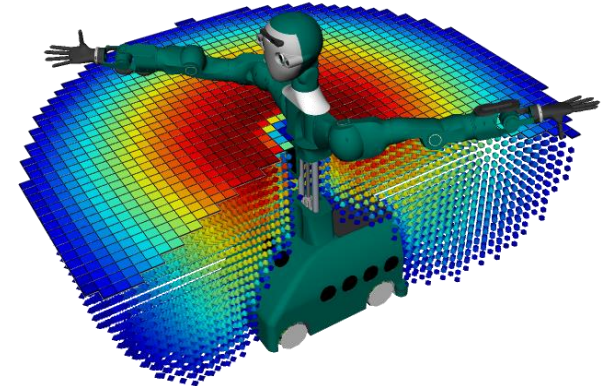
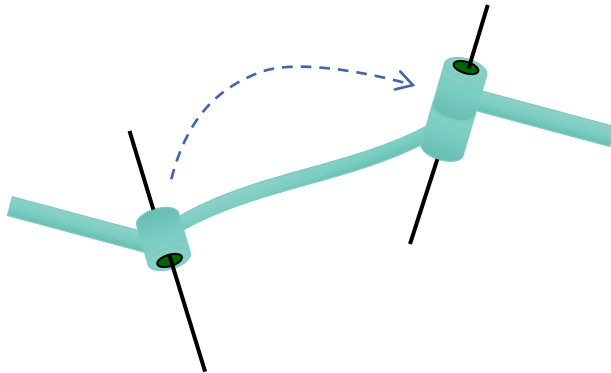


Robotik I: Introduction to Robotics

Chapter 2 – Kinematics

Tamim Asfour

<https://www.humanoids.kit.edu>



Models in Robotics – Outline

■ Kinematic Models

Kinematics studies of motion of bodies and systems based **only on geometry**, i.e. without considering the physical properties and the forces acting on them. The essential concept is a **pose** (position and orientation).

■ Dynamic Models


Dynamics studies the relationship between the **forces and moments** acting on a robot and accelerations they produce,

■ Geometric Models

Geometry: Mathematical description of the **shape of bodies**

Content

■ Kinematic Model

- Kinematic Chain
- Denavit-Hartenberg Convention
- Direct Kinematics Problem
- Examples
- Jacobian Matrix 
- Singularities and Manipulability
- Representation of Reachability

■ Geometric Model

- Areas of Application
- Classification
- Examples

Content

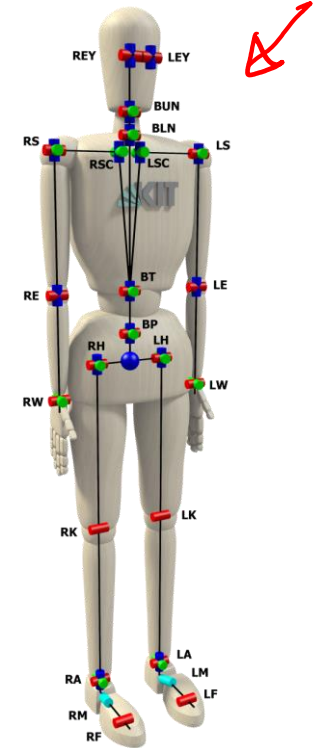
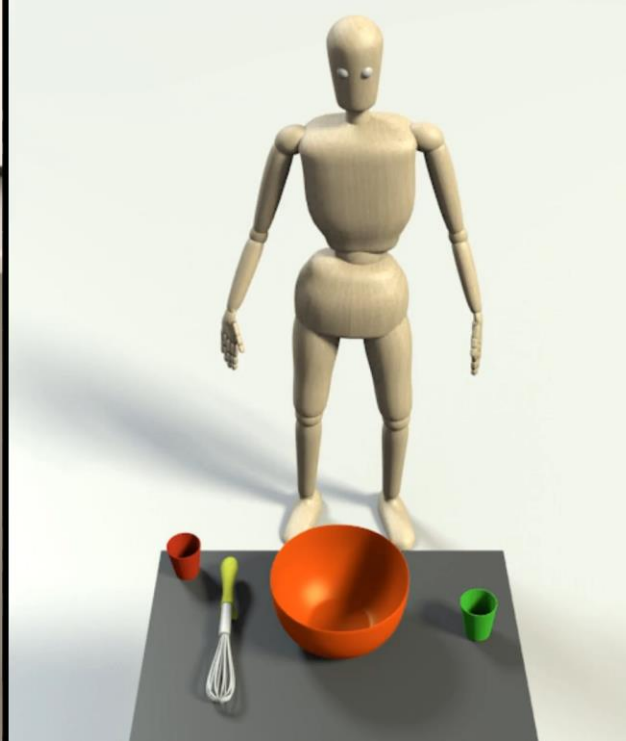
■ Kinematic Model

- Kinematic Chain
- Denavit-Hartenberg Convention
- Direct Kinematics Problem
- Examples
- Jacobian Matrix
- Singularities and Manipulability
- Representation of Reachability

■ Geometric Model

- Areas of Application
- Classification
- Examples

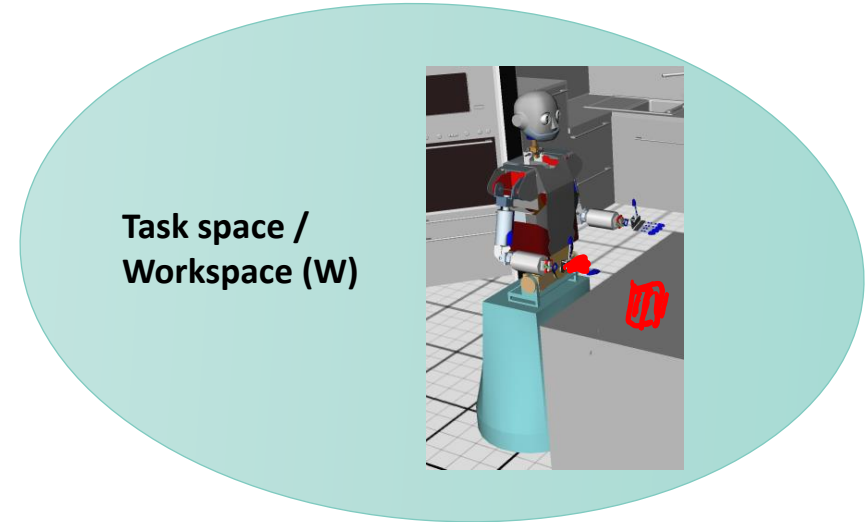
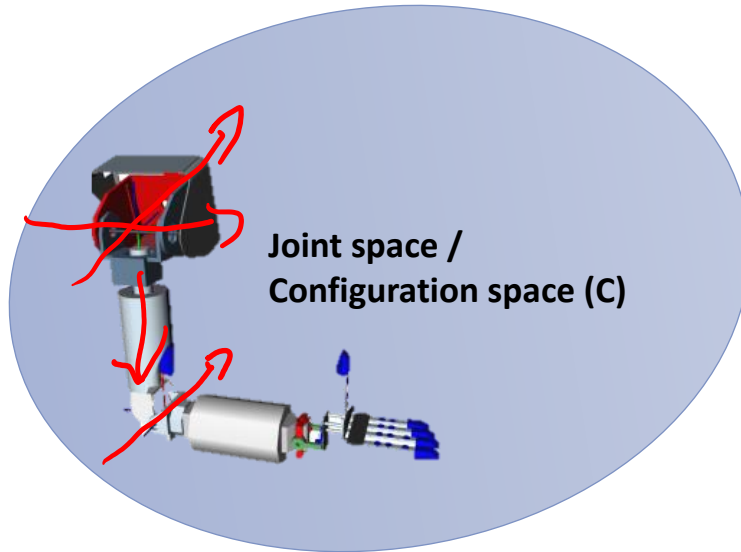
Kinematic Model (1)



Kinematic Model (2)

$$q = \begin{pmatrix} q_1 \\ \vdots \\ q_7 \end{pmatrix}$$

Two spaces



Kinematic Model (3)

Definition

The **kinematic model** of a robot describes the relationships between the **joint space** (robot coordinates, configuration space) and the **space of end effector poses** in world coordinates (task space, Cartesian space).

Areas of application

Relationship between **joint angles** and **poses** of the end effector

Reachability analysis

Geometric relation between the **body parts of the robot** (self-collision)

Geometric relation to the **environment** (collision detection)

Forward Kinematics

- **Direct** kinematics problem
 - Input: Joint angles of the robot
 - Output: **Pose of the end effector**

Forward Kinematics:
HERE!

End effector

Where is my
hand?

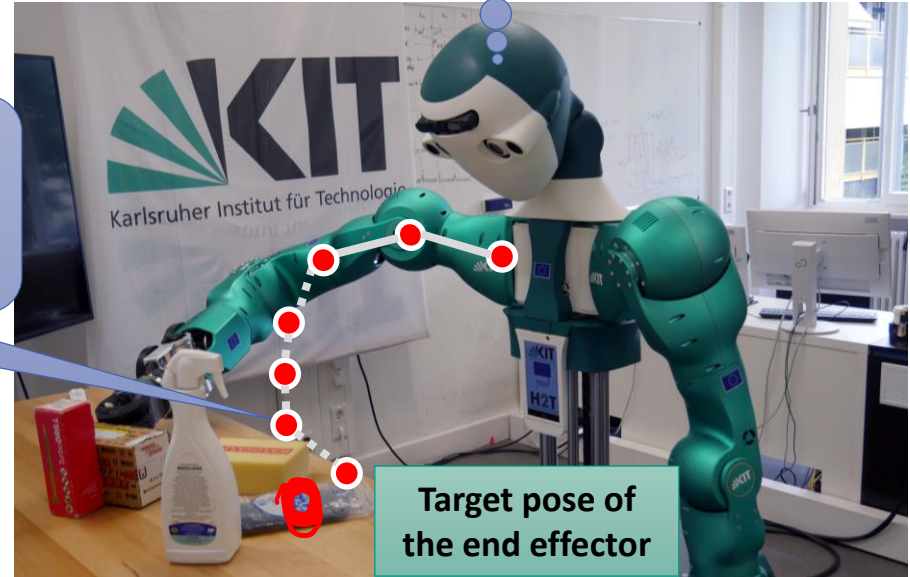


Inverse Kinematics

- **Inverse** kinematics problem
 - Input: Target pose of the end effector
 - Output: **Joint angles**

**Inverse Kinematics:
Determines the joint angles**

**How do I move my
hand to the target?**



Outline: Direct and Inverse Kinematics

Joint space
(configuration space)

Transformation

Cartesian coordinates
(task space)

$$(\theta_1, \dots, \theta_n) \in \mathcal{C} \subseteq \mathbb{R}^n$$

Direct Kinematics

$$\mathbf{x} = f(\boldsymbol{\theta})$$

Inverse Kinematics

$$\boldsymbol{\theta} = f^{-1}(\mathbf{x})$$

$$\mathbf{X} \subset \mathbb{R}^m$$

Position and orientation of
the end effector

$$\mathbf{x}_{EEF} = (x, y, z, \alpha, \beta, \gamma)$$

n : Joint degrees of freedom (DoF)

m : Cartesian degrees of freedom

Content

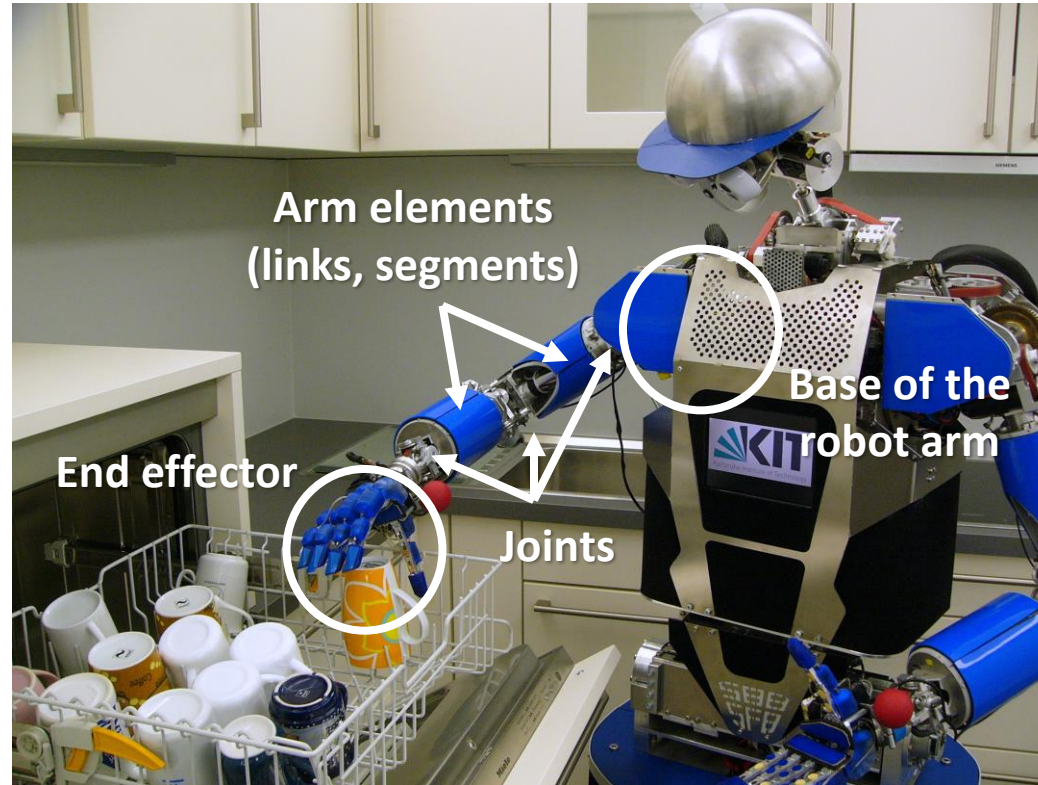
■ Kinematic Model

- **Kinematic Chain**
- Denavit-Hartenberg Convention
- Direct Kinematics Problem
- Examples
- Jacobian Matrix
- Singularities and Manipulability
- Representation of Reachability

■ Geometric Model

- Areas of Application
- Classification
- Examples

Elements of a Kinematic Chain

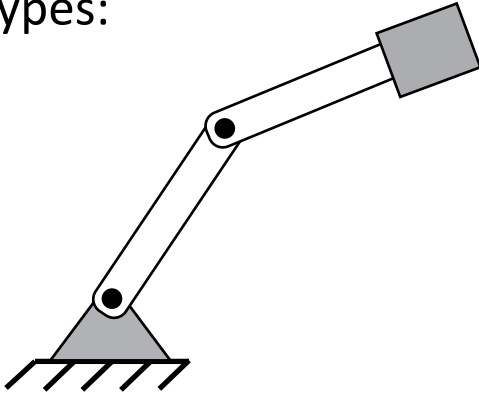


Kinematic Chain: Definition

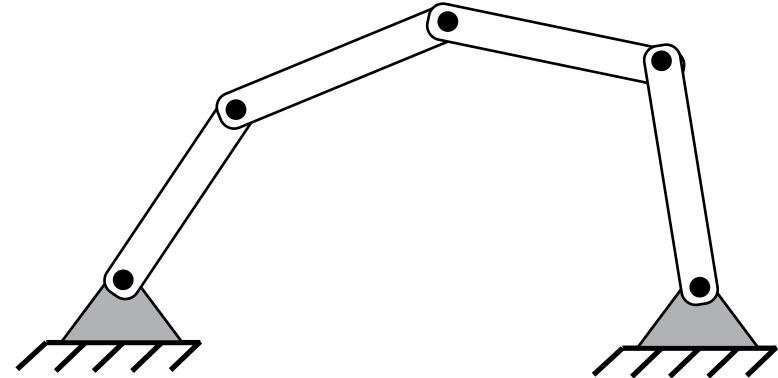
Definition:

A kinematic chain is formed by **several bodies** that are **kinematically connected by joints** (e.g. robot arm).

Types:



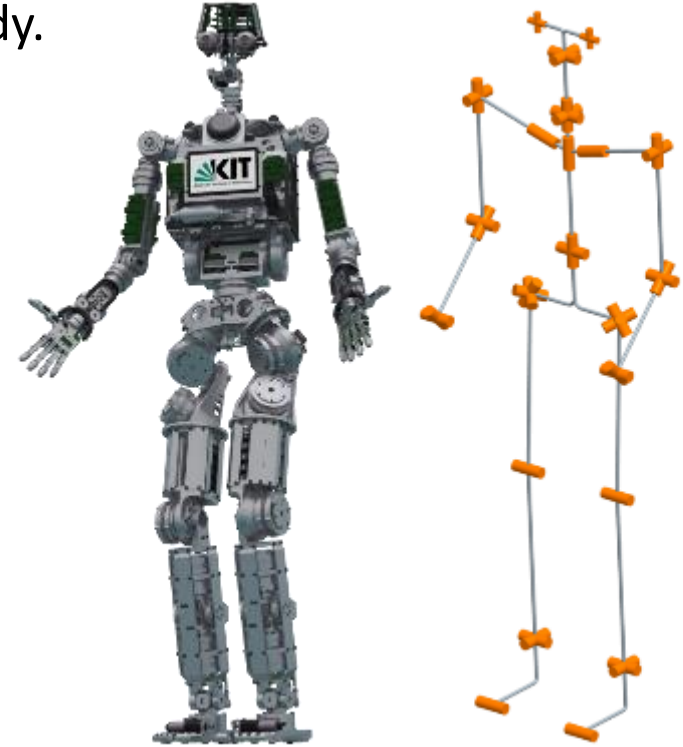
Open kinematic chain



Closed kinematic chain

Kinematic Chain: Conventions

- Each **arm element** corresponds to one **rigid** body.
- Each arm element is connected to the next one by a **joint**.
- For prismatic and rotational joints:
Each **joint** has only **one degree of freedom** (translation respectively rotation).
- **Kinematic parameters:**
 - Joint definition (e.g. rotation axis)
 - Transformation between joints



Kinematic Parameters

Joint parameters

Revolute joint: rotation axis

Prismatic joint: direction of translation

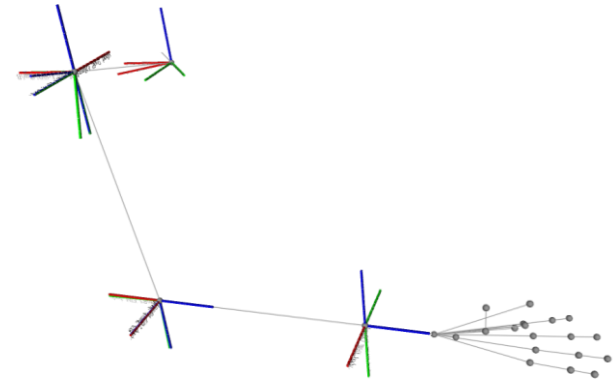
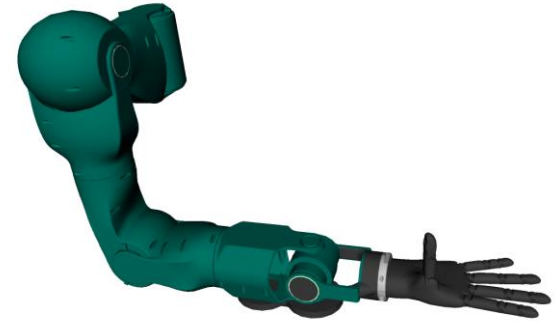
...

Specification of the **positions of joints relative to each other**

Fixed transformation between two joints

Defines the local coordinate systems of the joints

Transformation from joint $i - 1$ to joint i with transformation matrix ${}^{i-1}T_i$



Number of Parameters of the Kinematic Chain

A **transformation** must be determined for each link:

3 rotation parameters

3 translation parameters

→ **6 parameters** per link of the kinematic chain

Contents

■ Kinematic Model

- Kinematic Chain
- **Denavit-Hartenberg Convention**
- Direct Kinematics Problem
- Examples
- Jacobian Matrix
- Singularities and Manipulability
- Representation of Reachability

■ Geometric Model

- Areas of Application
- Classification
- Examples

Denavit-Hartenberg (DH) Convention

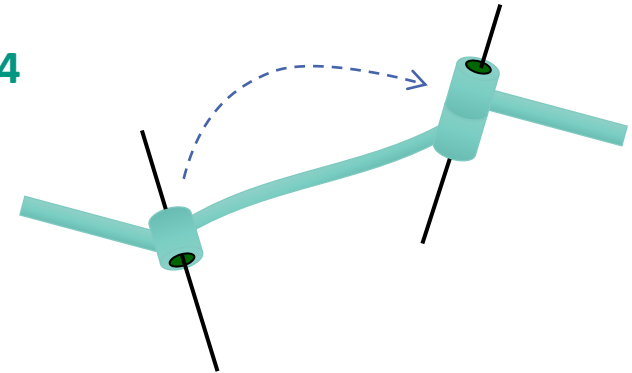
Goal: **Reduction of the parameters** for describing an arm element

Properties

Systematic description of relations (**translations** and **rotations**)
between adjacent joints

Reduction of the number of **parameters from 6 to 4**

Description with homogeneous matrices



Literature: Denavit, Hartenberg: „A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices“,
Journal of Applied Mechanics, 1955, vol 77, pp 215-221

DH Convention for the Choice of Coordinate Systems

■ Each coordinate system is determined on the basis of the following three rules:

1. The z_{i-1} -axis lies along the **axis of movement** of the i -th joint
2. The x_i -axis lies along the **common normal** of z_{i-1} and z_i
(direction via cross product: $x_i = z_{i-1} \times z_i$)
3. The y_i -axis completes the coordinate system according to the **right-hand rule**

$$i \in \{\text{base}, 1, \dots, n\}$$

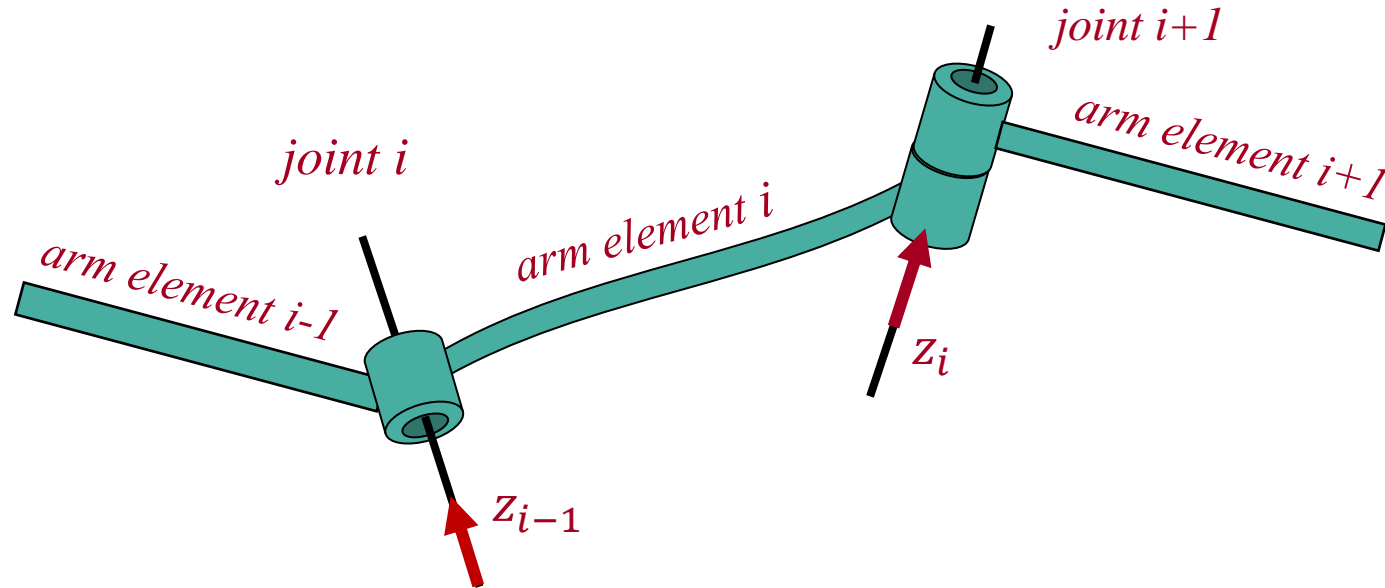
→ Derivation of parameters for arm element and joint

■ Remark

- Other variants of the DH convention can also be found in the literature
- In this lecture we consider the modified variant of Waldron and Paul

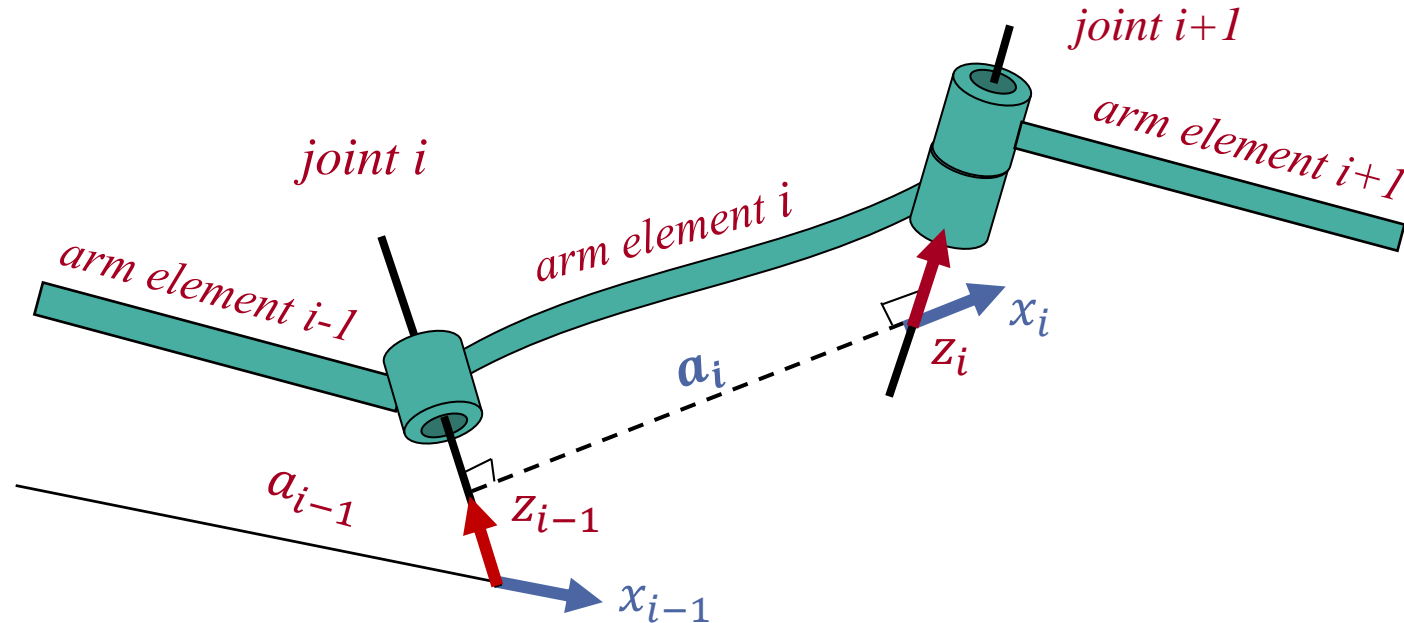
DH Convention: Parameters of the Arm Element (1)

- Each **arm element** i is embedded between two **joints** i and $i + 1$
- z_i runs along the **joint axis** $i + 1$



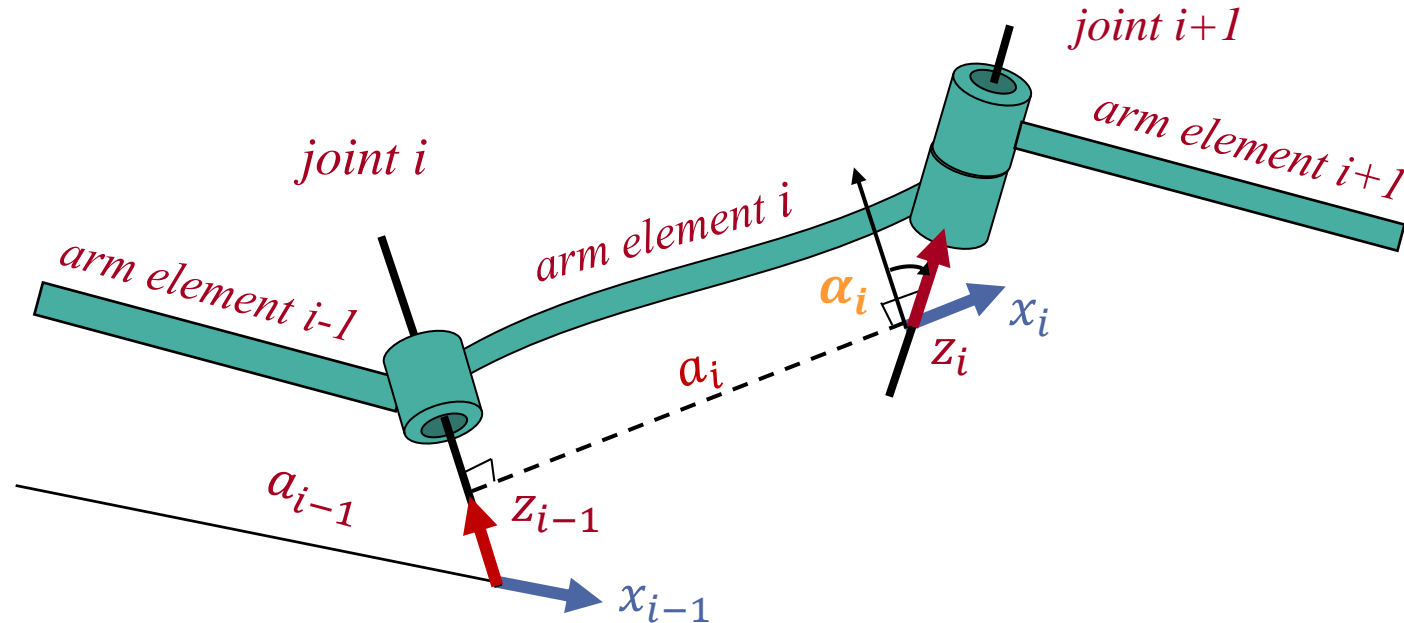
DH Convention: Parameters of the Arm Element (2)

- Link length a_i of an arm element i describes the **distance** from z_{i-1} to z_i
- x_i lies along the **normal of z_{i-1} and z_i** (cross product)



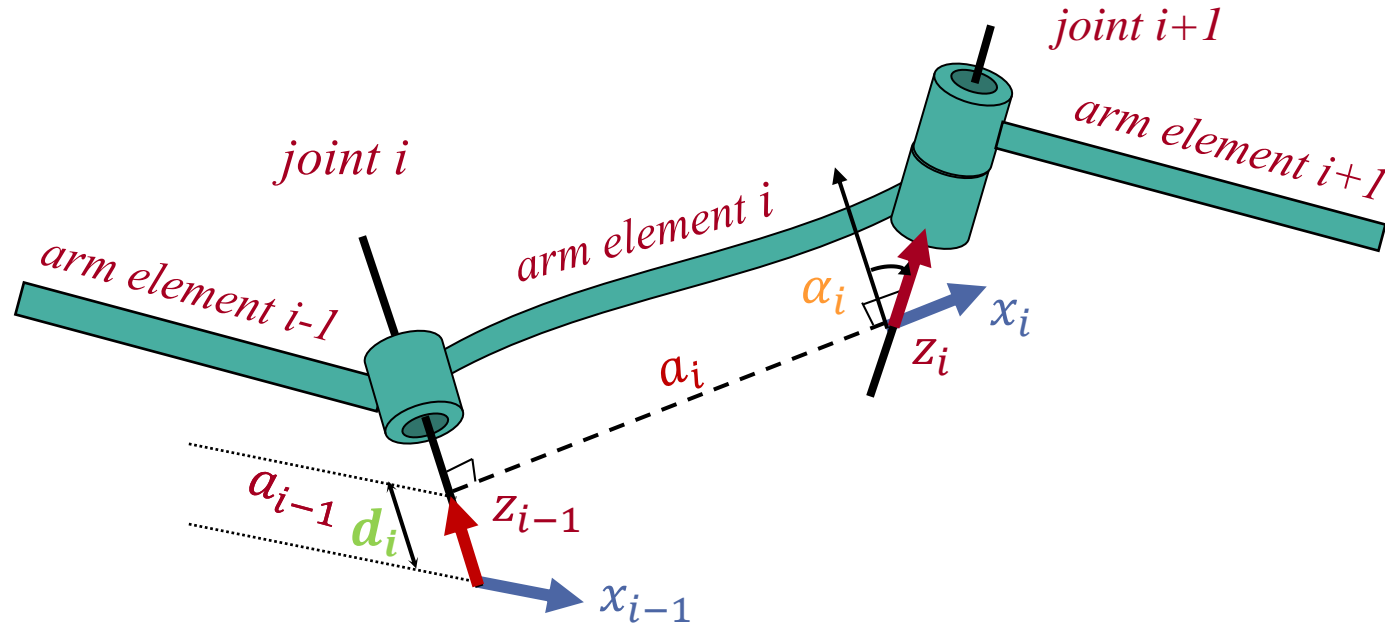
DH Convention: Parameters of the Arm Element (3)

- **Link twist** α_i describes the **angle** from z_{i-1} to z_i **around** x_i .



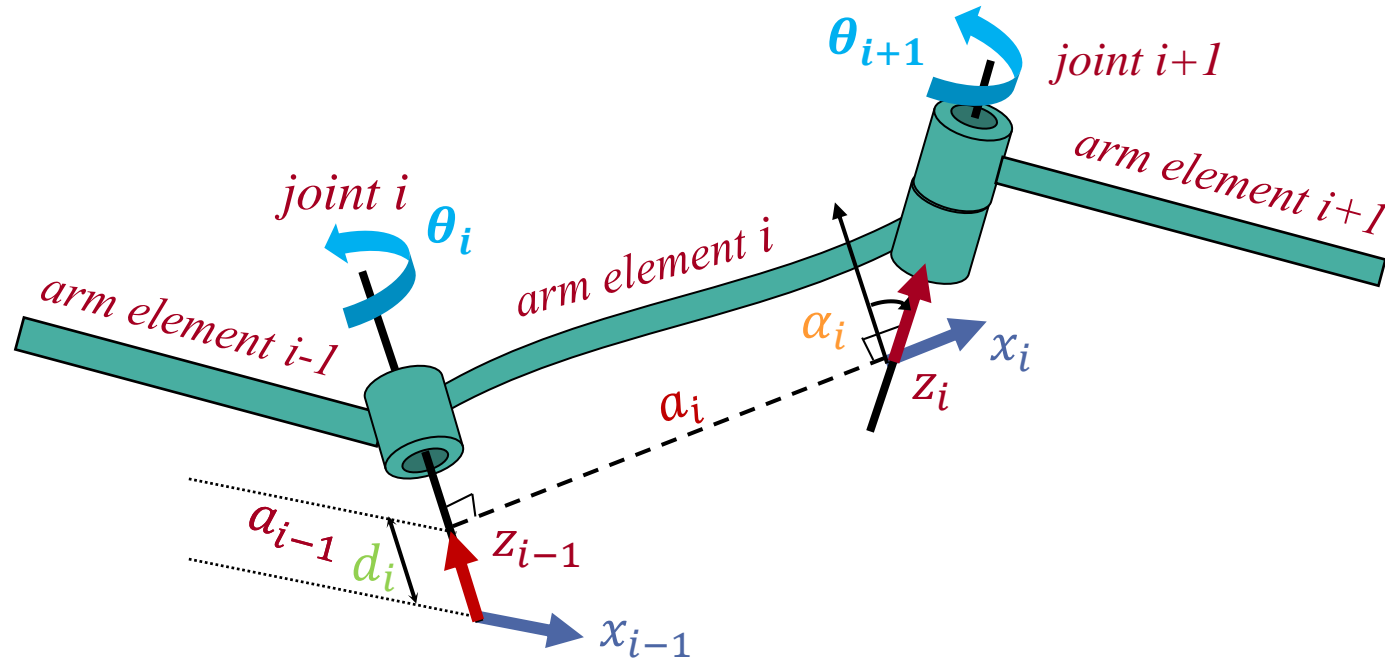
DH Convention: Parameters of the Arm Element (4)

- **Link offset d_i** is the distance between x_{i-1} -axis and x_i -axis along the z_{i-1} -axis



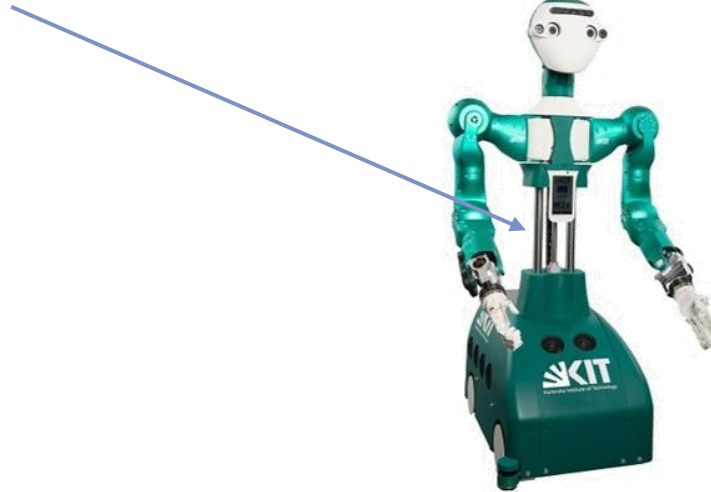
DH Convention: Parameters of the Arm Element (5)

- Joint angle θ_i is the angle from x_{i-1} to x_i around z_{i-1}



DH Parameters

- The **four parameters** a_i , α_i , d_i and θ_i are called **DH parameters**.
- They describe the transformations between two successive rotational or translational robot joints



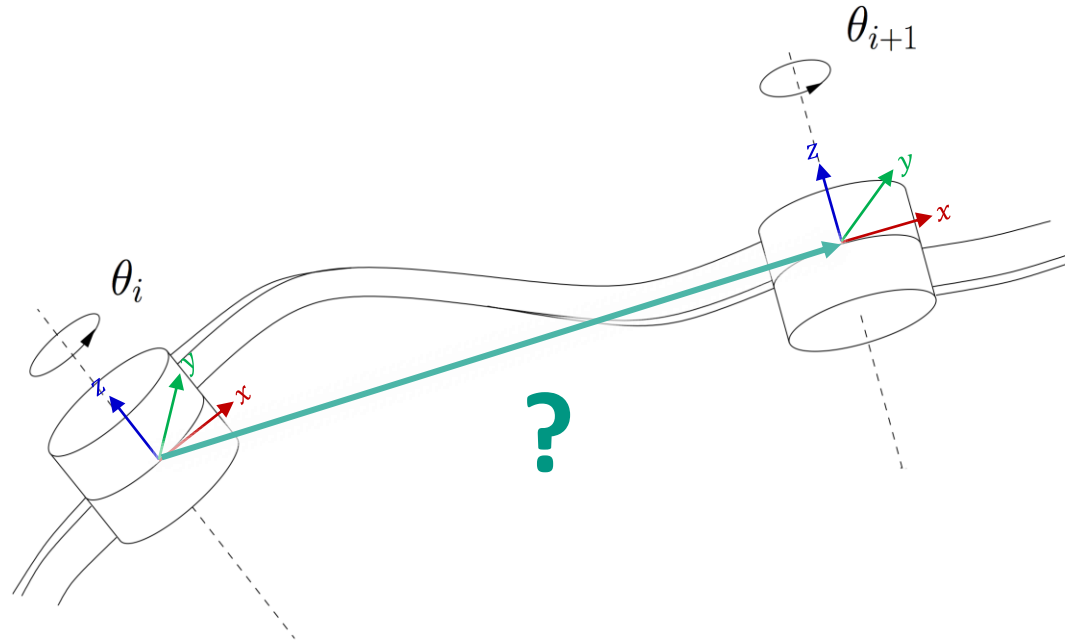
DH Parameters (Denavit-Hartenberg Parameters)

旋转关节

滑动关节

Parameter	Symbol	Revolute joint	Prismatic joint
Link length	a	constant	constant
Link twist	α	constant	constant
Link offset	d	constant	variable
Joint angle	θ	variable	constant

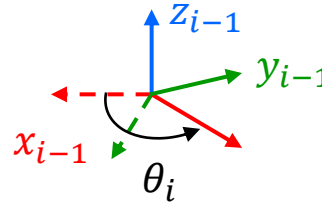
Transformation Between Two Robot Joints



DH Transformation Matrices (1)

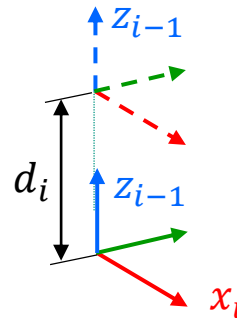
Transformation from LCS_{i-1} to LCS_i

1. A **rotation** θ_i around the z_{i-1} -axis so that the x_{i-1} -axis is parallel to the x_i -axis.



$$R_{z_{i-1}}(\theta_i) = \begin{pmatrix} \cos \theta_i & -\sin \theta_i & 0 & 0 \\ \sin \theta_i & \cos \theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

2. A **translation** d_i along the z_{i-1} -axis to the point where z_{i-1} and x_i intersect.



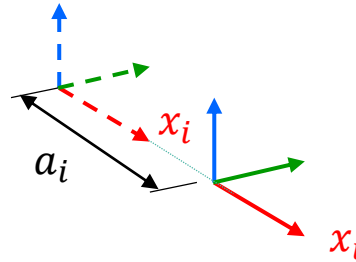
$$T_{z_{i-1}}(d_i) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

LCS_i : Local Coordinate System of joint i

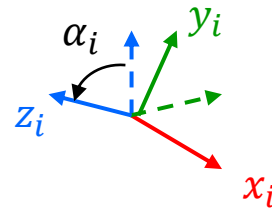
DH Transformation Matrices (2)

Transformation from LCS_{i-1} to LCS_i

3. A **translation** a_i along the x_{i-1} -axis to align the origins of the coordinate systems.
4. A **rotation** α_i around the x_i -axis to convert the z_{i-1} -axis into the z_i -axis.



$$T_{x_i}(a_i) = \begin{pmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



$$R_{x_i}(\alpha_i) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i & -\sin \alpha_i & 0 \\ 0 & \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

LCS_i : Local Coordinate System of joint i

DH Transformation Matrices (3)

Transformation LCS_{i-1} to LCS_i

$$\begin{aligned} A_{i-1,i} &= R_{z_{i-1}}(\theta_i) \cdot T_{z_{i-1}}(d_i) \cdot T_{x_i}(a_i) \cdot R_{x_i}(\alpha_i) \\ &= \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cdot \cos \alpha_i & \sin \theta_i \cdot \sin \alpha_i & a_i \cdot \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cdot \cos \alpha_i & -\cos \theta_i \cdot \sin \alpha_i & a_i \cdot \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

Inverse DH Transformation

Transformation from LCS_{i-1} to LCS_i

$$A_{i-1,i}^{-1} = A_{i,i-1}$$

$$= \begin{bmatrix} \cos \theta_i & \sin \theta_i & 0 & -a_i \\ -\cos \alpha_i \cdot \sin \theta_i & \cos \theta_i \cdot \cos \alpha_i & \sin \alpha_i & -d_i \cdot \sin \alpha_i \\ \sin \theta_i \cdot \sin \alpha_i & -\sin \alpha_i \cdot \cos \theta_i & \cos \alpha_i & -d_i \cdot \cos \alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T = \begin{pmatrix} n_x & o_x & a_x & u_x \\ n_y & o_y & a_y & u_y \\ n_z & o_z & a_z & u_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T^{-1} = \begin{pmatrix} n_x & n_y & n_z & -n^T \mathbf{u} \\ o_x & o_y & o_z & -o^T \mathbf{u} \\ a_x & a_y & a_z & -a^T \mathbf{u} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

See chapter 1

Concatenation of DH Transformations

- By concatenating the DH matrices, the pose of individual **coordinate systems relative** to the **reference coordinate system** can be determined.
- Position of the m -th coordinate system relative to the base:

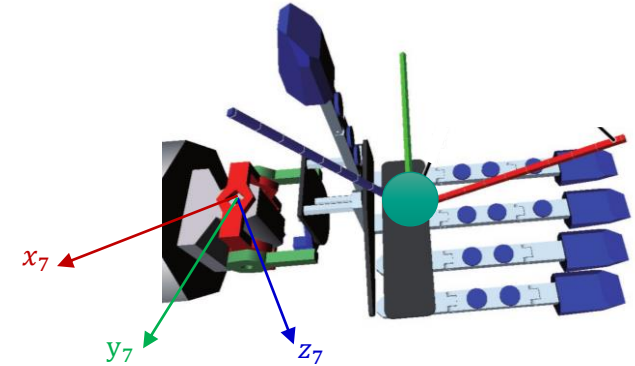
$$\begin{aligned} S_{\text{base},m}(\boldsymbol{\theta}) &= A_{0,1}(\theta_1) \cdot A_{1,2}(\theta_2) \cdot \dots \cdot A_{m-2,m-1}(\theta_{m-1}) \cdot A_{m-1,m}(\theta_m) \\ &= \begin{pmatrix} R & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{pmatrix} \end{aligned}$$

- This is a mapping of the configuration space $C \subset \mathbb{R}^n$ to the workspace $W \subset \mathbb{R}^m$

$$\mathbb{R}^n \rightarrow \mathbb{R}^m: \quad \mathbf{x} = \mathbf{f}(\boldsymbol{\theta})$$

DH Parameters – Notes

- The **four parameters** a_i , α_i , d_i and θ_i are called **DH parameters**.
- **Important:** Reference coordinate system (RCS) and end effector coordinate system (ECS) of the kinematic chain
 - As intuitive as possible; set so that the associated DH parameters are simple (preferably zero)
 - RCS as the coordinate system of the first joint in zero position
 - End effector coordinate system at an 'important reference point' at the end effector



Summary: Determination of the DH Parameters

1. **Sketch** of the manipulator
2. Identify and **enumerate** the **joints** (1, ..., last link = n)
3. Draw the **axes** z_{i-1} for **each joint i**
4. Determine the **parameters** a_i between z_{i-1} and z_i
5. Draw the **x_i -axes**
6. Determine the **parameters** α_i (twist around the x_i -axes)
7. Determine the **parameters** d_i (link offset)
8. Determine the **angles** θ_i around the z_{i-1} -axes
9. Compose the **joint transformation matrices** $A_{i-1,i}$

Contents

■ Kinematic Model

- Kinematic Chain
- Denavit-Hartenberg Convention
- **Direct Kinematics Problem**
- Examples
- Jacobian Matrix
- Singularities and Manipulability
- Representation of Reachability

■ Geometric Model

- Areas of Application
- Classification
- Examples

Direct Kinematics Problem (1)

- **Direct** kinematics problem
 - Input: Joint angles of the robot
 - Output: **Pose of the end effector**

**Forward Kinematics:
HERE!**

End effector

**Where is my
hand?**



Direct Kinematics Problem (2)

- The pose of the end effector (EEF) is to be determined from the DH parameters and the joint angles.
- The pose of the end effector (EEF) in relation to the RCS is given by:

$$\begin{aligned} S_{base,EEF}(\theta) &= A_{0,1}(\theta_1) \cdot A_{1,2}(\theta_2) \cdot \dots \cdot A_{n-2,n-1}(\theta_{n-1}) \cdot A_{n-1,n}(\theta_n) \\ &= \begin{pmatrix} R & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{pmatrix} \end{aligned}$$

- Joint angles $\theta_1, \dots, \theta_n$ are given \Rightarrow **The pose of the EEF is obtained from the equation above by inserting the joint angle values.**

Contents

Kinematic Model

- Kinematic Chain

- Denavit-Hartenberg Convention

- Direct Kinematics Problem

Examples

- Jacobian Matrix

- Singularities and Manipulability

- Representation of Reachability

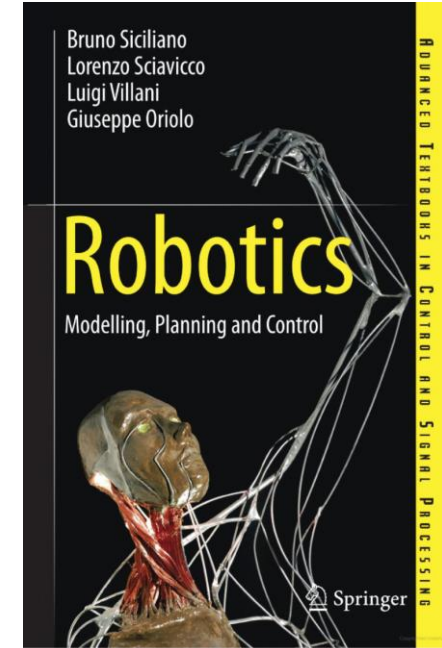
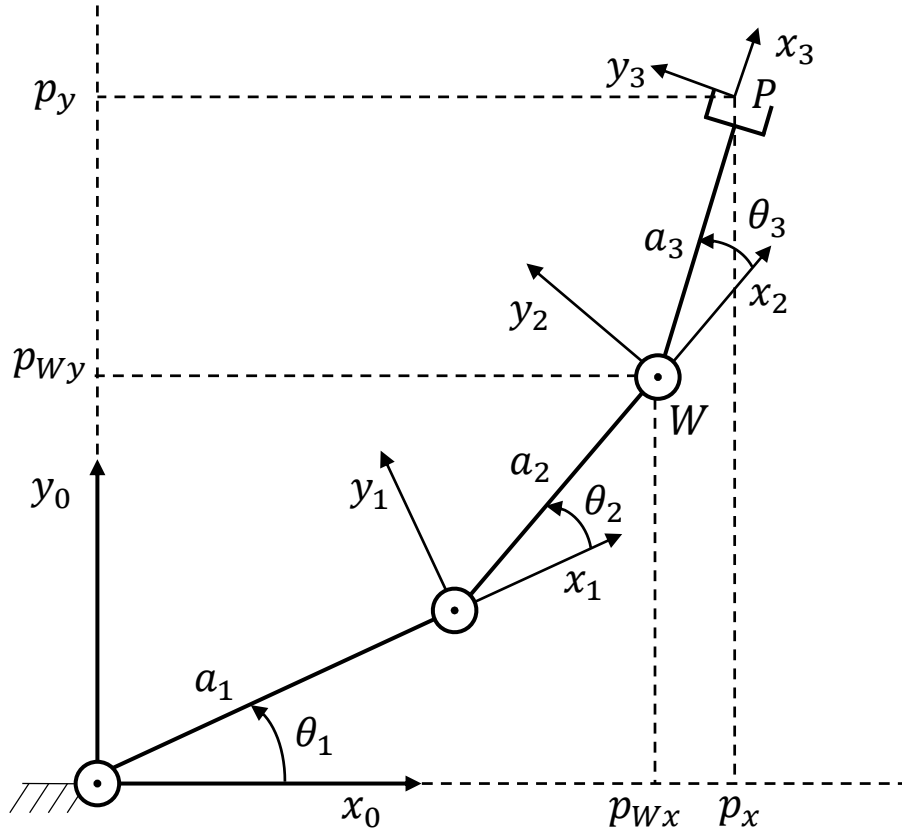
Geometric Model

- Areas of Application

- Classification

- Examples

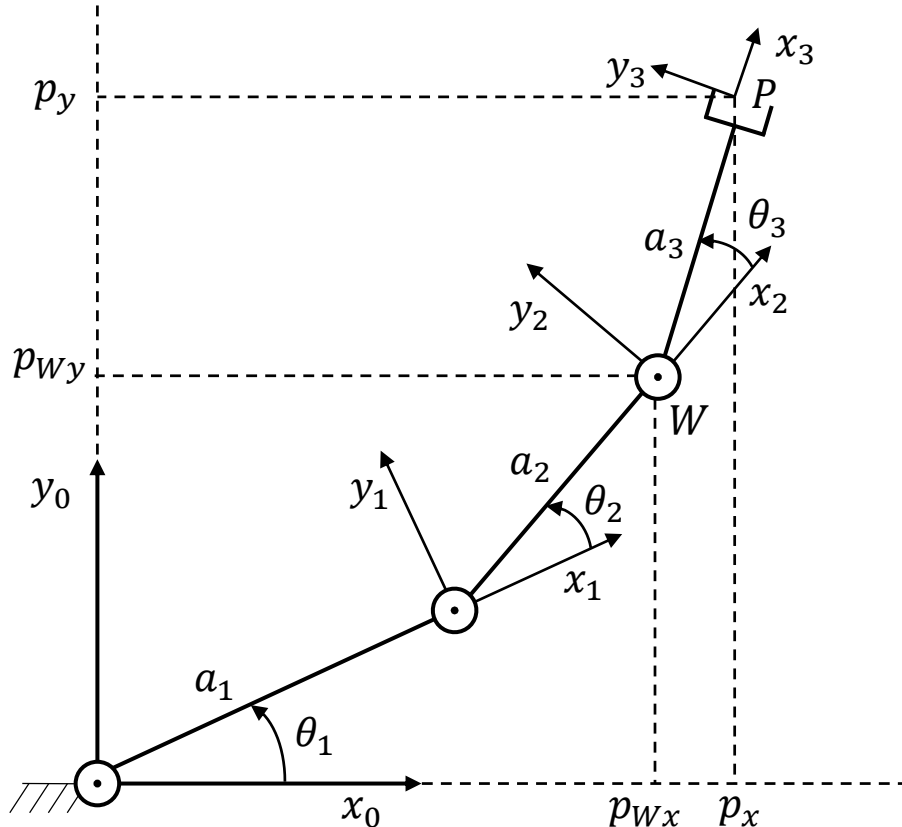
Example 1: Planar Robot (in xy-Plane)



Example 1: Planar Robot

z-axes are parallel

No translation in z-direction



Joint	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1
2	a_2	0	0	θ_2
3	a_3	0	0	θ_3

$$A_{i-1,i} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cdot \cos \alpha_i & \sin \theta_i \cdot \sin \alpha_i & a_i \cdot \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cdot \cos \alpha_i & -\cos \theta_i \cdot \sin \alpha_i & a_i \cdot \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

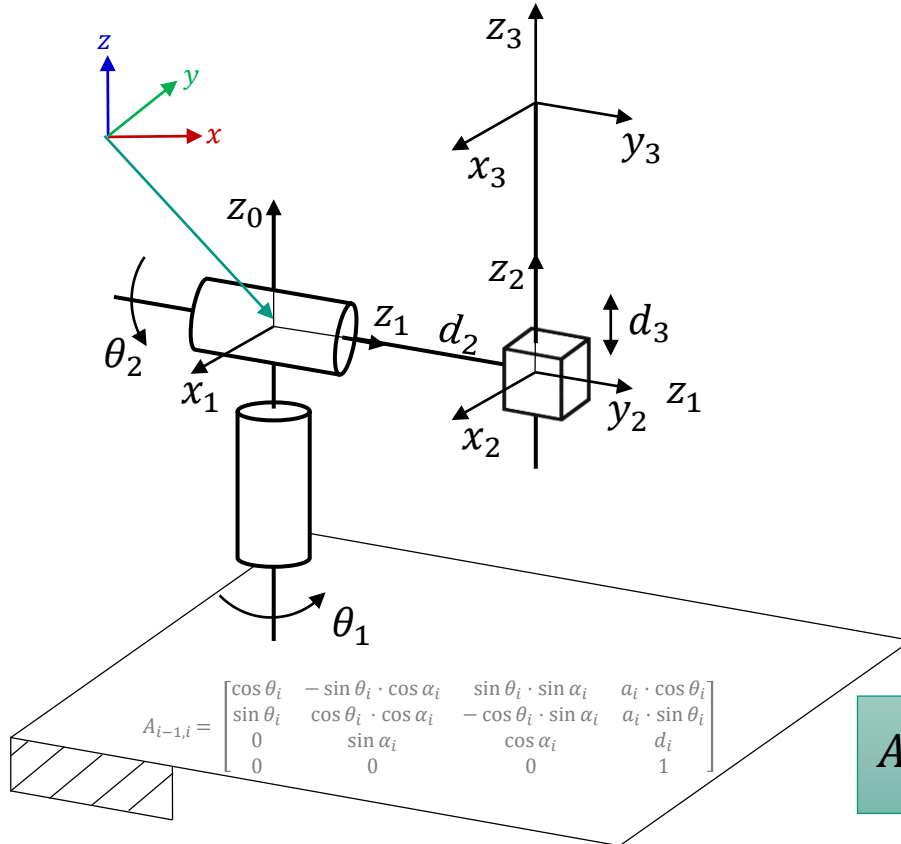
$$A_{i-1,i} = \begin{bmatrix} c_i & -s_i & 0 & a_i c_i \\ s_i & c_i & 0 & a_i s_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Example 1: Planar Robot

$$A_{0,3}(\theta) = A_{0,1} \cdot A_{1,2} \cdot A_{2,3} = \begin{bmatrix} c_{123} & -s_{123} & 0 & a_1 c_1 + a_2 c_{12} + a_3 c_{123} \\ s_{123} & c_{123} & 0 & a_1 s_1 + a_2 s_{12} + a_3 s_{123} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Abbreviations: $c_{123} = \cos(\theta_1 + \theta_2 + \theta_3)$, $s_{123} = \sin(\theta_1 + \theta_2 + \theta_3)$, etc.

Example 2: 3D Robot

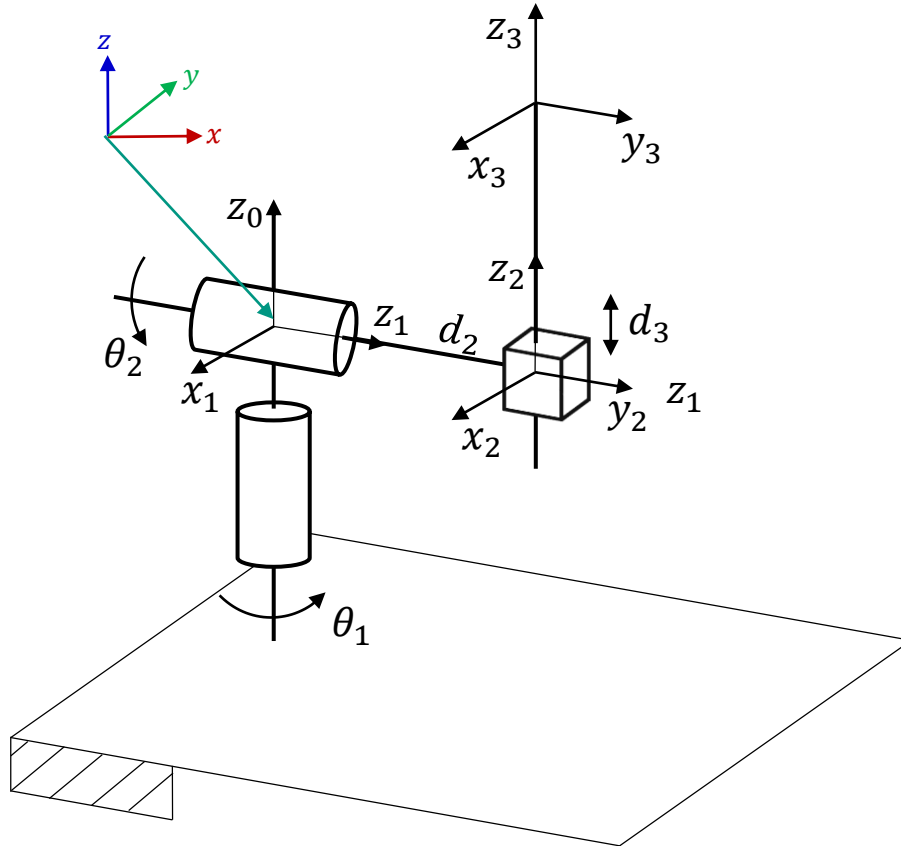


Joint	a_i	α_i	d_i	θ_i
1	0	-90	0	θ_1
2	0	90	d_2	θ_2
3	0	0	d_3	0

$$A_{0,1} = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_{i-1,i} = R_{z_{i-1}}(\theta_i) \cdot T_{z_{i-1}}(d_i) \cdot T_{x_i}(a_i) \cdot R_{x_i}(\alpha_i)$$

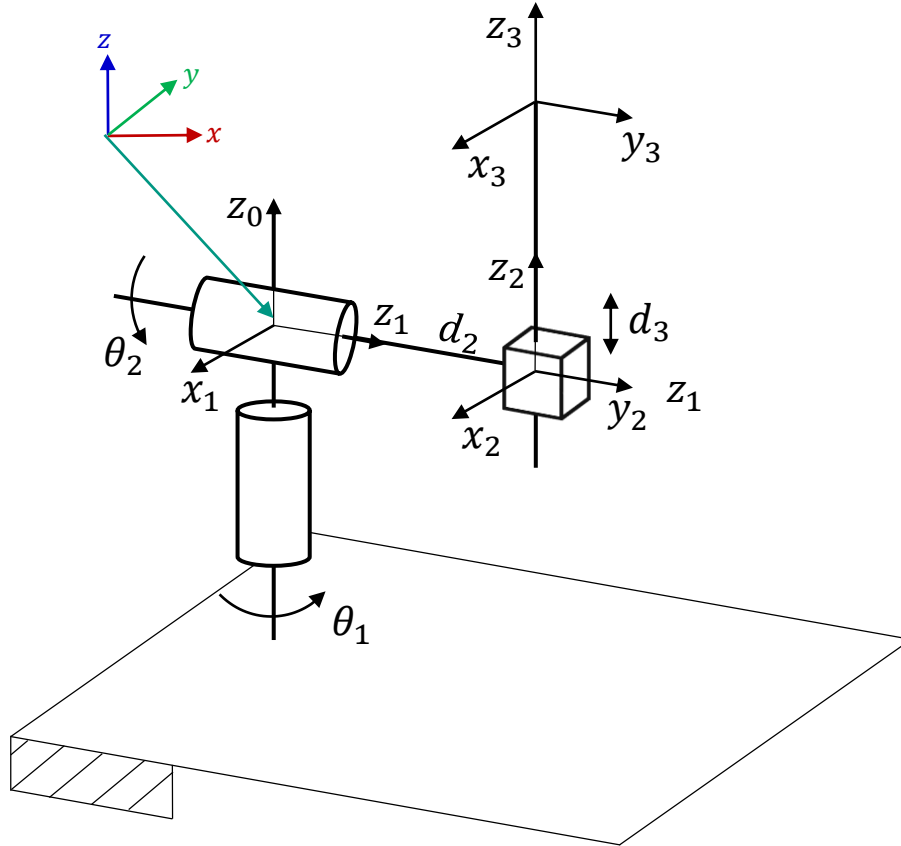
Example 2: 3D Robot



Joint	a_i	α_i	d_i	θ_i
1	0	-90	0	θ_1
2	0	90	d_2	θ_2
3	0	0	d_3	0

$$A_{1,2} = \begin{bmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & -c_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Example 2: 3D Robot



Joint	a_i	α_i	d_i	θ_i
1	0	-90	0	θ_1
2	0	90	d_2	θ_2
3	0	0	d_3	0

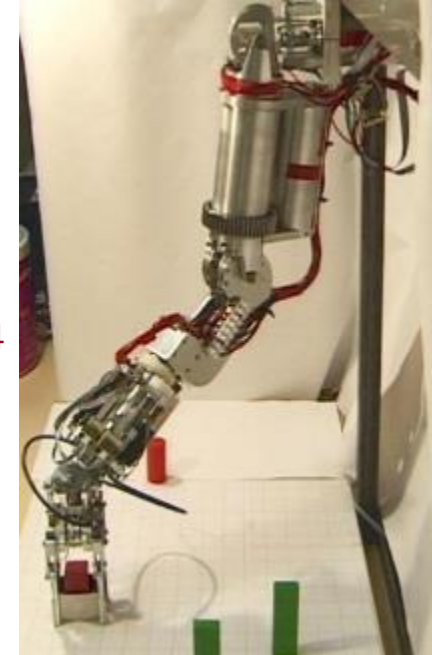
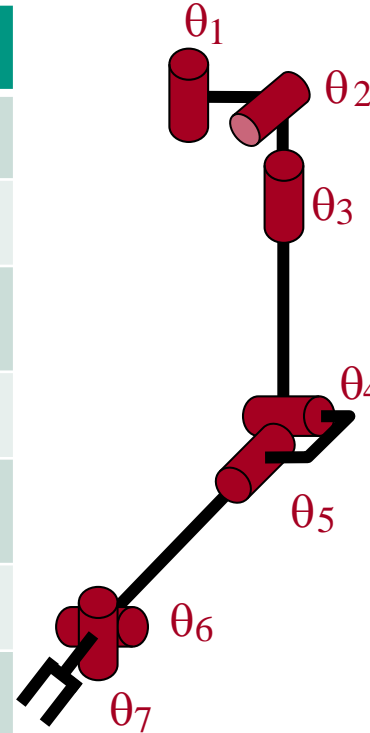
$$A_{2,3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Example 2: 3D Robot

$$A_{0,3}(\theta) = A_{0,1} \cdot A_{1,2} \cdot A_{2,3} = \begin{bmatrix} c_1 c_2 & -s_1 & c_1 s_2 & c_1 s_2 d_3 - s_1 d_2 \\ s_1 c_2 & c_1 & s_1 s_2 & s_1 s_2 d_3 + c_1 d_2 \\ -s_2 & 0 & c_2 & c_2 d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

DH Notation: Arm of ARMAR-III

Joint i	θ_i [°]	a_i [mm]	α_i [°]	d_i [mm]
1	θ_1	30	-90	0
2	$\theta_2 - 90$	0	-90	0
3	$\theta_3 + 90$	0	90	223,5
4	θ_4	0	-90	0
5	θ_5	0	90	270
6	$\theta_6 + 90$	0	-90	0
7	θ_7	140	90	0



Forward Kinematics (1)

■ **Given:** $\theta, A_{i-1,i}(\theta)$

■ **Desired:** $S_{base,EEF}(\theta)$

$$A_{i-1,i}(\theta) = R_{z_{i-1}}(\theta_i) \cdot T_{z_{i-1}}(d_i) \cdot T_{x_i}(a_i) \cdot R_{x_i}(\alpha_i)$$

$$= \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cdot \cos \alpha_i & \sin \theta_i \cdot \sin \alpha_i & a_i \cdot \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cdot \cos \alpha_i & -\cos \theta_i \cdot \sin \alpha_i & a_i \cdot \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$S_{base,EEF}(\theta) = A_{0,1}(\theta_1) \cdot A_{1,2}(\theta_2) \cdot \dots \cdot A_{n-2,n-1}(\theta_{n-1}) \cdot A_{n-1,n}(\theta_n)$$

$$= \begin{pmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix}$$

Forward Kinematics (2)

- Pose of the end effector coordinate system relative to the base:

$$S_{base,EEF}(\boldsymbol{\theta}) = A_{0,1}(\theta_1) \cdot A_{1,2}(\theta_2) \cdot \dots \cdot A_{n-2,n-1}(\theta_{n-1}) \cdot A_{n-1,n}(\theta_n)$$

- This is a mapping of the configuration space $C \subset \mathbb{R}^n$ to the workspace $W \subset \mathbb{R}^m$

$$\mathbb{R}^n \rightarrow \mathbb{R}^m: \quad \boldsymbol{x} = \boldsymbol{f}(\boldsymbol{\theta})$$

Derivation of the Forward Kinematics

- Forward Kinematics: **Joint angle position** → **end effector pose**

$$\mathbb{R}^n \rightarrow \mathbb{R}^m: \mathbf{x}(t) = \mathbf{f}(\boldsymbol{\theta}(t))$$

Pose of the EEF in W

Joint angle vector in C

- How do the corresponding relationships look like?

- Joint angular velocities → end effector velocities
- Joint torques → end effector forces and torques

- **Approach**: Derive forward kinematics → **Jacobian matrix**

Contents

■ Kinematic Model

- Kinematic Chain
- Denavit-Hartenberg Convention
- Direct Kinematics Problem
- Examples
- **Jacobian Matrix**
- Singularities and Manipulability
- Representation of Reachability

■ Geometric Model

- Areas of Application
- Classification
- Examples

Reminder: Jacobian Matrix

- Given a differentiable function

$$f: \mathbb{R}^n \rightarrow \mathbb{R}^m \quad \left\{ \begin{array}{l} \text{i.e. } \mathbf{f}(x) = \begin{pmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{pmatrix} \end{array} \right.$$

- The **Jacobian Matrix** contains all first-order partial derivatives of f .
For $\mathbf{a} \in \mathbb{R}^n$:

$$J_f(\mathbf{a}) = \left(\frac{\partial f_i}{\partial x_j}(\mathbf{a}) \right)_{i,j} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(\mathbf{a}) & \dots & \frac{\partial f_1}{\partial x_n}(\mathbf{a}) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1}(\mathbf{a}) & \dots & \frac{\partial f_m}{\partial x_n}(\mathbf{a}) \end{pmatrix} \in \mathbb{R}^{m \times n}$$

f_1, \dots, f_m denote the component functions of f and x_1, \dots, x_n the coordinates in \mathbb{R}^n .

Jacobian Matrix in Forward Kinematics

■ **Problem:** Forward kinematics is matrix-valued (n : number of joints)

$$f: \mathbb{R}^n \rightarrow \text{SE}(3)$$

⇒ Jacobian matrix not defined

■ **Solution:** Select vector representation,
e.g. use roll, pitch and yaw angles to represent orientations

$$f: \mathbb{R}^n \rightarrow \mathbb{R}^6 \left\{ \begin{pmatrix} x \\ y \\ z \\ \alpha \\ \beta \\ \gamma \end{pmatrix} \right.$$

End Effector Velocities

- **Assumption:** The kinematic chain moves along a trajectory

$$\theta: \mathbb{R} \rightarrow \mathbb{R}^n$$

- Pose of the end effector $\mathbf{x}(t) \in \mathbb{R}^6$ at time t :

$$\mathbf{x}(t) = f(\theta(t))$$

- The end effector velocity depends linearly on the joint velocities (**chain rule**):

$$\dot{\mathbf{x}}(t) = \frac{\partial f(\theta(t))}{\partial t} = \frac{\partial f(\theta(t))}{\partial \theta} \cdot \frac{\partial \theta(t)}{\partial t} = J_f(\theta(t)) \cdot \dot{\theta}(t)$$

End Effector Velocities

- The Jacobian matrix relates Cartesian end effector velocities to joint angle velocities

$$\dot{\mathbf{x}}(t) = J_f(\theta(t)) \cdot \dot{\theta}(t)$$

- The following problems can be solved with this relation:
 - Forward kinematics in the velocity space:
Given joint angle velocities,
which Cartesian end effector velocities are realized?
 - Inverse kinematics in the velocity space:
Given Cartesian end effector velocities,
which joint angle velocities are necessary to realize them?

Kinematics using the Jacobian Matrix (1)

■ Forward kinematics:

Given the joint angle velocities $\dot{\theta}(t)$,
which Cartesian end effector velocities $\dot{x}(t)$ are realized?

■ Insert $\dot{\theta}(t)$:


$$\dot{x}(t) = J_f(\theta(t)) \cdot \dot{\theta}(t)$$

Kinematics using the Jacobian Matrix (2)

Inverse kinematics:

Given a Cartesian end effector velocities $\dot{\mathbf{x}}(t)$,
which joint angle velocities $\dot{\boldsymbol{\theta}}(t)$ are necessary to realize them?

$$\dot{\mathbf{x}}(t) = J_f(\boldsymbol{\theta}(t)) \cdot \dot{\boldsymbol{\theta}}(t)$$

 $J_f^{-1}(\boldsymbol{\theta}(t)) \cdot [\quad]$

$$\dot{\boldsymbol{\theta}}(t) = J_f^{-1}(\boldsymbol{\theta}(t)) \cdot \dot{\mathbf{x}}(t)$$

Forces and Torques at the End Effector

- **Assumption:** The kinematic chain moves along a trajectory

$$\theta: \mathbb{R} \rightarrow \mathbb{R}^n$$

- The **work** done (force · distance) must remain constant regardless of the reference system (**friction neglected**) 忽略摩擦

$$\int_{t_1}^{t_2} \dot{\theta}(t)^T \cdot \tau(t) dt = W = \int_{t_1}^{t_2} \dot{x}(t)^T \cdot F(t) dt$$

- With:

$\dot{\theta}(t): \mathbb{R} \rightarrow \mathbb{R}^n$, Joint velocities

$\tau(t): \mathbb{R} \rightarrow \mathbb{R}^n$, Joint torques

$\dot{x}(t): \mathbb{R} \rightarrow \mathbb{R}^6$, End effector velocities

$F(t): \mathbb{R} \rightarrow \mathbb{R}^6$, Force-torque vector at the end effector

Forces and Torques at the End Effector

$$\int_{t_1}^{t_2} \dot{\theta}(t)^T \cdot \tau(t) dt = W = \int_{t_1}^{t_2} \dot{x}(t)^T \cdot F(t) dt$$

- The relation must apply for each time interval $[t_1, t_2]$, therefore:

$$\dot{\theta}(t)^T \cdot \tau(t) = \dot{x}(t)^T \cdot F(t)$$

- Known relation between end effector velocity and Jacobian matrix:

$$\dot{\theta}(t)^T \cdot \tau(t) = \dot{\theta}(t)^T \cdot J_f^T(\theta(t)) \cdot F(t) \quad \dot{x}(t) = J_f(\theta(t)) \cdot \dot{\theta}(t)$$

- Since $\dot{\theta}(t)$ is arbitrary, it follows that:

$$\tau(t) = J_f^T(\theta(t)) \cdot F(t)$$

Forces and Torques at the End Effector

- The Jacobian matrix relates forces and torques at the end effector to the torques in the joints:

$$\tau(t) = J_f^T(\theta(t)) \cdot F(t)$$

- The following problems can be solved with this relation:
 - Given forces/torques at the end effector,
which torques must act in the joints to resist this force?
 - Given the torques in the joints,
which resulting forces and torques act on the (fixed) end-effector?

Recap – DH Transformation Matrices

Transformation from LCS_{i-1} to LCS_i

$$\begin{aligned} A_{i-1,i} &= R_{z_{i-1}}(\theta_i) \cdot T_{z_{i-1}}(d_i) \cdot T_{x_i}(a_i) \cdot R_{x_i}(\alpha_i) \\ &= \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cdot \cos \alpha_i & \sin \theta_i \cdot \sin \alpha_i & a_i \cdot \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cdot \cos \alpha_i & -\cos \theta_i \cdot \sin \alpha_i & a_i \cdot \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

Recap – Direct and Inverse Kinematics

Joint angle space
(configuration space)

Transformation

Cartesian coordinates
(workspace)

Direct Kinematics

$$x = f(\theta)$$

$$(\theta_1, \dots, \theta_n) \in \mathcal{C} \subseteq \mathbb{R}^n$$

$$X \subset \mathbb{R}^m$$

Inverse Kinematics

$$\theta = f^{-1}(x)$$

E.g. position and orientation
of the end effector

$$x_{EEF} = (x, y, z, \alpha, \beta, \gamma)$$

n : Joint degrees of freedom (DoF)
 m : Cartesian degrees of freedom

Recap – Jacobian Matrix

$$\mathbf{x} = f(\boldsymbol{\theta})$$

$$\dot{\mathbf{x}} = J_f(\boldsymbol{\theta}) \cdot \dot{\boldsymbol{\theta}}$$

$$J_f(\boldsymbol{\theta}) = \left(\frac{\partial f_i}{\partial \theta_j}(\boldsymbol{\theta}) \right)_{i,j} = \begin{pmatrix} \frac{\partial f_1}{\partial \theta_1}(\boldsymbol{\theta}) & \dots & \frac{\partial f_1}{\partial \theta_n}(\boldsymbol{\theta}) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial \theta_1}(\boldsymbol{\theta}) & \dots & \frac{\partial f_m}{\partial \theta_n}(\boldsymbol{\theta}) \end{pmatrix} \in \mathbb{R}^{m \times n}$$

$$\mathbf{x} = (\mathbf{x}, \mathbf{y}, z, \alpha, \beta, \boldsymbol{\gamma})^T \in \mathbb{R}^{m=6} \text{ and } \boldsymbol{\theta} \in \mathbb{R}^n$$

$$J_f(\boldsymbol{\theta}) = \left(\frac{\partial f_i}{\partial \theta_j}(\boldsymbol{\theta}) \right)_{i,j} = \begin{pmatrix} \frac{\partial \mathbf{x}}{\partial \theta_1}(\boldsymbol{\theta}) & \dots & \frac{\partial \mathbf{x}}{\partial \theta_n}(\boldsymbol{\theta}) \\ \frac{\partial \mathbf{y}}{\partial \theta_1}(\boldsymbol{\theta}) & \dots & \frac{\partial \mathbf{y}}{\partial \theta_n}(\boldsymbol{\theta}) \\ \vdots & \ddots & \vdots \\ \frac{\partial \boldsymbol{\gamma}}{\partial \theta_1}(\boldsymbol{\theta}) & \dots & \frac{\partial \boldsymbol{\gamma}}{\partial \theta_n}(\boldsymbol{\theta}) \end{pmatrix} \in \mathbb{R}^{6 \times n}$$

Recap – Jacobian Matrix

■ Velocity space

$$\dot{x}(t) = J_f(\theta(t)) \cdot \dot{\theta}(t)$$

■ Force space

$$\tau(t) = J_f^T(\theta(t)) \cdot F(t)$$

Calculation of the Jacobian Matrix

- Each column of the Jacobian matrix corresponds to a joint θ_i of the kinematic chain

$$J_f = \begin{pmatrix} \frac{\partial f}{\partial \theta_1} & \dots & \frac{\partial f}{\partial \theta_n} \end{pmatrix} \in \mathbb{R}^{m \times n}$$

$$J_f(\boldsymbol{\theta}) = \left(\frac{\partial f_i}{\partial x_j}(\boldsymbol{\theta}) \right)_{i,j} = \begin{pmatrix} \frac{\partial f_1}{\partial \theta_1}(\boldsymbol{\theta}) & \dots & \frac{\partial f_1}{\partial \theta_n}(\boldsymbol{\theta}) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial \theta_1}(\boldsymbol{\theta}) & \dots & \frac{\partial f_m}{\partial \theta_n}(\boldsymbol{\theta}) \end{pmatrix} \in \mathbb{R}^{m \times n}$$

- **Approach:**

The numerical calculation of the Jacobian matrix is carried out **column by column**

⇒ joint by joint

Geometric Calculation of the Jacobian Matrix

$$\mathbf{x} = f(\theta)$$

$$\mathbf{x} = (x, y, z, \alpha, \beta, \gamma)^T \in \mathbb{R}^{m=6} \text{ and } \theta \in \mathbb{R}^{n=6}$$

$$\dot{\mathbf{x}} = J(\theta) \cdot \dot{\theta}$$

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{pmatrix} = \begin{pmatrix} j_{11} & j_{12} \\ j_{21} & j_{22} \\ j_{31} & j_{32} \\ j_{41} & j_{42} \\ j_{51} & j_{52} \\ j_{61} & j_{62} \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{pmatrix} \Rightarrow \begin{pmatrix} \dot{v} \\ \dot{\omega} \end{pmatrix} = (J_1(\theta), J_2(\theta), \dots, J_6(\theta)) \cdot \dot{\theta}$$

Geometric Calculation of the Jacobian Matrix

1. Case: **Prismatic joint** 移动关节

- **Assumption:** The j -th joint performs a translation in direction of the unit vector $\mathbf{z}_j \in \mathbb{R}^3$.
- It follows:

$$J_j(\boldsymbol{\theta}) = \frac{\partial f(\boldsymbol{\theta})}{\partial \theta_j} = \begin{bmatrix} \mathbf{z}_j \\ \mathbf{0} \end{bmatrix} \in \mathbb{R}^6$$

2. Case: **Revolute joint** 旋转关节

- **Assumption:** The j -th joint performs a rotation around the axis $\mathbf{z}_j \in \mathbb{R}^3$ at the position $\mathbf{p}_j \in \mathbb{R}^3$.
- It follows:

$$J_j(\boldsymbol{\theta}) = \frac{\partial f(\boldsymbol{\theta})}{\partial \theta_j} = \begin{bmatrix} \mathbf{z}_j \times (f(\boldsymbol{\theta}) - \mathbf{p}_j) \\ \mathbf{z}_j \end{bmatrix} \in \mathbb{R}^6$$

Geometric Calculation of the Jacobian Matrix

2. Case: **Revolute joint**

$$J_j(\boldsymbol{\theta}) = \frac{\partial f(\boldsymbol{\theta})}{\partial \theta_j} = \begin{bmatrix} \mathbf{z}_j \times (f(\boldsymbol{\theta}) - \mathbf{p}_j) \\ \mathbf{z}_j \end{bmatrix} \in \mathbb{R}^6$$

■ Manipulator with n joints

$$J(\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{z}_0 \times (f(\boldsymbol{\theta}) - \mathbf{p}_0) & \mathbf{z}_1 \times (f(\boldsymbol{\theta}) - \mathbf{p}_1) & \dots & \mathbf{z}_{n-1} \times (f(\boldsymbol{\theta}) - \mathbf{p}_{n-1}) \\ \mathbf{z}_0 & \mathbf{z}_1 & \dots & \mathbf{z}_{n-1} \end{bmatrix}$$

Summary: Jacobian Matrix

■ Forward kinematics:

$$f: \mathbb{R}^n \rightarrow \mathbb{R}^6, \quad f(\boldsymbol{\theta}) = \boldsymbol{x} = (x, y, z, \alpha, \beta, \gamma)$$

■ Jacobian matrix:

$$J_f = \begin{pmatrix} \frac{\partial f}{\partial \theta_1} & \dots & \frac{\partial f}{\partial \theta_n} \end{pmatrix} \in \mathbb{R}^{6 \times n}$$

■ Properties:

■ J_f describes the relations between

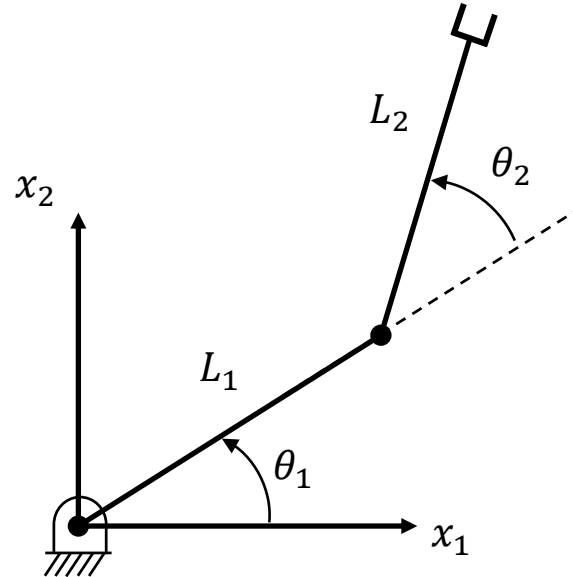
- Joint angle velocities (n -dimensional)
and end effector velocities (6-dimensional)
- Joint torques (n -dimensional)
and forces and torques at the end effector (6-dimensional)

■ The Jacobian matrix depends on the joint angle configuration

Jacobian Matrix: Example (1)

■ Manipulator with two joints θ_1, θ_2

■ Find $\dot{\mathbf{x}}$



Jacobian Matrix: Example (2)

■ Forward kinematics

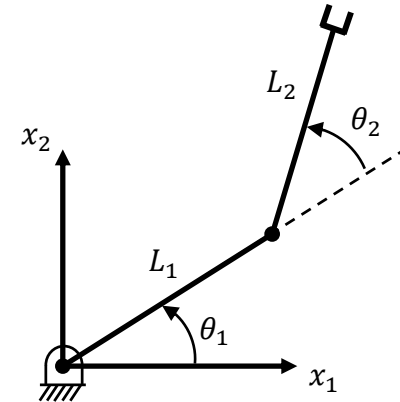
$$\mathbf{x} = f(\boldsymbol{\theta})$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = f \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}$$

■ Velocity of the end effector

$$\dot{\mathbf{x}} = J_f(\boldsymbol{\theta}) \cdot \dot{\boldsymbol{\theta}}$$

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = J_f(\boldsymbol{\theta}) \cdot \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix}$$



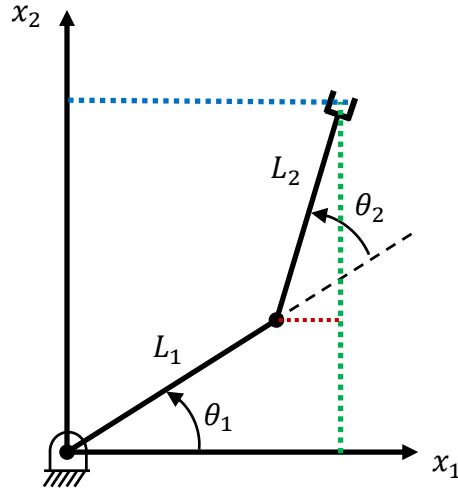
Jacobian Matrix: Example (3)

■ Forward kinematics

$$x_1 = L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2)$$

$$x_2 = L_1 \sin \theta_1 + L_2 \sin(\theta_1 + \theta_2)$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = f \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}$$



Jacobian Matrix: Example (4)

Forward kinematics

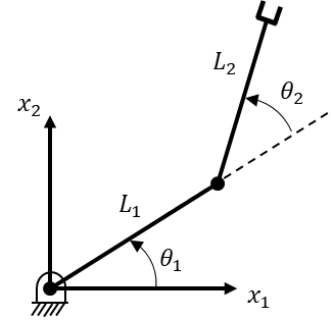
$$\begin{aligned}x_1 &= L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) \\x_2 &= L_1 \sin \theta_1 + L_2 \sin(\theta_1 + \theta_2)\end{aligned}$$

Derivation

$$\begin{aligned}\dot{x}_1 &= -L_1 \dot{\theta}_1 \sin \theta_1 - L_2 (\dot{\theta}_1 + \dot{\theta}_2) \sin(\theta_1 + \theta_2) \\ \dot{x}_2 &= L_1 \dot{\theta}_1 \cos \theta_1 + L_2 (\dot{\theta}_1 + \dot{\theta}_2) \cos(\theta_1 + \theta_2)\end{aligned}$$

Jacobian matrix

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \underbrace{\begin{pmatrix} -L_1 \sin \theta_1 - L_2 \sin(\theta_1 + \theta_2) \\ L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) \end{pmatrix}}_{J_1(\theta)} \underbrace{\begin{pmatrix} -L_2 \sin(\theta_1 + \theta_2) \\ L_2 \cos(\theta_1 + \theta_2) \end{pmatrix}}_{J_2(\theta)} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix}$$



Jacobian Matrix: Example (5)

End effector velocity

$$\mathbf{v}_{EEF} = J_1(\boldsymbol{\theta})\dot{\theta}_1 + J_2(\boldsymbol{\theta})\dot{\theta}_2$$

As long as $J_1(\boldsymbol{\theta})$ and $J_2(\boldsymbol{\theta})$ are **not linearly dependent**, \mathbf{v}_{EEF} can be generated in any direction in the x_1x_2 -plane.

Singularities

$J_1(\theta)$ and $J_2(\theta)$ linearly dependent
 $\rightarrow J(\theta)$ becomes singular

E.g. if $\theta_2 = 0^\circ$

The possible movements of the end effector are restricted.

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -L_1 \sin \theta_1 - L_2 \sin(\theta_1 + \theta_2) & -L_2 \sin(\theta_1 + \theta_2) \\ L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) & L_2 \cos(\theta_1 + \theta_2) \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix}$$

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \underbrace{\begin{pmatrix} -L_1 \sin \theta_1 - L_2 \sin(\theta_1) & -L_2 \sin(\theta_1) \\ L_1 \cos \theta_1 + L_2 \cos(\theta_1) & L_2 \cos(\theta_1) \end{pmatrix}}_{\begin{pmatrix} -(L_1 + L_2) \sin \theta_1 & -L_2 \sin \theta_1 \\ (L_1 + L_2) \cos \theta_1 & L_2 \cos \theta_1 \end{pmatrix}} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix}$$

Contents

■ Kinematic Model

- Kinematic Chain
- Denavit-Hartenberg Convention
- Direct Kinematics Problem
- Examples
- Jacobian Matrix
- **Singularities and Manipulability**
- Representation of Reachability

■ Geometric Model

- Areas of Application
- Classification
- Examples

Recap – Jacobian Matrix

■ Velocity space

$$\dot{x}(t) = J_f(\theta(t)) \cdot \dot{\theta}(t)$$

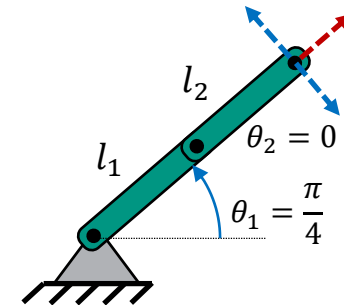
■ Force space

$$\tau(t) = J_f^T(\theta(t)) \cdot F(t)$$

Kinematic Singularities

- If a robot is in a configuration $\theta_{singular} \in C$ in which it is no longer able to move instantaneously in one or more directions, this is referred to as a **kinematic singularity**.
- Configurations $\theta_{singular} \in C$ that lead to a kinematic singularity are called **singular**.
- Can we distinguish singular from non-singular configurations?
→ Via the Jacobian matrix

There is no joint angular velocity that generates an end effector velocity in the **red direction**.
⇒ The configuration is singular.



Kinematic Singularities: Example

Forward kinematics: $\mathbf{x} = f(\boldsymbol{\theta}) = \begin{pmatrix} l_1 \cdot \cos \theta_1 + l_2 \cdot \cos(\theta_1 + \theta_2) \\ l_1 \cdot \sin \theta_1 + l_2 \cdot \sin(\theta_1 + \theta_2) \end{pmatrix}$

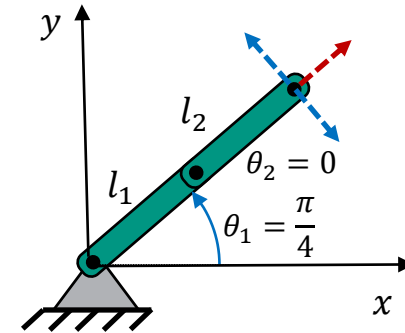
Jacobian matrix: $J(\boldsymbol{\theta}) = \begin{pmatrix} -l_1 \cdot \sin \theta_1 - l_2 \cdot \sin(\theta_1 + \theta_2) & -l_2 \cdot \sin(\theta_1 + \theta_2) \\ l_1 \cdot \cos \theta_1 + l_2 \cdot \cos(\theta_1 + \theta_2) & l_2 \cdot \cos(\theta_1 + \theta_2) \end{pmatrix}$

For the singular configuration $\boldsymbol{\theta} = \left(\frac{\pi}{4}, 0\right)^T$:

$$J\left(\begin{pmatrix} \pi/4 \\ 0 \end{pmatrix}\right) = (J_1, J_2) = \begin{pmatrix} -(l_1 + l_2) \cdot \frac{1}{\sqrt{2}} & -l_2 \cdot \frac{1}{\sqrt{2}} \\ (l_1 + l_2) \cdot \frac{1}{\sqrt{2}} & l_2 \cdot \frac{1}{\sqrt{2}} \end{pmatrix}$$

The first and second column are linearly dependent

$$J_1 = \frac{l_1 + l_2}{l_2} \cdot J_2$$



Kinematic Singularities: Jacobian Matrix (1)

Forward kinematics in the velocity space:

The end effector velocity is a **linear combination** of the columns of the Jacobian matrix.

$$J(\boldsymbol{\theta}) = \left(\frac{\partial f}{\partial \theta_1}, \quad \frac{\partial f}{\partial \theta_2}, \quad \dots, \quad \frac{\partial f}{\partial \theta_n} \right) = (J_1, J_2, \dots, J_n)$$

$$\dot{\mathbf{x}} = J(\boldsymbol{\theta}) \cdot \dot{\boldsymbol{\theta}}$$

$$\dot{\mathbf{x}} = (J_1, J_2, \dots, J_n) \cdot \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \vdots \\ \dot{\theta}_n \end{pmatrix} = J_1 \cdot \dot{\theta}_1 + J_2 \cdot \dot{\theta}_2 + \dots + J_n \cdot \dot{\theta}_n$$

Kinematic Singularities: Jacobian Matrix (2)

The end effector velocity is a **linear combination** of the columns of the Jacobian matrix.

$$\dot{\mathbf{x}} = \mathbf{J}_1 \cdot \dot{\theta}_1 + \mathbf{J}_2 \cdot \dot{\theta}_2 + \cdots + \mathbf{J}_n \cdot \dot{\theta}_n \qquad \mathbf{J}(\boldsymbol{\theta}) = (\mathbf{J}_1, \mathbf{J}_2 \dots, \mathbf{J}_n)$$

If a robot is in a configuration $\boldsymbol{\theta}_{singular} \in \mathcal{C}$ in which it is no longer able to move instantaneously in one or more directions, this is referred to as a **kinematic singularity**.

In mathematical terms, kinematic singularity means that the linear combination of Jacobian columns **does not span the entire end effector velocity space**.

The Jacobian matrix $\mathbf{J}(\boldsymbol{\theta})$ has a rank smaller than the workspace dimension.

$$rank \mathbf{J}(\boldsymbol{\theta}) < m, \qquad \dot{\mathbf{x}} \in \mathbb{R}^m$$

Kinematic Singularities: Definition

Given a forward kinematics function f

$$\mathbf{x} = f(\boldsymbol{\theta}), \quad \boldsymbol{\theta} \in \mathcal{C} \subset \mathbb{R}^n, \quad \mathbf{x} \in \mathcal{W} \subset \mathbb{R}^m$$

and the corresponding Jacobian matrix

$$J(\boldsymbol{\theta}) = \left(\frac{\partial f}{\partial \theta_1}, \quad \frac{\partial f}{\partial \theta_2}, \quad \dots, \quad \frac{\partial f}{\partial \theta_n} \right) \in \mathbb{R}^{m \times n},$$

a configuration $\boldsymbol{\theta}_{\text{singular}} \in \mathcal{C}$ is called **singular** if the rank of the Jacobian matrix is smaller than the dimension of the workspace.

$$\text{rank } J(\boldsymbol{\theta}) < m$$

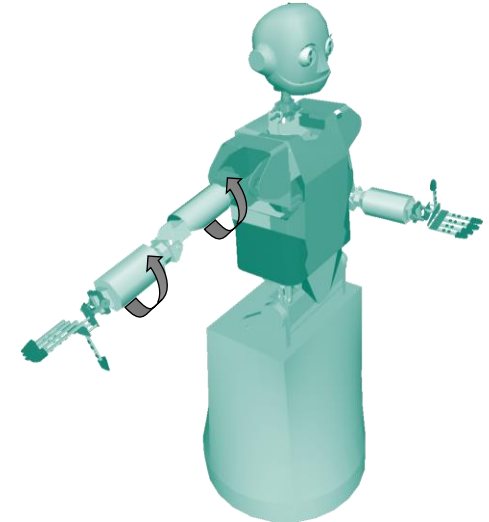
Singularities

■ Definition:

A kinematic chain is in a singular configuration if the associated Jacobian matrix is not of full rank, i.e. two or more columns of $J(\theta)$ are linearly dependent.

- A singular Jacobian matrix cannot be inverted
⇒ Certain end effector movements are impossible
- In the vicinity of singularities, large joint velocities may be necessary to maintain an end effector velocity.

在奇异点附近，可能需要较大的关节速度来保持末端执行器的速度



Manipulability

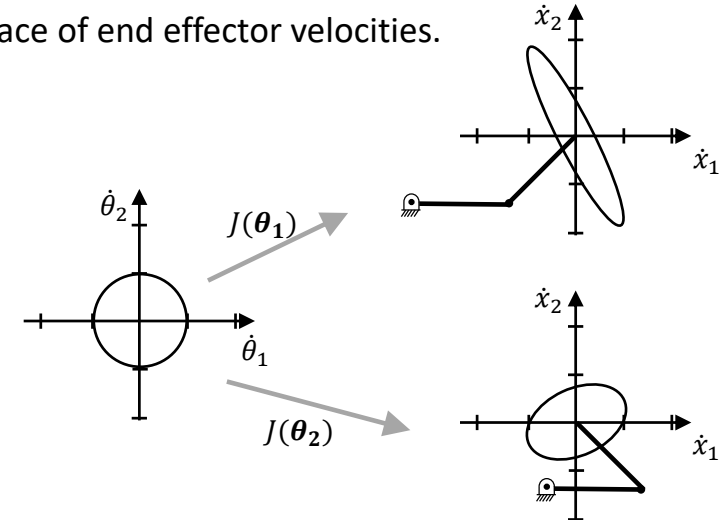
■ **Manipulability:** **Measure** of the freedom of movement of the end effector; also how 'close' a configuration is to a singularity

■ Manipulability ellipsoid

- Describes the end effector velocities for joint angle velocities with $\|\dot{\theta}\| = 1$
- Use $J(\theta)$ to map the **unit circle** of joint angle velocities to the space of end effector velocities.
- Result: **Manipulability ellipsoid**
- Depends on joint angle configuration

■ Analysis

- **Circle ('large ellipsoid'):**
End effector movement is possible without restriction in any direction.
- **Degenerate cases (compressed ellipsoid):**
End effector movement is restricted in certain directions.



Manipulability: Eigenvalue Analysis

■ Calculate $A(\theta) = J(\theta) \cdot J(\theta)^T \in R^{m \times m}$

■ $A(\theta)$ is

- Quadratic 二次型
- Symmetric 对称
- Positive definite 正定
- Invertible 可逆

■ Eigenvalues λ_i and Eigenvectors v_i of A

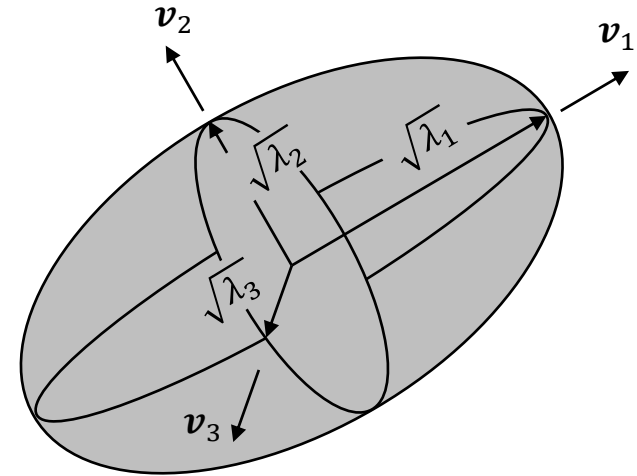
- $A v_i = \lambda_i v_i$
- $(\lambda_i I - A) v_i = 0$

■ Singular values

- $\sigma_i = \sqrt{\lambda_i}$ 奇异值与特征值关系

Volume V is proportional to

$$\sqrt{\lambda_1 \lambda_2 \dots \lambda_m} = \sqrt{\det(A)} = \sqrt{\det(JJ^T)}$$



Manipulability ellipsoid:
Geometric representation of the manipulability

Manipulability: Calculation

■ **Scalar** measures for manipulability

■ **Smallest singular value**

$$\mu_1(\theta) = \sigma_{\min}(A(\theta))$$

■ **Inverse condition**

$$\mu_2(\theta) = \frac{\sigma_{\min}(A(\theta))}{\sigma_{\max}(A(\theta))}$$

■ **Determinant** 決定因素

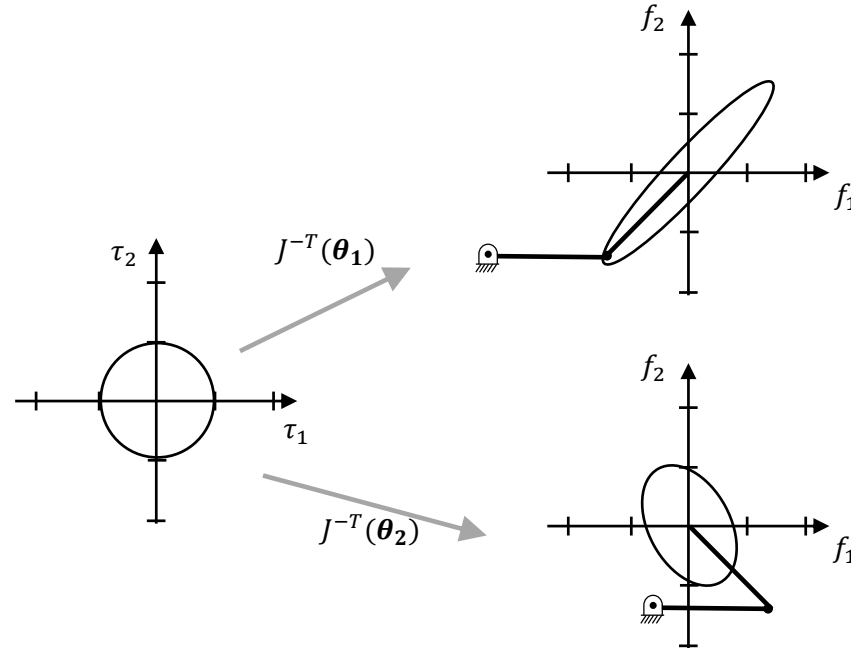
$$\mu_3(\theta) = \det A(\theta)$$

■ **Application:**

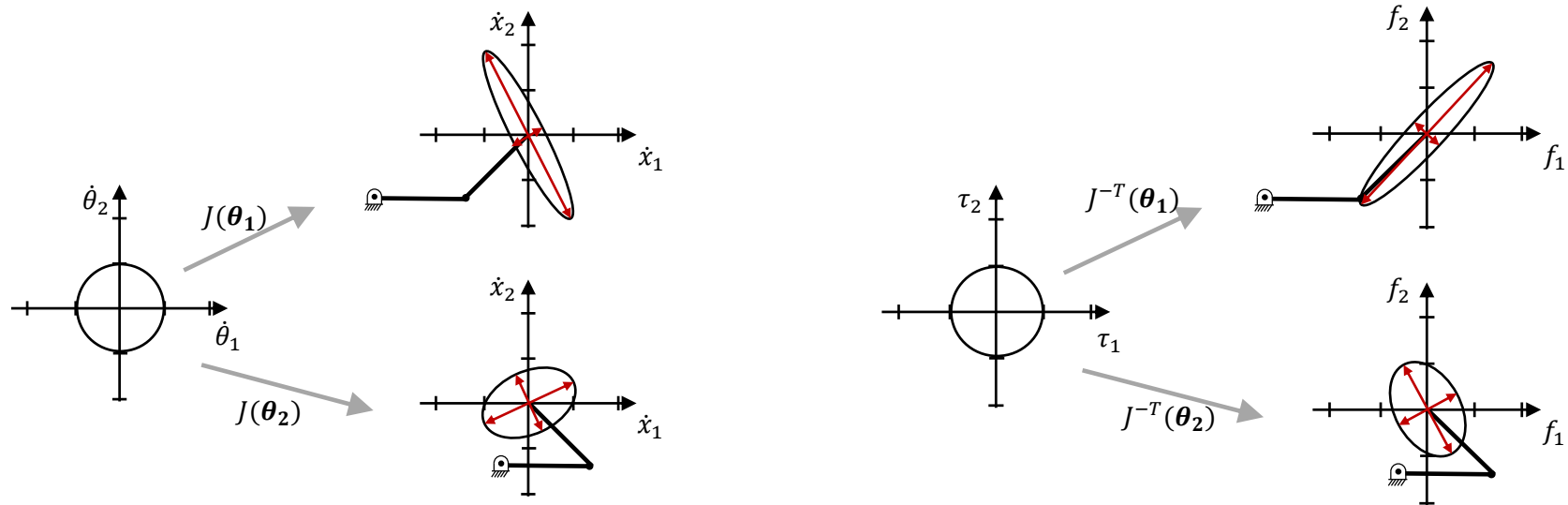
- Analysis of joint angle configurations
- Singularity avoidance

Force Ellipsoid

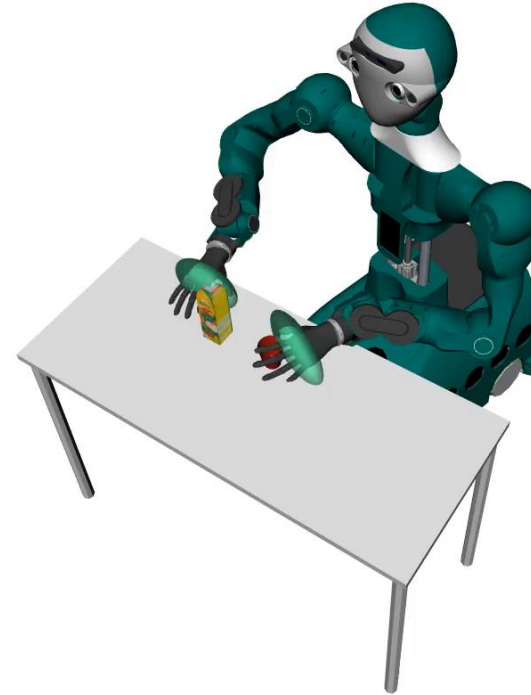
$$\tau(t) = J_f^T(\theta(t)) \cdot F(t) \quad \rightarrow \quad F(t) = J_f^{-T}(\theta(t)) \cdot \tau(t)$$



Manipulability and Force Ellipsoid



Manipulability – Examples



Contents

■ Kinematic Model

- Kinematic Chain
- Denavit-Hartenberg Convention
- Direct Kinematics Problem
- Examples
- Jacobian Matrix
- Singularities and Manipulability
- **Representation of Reachability**

■ Geometric Model

- Areas of Application
- Classification
- Examples

Joint Angle Limits

- A robot with the configuration space $C \subset \mathbb{R}^n$ generally only covers part of the underlying \mathbb{R}^n as there are **joint angle limits**.

- There is a minimum and maximum value for each joint

$$\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_n) \in C$$

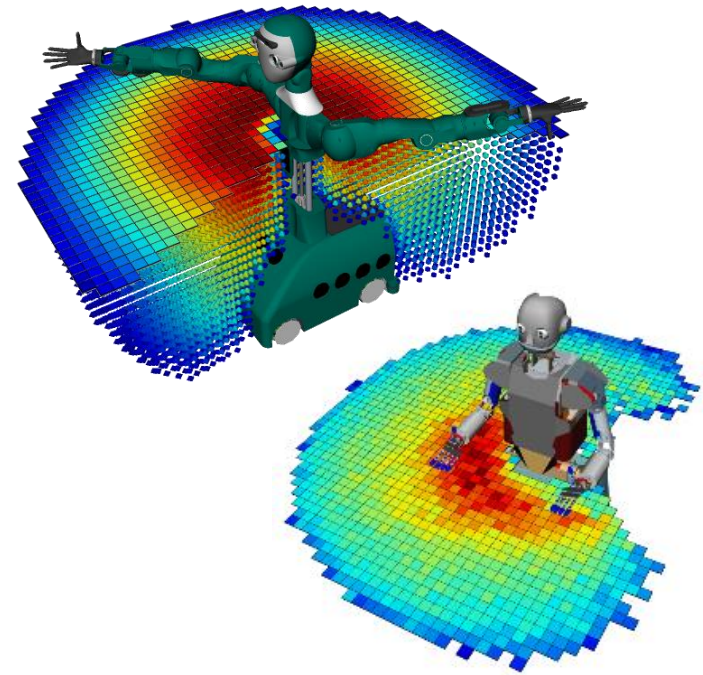
$$\theta_i \in [\theta_{i,\min}, \theta_{i,\max}]$$

- Exception: Continuous rotation joints (ARMAR-6)
- Joint angle limits restrict the **reachable part of the workspace**

$$W_{\text{reachable}} \subseteq W \subset \mathbb{R}^6$$

Representation of Reachability (1)

- Reachable part of the workspace of the robot in \mathbb{R}^6
- Approximation using a 6-dimensional grid
- Entry in each grid cell:
 - **Reachability:**
Binary: Is there at least one joint angle configuration so that the Tool Center Point (TCP) lies within the 6D grid cell?
 - **Manipulability:**
Maximum manipulability value of a grid cell, e.g. $\mu_1(\theta)$



Visualization of reachability and manipulability for the ARMAR-6 and ARMAR-III robots

Vahrenkamp, N., Asfour, T. and Dillmann, R., *Efficient Inverse Kinematics Computation based on Reachability Analysis*, International Journal of Humanoid Robotics (IJHR), vol. 9, no. 4, 2012

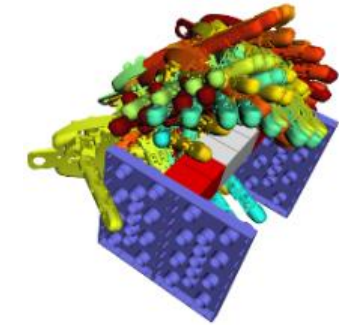
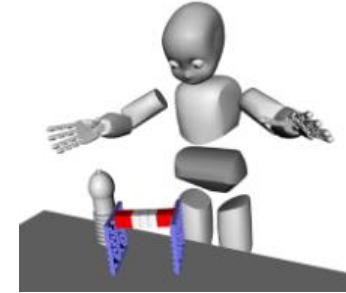
Representation of Reachability (2)

■ Generation

- Offline process in simulation
- Check all joint angles
 - in x steps (e.g. $x = 5^\circ$)
 - Determine the pose of the TCP using forward kinematics
 - Determine the grid cell and set the entry

■ Application

- Pre-calculated reachability information
- Quick decision whether a pose is reachable with the end effector.
Effort: $O(1)$
- Can be used for grasp selection



Grasps that cannot be reached can be efficiently sorted out.

Contents

Kinematic Model

- Kinematic Chain

- Denavit-Hartenberg Convention

- Direct Kinematic Problem

- Examples

- Jacobian Matrix

- Singularities and Manipulability

- Representation of Reachability

Geometric Model

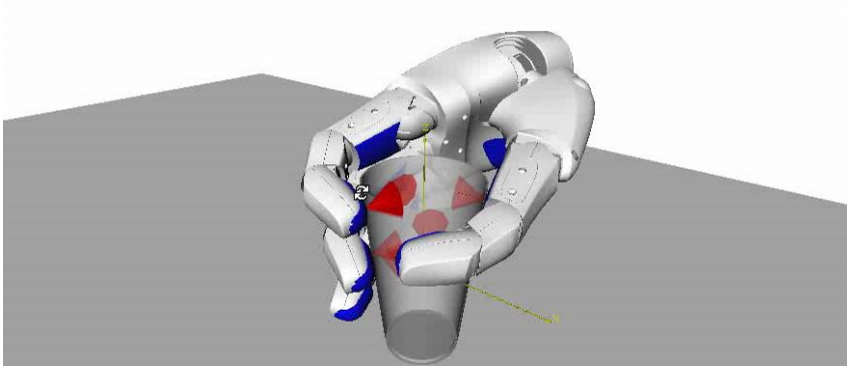
- Areas of Application

- Classification

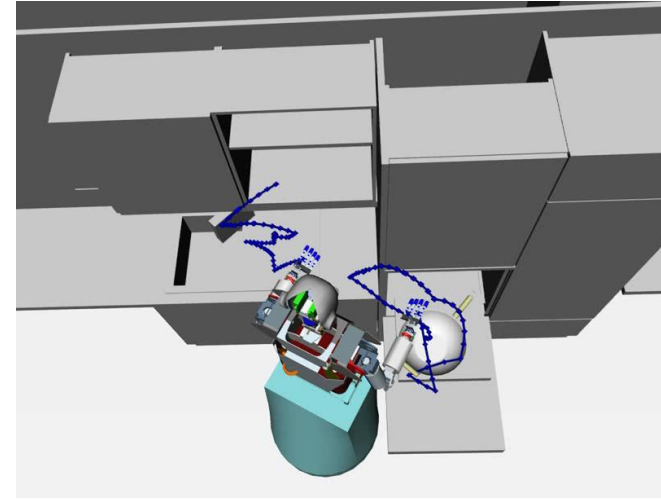
- Examples

Geometric Model: Motivation (1)

■ Collision and contact calculation

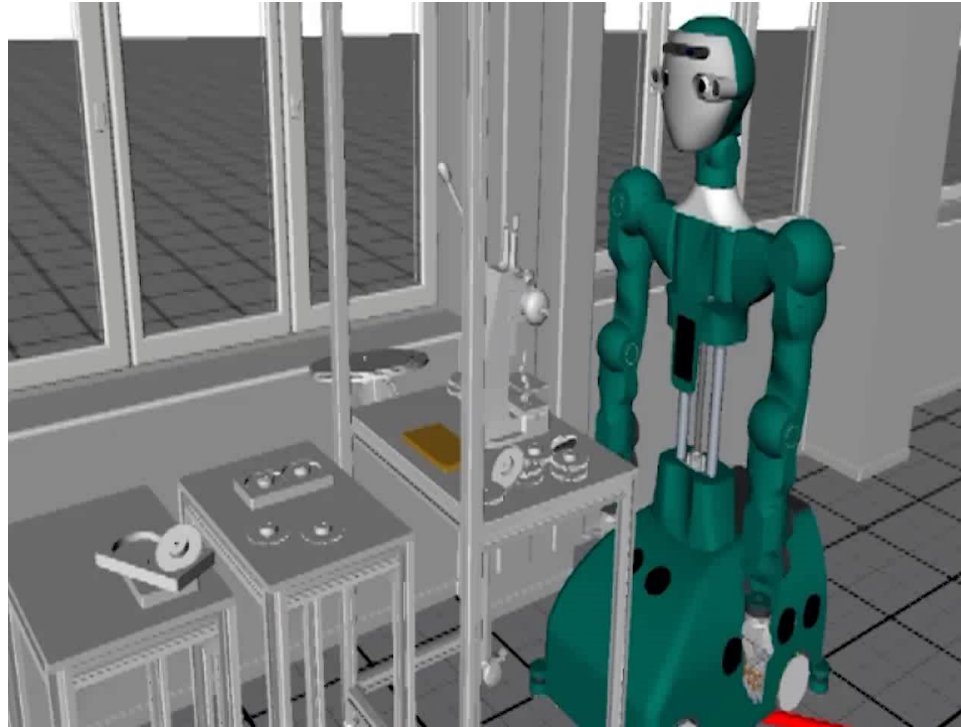


Grasping



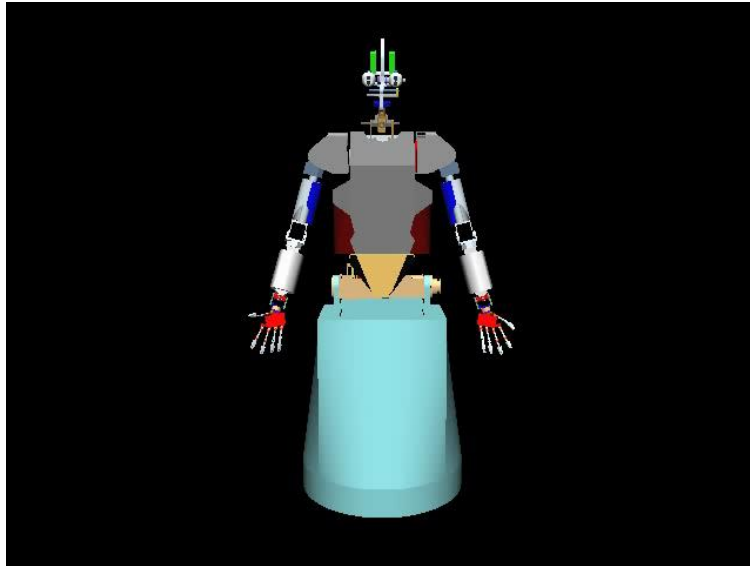
Motion planning

Geometric Model: Motivation (2)

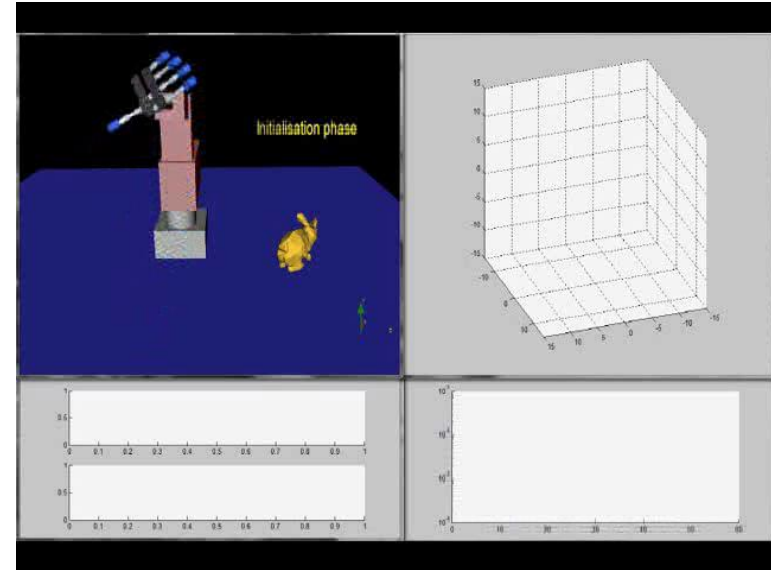


Geometric Model: Motivation (3)

■ Simulation



Imitation

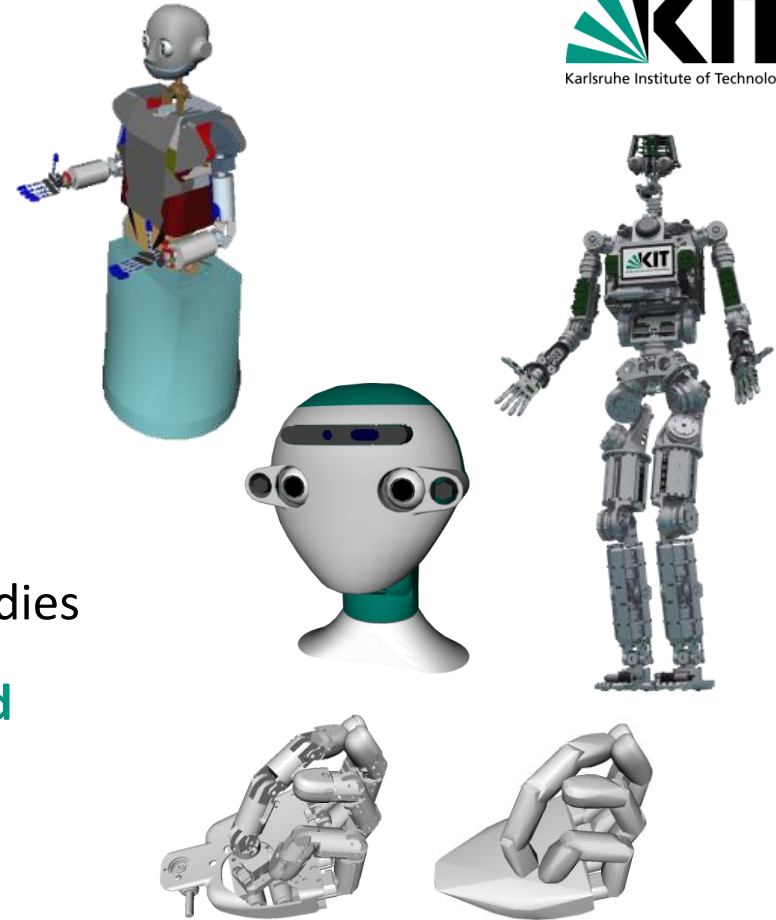


Haptic exploration

Geometric Model

Application

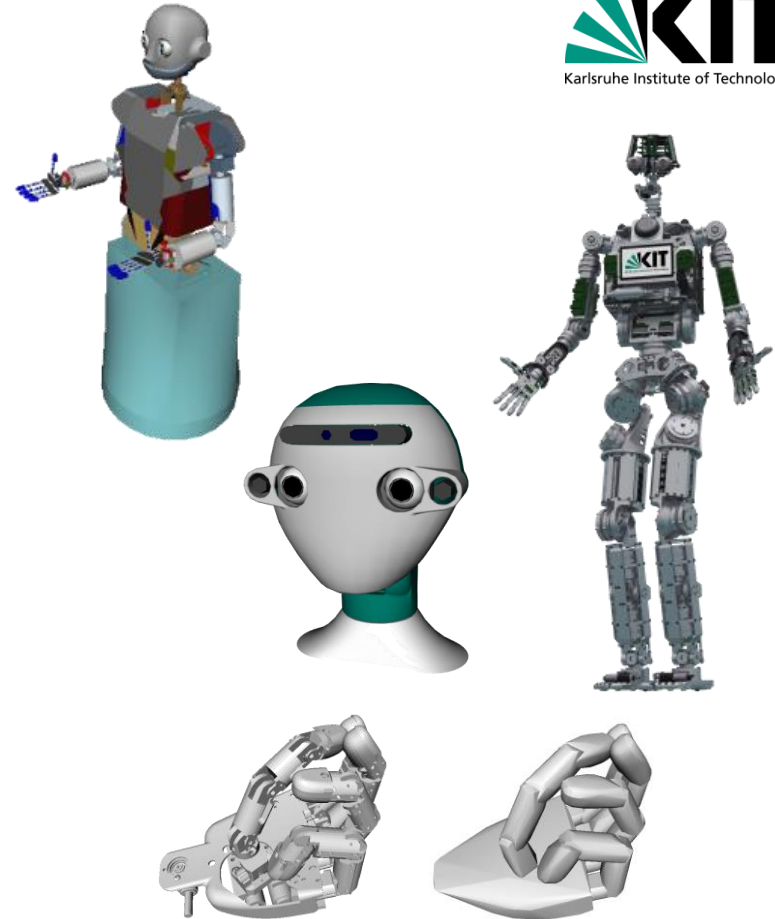
- Graphical **representation** of bodies (visualization)
- Starting point for **distance measurements** and **collision detection**
- Basis for calculating the **movements** of bodies
- Basis for determining the acting **forces and torques**



Geometric Model: Classification

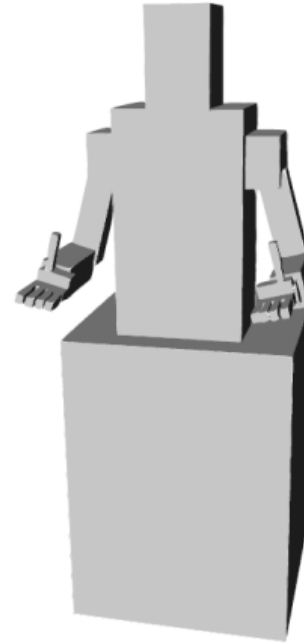
- Classification according to **spaces**
 - 2D models
 - 3D models
- Classification according to **basic primitives**
 - Edge or wireframe models
 - Surface models
 - Volume models

基本元素



Block World

- The bodies are represented by **bounding boxes**.
- Used in the first steps of collision avoidance.
- **Class:** 3D, volumes or surfaces



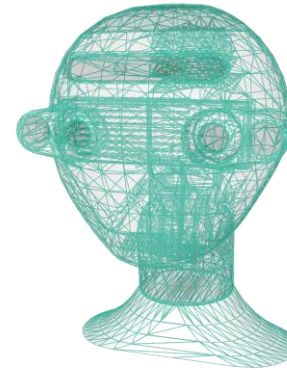
ARMAR-III block world model

Edge Model

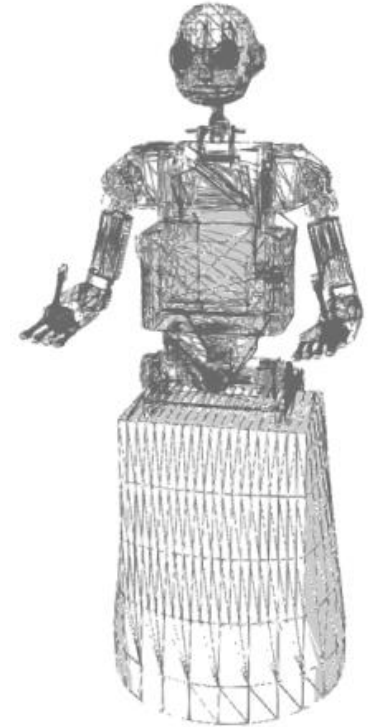
The bodies are represented by **polygons** (edges).

Used for quick visualization.

Class: 3D, edges or surfaces



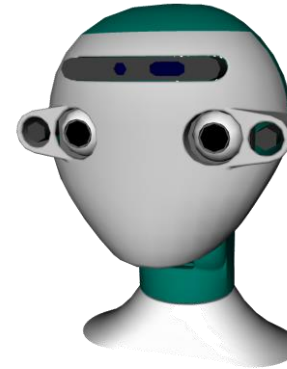
ARMAR-6 head model



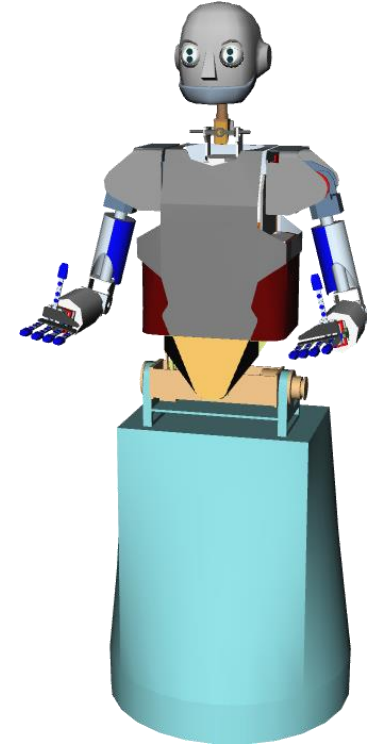
ARMAR-III edge model

Volume Model

- The bodies are represented **accurately**.
- Precise collision detection possible.
- Used for displaying animations.
- **Class:** 3D, volume



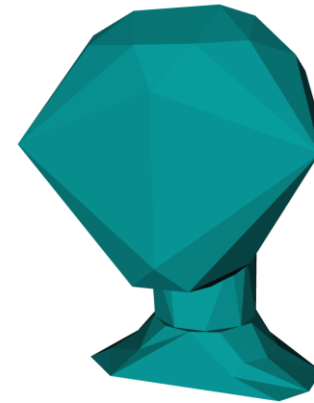
ARMAR-6 head model



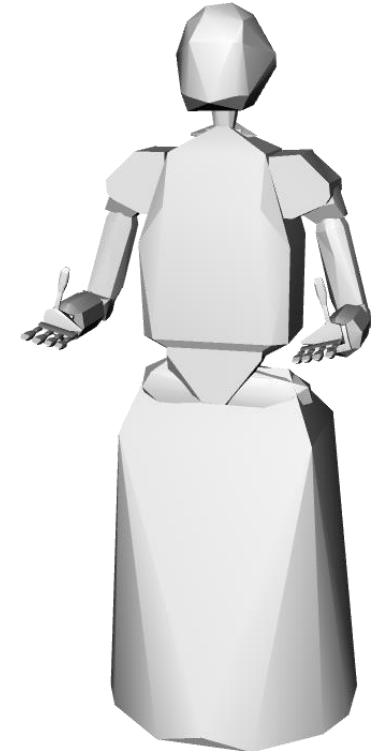
ARMAR-III volume model

Collision Model

- The bodies are represented in **simplified** form.
- Fast collision detection possible
- **Class:** 3D, volume



ARMAR-6 collision model



ARMAR-III collision model

Summary

Kinematic model

Denavit-Hartenberg convention: Minimum number of parameters to describe transformation between consecutive joints

Direct kinematics problem: Calculate end-effector pose from joint angles

Jacobian matrix: The solution for everything 😊

Singularities and manipulability

Reachability

Geometric model

Classification according to space (2D/3D) and basic primitives (edge or wireframe models, surface models and volume models)