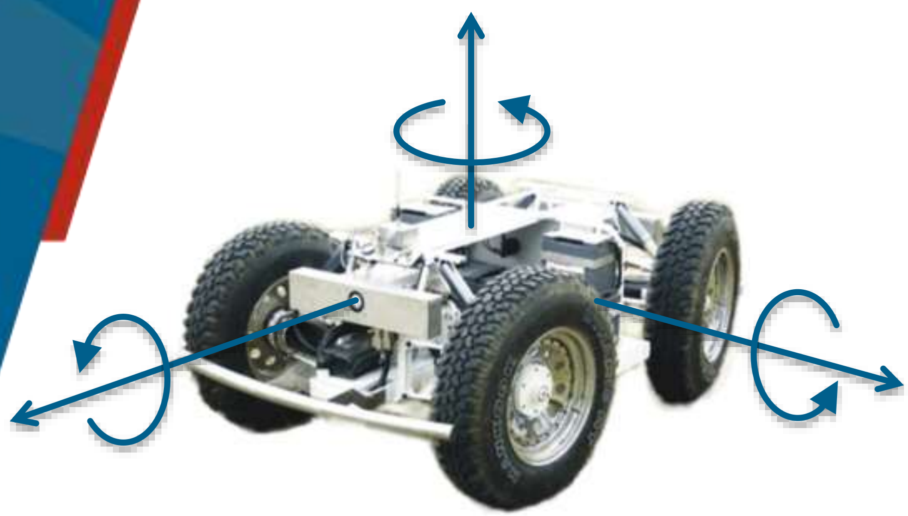
Spatial Kinematics II



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

Content

- Object pose in a 3D Euclidian space (E_3)
- Describing orientations with 3×3 matrices
- 6-dimensional description vectors
- Homogeneous coordinate transformations and transformation matrices
- Sequence of rotations
- Quaternions

Orientation of a Rigid Body

- Every orientation of a rigid body in E_3 is reachable by 3 rotations around the axes of the coordinate system
- Every rotation around a axis can be constructed as a 3×3 rotation matrix
- Composition of rotations by multiplying the corresponding matrices
- Matrix multiplication has no commutative but an associative property
- Interpret $R_1 \cdot R_2 \cdots R_n$ from left to right
  - Rotation of R_i according to the coordinate system defined by $R_1 \cdot \ldots \cdot R_{i-1}$
  - R_1 rotates the BCS
- Euler angles: $R_S = R_z(\alpha) \cdot R_{y'}(\beta) \cdot R_{z''}(\gamma)$

6-Dimensional Description Vector

- The pose of an object in E_3 can be described by a 6-tuple $(x, y, z, \alpha, \beta, \gamma) \in \mathbb{R}^6$

$$ {}^{B}\vec{u} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} + {}^{B}_{O}R\,(\alpha, \beta, \gamma)\; {}^{O}\vec{u} $$

- x, y, z: Coordinates of the origin of OCS relative to the BCS (describing position)
- α, β, γ: Angle of rotation, corresponding to the axes of rotation (describing orientation)
- Position vector and rotation matrix are intuitive and mainly used for the pose description of objects and end effectors
- Cons: Vector and matrix operators are separate

Homogeneous Coordinate Transform

- Replacement of translations and rotations operators through homogeneous matrices

  -> Transformation from Cartesian to homogeneous coordinates

- Let P be a point with Cartesian coordinates (p_x, p_y, p_z), let $s \in \mathbb{R}$ then $P' = (sp_x, sp_y, sp_z, s) \in \mathbb{R}^4$ is the representation of P in homogeneous coordinates
- For each P there is an infinite number of homogeneous points P'

Homogeneous Coordinates

- Cartesian coordinates of a homogeneous point

$P\ (x, y, z, s)$ are $\left(\frac{x}{s}, \frac{y}{s}, \frac{z}{s}\right)$

- Homogeneous coordinates in E_3 allows to create 4×4 transformation matrices containing

rotation, translation, scaling and perspective transforms

- In robotics, $s = 1$

Object Pose in Homogeneous Coordinates

- Scaling factor s
- Perspective transformation P
  - Here $(0,0,0)^T$
- Translation vector $u = (u_x, u_y, u_z)$
  - Position: Origin of OCS relative to BCS
- Rotations $R = (n_x, n_y, n_z), (o_x, o_y, o_z), (a_x, a_y, a_z)$
  - Orientation: 3 unit vectors in x-, y- and z-direction of the OCS

$$A = \begin{bmatrix} R & \vec{u} \\ \vec{P} & s \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & u_x \\ R_{21} & R_{22} & R_{23} & u_y \\ R_{31} & R_{32} & R_{33} & u_z \\ P_1 & P_2 & P_3 & s \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & u_x \\ n_y & o_y & a_y & u_y \\ n_z & o_z & a_z & u_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Homogeneous Translation Matrix T

- Let p, p' be position vectors in homogeneous coordinates
- Let a be a homogeneous translation vector where $a = (a_x, a_y, a_z, 1)^T$
- A Cartesian translation $p = a + p'$ can be represented by the translation matrix T:

$$p = T(a_x, a_y, a_z)p' \text{with T} = \begin{bmatrix} 1 & 0 & 0 & a_x \\ 0 & 1 & 0 & a_y \\ 0 & 0 & 1 & a_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow T(a_x, a_y, a_z)^{-1} = \begin{bmatrix} 1 & 0 & 0 & -a_x \\ 0 & 1 & 0 & -a_y \\ 0 & 0 & 1 & -a_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Local and Global Scaling

- Local (anisotropic) scaling

$$\begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} ax \\ by \\ cz \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} ax \\ by \\ cz \end{bmatrix}$$

- Global (isotropic) scaling

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & s \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ s \end{bmatrix} \rightarrow \begin{bmatrix} \frac{x}{s} \\ \frac{y}{s} \\ \frac{z}{s} \end{bmatrix}$$

  - $s > 1$ down scaling
  - $s < 1$ up scaling

Homogeneous Rotation Matrices

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

Inverse Homogeneous Rotation Matrix R^{-1}

Let R_3 be a 3×3 rotation matrix, then:

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow R^{-1} = \begin{bmatrix} r_{11} & r_{21} & r_{31} & 0 \\ r_{12} & r_{22} & r_{32} & 0 \\ r_{13} & r_{23} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Inverse Homogeneous Matrix A

$$A = T(a_x, a_y, a_z)R = \begin{bmatrix} r_{11} & r_{12} & r_{13} & a_x \\ r_{21} & r_{22} & r_{23} & a_y \\ r_{31} & r_{32} & r_{33} & a_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

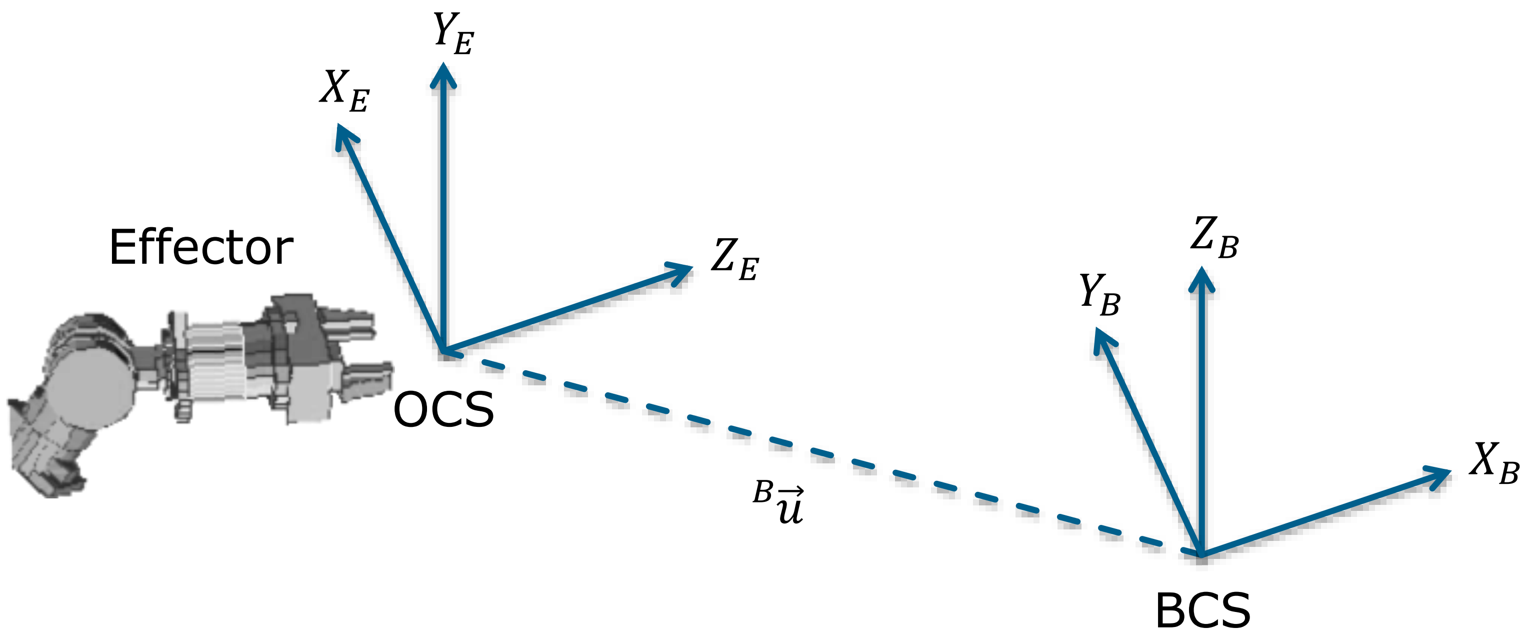
$$\Rightarrow A^{-1} = R^{-1}T^{-1} = \begin{bmatrix} r_{11} & r_{21} & r_{31} & 0 \\ r_{12} & r_{22} & r_{32} & 0 \\ r_{13} & r_{23} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -a_x \\ 0 & 1 & 0 & -a_y \\ 0 & 0 & 1 & -a_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} r_{11} & r_{21} & r_{31} & -r_{11}a_x - r_{21}a_y - r_{31}a_z \\ r_{12} & r_{22} & r_{32} & -r_{12}a_x - r_{22}a_y - r_{32}a_z \\ r_{13} & r_{23} & r_{33} & -r_{13}a_x - r_{23}a_y - r_{33}a_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Properties of Homogeneous Matrices

- In a homogeneous 4×4 matrix there are 12 $(\vec{n}, \vec{o}, \vec{a}, \vec{u})$ nontrivial parameters, but only 6 $(x, y, z, \alpha, \beta, \gamma)$ are necessary
- Redundancy because of orthogonality

Homogenous Transformation Matrix



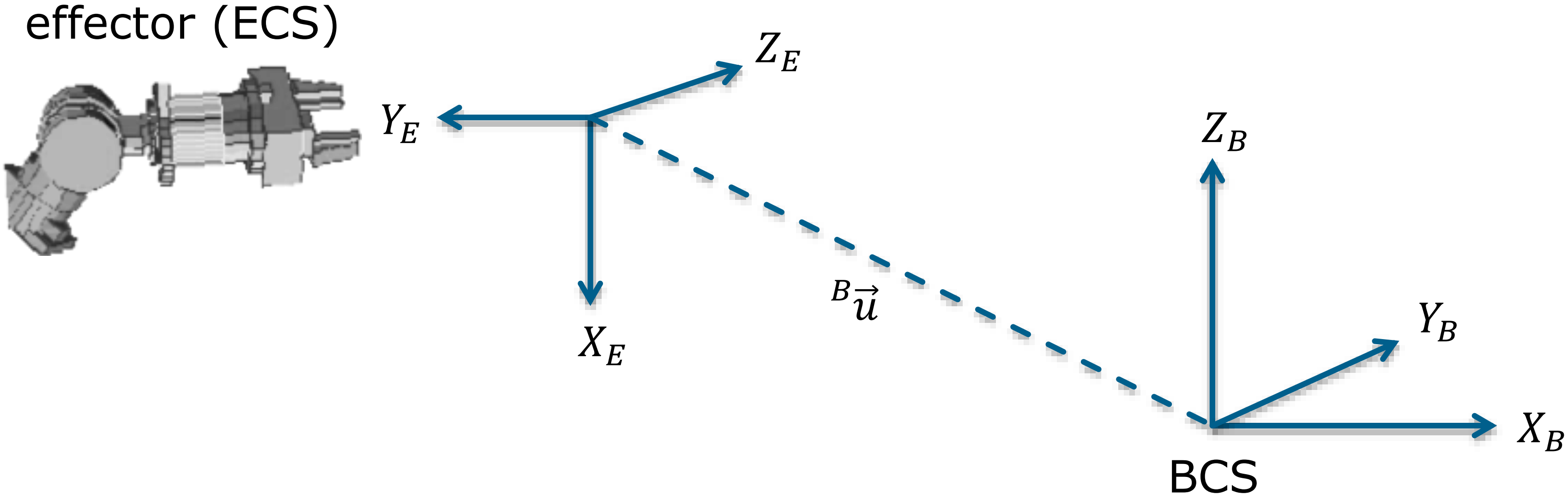
Orientation | Position

$$\left({}^{E}x,\, {}^{E}y,\, {}^{E}z, R\right) = \left[\begin{array}{ccc|c} 0 & 0 & 1 & {}^{B}u_x \\ 1 & 0 & 0 & {}^{B}u_y \\ 0 & 1 & 0 & {}^{B}u_z \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Calculation of a Homog. Transformation Matrix

Given: Description vector: $(-7{,}0{,}8{,}0°, 90°, 90°)$, roll-pitch-yaw

Wanted: Homogeneous representation of the pose as a matrix A



Calculation of a Homog. Transformation Matrix

- Compute $R = R_z(90°)R_y(90°)R_x(0°)$
- $$R_z(90) = \begin{bmatrix} \cos 90 & -\sin 90 & 0 \\ \sin 90 & \cos 90 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
- $$R_y(90) = \begin{bmatrix} \cos 90 & 0 & \sin 90 \\ 0 & 1 & 0 \\ -\sin 90 & 0 & \cos 90 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$
- $$R_x(0) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 0 & -\sin 0 \\ 0 & \sin 0 & \cos 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Calculation of a Homog. Transformation Matrix

$$R = R_z(90°)R_y(90°)R_x(0°)$$

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

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

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

$$\Rightarrow A = \begin{bmatrix} 0 & -1 & 0 & -7 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Calculation of a the Description Vector

Given: Homogeneous matrix: A

Wanted: Description vector: $\vec{v} = (x, y, z, \alpha, \beta, \gamma)$

$$A = \begin{bmatrix} 0 & 0 & 1 & -7 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Position: $x = -7, y = 0, z = 8$
- Orientation: Calculated by matrix equation:
  - $R_s = R_1 \cdot R_2 \cdot R_3$
  - 3×3-orientation part of homogeneous matrix
  - Product of rotation matrices corresponding to axes of rotation (Euler, roll-pitch-yaw)

Reformulating Matrix Equations

- The matrix equations can be represented as 16 individual equations, where 12 are non trivial

- The matrix equation can be solved by multiplying with the inverse matrix R_1^{-1} or R_3^{-1}
  - $R_1^{-1} \cdot R_s = R_2 \cdot R_3$
  - $R_s \cdot R_3^{-1} = R_1 \cdot R_2$

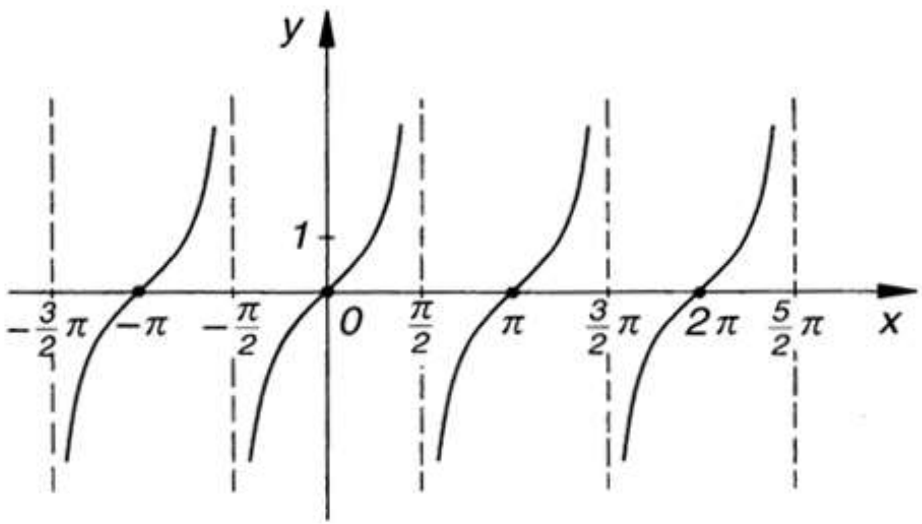
Form of the Equations

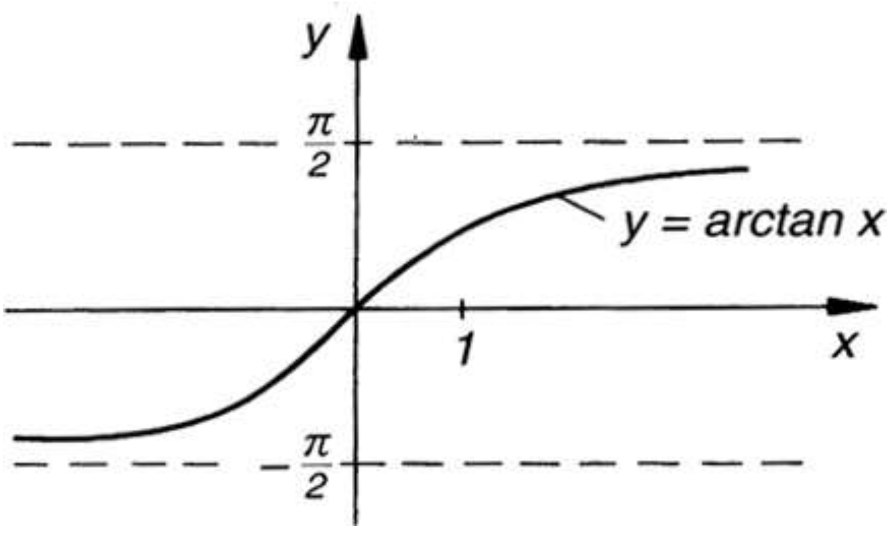
- After these reformulations there exists for each α, β, γ:
  (1) An equation of the form $b \cdot \sin \alpha - a \cdot \cos \alpha = 0$
  (2) Or a set of equations of the form $\sin \alpha = a, \cos \alpha = b$

- Approaches to determine the angle
  - An unknown angle should/must not be calculated with $\sin$ or $\cos$ because of its ambiguity
  - Approach: Use $\arctan$ (arcus tangent) and projection of the angle to the correct quadrant

Solving the Equations

- With (1) or (2) one can compute α, since from (1), and from (2) it follows that $\alpha = \arctan\frac{a}{b}$
- Problem: $\arctan x$ is ambiguous
  - Restriction to angles between $-\frac{\pi}{2}$ and $+\frac{\pi}{2}$





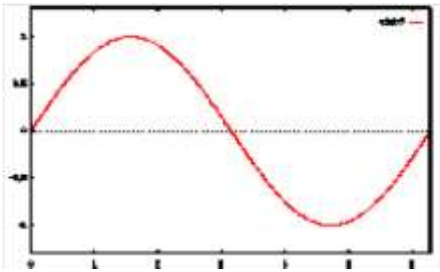
ATAN2

- To calculate α we do not use $\arctan$, but the 2-argument arctangent ATAN2
- Starting point: Equations of the form $\sin\alpha = a, \cos\alpha = b$
- Are only solvable for α,
  if $a^2 + b^2 = 1$, then $\sin^2\alpha + \cos^2\alpha = 1$
- Especially a and b can not be zero simultaneously

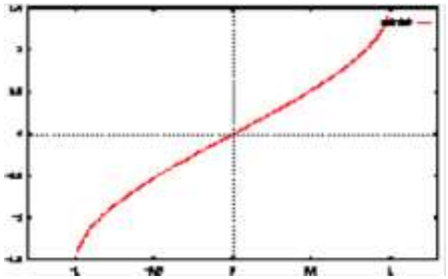
Arc-Functions of Sine and Cosine

(3) $\arcsin a = \alpha$ where $\alpha \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$

$\sin x$

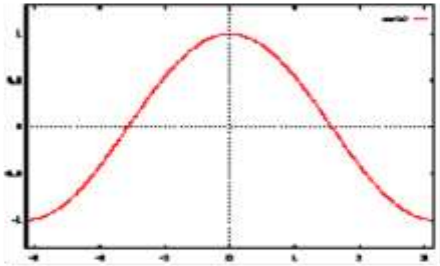


$\arcsin x$

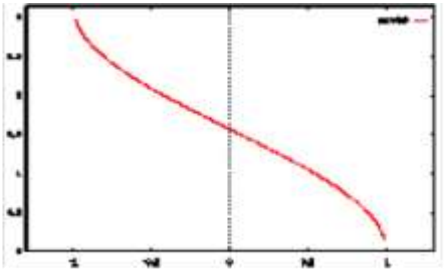


(4) $\arccos b = \alpha$ where $\alpha \in [0, \pi]$

$\cos x$



$\arccos x$



(5) Tangent $\tan \alpha = \dfrac{\sin \alpha}{\cos \alpha}$

Definition ATAN2

- $\mathrm{ATAN2}(a, b)$, where $a = \sin$ and $b = \cos$ due to (5)
- $\mathrm{ATAN2}(a, b) = \begin{cases} \arccos b \text{ , if } a \geq 0 \\ -\arccos b \text{ , if } a < 0 \end{cases}$

  Due to (3) and (4)

- Can also be computed for $\mathrm{ATAN2}(a, b) = \tan^{-1}\left(\frac{a}{b}\right)$
  - Choose quadrant with the signs of a and b
  - Example: $\mathrm{ATAN2}(-2, -2) = -135°$ (third quadrant)
  - Example: $\mathrm{ATAN2}(2{,}2) = -45°$ (first quadrant)

Computing Angles with ATAN2

- $\sin\theta = a \Rightarrow \theta = \pm\mathrm{ATAN2}\left(\sqrt{1-a^2}, a\right)$
- $\cos\theta = b \Rightarrow \theta = \pm\mathrm{ATAN2}\left(b, \pm\sqrt{1-b^2}\right)$
- $a \cdot \cos\theta + b \cdot \sin\theta = 0$ has two solutions
  - $\theta = \mathrm{ATAN2}(a, -b)$
  - $\theta = \mathrm{ATAN2}(-a, b)$
- $a \cdot \cos\theta + \mathrm{b} \cdot \sin\theta = c \Rightarrow \theta = \mathrm{ATAN2}(b, a) \pm \mathrm{ATAN2}\left(\sqrt{a^2 + b^2 + c^2}, c\right)$
- $a \cdot \cos\theta - b \cdot \sin\theta = c$ and $a \cdot \sin\theta + b \cdot \cos\theta = d$

  $$\Rightarrow \theta = \mathrm{ATAN2}(ad - bc, ac + bd)$$

Euler Angles

Computation of Euler angles for a general orientation matrix

- Matrix equation $R = R_z(\alpha) \cdot R_{y'}(\beta) \cdot R_{z''}(\gamma) = \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix}$
- Multiplying from the left with $R_z(\alpha)^{-1}$:
$$R_z(\alpha)^{-1} \cdot R_z(\alpha) \cdot R_{y'}(\beta) \cdot R_{z''}(\gamma) = R_z(\alpha)^{-1} \cdot R$$
- For orthogonal matrices it holds that $A^{-1} = A^T$ therefore
$$R_{y'}(\beta) \cdot R_{z''}(\gamma) = R_z(\alpha)^{-1} \cdot R = R_z(\alpha)^T \cdot R$$

$$\begin{bmatrix} C\beta \cdot C\gamma & -C\beta \cdot S\gamma & S\beta \\ S\gamma & C\gamma & 0 \\ -S\beta \cdot C\gamma & S\beta \cdot S\gamma & C\beta \end{bmatrix} = \begin{bmatrix} C\alpha n_x + S\alpha n_y & C\alpha o_x + S\alpha o_y & C\alpha a_x + S\alpha a_y \\ -S\alpha n_x + C\alpha n_y & -S\alpha o_x + C\alpha o_y & -S\alpha a_x + C\alpha a_y \\ n_z & o_z & a_z \end{bmatrix}$$

Computation of Euler Angles - an Example

Example gives the following equations:

(row.column)

(1.1) $C\beta \cdot C\gamma = C\alpha \cdot n_x + S\alpha \cdot n_y$

(1.2) $-C\beta \cdot S\gamma = C\alpha \cdot o_x + S\alpha \cdot o_y$

(1.3) $S\beta = C\alpha \cdot a_x + S\alpha \cdot a_y$

(2.1) $S\gamma = -S\alpha \cdot n_x + C\alpha \cdot n_y$

(2.2) $C\gamma = -S\alpha \cdot o_x + C\alpha \cdot o_y$

(2.3) $0 = -S\alpha \cdot a_x + C\alpha \cdot a_y$

(3.1) $-S\beta \cdot C\gamma = n_z$

(3.2) $S\beta \cdot S\gamma = o_z$

(3.3) $C\beta = a_z$

Computation of Euler Angles - an Example

- Angle α: From (2.3) it follows that:
  - $S\alpha \cdot a_x = C\alpha \cdot a_y \Leftrightarrow \frac{S\alpha}{C\alpha} = \tan\alpha = \frac{a_y}{a_x}$
  - Therefore $\alpha = \mathrm{ATAN2}(ay, ax)$
- Angle β: From $(1.3), (3.3)$ it follows that:
  - $\beta = \mathrm{ATAN2}\left(C\alpha a_x + S\alpha a_y, a_z\right)$
- Angle γ: From $(2.1), (2.2)$ it follows that
  - $\gamma = \mathrm{ATAN2}\left(C\alpha n_x + S\alpha n_y, -S\alpha o_x + C\alpha o_y\right)$
- Note: α is present in the solutions of β, γ

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:

$$(1.1)\quad C\beta = 0$$
$$(1.2)\quad 0 = 0$$
$$(1.3)\quad S\beta = 1$$
$$(2.1)\quad S\beta \cdot S\alpha = C\gamma$$
$$(2.2)\quad C\alpha = S\gamma$$
$$(2.3)\quad -S\alpha \cdot C\beta = 0$$
$$(3.1)\quad -C\alpha \cdot S\beta = -S\gamma$$
$$(3.2)\quad S\alpha = C\gamma$$
$$(3.3)\quad C\alpha \cdot C\beta = 0$$

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 $(0°, 90°, 90°)$

Concatenated Poses

- Poses are most of the time not given relative to the BCS, but relative to a more suitable CS (relative definition)
- Transformation between different coordinate system (e.g. BCS) necessary
- Pros of the relative pose definition
  - Less effort needed to track an objects motion
  - Individual coordinates only cover small distances

Rotation/Translation of Poses

- Let ${}^{BCS}_{A}H_{obj} = (4 \times 4)$ be the pose of an object in frame A relative to BCS
- Let ${}^{A}_{B}H_{obj} = (4 \times 4)$ be the pose of an object in frame B relative to the OCS of A
- Let ${}^{BCS}_{B}H_{obj} = (4 \times 4)$ be the pose of an object in frame B relative to BCS
  - It holds: ${}^{BCS}_{B}H_{obj} = {}^{BCS}_{A}H_{obj} \cdot {}^{A}_{B}H_{obj}$
- Notation is more compact, compared to the Cartesian notation:

$$\begin{aligned} {}^{BKS}R + {}^{BKS}\vec{v} &= {}^{BKS}_{A}R_1 \cdot \left({}^{A}_{B}R_2 + {}^{A}\vec{v}_{\mathrm{B}}\right) + {}^{BKS}\vec{v}_{\mathrm{A}} \\ &= {}^{BKS}_{A}R_1 \cdot {}^{A}_{B}R_2 + \left({}^{BKS}_{A}R_1 \cdot {}^{A}\vec{v}_{B}\right. \end{aligned}$$

Concatenated Poses - an Example

- Pose of object in frame 1 relative to BCS: ${}^{BCS}_{1}H_{obj}$
- Pose of object in frame 2 relative to frame 1: ${}^{1}_{2}H_{obj}$
- Pose of object in frame 3 relative to frame 2 : ${}^{2}_{3}H_{obj}$
- Pose of object in frame 3 relative to BCS : ${}^{BCS}_{3}H_{obj}$
  - ${}^{BCS}_{3}H_{obj} = {}^{BCS}_{1}H_{obj} \cdot {}^{1}_{2}H_{obj} \cdot {}^{2}_{3}H_{obj}$
- In order to use concatenated poses each matrix must be defined relative to the frame defined by its left matrix
  - $\prod_{i=1}^{n} {}^{i-1}_{i}H$ with $1 \leq i \leq n$ and ${}^{0}H = \mathrm{BCS}$

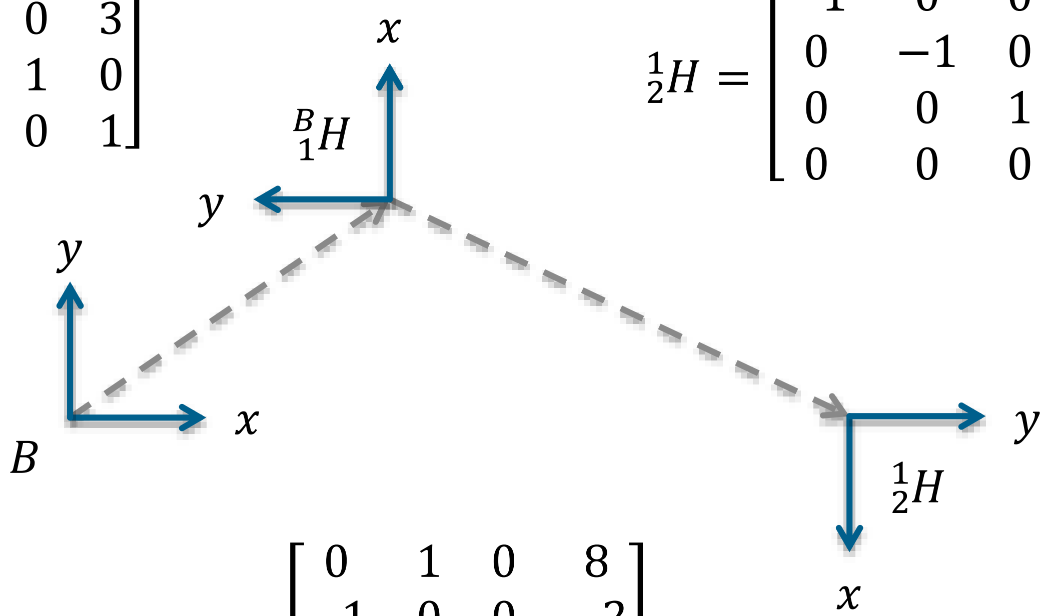
Concatenated Poses - an Example

- System of object H_1, is generated by a transformation $\left((3,3,0)^T, R_z(90°)\right)$ of an arbitrary coordinate system B: ${}^{B}_{1}H$
- System of object H_2, is generated by a transformation $\left((-5,-5,0)^T, R_z(-180°)\right)$ of the system of object H_1: ${}^{1}_{2}H$

Concatenated Poses - an Example

$$ {}^{BCS}_{1}H = \begin{bmatrix} 0 & -1 & 0 & 3 \\ 1 & 0 & 0 & 3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{1}_{2}H = \begin{bmatrix} -1 & 0 & 0 & -5 \\ 0 & -1 & 0 & -5 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$



$$ {}^{BCS}_{2}H = {}^{BCS}_{1}H \cdot {}^{1}_{2}H = \begin{bmatrix} 0 & 1 & 0 & 8 \\ -1 & 0 & 0 & -2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Chasles Theorem

- For all homogenous Matrices

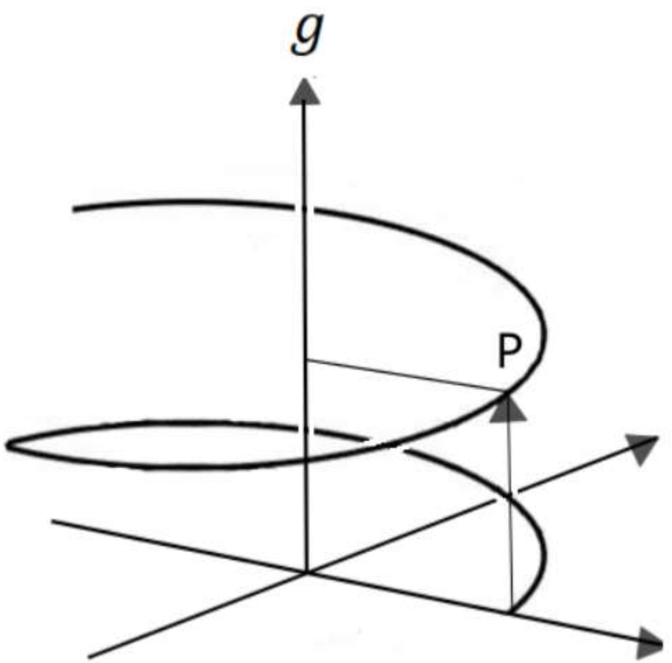
$$\begin{bmatrix} R & \vec{u} \\ 0 & 1 \end{bmatrix}$$

- There exists $\vec{g}$ and θ such that:

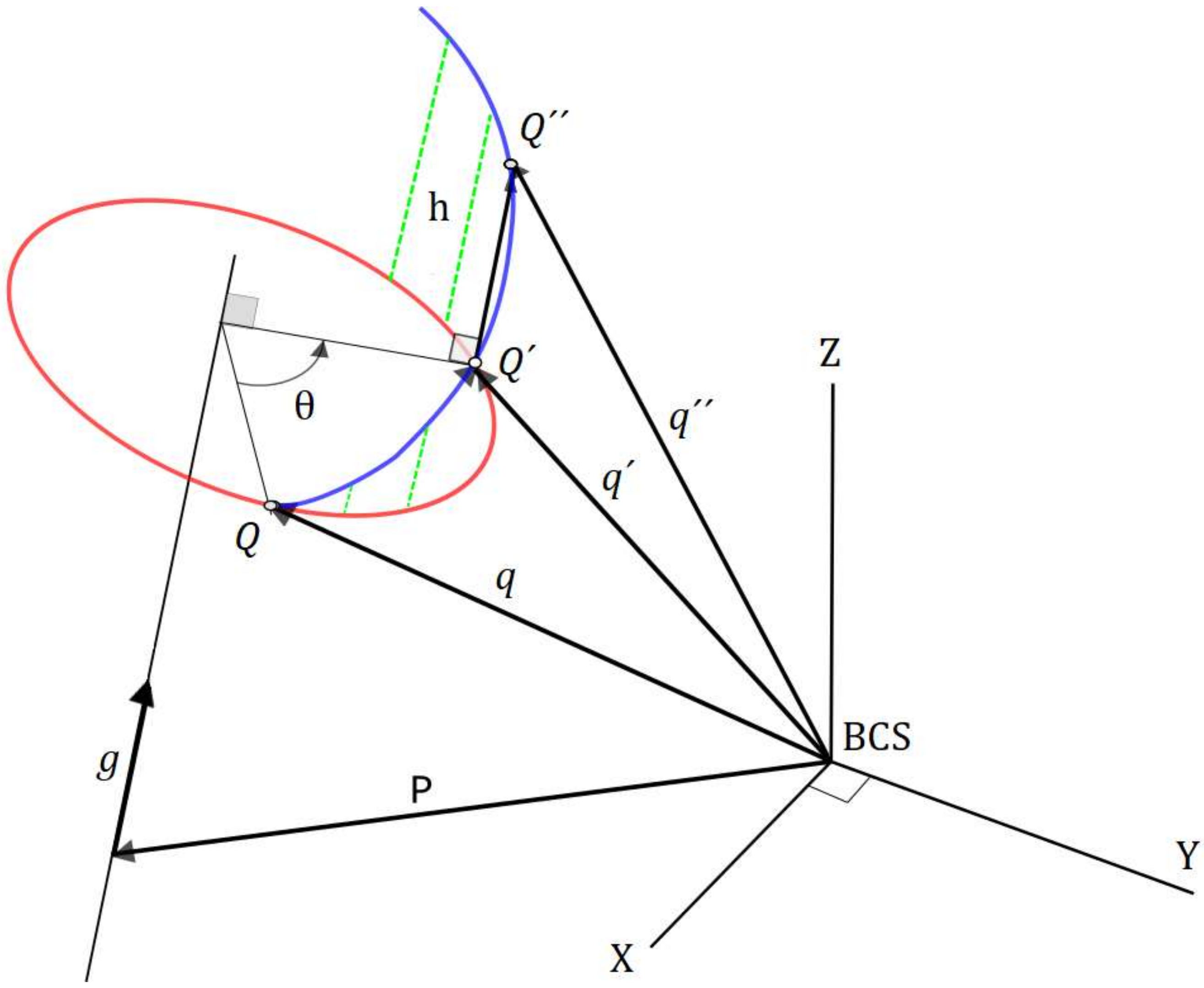
$\begin{bmatrix} R & \vec{u} \\ 0 & 1 \end{bmatrix}$ can be describe as

$$\begin{bmatrix} R_{\vec{g}}(\theta) & \vec{g} \\ 0 & 1 \end{bmatrix}$$

- $\vec{g}$ is called the screw axis
- θ is called the twist angle
  - Direction of $\vec{g}$ is given by Rodrigues formula, but needs to be scaled.



Screw Motion



Screws

- A screw $S = S(h, \theta, \vec{g}, \vec{P})$ is defined by:
  - a normalized screw axis $\vec{g}$
  - a twist angle θ
  - a translation h
  - a location $\vec{P}$

Screw types

- $p = \frac{h}{\theta}$ is called the pitch of screw $S = S(h, \theta, \vec{g}\,, P)$

- If $p > 0$, S is called right handed
- If $p < 0$, S is called left handed

- If $\vec{P} = 0$, then we write $S(h, \theta, \vec{g}) := S(h, \theta, \vec{g}, 0)$, and S is called a central screw.

From central Screws to Homogenous Matrices

- If we have $S(h, \theta, \vec{g})$ then,

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

and hence,

$$A_S(h, \theta, \vec{g}) = \begin{bmatrix} R_{\vec{g}}(\theta) & h\vec{g} \\ 0 & 1 \end{bmatrix}$$

From Screws to Homogenous Matrices

- $S\left(h, \theta, \vec{g}, \vec{P}\right) =$

$$\begin{bmatrix} C\theta I_3 + \vec{g}\vec{g}^T\eta\theta + \hat{g}S\theta & \left((I - \vec{g}\vec{g}^T)\eta\theta - \hat{g}S\theta\right)\vec{\hat{P}} + h\vec{g} \\ 0 & 1 \end{bmatrix}$$

From Homogenous Matrices to Screws

- By Rodrigues formula we get
  - $\theta = \cos^{-1}\left(\frac{tr(R)-1}{2}\right) \in [0, \pi]$, and
  - $\vec{g} = \frac{1}{2S_\theta}\begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$
- $\vec{P}$ is any point on the screw.
- h and $\vec{P}$ are still missing.

From Homogenous Matrices to Screws

- Goal:

Find $\vec{P} = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}$ and h, such that:

$$\begin{bmatrix} R & \vec{u} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} = \begin{bmatrix} R_{\vec{g}}(\theta) & h\vec{g} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} R - R_{\vec{g}}(\theta) & \vec{u} - h\vec{g} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\Leftrightarrow \left(R - R_{\vec{g}}(\theta)\right) \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} + \vec{u} - h\vec{g} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

From Homogenous Matrices to Screws

$$\Leftrightarrow \quad \vec{u} = h\vec{g} - \left(R - R_{\vec{g}}(\theta)\right)\begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

- W.l.o.g., $P_1 = 0$, since any point on the screw is fine

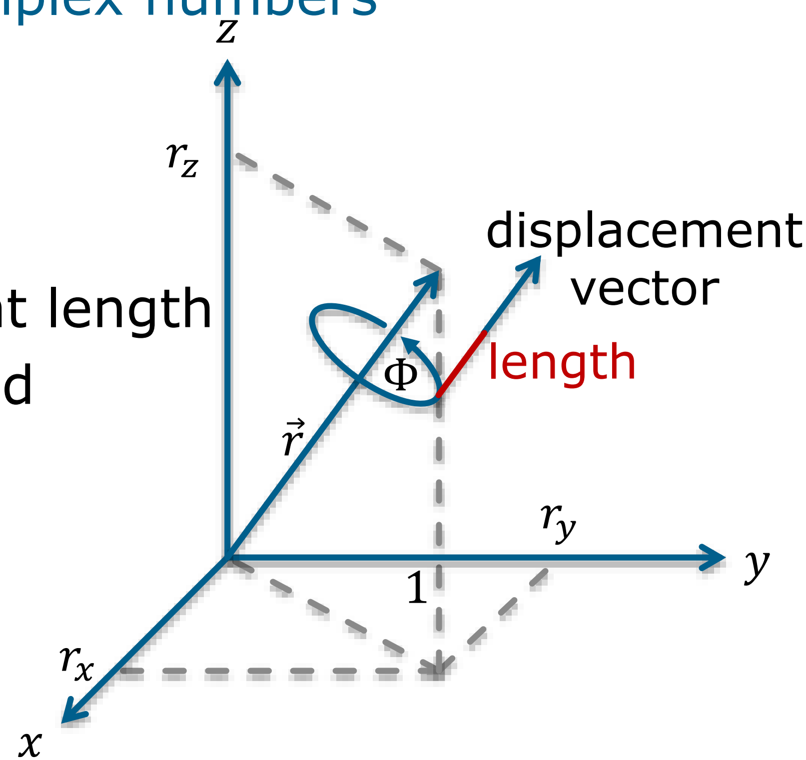
$$\Longrightarrow \vec{u} = \left[\vec{g},\ \left(R - R_{\vec{g}}(\theta)\right)_2,\ \ \left(R - R_{\vec{g}}(\theta)\right)_3\right]\begin{bmatrix} h \\ P_2 \\ P_3 \end{bmatrix}$$

$$\Longrightarrow \begin{bmatrix} h \\ P_2 \\ P_3 \end{bmatrix} = \left[\vec{g},\ \left(R - R_{\vec{g}}(\theta)\right)_2,\ \ \left(R - R_{\vec{g}}(\theta)\right)_3\right]^{-1} \vec{u}$$

- Here $(\cdot)_i$ denotes the i–th column

Dual Quaternions

- Quaternions are suitable for the description of the orientation, but not the position of an object
- Position and orientation can be expressed by quaternions
- Real numbers are replaced by complex numbers
  - $D_q = (d_1, d_2, d_3, d_4)$
  - $d_i = dp_i + \varepsilon \cdot ds_i$
  - $\varepsilon^2 = 0$
  - d_1: Angle value and displacement length
  - d_2, d_3, d_4: Description of a directed straight line in space in which the rotation and translation take place



Properties of Dual Quaternions

- Dual quaternions suitable for location description
- Operations on dual quaternions allow all needed transformations
- Low redundancy, as only 8 characteristics
- Gimbal lock does not exist
- Weaknesses
  - Difficulty for the user to describe a location by specifying a dual quaternion
  - Complex processing rules (e.g. multiplication)

Next Lecture

Modelling in robotics

- Degree of freedom
- Geometric model
- Direct kinematic model