{O_y} \cdot {}^{B}p = \vec{e}_{O_y} \cdot \vec{e}_{B_x} \cdot {}^{B}p_x + \vec{e}_{O_y} \cdot \vec{e}_{B_y} \cdot {}^{B}p_y + \vec{e}_{O_y} \cdot \vec{e}_{B_z} \cdot {}^{B}p_z$
  - $^{O}p_z = \vec{e}_{O_z} \cdot {}^{B}p = \vec{e}_{O_z} \cdot \vec{e}_{B_x} \cdot {}^{B}p_x + \vec{e}_{O_z} \cdot \vec{e}_{B_y} \cdot {}^{B}p_y + \vec{e}_{O_z} \cdot \vec{e}_{B_z} \cdot {}^{B}p_z$

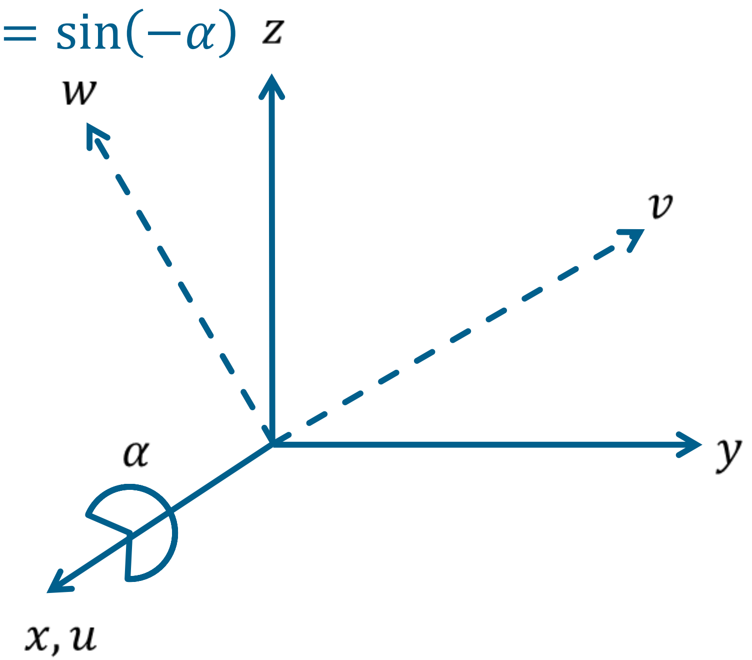
Matrix Notation

- $${}^{B}_{O}R_1 = \begin{bmatrix} \vec{e}_{B_x} \cdot \vec{e}_{O_x} & \vec{e}_{B_x} \cdot \vec{e}_{O_y} & \vec{e}_{B_x} \cdot \vec{e}_{O_z} \\ \vec{e}_{B_y} \cdot \vec{e}_{O_x} & \vec{e}_{B_y} \cdot \vec{e}_{O_y} & \vec{e}_{B_y} \cdot \vec{e}_{O_z} \\ \vec{e}_{B_z} \cdot \vec{e}_{O_x} & \vec{e}_{B_z} \cdot \vec{e}_{O_y} & \vec{e}_{B_z} \cdot \vec{e}_{O_z} \end{bmatrix} \text{ and } {}^{O}\vec{p} = \begin{bmatrix} {}^{O}p_x \\ {}^{O}p_y \\ {}^{O}p_z \end{bmatrix}$$
- $${}^{O}_{B}R_2 = \begin{bmatrix} \vec{e}_{O_x} \cdot \vec{e}_{B_x} & \vec{e}_{O_x} . \vec{e}_{B_y} & \vec{e}_{O_x} \cdot \vec{e}_{B_z} \\ \vec{e}_{O_y} \cdot \vec{e}_{B_x} & \vec{e}_{O_y} \cdot \vec{e}_{B_y} & \vec{e}_{O_y} \cdot \vec{e}_{B_z} \\ \vec{e}_{O_z} \cdot \vec{e}_{B_x} & \vec{e}_{O_z} \cdot \vec{e}_{B_y} & \vec{e}_{O_z} \cdot \vec{e}_{B_z} \end{bmatrix} \text{ and } {}^{B}\vec{p} = \begin{bmatrix} {}^{B}p_x \\ {}^{B}p_y \\ {}^{B}p_z \end{bmatrix}$$
- $${}^{B}\vec{p} = {}^{B}_{O}R_1 \cdot {}^{O}\vec{p} = {}^{O}_{B}R_2{}^{-1} \cdot {}^{O}\vec{p}$$
- $${}^{O}\vec{p} = {}^{O}_{B}R_2 \cdot {}^{B}\vec{p} = {}^{B}_{O}R_1{}^{-1} \cdot {}^{B}\vec{p}$$
- 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

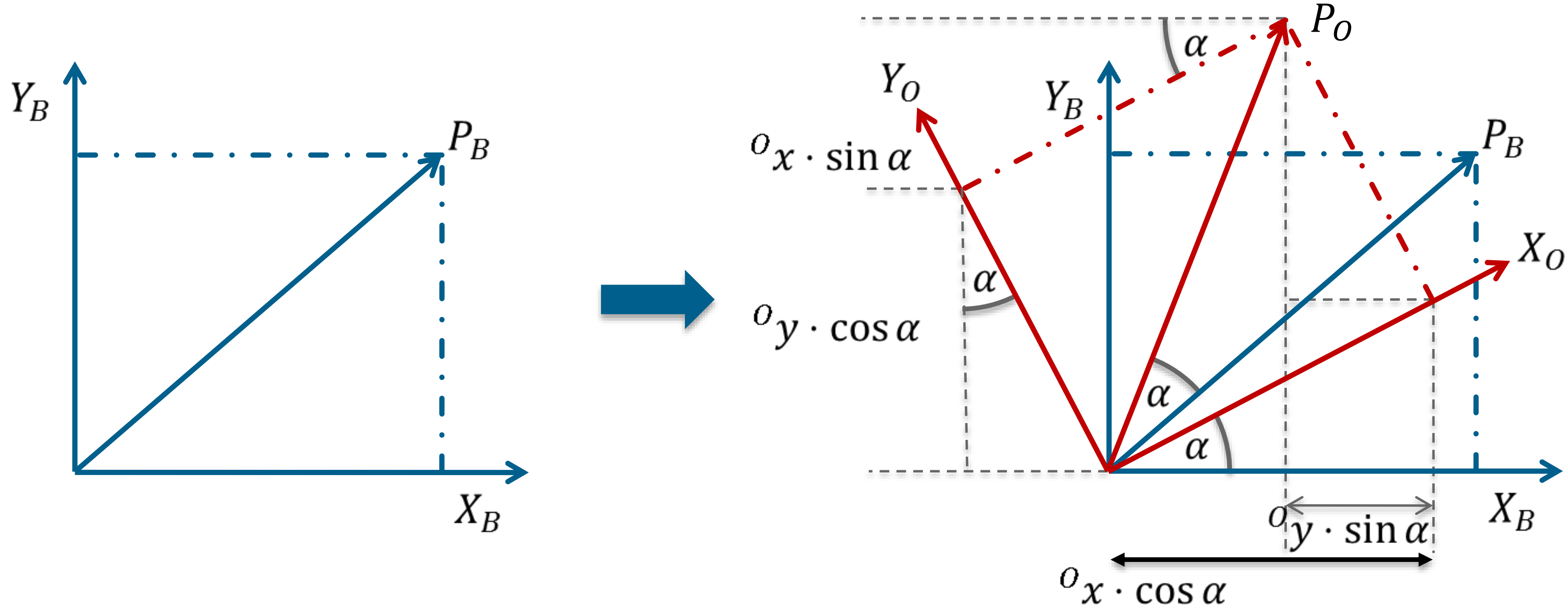
Using scalar product: $\vec{a} \cdot \vec{b} = |\vec{a}| \cdot |\vec{b}| \cos\alpha$

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



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
  - $R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$
  - $R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$

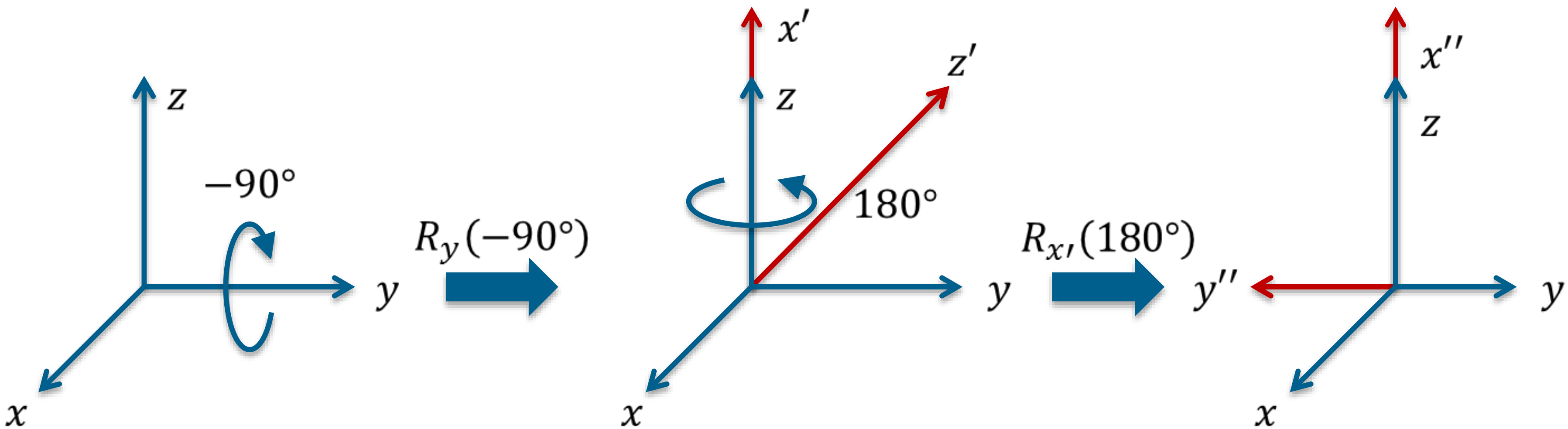
Rotation Matrix - Properties

- 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 $R_g(R) = 3$
  - $R^T = R^{-1}$
  - $R \cdot R^{-1} = R^{-1} \cdot R = I$
  - $\det R = 1$

Several Elementary Rotations

Basic Rotations:

- Let OCS result based on 2 rotations from BCS



BCS OCS1 OCS2

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

Vector Coordinates due to a new Frame

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

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} {}^{O2}\vec{u}_1 \\ {}^{O2}\vec{u}_2 \\ {}^{O2}\vec{u}_3 \end{bmatrix} = \begin{bmatrix} {}^{O2}\vec{u}_3 \\ -{}^{O2}\vec{u}_2 \\ {}^{O2}\vec{u}_1 \end{bmatrix}$$

- ${}^{B}\vec{e}_{O2_x} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$
- ${}^{B}\vec{e}_{O2_y} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$
- ${}^{B}\vec{e}_{O2_z} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

Interpretation of several, elementary Rotations

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

- Post-multiplication $\big((\cdots (R_1 R_2) \cdots R_{n-1}) R_n\big)$:
  - 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 unique axis
    - Trade-off between others
  - Euler angels
    - Follows chained Joint-Setup
  - Roll-Pitch-Yaw
    - Easy to interpret by humans
  - Quaternions
    - Computationally fast
  - Exponential coordinates
    - More similar to its kinematic

Rotation around unique Axis

- Instead of rotation with BCS-axis, rotate around unique 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:

- Given a transformation matrix $\mathrm{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$ we can obtain:
- $\hat{g} = \frac{1}{2S\theta}(R - R^T)$, and $\theta = \cos^{-1}\left(\frac{tr(R)-1}{2}\right) \in [0, \pi]$.

Hence,

- $R = R_{\vec{g}}(\theta) = \begin{bmatrix} g_1^2\eta\theta + C\theta & g_1 g_2\eta\theta - g_3 S\theta & g_1 g_3\eta\theta + g_2 S\theta \\ g_1 g_2\eta\theta + g_3 S\theta & g_2^2\eta\theta + C\theta & g_2 g_3\eta\theta - g_1 S\theta \\ g_1 g_3\eta\theta - g_2 S\theta & g_2 g_3\eta\theta + g_1 S\theta & g_3^2\eta\theta + C\theta \end{bmatrix}$

with:

$$\mathrm{S}\theta = \sin\theta,\ \ C\theta = \cos\theta,\ \ \eta\theta = 1 - \cos\theta\ \ , \vec{g} = (g_1, g_2, g_3)^T = \frac{1}{2S\theta}\begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$$

Rodrigues Formula:

- $\hat{g}$ is the skew-symmetric matrix corresponding to the vector $\vec{g}$:

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

- The matrix R can be decomposed to

$$R = R_{\vec{g}}(\theta) = \cos(\theta)\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + (1-\cos(\theta))\begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix}\begin{bmatrix} g_1 & g_2 & g_3 \end{bmatrix} + \sin(\theta)\begin{bmatrix} 0 & -g_3 & g_2 \\ g_3 & 0 & -g_1 \\ -g_2 & g_1 & 0 \end{bmatrix}$$

- To be equal to:

$$R = R_{\vec{g}}(\theta) = C\theta I_3 + \left(1-cos(\theta)\right)\vec{g}\vec{g}^T + S\theta\,\hat{g}$$

THE ROBOTICS RESEARCH LAB

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:

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

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

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 $\lambda_i$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:

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

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

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

Computation of Roll-Pitch-Yaw-Angles

- Multiplying from the right with $R_x(\alpha)^{-1}$:
$$R_z(\gamma) \cdot R_y(\beta) \cdot R_x(\alpha) \cdot R_x(\alpha)^{-1} = R \cdot R_x(\alpha)^{-1}$$
- Simplified: $R_z(\gamma) \cdot R_y(\beta) = R \cdot R_x(\alpha)^T$
- → Exercise

Roll-Pitch-Yaw-Angles - an Example

Matrix from slides 16-17 gives the following equations:

$$\begin{bmatrix} C\beta & 0 & S\beta \\ S\beta \cdot S\alpha & C\alpha & -S\alpha \cdot C\beta \\ -C\alpha \cdot S\beta & S\alpha & C\alpha \cdot C\beta \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ C\gamma & S\gamma & 0 \\ -S\gamma & C\gamma & 0 \end{bmatrix}$$

Roll-Pitch-Yaw-Angles - an Example

Or equivalently:

$$
\begin{array}{ll}
(1.1) & C\beta = 0 \\
(1.2) & 0 = 0 \\
(1.3) & S\beta = 1 \\
(2.1) & S\beta \cdot S\alpha = C\gamma \\
(2.2) & C\alpha = S\gamma \\
(2.3) & -S\alpha \cdot C\beta = 0 \\
(3.1) & -C\alpha \cdot S\beta = -S\gamma \\
(3.2) & S\alpha = C\gamma \\
(3.3) & C\alpha \cdot C\beta = 0
\end{array}
$$

Roll-Pitch-Yaw-Angles: An Example

- Angle β: From $(1.1), (1.3)$ it follows that

$$\beta = 90°$$

- Angle α and γ: From $(2.2), (3.2)$ it follows that

$$\gamma = 90° - \alpha$$

- With $\beta = 90°$ you can simplify $(2.1), (2.3), (3.1), (3.3)$ to (2.2) and (3.2)
- No equations for α or γ:
  - α can be chosen - γ arbitrarily
- Choose $\alpha = 0°$ → Solutions $(0°, 90°, 90°)$

Axes of Rotation in Robotics

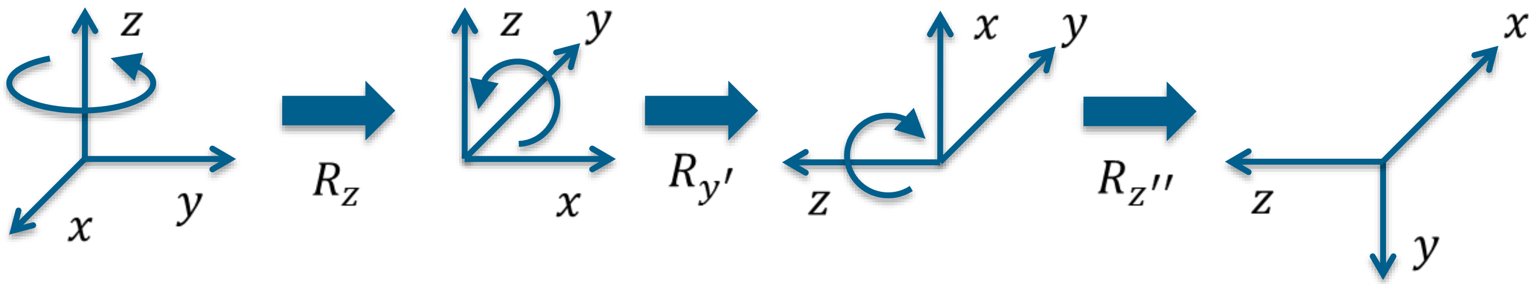
- Rotation axes usually BCS
- Convention of rotation axes and their order usually in …
  - Euler-angles
  - Roll, Pitch, Yaw

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 $R_{z,\alpha}, R_{y',\beta}, R_{z'',\gamma}$

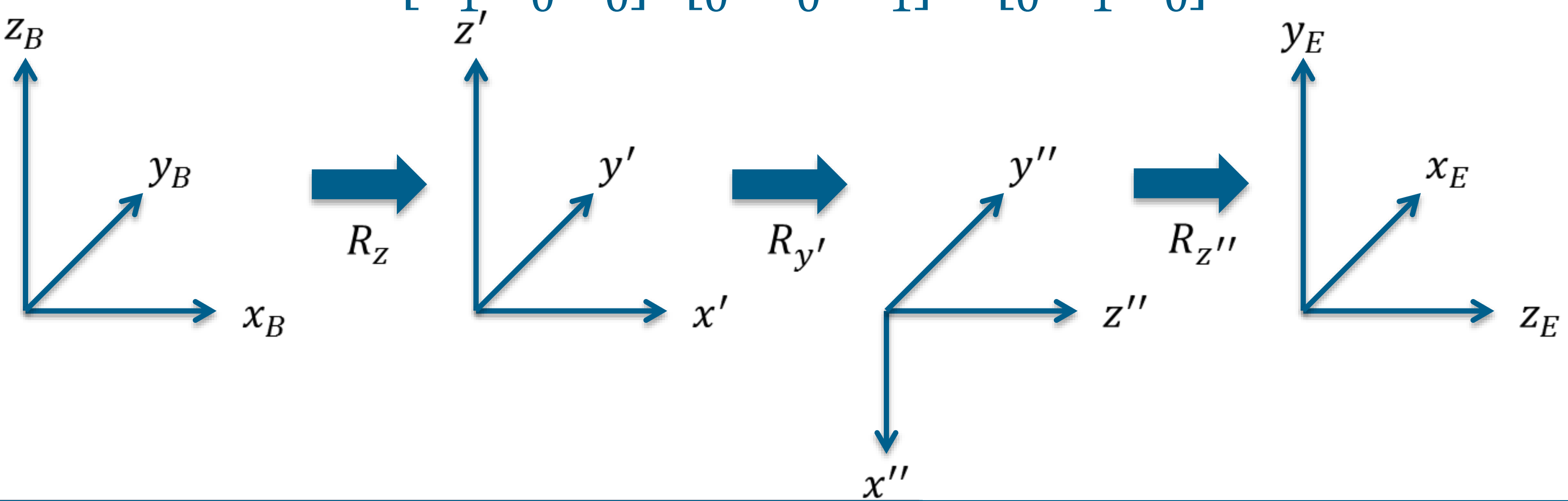


Euler-Angles - Example

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

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

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



Euler-Angles: Derivation

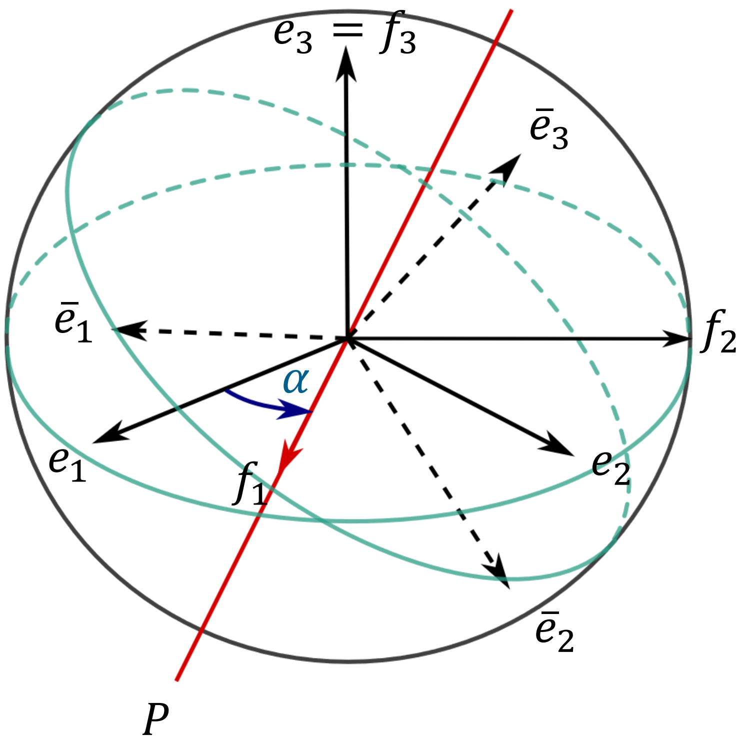
Theorem: If two right-handed Cartesian coordinate systems $R = \{U, e_1, e_2, e_3\}$ and $\{U, \bar{e}_1, \bar{e}_2, \bar{e}_3\}$ with a common origin exist, then there exists an orthogonal matrix A that maps R to $\bar{R}$

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

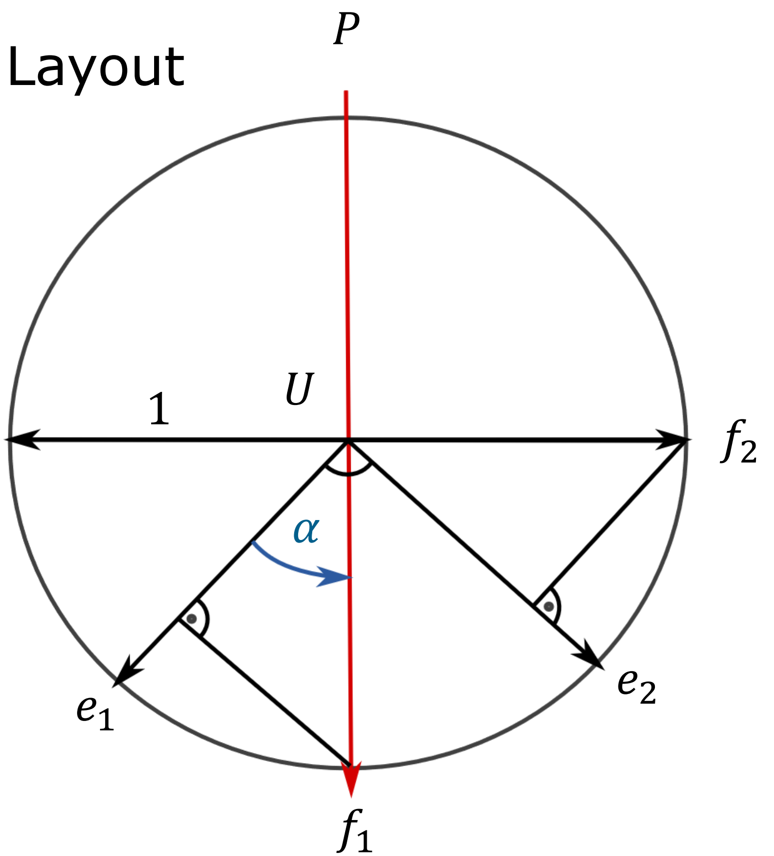
Euler-Angles: Derivation

1. Rotation around $e_3$with the positive angle α so that e_1 is mapped onto f_1
   - f_1, constructed by positive rotation with α with $0 \leq \alpha \leq \pi$, lies on P
   - R transforms into $R = \{U, f_1, f_2, f_3 = e_3\}$
   - $A_1 = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} R_1 = RA_1$
   - $f_1 \perp e_3$ and $f_1 \perp \bar{e}_3$

Euler-Angles - Coordinate Systems



Layout

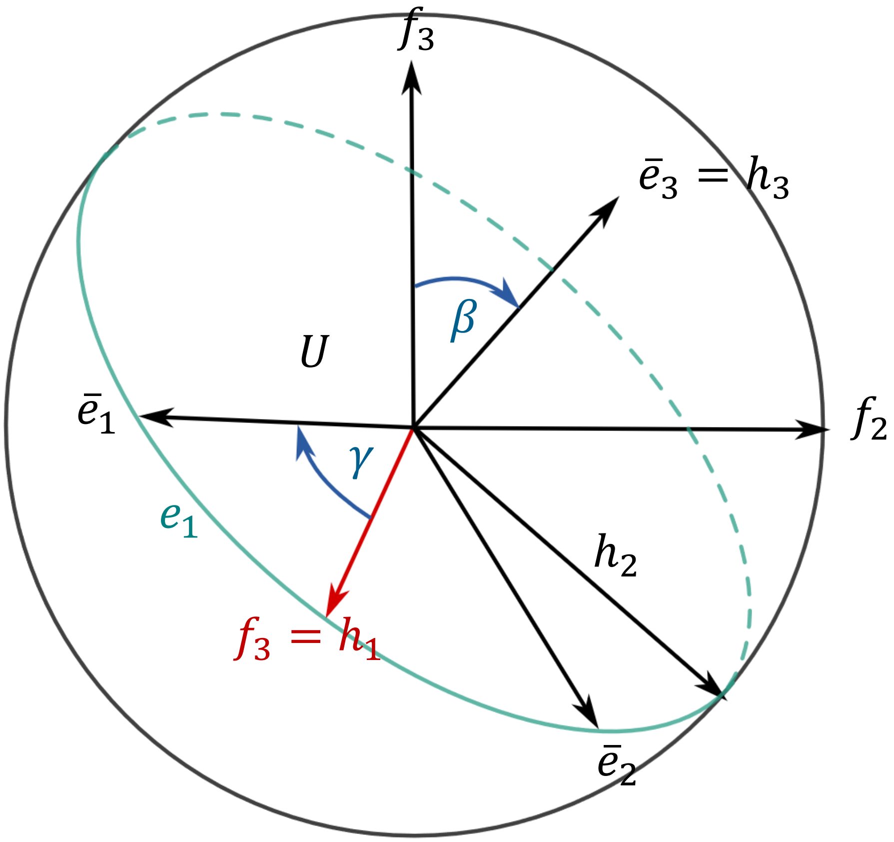


- Plane E_1 (spanned by e_1 and e_2) intersects E_2 (spanned by $\bar{e}_1$and $\bar{e}_2$) in line P.

Euler-Angles: Derivation

2. Rotate R_1 around axis $f_1$with angle α so that $e_3 = \bar{e}_3$ falls together with $\bar{e}_3$
   - R transforms to $R_2 = \{U, f_1 = h_1, h_2, h_3 \ = \bar{e}_3\}$
   - $A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\beta) & -\sin(\beta) \\ 0 & \sin(\beta) & \cos(\beta) \end{bmatrix} R_2 = R_1 A_2$
   - f_2 is mapped onto h_2
   - h_2 lies in the plane spanned by $\bar{e}_1$and $\bar{e}_2$

Euler-Angles - Coordinate Systems



Euler-Angles: Derivation

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

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

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
- $A = \begin{bmatrix} C\alpha\, C\gamma - S\alpha C\beta S\gamma & -C\alpha S\gamma - S\alpha C\beta C\gamma & S\alpha S\beta \\ S\alpha C\delta - C\alpha\, C\beta S\gamma & -S\alpha S\gamma + C\alpha C\beta C\gamma & -C\alpha S\beta \\ S\beta S\gamma & S\beta C\gamma & C\beta \end{bmatrix}$

- Through equating coefficients it is possible to uniquely identify α, β, γ with $0 \leq \alpha \leq \pi$
  - $a_{13} = \sin\alpha \sin\beta \quad a_{23} = -\sin\beta\cos\alpha \quad a_{33} = \cos\beta$
  - $a_{31} = \sin\beta\sin\gamma \quad a_{32} = \sin\beta\cos\gamma$

Rotation Axis and Angle of Rotation

- Every orthogonal 3×3 matrix $A = (a_{ik})$ with $\det(A) = 1$ describes a rotation around an axis g by a rotation angle α.
- The following applies to the angle of rotation:

$$\cos(\alpha) = \frac{1}{2}(a_{11} + a_{22} + a_{33})$$

- And, if $\alpha \neq 0^{\circ}$ and $\alpha \neq 180^{\circ}$, the axis of rotation g is determines by:

$$g_1 = (a_{32} - a_{23})$$
$$g_2 = (a_{13} - a_{31})$$
$$g_3 = (a_{21} - a_{12})$$

Rodriques´ Theorem

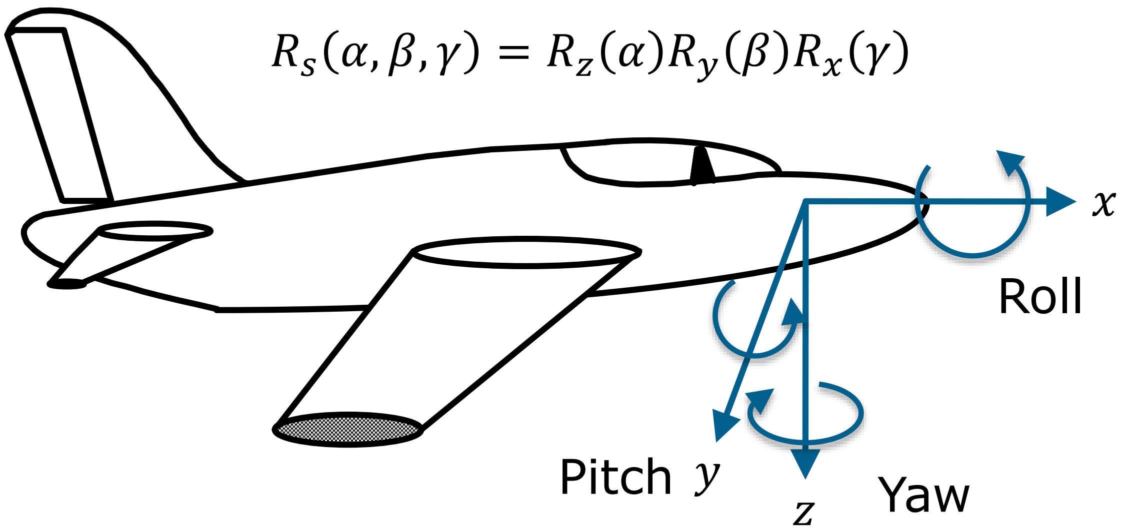
- Rotation of the vector $\vec{q}$ around the axis, which is described by the vector $\vec{k}$,with the angle α.

$$\vec{q}\,' = \vec{q}\cos(\alpha) + \sin(\alpha)\left(\vec{k} \times \vec{q}\right) + (1 - \cos(\alpha))\left(\vec{k}.\vec{q}\right) \times \vec{k}$$

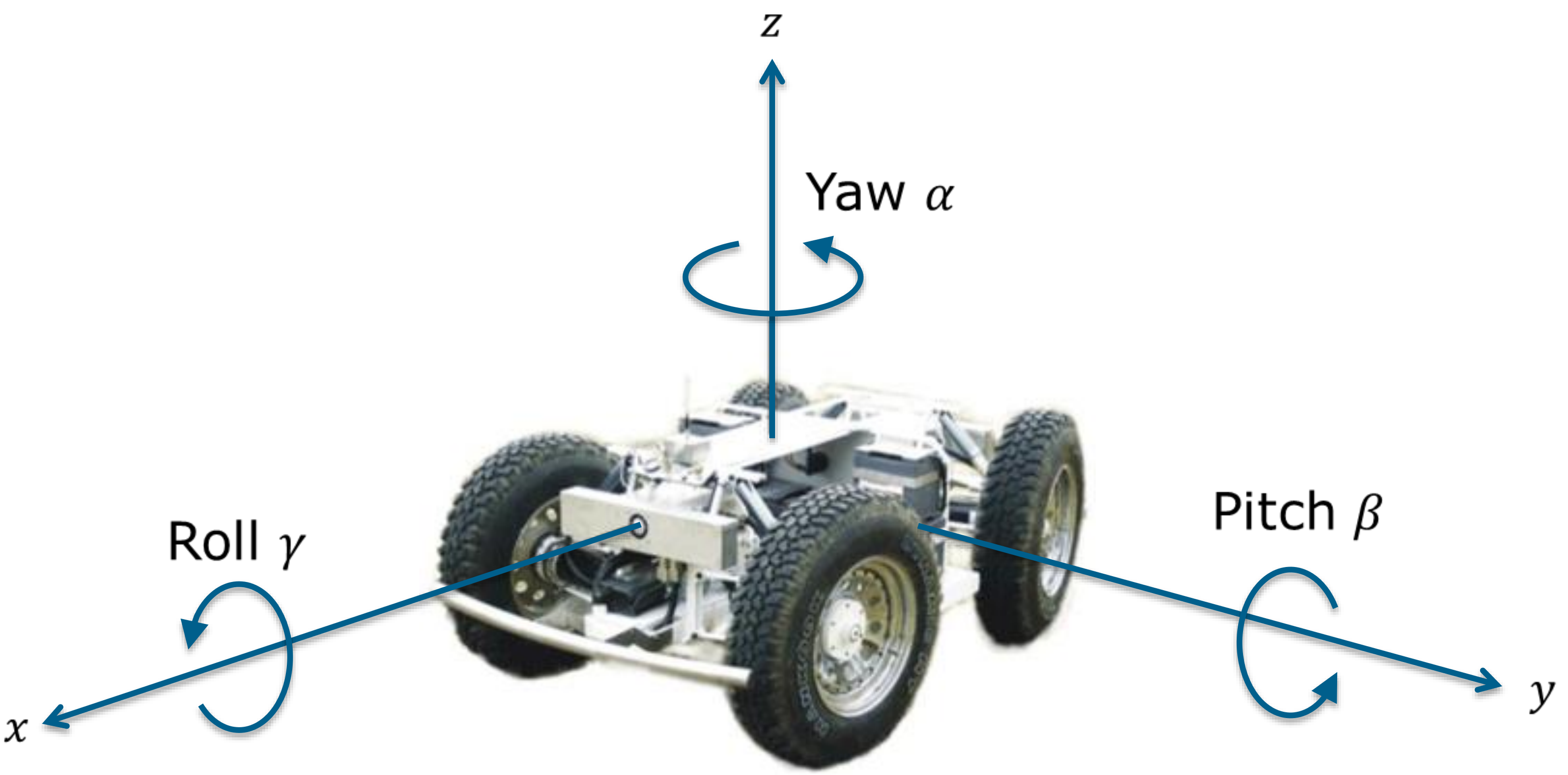
Roll-Pitch-Yaw

- 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

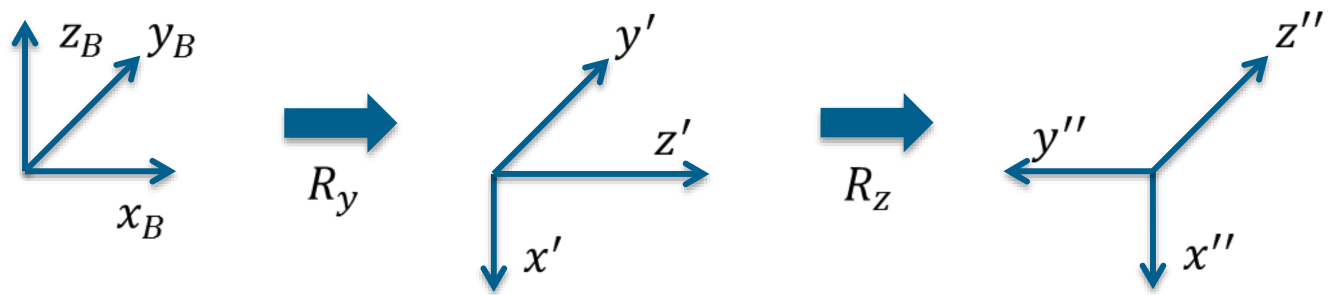


Roll-Pitch-Yaw - Rotation Matrix

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

Roll-Pitch-Yaw - Example

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



Coming up next ...

Object pose in a 3D Euclidian space ($E3$)