# Introduction to Robotics

## Lecture 1

**Jianwei Zhang, Lasse Einig**
[zhang, einig]@informatik.uni-hamburg.de

University of Hamburg
Faculty of Mathematics, Informatics and Natural Sciences
Department of Informatics
**Technical Aspects of Multimodal Systems**

April 07, 2017

# General Information

| | |
|---|---|
| **Lecture:** | Friday 10:00 c. t. - 12:00 c. t. |
| **Room:** | F-334 |
| **Web:** | http://tams.inf...burg.de/lectures/ |
| **Name:** | Prof. Dr. Jianwei Zhang |
| **Office:** | F-330a |
| **E-mail:** | zhang@informatik.uni-hamburg.de |
| **Consultation:** | by arrangement |
| **Secretary:** | Tatjana Tetsis |
| **Office:** | F-311 |
| **Tel.:** | +49 40 42883-2430 |
| **E-mail:** | tetsis@informatik.uni-hamburg.de |

# General Information (cont.)



| | |
|---|---|
| **Exercises /RPC:** | Friday 8:00 c. t. - 10:00 c. t. (alternating)<br>see website for dates |
| **Room:** | F-334/F-304 |
| **Web:** | `http://tams.inf...burg.de/lectures/` |



| | |
|---|---|
| **Name:** | Lasse Einig |
| **Office:** | F-324 |
| **Tel.:** | +49 40 42883-2504 |
| **E-mail:** | `einig@informatik.uni-hamburg.de` |
| **Consultation:** | by arrangement (E-mail) |

# Exercises/RPC



Criteria for Course Certificate:

- min. 50 % of points in the exercises
  - min. 33 % in each exercise
- regular presence in exercises and RPC
- presentation of three tasks
- solutions in groups of 2–3
  - no solo submission
  - each member of a group must be able to present the tasks
  - failure to present results in 0 points

You may use your own laptop for the RPC.

# Previous Knowledge



- Basics in physics
  - basics of electrical engineering
- Linear algebra
- Elementary algebra of matrices
- Programming knowledge
  - Python or Matlab (recommended)
  - JavaScript (RPC)

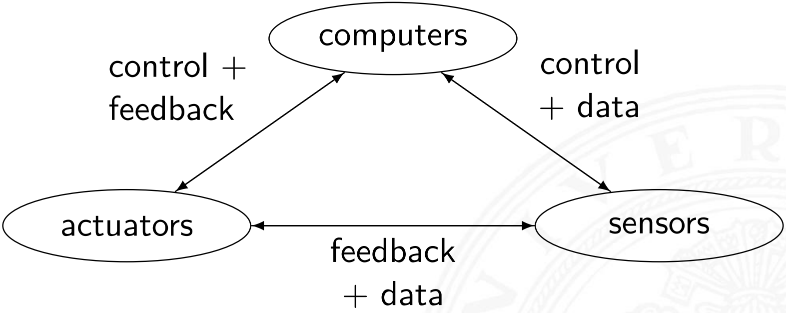
# Content

- Mathematic concepts
  - description of space and coordinate transformations
  - kinematics
  - dynamics
- Control concepts
  - movement execution
- Programming aspects
  - ROS, URDF, Kinematics Simulator
- Task-oriented movement and planning

# Outline

# Components of a robot

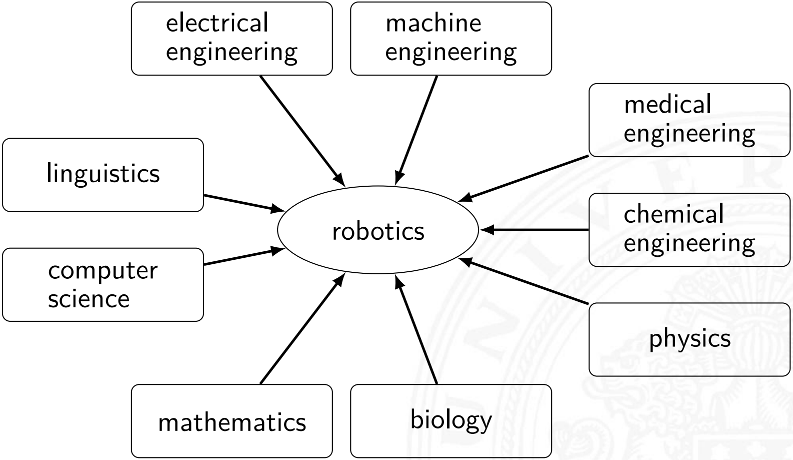





**Robotics**

Intelligent combination of computers, sensors and actuators.

# An Interdisciplinary Field



# Definition of Industry Robots



According to RIA[1], a robot is:

> *...a reprogrammable and multifunctional manipulator, devised for the transport of materials, parts, tools or specialized systems, with varied and programmed movements, with the aim of carrying out varied tasks.*

## Intelligent System

Is such a robot also an intelligent system?

[1] Robot Institute of America

# Background of Some Terms

**Robot** became popular through a stage play by Karel Čapek in 1923, being a capable servant.

**Robotics** was invented by Isaac Asimov in 1942.

**Autonomous** (literally) (gr.) "living by one's own laws" (*Auto:* Self; *nomos*: Law)

**Personal Robot** a small, mobile robot system with simple skills regarding vision system, speech, movement, etc. (from 1980).

**Service Robot** a mobile handling system featuring sensors for sophisticated operations in service areas (from 1989).

**Intelligent Robot, Cognitive Robot, Intelligent System ...**

# Degree of Freedom (DOF)



The number of independent coordinate planes or orientations on which a joint or end-point of a robot can move.
The DOF are determined by the number of independent variables of the control system.

- Point on a line
- Point on a plane
- Point in space
- Rigid body
  - one a surface
  - on a plane
  - in space
- Non-rigid body
- Manipulator
  - number of independently controllable joints
  - a robot should have at least two

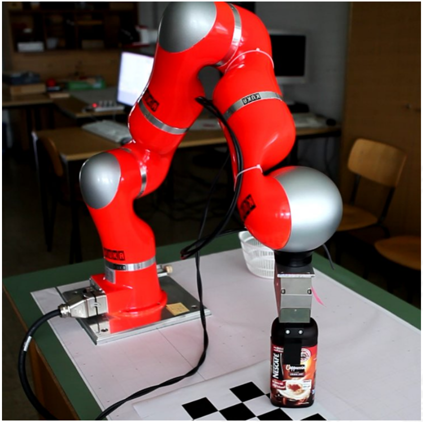
# DOF Examples




80's toy robot (Quickshot)

4-DOF + 1-DOF gripper

# DOF Examples (cont.)

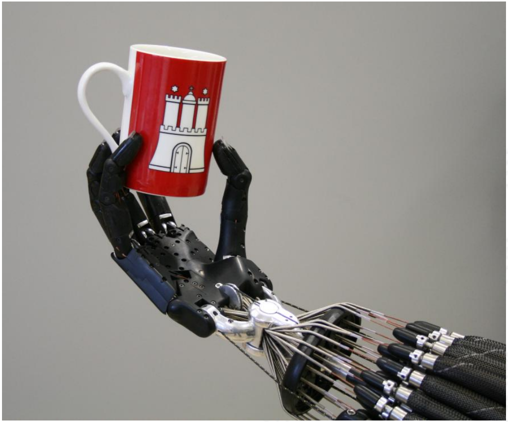



KUKA LWR 4+ arm with Schunk gripper

7-DOF + 1-DOF gripper

# DOF Examples (cont.)





Shadow C5 Air Muscle hand

20-DOF + 4 unactuated joints

# DOF Examples (cont.)



Boston Dynamics Atlas (2013)

28-DOF

# DOF Examples (cont.)




PR2 service robot with Shadow C6 electrical hand

19-DOF + 20-DOF gripper

# Robot classification

by input power source

- electrical
- hydraulic
- pneumatic

# Robot Classification (cont.)



by field of work

- stationary
  - arms with 2 DOF
  - arms with 3 DOF
  - ...
  - arms with 6 DOF
  - redundant arms ($>$ 6 DOF)
  - multi-finger hand
- mobile
  - automated guided vehicles
  - portal robot
  - mobile platform
  - running machines and flying robots
  - anthropomorphic robots (humanoids)

# Robot Classification (cont.)



by type of joint

- translatory
  - linear joint
  - translational
  - cartesian
  - prismatic
- rotatory
- combinations

by robot coordinate system

- cartesian
- cylindrical
- spherical

# Robot Classification (cont.)



by usage

- object manipulation
- object modification
- object processing
- transport
- assembly
- quality testing
- deployment in non-accessible areas
- agriculture and forestry
- underwater
- building industry
- service robot in medicine, housework, ...

# Robot Classification (cont.)



by intelligence

- manual control
- programmable for repeated movements
- featuring cognitive ability and responsiveness
- adaptive on task level

# Robotics is Fun!



- robots move — computers don't
- interdisciplinarity
  - soft- and hardware technology
  - sensor technology
  - mechatronics
  - control engineering
  - multimedia, . . .
- A dream of mankind:

  *Computers are the most ingenious product of human laziness to date.*

  computers ⇒ robots

# Literature



Slides and literature references @
`http://tams.informatik.uni-hamburg.de/lectures/`

K. Fu, R. González, and C. Lee, *Robotics: Control, Sensing, Vision, and Intelligence*. McGraw-Hill series in CAD/CAM robotics and computer vision, McGraw-Hill, 1987

R. Paul, *Robot Manipulators: Mathematics, Programming, and Control : the Computer Control of Robot Manipulators*. Artificial Intelligence Series, MIT Press, 1981

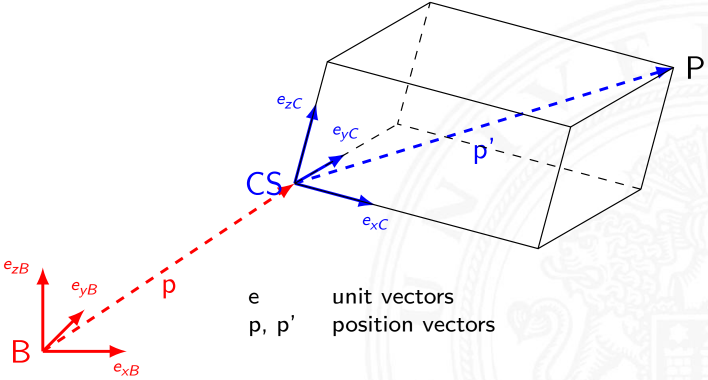
J. Craig, *Introduction to Robotics: Pearson New International Edition: Mechanics and Control*. Always learning, Pearson Education, Limited, 2013

# Outline

# Outline (cont.)

# Coordinate systems

The **pose** of objects, in other words their **location** and **orientation** in Euclidian space can be described through specification of a cartesian coordinate system (CS) in relation to a base coordinate system (B).

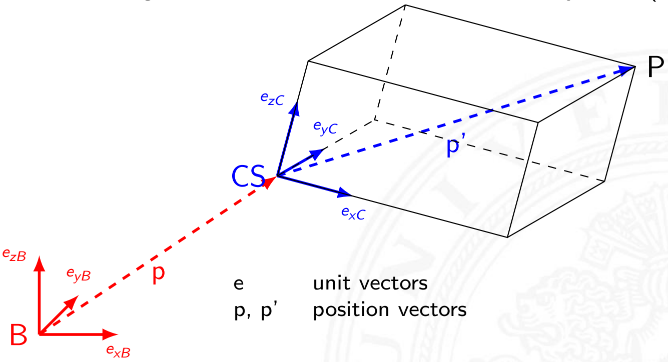



| | |
|---|---|
| e | unit vectors |
| p, p' | position vectors |

# Specification of location and orientation

Position (object coordinates):

- translation along the axes of the base coordinate system (B)

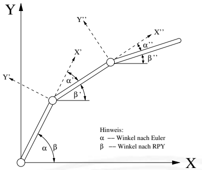



| | |
|---|---|
| e | unit vectors |
| p, p' | position vectors |

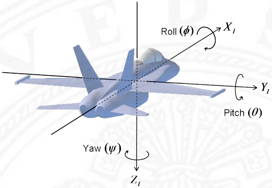
- given by $\mathbf{p} = [p_x, p_y, p_z]^T \in \mathcal{R}^3$

`Orientation` (in space):

- Euler-angles $\phi, \theta, \psi$
  - rotations are performed successively around the axes of the new coordinate systems, e. g. $ZY'X''$ or $ZX'Z''$ (12 possibilities)




- Gimbal-angles (Roll-Pitch-Yaw)
  - relative to object coordinates (used in aviation and maritime)
  - rotation with respect to **fixed** axes (X – Roll, Y – Pitch, Z – Yaw)




- given by Rotation-matrix $R \in \mathcal{R}^{3\times 3}$
  - redundant; 9 parameters for 3 DOF

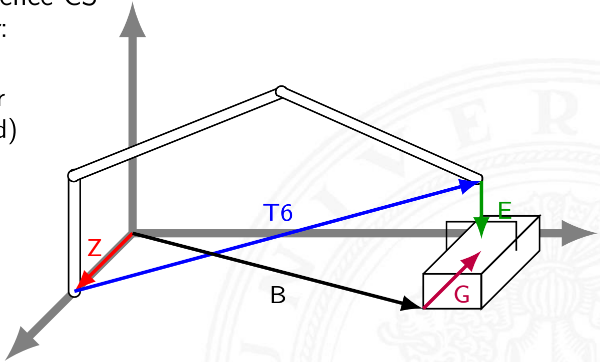
$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

# Specification of location and orientation (cont.)

- **Position:**
  - given through $\vec{p} \in \mathcal{R}^3$
- **Orientation:**
  - given through projection $\vec{n}, \vec{o}, \vec{a} \in \mathcal{R}^3$ of the axes of the CS to the origin system
  - summarized to rotation matrix $R = \begin{bmatrix} \vec{n} & \vec{o} & \vec{a} \end{bmatrix} \in \mathcal{R}^{3\times 3}$
  - redundant, since there are 9 parameters for 3 degrees of freedom
  - other kinds of representation possible, e.g. *roll*, *pitch*, *yaw* angle, qaternions etc.

# Coordinate transformations

- `Transform of Coordinate systems`:
  frame: a reference CS
  *typical frames*:
  - robot base
  - end-effector
  - table (world)
  - object
  - camera
  - screen
  - ...




Frame-transformations transform one frame into another.
${}^{A}T_{B}$ transforms frame $A$ to frame $B$ (Latex: `$^{A}T_{B}$`)

# Homogenous transformation

- Combination of $\vec{p}$ and $R$ to $T = \begin{bmatrix} & R & & \vec{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} \in \mathcal{R}^{4\times 4}$
- Concatenation of several $T$ through matrix multiplication
  - ${}^{A}T_{B}\,{}^{B}T_{C} = {}^{A}T_{C}$
- not commutative, in other words ${}^{B}T_{C}\,{}^{A}T_{B} \neq {}^{A}T_{B}\,{}^{B}T_{C}$

# Homogenous transformation (cont.)

- `Homogeneous transformation matrices:`

$$H = \begin{bmatrix} R & T \\ P & S \end{bmatrix}$$

where $P$ depicts the perspective transformation and $S$ the scaling.

- In robotics, $P = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$ and $S = 1$. Other values are used for computer graphics.

# Translatory transformation



A translation with a vector $[p_x, p_y, p_z]^T$ is expressed through a transformation $H$:

$$H = T_{(p_x, p_y, p_z)} = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Rotatory transformation

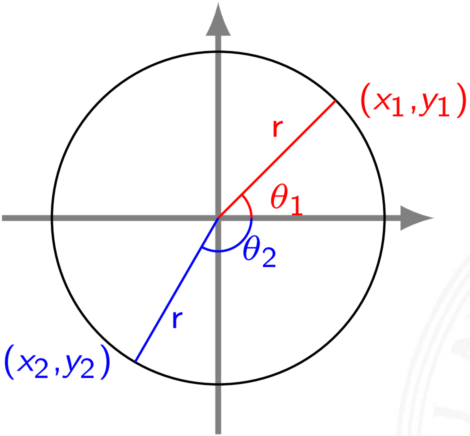
(shortened representation: $S$ : sin, $C$ : cos)

The transformation corresponding to a rotation around the $x$-axis with angle $\psi$ ($Psi$):

$$R_{x,\psi} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\psi & -S\psi & 0 \\ 0 & S\psi & C\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Rotatory transformation (cont.)






**Coordinates of a circle**

$x = r \sin\theta, \; y = r \cos\theta$

# Rotatory transformation (cont.)

The transformation corresponding to a rotation around the $y$-axis with angle $\theta$ (*Theta*):

$$R_{y,\theta} = \begin{bmatrix} C\theta & 0 & S\theta & 0 \\ 0 & 1 & 0 & 0 \\ -S\theta & 0 & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Rotatory transformation (cont.)



The transformation corresponding to a rotation around the $z$-axis with angle $\phi$ (*Phi*):

$$R_{z,\phi} = \begin{bmatrix} C\phi & -S\phi & 0 & 0 \\ S\phi & C\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Rotatory transformation (cont.)

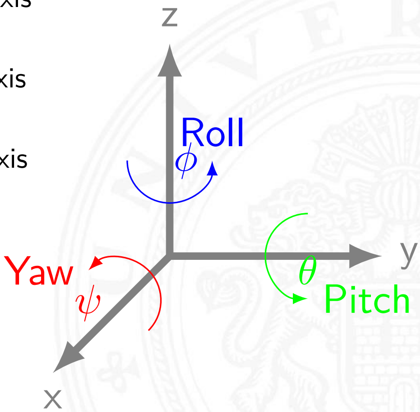
Signs of transformations:

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

# Multiple rotations

Sequential left-multiplication of the transformation matrices by order of rotation.

1. rotation $\psi$ around the $x$-axis
   $R_{x,\psi}$ – Yaw
2. rotation $\theta$ around the $y$-axis
   $R_{y,\theta}$ – Pitch
3. rotation $\phi$ around the $z$-axis
   $R_{z,\phi}$ – Roll



# Concatenation of rotation matrices

$$R_{\phi,\theta,\psi} = R_{z,\phi} R_{y,\theta} R_{x,\psi}$$

$$= \begin{bmatrix} C\phi & -S\phi & 0 & 0 \\ S\phi & C\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta & 0 \\ 0 & 1 & 0 & 0 \\ -S\theta & 0 & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\psi & -S\psi & 0 \\ 0 & S\psi & C\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} C\phi C\theta & C\phi S\theta S\psi - S\phi C\psi & C\phi S\theta C\psi + S\phi S\psi & 0 \\ S\phi C\theta & S\phi S\theta S\psi + C\phi C\psi & S\phi S\theta C\psi - C\phi S\psi & 0 \\ -S\theta & C\theta S\psi & C\theta C\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

*Remark:* Matrix multiplication is not commutative:

$$AB \neq BA$$

# Coordinate frames

They are represented as four vectors using the elements of homogeneous transformation.

$$H = \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

# Inverse transformation

The inverse of a rotation matrix is simply its transpose:

$$R^{-1} = R^T \text{ and } RR^T = I$$

whereas $I$ is the identity matrix.

The inverse of (1) is:

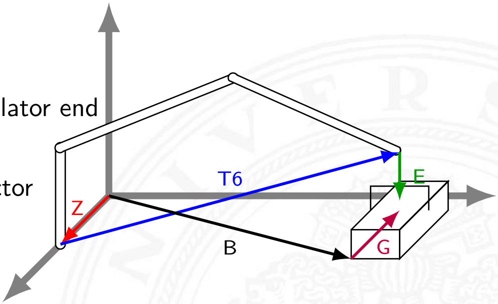
$$H^{-1} = \begin{bmatrix} r_{11} & r_{21} & r_{31} & -\mathbf{p}^\mathbf{T} \cdot \mathbf{r}_1 \\ r_{12} & r_{22} & r_{32} & -\mathbf{p}^\mathbf{T} \cdot \mathbf{r}_2 \\ r_{13} & r_{23} & r_{33} & -\mathbf{p}^\mathbf{T} \cdot \mathbf{r}_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

whereas $\mathbf{r}_1$, $\mathbf{r}_2$, $\mathbf{r}_3$ and $\mathbf{p}$ are the four column vectors of (1) and $\cdot$ represents the scalar product of vectors.

# Relative transformations

One has the following transformations:

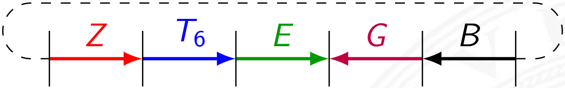
- $Z$:
  World $\rightarrow$ Manipulator base
- $T_6$:
  Manipulator base $\rightarrow$ Manipulator end
- $E$:
  Manipulator end $\rightarrow$ Endeffector
- $B$:
  World $\rightarrow$ Object
- $G$:
  Object $\rightarrow$ Endeffector



# Transformation equation

There are two descriptions for the desired endeffector position, one in relation to the object and the other in relation to the manipulator. Both descriptions should equal to each other for grasping:

$$Z T_6 E = B G$$




In order to find the manipulator transformation:

$$T_6 = Z^{-1} B G E^{-1}$$

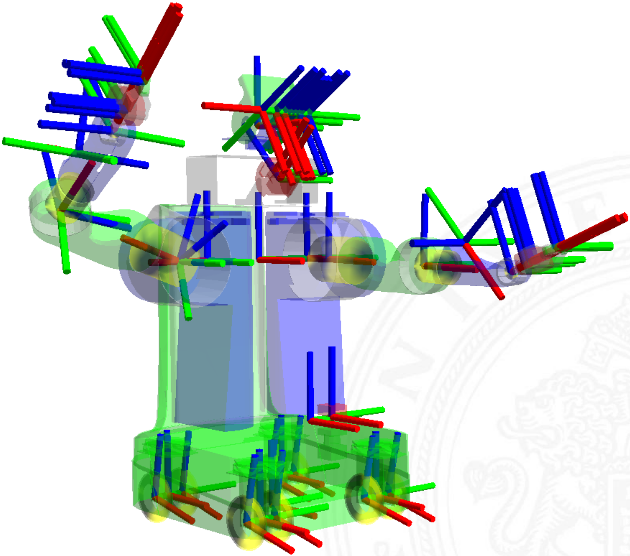
In order to determine the position of the object:

$$B = Z T_6 E G^{-1}$$

This is also called kinematic chain.

# Example: coordinate transformation





# Summary of homogeneous transformations



- `A homogeneous transformation` depicts the position and orientation of a coordinate frame in space.
- `If the coordinate frame` is defined in relation to a solid object, the position and orientation of the solid object is unambiguously specified.
- `The depiction` of an object $A$ can be derived from a homogeneous transformation relating to object $A'$. This is also possible the other way around using inverse transformation.

# Summary of homogeneous transformations (cont.)



- Several translations and rotations can be multiplied. The following applies:
  - If the rotations / translations are performed in relation to the current, newly defined (or changed) coordinate system, the newly added transformation matrices need to be multiplicatively appended on the right-hand side.
  - If all of them are performed in relation to the fixed reference coordinate system, the transformation matrices need to be multiplicatively appended on the left-hand side.
- A homogeneous transformation can be segemented into a rotational and a translational part.

## Reduction to area of interest

For grasping, position and orientation of the robot **gripper** are of interest.

The robot itself is reduced to a single transformation and treated as a solid object.

# Coordinates of a manipulator

- `Joint coordinates:`
  A vector $\mathbf{q}(t) = (q_1(t), q_2(t), ..., q_n(t))^T$
  (a robot configuration)
- `Endeffector coordinates`
  (Object coordinates):
  A Vector $\mathbf{p} = [p_x, p_y, p_z]^T$
- `Description of orientations:`
  - Euler angle $\phi, \theta, \psi$
  - Rotation matrix:

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

# Outlook: Denavit-Hartenberg Convention

- `Definition` of one coordinate system per segment $i = 1..n$
- `Definition` of 4 parameters per segment $i = 1..n$
- `Definition` of one transformation $A_i$ per segment $i = 1..n$
- $T_6 = \prod_{i=1}^{n} A_i$

**Outlook**

Later Denavit Hartenberg Convention will be presented in more detail!

# Outlook: Kinematics



- `The direct kinematic problem`:
  Given the joint values and geometrical parameters of all joints of a manipulator, how is it possible to determine the position and orientation of the manipulator's endeffector?
- `The inverse kinematic problem`:
  Given a desired position and orientation of the manipulator's endeffector and the geometrical parameters of all joints, is it possible for the manipulator to reach this position / orientation? If it is, how many manipulator configurations are capable of matching these conditions?

**Example**

A two-joint-manipulator moving on a plane

# Outlook: Position

$T_6$ defines, how the $n$ joint angles are supposed to be consolidated to 12 non-linear formulas in order to describe 6 cartesian degrees of freedom.

- `Forward kinematics` $K$ defined as:
  - $K : \vec{\theta} \in \mathcal{R}^n \rightarrow \vec{x} \in \mathcal{R}^6$
  - Joint angle $\rightarrow$ Position $+$ Orientation
- `Inverse kinematics` $K^{-1}$ defined as:
  - $K^{-1} : \vec{x} \in \mathcal{R}^6 \rightarrow \vec{\theta} \in \mathcal{R}^n$
  - Position $+$ Orientation $\rightarrow$ Joint angle
  - non-trivial, since $K$ is usually not unambiguously invertible

# Outlook: Differential movement

Non-linear kinematics $K$ can be linearized through the *Taylor series* $f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n$.

- `The Jacobian matrix` $J$ as factor for $n = 1$ of the multi-dimensional Taylor series is defined as:
  - $J(\vec{\theta}) : \dot{\vec{\theta}} \in \mathcal{R}^n \rightarrow \dot{\vec{x}} \in \mathcal{R}^6$
  - Joint speed $\rightarrow$ cartesian speed
- `Inverse Jacobian matrix` $J^{-1}$ defined as:
  - $J^{-1}(\vec{\theta}) : \dot{\vec{x}} \in \mathcal{R}^6 \rightarrow \dot{\vec{\theta}} \in \mathcal{R}^n$
  - cartesian speed $\rightarrow$ Joint speed
  - non-trivial, since $J$ not necessarily invertible (e.g. not quadratic)

# Outlook: Motion planning

Since $T_6$ describes only the target **position**, explicit generation of a trajectory is necessary.
Depending on *constraints* different for:

- `joint angle space`
- `cartesian space`

Interpolation through:

- `piecewise straight lines`
- `piecewise polynoms`
- `B-Splines`
- `...`

# Suggestions

- Read (available on google & library):
  - J. F. Engelberger, *Robotics in service*. MIT Press, 1989
  - K. Fu, R. González, and C. Lee, *Robotics: Control, Sensing, Vision, and Intelligence*. McGraw-Hill series in CAD/CAM robotics and computer vision, McGraw-Hill, 1987
  - R. Paul, *Robot Manipulators: Mathematics, Programming, and Control : the Computer Control of Robot Manipulators*. Artificial Intelligence Series, MIT Press, 1981
  - J. Craig, *Introduction to Robotics: Pearson New International Edition: Mechanics and Control*. Always learning, Pearson Education, Limited, 2013
- Repeat your linear algebra knowledge, especially regarding elementary algebra of matrices.

# Bibliography

[1] K. Fu, R. González, and C. Lee, *Robotics: Control, Sensing, Vision, and Intelligence*.
McGraw-Hill series in CAD/CAM robotics and computer vision, McGraw-Hill, 1987.

[2] R. Paul, *Robot Manipulators: Mathematics, Programming, and Control : the Computer Control of Robot Manipulators*.
Artificial Intelligence Series, MIT Press, 1981.

[3] J. Craig, *Introduction to Robotics: Pearson New International Edition: Mechanics and Control*.
Always learning, Pearson Education, Limited, 2013.

[4] J. F. Engelberger, *Robotics in service*.
MIT Press, 1989.

[5] W. Böhm, G. Farin, and J. Kahmann, "A Survey of Curve and Surface Methods in CAGD," *Comput. Aided Geom. Des.*, vol. 1, pp. 1–60, July 1984.

[6] J. Zhang and A. Knoll, "Constructing fuzzy controllers with B-spline