Industrial Internet of Things

Robotik II

Lehrstuhl für Wirtschaftsinformatik
Prozesse und Systeme
Universität Potsdam

Chair of Business Informatics
Processes and Systems
University of Potsdam

Univ.-Prof. Dr.–Ing. habil. Norbert Gronau
Lehrstuhlinhaber | Chairholder

August-Bebel-Str. 89 | 14482 Potsdam | Germany

Tel +49 331 977 3322
Fax +49 331 977 3406

E-Mail ngronau@lswi.de
Web lswi.de

Grundlegendes Modell

Vorwärtskinematik

Modellierungskonventionen

Lernziele

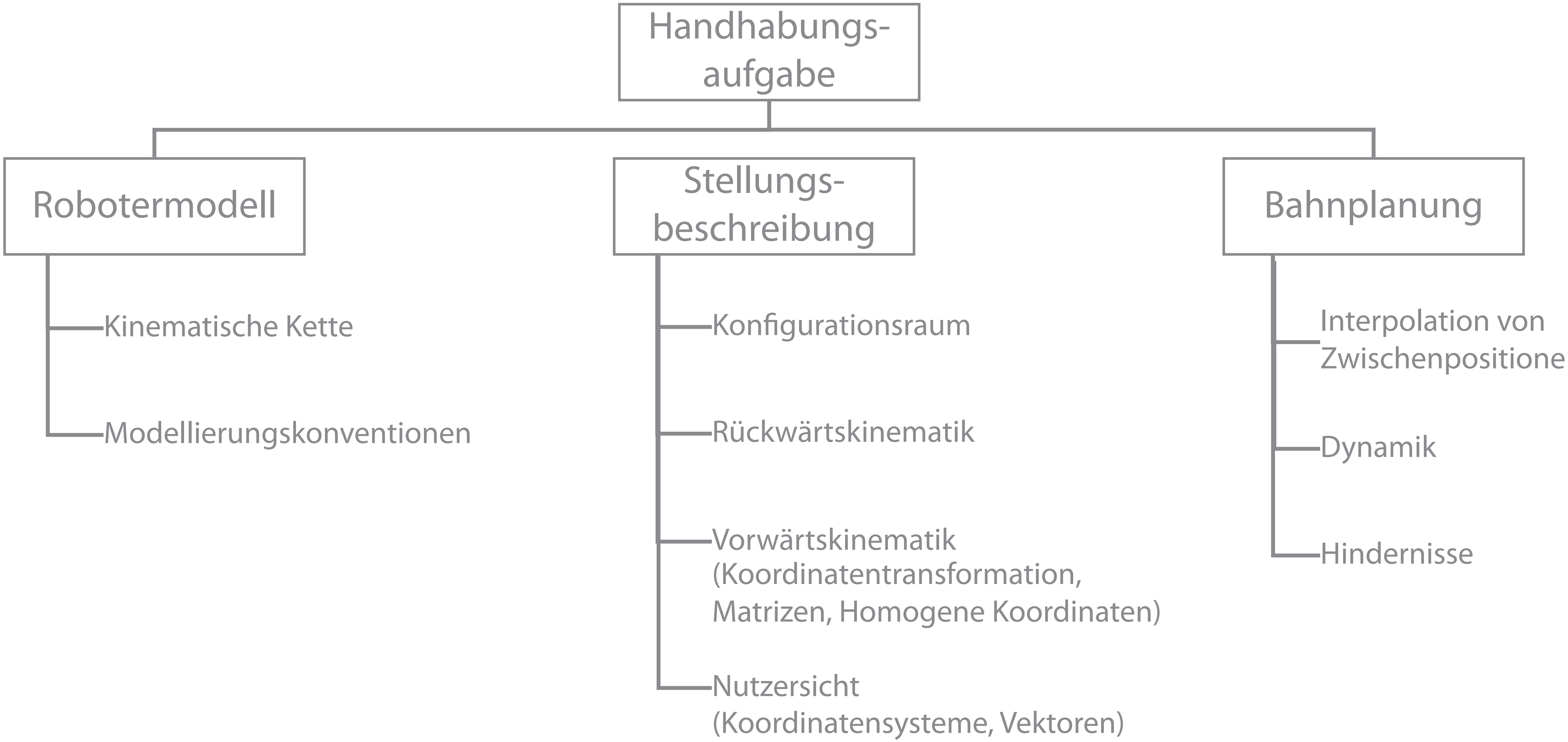
- Grundlegenden Modelle zur Beschreibung und Berechnung eines Roboterarms kennen
- Festlegung der Zwischensysteme verstehen
- Verstehen warum Rotation und Translation in einer Matrix zusammengefasst werden
- Geeignetes Vorgehen zur Berechnung der Stellung aus Nutzersicht kennen

Grundlegendes Modell

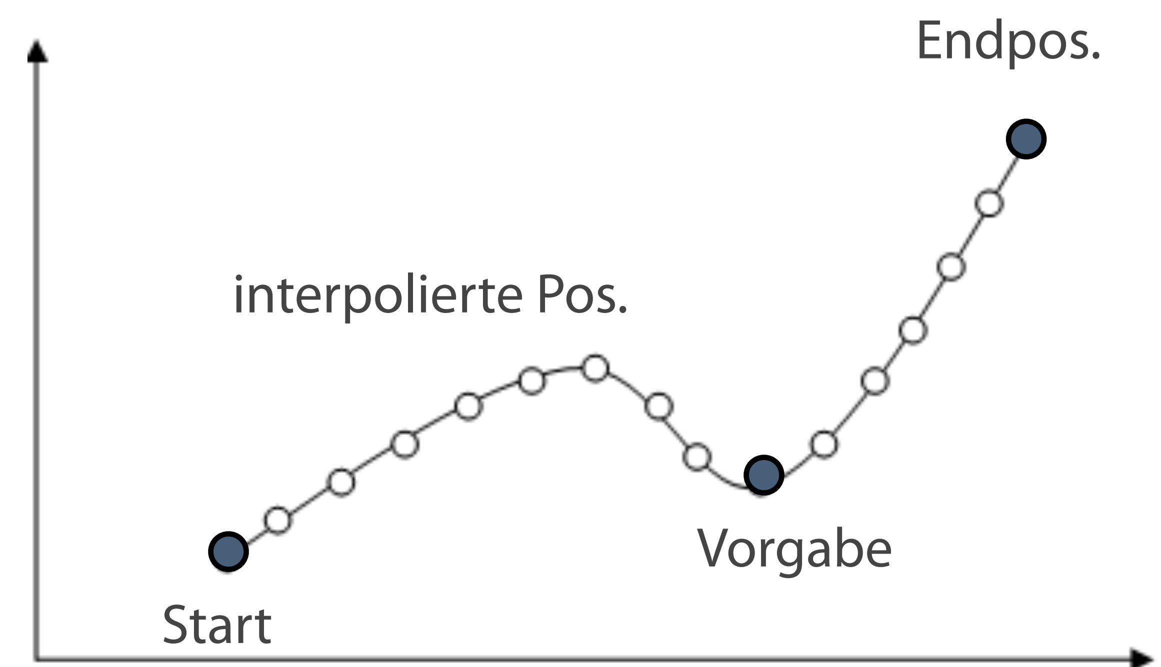
Vorwärtskinematik

Modellierungskonventionen

Strukturierung des Gebietes „Robotik“



Steuerung



Bewegungssteuerung

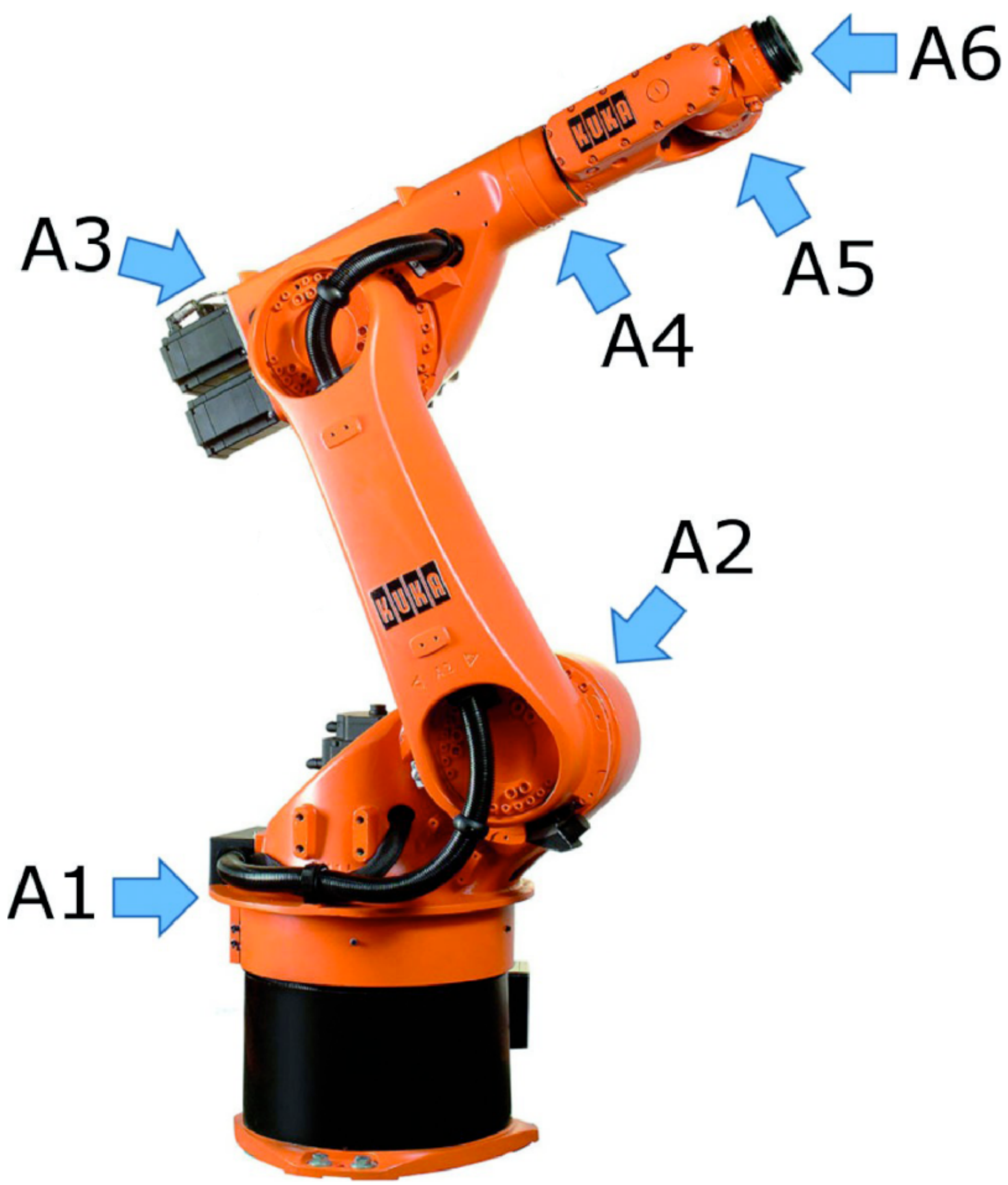
- Anfahren von **Stellungen**
- Koordination und Ausführung der Gelenkbewegung
- simultane Ansteuerung der Gelenke und gelenksynchrone Ausführung

Ablaufsteuerung

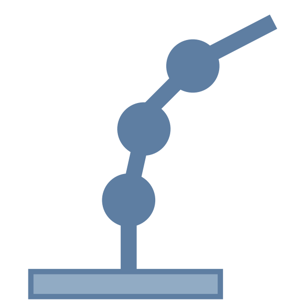
- Ablaufen bestimmter Bahnen
- Interpolation von Zwischenpositionen
- Verarbeitung von Hindernissen

Die Lösung dieser Aufgaben bedarf entsprechender Modelle des Roboters, welche durch das Zusammenwirken von mathematischem Erfassen und Betriebssoftware der Steuerung umgesetzt werden.

Kinematische Kette (KK)

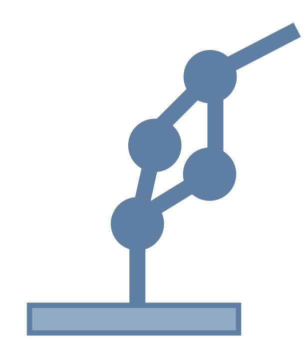


- Verknüpfung von Gelenkkörpern mittels Achsen (A)
- Menge zusammenhängender kinematischer Paare
- Rotatorische und translatorische Gelenke



Einfache KK

- serielle Kinematik
- paarweise Verknüpfung



Komplexe KK

- parallele Kinematik
- mehrfache Verknüpfung

Industrieroboter können als eine Kette von miteinander verbundenen Gelenken (Kinematische Kette) modelliert werden.

Stellung eines Endeffektors am Beispiel eines Roboters mit sechs Freiheitsgraden

- Stellung des Effektors = Position + Orientierung

Grundachsen

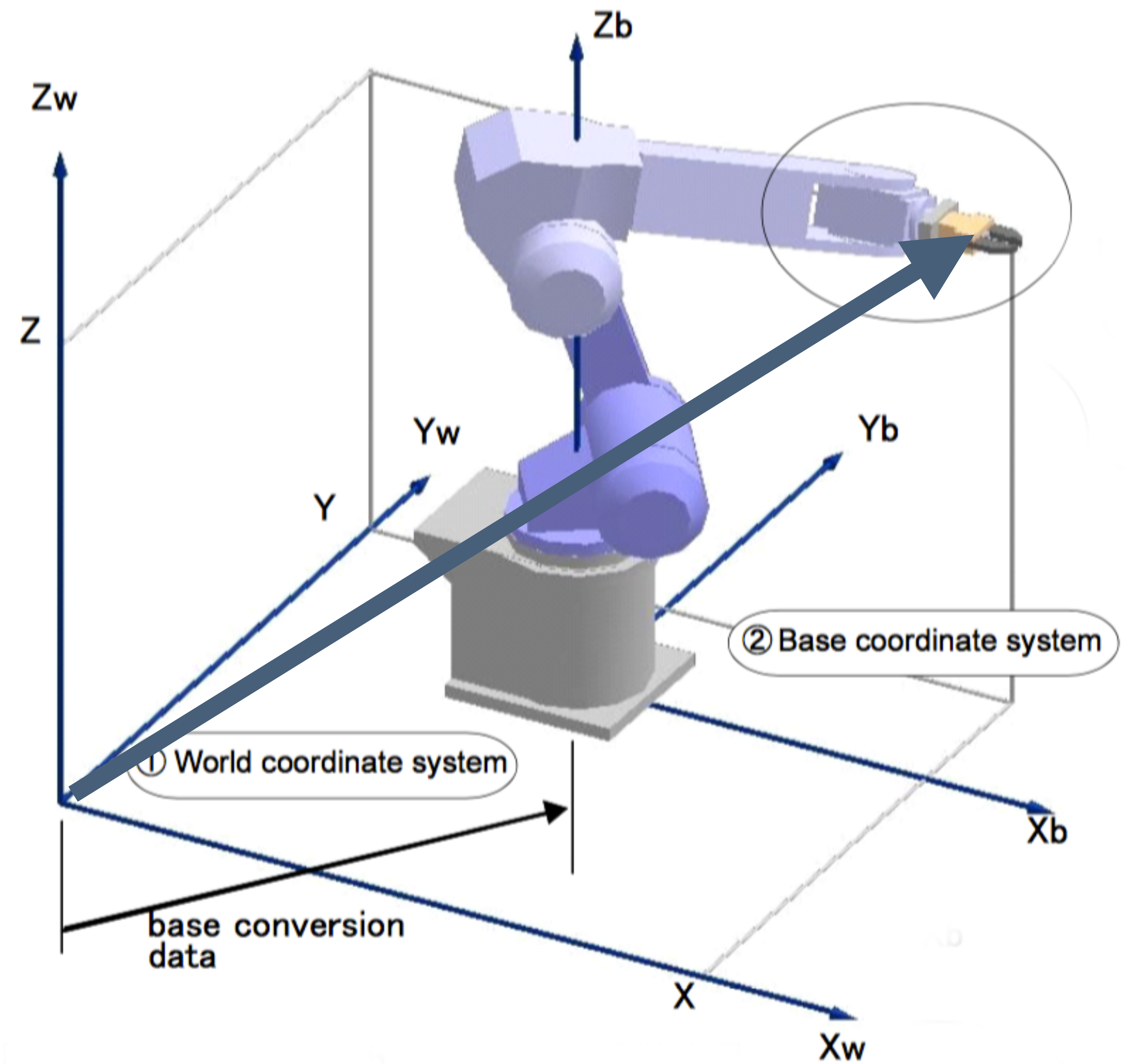
- Position des Effektors
- Rotationsachsen 1 bis 3

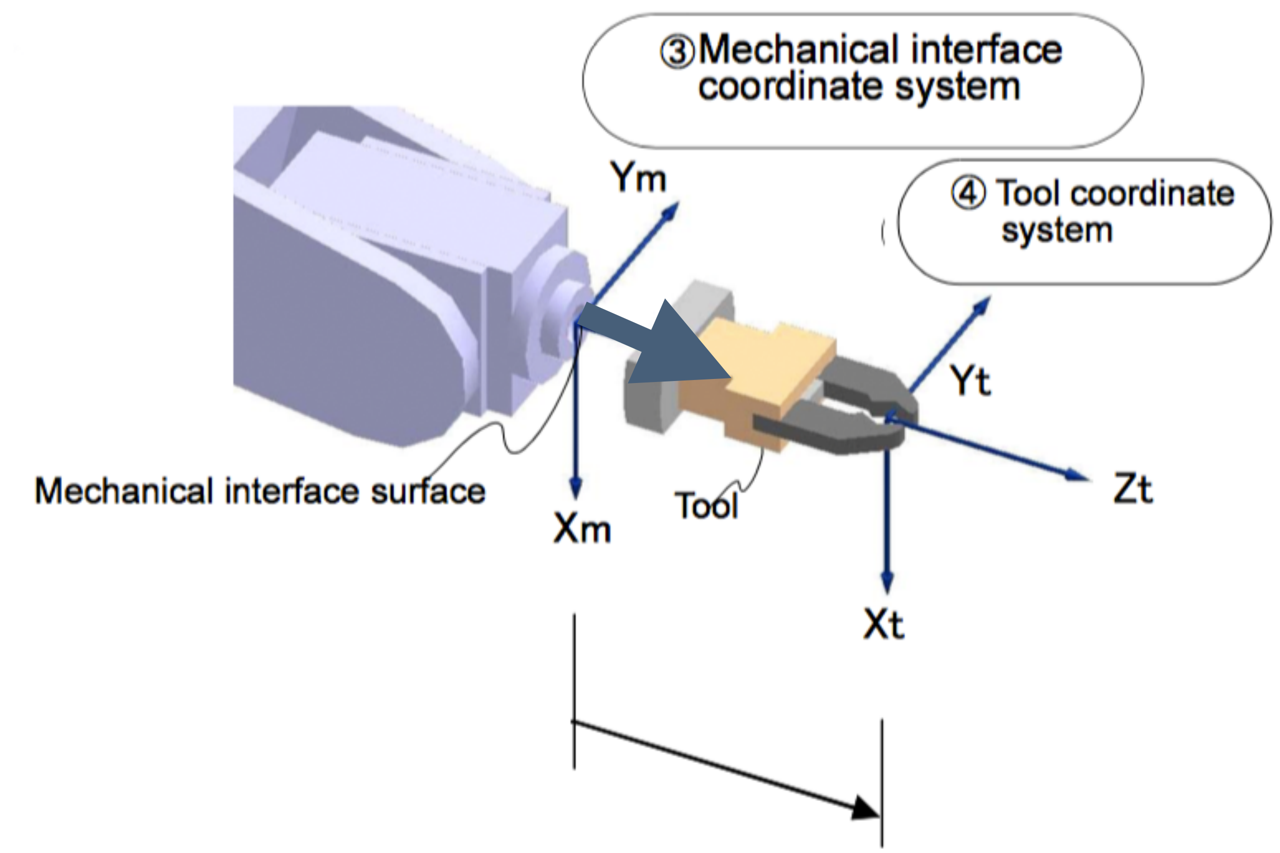
Handachsen

- Orientierung des Effektors
- Rotationsachsen 4 bis 6

Die freie Positionierung des Effektors bedarf sechs Freiheitsgrade also mindestens sechs Achsen.

Der Industrieroboter aus Nutzersicht
Stellungsbeschreibung



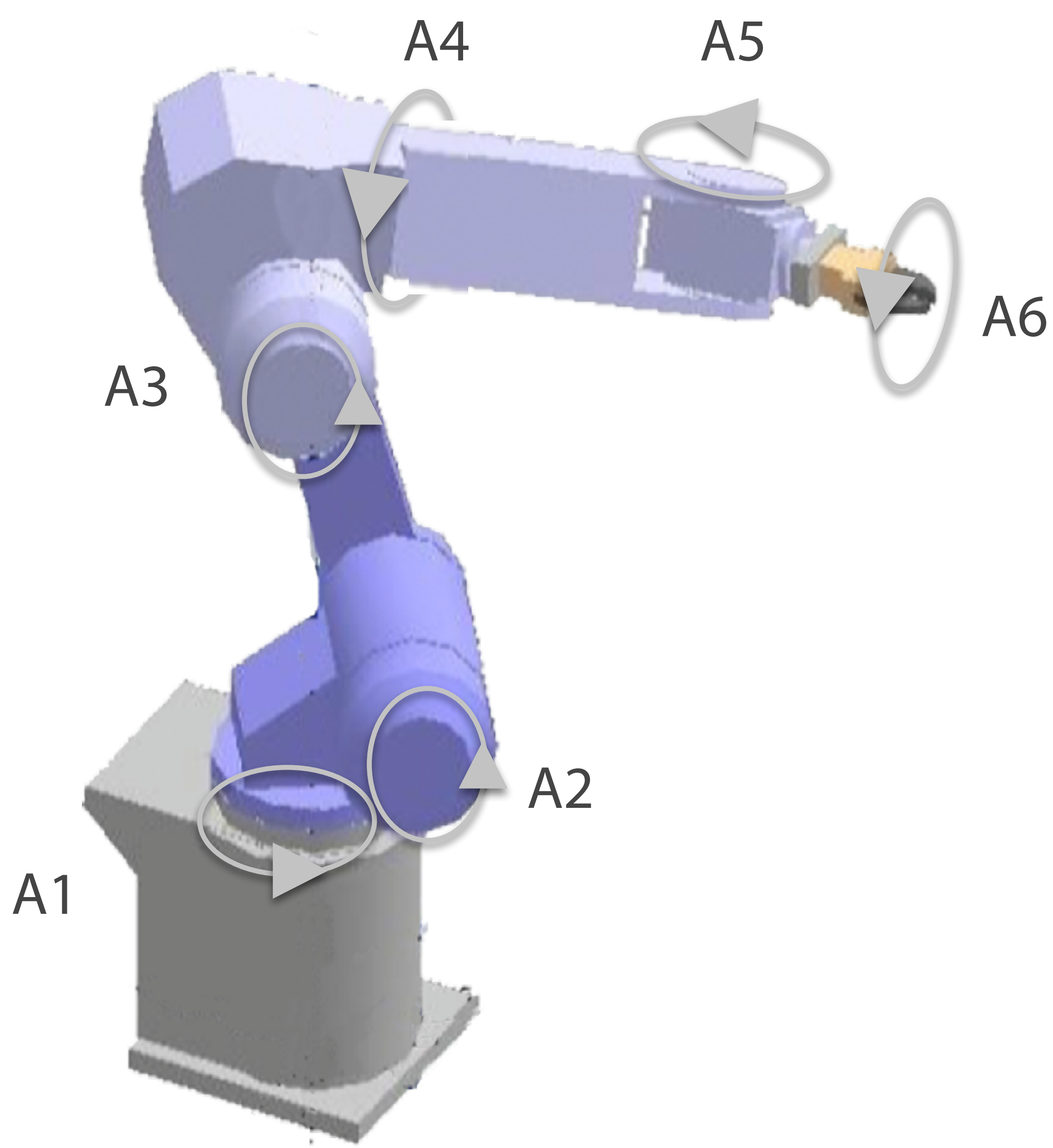


- Stellung = Position + Orientierung

Eine Stellung aus Nutzersicht läßt sich durch die sechs Parameter der Position und Orientierung des Effektors eindeutig beschreiben.

Quelle: Mitsubishi 2012

Der Industrieroboter aus Sicht der Steuerung
Stellungsbeschreibung

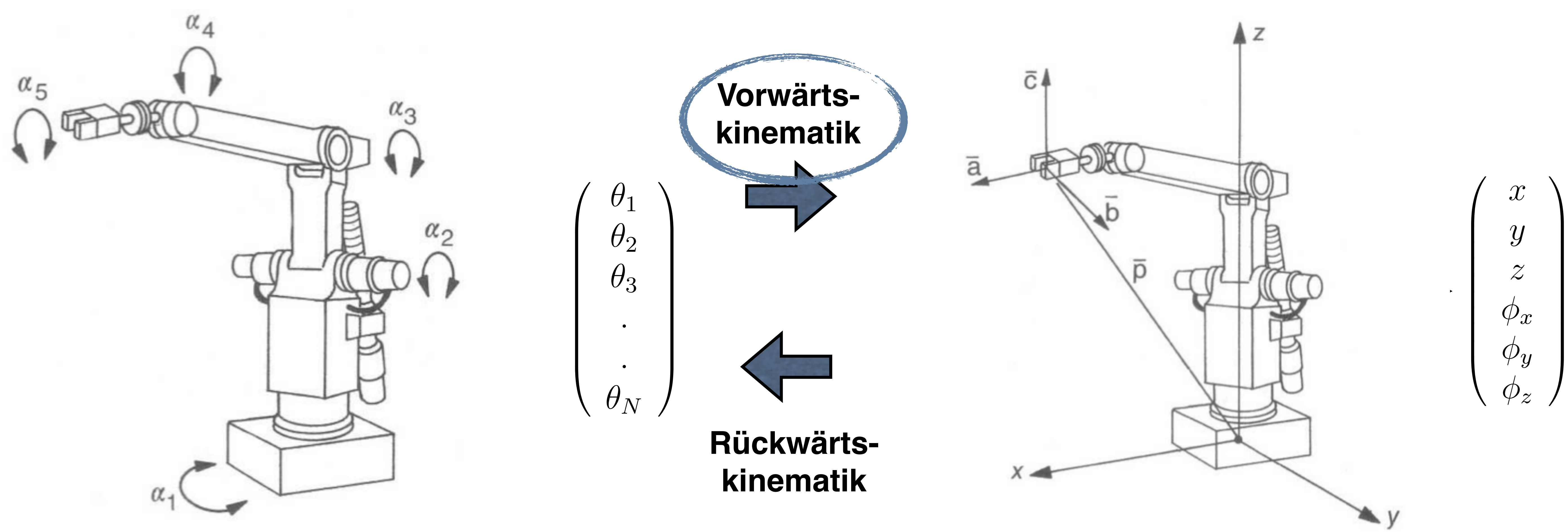


- A1: 30°
- A2: 20°
- A3: 110°
- A4: 10°
- A5: 0°
- A6: 80°

Eine Stellung aus Steuerungssicht läßt sich durch sechs Gelenkwinkel eindeutig beschreiben.

Quelle: Mitsubishi 2012

Stellungsbeschreibungen
Anwendungs- und technisch orientierte Betrachtungsperspektiven



Konfigurationsraum

- Vektor mit Gelenkstellungen
- Winkel oder Verschiebungen
- Relative Stellung eines kinematischen Paares
- Grundlage für die technische Steuerung

Nutzersicht

- **Vektor(en)** in einem **Koordinatensystem**
- Position und Orientierung des Effektors
- 3 kartesische Koordinaten
- 3 Winkel

Die Stellung eines Roboters kann auf unterschiedliche Weise beschrieben werden.
Durch geeignete Transformationen können die Stellungsdaten umgerechnet werden.

Grundlegendes Modell

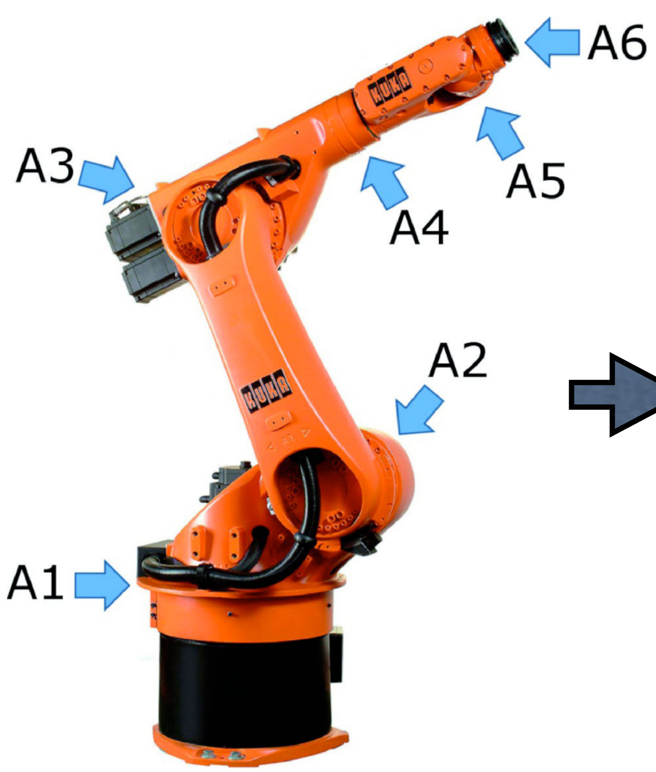
Vorwärtskinematik

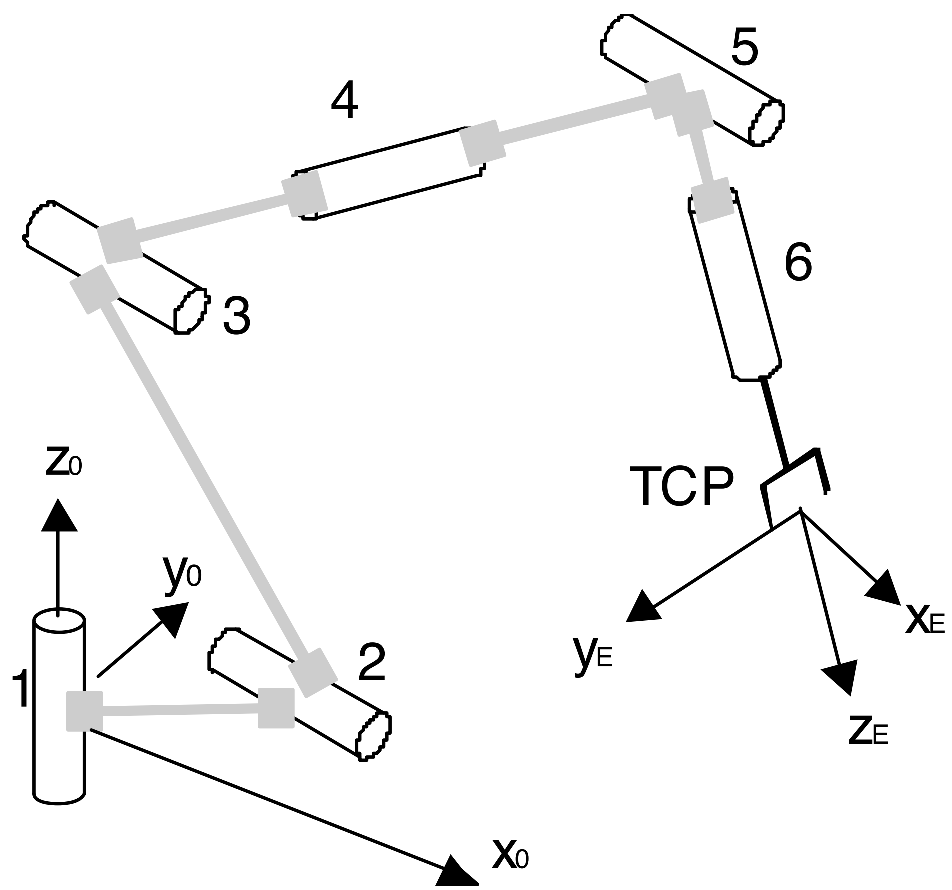
Modellierungskonventionen

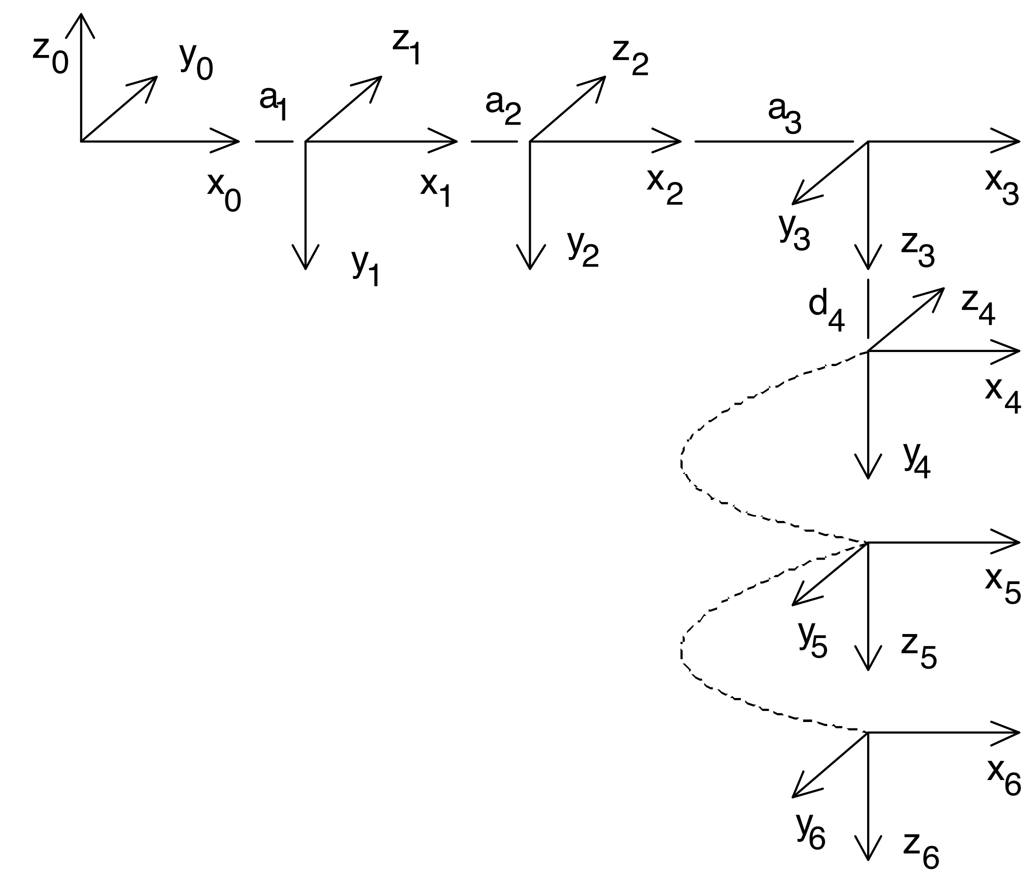
Lösungsidee Vorwärtskinematik
Schrittweise Transformationen

Gegeben $\begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ . \\ . \\ \theta_N \end{pmatrix}$ — $\begin{pmatrix} x \\ y \\ z \\ \phi_x \\ \phi_y \\ \phi_z \end{pmatrix}$ **Gesucht**

- Ein Koordinatensystem pro Achse
- Ermittlung der jeweiligen Übergänge von KS zu KS entlang der kinematischen Kette
- Zusammenfassung zu der Gesamttransformation

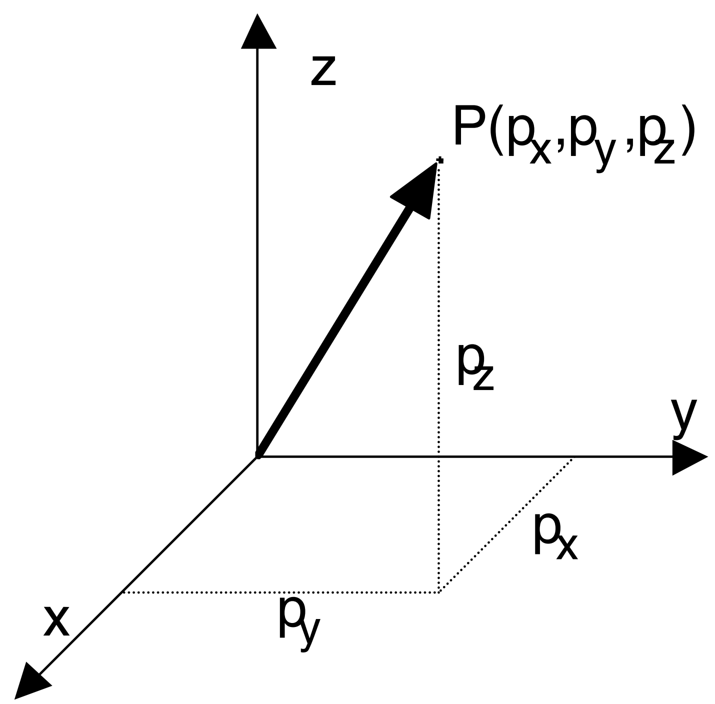


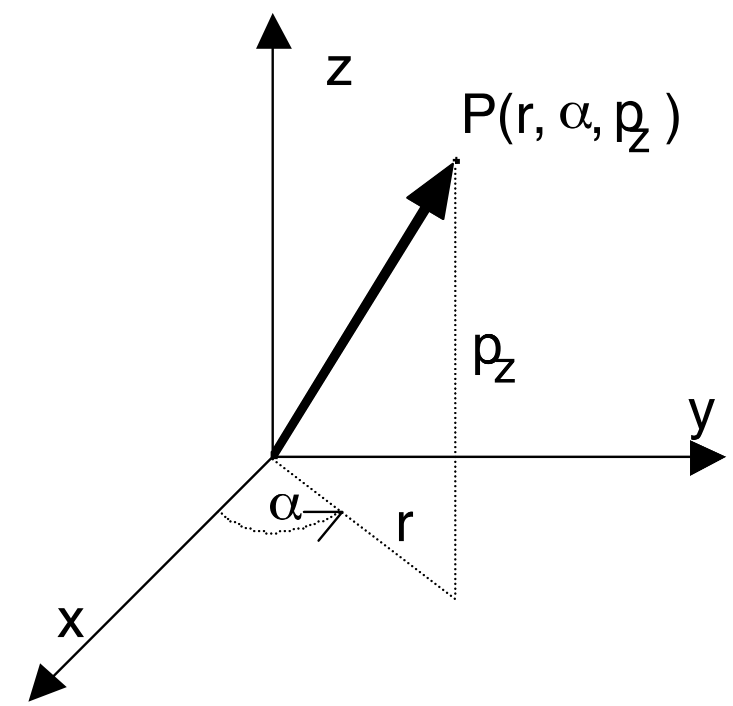


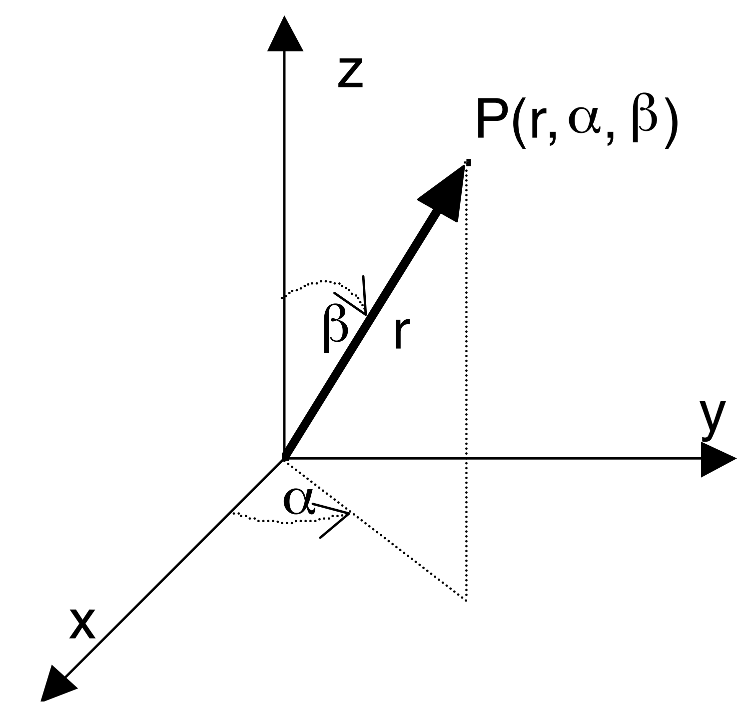


Ausgehend von der Geometrie des Roboters erfolgt die Formalisierung in ein System von Transformationen, die als Parameter die Drehwinkel beinhalten und die Stellung des Effekts im BKS errechnen.

Werkzeugkasten
Koordinatensysteme







Kartesische Koordinaten

- Länge in x, y und z-Richtung
- orthogonale Achsen
- Linearkombination der Einheitsvektoren

Zylinderkoordinaten

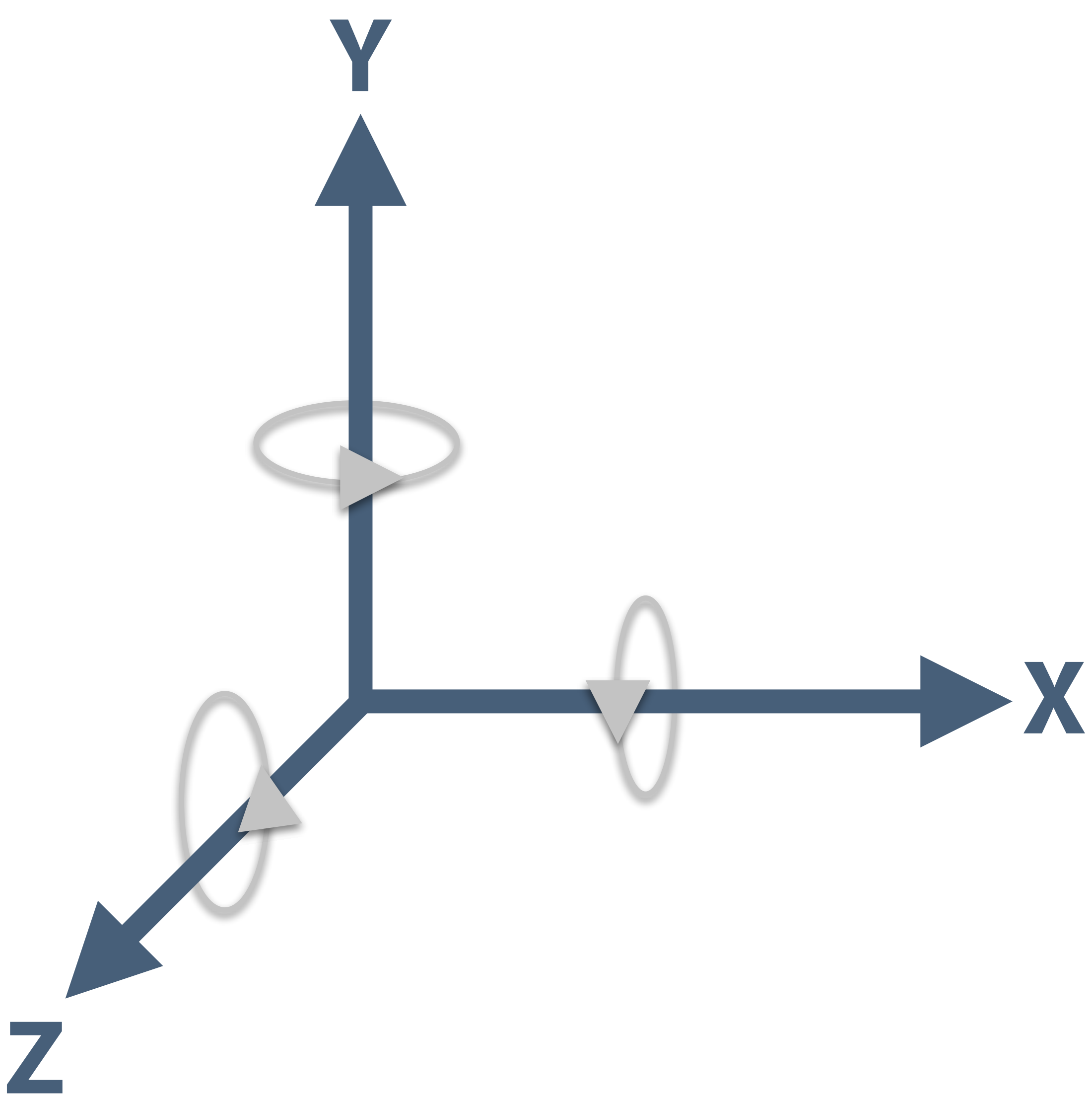
- Rotation des Radius r um z-Achse um Winkel
- Höhe in z-Richtung

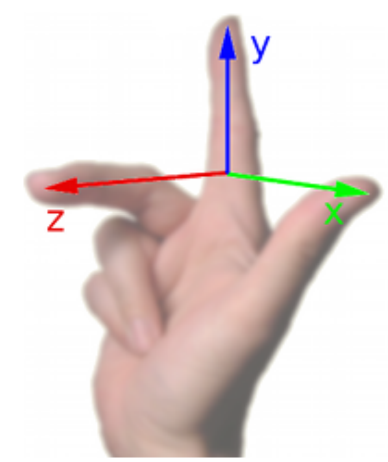
Kugelkoordinaten

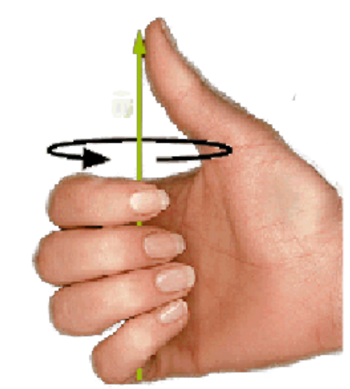
- Rotation des Radius r um z-Achse um Winkel
- Rotation in der z-r Ebene

Die Auswahl des Koordinatensystems kann nach Anwendungsfall erfolgen, eine Umrechnung von Koordinaten von einem System zum anderen ist ohne weiteres möglich.

Werkzeugkasten
Kartesisches Koordinatensystem (rechtshändig)







Rechte-Hand-Regel

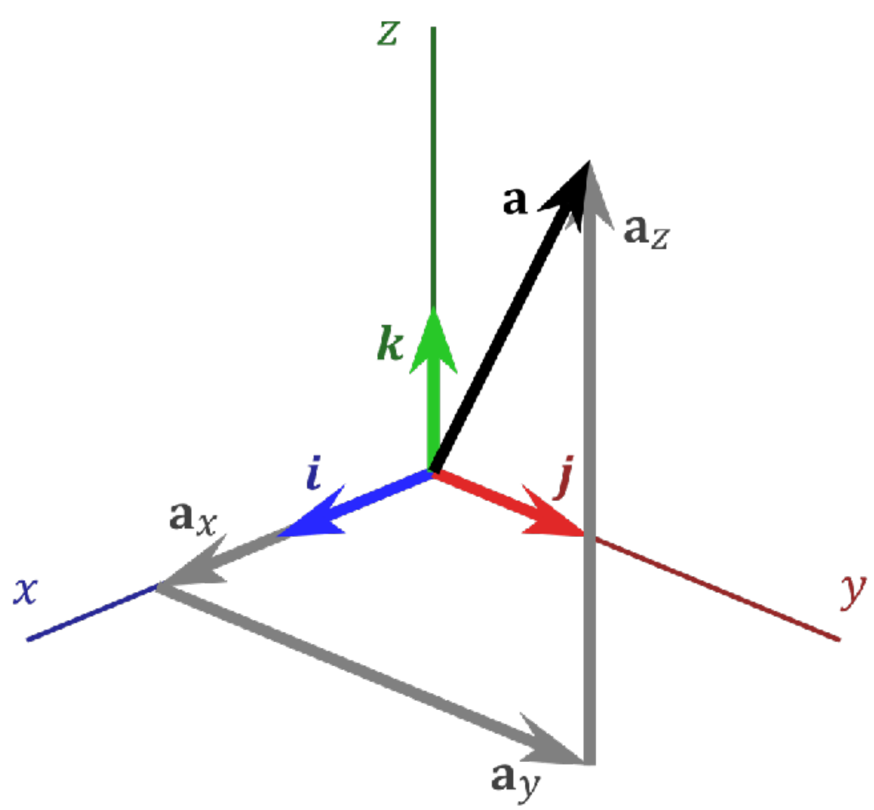
- X = Daumen
- Y = Zeigefinger
- Z = Mittelfinger

Drehrichtung

- Daumen in Richtung der Achse
- Richtung durch umschließende Finger

Rechtssysteme sind in der Robotik üblich und Grundlage der weiteren Betrachtungen.

Werkzeugkasten
Vektoren



$$\vec{a} + \vec{b} = \begin{pmatrix} x_a \\ y_a \\ z_a \end{pmatrix} + \begin{pmatrix} x_b \\ y_b \\ z_b \end{pmatrix} = \begin{pmatrix} x_a + x_b \\ y_a + y_b \\ z_a + z_b \end{pmatrix}$$

$$\lambda \cdot \vec{v} = \lambda \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \lambda \cdot x \\ \lambda \cdot y \\ \lambda \cdot z \end{pmatrix}$$

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

Kartesische Vektoren

- Einheitsvektoren **i, j, k**
- Vektor **a** als Linearkombination der Einheitsvektoren
- **a** = $(a_x, a_y, a_z)^T = a_x * \mathbf{i} + a_y * \mathbf{j} + a_z * \mathbf{k}$

Rechenoperationen

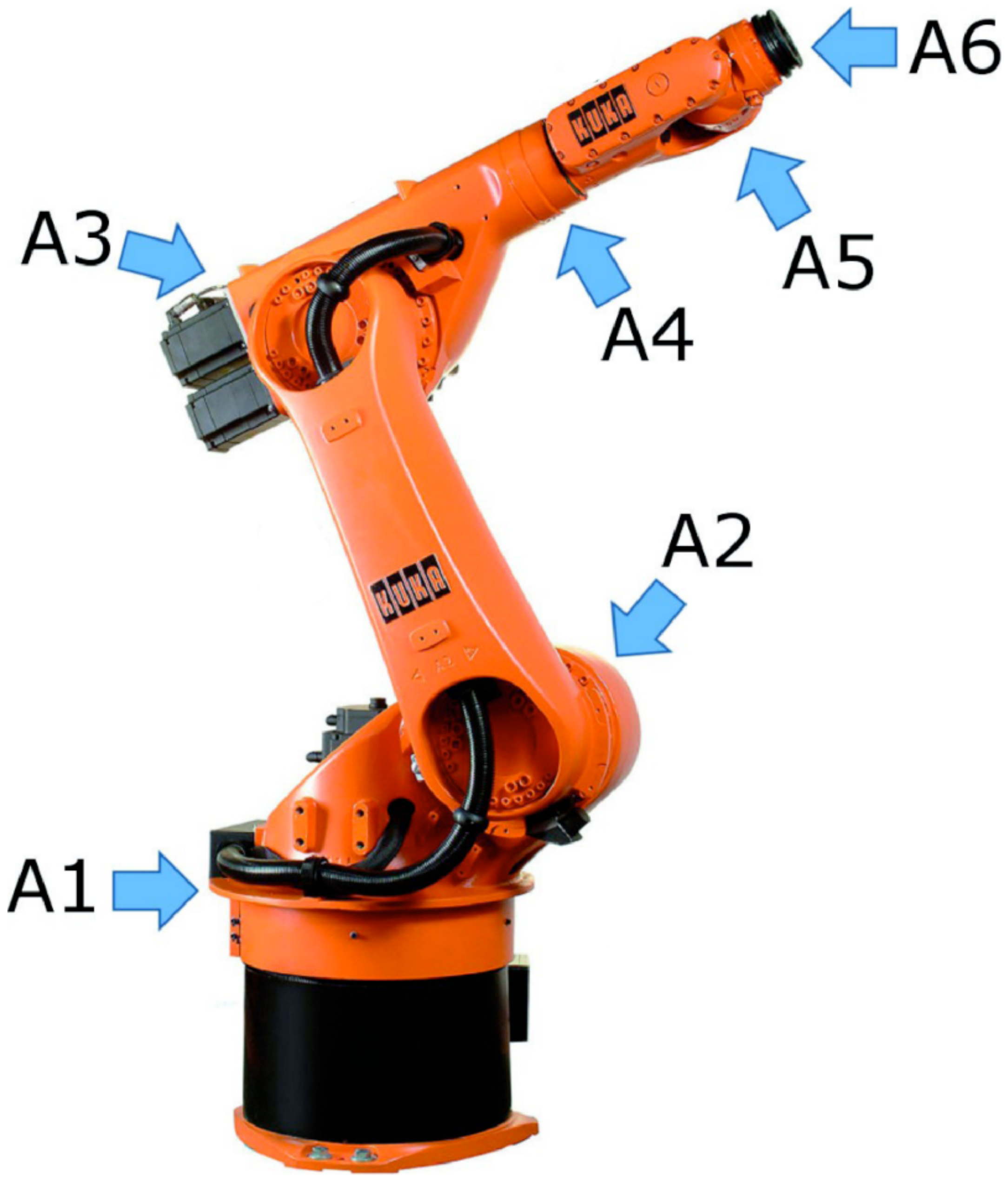
- Vektoraddition
- Skalare Multiplikation
- Kreuzprodukt

$$\vec{v} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = (x, y, z)^T$$

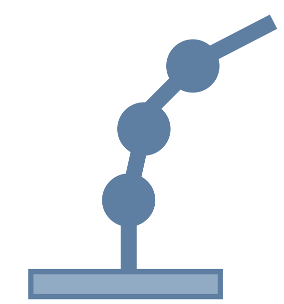
Vektoren dienen der Beschreibung von Objekten und Objektlagen im 3-dim. euklidischen Raum, in der Robotik zur Beschreibung von Positionen und Orientierungen.

Quelle: Bronstein 2012

Zurück zur kinematische Kette

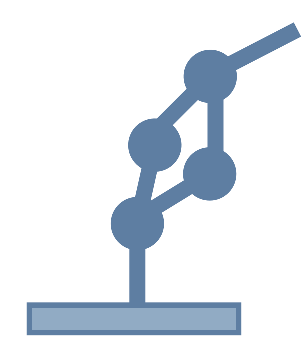


- Verknüpfung von Gelenkkörpern mittels Achse (A)
- Menge zusammenhängender kinematischer Paare
- **Rotatorische und translatorische Gelenke**



Einfache KK

- serielle Kinematik
- paarweise Verknüpfung

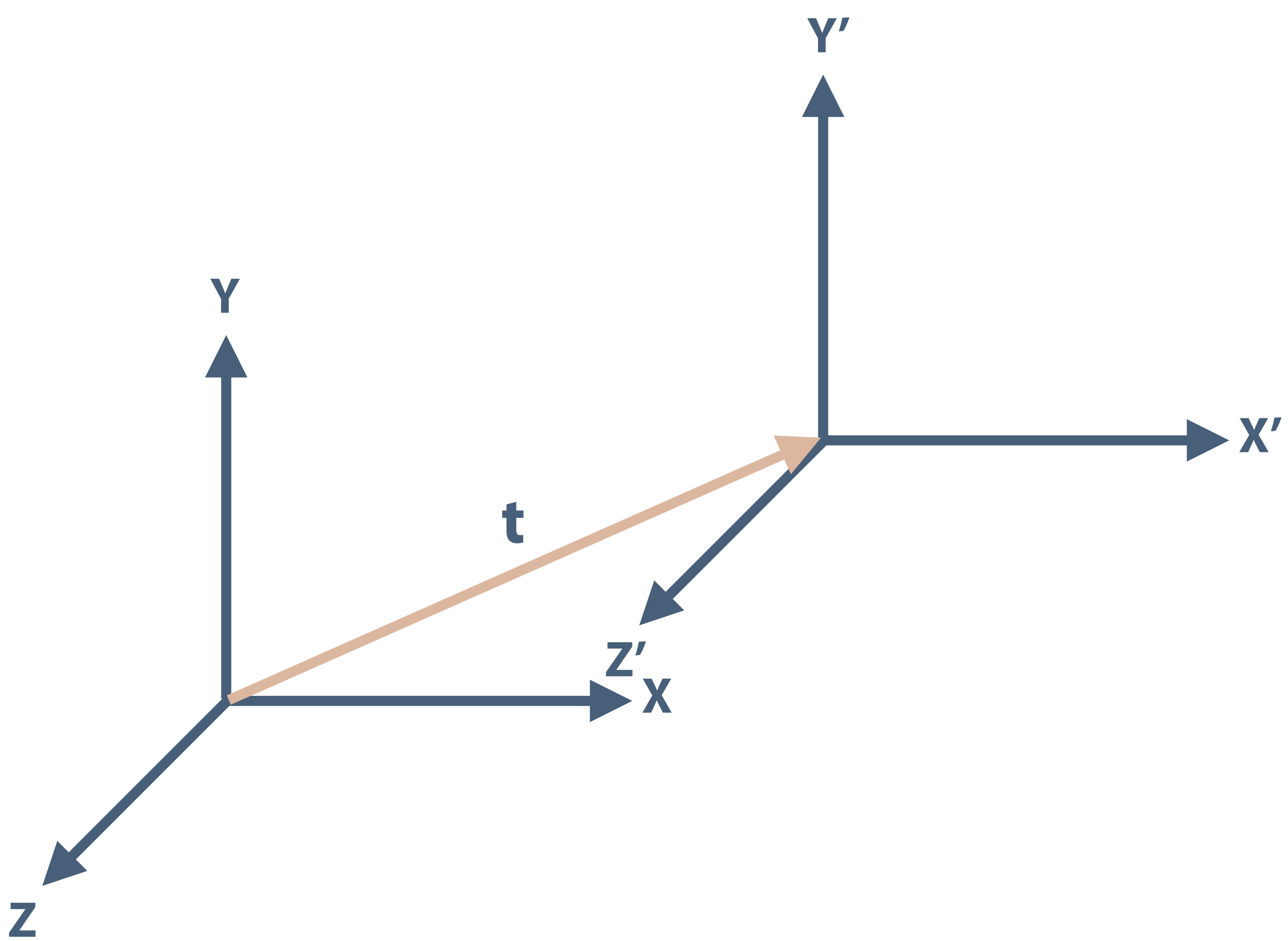


Komplexe KK

- parallele Kinematik
- mehrfache Verknüpfung

Industrieroboter können als eine Kette von miteinander verbundenen Gelenken modelliert werden.

Koordinatentransformation
Beispiel Translation um Vektor t



Linearkombination von Koordinaten und Einheitsvektoren

- $\mathbf{a} = (a_x, a_y, a_z)^T = a_x * \mathbf{i} + a_y * \mathbf{j} + a_z * \mathbf{k}$
- $\mathbf{i} = (1, 0, 0)^T$
- $\mathbf{j} = (0, 1, 0)^T$
- $\mathbf{k} = (0, 0, 1)^T$

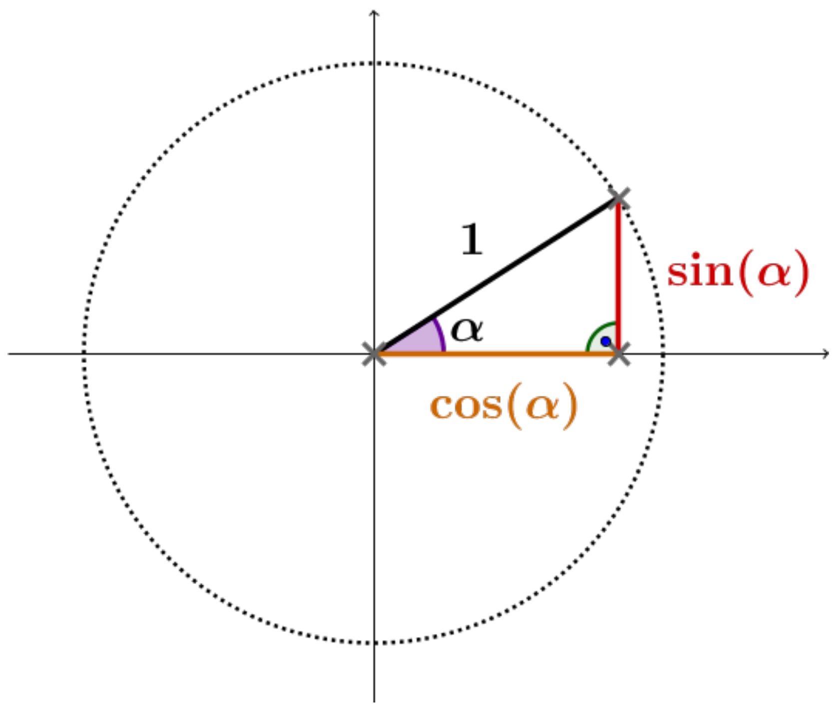
Translation um dx, dy, dz

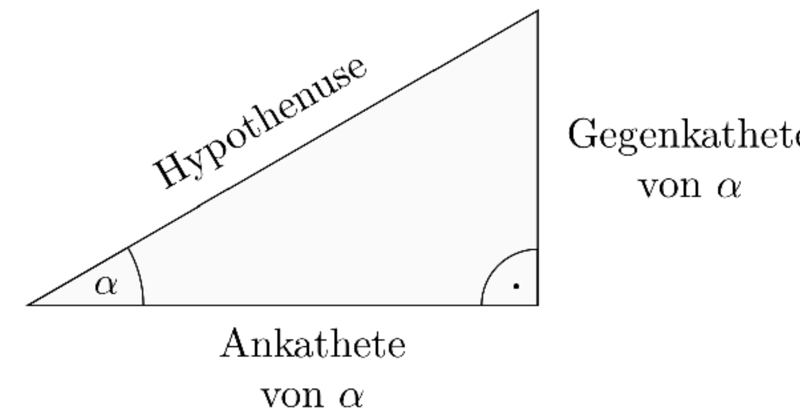
- $\mathbf{t} = (d_x, d_y, d_z)^T$
- Vektoraddition $\mathbf{a} + \mathbf{t}$

Anwendung auf Vektor a Translationsvektor t

- $\mathbf{a'} = (a'_x, a'_y, a'_z)^T = \mathbf{a} + \mathbf{t} = a_x * \mathbf{i} + a_y * \mathbf{j} + a_z * \mathbf{k} + d_x * \mathbf{i} + d_y * \mathbf{j} + d_z * \mathbf{k}$
- $\mathbf{a'} = (a_x + d_x) * \mathbf{i} + (a_y + d_y) * \mathbf{j} + (a_z + d_z) * \mathbf{k}$
- $\mathbf{a'} = (a_x + d_x, a_y + d_y, a_z + d_z)^T$

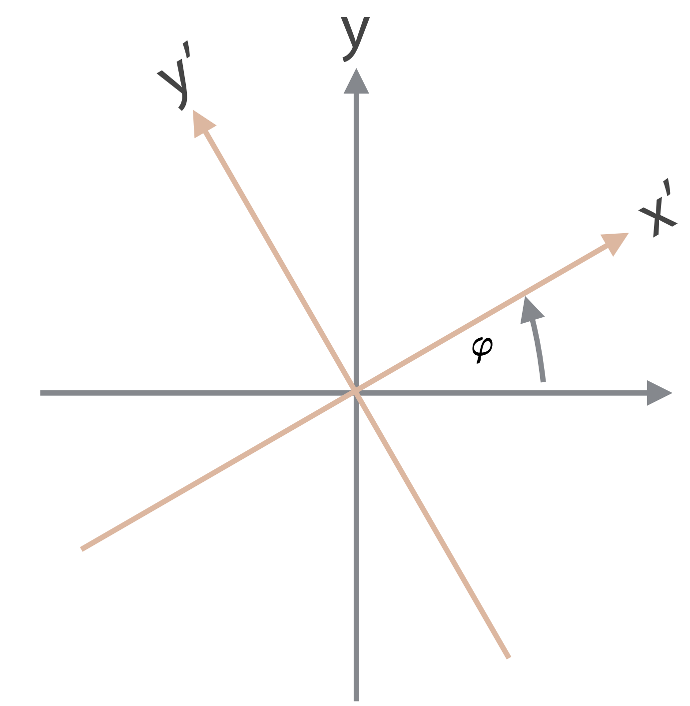
Werkzeugkasten
Winkelfunktionen Sinus und Cosinus





$$\sin\alpha = \frac{\text{Gegenkathete von }\alpha}{\text{Hypothenuse}}$$

$$\cos\alpha = \frac{\text{Ankathete von }\alpha}{\text{Hypothenuse}}$$



Sinus und Kosinus im Einheitskreis

- Länge von Ankathete (sin) und Gegenkathete (cos) in Abhängigkeit des Winkels
- Zuordnung von Längen zu einem Winkel
- Skalierung für Hypothenuse beliebiger Länge

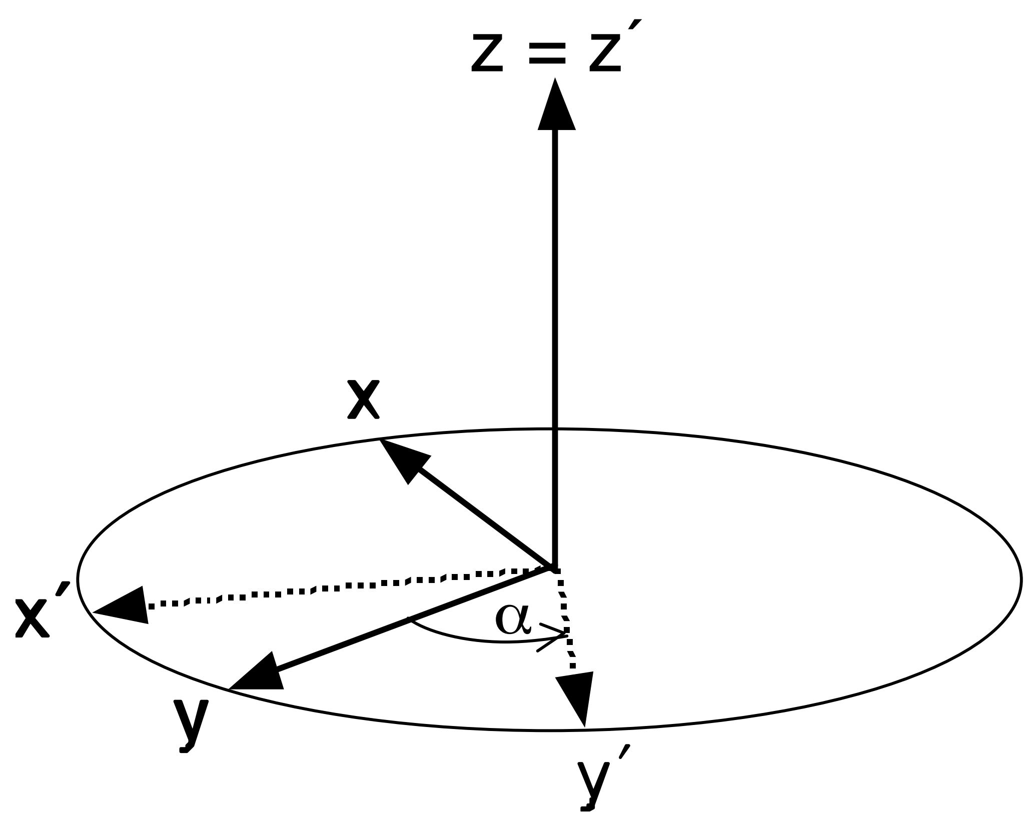
Anwendungsfall Rotation eines Koordinatensystems

- Drehung um die z-Achse
- Errechnen der Anteile von x' auf x und y
- Errechnen der Anteile von y' auf x und y

Sinus und Cosinus sind Basis der Berechnung von Rotationen.

Quelle: Bronstein 2012

Koordinatentransformation
Beispiel Rotation um die z-Achse

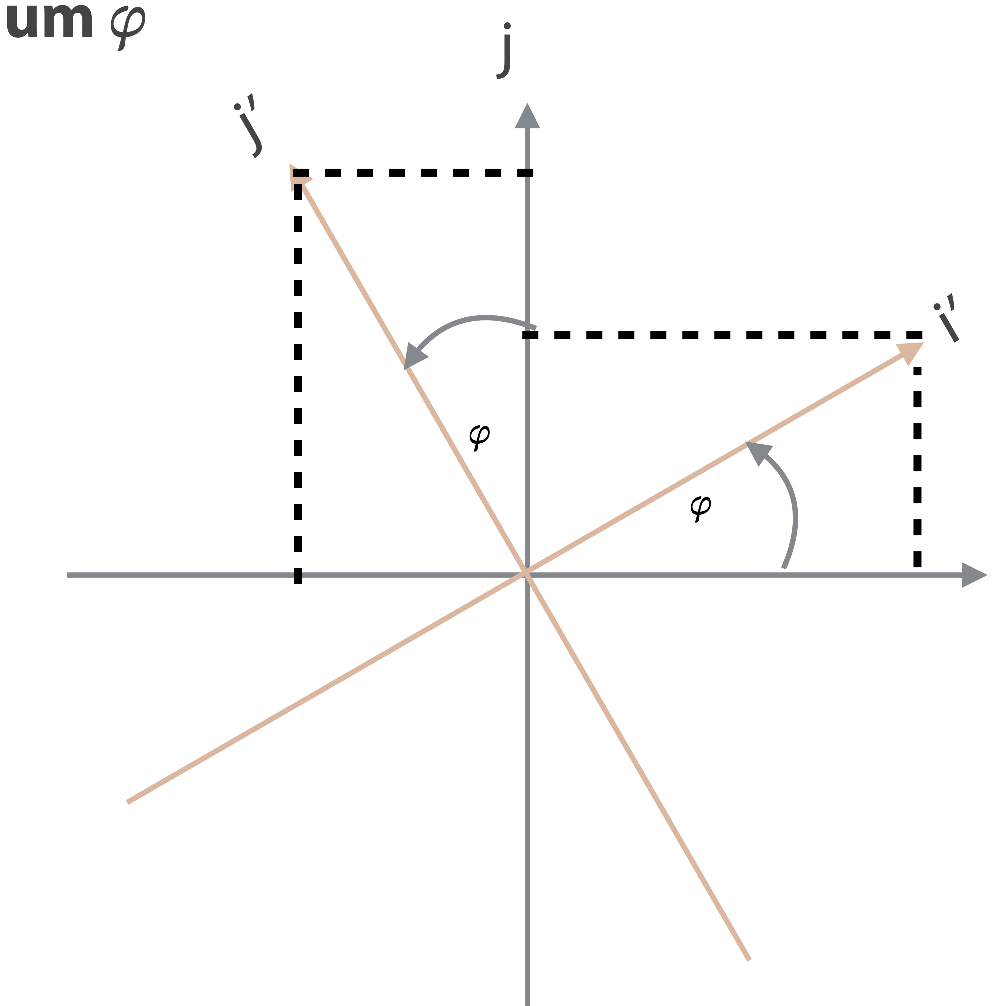


Linearkombination von Koordinaten und Einheitsvektoren

- $\mathbf{a} = (a_x, a_y, a_z)^T = a_x * \mathbf{i} + a_y * \mathbf{j} + a_z * \mathbf{k}$
- $\mathbf{i} = (1, 0, 0)^T$
- $\mathbf{j} = (0, 1, 0)^T$
- $\mathbf{k} = (0, 0, 1)^T$

Rotation der Einheitsvektoren um φ

- $\mathbf{i'} = (\cos\varphi, \sin\varphi, 0)^T$
- $\mathbf{j'} = (-\sin\varphi, \cos\varphi, 0)^T$
- $\mathbf{k}' = (0, 0, 1)^T = \mathbf{k}$



Anwendung auf Vektor a mit rot(a_x, a_y, a_z, φ)

- $\mathbf{a'} = (a'_x, a'_y, a'_z)^T = a_x * \mathbf{i'} + a_y * \mathbf{j'}, + a_z * \mathbf{k'}$
- $\mathbf{a'} = (a_x * (\cos\varphi, \sin\varphi, 0)^T, a_y * (-\sin\varphi, \cos\varphi, 0)^T, a_z * (0, 0, 1)^T)^T$
- $\mathbf{a'} = (a_x * \cos\varphi, a_x * \sin\varphi, 0)^T + (- a_y * \sin\varphi, a_y * \cos\varphi)^T$
- **$\mathbf{a'} = (a_x * \cos\varphi - a_y * \sin\varphi, a_x * \sin\varphi + a_y * \cos\varphi, 0)^T$**

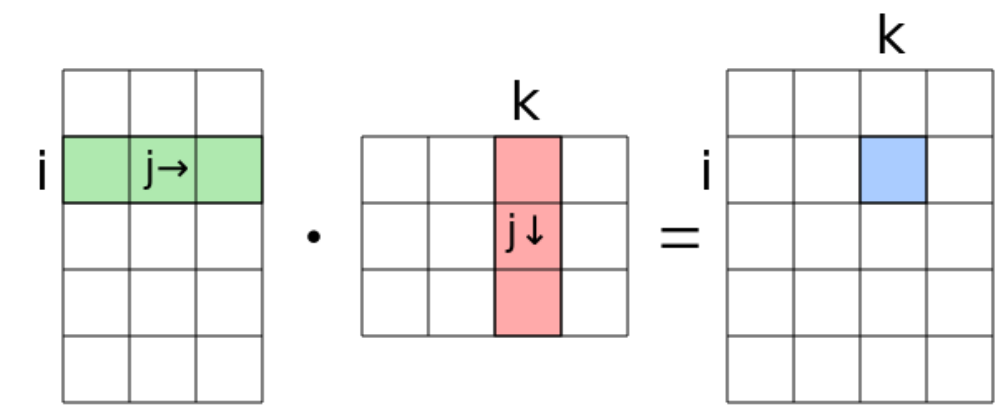
Werkzeugkasten
Matrizen

- $\mathbf{A} = (a_{ij})$ mit $\mathbf{A} \in \mathbf{R}^{m \times n}$; $i \in \{1, 2, ..., m\}$; $j \in \{1, 2, ..., n\}$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots \\ a_{21} & a_{22} & a_{23} & \cdots \\ a_{31} & a_{32} & a_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \; a_{mn} \end{bmatrix} \qquad A = \begin{bmatrix} -5 & 1 & -3 \\ 6 & 0 & 2 \\ 2 & 6 & 1 \end{bmatrix}$$

$$R^{l \times m} \times R^{m \times n} \to R^{l \times n}, \quad (A, B) \mapsto C = A \cdot B$$

$$c_{ik} = \sum_{j=1}^{m} a_{ij} \cdot b_{jk}$$



Matrizen

- Matrix A vom Typ (m,n) als System von m mal n Elementen
- Verallgemeinerung des Vektors

Matrizenmultiplikation

- komponentenweise Multiplikation und Summation der Einträge
- Zeilen von Matrix A mit Spalten der Matrix B
- nicht kommutativ!

Matrizen dienen u. a. zur Beschreibung und Kombination von Koordinatentransformationen.

Quelle: Bronstein 2012

Koordinatentransformation
Homogene Koordinaten

Beispiel mit zwei Dimensionen

$\mathbf{a} = (a_x, a_y, a_z)^T = a * \mathbf{E}$ $\quad E = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Linearkombination von Koordinaten und Einheitsvektoren

- $\mathbf{a} = (a_x, a_y, a_z)^T = a_x * \mathbf{i} + a_y * \mathbf{j} + a_z * \mathbf{k}$
- $\mathbf{i} = (1, 0, 0)^T \quad \mathbf{j} = (0, 1, 0)^T \quad \mathbf{k} = (0, 0, 1)^T$

$$T(z, \phi) = \begin{bmatrix} cos\ \phi & -sin\phi & 0 \\ sin\phi & cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotation der Einheitsvektoren um φ

- $\mathbf{i}' = (\cos\varphi, \sin\varphi, 0)^T$
- $\mathbf{j}' = (-\sin\varphi, \cos\varphi, 0)^T$
- $\mathbf{k}' = (0, 0, 1)^T = \mathbf{k}$

Translation um dx, dy, dz

- Vektoraddition $\mathbf{a} + \mathbf{t}$ mit $\mathbf{t} = (d_x, d_y, d_z)^T$

Erweiterung auf homogene Koordinaten:

$\mathbf{a} = (a_x, a_y, a_z, 1)^T = a * \mathbf{E}$

Kombination von Rotation und Translation

- Vektoraddition $\mathbf{a} = (a_x, a_y, a_z)^T = a_x * \mathbf{i} + a_y * \mathbf{j} + a_z * \mathbf{k} + \mathbf{t}$

Homogene Koordinaten erlauben die formale Kombination von Transformationen in einer Transformationsmatix. Die Multiplikation der Vektorkoordinaten mit der Transformationsmatrix führt Rotation und Translation aus.

Werkzeugkasten
Transformationsmatrizen

$$Trans(x, y, z) = \begin{pmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$Rot(x, \theta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$Rot(z, \theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$Rot(y, \theta) = \begin{pmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Die (fallspezifische) Multiplikation der Transformationsmatrizen ergibt die Gesamttransformation.

Grundlegendes Modell

Vorwärtskinematik

Modellierungskonventionen

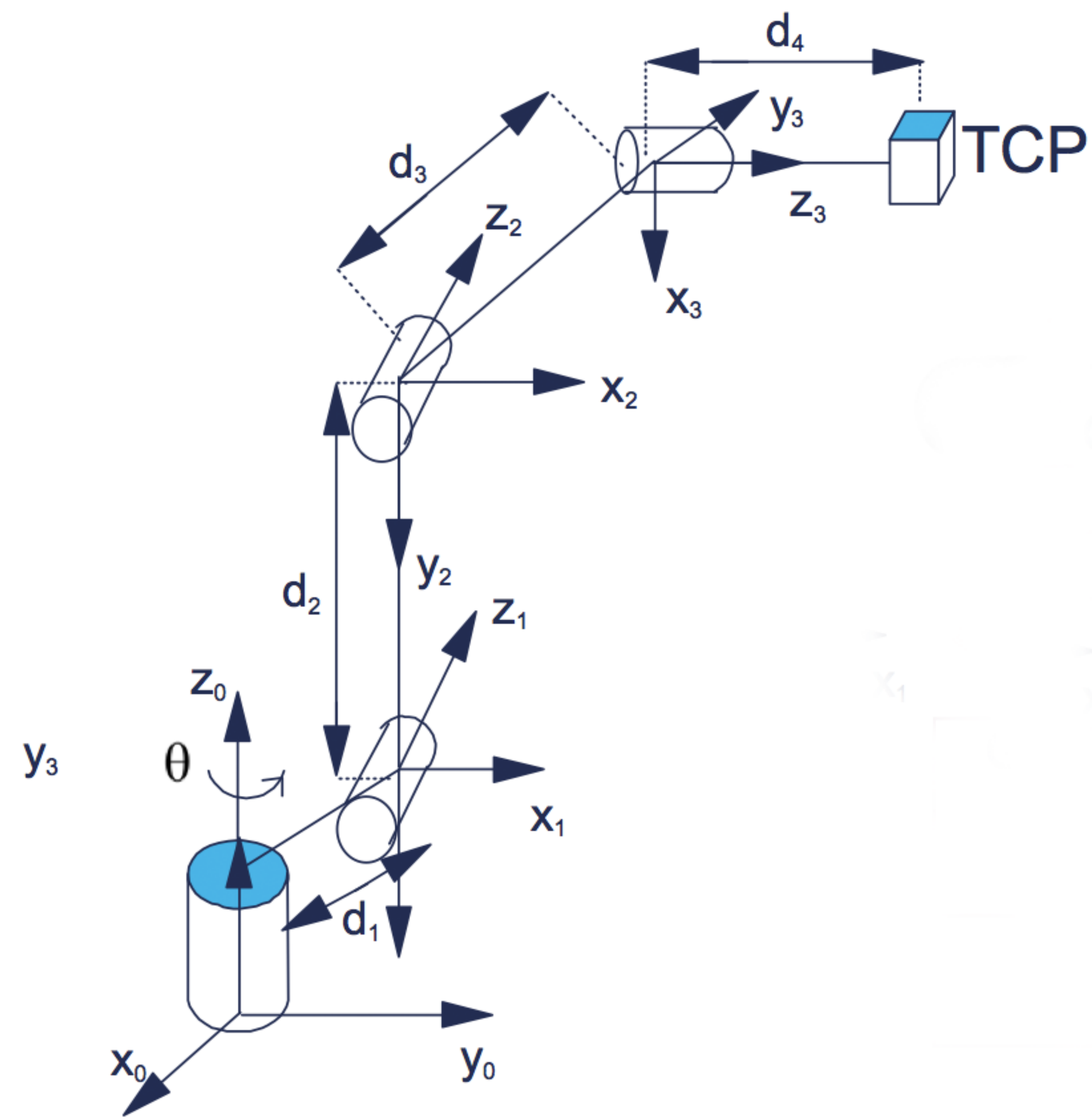
Modell eines Roboters
Kinematische Kette

Konvention nach Denavit Hartenberg (DH-Parameter):

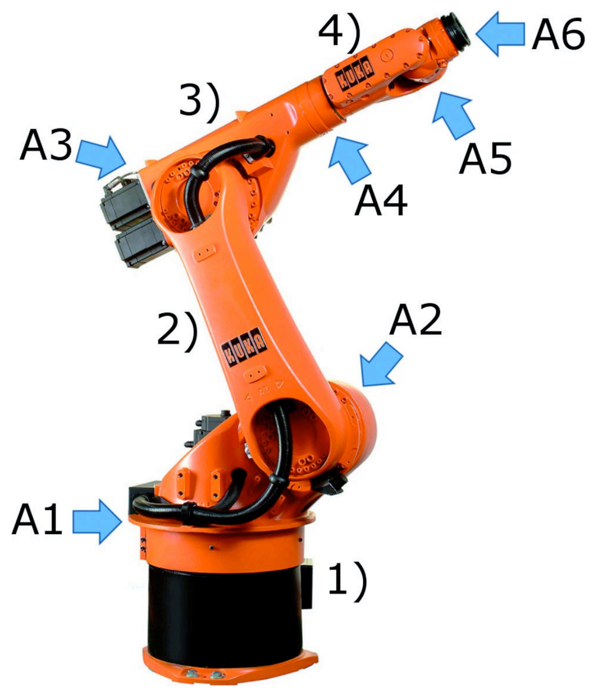
- Richtlinie zur Festlegung der Gelenkkoordinatensystem
- Erleichtert Transformation der KS
- DH-Parameter als Beschreibung des jeweiligen Aufbaus

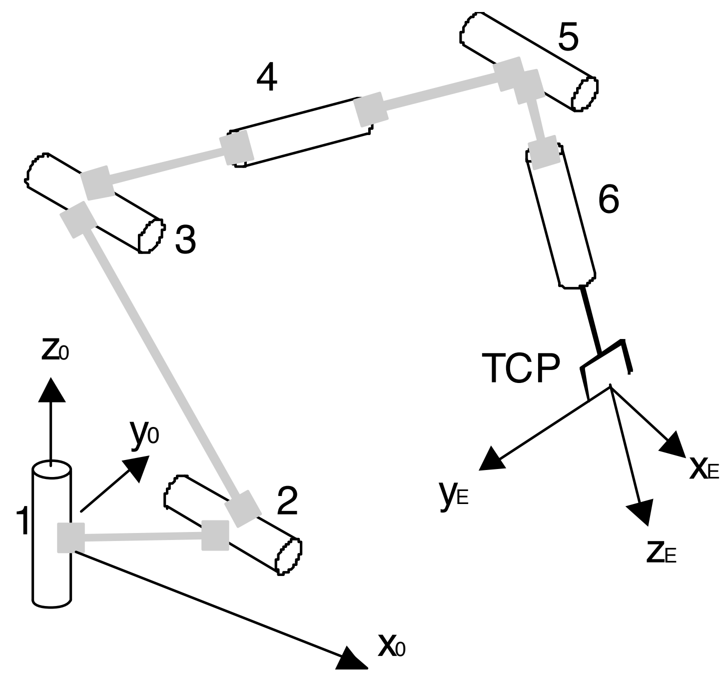
- a_i Translation um Armlänge
- α_i Verwindung um die x-Achse
- d_i Gelenkabstand, Translation (Höhe)
- θ_i Rotation um die Z_{n-1}-Achse

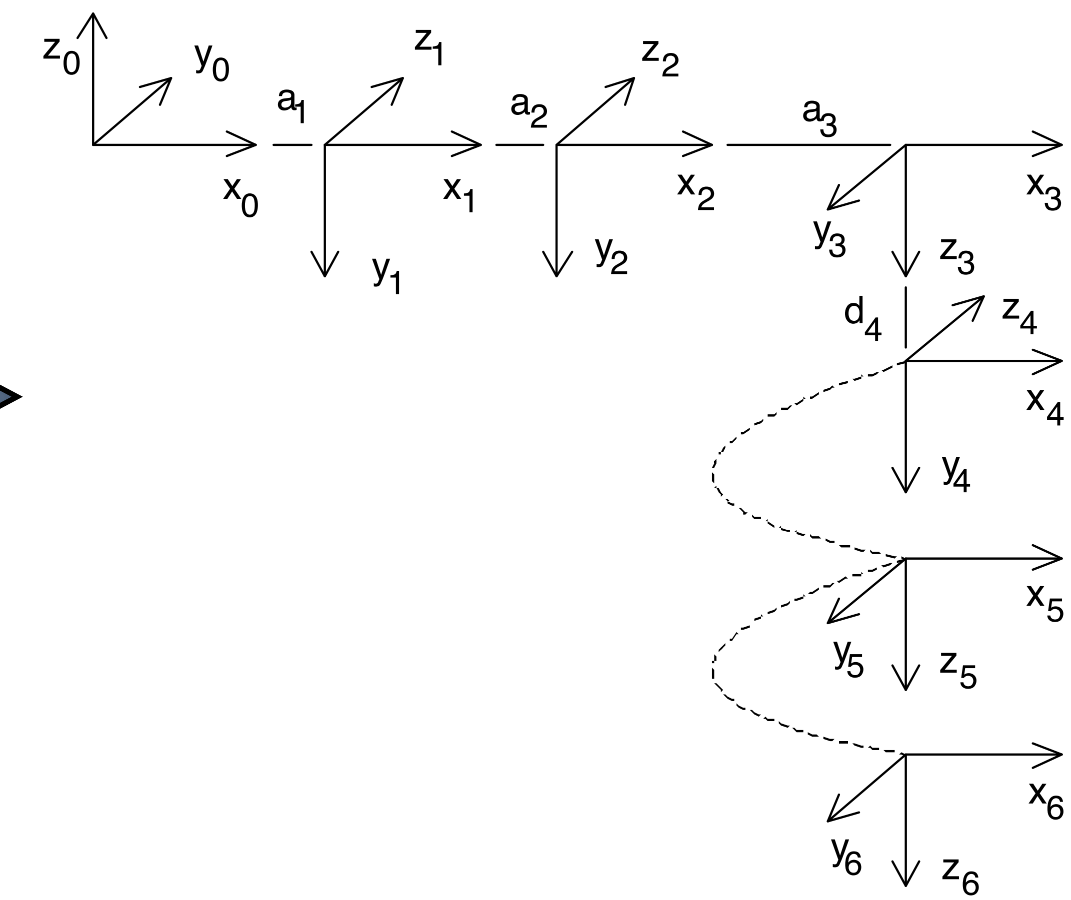
Gelenk	a_i	α_i	d_i	θ_i
1	0	0	d_1	θ_1
2	0	0	d_2	θ_2
3	0	-90°	d_3	θ_3
4	0	0	d_4	θ_4



Vorgehen Gesamttransformation Denavit-Hartenberg







Achse	θ	d	a	α
1	θ_1	0	a_1 (80)	$-\pi/2$
2	θ_2	0	a_2 (60)	0
3	θ_3	0	a_3 (40)	$-\pi/2$
4	θ_4	d_4 (20)	0	$\pi/2$
5	θ_5	0	0	$-\pi/2$
6	θ_6	0	0	0

- Konstruktive Parameter
- Gelenkstellung als variabler Parameter

Festlegung der Koordinatensysteme nach Vorschlag von Denavit und Hartenberg führt zu einer Vereinheitlichung und einer Vereinfachung der Transformationen.

Vorgehen
Festlegung der Bezugssysteme

Wie muss ich KS_{n-1} schrittweise drehen/ verschieben, so dass es mit KS_n übereinstimmt?

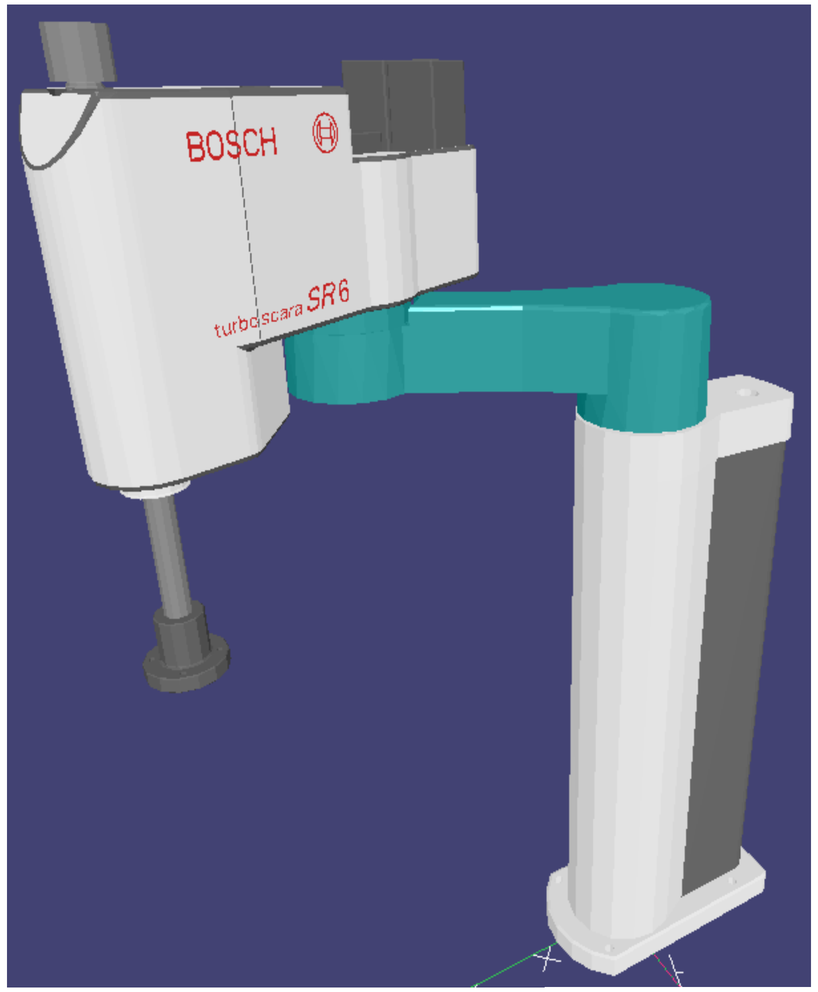
Denavit-Hartenberg

- Regeln/Vorgehen zur Beschreibung zweier benachbarter Bezugssysteme durch vier Transformationen als D-H-Parameter
- Koordinatensystem n-1 in Koordinatensystem n mittels 2 Rotationen und 2 Translationen
- abhängig von Gelenkwinkel und Konstruktion (Achsversatz und Armlänge) des Roboters

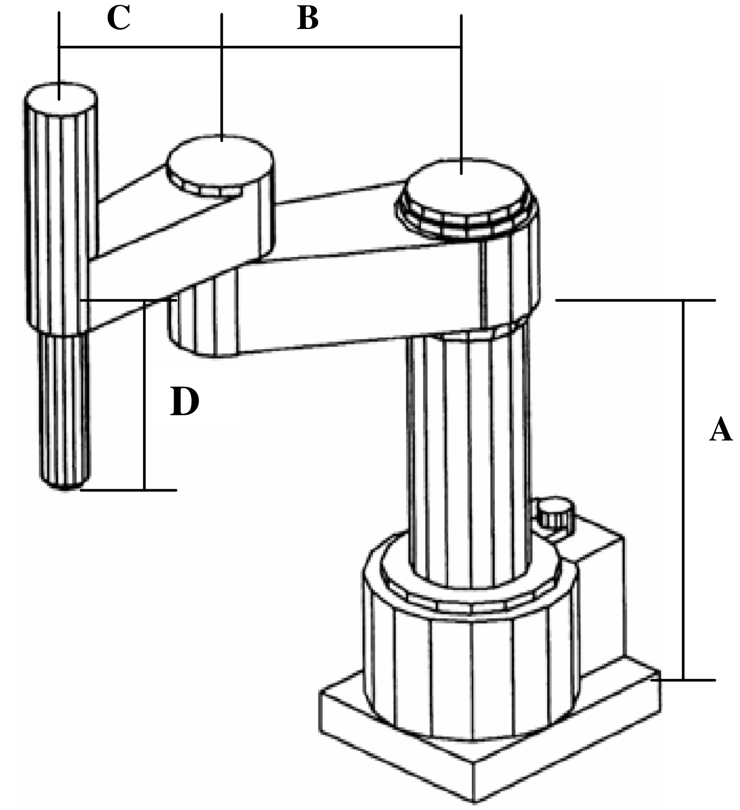
- 1. Rotation um n um die Achse z_{n-1} bis x_{n-1} parallel zu x_n liegt
- 2. Verschiebung um d_n in Richtung z_{n-1} bis sich x_{n-1} und x_n decken
- 3. Verschiebung um a_n in Richtung x_n bis Koordinatenursprünge gleich sind
- 4. Rotation um n um die Achse x_n bis die Koordinatensysteme identisch sind

D-H-Parameter liefern Vereinheitlichung und einfachere Transformationen!

D-H-Parameter
Beispiel SCARA



turbo SCARA SR6



SCARA1

Achse	Maß in mm
A	700
B	330
C	270
D	70

Achse	Θ in °	d in mm	a in mm	α in °
1	R_Z=0°	0	330	0
2	R_Z=0°	0	270	0
3	T_Z=0°	0	0	0
4	R_Z=0°	-70	0	0

Vorgehen
Transformationsmatrix für D-H

Gesamttransformation einer Achse:

$$T_{n-1,n} = \left[Rot(z_{n-1},\Theta)\right] * \left[Trans(0,0,d_n)\right] * \left[Trans(a_n,0,0)\right] * \left[Rot(x_n,\alpha_n)\right]$$

$$T_{n-1,n} = \begin{bmatrix} \cos\Theta & -\sin\Theta & 0 & 0 \\ \sin\Theta & \cos\Theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_n = \begin{bmatrix} \cos\Theta & -\sin\Theta\cos\alpha & \sin\Theta\sin\alpha & a * cos\Theta \\ \sin\Theta & \cos\Theta\cos\alpha & -\cos\Theta\sin\alpha & a * sin\Theta \\ 0 & \sin\alpha & \cos\alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix} = f(\Theta, d, \alpha, a)$$

Ergebnis ist die Transformationsmatrix für die Achse n. Die Kombination aller Achstransformation ergibt die Gesamttransformation.

Vorgehen Vorwärtskinematik

Gesamttransformation Matrizenprodukt aller Achstransformationen:

$T_G = T_1 * T_2 * \ldots. T_N$ mit N = Anzahl der Achsen

Eigenschaften/Elemente der Matrix:

$$T_G = \begin{pmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Position des Effektors:

$\mathbf{p} = (p_x, p_y, p_z)^T$

Orientierung des Effektors:

$$\begin{aligned} \phi_z &= \arctan(\frac{n_y}{n_x}) \\ \phi_y &= \arctan(\frac{-n_z}{n_x \cos\phi_z + n_y \sin\phi_z}) \\ \phi_x &= \arctan(\frac{a_x \sin\phi_z - a_y \cos\phi_z}{o_y \cos\phi_z - o_x \sin\phi_z}) \end{aligned}$$

Durch Auswertung der berechneten Gesamttransformationsmatrix und deren Eigenschaften läßt sich die Stellung des Effektors bestimmen.

Literatur

Bronstein, Ilja N., et al. Taschenbuch der Mathematik. Hrsg. Zeidler, E., Hackbusch, W., Vol. 1. Springer-Verlag, 2012.

Mitsubishi INSTRUCTION MANUAL CR750/CR751 Series Controller - Detailed explanations of functions and operations, 2012

Müller, W. H., Ferber, F.: Technische Mechanik für Ingenieure. Fachbuchverlag Leipzig im Carl-Hanser-Verlag, München u. a. 2019

Eberhard Brommundt, E., Sachs, G., Sachau, D.: Technische Mechanik. Eine Einführung. 4., verbesserte und erweiterte Auflage. Oldenbourg. München u. a. 2007, ISBN 978-3-486-58111-9, S. 47 ff

Dubbel, H.: Taschenbuch für den Maschinenbau. Eds. Wolfgang Beitz, and Karl-Heinz Küttner. Springer-Verlag, 2014

Snyder, W. E. "Computergesteuerte Industrieroboter." Robotik. Weinheim etc.: VCH, 1990.

Hesse, S.: Golems Enkel, Urania Verlag Leipzig/Jena/Berlin 1986

Asimov, Isaac, and Moray Powell. Isaac Asimov's book of facts. Grosset & Dunlap, 1979.

Ichbiah, D.; Roboter, Geschichte, Technik, Entwicklung; München 2005, S. 11

Ktschinksi, V,: Vorlesung Robotertechnik, 2008, Technische Universität Berlin