

# Grundlagen der Robotik

**Zusammenfassung**  
Grundlagen der Robotik  
15. Mai 2020



TECHNISCHE  
UNIVERSITÄT  
DARMSTADT



---

## Inhaltsverzeichnis

---



---

## Abbildungsverzeichnis

---



---

## Tabellenverzeichnis

---



---

## List of Algorithms

---

---

# 1 Einleitung

---

---

## 1.1 Was ist ein Roboter?

---

---

## 1.2 Was ist KI?

---

---

## 1.3 Was ist Robotik?

---

---

## 1.4 Sense – Plan – Act

---

### Act

Kinematik

Dynamik

Steuerung

### Sense

Sensoren

### Plan

Lokalisierung, Kartographie, Navigation, Bahnplanung

---

## **1.5 Geschichte der Robotik**

---

### **1.5.1 Historische Entwicklung**

---

### **1.5.2 Die drei Gebote der Robotik**

---

### **1.5.3 Autonome Fahrzeuge**

---

### **1.5.4 Entwicklungstrend**

---

## **1.6 Herausforderungen**

---

### **1.6.1 Humanoide Bewegung**

---

### **1.6.2 Roboter für menschliche Mobilität**

---

### **1.6.3 Roboter-Avatare**

---

### **Beine**

---

### **Katastrophenbewältigung und -hilfe**

---

### **Objekt-Vorlagen**

---

### **Greifen und Manipulation**

---

### **1.6.4 Die Robotik an sich**

---

---

## 2 Räumliche Darstellungen und Transformationen

---

---

### 2.1 Mathematische Grundlagen und Notation

---

---

#### 2.1.1 Vektoren

---

- Vektor:  $\mathbf{p}$  (fett)
- Nullvektor:  $\mathbf{0}$
- Transponierter Vektor:  $\mathbf{p}^T$
- Euklidische Norm eines Vektor:  $\|\mathbf{p}\|$
- Standard-Skalarprodukt:  $\mathbf{p} \circ \mathbf{r}$
- Orthogonale Vektoren:  $\mathbf{p} \perp \mathbf{r}$
- Winkel zwischen  $\mathbf{p}, \mathbf{r}$ :  $\phi$
- Kreuzprodukt:  $\mathbf{p} \times \mathbf{r}$
- Euklidische Einheitsvektoren:  $\mathbf{e}_x, \mathbf{e}_y, \mathbf{e}_z$
- Positionsvektor:  $\mathbf{p} = \begin{bmatrix} p_x & p_y & p_z \end{bmatrix}^T$

---

#### 2.1.2 Matrizen

---

- Matrix:  $\mathbf{R}$  (fett)
- Transponierte Matrix:  $\mathbf{R}^T$
- Einheitsmatrix:  $\mathbf{E}$
- Nullmatrix:  $\mathbf{O}$
- Inverse Matrix:  $\mathbf{R}^{-1}$

---

### 2.2 Klassische Transformationsbeziehungen

---

Darstellung von  $\cdot$  bzgl. einem Koordinatensystem  $S_a$ :  ${}^a\cdot$ .

Klassische Transformationsbeziehung:

$${}^a\mathbf{p} = {}^a\mathbf{r}_b + {}^a\mathbf{R}_b {}^b\mathbf{p}$$

mit

- ${}^a\mathbf{p}$ : Koordinaten von Punkt  $P$  bzgl.  $S_a$ ,
- ${}^b\mathbf{p}$ : Koordinaten von Punkt  $P$  bzgl.  $S_b$ ,
- ${}^a\mathbf{r}_b$ : Translationsvektor,
- ${}^a\mathbf{R}_b$ : Rotationsmatrix zwischen  $S_a$  und  $S_b$ .



---

## 2.3 Rotation eines Koordinatensystems

---

### 2.3.1 Rotationsmatrizen

---

Eigenschaften einer Rotationsmatrix  $\mathbf{R}$ :

- Orthonormalität:  $\mathbf{R}\mathbf{R}^T = \mathbf{E}$
- Orthonormale Spalten (d. h. Einheitsvektoren und paarweise orthogonal).

#### Elementare Rotationsmatrizen

- Rotation um  $x$ -Achse um  $\theta_x$ :

$$\mathbf{R}(x; \theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_x & -\sin \theta_x \\ 0 & \sin \theta_x & \cos \theta_x \end{bmatrix}$$

- Rotation um  $y$ -Achse um  $\theta_y$ :

$$\mathbf{R}(y; \theta_y) = \begin{bmatrix} \cos \theta_y & 0 & \sin \theta_y \\ 0 & 1 & 0 \\ -\sin \theta_y & 0 & \cos \theta_y \end{bmatrix}$$

- Rotation um  $z$ -Achse um  $\theta_z$ :

$$\mathbf{R}(z; \theta_z) = \begin{bmatrix} \cos \theta_z & -\sin \theta_z & 0 \\ \sin \theta_z & \cos \theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

---

### 2.3.2 Verkettete Rotationen

---

Im Allgemeinen können mehrere Rotationen direkt hintereinander ausgeführt werden, wobei dafür die Rotationsmatrizen multipliziert werden. Dafür gibt es zwei mögliche Interpretationsmöglichkeiten:

$${}^a\mathbf{R}_b = \overbrace{\mathbf{R}_n \cdot \mathbf{R}_{n-1} \cdots \mathbf{R}_2 \cdot \mathbf{R}_1}^{\text{Vormultiplikation}} \quad \text{Nachmultiplikation}$$

- *Vormultiplikation*  
Die Rotationen finden in der Reihenfolge  $1, 2, \dots, n$  statt und drehen das (momentane)  $S_b$ -System um die festen  $S_a$ -Achsen, das Bezugssystem ist immer  $S_a$ .
- *Nachmultiplikation*  
Die Rotationen finden in der Reihenfolge  $n, n-1, \dots, 2, 1$  statt und drehen das (momentane)  $S_b$ -System um die momentanen  $S_b$ -Achsen, das Bezugssystem ist immer das momentane  $S_b$ .

Prinzipiell liefern beide Interpretationen jedoch die gleiche Endmatrix (da Matrixmultiplikation assoziativ ist).

---

### 2.3.3 Winkelparameter

---

Eine allgemeine Rotation um alle Achsen benötigt nur 3 Parameter (es existieren nur drei Freiheitsgrade). Dabei können die verketteten Rotationen in sechs verschiedenen Reihenfolgen um drei bzw. zwei unterschiedliche Achsen, also insgesamt auf zwölf Arten geschehen. Geläufige Winkelkonventionen sind z. B. RPY- oder Euler-Winkel.

---

#### Kardan-Winkel

---

Bei *Kardan-Winkeln* erfolgen die drei verketteten Rotationen um die Winkel  $\psi$  (Yaw),  $\theta$  (Pitch),  $\phi$  (Roll) in dieser Reihenfolge um die Achsen des festen Bezugssystems  $S_a$ , d. h. durch Vormultiplikation.

Am häufigsten werden *X-Y-Z-Winkel* (auch *RPY-Winkel*) verwendet:

$$\begin{aligned} {}^a\mathbf{R}_b(\psi, \theta, \phi) &= \mathbf{R}(z; \phi) \cdot (\mathbf{R}(y; \theta) \cdot \mathbf{R}(x; \psi)) \\ &= \begin{bmatrix} \cos \phi \cos \theta & -\sin \phi \cos \theta + \cos \phi \sin \theta \sin \psi & \sin \phi \sin \theta \sin \psi + \cos \phi \cos \theta \cos \psi \\ \sin \phi \cos \theta & \cos \phi \cos \theta + \sin \phi \sin \theta \sin \psi & -\cos \phi \sin \theta \sin \psi + \sin \phi \cos \theta \cos \psi \\ -\sin \phi & \cos \phi \sin \psi & \cos \phi \cos \psi \end{bmatrix} \end{aligned}$$

Die inverse Umwandlung (von einer Rotationsmatrix zu RPY-Winkeln) kann dann wie folgt erfolgen:

1. Berechnung von  $\theta$  aus  $R_{31}$ .
2. Berechnung von  $\psi$  aus  $R_{23}$  oder  $R_{33}$ .
3. Berechnung von  $\psi$  aus einem der anderen Matrixelemente.

Diese Umwandlung ist nicht immer eindeutig!

---

### Euler-Winkel

---

Bei *Euler-Winkeln* erfolgen die drei verketteten Rotationen um die Winkel  $\alpha, \beta, \gamma$  in dieser Reihenfolge um die Achsen des momentanen Bezugssystems  $S_b$ , d. h. durch Nachmultiplikation.

Am häufigsten werden dabei *Z-Y-Z-Winkel* verwendet:

$${}^a\mathbf{R}_b = (\mathbf{R}(z; \alpha) \cdot \mathbf{R}(y; \beta)) \cdot \mathbf{R}(z; \gamma)$$

Andere Möglichkeiten sind z. B. *X-Y-Z-Winkel*

$${}^a\mathbf{R}_b = (\mathbf{R}(z; \alpha) \cdot \mathbf{R}(y; \beta)) \cdot \mathbf{R}(x; \gamma)$$

---

## 2.4 Homogene Transformationen

---

Die klassische Transformationsbeziehung

$${}^a\mathbf{p} = {}^a\mathbf{r}_b + {}^a\mathbf{R}_b {}^b\mathbf{p}$$

lässt sich durch *homogene Transformationen* auch durch eine einzigen Matrixmultiplikation darstellen:

$${}^a\hat{\mathbf{p}} = \begin{bmatrix} {}^a\mathbf{p} \\ 1 \end{bmatrix} = \begin{bmatrix} {}^a\mathbf{R}_b & {}^a\mathbf{r}_b \\ \mathbf{0}^T & 1 \end{bmatrix} \cdot \begin{bmatrix} {}^b\mathbf{p} \\ 1 \end{bmatrix} = {}^a\mathbf{T}_b \cdot {}^b\hat{\mathbf{p}}$$

Somit lautet die inverse einer homogenen Transformationsmatrix:

$$({}^a\mathbf{T}_b)^{-1} = \begin{bmatrix} ({}^a\mathbf{R}_b)^T & -({}^a\mathbf{R}_b)^T \cdot {}^a\mathbf{r}_b \\ \mathbf{0}^T & 1 \end{bmatrix}$$

---

## 3 Roboterkinematik

---

### 3.1 Vorwärtskinematik

---

Die Vorwärtskinematik DIR KIN berechnet die Endeffektor-Position/-Orientierung aus den Gelenkstellungen.

#### 3.1.1 Kinematische Ketten

---

Der Roboter wird dabei als *offene kinematische Kette* betrachtet, d. h. die

- Schub- (mit Ausfahrweite  $d_i$ ) und
- Drehgelenke (mit Rotation  $\theta_i$ )

als auch die Glieder werden als starr-, massen- sowie schleifenlos betrachtet.

Dabei hat jedes Gelenk einen Freiheitsgrad (Degree of Freedom, DOF). Ein Manipulator mit  $n$  Gelenken hat somit  $n$  DOF, wobei

1. die Gelenke von 1 bis  $n$  nummeriert werden und
2. das Gelenk  $i$  die Glieder  $i - 1$  und  $i$  verbindet.

Es gibt also  $n + 1$  Glieder wobei Glied 0 die Basis ist und Glied  $n$  den Endeffektor trägt.

#### 3.1.2 Kinematische Modellbildung

---

Mit jedem Glied  $i$  wird am Gelenk  $i + 1$  ein gliedfestes Koordinatensystem  $S_i$  befestigt, wobei zusätzlich das Basiskoordinatensystem  $S_0$  und das Endeffektorkoordinatensystem  $S_n$  definiert wird. Ein Manipulator wird also durch  $n + 1$  Koordinatensysteme beschrieben.

Ziel ist es nun, eine kinematische Modellstruktur

$${}^0\mathbf{T}_n = {}^0\mathbf{T}_1 \cdot {}^1\mathbf{T}_2 \cdot \dots \cdot {}^{n-2}\mathbf{T}_{n-1} \cdot {}^{n-1}\mathbf{T}_n = \prod_{i=1}^n {}^{i-1}\mathbf{T}_i$$

wobei die homogene Transformationsmatrix  ${}^0\mathbf{T}_n = {}^0\mathbf{T}_n(\mathbf{q}, \mathbf{f})$  eine Funktion der verallgemeinerten Gelenkvariablen  $\mathbf{q}$  (mit  $q_i = \theta_i$  für ein Drehgelenk und  $q_i = d_i$  für ein Schubgelenk) und der geometrischen Parameter  $\mathbf{f}$  ist.

#### 3.1.3 Denavit-Hartenberg (DH) Konventionen

---

Die Denavit-Hartenberg Konventionen (DH-Konventionen) ist ein internationaler, formaler Standard zur Festlegung der Koordinatensysteme. Einige der Basiseigenschaften von per DH-Konvention festgelegten Koordinatensystemen sind:

- Die Koordinatensysteme liegen in den jeweiligen Bewegungsachsen.
- Die  $z_{i-1}$ -Achse liegt entlang der Bewegungsachse des  $i$ -ten Gelenks.
- Die  $x_i$ -Achse steht senkrecht zur  $z_{i-1}$ -Achse und zeigt von ihr weg.
- Die  $x_i$ -Achse und die  $z_{i-1}$ -Achse haben einen Schnittpunkt.

#### Pseudo-Algorithmus zur Festlegung der Koordinatensysteme

---

Voraussetzung: Der Manipulator befindet sich in allen  $n$  Gelenken in Nullstellung (z. B. ausgestreckt nach oben).

Legt der folgende Algorithmus in einzelnen Schritten nicht eindeutig sein, so ist zuerst die Einhaltung der DH-Eigenschaften zu gewährleisten und ansonsten die Wahl einer Lösung mit möglichst geringer Komplexität der Transformationen zu bevorzugen.

##### Schritt 1

**Schritt 1a** Nummerierung der Glieder von 0 (Basis) bis  $n$ .

**Schritt 1b (für  $i = 0, \dots, n-1$ )** Festlegung der  $z_i$ -Achsen als koinzident mit der Bewegungsachse des  $(i+1)$ -ten Gelenks:

- Bei Schubgelenken in Richtung weg von Gelenk  $i+1$ .
- Bei Drehgelenken als Rotationsachse in Richtung positiver Drehwinkel (wird festgelegt).

**Schritt 2** Festlegung von  $S_0$  mit Ursprung auf der  $z_0$ -Achse, sodass sich ein Rechtskoordinatensystem ergibt (oft werden  $x_0$  und  $y_0$  parallel zum Welt-Koordinatensystem gewählt).

**Schritt 3 (für  $i = 1, \dots, n-1$ )** Festlegung des Ursprungs von  $S_i$ :

- Falls  $z_{i-1}$  und  $z_i$  sich schneiden:  
Schnittpunkt wird der Ursprung.
- Falls  $z_{i-1}$  und  $z_i$  parallel sind:  
Festlegung auf der  $z_i$ -Achse am Gelenk  $i+1$ .
- Sonst ( $z_{i-1}$  und  $z_i$  windschief):  
Bilde eine gemeinsame Normale zu  $z_i$  und  $z_{i-1}$ , Schnittpunkt der Normalen mit  $z_i$  wird der Ursprung.

**Kurzschreibweise**

$$S_i = \begin{cases} z_{i-1} \cap z_i & \text{falls } |z_{i-1} \cap z_i| = 1 \\ z_i \cap \text{Gelenk}_{i+1} & \text{falls } z_{i-1} \parallel z_i \\ z_i \cap (\perp(z_i, z_{i-1})) & \text{sonst} \end{cases}$$

**Schritt 4 (für  $i = 1, \dots, n-1$ )** Festlegung der  $x_i$ -Achse:

- Falls  $z_{i-1}$  und  $z_i$  sich schneiden:  
 $x_i = z_{i-1} \times z_i$  oder  $x_i = z_i \times z_{i-1}$  (unter Berücksichtigung der DH-Eigenschaften!).
- Sonst ( $z_{i-1}$  und  $z_i$  parallel oder windschief):  
 $x_i$ -Achse in Richtung der gemeinsamen Normalen von  $z_{i-1}$  und  $z_i$ , sodass sich  $x_i$  und  $z_{i-1}$  schneiden.

**Kurzschreibweise**

$$x_i = \begin{cases} x_i \in \{ z_{i-1} \times z_i, z_i \times z_{i-1} \} & \text{falls } |z_{i-1} \cap z_i| = 1 \\ \perp(z_{i-1}, z_i) & \text{sonst} \end{cases}$$

**Schritt 5 (für  $i = 1, \dots, n-1$ )** Festlegung der  $y_i$ -Achse, sodass  $x_i, y_i, z_i$  ein Rechtskoordinatensystem bilden (d. h.  $z_i = x_i \times y_i$ , Rechte-Hand-Regel!).

**Kurzschreibweise**

$$z_i \stackrel{!}{=} x_i \times y_i$$

**Schritt 6** Festlegung des Endeffektor-Koordinatensystems  $S_n$ : Der Ursprung wird meistens in den Tool-Center-Point (TCP) gelegt.

- Liegen keine besonderen Bedingungen vor:  
Erstellung einer möglichst einfachen Transformation (häufig eine reine Translation).
- Ist der Endeffektor ein einfacher Greifer:  
Meist eine solche Festlegung, dass Yaw-, Pitch- und Roll-Winkel verwendet werden können.

**Schritt 7 (für  $i = 1, \dots, n$ )** Erstellen einer Tabelle von Gliedparametern  $\theta_i, d_i, a_i, \alpha_i$ :

$\theta_i$  Winkel zwischen  $x_{i-1}$  und  $x_i$ , gemessen um  $z_{i-1}$ . Variabel, falls  $i$  ein Drehgelenk ist.

$d_i$  Entfernung vom  $S_{i-1}$ -Ursprung entlang  $z_{i-1}$  zum Schnittpunkt mit  $x_i$ . Variablen, falls  $i$  ein Schubgelenk ist.

$a_i$  Entfernung vom Schnittpunkt von  $z_{i-1}$  und  $x_i$  entlang  $x_i$  zum  $S_i$ -Ursprung. Je nach Orientierung von  $x_i$  kann  $a_i$  auch negativ sein.

$\alpha_i$  Winkel zwischen  $z_{i-1}$  und  $z_i$ , gemessen um  $x_i$ .

### Kurzschreibweise

$$\begin{aligned}\theta_i &:= x_{i-1} \angle_{z_{i-1}} x_i \\ d_i &:= |S_{i-1} \rightarrow_{z_{i-1}} (z_{i-1} \cap x_i)| \\ a_i &:= |(z_{i-1} \cap x_i) \rightarrow_{x_i} S_i| \\ \alpha_i &:= z_{i-1} \angle_{x_i} z_i\end{aligned}$$

**Schritt 8** (für  $i = 1, \dots, n$ ) Bildung der homogenen Transformationsmatrizen:

$${}^{i-1}\mathbf{T}_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

**Schritt 9** Berechnung von

$${}^0\mathbf{T}_n = {}^0\mathbf{T}_1 \cdot {}^1\mathbf{T}_2 \cdot \dots \cdot {}^{n-2}\mathbf{T}_{n-1} \cdot {}^{n-1}\mathbf{T}_n = \prod_{i=1}^n {}^{i-1}\mathbf{T}_i$$

## 3.2 Rückwärtskinematik (Inverse Kinematik)

Die Rückwärtskinematik INV KIN berechnet die benötigten Gelenkstellungen für eine Endeffektor-Position/-Orientierung.

Gegeben einer Endeffektor-Position/-Orientierung  ${}^0\tilde{\mathbf{T}}_E$  ist ein Vektor  $\mathbf{q} \in \mathbb{R}^n$  gesucht, sodass

$${}^0\mathbf{T}_E(\mathbf{q}) = {}^0\tilde{\mathbf{T}}_E$$

gilt. Dies liefert ein System von 6 unabhängigen, nichtlinearen, gekoppelten Gleichungen.

Allgemein gilt: Wird eine RAN beliebig vorgegeben, so existiert

- für  $n < 6$  Gelenkvariablen i. A. keine Lösung (es dürfen maximal  $n$  Werte vorgegeben werden),
- für  $n = 6$  Gelenkvariablen i. A. genau eine Lösung (dies ist der Fall für viele 6-DOF Industrieroboter),
- für  $n > 6$  Gelenkvariablen i. A. unendliche viele Lösungen.

Definition: Der *erreichbare Arbeitsraum*  $\mathcal{A} \subseteq \mathbb{R}^3$  ist der Raum, der von dem Manipulator in mindestens einer Stellung erreicht werden kann.

Definition: Der *vollmanipulierbare Arbeitsraum* die ist Teilmenge von  $\mathcal{A}$ , die von dem Manipulator mit allen Orientierungen erreicht werden kann.

### 3.2.1 Numerische Berechnung

Es kann bspw. das klassische Newton-Verfahren zur Lösung nichtlinearer Gleichungssysteme eingesetzt werden (sofern DIR KIN differenzierbar ist).

**Nachteile:**

- Kein iteratives Verfahren kann alle Lösungen garantiert berechnen. Dies wird jedoch oftmals benötigt.
- Iterative Berechnungsverfahren sind wesentlich langsamer als explizite, geschlossene Verfahren. Außerdem sind sie nicht mit den Echtzeitanforderungen vereinbar.
- Das Konvergenzverhalten ist oftmals problematisch.

### 3.2.2 Analytische Lösung

- **Vorteil:** Es können alle Lösungen in garantierter Berechnungszeit berechnet werden.
- **Nachteil:** Aufwendig zu bestimmen.
- Dennoch sollten, wenn möglich, explizite Lösungsformeln zumindest für einen Teil gefunden werden.

---

### Algebraische Ermittlung

---

Die Rückwärtslösung wird rein durch umformen von Gleichungen gefunden.

---

### Geometrische Lösung

---

Die Rückwärtslösung wird durch geometrische (bspw. trigonometrische) Betrachtung des Roboters ermittelt, wobei das Problem in mehrere Teilprobleme zerlegt werden kann.

---

### Algorithmische Ermittlung

---

- **Vorteil:** Falls eine Lösungsformel bestimmt werden kann schneller als iterative Verfahren.
- **Nachteil:** Eine algorithmische Lösung garantiert nicht das finden einer Lösung, auch wenn eine solche existiert.

---

## 3.3 Genauigkeit des kinematischen Modells

---

Es wird zwischen mehreren, unterschiedlichen „Genauigkeiten“ unterschieden:

- *Positionsgenauigkeit*  
Die Differenz zwischen tatsächlicher und vorgegebener RAN.
- *Wiederholungsgenauigkeit*  
Die Variation der Positionsgenauigkeit bei mehrfacher Wiederholung.
- *Auflösungsgenauigkeit*  
Die kleinstmögliche Distanz, über die der TCP garantiert bewegt werden kann.

Für die meisten Roboter sind Wiederholungsgenauigkeit und Auflösungsgenauigkeit viel besser als die Positionsgenauigkeit (z. B. durch Unsicherheiten in den geometrischen Parametern und die Nichtberücksichtigung von Reibung, Spiel, temperaturabhängigen Ausdehnungen, ...).

---

## 4 Geschwindigkeit, Jacobi-Matrix und statische Kräfte

---

Bisher wurde ausschließlich die Positionen von Gelenken, Gliedern und dem Endeffektor untersucht. In diesem Kapitel werden die Geschwindigkeiten und Beschleunigungen untersucht.

---

### 4.1 Vektor der Winkelgeschwindigkeiten

---

Sei  $\mathbf{R}(\mathbf{q}(t))$  eine Rotationsmatrix in impliziter Abhängigkeit von der Zeit. Dann gilt

$$\frac{d}{dt}(\mathbf{R}(\mathbf{q}(t))) = \mathbf{B}(\boldsymbol{\omega}(t)) \cdot \mathbf{R}(\mathbf{q}(t))$$

mit der schiefsymmetrischen Matrix

$$\mathbf{B}(\boldsymbol{\omega}) := \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}$$

wobei  $\boldsymbol{\omega} = [\omega_x \quad \omega_y \quad \omega_z]^T$  den Vektor der Winkelgeschwindigkeiten der Rotation darstellt.

Über die Beziehung

$$\mathbf{B}(\boldsymbol{\omega}(\mathbf{q}(t), \dot{\mathbf{q}}(t))) = \left( \sum_{i=1}^n \frac{\partial \mathbf{R}(\mathbf{q}(t))}{\partial q_i(t)} \cdot \dot{q}_i(t) \right) \cdot \mathbf{R}(\mathbf{q}(t))^T$$

kann damit die Winkelgeschwindigkeit der Rotation des Endeffektor bestimmt werden (durch Ablesen der Elemente von  $\mathbf{B}$ ).

---

### 4.2 Jacobi-Matrix eines Manipulators

---

**Achtung:** Die Jacobi-Matrix eines Manipulators darf nicht mit der Jacobi-Matrix einer Funktion mehrerer Veränderlicher verwechselt werden!

---

#### 4.2.1 Verkettung von Rotationen

---

Für verkettete Rotationen

$${}^0\mathbf{R}_n = {}^0\mathbf{R}_1 \cdot {}^1\mathbf{R}_2 \cdot \dots \cdot {}^{n-2}\mathbf{R}_{n-1} \cdot {}^{n-1}\mathbf{R}_n$$

gilt

$$\frac{d}{dt}({}^0\mathbf{R}_n) = \mathbf{B}({}^0\boldsymbol{\omega}_n) \cdot {}^0\mathbf{R}_n$$

mit dem Winkelgeschwindigkeitsvektor

$${}^0\boldsymbol{\omega}_n = {}^0\boldsymbol{\omega}_1 + {}^0\mathbf{R}_1 \cdot {}^1\boldsymbol{\omega}_2 + \dots + {}^0\mathbf{R}_{n-1} \cdot {}^{n-1}\boldsymbol{\omega}_n$$

---

#### 4.2.2 Zusammenfassung

---

Das folgende gilt nur, wenn die Koordinatensysteme entsprechend der DH-Konvention platziert sind!

Für die Jacobi-Matrix  ${}^0\mathbf{J}$  gilt:

$$\begin{bmatrix} {}^0\mathbf{v}(t) \\ {}^0\boldsymbol{\omega}(t) \end{bmatrix} = {}^0\mathbf{J}_n(\mathbf{q}(t)) \cdot \dot{\mathbf{q}}(t)$$

mit der Lineargeschwindigkeit  ${}^0\mathbf{v}(t)$  und Winkelgeschwindigkeit  ${}^0\boldsymbol{\omega}(t)$  des Endeffektors in Bezug auf das Basiskoordinatensystem.

Die  $i$ -te Spalte  ${}^0\mathbf{J}_{n,i}$  der Manipulator-Jacobi-Matrix berechnet sich wie folgt. Ist Gelenk  $i$  ein

- Drehgelenk:

$${}^0\mathbf{J}_{n,i} = \begin{bmatrix} {}^0\mathbf{e}_{z_{i-1}} \times ({}^0\mathbf{r}_n - {}^0\mathbf{r}_{i-1}) \\ {}^0\mathbf{e}_{z_{i-1}} \end{bmatrix}$$

- Schubgelenk:

$${}^0\mathbf{J}_{n,i} = \begin{bmatrix} {}^0\mathbf{e}_{z_{i-1}} \\ \mathbf{0} \end{bmatrix}$$

Somit kann die Jacobi-Matrix direkt aus DIR KIN berechnet werden.

Anmerkung: Oft kann es (insbesondere bei der linearen Geschwindigkeit eines Drehgelenks) einfacher sein,  ${}^0\mathbf{r}_n$  direkt nach den Gelenkvariablen abzuleiten.

---

### 4.3 Inverses Jacobi-Modell

---

Das inverse Jacobi-Modell (INV KIN) berechnet die benötigten Gelenkgeschwindigkeiten für eine Endeffektor-Geschwindigkeit.

Gegeben einer Linear- und Winkelgeschwindigkeit  ${}^0\mathbf{v}_n, {}^0\boldsymbol{\omega}_n$  ist ein Vektor  $\dot{\mathbf{q}}$  gesucht, sodass

$$\begin{bmatrix} {}^0\mathbf{v}_n \\ {}^0\boldsymbol{\omega}_n \end{bmatrix} = {}^0\mathbf{J}_n \cdot \dot{\mathbf{q}}$$

gilt. Dies liefert ein System von 6 unabhängigen, nichtlinearen, gekoppelten Gleichungen.

Allgemein gilt: Werden Geschwindigkeiten  ${}^0\mathbf{v}_n, {}^0\boldsymbol{\omega}_n$  beliebig vorgegeben, so existiert

- für  $n < 6$  Gelenkvariablen i. A. keine Lösung (es dürfen maximal  $n$  Werte vorgegeben werden),
- für  $n = 6$  Gelenkvariablen i. A. genau eine Lösung (dies ist der Fall für viele 6-DOF Industrieroboter),
- für  $n > 6$  Gelenkvariablen i. A. unendliche viele Lösungen.

Für solche „redundanten“ Manipulatoren können die üblichen Freiheitsgrade z. B. durch Hinzunahme von kinematischen Zwangsbedingungen oder durch die Maximierung eines Gütekriteriums, z. B. des Manipulierbarkeitsmaßes nach Yoshikawa

$$\mu(\mathbf{q}) = \sqrt{\det({}^0\mathbf{J}_n \cdot {}^0\mathbf{J}_n^T)}$$

festgelegt werden.

---

### 4.4 Kinematische Singularitäten

---

Sei

$$m := \max_{\mathbf{q} \in \mathcal{A}} \text{rank}({}^0\mathbf{J}_n(\mathbf{q}))$$

der maximale Rang der Jacobi-Matrix. Dann heißt jede Konfiguration  $\mathbf{q}_s$  mit

$$\text{rank}({}^0\mathbf{J}_n(\mathbf{q}_s)) < m$$

*kinematische Singularität.*

Bei einer solchen Singularität ist die Jacobi-Matrix singulär und es müssten unendliche hohe Geschwindigkeiten erreicht werden (ebenso nehmen die Geschwindigkeit bei der Annäherung an eine solche Singularität i. A. zu).

Zur Berechnung der kinematischen Singularitäten (z. B. um diese bei der Bahnplanung zu vermeiden), ist folgendes vorgehen möglich: Zunächst wird die Jacobi-Matrix auf eine  $(m \times m)$ -Untermatrix  ${}^0\hat{\mathbf{J}}_n(\mathbf{q})$  mit  $m = \text{rank}({}^0\hat{\mathbf{J}}_n(\mathbf{q}))$  eingeschränkt. Dann ist eine Konfiguration  $\mathbf{q}_s$  genau dann singulär, wenn

$$\det({}^0\hat{\mathbf{J}}_n(\mathbf{q}_s)) = 0$$

gilt, d. h. die singulären Konfigurationen können durch Nullsetzen der Determinante der eingeschränkten Jacobi-Matrix gefunden werden.



---

#### **4.4.1 Vermeidung**

---

#### **4.4.2 Umgang mit unvermeidbaren Singularitäten**

---

### **4.5 Nicht-holonome Kinematik mehrrädriger Fahrzeuge**

---

#### **4.5.1 Differentialantrieb**

---

#### **4.5.2 Allgemeines Vorwärtskinematikproblem für Fahrzeuge**

---

#### **4.5.3 Inverses Kinematikproblem**

---

#### **4.5.4 Omnidirektionale Dreirad-Kinematik**

---

#### **4.5.5 Weitere Antriebsarten von Fahrzeugen**

---

### **4.6 Statische Kräfte bei Manipulatoren**

---

---

## 5 Roboterdynamik

---

Bisher wurden keine (statischen) Kräfte, sondern nur Positionen und Geschwindigkeiten betrachtet. Die inverse Dynamik/Kinetik INV DYN berechnet für gegebene Kräfte (z. B. die Gravitation) die Kräfte/Momente, die die Gelenke aufbringen müssen.

Seien  $\tau_i$  die verallgemeinerten Kräfte/Drehmomente wobei

- $\tau_i = n_i$  für ein Drehgelenk (Drehmoment) und
- $\tau_i = f_i$  für ein Schubgelenk (Kraft) gilt.

---

### 5.1 Massenverteilung eines Starrkörpers

---

Die Massenverteilung eines Starrkörpers wird durch den symmetrischen Trägheitstensor

$${}^a\mathbf{I} \in \mathbb{R}^{3 \times 3} = \begin{bmatrix} {}^aI_{xx} & -{}^aI_{xy} & -{}^aI_{xz} \\ -{}^aI_{xy} & {}^aI_{yy} & -{}^aI_{yz} \\ -{}^aI_{xz} & -{}^aI_{yz} & {}^aI_{zz} \end{bmatrix}$$

bezüglich eines körperfesten Koordinatensystems  $S_a$  beschrieben. Die Elemente der Hauptdiagonale heißen dabei *Massenträgheitsmomente* und die restlichen *Massenträgheitsprodukte*, die sich wie folgt berechnen (mit dem Volumen  $\mathcal{V}$ , einem differentiellen Volumenelement  $dv$  und der Materialdichte  $\rho(x, y, z)$ ):

$$\begin{aligned} {}^aI_{xx} &= \iiint_{\mathcal{V}} (y^2 + z^2) \rho(x, y, z) dv & {}^aI_{xy} &= \iiint_{\mathcal{V}} xy \rho(x, y, z) dv \\ {}^aI_{yy} &= \iiint_{\mathcal{V}} (x^2 + z^2) \rho(x, y, z) dv & {}^aI_{xz} &= \iiint_{\mathcal{V}} xz \rho(x, y, z) dv \\ {}^aI_{zz} &= \iiint_{\mathcal{V}} (x^2 + y^2) \rho(x, y, z) dv & {}^aI_{yz} &= \iiint_{\mathcal{V}} yz \rho(x, y, z) dv \end{aligned}$$

Sind  $x_a, y_a$  und  $z_a$  die Hauptträgheitsachsen, so verschwinden die die Massenträgheitsprodukte. Dies ist zum Beispiel der Fall, wenn die Achsen paarweise Symmetrieebenen des Körpers bilden (bilden zwei Achsen eine Symmetrieebene, so verschwinden die Trägheitsprodukte der dritten Achse).

---

#### 5.1.1 Transformation von Trägheitstensoren

---

**(Reine) Rotation des Bezugssystems** Ist das System  $S_b$  um  ${}^a\mathbf{R}_b$  zu  $S_a$  rotiert, so gilt:

$${}^b\mathbf{I} = {}^b\mathbf{R}_a \cdot {}^a\mathbf{I} \cdot {}^a\mathbf{R}_b$$

**Translation des Bezugssystems (Satz von Steiner)** Liegt  $S_c$  im *Schwerpunkt* des Körpers und das System  $S_b$  ist um  ${}^c\mathbf{r}_b$  zu  $S_c$  verschoben, aber nicht rotiert, so gilt:

$${}^b\mathbf{I} = {}^c\mathbf{I} + m \cdot ({}^c\mathbf{r}_b^T \cdot {}^c\mathbf{r}_b \cdot \mathbf{E} - {}^c\mathbf{r}_b \cdot {}^c\mathbf{r}_b^T)$$

**Zusammenfügen von Körpern** Werden die Trägheitstensoren  $\mathbf{I}_1$  und  $\mathbf{I}_2$  von zwei Körpern, die verbunden werden, im gleichen Koordinatensystem dargestellt, so ergibt sich der Gesamtträgheitstensor des zusammengeführten Körpers:

$$\mathbf{I} = \mathbf{I}_1 + \mathbf{I}_2$$

---

### 5.2 Newton-Euler Formulierung der Roboterdynamik

---

Seien für jedes Roboterglied  $i$  die folgenden Daten gegeben:

- Der Koordinatenvektor des Schwerpunkts
  - bezüglich  $S_i$  als  ${}^i\mathbf{r}_{c_i}$ ,

- bezüglich  $S_0$  als  ${}^0\mathbf{r}_{c_i}$  und
- bezüglich  $S_{i-1}$  als  ${}^{i-1}\mathbf{r}_{c_i}$ .

- Die Gesamtmasse  $m_i$ .
- Der Trägheitstensor  ${}^{c_i}\mathbf{I}_i$  bezüglich  $S_{c_i}$ .

Dabei sei  $S_{c_i}$  das Schwerpunktkoordinatensystem von Glied  $i$  mit der gleichen Orientierung wie  $S_i$ .  
Weitere Notationen:

- $\boldsymbol{\omega}_i := {}^i\mathbf{R}_0 \cdot {}^0\boldsymbol{\omega}_i$   
Winkelgeschwindigkeit des  $i$ -ten Gelenks, gemessen in  $S_0$  und dargestellt in  $S_i$ .
- $\dot{\boldsymbol{\omega}}_i := {}^i\mathbf{R}_0 \cdot {}^0\dot{\boldsymbol{\omega}}_i$   
Winkelbeschleunigung des  $i$ -ten Gelenks, gemessen in  $S_0$  und dargestellt in  $S_i$ .
- $\mathbf{v}_i := {}^i\mathbf{R}_0 \cdot {}^0\mathbf{v}_i$   
Linearer Geschwindigkeitsvektor des  $i$ -ten Glieds an Gelenk  $i + 1$ , gemessen in  $S_0$  und dargestellt in  $S_i$ .
- $\dot{\mathbf{v}}_i := {}^i\mathbf{R}_0 \cdot {}^0\dot{\mathbf{v}}_i$   
Linearer Beschleunigungsvektor des  $i$ -ten Glieds an Gelenk  $i + 1$ , gemessen in  $S_0$  und dargestellt in  $S_i$ .

### 5.2.1 Iterative Berechnung von INV DYN

Für die Basis werden die Werte  $\boldsymbol{\omega}_0$ ,  $\dot{\boldsymbol{\omega}}_0$ ,  $\mathbf{v}_0$  und  $\dot{\mathbf{v}}_0$  vorgegeben (z. B.  $\dot{\mathbf{v}}_0 = \mathbf{g}$  mit dem Gravitationsvektor  $\mathbf{g}$ ). Des weiteren werden die Werte  $\mathbf{f}_{n+1}$  und  $\mathbf{n}_{n+1}$  für den Endeffektor vorgegeben (eine am Endeffektor sorgt dabei nicht für die Änderung dieser Werte, sondern für eine Änderung der Masse  $m_n$ ).

Sei außerdem

$$\rho_i := \begin{cases} 1 & \text{falls } i \text{ Drehgelenk} \\ 0 & \text{falls } i \text{ Schubgelenk} \end{cases}$$

**Schritt NE.1** (für  $n = 1, \dots, n$ ) Berechnung der Linear- und Winkelbeschleunigungen (von innen nach außen).

$$\begin{aligned} \boldsymbol{\omega}_i &= {}^i\mathbf{R}_{i-1} \left( \boldsymbol{\omega}_{i-1} + \rho_i [{}^{i-1}\mathbf{e}_{z_{i-1}} \dot{q}_i] \right) \\ \dot{\boldsymbol{\omega}}_i &= {}^i\mathbf{R}_{i-1} \left( \dot{\boldsymbol{\omega}}_{i-1} + \rho_i \left[ {}^{i-1}\mathbf{e}_{z_{i-1}} \ddot{q}_i + \boldsymbol{\omega}_{i-1} \times ({}^{i-1}\mathbf{e}_{z_{i-1}} \dot{q}_i) \right] \right) \\ \mathbf{v}_i &= {}^i\mathbf{R}_{i-1} \mathbf{v}_{i-1} + \boldsymbol{\omega}_i \times ({}^i\mathbf{R}_{i-1} {}^{i-1}\mathbf{r}_i) + (1 - \rho_i) [{}^i\mathbf{e}_{z_{i-1}} \dot{q}_i] \\ \dot{\mathbf{v}}_i &= {}^i\mathbf{R}_{i-1} \dot{\mathbf{v}}_{i-1} + \dot{\boldsymbol{\omega}}_i \times ({}^i\mathbf{R}_{i-1} {}^{i-1}\mathbf{r}_i) + \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times ({}^i\mathbf{R}_{i-1} {}^{i-1}\mathbf{r}_i)) + (1 - \rho_i) [2\boldsymbol{\omega}_i \times ({}^i\mathbf{e}_{z_{i-1}} \dot{q}_i) + {}^i\mathbf{e}_{z_{i-1}} \ddot{q}_i] \\ \dot{\mathbf{v}}_{c_i} &= \dot{\mathbf{v}}_i + (\dot{\boldsymbol{\omega}}_i \times {}^i\mathbf{r}_{c_i}) + \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times {}^i\mathbf{r}_{c_i}) \end{aligned}$$

**Schritt NE.2** (für  $i = 1, \dots, n$ ) Berechnung der am Schwerpunkt wirkenden Kräfte  $\mathbf{F}_i$  und Drehmomente  $\mathbf{N}_i$ . Dieser Schritt kann auch in Schritt NE.1 integriert werden.

$$\begin{aligned} \mathbf{F}_i &= \begin{cases} m_i \dot{\mathbf{v}}_{c_i} - m_i \mathbf{g}_i & \text{falls der Einfluss der Gravitation im Schritt NE.2 berücksichtigt wird} \\ m_i \dot{\mathbf{v}}_{c_i} & \text{falls der Einfluss der Gravitation im Anfangswert } \dot{\mathbf{v}}_0 = -\mathbf{g} \text{ berücksichtigt wird} \end{cases} \\ \mathbf{N}_i &= {}^{c_i}\mathbf{I}_i \dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times ({}^{c_i}\mathbf{I}_i \boldsymbol{\omega}_i) \end{aligned}$$

**Schritt NE.3** (für  $i = n, \dots, 1$ ) Berechnung der Kräfte und Drehmomente  $\tau_i$  am Gelenk jedes Roboterglieds (von außen nach innen).

$$\begin{aligned} \mathbf{f}_i &= {}^i\mathbf{R}_{i+1} \mathbf{f}_{i+1} + \mathbf{F}_i \\ \mathbf{n}_i &= {}^i\mathbf{R}_{i+1} \mathbf{n}_{i+1} + ({}^i\mathbf{R}_{i-1} {}^{i-1}\mathbf{r}_i + {}^i\mathbf{r}_{c_i}) \times \mathbf{F}_i + ({}^i\mathbf{R}_{i-1} {}^{i-1}\mathbf{r}_i) \times ({}^i\mathbf{R}_{i+1} \mathbf{f}_{i+1}) + \mathbf{N}_i \\ \tau_i &= \begin{cases} ({}^i\mathbf{e}_{z_{i-1}})^T \mathbf{n}_i & \text{falls } i \text{ Drehgelenk} \\ ({}^i\mathbf{e}_{z_{i-1}})^T \mathbf{f}_i & \text{falls } i \text{ Schubgelenk} \end{cases} \end{aligned}$$

---

### 5.2.2 Bemerkungen und DIR DYN

---

Die inverse Dynamik INV DYN kann immer in der Form

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q})$$

mit der symmetrischen, positiv definiten Massenmatrix  $\mathbf{q} \in \mathbb{R}^{n \times n}$ , dem Vektor der Zentrifugal- und Coriolisanteile  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$  und dem Vektor der Gravitationsanteile  $\mathbf{G}(\mathbf{q})$ .

Die direkte Dynamik DIR DYN kann in Abhängigkeit von den Antriebsmomenten und -kräften  $\boldsymbol{\tau}$  formuliert werden

$$\ddot{\mathbf{q}} = \mathbf{M}(\mathbf{q})^{-1} (\boldsymbol{\tau} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{G}(\mathbf{q}))$$

wobei die inverse der Massenmatrix in der Praxis nie direkt ausgerechnet, sondern stattdessen das lineare Gleichungssystem  $\mathbf{M}\ddot{\mathbf{q}} = \dots$  gelöst wird.

---

### 5.3 Lagrangesche Formulierung der Roboterdynamik

---

Die Lagrangesche Formulierung der Dynamik liefert am Ende die gleichen Bewegungsgleichungen, funktioniert aber grundlegend anders (Energie-basiert).

**Kinetische Energie** Die kinetische Energie des  $i$ -ten Gliedes lautet:

$$K_i(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \left( m_i {}^0\mathbf{J}_{c_i,v}^T {}^0\mathbf{J}_{c_i,v} + {}^0\mathbf{J}_{i,\omega}^T {}^{c_i}\mathbf{I}_i {}^0\mathbf{J}_{i,\omega} \right) \dot{\mathbf{q}}$$

Damit lautet die gesamte kinetische Energie des Manipulators:

$$\begin{aligned} K(\mathbf{q}, \dot{\mathbf{q}}) &= \sum_{i=1}^n K_i(\mathbf{q}, \dot{\mathbf{q}}) \\ &= \frac{1}{2} \dot{\mathbf{q}}^T \left( \sum_{i=1}^n \underbrace{\left( m_i {}^0\mathbf{J}_{c_i,v}^T {}^0\mathbf{J}_{c_i,v} + {}^0\mathbf{J}_{i,\omega}^T {}^{c_i}\mathbf{I}_i {}^0\mathbf{J}_{i,\omega} \right)}_{\mathbf{M}(\mathbf{q})} \right) \dot{\mathbf{q}} \\ &= \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} \end{aligned}$$

Wobei  $\mathbf{M}(\mathbf{q})$  Massenmatrix darstellt.

**Potentielle Energie** Die potentielle Energie des  $i$ -ten Gliedes lautet

$$P_i = -m_i \mathbf{g}^T {}^0\mathbf{r}_{c_i} + P_{\text{Ref},i}$$

mit dem Gravitationsvektor  $\mathbf{g}$  und einer Konstante  $P_{\text{Ref},i}$ , die im späteren Verlauf der Ableitungen wieder verschwindet.

Damit lautet die gesamte potentielle Energie des Manipulators:

$$P(\mathbf{q}) = \sum_{i=1}^n P_i$$

**Lagrangefunktion und Lagrange-Gleichungen** Die Lagrangefunktion ist definiert als:

$$\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = K(\mathbf{q}, \dot{\mathbf{q}}) - P(\mathbf{q})$$

Nach dem Prinzip der kleinsten Wirkung (Hamiltonsches Prinzip) bewegt der Manipulator sich dann entsprechend der  $n$  Bewegungsgleichungen

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = \tau_i, \quad i = 1, \dots, n$$

welche, werden sie symbolisch ausgerechnet, das inverse Dynamikmodell von der Form

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q})$$

liefert.

---

## 5.4 Numerische Aspekte

---

Die Newton-Euler-Formulierung hat, wenn sie numerisch implementiert wird, deutlich weniger Rechenoperationen als die Lagrangesche Formulierung. Wird die inverse Dynamik jedoch symbolisch „von Hand“ ausgerechnet, so ist die Lagrangesche Formulierung geeigneter.

---

### 5.4.1 Modularität

---

---

### 5.4.2 Simulation

---

---

## 5.5 Rekursive Verfahren zur Berechnung der Vorwärtsdynamik

---

---

### 5.5.1 Verfahren mit expliziter Berechnung der Massenmatrix

---

Verfahren 1: Berechnung von  $M$  durch wiederholte Auswertung des Newton-Euler-Verfahrens

Verfahren 2: Ausnutzen der Symmetrie von  $M$

Verfahren 3: Aggregation von Teilmanipulatoren (CRBA)

Vergleich der Verfahren

---

### 5.5.2 Verfahren ohne explizite Berechnung der Massenmatrix

---

---

### 5.5.3 Multibody Systems Library MBSlib

---

Beispiele

---

## 5.6 Geschlossene kinematische Ketten

---

---

## 5.7 Berücksichtigung von Nichtstarrkörpereffekten

---

Bisher wurde, selbst bei der Dynamik, viele Vereinfachungen eingeführt, insbesondere wurden Reibung, Elastizitäten und Loser (Gelenkspiel) nicht berücksichtigt. In diesem Abschnitt sollen diese Effekte (vereinfacht) berücksichtigt werden.

---

### 5.7.1 Reibung

---

Berühren sich zwei Oberflächen, so sind die typischerweise nicht glatt, sodass auch tangential Kräfte wirken (die *Reibung*). In der Regel können die Reibungskräfte und -momente, die in den Gelenken wirken, nicht vernachlässigt werden.

Zur Berücksichtigung von Reibung wird den Bewegungsdifferentialgleichungen INV DYN ein Vektor  $\mathbf{F}$  für Reibungs- und Dämpfungskräfte und -momente hinzugefügt:

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\mathbf{q}, \dot{\mathbf{q}})$$

---

#### Reibungsmodell

---

Ein häufiges, einfaches Reibungsmodell (des  $i$ -ten Gelenks) ist

$$F_i(\mathbf{q}, \dot{\mathbf{q}}) = \underbrace{\gamma_{C,i} \cdot \text{sign } \dot{q}_i}_{\text{Coulombsche Reibung}} + \underbrace{\gamma_{v,i} \cdot \dot{q}_i}_{\text{Viskose Gleitreibung}}$$

mit  $\gamma_{C,i}, \gamma_{v,i} = \text{const.}$

Sobald die Zugkraft  $Z$  einen bestimmten Grenzwert überschreitet, wird die Haftung überwunden und die Oberflächen gleiten übereinander. Diese *Coulombsche Reibung* wird mit  $\gamma_{C,i}$  beschrieben, welches nach Charles Augustine proportional zur Normalkraft  $N_i$  ist (mit dem *Reibungskoeffizient*  $\mu_{R,i}$ ):

$$\gamma_{C,i} = \mu_{R,i} \cdot N_i$$

Der Grenzwert der Haftung  $\gamma_{s,i}$  ist mit dem *Haftungskoeffizienten*  $\mu_{H,i}$  ebenfalls proportional zur Normalkraft:

$$\gamma_{s,i} = \mu_{H,i} \cdot N_i$$

Dabei ist der Haftreibungskoeffizient typischerweise größer als der Reibungskoeffizient, was zu einem ruckartigen Gleiten (Losbrechen) bzw. Festsetzen führt. Das resultierende, klassische Reibungsmodell lautet:

$$F_i(\mathbf{q}, \dot{\mathbf{q}}) = \begin{cases} \gamma_{C,i} \cdot \text{sign } \dot{q}_i + \gamma_{v,i} \cdot \dot{q}_i & \text{falls } \dot{q}_i \neq 0 \\ Z & \text{falls } \dot{q}_i = 0 \text{ und } |Z| \leq \gamma_{s,i} \\ \gamma_{s,i} \cdot \text{sign } Z & \text{sonst} \end{cases}$$

Die *viskose Gleitreibung* wird häufig proportional zur Geschwindigkeit modelliert und berücksichtigt hydromechanische Effekte (z. B. Schmiermittel).

**Verfeinerungen** Das bisherige Modell modelliert einen sprunghaften Übergang zwischen Haft- und Gleitreibung. Dies entspricht jedoch nur annähernd der Realität, wobei Reibung mit zunehmender Geschwindigkeit zunächst abnimmt. Dieser Übergang kann mit der Stribeck-Reibung beschrieben werden (dabei ist  $\dot{q}_{\text{stribek}}$  die Geschwindigkeit, bei der die minimale Reibung auftritt):

$$F_i(\mathbf{q}, \dot{\mathbf{q}}) = \left( \gamma_{C,i} + (\gamma_{s,i} - \gamma_{C,i}) \exp \left\{ - \left| \frac{\dot{q}_i}{\dot{q}_{\text{stribek},i}} \right| \right\} \right) \cdot \text{sign } \dot{q}_i + \gamma_{v,i} \cdot \dot{q}_i$$

### Anmerkungen

- Reibungsmodelle sind immer mit Unsicherheiten behaftet (Last-, Temperatur-, Winkelabhängigkeiten, ...).
- Es gibt zahlreiche unterschiedliche Ansätze zur Modellierung von Reibung.
- Im Newton-Euler-Verfahren werden Reibungskräfte im letzten Schritt von NE.3 eingeführt:

$$\tau_{i,\text{neu}} = \tau_i + F_i(\mathbf{q}, \dot{\mathbf{q}})$$

## 5.7.2 Elastizität

In der Industrie und der klassischen Robotik werden möglichst starre kinematische Ketten zur genauen und schnellen Manipulation schwerer Objekte konstruiert. In Leichtbaurobotern oder in der Mensch-Roboter-Interaktion können elastische Gelenke und Glieder jedoch von Vorteil sein. Zur exakten Positionsregelung ist hier die Betrachtung der Elastizitäten nötig.

### Grundlagen

Ein elastischer Körper verformt sich dabei und Kräfteinwirkung, speichert diese Kraft als potentielle Energie und gibt die bei Rückfederung wieder frei. Zur Charakterisierung wird die sogenannte *Federsteifigkeit* eingeführt (lineare und rotatorisch):

$$c = \frac{\Delta f}{\Delta s} \quad c_\varphi = \frac{\Delta n}{\Delta \varphi}$$

Dabei beschreibt

- $\Delta f$ ,  $\Delta n$  die Änderung der Kraft/des Drehmoments und
- $\Delta s$ ,  $\Delta \varphi$  die Änderung der Auslenkung/des Drehwinkels.

Werden Kraft/Moment und Auslenkung/Drehung gegeneinander aufgetragen, so ergibt sich die *Federkennlinie*. Dabei werden drei Typen an Kennlinien unterschieden:

- Progressive Kennlinie: Die Federsteifigkeit nimmt mit der Zeit zu.
- Degressive Kennlinie: Die Federsteifigkeit nimmt mit der Zeit ab.
- Lineare/Hookesche Kennlinie: Die Federsteifigkeit bleibt konstant (d. h.  $c = \text{const}$ ).

## Elastizitäten in der Robotik

In der Robotik können zwei Typen von Elastizitäten auftreten:

- Elastische Glieder  
Die Deformation ist abhängig von der Gelenkwinkelstellung  $q_i$ , der Gliedlänge  $r$  und der Temperatur  $T$ .
- Elastische Gelenke  
Führt zu einem Unterschied in der Gelenkposition: antriebsseitig  $\theta$ , abtriebsseitig  $\theta_{\text{El}}$ .

In der Regel werden diese Fälle getrennt betrachtet.

Im Ersatzmodell wird jedes Gelenk mit zwei Gelenkvariablen modelliert:

$\theta_i$  Die Position des starren Gelenkantriebs vor der Elastizität (antriebsseitig).

$q_i = \theta_{\text{El},i}$  Die tatsächliche Gelenkposition (abtriebsseitig).

Dadurch verdoppelt sich die Anzahl an Bewegungsgleichungen und es werden  $2n$  Gelenkvariablen zur Beschreibung der Bewegung benötigt.

Die Dynamikgleichungen lassen sich am besten durch eine Modifikation des Lagrange-Formalismus herleiten. Dabei wird die potentielle Energie

$$P(\mathbf{q}, \boldsymbol{\theta}) = P_G(\mathbf{q}) + P_{\text{El}}(\mathbf{q}, \boldsymbol{\theta})$$

durch einen Term  $P_{\text{El}}$  der in Gelenkverformungen gespeicherten Energie ergänzt ( $P_G(\mathbf{q})$  ist das Gravitationspotential, welches offensichtlich nur von den abtriebsseitigen Stellungen abhängt). Die potentielle Energie der Verformungen ist gegeben durch:

$$P_{\text{El}}(\mathbf{q}, \boldsymbol{\theta}) = \frac{1}{2} (\boldsymbol{\theta} - \mathbf{q})^T \mathbf{E}_S (\boldsymbol{\theta} - \mathbf{q})$$

Mit der diagonalen Steifigkeitsmatrix  $\mathbf{E}_S$ . Die kinetische Energie  $K_R(\mathbf{q}, \boldsymbol{\theta}, \dot{\mathbf{q}}, \dot{\boldsymbol{\theta}})$  ist nun abhängig von den an- und abtriebsseitigen Positionen und Geschwindigkeiten. Mit den zusammengefassten verallgemeinerten Gelenkvariablen

$$\mathbf{p} = \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\theta} \end{bmatrix}$$

hat die kinetische Energie die Form

$$K_R(\mathbf{p}, \dot{\mathbf{p}}) = \frac{1}{2} \dot{\mathbf{p}}^T \mathbf{M}_R(\mathbf{p}) \dot{\mathbf{p}}$$

mit der erweiterten Trägheitsmatrix:

$$\mathbf{M}_R(\mathbf{p}) = \begin{bmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{S}(\mathbf{q}, \boldsymbol{\theta}) \\ \mathbf{S}^T(\mathbf{q}, \boldsymbol{\theta}) & \mathbf{J}_M(\boldsymbol{\theta}) \end{bmatrix}$$

Die Teilmatrizen haben jeweils die Dimension  $(n \times n)$ , wobei

- die Matrix  $\mathbf{M}$  der Massenmatrix des starren Roboters in Abhängigkeit von den abtriebsseitigen Position entspricht.
- $\mathbf{J}_M$  ist die Trägheitsmatrix der Motoren und theoretisch abhängig von der Motorposition. Meistens ist die Massenverteilung des Rotors jedoch symmetrisch und der Schwerpunkt liegt auf der Drehachse, wodurch die Matrix diagonal und konstant wird.
- Die Matrix  $\mathbf{S}$  ist die Trägheitsverkopplung zwischen Rotoren und den Gliedern. Sie entsprechen der kinetischen Energie der Rotoren um andere Achsen als der Motorachse. Meistens wird die kinetische Energie jedoch hauptsächlich durch die Drehgeschwindigkeit um die Motorachse verursacht, wodurch  $\mathbf{S} = \mathbf{O}$  gilt, d. h. die Matrix verschwindet.

Dadurch lässt sich die kinetische Energie schreiben als:

$$\begin{aligned} K_R &= K_A + K_M \\ \text{mit} \quad K_A(\mathbf{q}, \dot{\mathbf{q}}) &= \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} \\ K_M(\dot{\boldsymbol{\theta}}) &= \frac{1}{2} \dot{\boldsymbol{\theta}}^T \mathbf{J}_M \dot{\boldsymbol{\theta}} \end{aligned}$$

Wobei  $K_A$  die kinetische Energie in den Armgliedern und  $K_M$  die kinetische Energie in den rotierenden Motoren darstellt. Durch Lösen der Lagrange-Gleichungen ergibt sich ein gekoppeltes Differentialgleichungssystem:

$$\begin{aligned} \boldsymbol{\tau}_{\text{El}} &= \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) \\ \boldsymbol{\tau}_m &= \mathbf{J}_M \cdot \ddot{\boldsymbol{\theta}} + \boldsymbol{\tau}_{\text{El}} \\ \boldsymbol{\tau}_{\text{El}} &= \mathbf{E}_S \cdot (\boldsymbol{\theta} - \mathbf{q}) \end{aligned}$$

---

## Berechnung von INV DYN bei Drehgelenkelastizitäten

---

Zur Berechnung von INV DYN werden die folgenden Schritte ausgeführt:

1. Berechnung der Drehmomente  $\tau_{\text{El}}$  in den Gelenken mit dem starren Modell (z. B. mit Newton-Euler).
2. Berechnung von  $\ddot{\theta}$  durch zweifaches differenzieren und umstellen nach  $\ddot{\theta}$  von ??:

$$\ddot{\theta} = \mathbf{E}_S^{-1} \cdot \ddot{\tau}_{\text{El}} + \ddot{\mathbf{q}}$$

3. Berechnung der Motordrehmomente:

$$\tau_m = \mathbf{J}_M \cdot \ddot{\theta} + \tau_{\text{El}}$$

---

## Elastizitäten in der Biologie

---

### Menschlicher Bewegungsapparat

Gelenkmodelle

Skelettmuskulatur

Muskel-Sehnen-Komplex

### Muskelaktivierungsdynamik

Muskelmodell

Hebelarme

### Weichteilmodelle

### Dynamikmodell

Dynamiksimulation

### Software und Daten

### Einschränkungen

---

## Steuerung und Regelung bei Mensch und Tier

---

### Reafferenzprinzip

---

## 5.8 Spezielle Dynamikmodelle für zweibeinige, humanoide Roboter und deren Stabilitätsregelung

---

Heutige humanoide Roboter bestehen aus starren kinematischen Ketten mit steifen Drehgelenken (meistens sechs bis sieben Drehgelenke pro Bein), sind Kaskadengeregelt.

Dabei beruht die posturale Stabilität auf dem *Zero Moment Point*. Der Begriff von *statisch stabilen Laufen* bezeichnet dabei, dass die Laufbewegung jederzeit gestoppt werden kann, ohne dass der Roboter umfällt. *Dynamische Stabilität* ist hingegen gegeben, wenn der Roboter zwar nicht statisch stabil ist, aber trotzdem nicht umfällt.



---

### 5.8.1 Zero-Moment-Point (ZMP)

---

Es wird ein stabiler (rutschfester) Stand auf einem Bein mit starrer Sole in einem Weltkoordinatensystem so, dass der Roboter in  $x$ -Richtung läuft und  $z$  nach oben zeigt, angenommen. Daraus resultieren Bodenreaktionskräfte und -momente. Diese sind zusammengesetzt aus normalen Kräften

$$\mathbf{f}_{N_i} = [f_{N_i,z}]$$

und tangentialen Kräften

$$\mathbf{f}_{t_i} = \begin{bmatrix} f_{t_i,x} \\ f_{t_i,y} \\ 0 \end{bmatrix}$$

Der *Zero-Moment-Point* ist nun derjenige Punkt  $P$ , in dem die resultierende normale Kraft  $\mathbf{F}_N = \sum_i \mathbf{f}_{N_i}$  wirkt:

$$\mathbf{r}_{\text{ZMP}} := \frac{\sum_i \mathbf{r}_{N_i} f_{N_i,z}}{\sum_i f_{N_i,z}}$$

Dabei ist  $\mathbf{r}_{N_i}$  der Angriffspunkt der Kraft  $\mathbf{f}_{N_i}$ . Durch die tangentialen Kräfte kann  $\mathbf{N}_t$ , der Vektor der Drehmomente am Punkt  $P$ , bestimmt werden:

$$\mathbf{N}_t = \sum_i \mathbf{d}_i \times \mathbf{f}_{t_i} = \begin{bmatrix} * \\ * \\ 0 \end{bmatrix} \times \begin{bmatrix} * \\ * \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ * \end{bmatrix}$$

Somit wirkt am Zero-Moment-Point zwar ein Moment, dieses führt aber nicht zum Kippen.

Der Roboter kann nur dann kippen, wenn sich der ZMP am Rand der konvexen Hülle der Bodenkontaktpunkte liegt. Daher muss der ZMP strikt innerhalb der konvexen Hülle gehalten werden.

---

### 5.8.2 Center of Pressure (CoP)

---

Ein alternatives Stabilitätsmaß ist das *Center of Pressure* (CoP). Dieser ist die resultierende Ersatzkraft aus allen Kräften, die auf die Sole des Fußes wirken und den Bodenreaktionskräften entgegen wirken.

Achtung: ZMP und CoP beschreiben nicht das gleiche! Dies ist nur der Fall bei einem dynamisch ausbalancierten Laufroboter.

---

### 5.8.3 Inverses Pendel und Feder-Masse-Modell

---

Für humanoide Roboter (mit oft mehr als 30 Gelenken) ist das aufstellen des inversen Dynamikmodells häufig zu komplex und damit die Berechnung nicht echtzeitfähig. Daher werden vereinfachte Dynamikmodelle eingesetzt, z. B. ein inverses Pendel.

Die Energien eines seitwärts beschleunigten, inversen Pendels mit Länge  $l$  und Auslenkung  $\theta$  ergibt sich die (um die Nullauslenkung, d. h. aufrechte Haltung) linearisierte Bewegungsgleichung

$$l\ddot{\theta} - g\theta = -\ddot{x}$$

Dieses Modell liefert nur gute Ergebnisse für langsames Gehen. Alternativ kann ein Feder-Masse-Modell mit

- Masse  $m$ ,
- Ruhelänge  $l_0$ ,
- Federkonstante  $k$  und
- momentanem Zustand  $(r, \varphi)$  in Polarkoordinaten

verwendet werden. Die Bewegungsgleichungen lauten dann:

$$\begin{aligned} m\ddot{r} &= k(l_0 - r) + mr\dot{\varphi}^2 - mg \sin \varphi \\ r\ddot{\varphi} &= -2\dot{r}\dot{\varphi} - g \cos \varphi \end{aligned}$$

---

### 5.8.4 Globale Stabilitätsbegriffe

---

---

### 5.8.5 Ausblicke

---

---

#### Capture Steps

---

- Ein *Capture Point* ist ein Punkt, auf dem der Roboter beim Betreten vollständig zum Stillstand kommen kann. Ist dieser Punkt erreichbar, so wird er auch *Capture Step* genannt.
- Capture Steps sind eine effektive Methode, um (harte) externe Stöße abzufangen.

---

## Unebenes Terrain

---

Auf unebenem Terrain gelten die vorherigen Annahmen für den ZMP nicht mehr, weshalb andere Ansätze notwendig sind.

---

## 5.9 Spezielle Dynamikmodelle für zweibeinige, nicht-humanoide Roboter und deren Stabilitätsregelung

---

In diesem Abschnitt werden z. B. hüpfende Roboter betrachtet.

---

### 5.9.1 Hüpfende Roboter mit Teleskop-Beinen

---

Bei einem hüpfendem Roboter wird die absorbierte Energie beim Bodenkontakt durch Kompression zwischen den Sprüngen transferiert.

---

#### Ein Modell dynamischer Stabilität

---

Das Steuerungsproblem für Hüpfroboter kann in drei unabhängige, zeitabhängige Teilprobleme zerlegt werden:

1. Hüpfhöhe
2. Vorwärtsgeschwindigkeit
3. Haltung

Diese Probleme können mit je einer Differentialgleichung modelliert werden, wobei die drei Steuerungen durch einen diskreten Zustandsautomaten für den Hüpfzyklus synchronisiert werden.

Dieser hat fünf diskrete Zustände und Übergänge mit jeweils eigener, kontinuierlicher Systemdynamik (hybride diskret-kontinuierliche Systemdynamik). Dieses „Umschalten“ von kontinuierlichen Systemdynamikmodell für die verschiedenen Phasen einer Gangart ist ein heute gängiges Vorgehen bei laufenden Robotern. Sensoren messen dabei die Zeitpunkte

- des Bodenkontakts,
- wenn das Bein nahe voller Ausdehnung oder
- wenn das Bein nahe voller Kompression ist.

Diese Steuerung kann ebenfalls auf mehrbeinige, hüpfende Roboter übertragen werden.

---

### 5.9.2 Passive Dynamic Walkers

---

Passive Laufmaschinen haben keine Antriebe, kein Regelungssystem und sind rein durch die Gravitationskraft angetrieben (schiefe Ebene). Diese Art der Fortbewegung ist nur begrenzt auf zweibeinige Roboter übertragbar, da weder Bergauf-Gehen, Kurven-Gehen oder Manipulation von Lasten möglich ist.

---

## 6 Antriebssysteme

---

### 6.1 Gebräuchliche Antriebssysteme

---

#### 6.1.1 Hydraulische Antriebe

---

Hydraulische Antriebe nutzen Öl und Druckveränderungen dieses Öls in einer Kammer zur Bewegung.

- **Vorteile:**
  - Können hohe Lasten tragen.
  - Gutes Verhältnis von Leistung zu Gewicht.
  - Durch die Inkompressibilität von Öl kann der Antrieb gut in einer Stellung fixiert werden.
  - Der Antrieb schmiert und kühlt sich selbstständig.
  - Schnelle Reaktionszeit.
  - Sicher in Umgebungen mit brennbaren Gasen (da keine Funken entstehen).
  - Flüssige Bewegungen sind auch bei langsamer Geschwindigkeit möglich.
- **Nachteile:**
  - Teuer.
  - Die Dichtungen müssen gut gewartet werden, sonst entstehen Lecks.
  - Stark limitierte Geschwindigkeiten.
  - Benötigen eine Rückführleitung.
  - Können aufgrund der hohen Leitungsdrücke und -flüsse schlecht klein gebaut werden (Miniaturisierung).
  - Großer Platzbedarf für externe Leistungsverorgung.
  - Das Regel- und Positionierverhalten ist Temperaturabhängig.

Daher gibt es nur wenige mobile hydraulisch aktuierte Roboter (z. B. Atlas und BigDog von Boston Dynamics).

---

#### 6.1.2 Pneumatische Antriebe

---

Pneumatische Antriebe nutzen Pressluft und Veränderungen des Drucks zur Bewegung.

- **Vorteile:**
  - Preiswert.
  - Pressluft ist in der Industrie überall verfügbar.
  - Hohe Geschwindigkeiten sind möglich.
  - Keine Verunreinigung durch auslaufende Flüssigkeiten (wie Öl).
  - Reine Rückführleitungen nötig.
  - Aufgrund der Kompressibilität von Luft gibt der Antrieb auf natürliche Weise nach: vorteilhaft für Interaktionen mit Menschen.
- **Nachteile:**
  - Die Kompressibilität der Luft beschränkt die Regel- und Positionierbarkeit (der Roboter kann nicht gut in einer Position fixiert werden).
  - Die entweichende Luft erzeugt Lärm.
  - Zusätzliches Trocknen/Filtern der Luft kann erforderlich sein.
  - Das hochheben von Lasten und Druckabfälle in der Leitung erschweren die Regelung.
  - Ungeeignet für mobile Anwendung, da ein Kompressor benötigt wird.

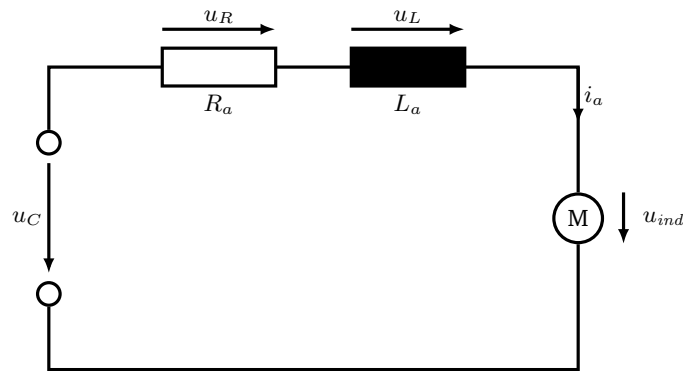


Abbildung 6.1: Stromkreis eines DC-Bürsten-Motors mit dem Widerstand der Spule  $R_a$ , der Induktivität der Spule  $L_a$ , dem Ankerstrom  $i_a$ , der angelegten Spannung  $u_a$  und der induzierten Spannung  $u_{ind}$ .

### 6.1.3 Elektrische Antriebe

- **Vorteile:**

- Schnelligkeit und Genauigkeit.
- Es sind ausgefeilte Regelungsmethoden anwendbar.
- Preiswert.
- Neue Motortypen werden schnell entwickelt, bzw. können schnell entwickelt werden.

- **Nachteile:**

- Typischerweise haben die Motoren hohe Geschwindigkeiten bei niedrigem Drehmoment, d. h. es sind Getriebe notwendig.
- Das Getriebespiel beschränkt die erreichbare Genauigkeit.
- Elektrische Lichtbögen (Funken, Blitze) sind problematisch bei entzündlicher Atmosphäre (z. B. innerhalb von entzündlichen Gasen).
- Im Dauereinsatz ist eine Überhitzung möglich.
- Es sind Bremsen notwendig, um eine Position dauerhaft zu fixieren.

## 6.2 DC-Bürsten-Motoren

Durch das anlegen einer Spannung an den *Bürsten*, die den *Kollektor* berühren, wird eine Spannung induziert. Durch die Rotorwicklung wird ein magnetisches Felds erzeugt, was das magnetische Feld des Permanentmagneten verzerrt. Das hieraus resultierende Drehmoment führt zur Bewegung des Kollektors. Durch das Reiben der Bürsten an dem Kollektor entstehen Funken, die in einer entzündlichen Umgebung zu Bränden führen können.

Abbildung ?? zeigt den elektrischen Stromkreis eines DC-Bürsten-Motors mit allen relevanten elektrischen Größen. Mit der Drehzahlkonstante  $k_v$  sowie der Drehmomentkonstante  $k_t$  und den mechanischen Größen des Motordrehmoments  $\tau_r$  und der Motorgeschwindigkeit  $\dot{\theta}_r$  wird der Stromkreis durch folgende (Differential-) Gleichungen beschrieben:

$$u_C = R_a i_a + L_a \frac{di_a}{dt} + u_{ind} \quad (\text{Ankerstromkreis})$$

$$u_{ind} = \dot{\theta}_r k_v \quad (\text{Induzierte Spannung})$$

$$\tau_r = i_a k_t \quad (\text{Motordrehmoment})$$

## 6.3 Getriebe

Direkte Elektrische Antriebe produzieren hohe Geschwindigkeiten bei geringen Drehmomenten. Daher ist ein *Getriebe* zur Übersetzung notwendig, wenn hohe Drehmomente gewünscht sind (dies ist meistens der Fall, bspw. wenn schwere Lasten gehoben werden sollen). Zu den gebräuchlichsten Getriebe-Bauarten zählen Gewinde, Riemen-, Ketten oder Seilzug-Getriebe sowie Rädergetriebe. Diese werden in den folgenden Abschnitten detaillierter behandelt.

---

Die *Getriebeübersetzung* bezeichnet im Allgemeinen das Verhältnis von Eingangs- zu Austrittsgeschwindigkeit aus dem Getriebe. Dieses ist dabei gleich dem Verhältnis von Ausgangs- zu Eingangsdrehmoment:

$$\text{Getriebeübersetzung} := \frac{\text{Eingangsgeschwindigkeit}}{\text{Ausgangsgeschwindigkeit}} = \frac{\text{Ausgangsdrehmoment}}{\text{Eingangsdrehmoment}}$$

Bei elektrischen Antrieben werden vor allem *Getriebeuntersetzungen* verwendet, die die Ausgangsdrehgeschwindigkeit reduzieren und das Ausgangsdrehmoment erhöhen.

---

### 6.3.1 Gewinde

Ein *Gewinde* funktioniert wie eine „inverse Schraube“, d. h. eine Schraube mit Gewinde wird durch eine Platte bewegt, die sich daraufhin linear bewegt. Dies führt zu einer großen Geschwindigkeitsreduktion und damit zu einer starken Momenterhöhung. Durch die hohe Steifigkeit des Systems sind außerdem hohe Traglasten möglich.

---

### 6.3.2 Riemen-/Seilzug-Getriebe

Bei einem *Riemen-* oder auch *Seilzug-Gewinde* bestehen aus zwei unterschiedlich großen Rädern (z. B. Zahnrädern) mit den Radien  $r_1$  und  $r_2$ , die über einen Riemen verbunden sind (dabei ist  $r_1$  der Eingang, d. h. dieses Rad wird von dem Motor gedreht.). Problematisch sind hier die auftretenden Elastizitäten im Riemen. Zur Vermeidung dieser muss der Riemen durchgehend gespannt bleiben.

Die Getriebeübersetzung gleich dem Quotient der beiden Radien:

$$\text{Getriebeübersetzung} = \frac{r_2}{r_1}$$

---

### 6.3.3 Rädergetriebe

Bei Zahnrädern treten einige elementare Probleme auf:

- (Verdreh-) Spiel (*Backlash*) durch ungenaue Verzahnung (d. h. die Räder können ein bisschen hin und her gewackelt werden).
- Reibung zwischen den Rädern.
- Durch eine enge Verzahnung kann das Spiel verringert werden, dies führt aber zu einer höheren Reibung.

Gängige Typen von Zahnradgetrieben sind

- Stirnradgetriebe,
- Planetengetriebe und
- Harmonic Drive Getriebe.

Diese werden in den kommenden Abschnitten näher betrachtet.

---

#### Stirnradgetriebe

Ein *Stirnradgetriebe* besteht aus ein oder mehreren Paarungen von Zahnrädern, wobei das erste Zahnrad direkt auf der Motorwelle montiert ist. Diese Getriebe sind preisgünstig und eignen sich für kleine Drehmomente.

---

#### Planetengetriebe

Ein *Planetengetriebe* besteht aus einem zentralen *Sonnenrad*, welches von mehreren *Planetenrädern*, die auf einem Planetenradträger montiert sind, umkreist wird. Um diese liegt wiederum das *Hohlrad*, welches mit allen Planetenrädern verzahnt wird. Das Getriebe hat drei Wellen (Sonnenrad, Planetenradträger, Hohlrad), wodurch sechs Übersetzungsmöglichkeiten existieren. Die üblichsten sind:

1. Sonnenrad angetrieben, Hohlrad festgehalten, Planetenradträger als (langsamere) Abtriebswelle
2. Sonnenrad angetrieben, Planetenradträger festgehalten, Hohlrad als (langsamere) Abtriebswelle (rückwärts)
3. Hohlrad angetrieben, Sonnenrad festgehalten, Planetenrad als (langsamere) Abtriebswelle
4. Sonnenrad und Hohlrad mit gleicher Drehzahl angetrieben, Planetenradträger als (gleich schnelle) Abtriebswelle

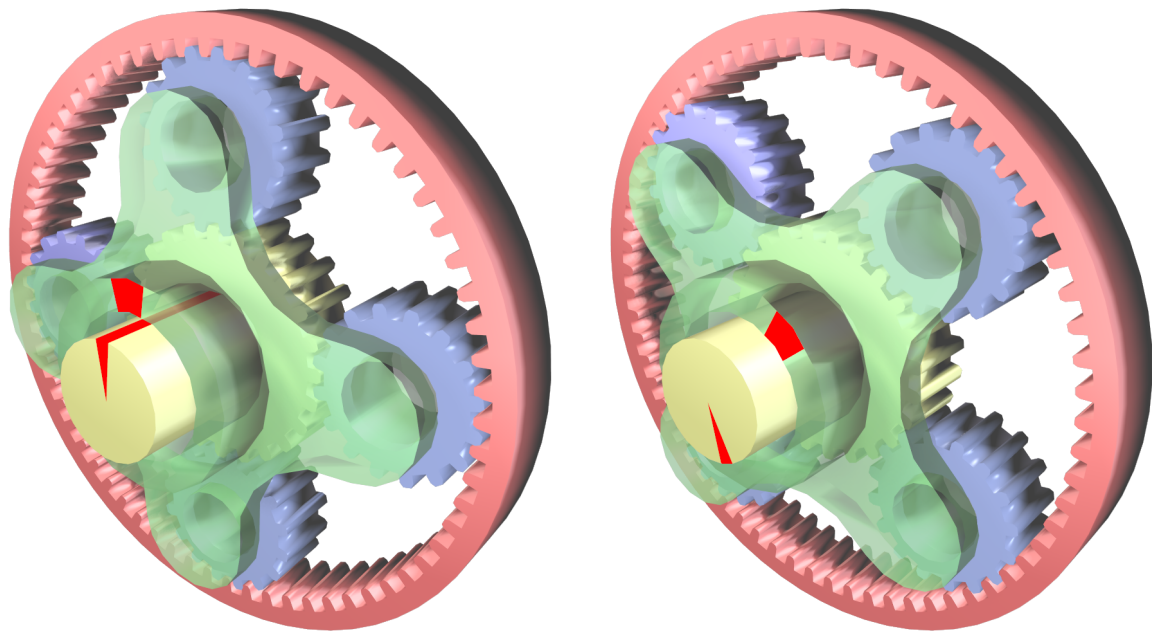


Abbildung 6.2: Planetengetriebe mit Sonnenrad (gelb), Planetenrädern (blau), Planetenradträger (grün) sowie Hohlrad (rot). Das rechte Bild zeigt die relative Verschiebung von Sonnenrad und Planetenradträger, nachdem letzterer um  $45^\circ$  rotiert wurde.

Quelle: [https://commons.wikimedia.org/wiki/File:Epicyclic\\_gear\\_ratios.png](https://commons.wikimedia.org/wiki/File:Epicyclic_gear_ratios.png)

---

### Harmonic Drive Getriebe

---

Ein *Harmonic Drive Getriebe* besteht aus:

- *Elliptischer Wave Generator*  
Elliptische Stahlscheibe mit zentrischer Nabe sowie aufgezogenem elliptisch Verformbaren Kugellager.
- *Flexspline*  
Zylindrische, verformbare Stahlbüchse mit Außenverzahnung.
- *Circular Spline*  
Zylindrischer Ring mit Innenverzahnung.

Dabei wird der elliptische Wave Generator angetrieben und verformt über das Kugellager den Flexspline. Dieser befindet sich gegenüber der großen Ellipsenachse mit dem Circular Spline im Eingriff. Drehen des Wave Generators führt zu einer Verlagerung der großen Ellipsenachse und damit zu einer Verlagerung des Zahneingriffsbereichs von Flexspline und Circular Spline. Eine halbe Umdrehung des Wave Generators führt dann zu einer Relativbewegung zwischen Flexspline und Circular Spline um einen Zahn.

- **Vorteile:**
  - Spielfreiheit (d. h. es existiert kein Spiel).
  - Sehr genaue Positionier- und Wiederholgenauigkeit
  - Kompakte Bauweise.
  - Hohe Drehmomentkapazität.
  - Hoher Wirkungsgrad.
  - Hohe Torsionssteifigkeit (d. h. das Getriebe ist sehr resistent gegenüber Verformungen).
  - Hohe Zuverlässigkeit und lange Lebensdauer.
- **Nachteile:**
  - Teuer.
  - War bislang nicht beliebig klein realisierbar, neue Entwicklungen ermöglichen aber auch kleine Varianten.

---

## 6.4 Alternative und elastische Antriebskonzepte

---

### 6.4.1 Beine

---

### 6.4.2 Neue Materialien

---

### 6.4.3 Compliant Robot Actuation

---

### 6.4.4 Elastische Antriebskonzepte

---

Die Grundidee ist die Kombination elektrischer Antriebe mit mechanischer (potentiell verstellbarer) Elastizität.

- Seriell-Elastische Antriebe (SEA):
  - **Stärken:**
    - \* Energiespeicherung reduziert die benötigte Leistungsspitze des Motors und den Energieverbrauch.
    - \* Die Entkopplung von Antrieb und Abtrieb liefert Stoßkompensation.
    - \* Sicherheitspotential.
  - **Schwächen:**
    - \* Erhöhte Masse des Antriebssystems.
    - \* Erhöhte Komplexität der mechanischen Konstruktion.
    - \* Erhöhte Komplexität der Regelung.
- Parallel-Elastische Antriebe (PAE):
  - **Stärken:**
    - \* Energiespeicherung reduziert die benötigte Leistungsspitze des Motors und den Energieverbrauch, allerdings potentiell auch das Motordrehmoment bei SPitzen.
  - **Schwächen:**
    - \* Erhöhte Masse des Antriebssystems.
    - \* Erhöhte Komplexität der mechanischen Konstruktion.
    - \* Erhöhte Komplexität der Regelung.

---

### Variable Stiffness Actuator

---

### Variable Impedance Actuators

---

### 6.4.5 Vom Muskel-Skelett-Apparat inspirierte Roboter

---

---

## 7 Sensoren

---

Ein *Sensor* ist ein technisches System, welches eine physikalische (analoge) Größe (z. B. Position) und gegebenenfalls deren Änderung (z. B. Geschwindigkeit) in geeignete (unsicherheitsbehaftete) elektronische (digitale) Signale transformiert.

Ein *internen Sensor* („proprioceptive sensor“) erfasst dabei die Roboter-internen Zustände, wobei ein *externer Sensor* („exteroceptive sensor“) Informationen über den Zustand der Umwelt sammelt.

---

### 7.1 Interne Sensoren

---

Interne Sensoren können grob in die Kategorien

- Positionssensoren
- Geschwindigkeitssensoren
- Beschleunigungssensoren
- Inertial Navigation System (INS)
- Kraft-Momenten-Sensoren

unterteilt werden.

---

#### 7.1.1 Positionssensoren

---

---

##### Potentiometer

---

Ein *Potentiometer* besteht aus einem *Widerstandselement* mit darüber verlaufender Spannung und einem gleitendem Kontakt (*Wischer*, „wiper“), der sich über das Widerstandselement bewegt. Dabei kann sich entweder der Wischer oder das Widerstandselement bewegen. Abbildung ?? zeigt ein lineares und ein rotatorisches Potentiometer.

Die Spannung im Kontakt ist dann abhängig von der Position auf dem Widerstandselement und damit proportional zur Gelenkposition, womit die Position gemessen werden kann. Die Genauigkeit liegt (unter der Voraussetzung einer stabilen Spannungsquelle) bei ca. 0.5 %.

---

##### Optische Codierer

---

Auf einer Scheibe wird ein kodierte Muster angebracht, welches den Strahl einer Lichtquelle periodisch unterbricht, wobei diese Quelle auf einen Photodetektor ausgerichtet ist. Durch Messung der Unterbrechung kann die Position gemessen werden. Dabei werden zwei Arten von optischen Kodierern unterschieden:

- Inkrementelle Codierer
- Absolute Codierer

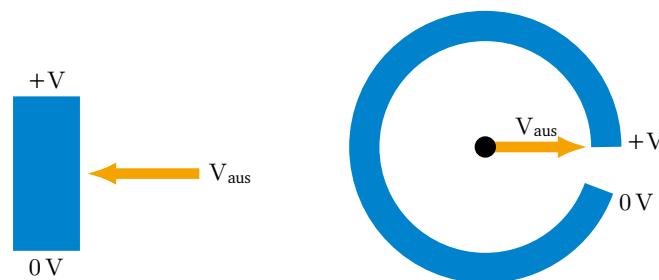


Abbildung 7.1: Lineares (links) und rotatorisches (rechts) Potentiometer mit Widerstandselement (blau) und Kontakt (orange).



---

**Inkrementell** Die Empfangseinheit zählt die Inkremente (die durch die periodischen Unterbrechungen erzeugt werden) und zählt diese (daher der Name *inkrementeller Codierer*). Daraus kann anschließend die Position berechnet werden. Ein großer Nachteil dieser Methode ist, dass die Startposition unbekannt ist, d. h. der Zähler muss zunächst kalibriert werden.

Durch mehrspurige Codierer können durch die Verschiebung der Inkremente mehr Daten gemessen werden:

- 1-spuriger Codierer: Winkel
- 2-spuriger Codierer: Winkel und Richtung
- 3-spuriger Codierer: Winkel, Richtung, Anfang/Ende

**Absolut** Im Gegensatz zum inkrementellen wird beim *absoluten Codierer* jeder Achsenposition ein individuelles Wortmuster zugeteilt, welches anschließend gemessen wird. Dadurch ist ein direktes Ablesen der Gelenkposition möglich und somit keine Kalibrierung notwendig. Allerdings ist die Konstruktion sehr aufwendig.

Mit dem Gray-Code unterscheidet sich der Code an jeder Stelle nur um ein Bit, wodurch große Fehler vermieden werden.

---

## Resolver

Resolver sind robust, störungssicher und haben eine hohe Lebensdauer, weshalb sie oft bei Industrierobotern eingesetzt werden.

---

### 7.1.2 Geschwindigkeitssensoren

Geschwindigkeitssensoren lassen sich bspw. durch Differentiation und Filterung aus mit hoher Frequenz abgetasteten Positionswerten erstellen (d. h. die Geschwindigkeit wird indirekt gemessen).

---

### 7.1.3 Beschleunigungssensoren (Silizium-Beschleunigungssensor)

In einem *Silizium-Beschleunigungssensor* befindet sich eine träge Masse in einem „Siliziumbad“, wobei Auslenkungen der Aufhängung die mechanischen Spannungen ändern, wodurch der piezoresistive Widerstandswert geändert wird. Dieser Wert ist messtechnisch erfassbar, wodurch Beschleunigungen erkannt werden können. Allerdings ist eine Erfassung von Richtung und Betrag der Gravitation nötig, um diese herausrechnen zu können. Durch die Verwendung von drei 1D-Sensoren lässt sich ein 3D-Beschleunigungssensor konstruieren (indem diese orthogonal zueinander angebracht werden).

---

### 7.1.4 Inertial Navigation System (INS)

Die Aufgabe eines *Inertial Navigation Systems* (INS) ist die Bestimmung der Orientierung relativ zu einem Inertialsystem. Dazu lassen sich bspw. folgende Sensoren einsetzen:

- Kompass
- Lagesensor
  - Hierbei wird eine elektrisch leitfähige Flüssigkeit in einem versiegeltem Glas mit drei Elektroden platziert.
  - Eine Lageänderung verursacht eine ungleiche Flüssigkeitsverteilung und damit ungleiche Widerstände relativ zum Winkel.
  - Daraus kann anschließend die Position berechnet werden.
- Gyroskop (mechanisch oder mikromechanisch)

---

## Mechanisches Gyroskop

Ein *Gyroskop* ist ein schnell rotierender, semi-kardanisch aufgehängter Kreisel. Durch die Erdrotation wird ein Drehmoment in Richtung des Medians verursacht, wodurch eine Richtungsanzeige möglich wird.

Angewendet werden Gyroskope an vielen Stellen:

- Linienflugzeug (ca. 12 Gyroskope)
- Raumstation MIR (11 Gyroskope zur Orientierungshaltung zur Sonne)
- Häufig werden 3D-Beschleunigungssensoren für die lineare Beschleunigung und Gyroskope für die 3D-Winkelgeschwindigkeiten verwendet. Die Integration über die Geschwindigkeiten ergibt die zurückgelegte Bewegung (*Odometrie*). Durch Filtern der Messwerte können Fehler und Drifts kompensiert werden.
- Ein normales Flugzeug-Gyroskop hat pro Betriebsstunde einen Drift von ca. 1.85 km, „High-End“-Gyroskope kommen auf einen Drift von weniger als 0.1 % der zurückgelegten Distanz.

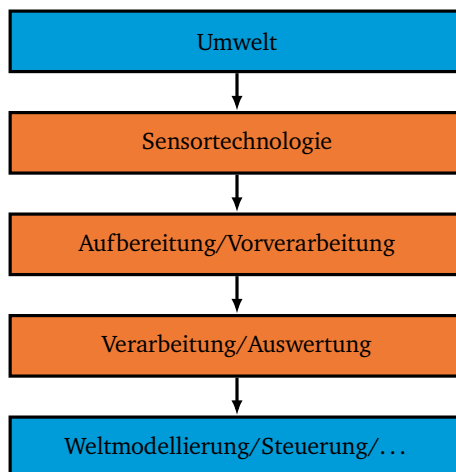


Abbildung 7.2: Typische Struktur eines Sensorsystems, wobei die blau hinterlegten Schritte (größtenteils) analog und die orange hinterlegten Schritte (größtenteils) digital ablaufen.

---

### Mikromechanische Gyroskope

---

Bei mikromechanischen Gyroskopen wird die Rotation durch schwingende, mechanische Elemente registriert. Sie werden z. B. als „Stimmgabel“ realisiert:

- Zwei Zinken mit unterschiedlichen, festen Schwingfrequenzen.
- Bei einer Rotation verursacht die Corioliskraft eine differentielle, sinusförmige Kraft orthogonal zur Hauptschwingung in jedem Zinken.
- Durch die differentielle Verbiegungen der Zinken oder die Torsionsschwingung am Stamm der Stimmgabel kann diese Kraft detektiert werden.
- Die Anregung der Resonanzfrequenz der Zinken kann elektrostatisch, elektromagnetisch oder piezoresistiv erfolgen
- Die durch die Corioliskraft verursachten Schwingungen können kapazitiv, piezoresistiv oder piezoelektrisch detektiert werden.
- Eine optische Detektion ist auch möglich, ist i. A. allerdings sehr teuer.

---

### 7.1.5 Kraft-Momenten-Sensoren

---

Zeil von *Kraft-Momenten-Sensoren* ist die Messung der Kräfte und Momente zwischen Effektor und Objekt. Oft wird hierfür eine sogenannte *Kraftmessdose* verwendet, die eine Kombination aus mehreren Messapparaturen darstellt. Typische Apparaturen sind hierbei:

- Dehnungsmessstreifen (DMS)
- Piezokristalle
- Optische Effekte

Die Dose wird dabei zwischen Effektor und Roboterhand und -fuß angebracht.

Oft haben Kraft-Momenten-Sensoren eine Speichenradform, wobei auf den Speichen/Stegen Dehnungsmessstreifen angebracht sind. Wirkt eine Kraft auf die Dose, so ändert sich die Länge der Stege. Auf den Dehnungsmessstreifen ist eine dünne Metallfolie in Matrixform angebracht, die bei Änderung der Länge ihren Widerstand ändert. Durch Messung dieser Änderung kann die Krafteinwirkung gemessen werden.

---

## 7.2 Externe und intelligente Sensoren

---

Abbildung ?? zeigt die typische Struktur eines (externen) Sensorsystems.

Externe Sensoren können grob in die Kategorien

- Taktile Sensoren
- Näherungssensoren
- Abstandssensoren
- Positionssensoren
- Visuelle Sensoren

unterteilt werden.

---

### 7.2.1 Abstandssensoren

---

*Abstandssensoren* messen den Abstand zwischen dem Sensor und einem Gegenstand. Diese Sensoren sind geeignet zur Erfassung von geometrischen Umweltinformation. Außerdem haben sie eine größere Reichweite als Näherungssensoren.

Es werden drei grundlegende Typen von Abstandssensoren unterschieden:

- Optisch (Nutzung von sichtbaren, elektromagnetischen Wellen)
- Radar (elektromagnetische Wellen mit sehr kurzen Längen)
- Akustisch (mechanische Wellen)

---

#### Akustische Abstandssensoren

---

*Schall* ist ein an Materie gebundener Energie- und Impulstransport, wobei eine *Schallwelle* die wellenförmige Ausbreitung der periodischen Anregung eines Übertragungsmediums beschreibt. Der hörbare Schall liegt dabei zwischen 16 Hz und 16 kHz, wobei alles darunter als *Infraschall* und alles darüber als *Ultraschall* bezeichnet wird.

In den 1980er und 1990er sind Ultraschallsensoren die wichtigsten Sensoren zur Erfassung der Umwelt. Seit Ende der 1990 werden jedoch vermehrt Laserscanner kombiniert mit Farbkameras eingesetzt. Im Nahbereich sind Ultraschallsensoren jedoch auch noch heute nützlich (Kollisionsdetektion, bzw.-vermeidung), sowie in Bereichen wo weder Laserscanner noch Kameras einsetzbar sind (z. B. in Gebäuden mit Rauch).

**Ultraschall** Die Messung von Abständen mittels Ultraschall heißt *sonar* („sound navigation and ranging“). In der Natur wird diese Art der Navigation häufig eingesetzt, z. B. von Delfinen und Fledermäusen.

Zur Erzeugung von Ultraschall wird eine schwingende Folie eingesetzt, die im Sende-Modus durch die elektrostatische Kraft, ausgelöst von einer geladenen Aluplatte, vibriert. Im Empfangs-Modus bewegt der Klangdruck die Folie, wodurch sich die Kapazität des Kondensators ändern (elektrostatisches Mikrofon).

Im Normalbetrieb sendet ein Sensor 1 ms bis 1.2 ms langes „Zwitschern“ pro 200 ms. Dieses Zwitschern wird aufgrund der unterschiedlichen Materialbeschaffenheit und deren unterschiedlichen Reflexionseigenschaften in unterschiedlichen Frequenzen gesendet.

#### Messsituationen

- Im Idealzustand ist die Achse der Schallkeule orthogonal zu einem flachen Objekt. In diesem Fall wird ein genauer Abstand gemessen.
- Befindet sich ein kleines Objekt vor der Wand, so wird der Abstand weiterhin genau gemessen, die Querposition ist jedoch ungenau.
- Steht der Sender rotiert zur Wand, so ist der gemessene Abstand vom Rand der Schallkeule zu kurz (Unterschätzung). Die Unsicherheit des Abstands ist eine Funktion des Rotationswinkels und des Keulenöffnungswinkels.
- Ist die Rotation größer als der Keulenöffnungswinkel, so ist die Wand unsicherbar.
- Eine Kante ist ebenfalls unsichtbar, solange die Ecke nicht ein wenig abgerundet ist.
- Mehrfachreflexionen (wie in einer Ecke) bewirkt eine Starke Überschätzung des Abstands.

**Abhilfe der Schwierigkeiten** Bei nur einem Empfänger darf die Umweltszene nur ein Objekt innerhalb der Schallkeule enthalten der reflektierende Objektteil muss orthogonal zur Strahlrichtung stehen.

Gebräuchliche Abhilfen sind die Verwendung von mehreren Ultraschallsensoren und algorithmische Verfahren wie Peilverfahren: Die Position wird aus den Laufzeitunterschieden, z. B. mit Triangulationsverfahren (ähnlich wie bei GPS), berechnet.

---

**Natur** Fledermäuse haben sehr unterschiedliche Ultraschalle (enge Schallkeulen, weite Schallkeulen, Zwitschern mit fester/veränderlicher Frequenz, ...). Der aktuelle Stand der Ultraschallerzeugung ist vergleichbar mit dem Ultraschall von Fledermäusen, allerdings liegen die technischen Fähigkeiten zur Verarbeitung noch weiter hinter jenen von Fledermäusen.

---

### Optische Abstandssensoren

---

Der Aufbau von optischen Abstandssensoren besteht aus einem Emitter (LED, Laserdioden) und einem Empfänger (Fototransistor). Optische Sensoren liefern sehr genaue Entfernungsmessungen und sind stabil gegen viele Fremdeinflüsse.

Es gibt verschiedene Messverfahren:

- Ermittlung der Flugzeit (Laufzeit)
- aktive Triangulation
- Interferometrie (Phasenverschiebung)
- Stereoskopie

**Laufzeitermittlung** Bei der *Laufzeitermittlung* wird die Hin- und Rücklaufzeit eines Impulses (z. B. Lichtimpuls) gemessen. Wurde zwischen Aussenden und Empfangen die Zeit  $\Delta t$  gemessen, so berechnet sich die Entfernung  $d$  wie folgt:

$$d = \frac{c \cdot \Delta t}{2}$$

Es muss die Hälfte genommen werden, da der Lichtstrahl zum Objekt hin und zurück verläuft.

Typische Probleme sind die Absorption des Strahls vom Objekt, Wegreflexion (z. B. bei einem Spiegel) sowie Mehrfachreflexion (wodurch der Weg verlängert wird).

**2D-Laserscanner** In einem *Laserscanner* wird ein gepulster Laserstrahl, abgelenkt durch einen beweglichen Spiegel, ausgesandt. Durch diesen Drehspiegel wird die Umgebung fächerförmig abgetastet. Trifft der Impuls auf ein Objekt, so wird er reflektiert und die Entfernung kann mittels Laufzeitermittlung berechnet werden. Durch die Abfolge der empfangenen Impulse kann die Entfernung, Lage und Kontur eines Objekts berechnet werden.

Bei der *Kartografie* werden alle gescannten Szenen (gescannt mit einem mobilen System) zu einer Karte zusammengefasst.

---

### 7.2.2 Visuelle Sensoren

---

- Visuelle Wahrnehmung („visual perception“)
  - Aufnahme und Verarbeitung von visuellen Reizen
  - Auge und Gehirn extrahieren relevante Informationen
  - Erkennen von Elementen und deren Interpretation und Abgleich mit Erinnerungen
- Maschinelles Sehen („machine vision“)
  - Digitale Bildverarbeitung und Mustererkennung
  - Eingesetzt z. B. bei Prüfverfahren in der industriellen Fertigung
- Bildverstehen („computer vision“)
  - Extraktion komplexer Interpretationen und Schlüsse aus 2D-Abbildungen realer 3D-Szenen
  - Viele langfristig orientierte Grundlagenforschungsfragen
- Robotersehen („robot vision“)
  - Maschinelles Sehen und Bildverstehen unter den Echtzeitbedingungen eines Roboters
  - Begrenzte Rechenleistung und schnelle Reaktionszeiten

Voraussetzungen für maschinelles Sehen sind:

1. „Technisches Auge“, d. h. ein Kamerachip mit Optik
2. „Intelligente“, bildverarbeitende Algorithmen
3. Ausreichende Strukturierung (Material, Merkmale der relevanten Objekte, ...) der Szene
4. Ausreichende Beleuchtung (Helligkeit, Lampen-Lichtspektrum, Reflexionen, ...) der Szene

Je nach Aufgabe ist eine Abstimmung dieser vier Voraussetzungen nötig, d. h. es gibt keine allgemeine Lösung.

Die klassische Hierarchie der Bildverarbeitungsoperationen ist in Abbildung ?? aufgezeigt.

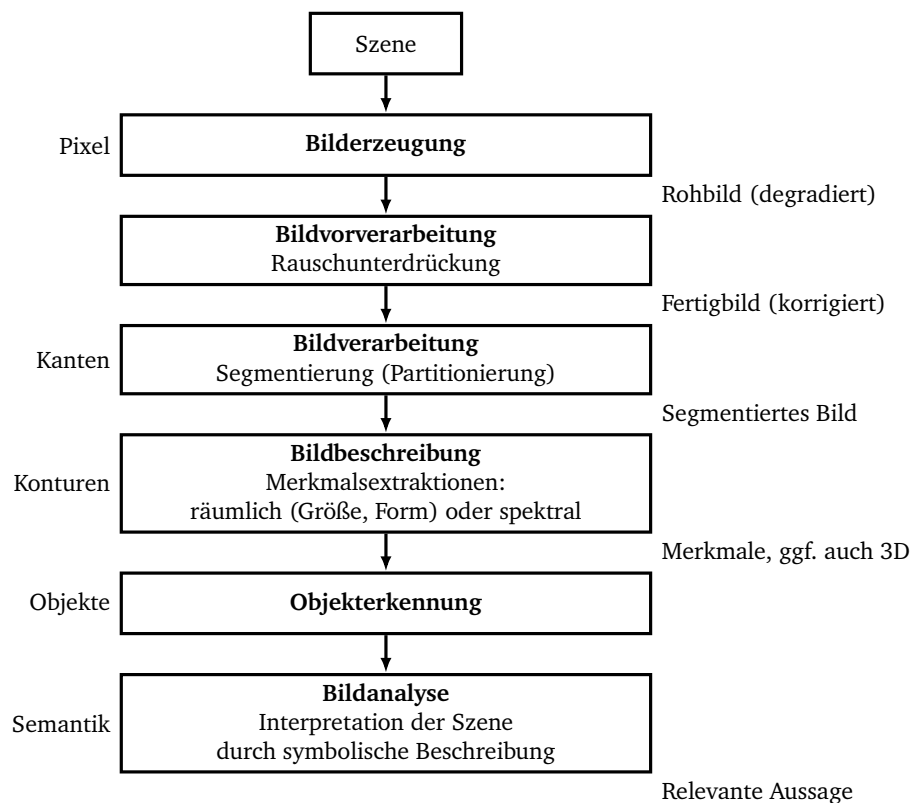


Abbildung 7.3: Klassische Hierarchie der Bildverarbeitung. Dabei findet alles bis zur Bildverarbeitung numerisch und alles ab der Bildbeschreibung symbolisch statt.

Ort	Illumination $i$
Sonne	10 000 cd
Wolken	1000 cd
Arbeitsplatz	100 cd

Tabelle 7.1: Beispielhafte Werte der Illumination  $i$ .

Material	Reflexion $r$
Schwarzer Samt	0.01
Edelstahl	0.65
Weißer Wand	0.80
Spiegel	0.90
Schnee	0.93

Tabelle 7.2: Beispielhafte Werte der Reflexion  $r$ .

## Bilderzeugung

Die *Kameramatrix* beschreibt die Projektionsabbildung von 3D-Punkten durch eine Lochkamera auf 2D-Punkte der Bildebene. Dadurch können spezifische Verzerrungseffekte der Kameralinse berücksichtigt werden.

Zur Bilderzeugung gibt es mehrere grundlegende Methoden von Bildwandlersystemen (Kamerachip):

- Binäre fotoelektrische Zellen  
Der Chip besteht aus einzelnen Fotodioden, wobei die Bildpunkte entweder 1 oder 0 wahrnehmen (hell oder dunkel).
- Zeilen von Fotodioden (Linearkamera)  
Die fotoelektrischen Zellen sind entlang einer Geraden angeordnet.
- Felder von Fotoelementen (Matrixkamera)  
Die Zellen sind zweidimensional angeordnet.
- CCD-Fernsehkameras  
Fernsehnorm, vorgeschaltete Kameraoptik, ...

## Grundlagen

Die *Lichtintensitätsfunktion*  $f^*$ , die für einen Pixel  $(x, y)$  die Lichtintensität  $[f^*] = \text{cd}$  angibt, kann in ein Produkt

$$f^*(x, y) = i(x, y) \cdot r(x, y)$$

aufgespalten werden mit der Illuminationsfunktion  $i(x, y)$  und der Reflexionsfunktion  $0 < r(x, y) < 1$ . Dabei hat die Illuminationsfunktion die Einheit  $[i] = \text{cd}$  (Candela) und die Reflexion ist einheitenlos. Beispielhafte Werte für die Illumination  $i$  und die Reflexion  $r$  sind in den Tabellen ?? und ?? gegeben.

Die Absorptionsfunktion

$$a(x, y) := 1 - r(x, y)$$

beschreibt, wie viel Licht an einem Pixel  $(x, y)$  absorbiert wird/wurde.

Zusammengefasst ergibt sich die *Bildmatrix*  $f = (f_{i,j})$

$$f = \begin{bmatrix} f_{0,0} & f_{0,1} & \cdots & f_{0,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ f_{M-1,0} & f_{M-1,1} & \cdots & f_{M-1,N-1} \end{bmatrix}$$

mit den Elementen als Werten der Lichtintensitätsfunktion. Typische Formate  $M \times N$  sind

- PAL:  $N = 768, \quad M = 576 \quad (576i50)$
- NTSC:  $N = 640, \quad M = 480 \quad (480i60)$
- HDTV:  $N = 1280, \quad M = 720 \quad (720p50, \text{WXGA})$
- HDTV:  $N = 1920, \quad M = 1080 \quad (1080i50, \text{Full HD})$