

# Robotik - Zusammenfassung

## I. Einführung

**Definition:** Automat

griechisch: mit eigenem Willen

**Definition:** Roboter

tschäfisch: Arbeit/Arbeiter

### Asimov Roboter Gesetze

1. Ein Roboter darf keine Menschen verletzen oder es zulassen, dass Menschen durch ihn zu Schaden kommen
2. Ein Roboter muss den Befehlen eines Menschen folge leisten, außer das 1. Gesetz wird dadurch verletzt
3. Ein Roboter muss seine Existenz wahren, solange dadurch das erste und zweite Gesetz nicht verletzt werden

## Robotik-Trends

### Klassische Robotik (Mitte der 1970er)

- Exakte Modelle
- keine Wahrnehmung der Umwelt

### Reaktive Robotik (Mitte der 1980er)

- keine Modelle
- verlässt sich stark auf eine gute Sensorik

### Hybride Ansätze (seit ca. 1990)

- Modellbasiert auf höheren Ebenen
- Reaktiv auf unteren Ebenen

### Probabilistische Robotik (seit Mitte der 1990er)

- Nahtlose Integration von Bewegung und Wahrnehmung
- Ungenaue Modelle, ungenaue Sensoren

### Kognitive Robotik (seit ca. 2000)

- Höhere kognitive Funktionen
- komplexe Umgebung

## Bestandteile von Robotern

- Sensoren
- Aktuatoren
- Steuerung

## 2. Sensoren

### Ultrasonar - Abstandssensoren

#### Anwendung:

- Abstandsmessung
- Hindernisvermeidung (kurze Reichweite)
- Kartierung der Umgebung (lange Reichweite)

#### Funktion:

- Akustischer Ping wird erzeugt ( $> 20 \text{ kHz}$ )  
→ Reflexion durch Objekte
- Berechnung der Entfernung durch Laufzeitmessung und Schallgeschwindigkeit
- Schallgeschwindigkeit: Luft:  
 $0^\circ\text{C}: 330 \text{ m/s}$   
 $20^\circ\text{C}: 343 \text{ m/s}$   
 $40^\circ\text{C}: 355 \text{ m/s}$

#### Eigenschaften:

- Ausbreitungsgeschwindigkeit hängt vom Medium und Temperatur ab
- Schallwellen können an bzw. in Objekt vom Sender weg reflektiert werden
- Minimale und maximale Distanz

### Ultrasseigenschaften

#### Vorteile:

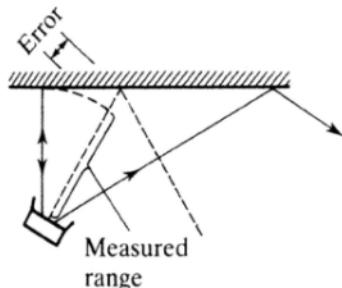
- große, rau Flächen werden gut erkannt
- optisch transparente Flächen können erkannt werden

#### Nachteile:

- glatte, kleine Flächen streuen oder reflektieren schlecht  
→ kleinere Wellenlängen erkennen solche Flächen besser

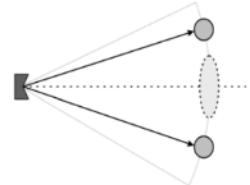
#### Problem: Kurzsichtigkeit

Bei diagonalem Auftreffen des Schalls auf eine Fläche entstehen unterschiedliche Laufzeiten. Da die kürzeste Laufzeit gemessen wird, erscheint das Objekt näher als es im Zentrum ist  
→ größerer Kegel  $\Rightarrow$  größerer möglicher Fehler



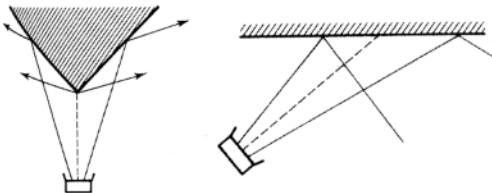
## Problem: Virtuelle Hindernisse

Mehrere kleine Hindernisse werden als ein großes wahrgenommen  
→ je größer der Schallkegel, desto größer müssen Lücken sein



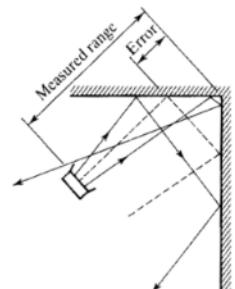
## Problem: Totalreflektion

Bei ungünstigen Auftreffwinkeln wird der Schall weg vom Sensor reflektiert



## Problem: Mehrfachreflektion

Schallwellen werden mehrfach reflektiert bis sie zurück zum Sender reflektiert werden. Objekte erscheinen weiter weg



## Infrarot-Abstandssensoren

### Anwendung

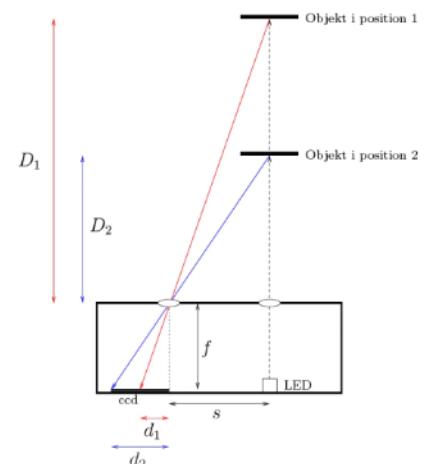
- Erkennung von Anwesenheit von Objekten und eventuelle Entfernungsmessung
- Sanitäranlagen

### Funktion

- Messung mit Infrarot-Strahl
- Triangulation der Entfernung

### Eigenschaften

- nicht-lineare Kennlinie ermöglicht verschiedene Messbereiche



## Laser-Scanner (meist rotierend)

### Anwendung

- Sicherheitsabschaltung, Lokalisierung, Kartierung

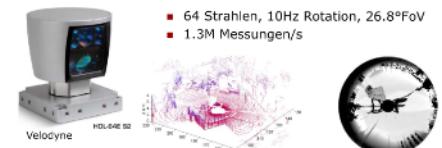
### Funktion

- Rotierender Sensor
- Laser tastet Umgebung rasterartig ab

### Eigenschaft

- Bewegungsfreiheit des Sensors ist entscheidend
- Rotation charakterisierend (2D oder 3D)

2D:



3D:



22

## Time-of-Flight-Kamera

### Anwendung:

- Abstandsmessung



### Funktion

- Umgebung wird durch Lichtpulse angelichtet
- für jeden Bildpunkt wird die Laufzeit gemessen



### Eigenschaften

- breiter Messbereich
- hohe Messfrequenz

### Beispiel: Mesa SR4000

- Messung der Phasenverschiebung modulierten Lichts
- 176 (h) x 144 (v) QCIF
- 43.6° x 34.6° Sichtbereich
- 0.3 to 5.0 m, +/-1cm Genauigkeit
- Bis zu 54 fps



25

## Stereo-Kameras

### Anwendung

- Tiefe erkennen

### Funktion

- Bildaufnahme aus 2 Blickwinkeln
- Finden von Korrespondenzen
- Disparität lässt auf Tiefe schließen

## CCD/Cmos Farbkameras

### Anwendung

- Objekterkennung

### Funktion

- Farberkennung durch Farbfilter im Bayer-Muster
- Farbinterpolation erforderlich

## Thermalkameras

### Anwendung

- Personen detektieren

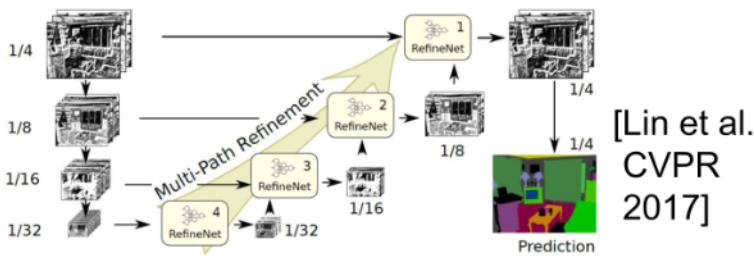
### Funktion

- Temperatur wird durch langwelliges Infrarot gemessen

## Semantische Segmentierung

Stufenweise Verarbeitung von Sensor-(Bild) Daten um zusammenhängende Regionen zu erkennen oder Objekte zu erkennen.

Erkennung z.B. durch Farbe oder Textur



[Lin et al.  
CVPR  
2017]



## Messung von Positionsänderungen

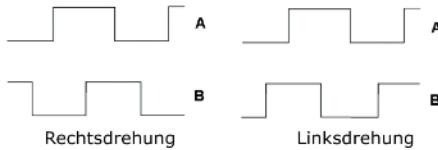
### Inkrementelle Encoder

Durch Drehung einer Scheibe können an 2 Stellen um  $90^\circ$  verschobene Signale gemessen werden und dadurch Winkeländerungen oder Positionsänderungen ermittelt werden



### Quadratur-Codierung

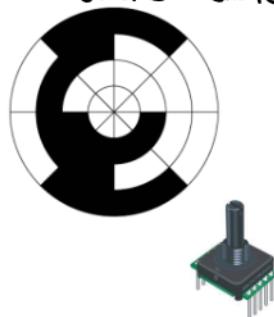
→ gibt Geschwindigkeit und Drehrichtung an



### Absolut-Encoder

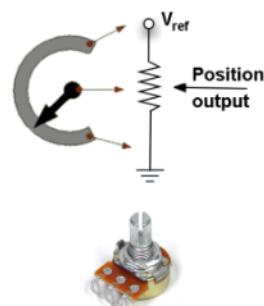
#### Gray-Code

→ alle Signale lassen sich voneinander unterscheiden durch abtasten



#### Potentiometer

→ Signale durch Abtaster und verschiedenen Spannungen erzeugt durch so veränderbaren Widerstand



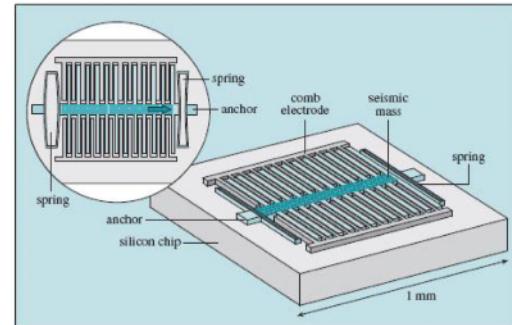
## Magnet-Encoder

- berührungslose Messung
- funktioniert durch Hall-Effekt
- mögliche Varianten: Lineare, rotierende, absolute und inkrementelle Encoder



## Beschleunigungssensoren

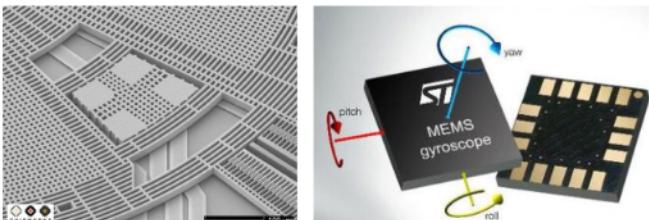
- mikromechanisch hergestellt
- Testmasse federnd aufgehängt
- Auslenkung wird durch Kondensatoren oder piezoelektrischen Effekten bestimmt



## Weitere Sensorenbeispiele:

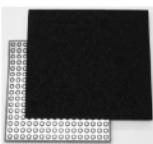
### Drehratensensoren

- Ähnlich wie rotierendes Objekte benötigen auch vibrierende Objekte eine Kraft, um sie aus der Vibrationsebene zu bewegen (*Corioliskraft*)
- mikromechanische Ausführungen mit bis zu drei Achsen



### Künstliche Haut

- Erfassung von Position und Druckkraft
- Schutzfunktion (Polsterung, Schmutz, ...)
- z.B. Matten aus Schaumstoff
  - Elektroden-Matrix
  - Kraftmessung durch Widerstandsänderung



### Kapazitive Sensoren

- Änderung der Kapazität von Kondensatoren durch leitfähige Objekte, wie Finger



### Hex-O-Skin (TUM)

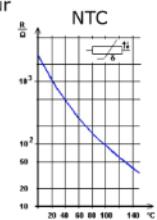
- Sensorzelle mit lokaler Intelligenz
- Mehrere Modalitäten
  - Nähe
  - Beschleunigung
  - Normalkraft
  - Temperatur
- Sensornetzwerk

[www.hex-o-skin.eu]

### Temperaturmessung

#### Thermoresistive Sensoren:

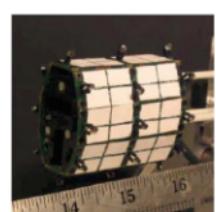
- Metalle
  - Widerstand ändert sich mit der Temperatur
  - Häufig genutzt: Platin
  - Messbereich -200°C bis +500°C
  - Ausführung als Draht oder Metallfilm
  - Problem: Stromfluss erzeugt Wärme
- Keramische Materialien
  - positiver (PTC, Kaltleiter) oder negativer (NTC, Heissleiter) Temperaturkoeffizient



#### Thermoelektrische Sensoren:

- Zwei verschiedene Metalle in Kontakt
- Thermoelektrischer (Seebeck-) Effekt erzeugt Spannung proportional zur Temperatur am Kontakt

### The Huggable (MIT)



- QTC Kraftsensoren
- Thermistoren
- Sensoren für elektrische Felder

### 3. Aktuatoren

#### Elektromagnetische Aktuatoren

- Bewegung durch magnetische Felder
- mittlere Kräfte und Wege

#### Gleichstrom (DC)-Motoren

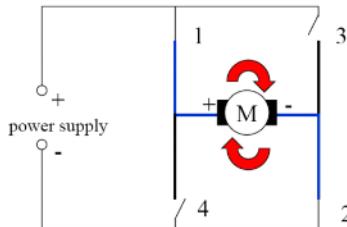
##### Funktion:

- Ansteuerung durch Schleifkontakte
- 1W - 1kW
- kurzfristige Überlastung möglich

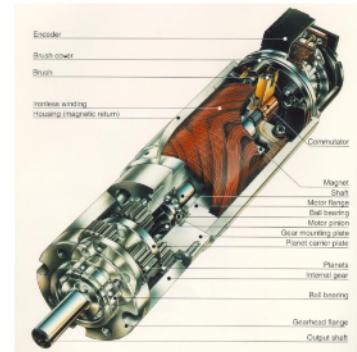
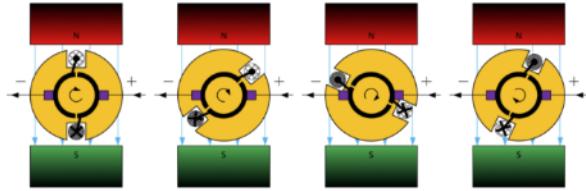
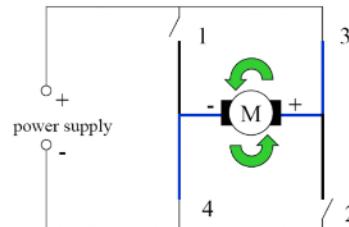
##### Ansteuerung:

- Controller und H-Brücken

Drive forward:



Drive backward:

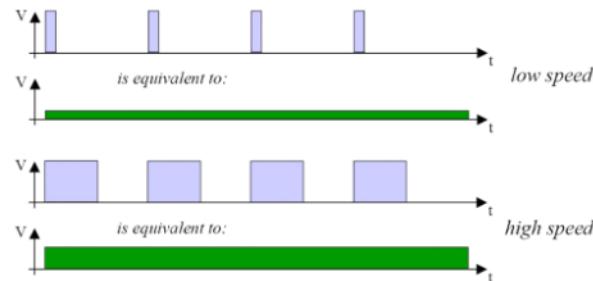


Erlaubt es die Drehrichtung zu ändern

- Ansteuerung durch Puls-Weiten-Modulation

→ hochfrequentes Ein und Aus schalten (z.B. 20 kHz)

→ Pulslänge steuert Geschwindigkeit



##### Vorteile:

- leicht anzusteuern
- einfach, preiswert

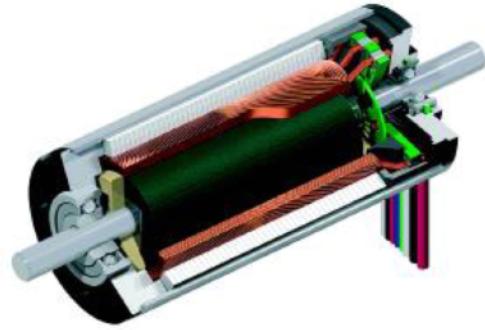
##### Nachteile

- Abnutzung der Bürsten
- Verlustleistung muss abgeführt werden

## Bürstenlose DC-Motoren

### Funktion:

- keine Bürsten
- benötigt Winkel- und Geschwindigkeitssensoren
- komplizierte Ansteuerung



### Vorteile:

- keine Abnutzung
- hohe Drehzahl
- viel Leistung pro Gewichtseinheit (10W - 50kW)
- einfache Mechanik

### Nachteile:

- komplizierter
- teurer

## Schrittmotoren

### Funktion:

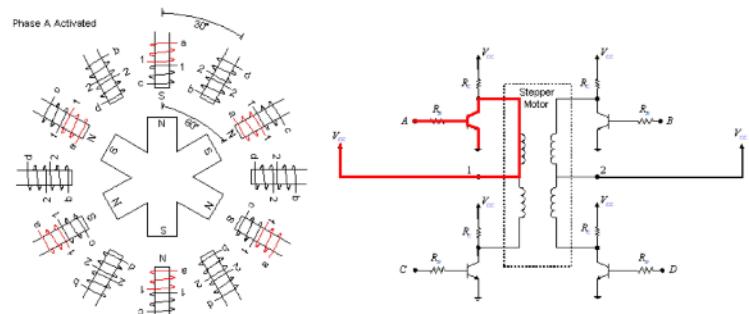
- inkrementelle Ansteuerung durch einzelne Schritte
- vorwärts oder rückwärts
- Winkelsteuerung

### Vorteile:

- präzise
- keine Sensorik benötigt

### Nachteile:

- langsam



## Linearmotoren

### Funktion:

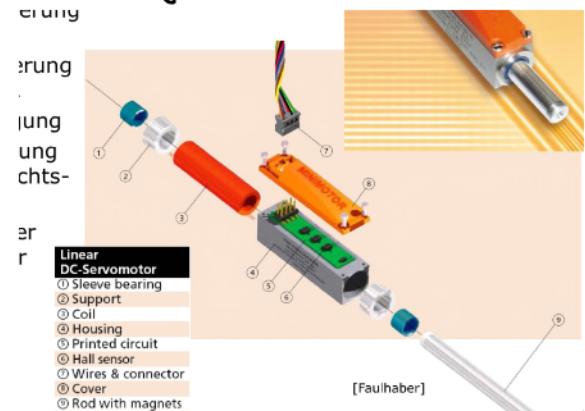
- Linear Bewegung der durch elektrische Spannung angesteuert wird
- komplexer Controller

### Vorteile:

- hohe Beschleunigung
- genaue Positionierung
- viel Leistung pro Gewichtseinheit

### Nachteil:

- komplexer Controller



## Seriell-elastische Aktuatoren

- Feder zwischen Getriebe und Abtrieb
- Positionsmessung auf beiden Seiten der Feder
  - Regelbares Drehmoment bzw. Kraft
  - Schützt Getriebe vor rüttelnder Kraft

## Pneumatische Aktuatoren

- Bewegung durch Luftdruck
- hohe Kräfte, mittlere Kräfte
- elastisch

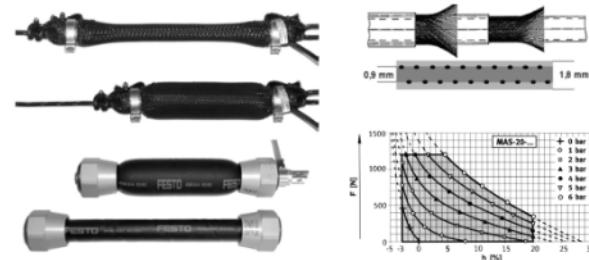


## Pneumatische Muskeln

- gefalteter Polyesterschlach
- nicht dehbare Fasern übertragen Zugkraft

## McKibben-Muskeln

- Schlauch aus drei Gummischichten
  - dazwischen nicht-dehbare Fasern

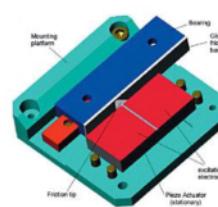


## Hydraulische Aktuatoren

- Bewegung durch Flüssigkeitsdruck
- auch kontinuierliche Drehbewegung

## Piezo-Aktuatoren

- Längenänderung von Kristallen bei Anlegen von elektrischer Spannung



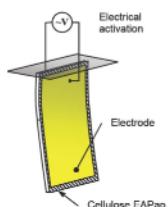
## Ultraschall-Piezomotoren

- hohe Geschwindigkeit
- geringe Kraft

## Gedächtnismetalle

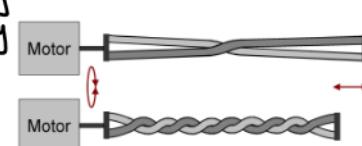
- Verkürzung bei Erwärmung

## Elektroaktive Polymere



## Verdrillungsantrieb

- Schnelle Drehbewegung wird in langsame Zugbewegung umgewandelt



# Getriebe

## Funktion

- Elektromotoren sind nur effizient bei schneller Rotation
- Getriebe reduzieren Geschwindigkeit und erhöhen Drehmoment

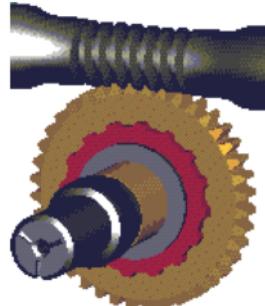
Übersetzungsverhältnis:  $i = \frac{\text{Zähne (Abtrieb)}}{\text{Zähne (Antrieb)}} = \frac{\text{Drehmoment (Abtrieb)}}{\text{Drehmoment (Antrieb)}}$

## Stirnrad-Getriebe

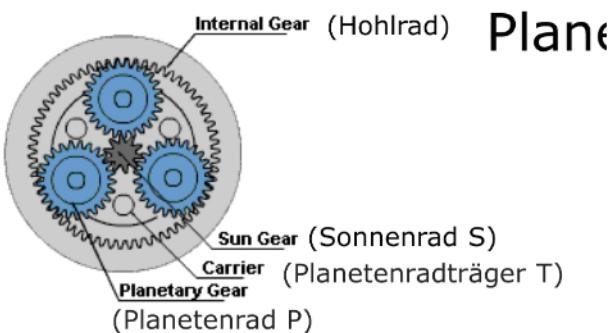


Stirnrad

## Schneckenrad-Getriebe



## Planetengetriebe



## Gleichungen:

- Wille-Gleichung:

$$n_P \cdot z_P = n_T(z_P + z_S) - n_S \cdot z_S$$

- Planetenradgleichung:

$$n_H \cdot z_H = n_T(z_H + z_S) - n_S \cdot z_S$$

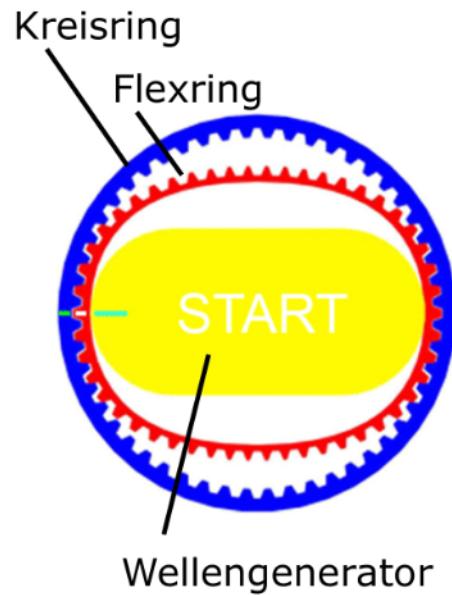
n-Drehzahlen, z-Zähnezahlen

## Harmonic-Drive - Getriebe

### Funktion

- axialer An- bzw. Abtrieb
- Flexring hat wenige Zähne weniger als der Kreisring
- Flexring dreht sich entgegengesetzt zum Wellengenerator  
→ pro Umdrehung werden die Zähne um eine Stelle weiter gedrückt

Übersetzungsverhältnis:  $i = \frac{Z_{\text{Flexring}} - Z_{\text{Kreisring}}}{Z_{\text{Flexring}}}$



### Vorteile:

- Spielfrei
- leicht, kompakt
- hohes Drehmoment (hohe Übersetzung)

### Nachteile:

- sehr langsam

## Quasi-Direct-Drive

### Funktion:

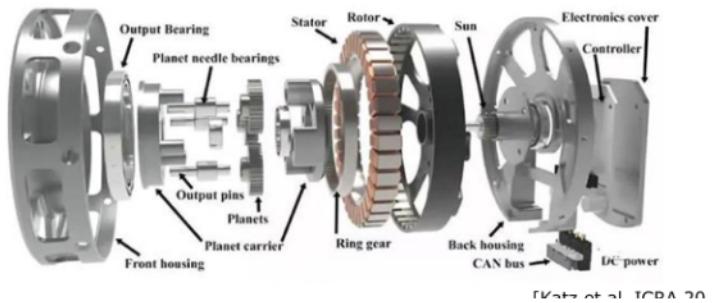
- Flache, bürstenlose DC-Motoren mit Planetengetriebe (geringe Übersetzung)

### Vorteile:

- rücktriebbar
- hohes Drehmoment
- hohe Explosivkraft

### Nachteile:

- Spielbehäftet



## Zyklod - Getriebe

### Funktion

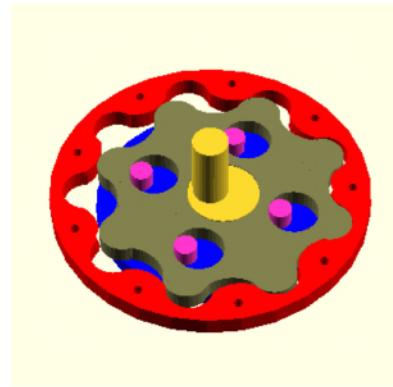
- keine Zähne, sondern Rollen

### Vorteile:

- Spielarm
- hohes Drehmoment

### Nachteile:

- langsam



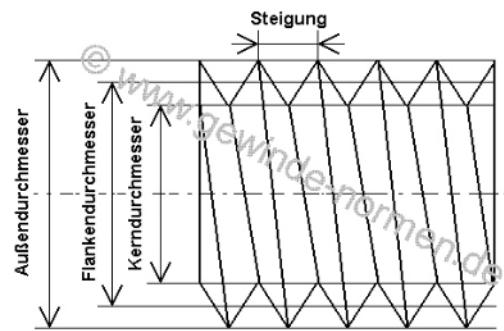
## Gewindestange

### Funktion

- Umwandlung von schneller Drehbewegung in langsame Drehbewegung

### Umrechnung:

$$\text{Energieerhaltung: } M \cdot w = F \cdot s$$



## Servos

Servos bestehen aus einem Elektromotor samt Steuerelektronik

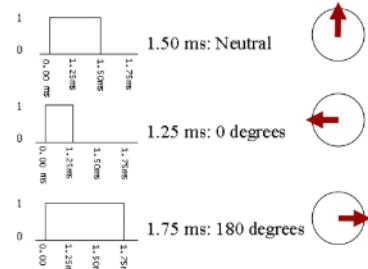
### Modellbau (RC) - Servos

#### Funktion

- PWM-Signal zur Ansteuerung
- Positionsmeßung durch Potenzialmeter
- Integrierte Regelelektronik

#### Vorteile:

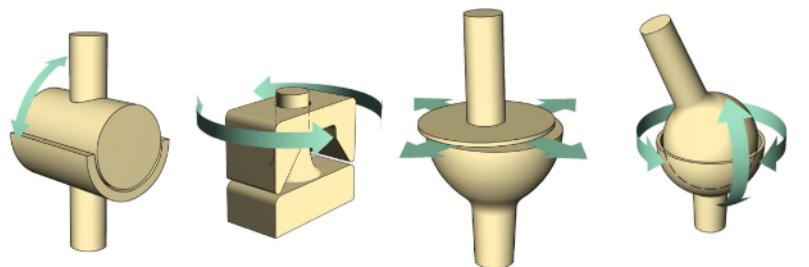
- geringes Gewicht
- hohes Drehmoment



## 4. Gelenke

Verbindung mechanischer Glieder

### Gelenkkarten:



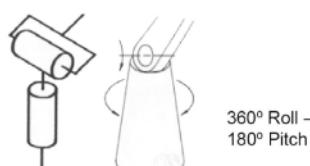
Scharnier

Drehgelenk

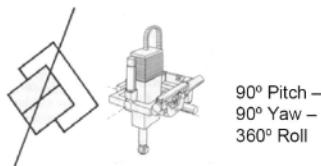
Planares Gelenk

Kugelgelenk

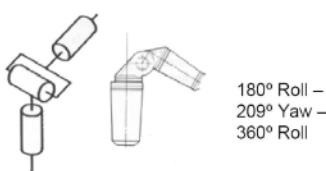
### Kombinierte Gelenke:



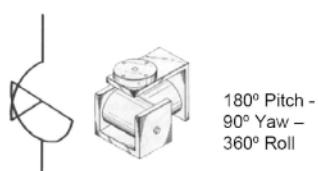
360° Roll –  
180° Pitch



90° Pitch –  
90° Yaw –  
360° Roll



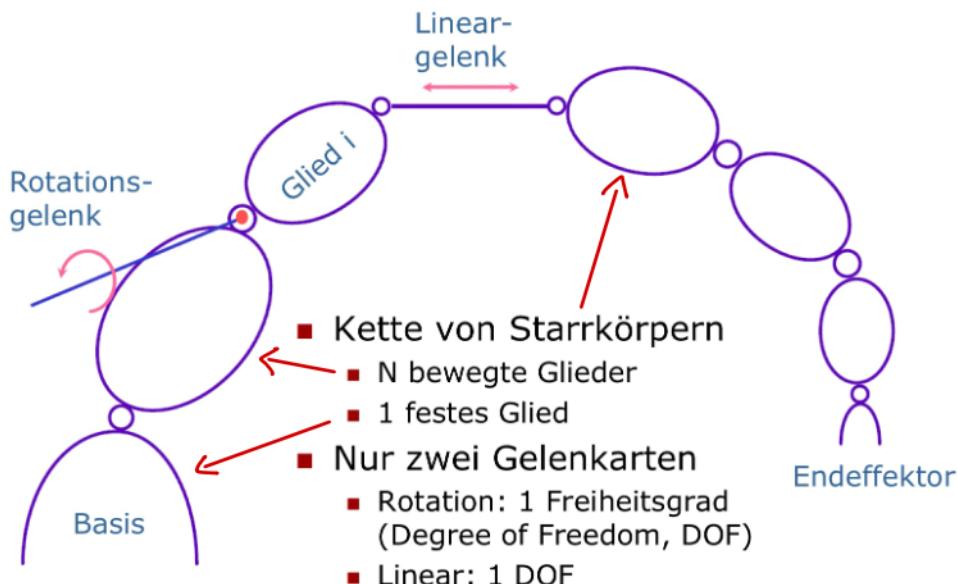
180° Roll –  
209° Yaw –  
360° Roll



180° Pitch –  
90° Yaw –  
360° Roll

## 5. kinematik

### Kinematische Kette von Gelenken: (Manipulator)

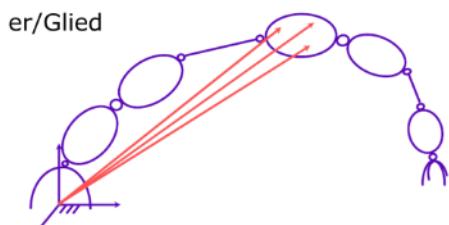


4

### Konfigurationsparameter:

Menge von Positionsparameter um die Ausrichtung / Konfiguration des Systems im Raum zu beschreiben

Beispiel: Drei Punkte um die Lage eines Gliedes im Raum zu beschreiben. Pro Glied 3 Parameter  
→ 9 Parameter pro Glied



### Generalisierte Koordinaten:

→ Menge unabhängiger Konfigurationsparameter

### Freiheitsgrade (DOF)

→ Anzahl generalisierter Koordinaten

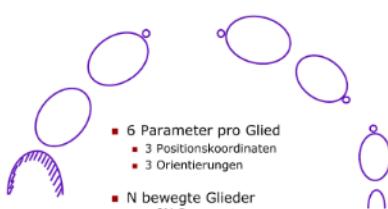
### Bestimmen Generalisierte Koordinaten

6 Parameter pro Glied

→ 3: Positionskoordinaten

→ 3: Orientierung

$n$  Bewegte Glieder ⇒  $6n$  Parameter



### Bestimmen Freiheitsgrade

Betrachtung von 1-DOF-Gelenken

⇒ schränkt 5 Freiheitsgrade ein

→ n 1-DOF-Gelenke:  $5 \cdot n$  Beschränkungen

→ N Glieder :  $6 \cdot n$  Parameter

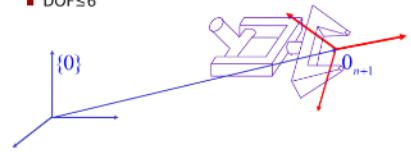
⇒  $n$  Freiheitsgrade



## Endeffektor - Konfigurationsparameter

Menge von Parametern  $(x_1, \dots, x_m)$ , welche die Position und Orientierung des Endeffektors beschreiben  
 $\rightarrow \text{DOF} \leq 6$

■ DOF  $\leq 6$



## Arbeitskoordinaten

Menge von  $M$  unabhängigen Parametern, welche die Endeffektorposition und Orientierung vollständig beschreiben

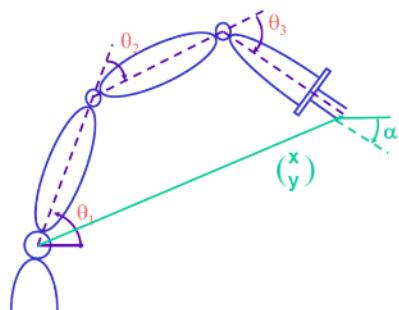
$M$ : Freiheitsgrad des Endeffektors

## Gelenkraum

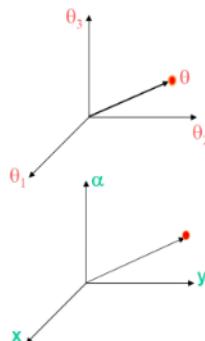
vs.

## Arbeitsraum

### Gelenkstellung



### Arbeitskoordinaten



$\rightarrow$  Konfigurationsraum

$\rightarrow$  Arbeitsraum

## Redundanz

Manipulator ist redundant, wenn er mehr Gelenke als Freiheitsgrade hat

$$N > M$$

$\Rightarrow$  Nullraum: Dimensionalität:  $N - M$

## Konfiguration von Starrkörper zueinander

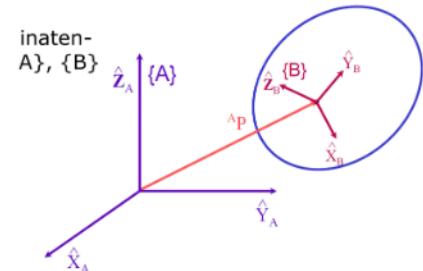
Jeder Starrkörper beschreibt ein eigenes Koordinatensystem  $\Sigma A\}$  und  $\Sigma B\}$

**Translation:** Beschreibt den Koordinatenursprung von  $\Sigma B\}$  in  $\Sigma A\}$

**Rotation:** Beschreibt die Orientierung  $\{\hat{X}_B, \hat{Y}_B, \hat{Z}_B\}$  der Achsen von  $\Sigma B\}$  in  $\Sigma A\}$

Beschreibung von  $\Sigma B\}$  in  $\Sigma A\}$ :

$$\{B\} = \left\{ \begin{matrix} {}^A_B R & {}^A P \end{matrix} \right\}$$



## Rotationsmatrix

Reine Rotation!

Transformation des Koordinatensystems  $\Sigma B\}$  in  $\Sigma A\}$ :

=> Koordinatenachsen transformieren (Darstellung in  $\Sigma A\}$ )

$${}^A \hat{X}_B = {}^A_B R \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad {}^A \hat{Y}_B = {}^A_B R \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad {}^A \hat{Z}_B = {}^A_B R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Rotationsmatrix  ${}^A_B R$

$${}^A_B R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad \begin{array}{l} \leftarrow {}^B \hat{X}_A^T \\ \leftarrow {}^B \hat{Y}_A^T \\ \leftarrow {}^B \hat{Z}_A^T \end{array}$$

$$\text{Also: } {}^A_B R = \begin{bmatrix} {}^A \hat{X}_B & {}^A \hat{Y}_B & {}^A \hat{Z}_B \end{bmatrix}$$

→ Spalten sind die Koordinatenachsen von  $\Sigma B\}$  in  $\Sigma A\}$

Berechnung:

Berechnung der Achsen durch Skalarprodukt:

$${}^A \hat{X}_B = \begin{bmatrix} \hat{X}_B \cdot \hat{X}_A \\ \hat{X}_B \cdot \hat{Y}_A \\ \hat{X}_B \cdot \hat{Z}_A \end{bmatrix} \Rightarrow {}^A_B R = \begin{bmatrix} \hat{X}_B \cdot \hat{X}_A & \hat{Y}_B \cdot \hat{X}_A & \hat{Z}_B \cdot \hat{X}_A \\ \hat{X}_B \cdot \hat{Y}_A & \hat{Y}_B \cdot \hat{Y}_A & \hat{Z}_B \cdot \hat{Y}_A \\ \hat{X}_B \cdot \hat{Z}_A & \hat{Y}_B \cdot \hat{Z}_A & \hat{Z}_B \cdot \hat{Z}_A \end{bmatrix}$$

Inverse Rotationsmatrix:

$${}^A_B R^{-1} = {}^A_B R^T$$

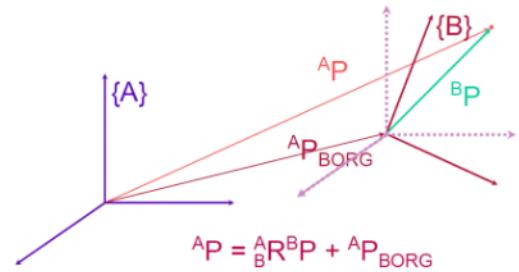
## Mapping: Wechsel des Koordinatensystems

Ziel: P ist in  $\mathcal{E}B\}$  gegeben und soll in  $\mathcal{E}A\}$  dargestellt werden  
 → Wechsel des Koordinatensystems

nur Rotation:  ${}^A P = {}_B R {}^B P$

nur Translation:  ${}^A P = {}^B P + {}^A P_{BORG}$

Allgemein:  ${}^A P = {}_B R {}^B P + {}^A P_{BORG}$



## Homogene Transformation

Verhindert aneinander Reihung von Transformationen  
 → Transformation durch Matrix

Homogene Transformationsmatrix:

$$\begin{bmatrix} {}^A P \\ 1 \end{bmatrix} = \begin{bmatrix} {}_B R & {}^A P_{BORG} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}^B P \\ 1 \end{bmatrix}$$

## Operator: Abbildung eines Punktes

Ziel: Punkt  ${}^A P_1$  soll auf  ${}^A P_2$  abgebildet werden

→ T:  $P_1 \rightarrow P_2$

Rotation um Achse k mit Winkel  $\theta$ :

Rotationsmatrizen:

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

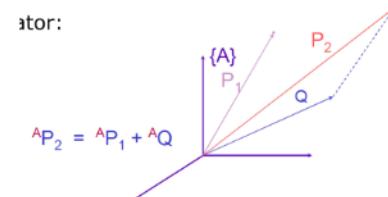
$R_K(\theta)$ :  $P_1 \rightarrow P_2$

$$P_2 = R_K(\theta) P_1$$

Translation:  ${}^A P_2 = {}^A P_1 + {}^A Q$

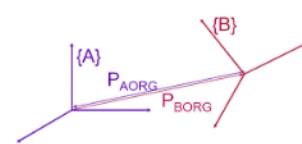
Allgemein:  $P_2 = \begin{pmatrix} R_K(\theta) & Q \\ 0 & 0 & 0 & 1 \end{pmatrix} P_1$

$\underbrace{\qquad\qquad}_{T}$



## Inverse Transformation:

$${}_B T^{-1} = {}_A T = \begin{bmatrix} {}_B R^T & -{}_B R^T \cdot {}^A P_{BORG} \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^B P_{AORG}$$

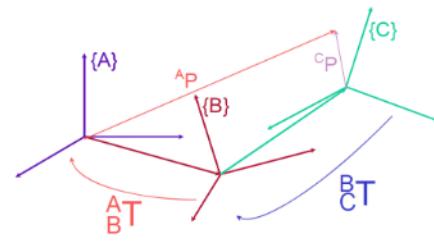


## Transformationsketten

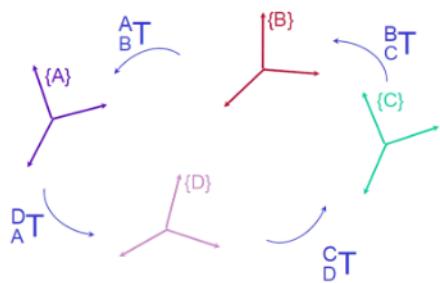
Transformation über mehrere Koordinatensysteme

Resultierende Transformationsmatrix:  ${}^C T = {}^A T {}^B T {}^C T$

$${}^A T = \begin{bmatrix} {}^A R {}^B C & {}^A R {}^B P_{Corg} + {}^A P_{Borg} \\ 0 & 1 \end{bmatrix}$$



## Zyklische Transformation



$${}^A T {}^B T {}^C T {}^D T {}^A T = I$$

→ Fehlende Transformationen können berechnet werden

$$\rightarrow {}^B T = {}^B T {}^C T {}^D T {}^A T$$

→ Falls die Richtung der Transformation falsch ist, kann mit der Inversen multipliziert werden

## Endeffektor-Konfiguration

→ Bewegung relativ zur Basis

Transformationsmatrix:  ${}^B E T$ : Position + Orientierung

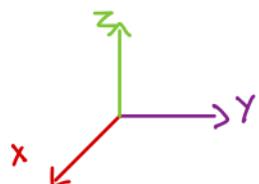
Endeffektor-Konfigurationsparameter:  $X = \begin{bmatrix} X_P \\ X_R \end{bmatrix}$

Position  
Orientierung

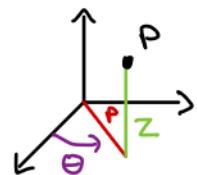


## Positionsrepräsentation

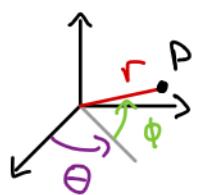
Kartesisch:  $(x, y, z)$



Zylindrisch:  $(\rho, \theta, z)$



Sphärisch:  $(r, \theta, \phi)$



## Umrechnung

Cartesian  $\rightarrow$  Cylindrical:  $\rho = \sqrt{x^2 + y^2}$   $\Theta = \arctan\left(\frac{y}{x}\right)$   $z = z$

Cartesian  $\rightarrow$  Spherical:  $r = \sqrt{x^2 + y^2 + z^2}$   $\Theta = \arctan\left(\frac{y}{x}\right)$

$$\Phi = \arctan\left(\frac{z}{\sqrt{x^2 + y^2}}\right)$$

## Orientierungsrepräsentation:

### Euler Winkel

Rotation um bewegliche Rotationsachsen  
→ es wird um rotierte Achsen rotiert

Es gilt: Rotationsmatrizen werden entgegen der Reihenfolge der Rotationen multipliziert:  $R_{\text{ges}} = R_1 \dots R_n R_1$

Beispiel: (z-y-x) Eulerwinkel:  $\overset{A}{B}R = R_z(\alpha) \cdot R_y(\beta) \cdot R_x(\gamma)$

### Absolut Winkel

Rotation um feste Rotationsachsen  
→ es wird um ursprüngliche Achsen rotiert

Es gilt: Rotationsmatrizen werden in Reihenfolge der Rotationen multipliziert:  $R_{\text{ges}} = R_1 R_2 \dots R_n$

Beispiel: (x-y-z) Absolutwinkel:  $\overset{A}{B}R = \overset{A}{B}R_{XYZ}(\gamma, \beta, \alpha) = R_z(\alpha) \cdot R_y(\beta) \cdot R_x(\gamma)$

### Allgemein gilt:

$$\text{Abs: } R_{XYZ}(\gamma, \beta, \alpha) = R_z(\alpha) \cdot R_y(\beta) \cdot R_x(\gamma)$$

$$\text{Eul: } R_{ZYX''}(\alpha, \beta, \gamma) = R_z(\alpha) \cdot R_y(\beta) \cdot R_x(\gamma) \Rightarrow R_{ZYX''}(\alpha, \beta, \gamma) = R_{XYZ}(\gamma, \beta, \alpha)$$

### Inverses Problem

Gegeben: Rotationsmatrix  $\overset{A}{B}R \rightarrow$  Gesucht: Winkel  $(\alpha, \beta, \gamma)$

Problem: Durch Umformen erhält man Ausdrücke, wie  
 $\cos(\beta) = 0$  oder  $\sin(\beta) = 0$

### Singularität der Repräsentation

→ Winkel können nicht eindeutig bestimmt werden

$$\cos(\beta) = 0 \Rightarrow \beta = (\pm 90^\circ)$$

$$\sin(\beta) = 0 \Rightarrow \beta = 0^\circ \vee \beta = 180^\circ$$

⇒ jede Rotationsrepräsentation mit 3 Winkeln hat Singularität

## Euler-Parameter (Quaternion)

→ keine Singularität

Definition:

$$\varepsilon_1 = W_x \cdot \sin \frac{\theta}{2}$$

Es gilt:  $|W| = 1, \quad \varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 + \varepsilon_4^2 = 1$

$$\varepsilon_2 = W_y \cdot \sin \frac{\theta}{2}$$

$$\varepsilon_3 = W_z \cdot \sin \frac{\theta}{2}$$

$$\varepsilon_4 = \cos \frac{\theta}{2}$$

## Inverses Problem

Gegeben Rotationsmatrix  ${}^A_R$  → Gesucht: Euler-Parameter  $\varepsilon_i$

Ansatz:

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \equiv \begin{bmatrix} 1 - 2\varepsilon_2^2 - 2\varepsilon_3^2 & 2(\varepsilon_1\varepsilon_2 - \varepsilon_3\varepsilon_4) & 2(\varepsilon_1\varepsilon_3 + \varepsilon_2\varepsilon_4) \\ 2(\varepsilon_1\varepsilon_2 + \varepsilon_3\varepsilon_4) & 1 - 2\varepsilon_1^2 - 2\varepsilon_3^2 & 2(\varepsilon_2\varepsilon_3 - \varepsilon_1\varepsilon_4) \\ 2(\varepsilon_1\varepsilon_3 - \varepsilon_2\varepsilon_4) & 2(\varepsilon_2\varepsilon_3 + \varepsilon_1\varepsilon_4) & 1 - 2\varepsilon_1^2 - 2\varepsilon_2^2 \end{bmatrix}$$

Lösung:

$$\varepsilon_4 = \frac{1}{2} \sqrt{1 + r_{11} + r_{22} + r_{33}}$$

$$\varepsilon_1 = \frac{r_{32} - r_{23}}{4\varepsilon_4}, \quad \varepsilon_2 = \frac{r_{13} - r_{31}}{4\varepsilon_4}, \quad \varepsilon_3 = \frac{r_{21} - r_{12}}{4\varepsilon_4}$$

Falls  $\varepsilon_4 = 0$ : Teile durch den größten Parameter:

$$\varepsilon_1 \text{ maximal: } \varepsilon_1 = \frac{1}{2} \sqrt{r_{11} - r_{22} - r_{33} + 1}$$

$$\varepsilon_2 = \frac{(r_{21} + r_{12})}{4\varepsilon_1}, \quad \varepsilon_3 = \frac{(r_{31} + r_{13})}{4\varepsilon_1}, \quad \varepsilon_4 = \frac{(r_{32} - r_{23})}{4\varepsilon_1}$$

$$\varepsilon_2 \text{ maximal: } \varepsilon_2 = \frac{1}{2} \sqrt{-r_{11} + r_{22} - r_{33} + 1}$$

$$\varepsilon_3 \text{ maximal: } \varepsilon_3 = \frac{1}{2} \sqrt{-r_{11} - r_{22} + r_{33} + 1}$$

$$\varepsilon_4 \text{ maximal: } \varepsilon_4 = \frac{1}{2} \sqrt{1 + r_{11} + r_{22} + r_{33}}$$

## Berechnung von Euler-Parametern aus Euler Winkeln

$$\varepsilon_1 = \sin \frac{\beta}{2} \cos \frac{\alpha - \gamma}{2}$$

$$\varepsilon_2 = \sin \frac{\beta}{2} \sin \frac{\alpha - \gamma}{2}$$

$$\varepsilon_3 = \cos \frac{\beta}{2} \sin \frac{\alpha + \gamma}{2}$$

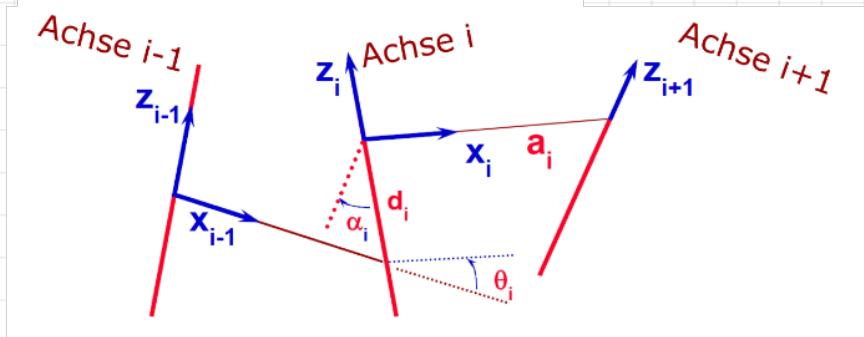
$$\varepsilon_4 = \cos \frac{\beta}{2} \cos \frac{\alpha + \gamma}{2}$$

## 6. Denavit-Hartenberg-Parameter

- Verfahren, was die direkte Kinematik erleichtert
- Richtlinien zur Festlegung von Gelenkkoordinatensystemen

### DH-Parameter

**DH-Parameter:** ( $\alpha_i, a_i, d_i, \theta_i$ )

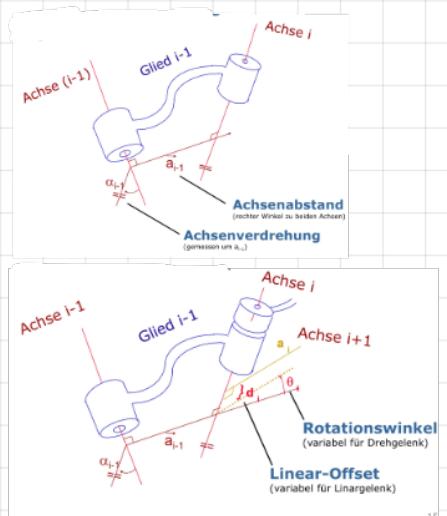


$\alpha_i$ : Achsenverdrehung von i und i+1 zueinander  
→ wird an i gemessen

$a_i$ : Achsenabstand von i und i+1  
→ gerade mit rechtem Winkel zu beiden

$d_i$ : Abstand der Verbindungsgerade  $x_{i-1}$  und  $x_i$   
→ Abstand wird auf Achse i gemessen  
→ Variabel für Lineargelenke

$\theta_i$ : Drehung der Verbindungsgeraden  $x_{i-1}$  und  $x_i$   
→ gemessen an Achse i  
→ Variabel für Drehgelenke

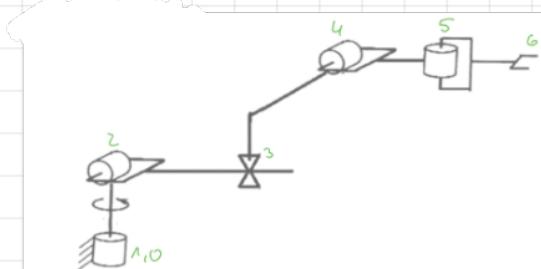


# Verfahren zur Bestimmung der DH-Parameter

## Festlegung der Koordinatensysteme

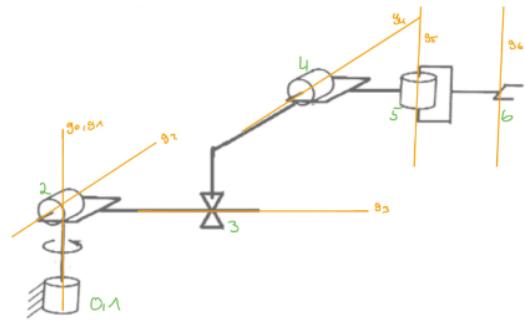
### 1. Nummerierung der Armteile

- (Fuß → Armteil 0)
- erstes Gelenk → Armteil 1, ...
- letztes Gelenk → Armteil i
- (Endeffektor → Armteil i+1)



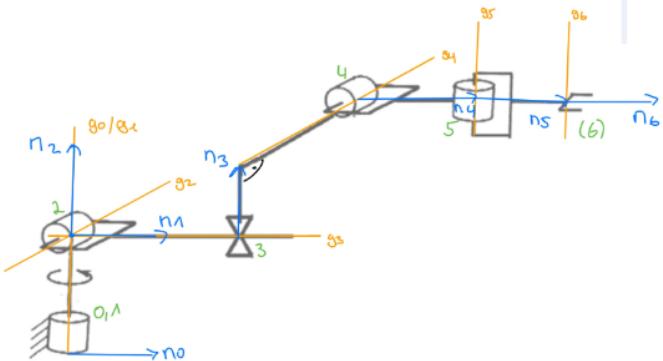
### 2. Kennzeichnen der Gelenkachsen $g_i$

- Linear-/Gleitachsen oder Rotations-/Drehachsen
- Achse  $g_0$  am Fuß parallel zu  $g_1$
- Achse  $g_{i+1}$  am Endeffektor parallel zu  $g_i$



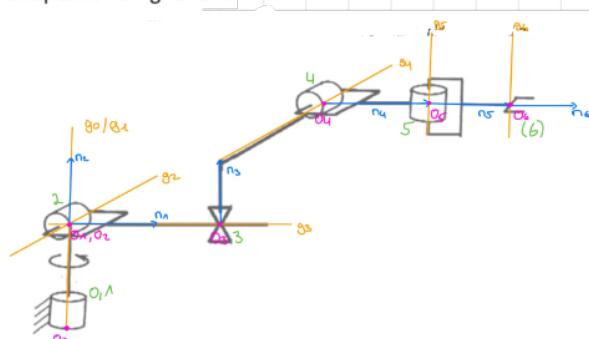
### 3. Normalen $n_i$ von $g_i$ und $g_{i+1}$ festlegen

- Fall 1:  $g_i$  und  $g_{i+1}$  parallel →  $n_i$  beliebig entlang  $g_i$
- Fall 2:  $g_i$  und  $g_{i+1}$  schneiden sich →  $n_i$  in Schnittpunkt (Richtung bestimmt Vorzeichen von  $a_i$ )
- Fall 3:  $g_i$  und  $g_{i+1}$  windschief:  $n_i$  mit minimalen Abstand zu  $g_{i+1}$
- $n_0$  möglichst parallel zu  $n_1$
- $n_{i+1}$  möglichst parallel zu  $n_i$  und senkrecht zu  $g_i$



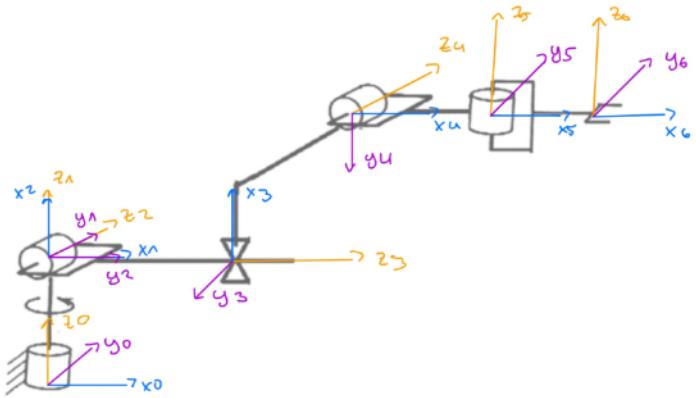
### 4. Koordinatensystemursprünge $O_i$ festlegen

- $O_0$  in Basis der kinematischen Kette
- $O_{i+1}$  in Endeffektor
- $O_i$  in Schnittpunkt von  $g_i$  und  $g_{i+1}$ , falls dieser nicht existiert in Schnittpunkt von  $g_i$  und  $n_i$



## 5. Koordinatensystemachsen eintragen

- $z_i$  entlang  $g_i$  an  $O_i$
- $x_i$  entlang  $n_i$  an  $O_i$
- $y_i$  gemäß der Rechten-Hand-Regel



## 6. Identifizieren der DH-Parameter

$a_i$ = Distanz ( $z_i, z_{i+1}$ ) entlang  $x_i$

$\alpha_i$ = Winkel ( $z_i, z_{i+1}$ ) um  $x_i$

$d_i$ = Distanz ( $x_{i-1}, x_i$ ) entlang  $z_i$

$\theta_i$ = Winkel ( $x_{i-1}, x_i$ ) um  $z_i$

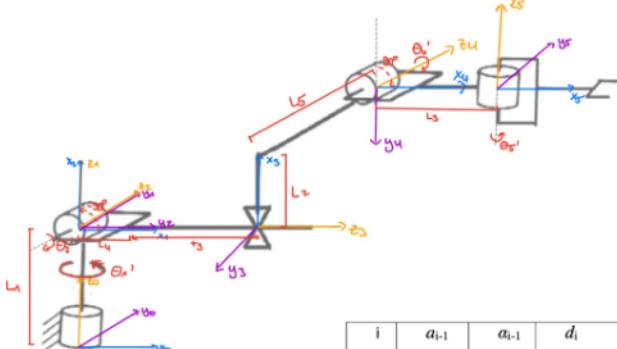
$a_{i-1}$ = Distanz ( $z_{i-1}, z_i$ ) entlang  $x_{i-1}$

$\alpha_{i-1}$ = Winkel ( $z_{i-1}, z_i$ ) um  $x_{i-1}$

$d_i$ = Distanz ( $x_{i-1}, x_i$ ) entlang  $z_i$

$\theta_i$ = Winkel ( $x_{i-1}, x_i$ ) um  $z_i$

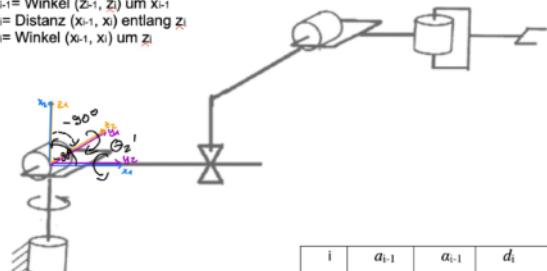
## 7. Denavit-Hartenberg-Paramter festlegen



i	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
1	0	$0^\circ$	$l_1$	$0^\circ + \theta_{1'}$ $= \theta_{1'}$
2	0	$-90^\circ$	0	$-90 + \theta_2'$
3	0	$-90^\circ$	$l_4 + l_5$	$0^\circ$
4	$l_2$	$90^\circ$	$l_5$	$90^\circ + \theta_4'$
5	$l_3$	$90^\circ$	0	$0^\circ + \theta_5' = 0.5^\circ$

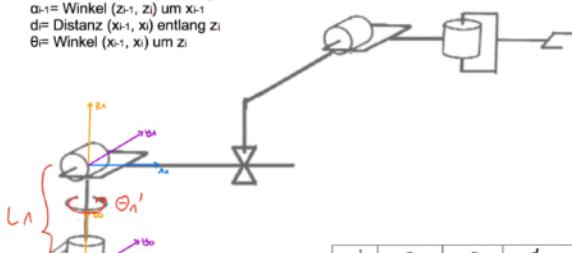
konstant      Variabel je nach Gelenktyp

$a_i$ = Distanz ( $z_{i-1}, z_i$ ) entlang  $x_{i-1}$   
 $\alpha_i$ = Winkel ( $z_{i-1}, z_i$ ) um  $x_{i-1}$   
 $d_i$ = Distanz ( $x_{i-1}, x_i$ ) entlang  $z_i$   
 $\theta_i$ = Winkel ( $x_{i-1}, x_i$ ) um  $z_i$



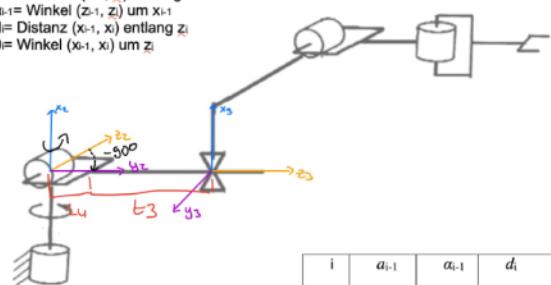
i	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
1	0	$0^\circ$	$l_1$	$0^\circ + \theta_{1'}$ $= \theta_{1'}$
2	0	$-90^\circ$	0	$-90^\circ + \theta_2'$
3				
4				
5				

$a_{i-1}$ = Distanz ( $z_{i-1}, z_i$ ) entlang  $x_{i-1}$   
 $\alpha_{i-1}$ = Winkel ( $z_{i-1}, z_i$ ) um  $x_{i-1}$   
 $d_i$ = Distanz ( $x_{i-1}, x_i$ ) entlang  $z_i$   
 $\theta_i$ = Winkel ( $x_{i-1}, x_i$ ) um  $z_i$

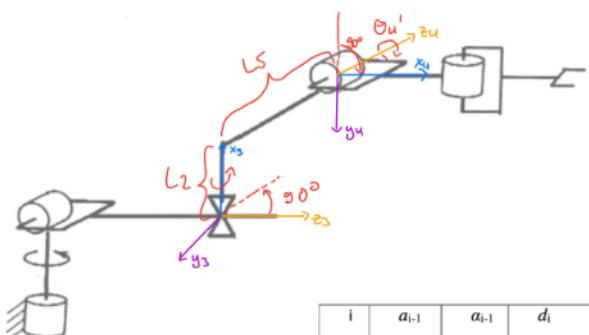


i	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
1	0	$0^\circ$	$l_1$	$0^\circ + \theta_{1'}$ $= \theta_{1'}$
2	0	$-90^\circ$	0	$-90 + \theta_2'$
3	0	$-90^\circ$	$l_4 + l_5$	$0^\circ$
4				
5				

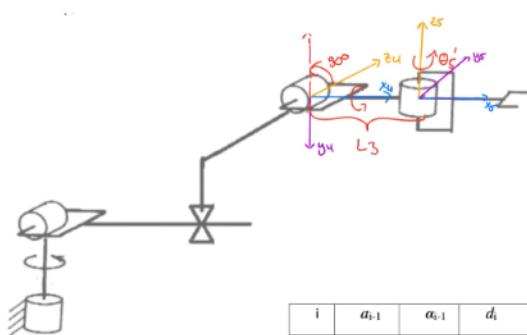
$a_i$ = Distanz ( $z_{i-1}, z_i$ ) entlang  $x_{i-1}$   
 $\alpha_i$ = Winkel ( $z_{i-1}, z_i$ ) um  $x_{i-1}$   
 $d_i$ = Distanz ( $x_{i-1}, x_i$ ) entlang  $z_i$   
 $\theta_i$ = Winkel ( $x_{i-1}, x_i$ ) um  $z_i$



i	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
1	0	$0^\circ$	$l_1$	$0^\circ + \theta_{1'}$ $= \theta_{1'}$
2	0	$-90^\circ$	0	$-90^\circ + \theta_2'$
3	0	$-90^\circ$	$l_4 + l_5$	$0^\circ$
4				
5				



i	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
1	0	0°	$L_1$	$0^\circ + \theta_{1'}$ = $\theta_{1'}$
2	0	-90°	0	$-90^\circ + \theta_{2'}$
3	0	-90°	$L_4 + t_3$	0°
4	$L_2$	90°	$L_6$	$90^\circ + \theta_{4'}$
5				



i	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
1	0	0°	$L_1$	$0^\circ + \theta_{1'}$ = $\theta_{1'}$
2	0	-90°	0	$-90^\circ + \theta_{2'}$
3	0	-90°	$L_4 + t_3$	0°
4	$L_2$	90°	$L_5$	$90^\circ + \theta_{4'}$
5	$L_3$	90°	0	$90^\circ + \theta_{5'}$ = $\theta_{5'}$

## DH-Vorwärtshinematik

Transformation durch DH-Parameter:

$${}^{i-1}T_{(\alpha_{i-1}, a_{i-1}, \theta_i, d_i)} = R_x(\alpha_{i-1}) D_x(a_{i-1}) R_z(\theta_i) D_z(d_i)$$

Allgemein:

$${}^i T = \left[ \begin{array}{ccc|c} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{array} \right]$$

Transformation von kinematischen Ketten

$${}^0 T_N = {}^0 T_1 {}^1 T_2 \dots {}^{N-1} T_N$$

## 7. Differentielle Kinematik

Veränderung von kleinen Winkeländerungen:

$$\theta + \delta\theta \rightarrow x + \delta x$$

Gelenkgeschwindigkeit:

Gelenkgeschwindigkeit —  $\dot{\theta} \leftrightarrow \dot{x}$  — Lineargeschwindigkeit  
Winkelgeschwindigkeit

Gelenkkoordinaten

→ Darstellung durch einen variablen Gelenkparameter  $i$ :

$$q_i = \bar{\varepsilon}_i \theta_i + \varepsilon_i d_i \quad \varepsilon_i = \begin{cases} 0 & \text{wenn Drehgelenk} \\ 1 & \text{wenn Lineargelenk} \end{cases}$$

Gelenk-koordinaten-Vektor fasst alle Gelenke zusammen:

$$q = (q_1, q_2, \dots, q_n)^T$$

Jacobi-Matrix

→ Gegeben sei eine Funktion  $x = f(q)$ :

→ Jacobi-Matrix besteht aus den partiellen Ableitungen von  $f$  und vermittelt zwischen  $dq$  und  $dx$

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} = \begin{pmatrix} f_1(q) \\ f_2(q) \\ \vdots \\ f_m(q) \end{pmatrix}$$

Jacobi Matrix:

$$\delta x = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \dots & \frac{\partial f_1}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial q_1} & \dots & \frac{\partial f_m}{\partial q_n} \end{bmatrix} \cdot \delta q$$

Es gilt:  $\delta x_{(m \times 1)} = J_{(m \times n)}(q) \delta q_{(n \times 1)}$

$$\dot{x}_{(m \times 1)} = J_{(m \times n)}(q) \dot{q}_{(n \times 1)} \quad J_{ij}(q) = \frac{\partial}{\partial q_j} f_i(q) \quad \begin{pmatrix} \dot{x}_P \\ \dot{x}_R \end{pmatrix} = \begin{pmatrix} J_{X_P}(q) \\ J_{X_R}(q) \end{pmatrix} \dot{q}$$

→ Jacobi-Matrix ist Repräsentationsabhängig

Position

Rotation

Basis Jacobi-Matrix

→ Bezug aufs Basis koordinatensystem  
→ repräsentationsunabhängig

$$\begin{pmatrix} v \\ \omega \end{pmatrix}_{(6 \times 1)} = J_0(q)_{(6 \times n)} \dot{q}_{(n \times 1)}$$

Repräsentation von  $x_P$

$$\dot{x}_P = E_P(x_P)v$$

$$\dot{x}_R = E_R(x_R)\omega$$

⇒ Jacobi-Matrix für eine Repräsentation kann aus der Basis-Jacobi-Matrix hergeleitet werden

$$J_x(q) = E(x) J_0(q) \quad J = \left( \begin{array}{c|c} J_{XP} & 0 \\ \hline J_{XR} & E_R \end{array} \right) = \left( \begin{array}{c|c} E_P & 0 \\ \hline 0 & E_R \end{array} \right) \left( \begin{array}{c|c} J_v & 0 \\ \hline J_w & 0 \end{array} \right)$$

## Lineargeschwindigkeit

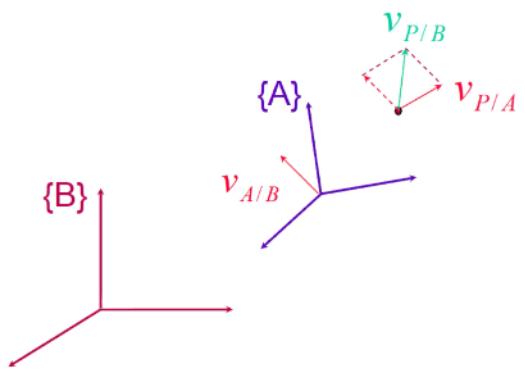
→ Beschreibung bezüglich verschiedener Koordinatensysteme

### Reine Translation:

→ Addition der relativ Geschwindigkeit

$$v_{P/B} = v_{A/B} + v_{P/A}$$

↑  
Geschwindigkeit  
Koordinatensystem



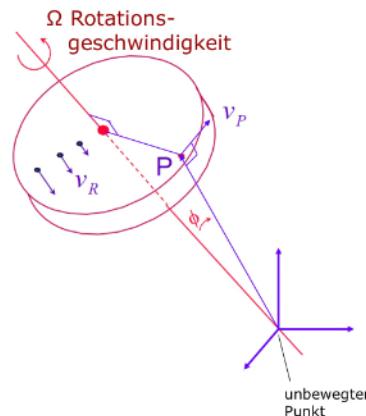
### Rotationsbewegung:

→ Linearbeschwindigkeit ist abhängig von Rotationsgeschwindigkeit  $\Omega$  und Abstand vom Ursprung:

$$v_P = \Omega \times P$$

Kreuzprodukt durch schiefsymmetrische Matrix:

$$c = \hat{a}b = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix}$$



### Operator für Rotation

$$\Omega = \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix}; P = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \Rightarrow v_P = \hat{\Omega}P = \begin{bmatrix} 0 & -\Omega_z & \Omega_y \\ \Omega_z & 0 & -\Omega_x \\ -\Omega_y & \Omega_x & 0 \end{bmatrix} \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}$$

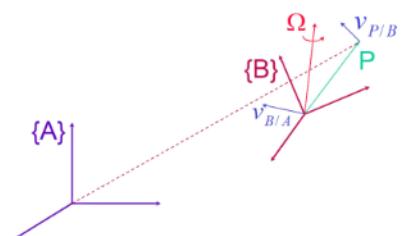
### Translation und Rotation

#### Allgemeine Linearbeschwindigkeit:

$$v_{P/A} = v_{B/A} + v_{P/B} + \Omega \times P_B$$

#### Linearbeschwindigkeit in $\{A\}$ :

$$\mathcal{Q}v_{P/A} = \mathcal{Q}v_{B/A} + {}_B^A R \cdot {}^B v_{P/B} + {}_B^A \Omega \times {}_B^A R \cdot {}^B P_B$$



## Geschwindigkeitspropagierung kinematischer Ketten

Linear:  $v_{i+1} = v_i + \omega_i \times P_{i+1} + \dot{d}_{i+1} \cdot Z_{i+1}$

Rotation:  $\omega_{i+1} = \omega_i + \Omega_{i+1}$

Achse des Vorgängers  $\Omega_{i+1} = \dot{\theta}_{i+1} \cdot Z_{i+1}$   
eigene Achse

Von Gelenk i nach i+1:

$${}^{i+1}\omega_{i+1} = {}^i\omega_i + \dot{\theta}_{i+1} \cdot {}^{i+1}Z_{i+1}$$

$${}^{i+1}v_{i+1} = {}^iR \cdot ({}^iv_i + {}^i\omega_i \times {}^iP_{i+1}) + \dot{d}_{i+1} \cdot {}^{i+1}Z_{i+1}$$

Richttransformation ins Basis-System

$$\begin{pmatrix} {}^0v_n \\ {}^0\omega_n \end{pmatrix} = \begin{pmatrix} {}^0R & 0 \\ 0 & {}^0R \end{pmatrix} \begin{pmatrix} {}^nv_n \\ {}^n\omega_n \end{pmatrix}$$

Jacobi Matrix: Explizite Form

→ Jacobi Matrix die Linear und Rotation Bewegung beschreibt

Einflüsse auf den Endeffektor

Lineargeschw.  $V_j$        $\Omega_i \times P_{in}$

Winkelgeschw. —       $\Omega_i$

Lineargeschwindigkeit des Endeffektors:

$$v = \sum_{i=1}^n [{}^iV_i + \bar{\epsilon}_i (\Omega_i \times P_{in})] \Leftrightarrow V_i = Z_i \dot{q}_i \Rightarrow v = J_v \dot{q}$$

Rotationsgeschwindigkeit des Endeffektors:

$$\omega = \sum_{i=1}^n \bar{\epsilon}_i \Omega_i \Leftrightarrow \Omega_i = Z_i \dot{q}_i \Rightarrow \omega = J_\omega \dot{q}$$

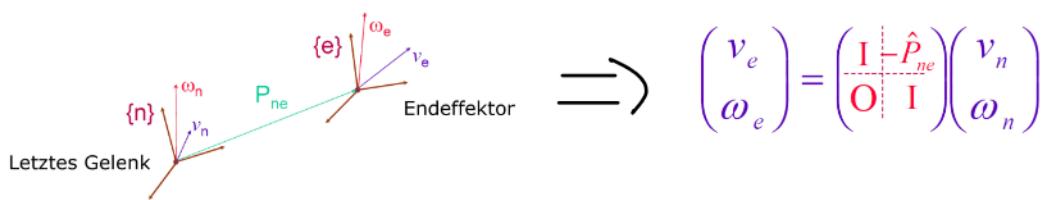
Allgemeine Jacobi-Matrix: Direkte Differentiation

$$J = \begin{pmatrix} \frac{\partial x_p}{\partial q_1} & \frac{\partial x_p}{\partial q_2} & \dots & \frac{\partial x_p}{\partial q_n} \\ \frac{\partial y_p}{\partial q_1} & \frac{\partial y_p}{\partial q_2} & \dots & \frac{\partial y_p}{\partial q_n} \\ \frac{\partial z_p}{\partial q_1} & \frac{\partial z_p}{\partial q_2} & \dots & \frac{\partial z_p}{\partial q_n} \end{pmatrix} \Rightarrow {}^0J = \begin{pmatrix} \frac{\partial}{\partial q_1}({}^0x_p) & \frac{\partial}{\partial q_2}({}^0x_p) & \dots & \frac{\partial}{\partial q_n}({}^0x_p) \\ \frac{\partial}{\partial q_1}({}^0y_p) & \frac{\partial}{\partial q_2}({}^0y_p) & \dots & \frac{\partial}{\partial q_n}({}^0y_p) \\ \frac{\partial}{\partial q_1}({}^0z_p) & \frac{\partial}{\partial q_2}({}^0z_p) & \dots & \frac{\partial}{\partial q_n}({}^0z_p) \end{pmatrix}$$

$${}^0Z_i = {}^0R \cdot {}^iZ_i \quad {}^iZ_i = Z = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

## Jacobi Matrix des Endeffektors

→ besteht nur aus Translation



## Wechsel des Koordinatensystems

$${}^0 J_e = \begin{pmatrix} {}^0 R & - {}^0 R {}^n \hat{P}_{ne} {}^0 R^T \\ 0 & {}^0 R \end{pmatrix} {}^n J_n$$

## Kinematische Singularität

→ der Endeffektor kann an bestimmten Stellen nicht mehr in bestimmte Richtungen verschoben werden

⇒ Spalten in Jacobi-Matrix sind abhängig

$$\Rightarrow \det(J) = 0$$

## Singuläre Konfigurationen

→ Determinante Null setzen

→ Nullstellen ausrechnen

→ Beschreiben singuläre Konfigurationen

$$\boxed{\begin{aligned} S_1(q) &= 0 \\ S_2(q) &= 0 \\ &\vdots \\ S_s(q) &= 0 \end{aligned}}$$

## Inverse Kinematik mit inverser Jacobi-Matrix

→ möglich, wenn es keine Singularität gibt

$$\Rightarrow J \text{ invertierbar: } \delta\theta = J^{-1}(\theta) \delta x$$

**Mehrdeutigkeit:** Entsteht bei redundanten Armen

→ Nullraum: Gelenkstellungen, die die gleiche Endeffektorpose erzeugen

Lösen durch Pseudo-Inverse:

$$J^+ = (J^T J)^{-1} J^T$$

$$\dot{\theta} = J^+ \dot{x}$$

## 8. Trajektoriengenerierung

### Erzeugung von Trajektorien

Ziel: Bewegung des Endeffektors von einer Aufangsposse  $\{T_A\}$  zu einer Endpose  $\{T_C\}$  mit möglichen Zwischenpunkten  $\{T_B\}$

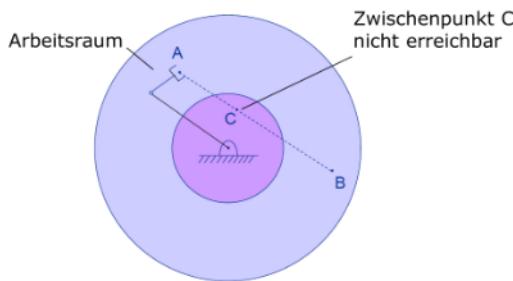
#### Nebenbedingungen:

- Räumlich (Hindernisse)
- zeitlich (Synchronisation)
- Glättheit (Energie)

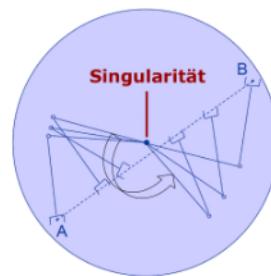
#### Ausgabe: Trajektorie

→ Position, Geschwindigkeit und Beschleunigung aller Gelenke

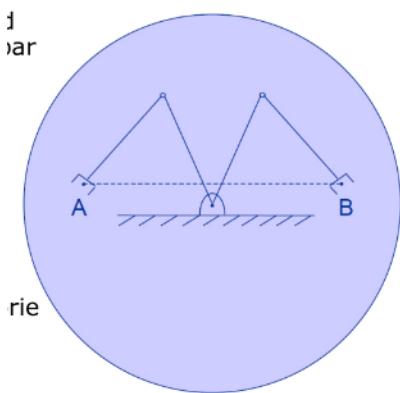
#### Nichterreichbarkeit:



#### Singularität:



#### Nicht verbundende Bereiche:



### Ansätze zur Trajektorienerzeugung

#### 1. Im Gelenakraum

- Löse inverse Kinematik an allen Pfadpunkten
- Interpoliere im Gelenakraum
- => wenig Rechenaufwand
- => keine Probleme mit Singularitäten
- => kann keiner geraden Line folgen

#### 2. Im Arbeitsraum

- Interpoliere die Koordinaten des Endeffektors
- Löse inverse Kinematik für jeden Zeitschritt
- => man kann genaue Trajektorie erzeugen
- => hoher Rechenaufwand
- => Probleme mit Erreichbarkeit, Singularitäten, mehrdeutigen Lösungen

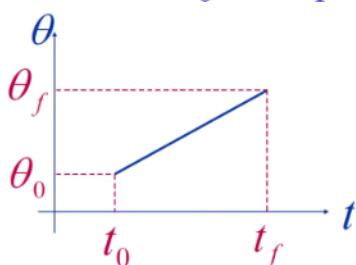
# Interpolationsmethoden

## 1. Lineare Interpolation



→ Benötigt 2 Koeffizienten:

$$\theta(t) = a_0 + a_1 t$$



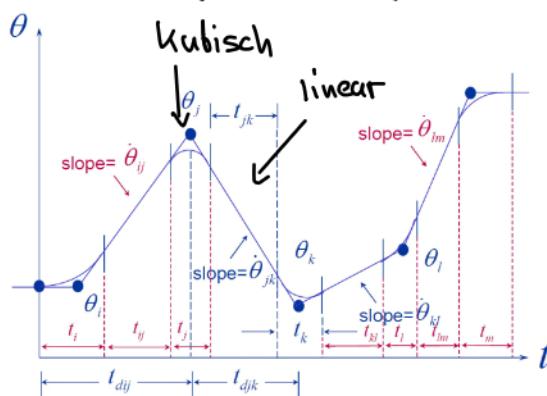
Randbedingungen

$$\theta(t_0) = \theta_0$$

$$\theta(t_f) = \theta_f$$

## 2. Lineare Interpolation mit glatten Übergängen

→ Aufteilung der Trajektorie in Teilstücke

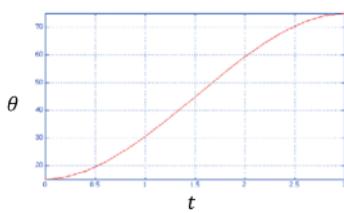


## 3. kubische Polynome



Einzelnes kubisches Modell:

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$



Lösung:  $a_0 = \theta_0$

$$a_1 = \dot{\theta}_0$$

$$a_2 = \frac{3}{t_f^2}(\theta_f - \theta_0) - \frac{2}{t_f} \dot{\theta}_0 - \frac{1}{t_f^2} \ddot{\theta}_0$$

$$a_3 = -\frac{2}{t_f^3}(\theta_f - \theta_0) + \frac{1}{t_f^2}(\dot{\theta}_f + \dot{\theta}_0)$$

Geschwindigkeit:

$$\dot{\theta}(t) = a_1 + 2a_2 t + 3a_3 t^2$$

Randbedingungen

$$\theta(0) = \theta_0 ; \quad \theta(t_f) = \theta_f \quad \dot{\theta}(0) = 0 ; \quad \dot{\theta}(t_f) = 0$$

Zwischenpunkte:

Kontinuität:  $\dot{\theta}_1(t_f) = \dot{\theta}_2(0)$

$$\ddot{\theta}_1(t_f) = \ddot{\theta}_2(0)$$

## 9. Systeme

Systeme sollen zur Regelungstechnik verwendet werden

### Konzepte zur Regelung

**Systembeschreibung:** Beschreibung von Eingabe, Ausgabe und Dynamik

$$\dot{x} = f(x, u) \quad x \in \mathbb{R}^n, x(0)$$

$$y = h(x, u) \quad u \in \mathbb{R}, y \in \mathbb{R}$$

**Stabilität:** System soll um Fixpunkt stabil sein

→ Regelungsverhalten  $u = \alpha(x)$  um den Fixpunkt

$$\lim_{t \rightarrow \infty} x(t) = x_e \text{ for all } x(0) \in \mathbb{R}^n$$

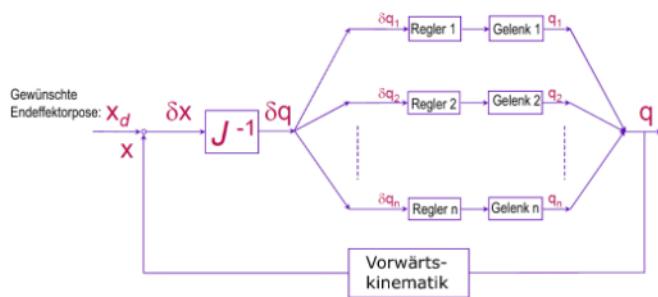
**Steuerbarkeit:** Das System soll in einen bestimmten Zustand gebracht werden → Kontrollaktion  $u(t)$  soll gefunden werden

$$\dot{x} = f(x, u(t)) \text{ das System in } x_f \text{ bringt: } x(t_0) = x_0 \rightarrow x(T) = x_f$$

**Folgeverhalten:** System soll einer Trajektorie folgen

Gegeben  $y_d(t)$ , finde  $u = \alpha(x, t)$  sodass

$$\lim_{t \rightarrow \infty} (y(t) - y_d(t)) = 0 \text{ for all } x(0) \in \mathbb{R}^n$$

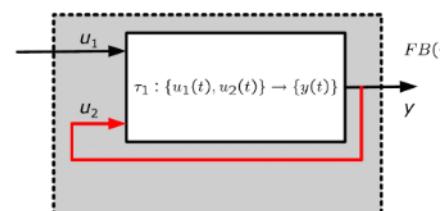


### Feedback-Systeme

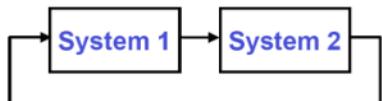
→ Position, Geschwindigkeit etc. soll durch ein System / Computer gesteuert werden

→ Rückkopplung des Zustands eines Systems

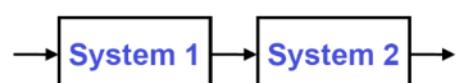
→ Feedback: Beeinflussung von mehreren Systemen gegenseitig



**Closed Loop:**



**Open Loop:**



Ziele

- statistische Performance: System hält erwünschten Zustand
- dynamische Performance: System reagiert schnell auf Veränderungen
- Robustheit: System toleriert Änderungen in der Dynamik

# Systemeigenschaften

## 1. Zustandslos vs. mit Zustand

Zustandslos: Ausgabe hängt nur von der aktuellen Eingabe ab  
Zustand: Ausgabe hängt zusätzlich von früheren Eingaben ab

## 2. Kausalität:

### System:

→ gleiche Eingabe erzeugt  
gleiche Ausgabe zu einem  
gewissem Zeitpunkt

$$u_1(t) = u_2(t) \forall t \leq \tau \Rightarrow T(u_1(t)) = T(u_2(t)) \forall t \leq \tau$$

### Signale:

$$f(t) = 0 \quad \forall t < 0$$

### Kausal machen:

$$f_c(t) = s(t)f(t)$$

## 3. Linearität

### Skalierungseigenschaft

### Superpositionseigenschaft

$$T\left\{\sum_{i=1}^n k_i u_i(t)\right\} = \sum_{i=1}^n k_i T\{u_i(t)\}$$

## 4. Zeitinvariante Systeme

→ Systemverhalten hängt nicht von der Zeit ab

$$T\{u(t - t_0)\} = y(t - t_0) \quad \forall t_0 \in T : t - t_0 \in T$$

## 5. Stabilität

→ Eingabe erzeugt endliche Ausgabe

$$|u(t)| < M < \infty \Rightarrow |T\{u(t)\}| < N < \infty$$

# Signaldekomposition

→ möglich bei **linearen Systemen**

Vorgehen:

1. Zerlege (komplexe) Eingabe in mehrere einfache Signale, für welche die Ausgabe berechnet werden kann
2. Berechne die Ausgaben getrennt
3. Addiere die einzelnen Ausgabesignale

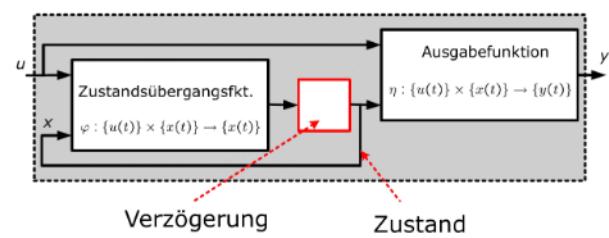
# Lineare zeitinvariante Systeme (LTI-Systeme)

Eigenschaft:

- **Linearität:** erlaubt, das Superpositionsprinzip anzuwenden
- **Zeitinvarianz:** erlaubt, Signale zeitlich zu verschieben

# 10. Dynamische Systeme

→ Systeme mit internen Zuständen



## Klassen

**Zeitdiskret:** Zustand ändert sich regelmäßig

- Zustandsübergangsfunktion als **Tabelle**:

$$\varphi : U \times X \rightarrow X$$

- Ausgabefunktion:  $\eta : X \rightarrow Y$

**Zeithintinuierlich:** Zustand ändert sich kontinuierlich

- $\varphi : X \times U \rightarrow X$   
Zustandsübergang beschrieben durch  
**Differentialgleichungen**, z.B.  $\dot{x} = f(x, u)$
- Ausgabefunktion:  
 $\eta : X \rightarrow Y$  oder  $\eta : X \times \{u(t)\} \rightarrow Y$

**Ereignisbasiert:** Zustand ändert sich bei Ereignissen

## Bestandteile eines Dynamischen Systems

**Zustand:** fasst Vergangenheit zusammen

→ unabhängige physikalische Größen, die das System beschreiben

**Eingabe:** beschreibt Anregung des Systems von außen

→ wird nicht durch Systemdynamik beeinflusst

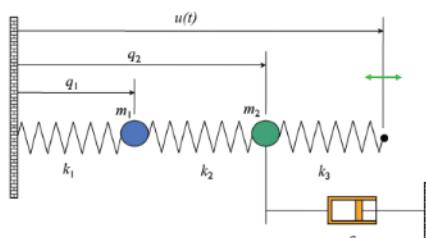
**Dynamik:** beschreibt Änderung des Zustands

→ Funktion des aktuellen Zustands und externen Eingaben

**Ausgabe:** beschreibt die gemessenen Größen

→ Funktion des Zustands und der Eingabe

## Beispiel



Beispiel: Feder-Massen-System

Zustand: Positionen und Geschwindigkeiten der Massen  $q_1, q_2, \dot{q}_1, \dot{q}_2$

Eingabe: Position der Feder am rechten Ende  $u(t)$

Dynamik: Newtonsche Mechanik

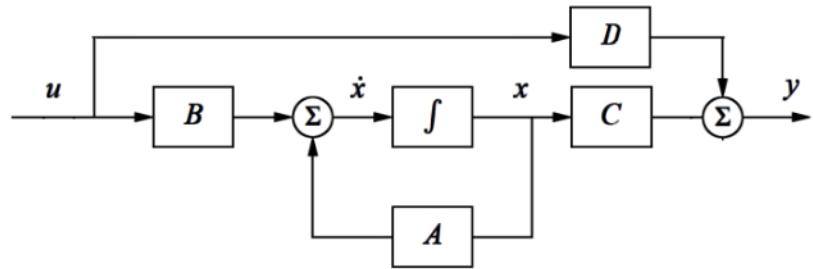
Ausgabe: Gemessene Positionen der Massen  $q_1, q_2$

## Lineare Systeme:

### Lineare Systeme

Dynamik:  $\frac{dx}{dt} = Ax + Bu$

Ausgabe:  $y = Cx + Du$



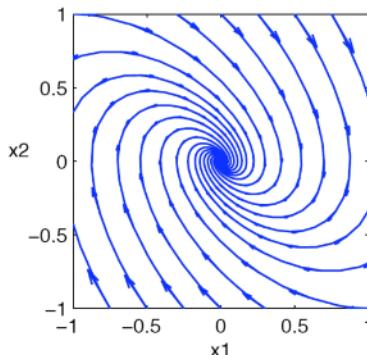
## Regler als Dynamische Systeme

- Ziel:
- Das closed-Loop soll stabil sein
  - Untersuchung des Verhaltens in dynamischen Situationen

⇒ Untersuchung durch Phasenportraits

### Phasenportraits

- Visualisierung einer 2D-Dynamik
- Darstellung autonomer Differentialgleichungen



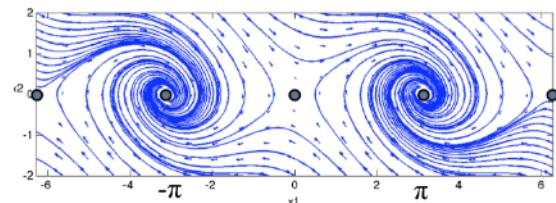
Begriffe:

Phasenraum: Menge aller möglichen Zustände  
Trajektorie: zeitliche Entwicklung eines Punktes

### Fixpunkte:

→ Fixpunkte  $x_c$  sind stationäre Zustände einer Dynamik:

Es gilt:  $\dot{x} = F(x)$        $F(x_c) = 0$



### Stabilität von Fixpunkten

**Stabil:** nahe Startzustände bleiben in der Nähe

Alle Eigenwerte haben Realteile  $\leq 0$   
& ein Eigenwert hat Realteil  $= 0$

**Asymptotisch Stabil:** Alle Anfangszustände konvergieren zum Fixpunkt

Alle Eigenwerte haben Realteile  $< 0$

**Instabil:** Es gibt Startzustände, die sich vom Fixpunkt entfernen

Es gibt einen Eigenwert mit Realteil  $> 0$

## Eigenschaften von Fixpunkten berechnen

### Lineare Systeme:

1. Bestimme Eigenwerte der Dynamikmatrix A

$$\det(\lambda \cdot I - A) = 0$$

2. Eigenwerte untersuchen

### Nicht-Lineare Systeme:

1. Linearisierung um den Fixpunkt

$$\rightarrow \text{Taylorreihenentwicklung: } T_f(x, x_e) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x_e)}{k!} (x - x_e)^k$$

$\rightarrow$  Taylorreihenentwicklung 1. Ordnung:

$$T_1 f(x, x_e) = \sum_{k=0}^1 \frac{f^{(k)}(x_e)}{k!} (x - x_e)^k = f'_1|_{x_e} \cdot (x - x_e)$$

$$\Rightarrow z = x - x_e$$

$$\dot{z} = Az$$

2. Bestimme Eigenwerte der Dynamikmatrix A

$$\det(\lambda \cdot I - A) = 0$$

3. Linearisierung instabil: Originalsystem lokal instabil

Linearisierung stabil: keine Aussage über die Stabilität des nicht-linearen Systems  
aber nicht asymptotisch stabil

## Stabilitätsanalyse mit Ljapanow-Funktion

Idee: Beschreibung der Entwicklung einer Energie im System

$\rightarrow$  Funktion soll Abstand von x zum Fixpunkt  $x_e$  beschreiben  
 $\rightarrow$  Analysiere deren Verhalten

■ Beispiel: Feder-Masse-System

- Können wir zeigen, dass alle Anfangszustände zum Ruhezustand laufen, ohne Differentialgleichungen zu lösen?

■ Systemdynamik:

$$\frac{dx}{dt} = \begin{bmatrix} x_2 \\ -\frac{k}{m}x_1 - \frac{c}{m}x_2 \end{bmatrix}$$

$$\begin{aligned} x_1 &= q \\ x_2 &= \dot{q} \end{aligned}$$

■ Berechne Energie und deren Ableitung

$$\begin{aligned} V(x) &= \frac{1}{2}kx_1^2 + \frac{1}{2}mx_2^2 & \frac{dV}{dt} &= kx_1\dot{x}_1 + mx_2\dot{x}_2 \\ &= kx_1x_2 + mx_2\left(-\frac{c}{m}x_2 - \frac{k}{m}x_1\right) & &= -cx_2^2 \end{aligned}$$

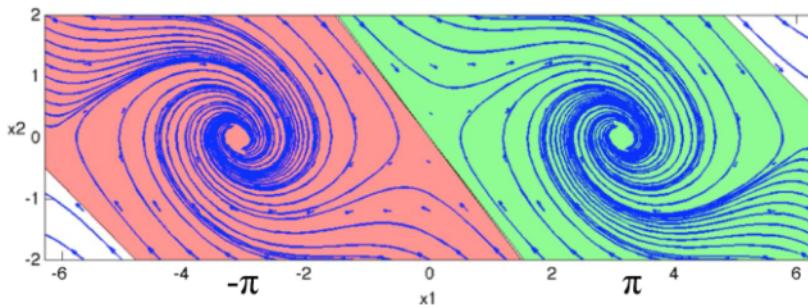
■ Energie ist nichtnegativ und fällt immer  $\Rightarrow$  muss sich Null annähern  
 $\Rightarrow x_1$  und  $x_2$  fallen auf Null

## Lokales und Globales Verhalten

→ Fixpunkte definieren lokales Verhalten

→ es kann stabile und instabile Fixpunkte geben

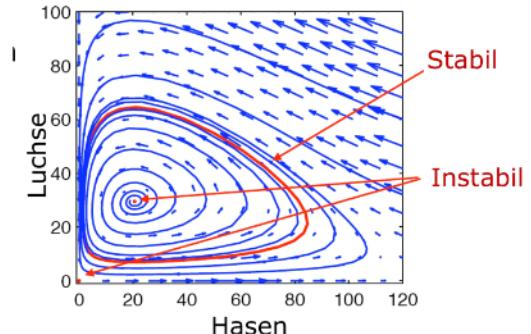
**Einzugsgebiete:** Startwerte die zu einem Fixpunkt konvergieren



## Periodische Attraktoren / Grenzzyklen

Grenzzyklus ist stabil

→ Startzustände konvergieren gegen den Grenzzyklus



## 10. Zustandsrückführung / Steuerung

### Steuerbarkeit

**Definition:** Ein System heißt steuerbar, wenn es ein Kontrollsignal  $u: [0, T] \rightarrow \mathbb{R}$  gibt, das den Systemzustand  $x(0) = x_0$  nach  $x(T) = x_f$  bewegt

### Steuerbarkeitstest

**Lineare Systeme:**

$$\dot{x} = Ax + Bu \quad x \in \mathbb{R}^n, \quad x(0)$$

$$y = Cx + Du \quad u \in \mathbb{R}, \quad y \in \mathbb{R}$$

1. **Steuerbarkeitsmatrix** aufstellen:  $[B \ AB \ A^2B \ \dots \ A^{n-1}B]$
2. **Steuerbar**  $\Leftrightarrow$  Steuerbarkeitsmatrix hat vollen Rang  
 $\Leftrightarrow \det(\text{Steuerbarkeitsmatrix}) \neq 0$

### Nicht-Lineare Systeme

1. Linearisierung um den Fixpunkt

$\rightarrow$  Taylorreihenentwicklung:  $T f(x, x_e) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x_e)}{k!} (x - x_e)^k$

$\rightarrow$  Taylorreihenentwicklung 1. Ordnung:

$$T_1 f(x, x_e) = \sum_{k=0}^1 \frac{f^{(k)}(x_e)}{k!} (x - x_e)^k = f'_{|x_e} \cdot (x - x_e)$$

$$\Rightarrow z = x - x_e$$

$$\dot{z} = Az$$

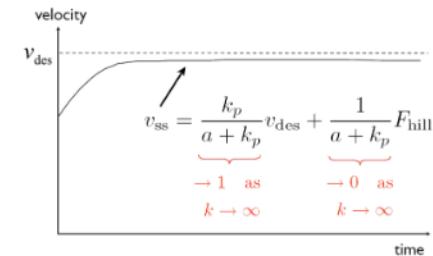
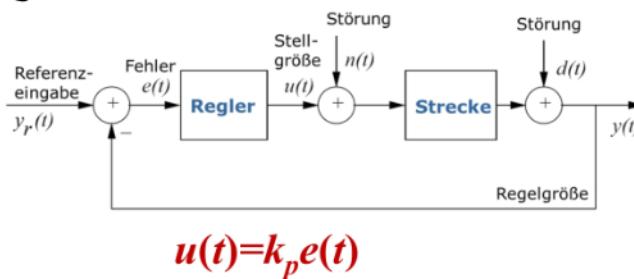
2. **Steuerbarkeitsmatrix** aufstellen:  $[B \ AB \ A^2B \ \dots \ A^{n-1}B]$

3. **lokal steuerbar**  $\Leftrightarrow$  Steuerbarkeitsmatrix hat vollen Rang

$\Leftrightarrow \det(\text{Steuerbarkeitsmatrix}) \neq 0$

## Proportional-Regler

→ Der Zustand wird zurückgeführt und mit einem Soll-Wert verglichen. Der so entstandene Fehler wird mit einem Gain-Faktor  $K$  in die Regelung eingespeist. So nähert sich der Zustand einem Soll-Wert an.

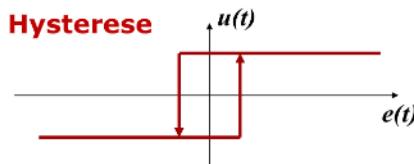


### Eigenschaften:

- Soll-Wert wird nicht ganz erreicht
- Fehler werden kleiner bei größeren  $P$

→ Überschwingen  $\Rightarrow$  mögliche Oszillation

Lösung: **Hysterese**: Tot-Bereich, in dem der Fehler nicht korrigiert wird



- kleine  $k \rightarrow$  Fehler wird zu langsam korrigiert

### Design eines Reglers für ein lineares System

**Gegeben:**  $\dot{x} = Ax + Bu$   
 $y = Cx + Du$

**Ziel:** Find ein Reglerverhalten:  $u = -Kx$   
 sodass:  $\dot{x} = Ax - BKx = (A - BK)x$

am Nullpunkt  $x=0$  stabil ist  
 $\Rightarrow$  Eigenwert müssen  $\leq 0$

1. Matrix  $\tilde{A} = A - BK$  aufstellen

2. Eigenwerte berechnen mit Hilfe des charakteristischen Polynoms

$$\det(\tilde{A} - \lambda \cdot I)$$

3. Gain-Faktoren so wählen, dass die Realteile der Eigenwerte  $< 0$  sind

1. Möglichkeit:  $\det(\tilde{A} - \lambda \cdot I) = 0$

→ Nach  $\lambda$  auflösen und  $K$  nach der obigen Bedingung wählen

2. Möglichkeit: charakteristisches Polynom mit einem Polynom vergleichen, das entsprechende Eigenwerte hat und dem entsprechenden  $K$  wählen. z.B.:  $(x+1)^4$

4. Regler mit den Gain-Faktoren bauen

## Normalform zur Steuerung

→ Koordinatenänderung durch Transformation im Zustandsraum z:

$$T: \mathbf{z} = \mathbf{T}\mathbf{x}$$

→ Jede Zustandsdimension kann eine Höhere beeinflussen

### Normalform:

$$\frac{dz}{dt} = \begin{bmatrix} -a_1 & -a_2 & -a_3 & \dots & -a_n \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & & & 1 & 0 \end{bmatrix} z + \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} b_1 & b_2 & b_3 & \dots & b_n \end{bmatrix} z + du.$$

### Charakteristisches Polynom:

$$\lambda(s) = s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n$$

### Steuerbarkeitsmatrix (voller Rang):

$$W_r = \begin{bmatrix} B & AB & \dots & A^{n-1}B \end{bmatrix} = \begin{bmatrix} 1 & -a_1 & a_1^2 - a_2 & \dots & * \\ 0 & 1 & -a_1 & \dots & * \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 1 & * \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}$$

### Transformation in die Normalform:

Ziel: T finden

- Bestimmung der  $a_1, \dots, a_n$  Koeffizienten aus dem char. Polynoms von A
- Aufstellen der Dynamik in z mit A, B

$$z = Tx \quad \tilde{A} = TAT^{-1}, \quad \tilde{B} = TB$$

- Steuerbarkeitsmatrix aufstellen und vereinfachen

$$\tilde{W}_r = \begin{bmatrix} \tilde{B} & \tilde{A}\tilde{B} & \dots & \tilde{A}^{n-1}\tilde{B} \end{bmatrix}$$

$$\tilde{A}\tilde{B} = TAT^{-1}TB = TAB$$

$$\tilde{A}^2\tilde{B} = (TAT^{-1})^2TB = TAT^{-1}TAT^{-1}TB = TA^2B$$

⋮

$$\tilde{A}^n\tilde{B} = TA^nB$$

$$\tilde{W}_r = T \begin{bmatrix} B & AB & \dots & A^{n-1}B \end{bmatrix} = TW_r$$

- Nach T auflösen:  $T = \tilde{W}_r W_r^{-1}$

und Koordinaten transformieren

## Beispiel:

- Gegeben: Lineares System  $\frac{dx}{dt} = \begin{pmatrix} \alpha & \omega \\ -\omega & \alpha \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u$
- Gesucht: Normalform  $\tilde{A} = \begin{pmatrix} -a_1 & -a_2 \\ 1 & 0 \end{pmatrix}$ ,  $\tilde{B} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
- Bestimmung der Koeffizienten aus charakt. Polynom:  
 $\lambda(s) = \det(sI - A) = s^2 - 2as + (\alpha^2 + \omega^2) \implies a_1 = -2a, a_2 = \alpha^2 + \omega^2$
- Steuerbarkeitsmatrix:  
 $W_r = \begin{pmatrix} 0 & \omega \\ 1 & \alpha \end{pmatrix}, \quad \tilde{W}_r = \begin{pmatrix} 1 & -a_1 \\ 0 & 1 \end{pmatrix}$
- Transformation:  
 $T = \tilde{W}_r W_r^{-1} = \begin{pmatrix} -(a_1 + \alpha)/\omega & 1 \\ 1/\omega & 0 \end{pmatrix} = \begin{pmatrix} \alpha/\omega & 1 \\ 1/\omega & 0 \end{pmatrix}$
- Koordinaten:  
 $\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = Tx = \begin{pmatrix} ax_1/\omega + x_2 \\ x_2/\omega \end{pmatrix}$

## Kanonische Zustandsrückführung

### open Loop:

$$\frac{dz}{dt} = \tilde{A}z + \tilde{B}u = \begin{bmatrix} -a_1 & -a_2 & -a_3 & \dots & -a_n \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & & & 1 & 0 \end{bmatrix} z + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} u$$

$$y = \tilde{C}z = \begin{bmatrix} b_1 & b_2 & \dots & b_n \end{bmatrix} z.$$

charakteristisches Polynom:  $\lambda(s) = s^n + a_1s^{n-1} + \dots + a_{n-1}s + a_n$

Regler:  $u = -\tilde{K}z + k_r r = -\tilde{k}_1 z_1 - \tilde{k}_2 z_2 - \dots - \tilde{k}_n z_n + k_r r$

### closed Loop:

$$\frac{dz}{dt} = \begin{bmatrix} -a_1 - \tilde{k}_1 & -a_2 - \tilde{k}_2 & -a_3 - \tilde{k}_3 & \dots & -a_n - \tilde{k}_n \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & & & 1 & 0 \end{bmatrix} z + \begin{bmatrix} k_r \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} r$$

$$y = \begin{bmatrix} b_n & \dots & b_2 & b_1 \end{bmatrix} z.$$

charakteristisches Polynom:  $s^n + (a_1 + \tilde{k}_1)s^{n-1} + (a_2 + \tilde{k}_2)s^{n-2} + \dots + (a_{n-1} + \tilde{k}_{n-1})s + a_n + \tilde{k}_n$

Gewünschtes Polynom:  $p(s) = s^n + p_1s^{n-1} + \dots + p_{n-1}s + p_n$

Wahl von  $K$ :  $\tilde{k}_1 = p_1 - a_1, \quad \tilde{k}_2 = p_2 - a_2, \quad \dots \quad \tilde{k}_n = p_n - a_n$

$$\tilde{K} = \begin{bmatrix} p_1 - a_1 & p_2 - a_2 & \dots & p_n - a_n \end{bmatrix} \quad k_r = \frac{a_n + \tilde{k}_n}{b_n} = \frac{p_n}{b_n}$$

## Zuweisung von Eigenwerten

Lineares System:  $\frac{dx}{dt} = Ax + Bu, \quad y = Cx + Du$

Koordinatentransformation in die Normalform:  $z = Tx$

Regler:  $u = -\tilde{K}z + k_r r = -\tilde{K}Tx + k_r r$

Wenn  $(A, B)$  steuerbar, existiert ein Regler  $u = -Kx + k_r r$   
mit charakteristischem Polynom:

$$p(s) = s^n + p_1s^{n-1} + \dots + p_{n-1}s + p_n$$

$$\Rightarrow \lambda(s) = \det(sI - A + BK)$$

Ackermann-Formel:

$$K = \tilde{K}T = \begin{bmatrix} p_1 - a_1 & p_2 - a_2 & \dots & p_n - a_n \end{bmatrix} \tilde{W}_r W_r^{-1}, \quad k_r = \frac{p_n}{a_n}$$

$$W_r = \begin{bmatrix} B & AB & \dots & A^{n-1}B \end{bmatrix}, \quad \tilde{W}_r = \begin{bmatrix} 1 & a_1 & a_2 & \dots & a_{n-1} \\ 0 & 1 & a_1 & \dots & a_{n-2} \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & a_1 \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}^{-1}$$

# Regler mit Integralteil

**Problem:** Die Zustandsrückführung erreicht den Soll-Wert nur bei genauer Kalibrierung des Referenzgains

**Idee:** Integriere Fehler in zusätzlicher Zustandsdimension  
Fehler wird verstärkt, desto länger er besteht.

$$\frac{d}{dt} \begin{bmatrix} x \\ z \end{bmatrix} = \begin{bmatrix} Ax + Bu \\ y - r \end{bmatrix} = \begin{bmatrix} Ax + Bu \\ Cx - r \end{bmatrix}$$

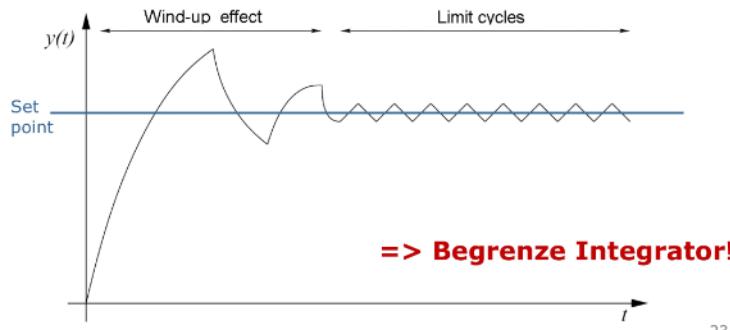
→ Wenn das System eingeschwungen ist, ist  $\dot{z}=0$  und  $z=r$

**Zustandsrückführung:**  $u = -Kx - \underline{k_i z} + k_r r$

**Fixpunkt:**  $x_e = -(A - BK)^{-1} B(k_r r - \underline{k_i z_e})$

## Wind-up-Effekt

- Stellgröße  $u(t)$  ist begrenzt → nicht linear
- Integrator akkumuliert Regelabweichung  $e(t)$
- Integrator ist zeitlich versetzt, Fehler werden nicht direkt erkannt
- Überschwingen



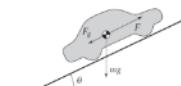
23

## Beispiel:

- Linearisierte Dynamik um Fixpunkt:  $v_e, u_e$

$$\frac{dx}{dt} = ax - b_g \theta + bw, \quad y = v = x + v_e$$

$$x = v - v_e, w = u - u_e$$



- Zustand erweitert durch Integrator:

$$\frac{dx}{dt} = ax - b_g \theta + bw, \quad \frac{dz}{dt} = y - v_r = v_e + x - v_r$$

$$\frac{d}{dt} \begin{bmatrix} x \\ z \end{bmatrix} = \begin{bmatrix} a & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} b \\ 0 \end{bmatrix} w + \begin{bmatrix} -b_g \\ 0 \end{bmatrix} \theta + \begin{bmatrix} 0 \\ v_e - v_r \end{bmatrix}$$

- Regler mit Zustandsrückführung:

$$w = -k_p x - k_i z + k_r v_r$$

- Wähle Gain-Faktoren  $k_p, k_i$  und  $k_r$  so, dass System stabil ist und Referenz folgt

2:

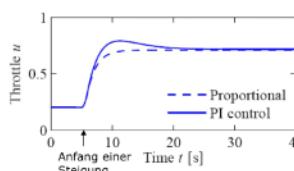
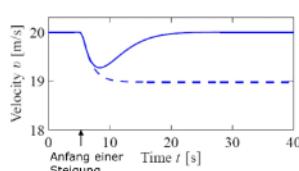
- Gewünschtes charakteristisches Polynom:

$$\lambda(s) = s^2 + a_1 s + a_2$$

- Mit Störung  $\theta = 0$ :

$$\det(sI - (A - BK)) = s^2 + (bk_p - a)s + bk_i$$

$$k_p = \frac{a_1 + a}{b}, \quad k_i = \frac{a_2}{b}, \quad k_r = -1/(C(A - BK)^{-1} B) = \frac{a}{b}$$

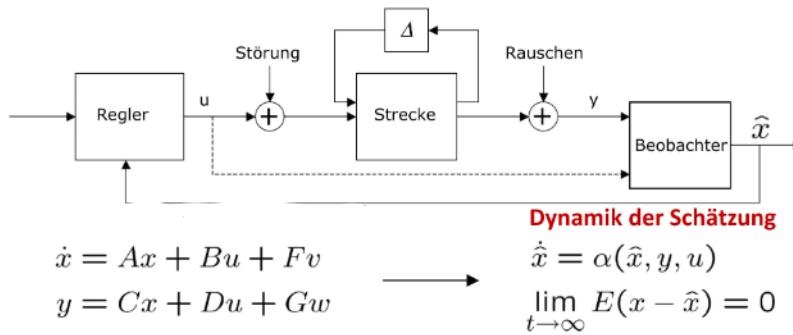


## II. Beobachter

**Problem:** Der Zustand eines Systems kann nicht immer direkt/genau gemessen werden

**Idee:** Der Zustand soll geschätzt werden aus gegebenen Messungen  
Dabei müssen Unsicherheiten, die meist nur aus Effekten in der Ausgabe ersichtlich sind.

Außerdem muss die Ausgabe genug Informationen enthalten um den Zustand zu schätzen



## Beobachtbarkeit

**Definition:** Ein dynamisches System heißt beobachtbar, wenn es für jedes  $T > 0$  möglich ist, den Zustand  $x(T)$  aus den Messungen  $y(T)$  und  $u(T)$  aus dem Intervall  $[0, T]$  zu bestimmen

### Beobachtbarkeitstest:

→ Taylorentwicklung

### Beobachtbarkeitsmatrix:

$$\begin{array}{ll} \dot{x} = Ax + Bu & y = \underline{Cx} \\ y = Cx & \dot{y} = \underline{C\dot{x}} = \underline{CAx} + \underline{CBu} \\ & \ddot{y} = \underline{CA^2x} + CABu + CB\dot{u} \\ & \vdots \end{array} \quad W_o = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

System ist **beobachtbar**  $\Leftrightarrow$  Beobachtbarkeitsmatrix hat vollen Rang

$$\Leftrightarrow \det(W_o) \neq 0$$

## Normalform zur Beobachtung

$$\frac{dz}{dt} = \begin{pmatrix} -a_1 & 1 & 0 & \cdots & 0 \\ -a_2 & 0 & 1 & & 0 \\ \vdots & & \ddots & & \\ -a_{n-1} & 0 & 0 & & 1 \\ -a_n & 0 & 0 & \cdots & 0 \end{pmatrix} z + \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-1} \\ b_n \end{pmatrix} u$$

$$y = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \end{pmatrix} z + Du.$$

**char. Polynom:**  $\lambda(s) = s^n + a_1s^{n-1} + \cdots + a_{n-1}s + a_n$

**Beobachtbarkeitsmatrix:**

$$W_o = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -a_1 & 1 & 0 & \cdots & 0 \\ -a_1^2 - a_1a_2 & -a_1 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ * & * & \cdots & 1 \end{bmatrix}$$

$$W_o^{-1} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ a_1 & 1 & 0 & \cdots & 0 \\ a_2 & a_1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ a_{n-1} & a_{n-2} & a_{n-3} & \cdots & 1 \end{bmatrix}$$

## Design von Beobachtern

Gegeben:  $\dot{x} = Ax + Bu$

$$y = Cx$$

Form des Beobachters:

$$\dot{\hat{x}} = \underbrace{A\hat{x} + Bu}_{\text{Zustandsvorhersage}} + L(y - C\hat{x})$$

Korrektur  
durch  
Ausgabefehler

→  $L$  ist die Gain-Matrix des Beobachters:

Beschreibt Änderung der Zustandsschätzung auf Grund des Ausgabefehlers

Beobachtungsfehler:

$$\tilde{x} = x - \hat{x}$$
 soll möglichst klein werden

Zeitliche Ableitung:  $\dot{\tilde{x}} = \dot{x} - \dot{\hat{x}} = Ax + Bu - (A\hat{x} + Bu + LC(x - \hat{x})) = (A - LC)\tilde{x}$

Ziel:  $\lim_{t \rightarrow \infty} x - \hat{x} = 0$

⇒ Das gilt genau dann, wenn alle Eigenwerte der Matrix  $(A - LC)$  negative Realteile haben

Vorgehensweise:

1.  $A - LC$  bestimmen

2.  $\det((A - LC) - \lambda \cdot I)$  bestimmen

3.  $L$  wählen, sodass  $\operatorname{Re}(\lambda) < 0$  sind

1. Möglichkeit:  $\det((A - LC) - \lambda \cdot I) = 0$  setzen und nach  $\lambda$  umformen  
→  $L$  entsprechend wählen

2. Möglichkeit: char. Polynom mit einem gewünschten Polynom vergleichen

4. Beobachter mit  $L$  designen

# Vergleich von Beobachter und Steuerbarkeit und Regelung

Es gilt:

1.  $(A, C)$  beobachtbar  $\Leftrightarrow (A^T, C^T)$  steuerbar
2. Eigenwerte von  $(A - LC)$  können frei gewählt werden  $\Leftrightarrow$  Eigenwerte von  $(A^T - C^T L^T)$  können frei gewählt werden
3. (Steuerbarkeit und Beobachtbarkeit sind Eigenschaften des Systems, d.h. hängen nicht von gewähltem Zustandsraum ab)
4. Auslegung von Beobachtern ist **dual** zu Auslegung von Reglern mit Zustandsrückführung

## Separationsprinzip

**Lege Regler und Beobachter getrennt aus.**

Dann kombiniere beide und das Gesamtsystem ist stabil

→ Wenn  $K$  ein stabiler Regler für  $(A, B)$  ist und  $L$  ein stabiler Beobachter für  $(A, C)$  ist

Dann ist:

$$u = -K(\hat{x} - x_d) + u_d \text{ stabil (wenn } x_d, u_d \text{ ein Fixpunkt)}$$

Also: Regelung auf Grundlage des Beobachters

Beweis:

Systemdynamik für das Gesamtsystem (o.B.d.A  $x_d = 0$ ):

$$\begin{aligned} \dot{x} &= Ax + Bu & \dot{\hat{x}} &= A\hat{x} + Bu + L(y - C\hat{x}) \\ y &= Cx & u &= -K\hat{x} + u_d \end{aligned}$$

Betrachte Fehlerdynamik  $\tilde{x} = x - \hat{x}$   
und kombinierten Zustand  $x, \tilde{x}$ :

$$\dot{\tilde{x}} = (A - LC)\tilde{x} \quad \frac{d}{dt} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} + \begin{bmatrix} u_d \\ 0 \end{bmatrix}$$

Charakteristisches Polynom des Gesamtsystems:

$$\det(sI - A + BK) \det(sI - A + LC).$$

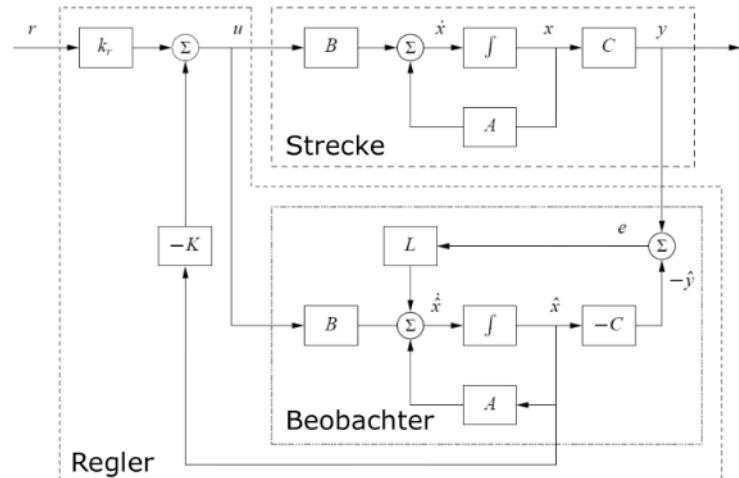
Dies ist ein Produkt aus zwei Polynomen, die dem Regler und dem Beobachter entsprechen.

=> Wenn beide stabil, dann auch Gesamtsystem.

## Beobachter-basierter Regler

→ Schätzung wird für den Regler mit Zustandsrückführung verwendet

→ Regler beinhaltet ein Modell des geregelten Prozesses



## Vorgehensweise

1. Beobachter und Regler getrennt designen
2. Regler in Beobachter einsetzen

## Beispiel

### Beispiel: Fahrzeuglenkung

- Linearisierte seitliche Dynamik:

$$\frac{dx}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} \gamma \\ 1 \end{bmatrix} u, \quad y = \begin{bmatrix} 1 & 0 \end{bmatrix} x \quad \begin{array}{l} x_1 \dots \text{seitliche Position} \\ x_2 \dots \text{seitliche Geschwindigkeit} \end{array}$$

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

- Charakteristisches Polynom des Beobachters  $A - LC = \begin{bmatrix} -l_1 & 1 \\ -l_2 & 0 \end{bmatrix}$ :

$$\det(sI - A + LC) = \det \begin{bmatrix} s + l_1 & -1 \\ l_2 & s \end{bmatrix} = s^2 + l_1 s + l_2$$

- Gewünschtes charakteristisches Polynom:  $s^2 + p_1 s + p_2 = s^2 + 2\zeta_o \omega_o s + \omega_o^2$

- Beobachter-Gain:  $l_1 = p_1 = 2\zeta_o \omega_o, \quad l_2 = p_2 = \omega_o^2$

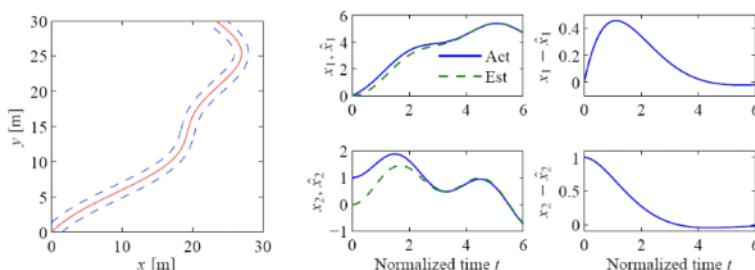
- Beobachter:

$$\frac{d\hat{x}}{dt} = A\hat{x} + Bu + L(y - C\hat{x}) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \hat{x} + \begin{bmatrix} \gamma \\ 1 \end{bmatrix} u + \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} (y - \hat{x}_1)$$

- Kurvige Straße

- Initialer Geschwindigkeitsfehler

$$\frac{d\hat{x}}{dt} = A\hat{x} + Bu + L(y - C\hat{x}) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \hat{x} + \begin{bmatrix} \gamma \\ 1 \end{bmatrix} u + \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} (y - \hat{x}_1)$$



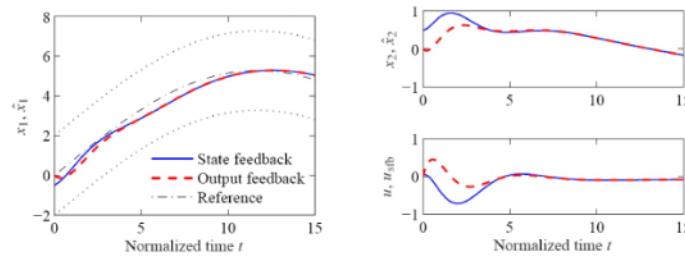
- Dynamik, die Lenkwinkel  $u$  zur seitlichen Abweichung  $y$  macht:

$$\frac{d\hat{x}}{dt} = A\hat{x} + Bu + L(y - C\hat{x}) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \hat{x} + \begin{bmatrix} \gamma \\ 1 \end{bmatrix} u + \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} (y - \hat{x}_1)$$

$$u = -K\hat{x} + k_r r = k_1(r - x_1) - k_2 x_2$$

- $u$  einsetzen:  $\frac{d\hat{x}}{dt} = (A - BK - LC)\hat{x} + Ly + Bk_r r$

$$= \begin{bmatrix} -l_1 - \gamma k_1 & 1 - \gamma k_2 \\ -k_1 - l_2 & -k_2 \end{bmatrix} \hat{x} + \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} y + \begin{bmatrix} \gamma \\ 1 \end{bmatrix} k_1 r$$



## 12. Konvolution

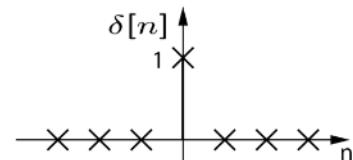
→ ist auf LTI-Systemen definiert

### Einheitsimpuls

→ Ableitung der Sprungfunktion

### zeitdiskreter Impuls

**Definition:**  $\delta[n] = \begin{cases} 1 & \text{für } n = 0 \\ 0 & \text{für } n \neq 0 \end{cases}$



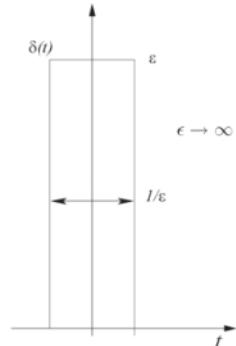
→ springt bei 0 kurz auf 1

$\delta[n]$  wird auch als **Dirac-Impuls** bezeichnet

### Zeitkontinuierlicher Impuls

**Definition:** 1.  $\delta(t) = \begin{cases} 0 & \forall t \neq 0 \\ \infty & t = 0 \end{cases}$  2.  $\int_{-\infty}^{\infty} \delta(t) dt = 1$

$\delta(t)$  wird auch **Dirac-Delta** genannt

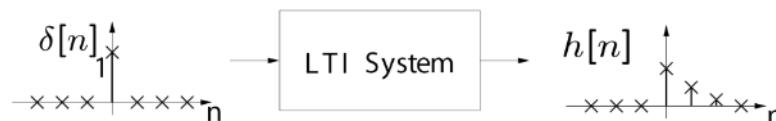


### Impulsantwort

→ beschreibt LTI-Systeme vollständig

**Definition:** Ausgabe  $h[n]$  ist die Ausgabe eines Systems bei der Eingabe des Einheitsimpulses  $\delta[n]$

$$h[n] = \mathcal{T}\{\delta[n]\}$$



**Berechnung:**  $y[n] = u[n] - 3/4 \cdot y[n-1]$

1. Darstellung der Ausgabe finden, die nicht mehr von der Ausgabe abhängt

$$\rightarrow \text{Reihendarstellung} \quad y[n] = \sum_{k=0}^{\infty} (-1)^k \left(\frac{3}{4}\right)^k \cdot u[n-k]$$

2. Darstellung umformen, sodass sie der Konvolution gleicht

$$y[n] = \sum_{k=-\infty}^{\infty} s[k] \left(-\frac{3}{4}\right)^k u[n-k]$$

3. Teilterm, der nicht die Eingabe ist, als Impulsantwort interpretieren

$$y[n] = \sum_{k=-\infty}^{\infty} s[k] \underbrace{\left(-\frac{3}{4}\right)^k}_{h[n]} u[n-k] \Rightarrow h[n] = s[n] \cdot \left(-\frac{3}{4}\right)^n$$

## Sprungantwort

- Ausgabe  $a[n]$  eines Systems bei Eingabe der Sprungfunktion  $s[n]$
- beschreibt LTI-Systeme vollständig

**Definition**  $a[n] = \mathcal{T}\{s[n]\}$

**Berechnung:**

1.  $\mathcal{T}\{s[n]\}$  mit Hilfe der Konvolution berechnen

$$a[n] = \mathcal{T}\{s[n]\} = h[n] * s[n] = \sum_{k=-\infty}^{\infty} s[k] \left(-\frac{3}{4}\right)^k s[n-k]$$

2. Term so weit, wie möglich umformen

$$a[n] = \sum_{k=0}^{\infty} \left(-\frac{3}{4}\right)^k s[n-k] = \sum_{k=0}^n \left(-\frac{3}{4}\right)^k = \frac{1 - \left(-\frac{3}{4}\right)^{n+1}}{1 + \frac{3}{4}}$$

## Signalzerlegung

Beliebige Folgen  $f[n]$  können als Linear kombination von Einheitsimpulsen  $s[n]$  dargestellt werden

$$f[n] = \sum_{\nu=-\infty}^{\infty} f[\nu] \delta[n-\nu]$$

$f[\nu] \rightarrow$  Wert der Folge an Stelle  $n=\nu$  (Skalierung)

$\delta[n-\nu] \rightarrow$  Einheitsimpuls wird an Stelle  $n=\nu$  verschoben

## Konvolution / Faltung

→ Konvolution bildet die Signale  $u[n]$  und  $h[n]$  auf eine Ausgabe  $y[n]$  ab

## zeitdiskrete Konvolution

**Definition:**  $y[n] = u[n] * h[n] : y[n] = \sum_{\nu=-\infty}^{\infty} u[\nu] h[n-\nu]$

## zeitkontinuierliche Konvolution

**Definition:**  $y(t) = h(t) * u(t) : y(t) = \int_{-\infty}^{\infty} u(\tau) h(t-\tau) d\tau$

**Eigenschaften:** (beide Fälle)

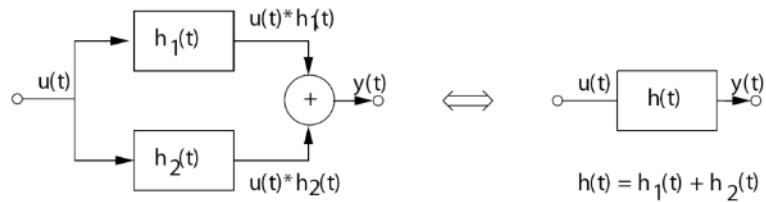
**Kommutativität:**  $x(t) * h(t) = h(t) * x(t) : \int_{-\infty}^{\infty} u(\tau) h(t-\tau) d\tau = \int_{-\infty}^{\infty} h(\tau) u(t-\tau) d\tau$

**Assoziativität:**  $x(t) * [h_1(t) * h_2(t)] = [x(t) * h_1(t)] * h_2(t)$

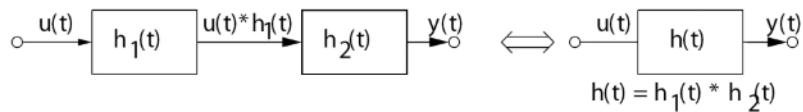
**Distributivität:**  $x(t) * [h_1(t) + h_2(t)] = [x(t) * h_1(t)] + [x(t) * h_2(t)]$

# Kombination von Teilsystemen

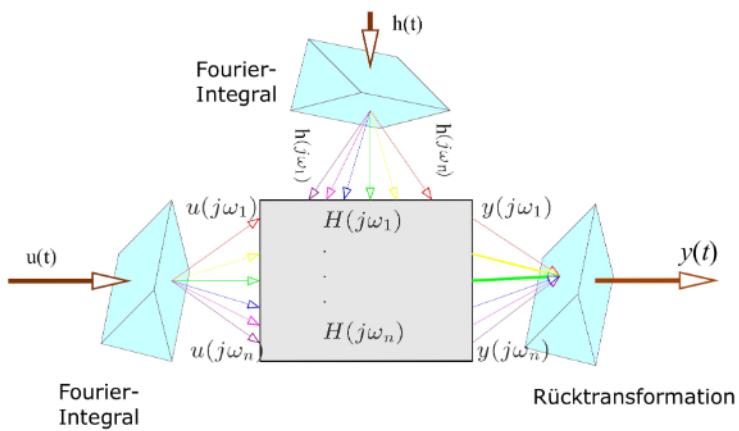
## Parallel: Addition



## Seriell: Faltung



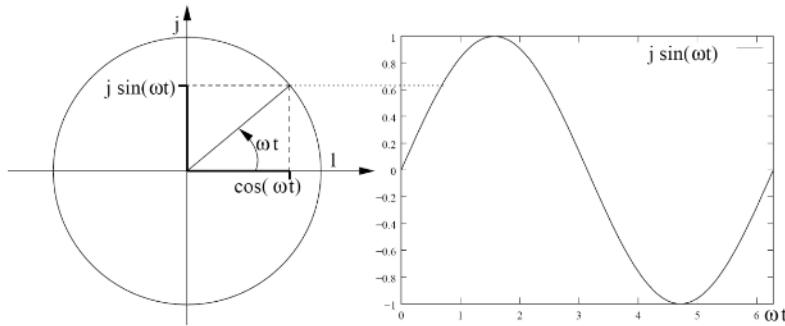
## 13. Frequenzraum (zeitkontinuierlich)



### Komplexe Exponentialfunktion

Euler'sche Formel  $\cos(\omega t) + j \sin(\omega t) = e^{j\omega t}$

Realteil:  $\cos(\omega t) = \operatorname{Re}\{e^{j\omega t}\}$



### Eigenfunktion von LTI-Systemen

→ LTI-Systeme liefern bei einem sinusoiden Signal  $u(t) = \hat{u} e^{j\omega t}$  als Eingabe eine sinusoidale Ausgabe gleicher Frequenz, multipliziert mit  $H \in \mathbb{C}$

$$y(t) = \mathcal{T}\{u(t)\} = H \hat{u} e^{j\omega t}$$

$H$  ändert:

- Phase
- Amplitude

### Klirrfaktor

→ Charakterisierung der Nichtlinearität eines Systems

Nichtlinearität erzeugt harmonisch Oberwellen, deren Frequenzen ganzzahlige Vielfache der Eingabefrequenz sind

Definition: Verhältnis der harmonischen Sinuside mit Frequenz  $\neq \omega$  zum Gesamtsignal

$$k = \frac{\bar{U}_{\text{Harmonics}}}{\bar{U}_{\text{Entire_Signal}}} = \frac{\sqrt{u_2^2 + u_3^2 + u_4^2 \dots}}{\sqrt{u_1^2 + u_2^2 + u_3^2 + u_4^2 \dots}}$$

## Transferfunktion $H(j\omega)$

→ ist charakterisiert durch  $h(t)$

Eingabe:  $u(t) = \hat{u}e^{j\omega t}$

erwartete Ausgabe:  $Hu(t)$

### Berechnung

$$y(t) = h(t) * u(t) = \int_{-\infty}^{\infty} h(\tau) \cdot u(t-\tau) d\tau$$

$$= \int_{-\infty}^{\infty} h(\tau) \cdot \hat{u} e^{j\omega(t-\tau)} d\tau = \underbrace{\int_{-\infty}^{\infty} h(\tau) e^{-j\omega\tau} d\tau}_{H(j\omega)} \underbrace{\hat{u} e^{j\omega t}}_{u(t)}$$

### Fourier-Integral

→ Mit dem Fourier-Integral kann man die Transferfunktion  $H(j\omega)$  berechnen:

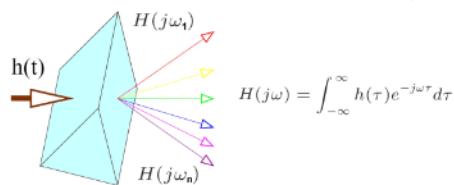
$$H(j\omega) = \int_{-\infty}^{\infty} h(\tau) e^{-j\omega\tau} d\tau$$

→ Damit können leicht Ausgaben für sinusoide Eingaben ohne Faltung berechnet werden:

$$y(t) = H(j\omega)u(t) \Big|_{u(t)=\hat{u}e^{j\omega t}}$$

- Skalierung der Amplitude  
- Phasenverschiebung

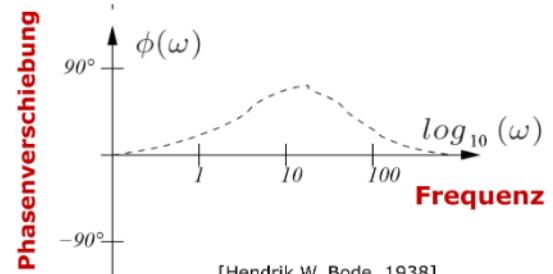
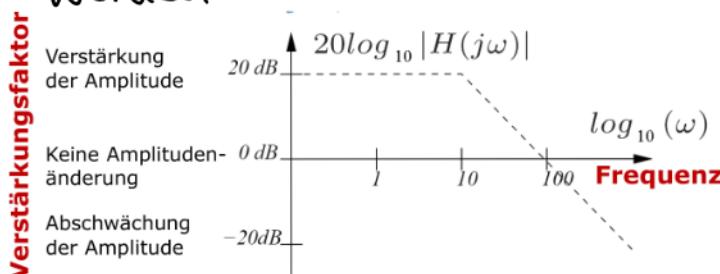
**Prinzip:** Impulsantwort wird in einzelne Frequenzen aufgespalten



Systemantwort kann für jede Eingabefrequenz  $j\omega$  einzeln durch Multiplikation mit  $H(j\omega)$  berechnet werden

### Bode-Diagramm: Grafische Darstellung

→ Zeigt welche Frequenzen verstärkt bzw gedämpft werden und wie die Frequenz in der Phase verschoben werden



[Hendrik W. Bode, 1938]

## Anwendung auf Signale:

$H(j\omega)$  repräsentiert Signale  $f(t)$  als Superposition einzelner Frequenzen mit Gewicht

$$F(j\omega) = \int_{-\infty}^{\infty} f(\tau) e^{-j\omega\tau} d\tau$$

In der Gegenrichtung kann  $f(t)$  als Integral über verschiedene Frequenzen repräsentiert werden

$$f(t) = 1/(2\pi) \int_{-\infty}^{\infty} F(j\omega) e^{j\omega t} d\omega$$

## Fourier-Transformation

→ Anwendung auf  $h(t)$  und  $u(t)$  kann als Transformation verstanden werden:

Notation:  $f(t) \xrightarrow{\quad} F(j\omega)$  (Transformation in Frequenzraum)

$F(j\omega) \xrightarrow{\quad} f(t)$  (Transformation in Zeit-Domäne)

Dualität: Vorwärts- und Rückwärtstransformation entscheiden sich hauptsächlich durch  $1/(2\pi)$

$$f(t) \xrightarrow{\quad} F(j\omega) \Leftrightarrow F(jt) \xrightarrow{\quad} 2\pi f(-\omega)$$

## Bekannte Transformationen

### Signal in der Zeit      Frequenzdarstellung

$\delta(t)$	$\xrightarrow{\quad} 1$
1	$\xrightarrow{\quad} 2\pi\delta(\omega)$
$s(t)$	$\xrightarrow{\quad} \pi\delta(\omega) + \frac{1}{j\omega}$
$\text{rect}(t/T)$	$\xrightarrow{\quad} T \text{ si}(\omega T/2)$ $\text{si}(x)$ steht für $\sin(x)/x$
$e^{j\omega_0 t}$	$\xrightarrow{\quad} 2\pi\delta(\omega - \omega_0)$
$\cos(\omega_0 t)$	$\xrightarrow{\quad} \pi[\delta(\omega - \omega_0) + \delta(\omega + \omega_0)]$
$\sin(\omega_0 t)$	$\xrightarrow{\quad} \pi/j[\delta(\omega - \omega_0) - \delta(\omega + \omega_0)]$

Zeitliche Ableitung:  $\frac{d}{dt} f(t) \xrightarrow{\quad} j\omega F(j\omega)$

Linearität:

$$k_1 f_1(t) + k_2 f_2(t) \xrightarrow{\quad} k_1 F_1(j\omega) + k_2 F_2(j\omega)$$

Zeitliche Skalierung:

$$f(at) \xrightarrow{\quad} \frac{1}{|a|} F(j\frac{\omega}{a}), \quad a \in \mathbb{R}$$

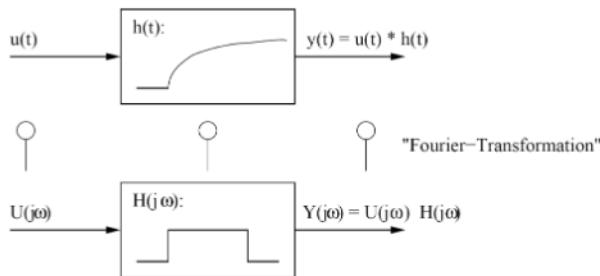
## Verschiebungsinvarianz

→ Zeitverschiebung erzeugt lediglich eine Phasenverschiebung

$$f(t - t_0) \xrightarrow{\quad} e^{-j\omega t_0} F(j\omega)$$

## Konvolutionstheorem

Die Berechnung der Ausgabe entspricht der Konvolution im Zeitraum und **Multiplikation im Frequenzraum**



$$Y(j\omega) = U(j\omega)H(j\omega) \xrightarrow{\quad} y(t) = u(t) * h(t)$$

**Multiplikation im Frequenzraum**

**Konvolution im Zeitraum**

**Beispiel:** Gegeben:  $u(t) = \delta(t)$  und  $H(j\omega) = \text{rect}(j\omega)$   
(idealer Tiefpass)

Gesucht:  $Y(j\omega)$  und  $y(t)$

Lösung:

$$Y(j\omega) = U(j\omega)H(j\omega) = 1 \text{ rect}(j\omega) = \text{rect}(j\omega)$$

Mit  $\text{rect}(t) \xrightarrow{\quad} \text{si}(\omega/2)$  erhalten wir durch Dualität

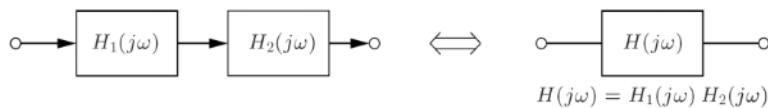
$$\text{rect}(j\omega) \xrightarrow{\quad} 1/(2\pi) \text{ si}(t/2)$$

Impulsantwort des idealen Tiefpasses:

$$y(t) = 1/(2\pi) \text{ si}(t/2)$$

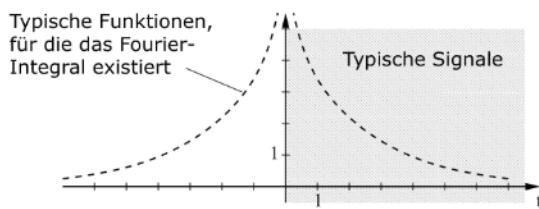
## Serielle Kombination

$$H(j\omega) = H_1(j\omega) H_2(j\omega)$$



## Problem: Konvergenz

Konvergenz nur, wenn  $\int_{-\infty}^{\infty} |f(t)| dt$  existiert



→  $f(t)$  kausal machen:  $f_{\text{kausal}}(t) = s(t)f(t)$

# Laplace-Transformation

Idee: Nutze statt komplexer Frequenz:  $s := \sigma + j\omega$  mit additiver Gewichtskonstante

$$F(\sigma + j\omega) = \int_{0^-}^{\infty} f(t) e^{-(\sigma+j\omega)t} dt$$

$\xleftarrow{\lim_{\varepsilon \rightarrow 0^+} \int_{-\varepsilon}^{\infty}}$

$$= \int_{0^-}^{\infty} f(t) e^{-\sigma t} e^{-j\omega t} dt$$

**Exponentiell abfallendes Gewicht** für  $f(t)$

→ Durch geeignete Wahl von  $\sigma$  wird die Konvergenz sichergestellt

Definition:

$$F(\sigma + j\omega) = \int_{0^-}^{\infty} f(t) e^{-(\sigma+j\omega)t} dt$$

→ ersetze  $\sigma + j\omega$  durch komplexe Zahl  $s$

$$F(s) = \int_{0^-}^{\infty} f(t) e^{-st} dt$$

Rücktransformation:

$$f(t) = \frac{1}{2\pi j} \int_{\sigma-j\infty}^{\sigma+j\infty} F(s) e^{st} ds$$

Korrespondenzen:

Zeitliche Ableitung:

$$\frac{d}{dt} f(t) \quad \circ \bullet \quad s F(s) - f(0^-)$$

Linearität:

$$k_1 f_1(t) + k_2 f_2(t) \quad \circ \bullet \quad k_1 F_1(s) + k_2 F_2(s)$$

Zeitverschiebung:

$$f(t - t_0) \quad \circ \bullet \quad F(s) e^{-st_0}$$

Konvolutionstheorem:

$$f_1(t) * f_2(t) \quad \circ \bullet \quad F_1(s) F_2(s)$$

**Signal in der Zeit**

**Frequenzdarstellung**

$$\delta(t) \quad \circ \bullet \quad 1$$

$$s(t) \quad \circ \bullet \quad \frac{1}{s} \quad Re(s) > 0$$

$$s(t) e^{-at} \quad \circ \bullet \quad \frac{1}{s+a} \quad Re(s+a) > 0$$

$$s(t) t e^{-at} \quad \circ \bullet \quad \frac{1}{(s+a)^2} \quad Re(s+a) > 0$$

$$s(t)(1 - e^{-at}) \quad \circ \bullet \quad \frac{a}{s(s+a)} \quad Re(s+a) > 0$$

$$A \operatorname{rect}(t/T) \quad \circ \bullet \quad A T \frac{\sinh(sT/2)}{sT/2}$$

$$s(t) \cos(\omega t) \quad \circ \bullet \quad \frac{s}{s^2 + \omega^2} \quad Re(s) > 0$$

$$s(t) \sin(\omega t) \quad \circ \bullet \quad \frac{\omega}{s^2 + \omega^2} \quad Re(s) > 0$$

$$\sum_{n=0}^{\infty} \delta(t - nT) \quad \circ \bullet \quad \frac{1}{1 - e^{-sT}}$$

## Rationale Laplacetransformation

Viele Systeme werden durch Differentialgleichungen in folgender Form beschrieben:

$$\begin{aligned} b_0 y(t) + b_1 \frac{d}{dt} y(t) + \dots + b_n \frac{d^n}{dt^n} y(t) & \quad \text{Eingabe} \\ = a_0 u(t) + a_1 \frac{d}{dt} u(t) + \dots + a_m \frac{d^m}{dt^m} u(t) & \quad \text{Ausgabe} \end{aligned}$$

Mit Startwert 0 und Linearität korrespondiert das zur Laplace-Transformation:

$$\begin{aligned} b_0 Y(s) + b_1 s Y(s) + \dots + b_n s^n Y(s) \\ = a_0 U(s) + a_1 s U(s) + \dots + a_m s^m U(s) \end{aligned}$$

## Transferfunktion:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{a_0 + a_1 s + \dots + a_m s^m}{b_0 + b_1 s + \dots + b_n s^n}$$

## Rationale Transferfunktion

$$\begin{aligned} H(s) &= \frac{a_0 + a_1 s + \dots + a_m s^m}{b_0 + b_1 s + \dots + b_n s^n} \\ &= \frac{\sum_{j=0}^m a_j s^j}{\sum_{i=0}^n b_i s^i} \end{aligned}$$

- Zähler und Nenner sind **Polynome von s**
- Praktische Bedeutung
  - Lösen von Differentialgleichungen
  - Einfache Rücktransformation
  - Aussagen zur Stabilität von Systemen

## Produktform:

$$H(s) = \frac{a_m}{b_n} \frac{(s - s_{0,1})(s - s_{0,2}) \dots (s - s_{0,m})}{(s - s_{\infty,1})(s - s_{\infty,2}) \dots (s - s_{\infty,n})}$$

$s_{0,1} \dots s_{0,m}$ : Nullstellen der Transferfunktion  $H(s)$   
=> Signale dieser Frequenz werden ausgelöscht

$s_{\infty,1} \dots s_{\infty,n}$ : Polstellen der Transferfunktion  $H(s)$   
=> Signale dieser Frequenz werden unendlich verstärkt  
=> Instabilität

## Partialbruchzerlegung:

### Voraussetzung:

1. Zählergrad  $m$  kleiner als Nennergrad  $n$
2. Nur einfache Polstellen

### Darstellung:

$$H(s) = \sum_{i=0}^n \frac{A_i}{s - s_{\infty,i}}$$

→ leichtere Rücktransformation in die Zeitdomäne

### Korrespondenz:

$$\sum_{i=1}^n \frac{A_i}{s - s_{\infty,i}} \bullet \circ \sum_{i=1}^n A_i s(t) e^{s_{\infty,i} t}$$

Berechnung:  $H(s) = \frac{s}{s^2 + 3s + 2}$

1. Nullstellen des Nenners berechnen

$$\rightarrow pq\text{-Formel} \Rightarrow (s+1)(s+2)$$

2. Nenner in Produktform schreiben

$$H(s) = \frac{s}{(s+1)(s+2)}$$

3. Jeder Nullstelle Partialbruch zuordnen

$$H(s) = \frac{A_1}{(s+1)} + \frac{A_2}{(s+2)}$$

4. A's bestimmen

1. Methode:

1. Brüche auf gleichen Nenner bringen

$$H(s) = \frac{A_1(s+2)}{(s+1)(s+2)} + \frac{A_2(s+1)}{(s+1)(s+2)}$$

2. Brüche addieren

$$H(s) = \frac{A_1(s+2) + A_2(s+1)}{(s+1)(s+2)}$$

3. Zähler ausmultiplizieren und nach s sortieren

$$H(s) = \frac{s \cdot (A_1 + A_2) + (2A_1 + A_2)}{(s+1)(s+2)}$$

4. Koeffizientenvergleich der Zähler

$$s(A_1 + A_2) + (2A_1 + A_2) = s$$

$$\Rightarrow \begin{cases} A_1 + A_2 = 0 \\ 2A_1 + A_2 = 1 \end{cases} \quad \text{Lösen: } A_1 = -1 \text{ und } A_2 = 2$$

2. Methode

Finde  $A_i$  durch multiplizieren von  $H(s)$  mit zugehörigem Nenner und Einsetzen von zugehöriger Polstelle

$$\begin{aligned} A_1 &= (s+1) H(s) \Big|_{s=-1} = \frac{s}{(s+1)(s+2)} \cdot (s+1) \Big|_{s=-1} \\ &= \frac{s}{s+2} \Big|_{s=-1} \\ &= -1 \end{aligned}$$

5.  $A_i$ 's einsetzen

$$H(s) = \frac{-1}{s+1} + \frac{2}{s+2}$$

## Konvergenz

Sei  $f(t)$  die Zeitfunktion für eine Laplacefunktion in Partialbruchdarstellung:

$$f(t) = \sum_{i=1}^n A_i e^{s_{\infty,i} t} s(t)$$

Dann:

$$\begin{aligned} F(s) &= \int_0^\infty \sum_{i=1}^n A_i e^{s_{\infty,i} t} e^{-st} dt \\ &= \sum_{i=1}^n A_i \int_0^\infty e^{(s_{\infty,i}-s)t} dt \end{aligned}$$

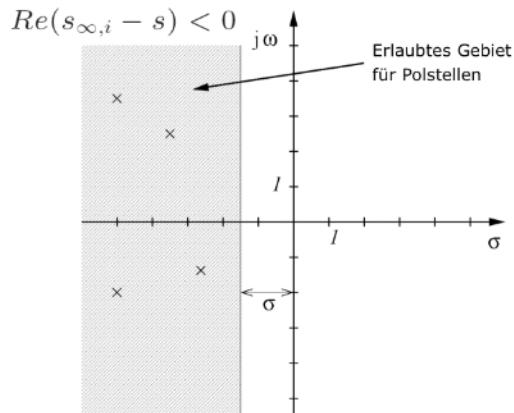
Laplacetransformation konvergiert nur, wenn das Integral für jede Polstelle existiert

$$\int_0^\infty e^{(s_{\infty,i}-s)t} dt = \frac{1}{s_{\infty,i} - s} e^{(s_{\infty,i}-s)t} \Big|_{t=0}^{t=\infty}$$

$$\Rightarrow \lim_{t \rightarrow \infty} \frac{1}{s_{\infty,i} - s} e^{(s_{\infty,i}-s)t} \quad \text{Grenzwert muss existieren}$$

Also: Die rationale Laplacetransformation konvergiert genau dann, wenn:

$$\operatorname{Re}(s_{\infty,i} - s) < 0$$



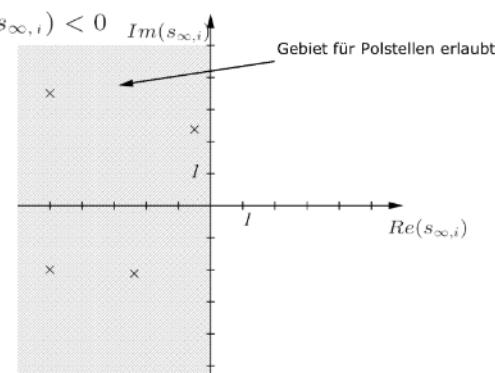
## Stabilität

Sei  $H(s)$  rationale Transferfunktion

Dann ist die Impulsantwort:  $h(t) = \sum_{i=1}^n A_i e^{s_{\infty,i} t} s(t)$

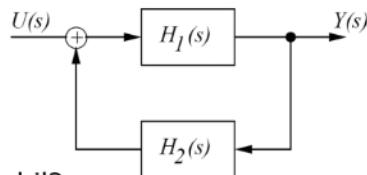
System ist stabil  $\Leftrightarrow$  alle Realteile der Polstellen sind negativ

$$\operatorname{Re}(s_{\infty,i}) < 0 \quad \forall i \in [0 \dots n]$$



## Beispiel

Zwei Systeme  $H_1(s) = \frac{6}{1+s}$  und  $H_2(s) = \frac{1}{2+s}$   
sind wie folgt verbunden:



Ist das System stabil?

Mit Superpositionsprinzip:

$$Y(s) = H_1(s)U(s) + H_1(s)H_2(s)Y(s)$$

$$\Rightarrow H(s) = \frac{Y(s)}{U(s)} = \frac{H_1(s)}{1 - H_1(s)H_2(s)}$$

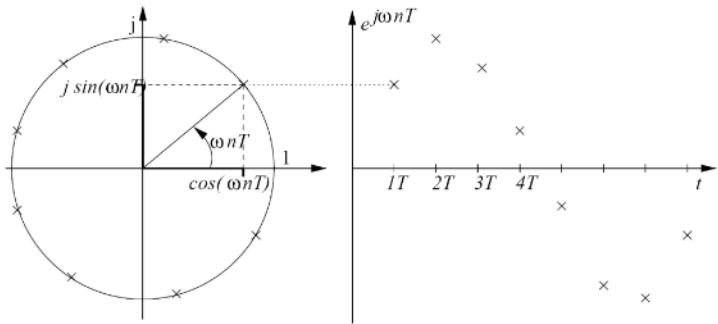
$$\begin{aligned}
 H(s) &= \frac{H_1(s)}{1 - H_1(s)H_2(s)} & H_1(s) &= \frac{6}{1+s} \\
 &= \frac{6/(1+s)}{1 - \frac{6}{(1+s)(2+s)}} & H_2(s) &= \frac{1}{2+s} \\
 &= \frac{6(2+s)}{(1+s)(2+s) - 6} \\
 &= \frac{12+6s}{s^2+3s-4} = \frac{12+6s}{(s-1)(s+4)} & \text{Stabil?}
 \end{aligned}$$

- System hat Polstelle bei  $s=1$
- => **System ist nicht stabil!**

## 14. Frequenzraum (zeitdiskret)

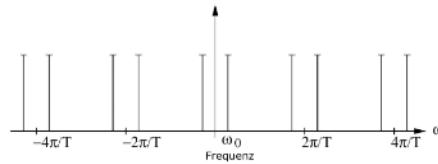
### zeitdiskrete komplexe Exponentialfunktion

**Definition:**  $f[n] = f[nT] = e^{j\omega nT}$



### Spektrum der Sinus-Folge

Periodisch mit Periode  $2\pi/T$



### Transferfunktion

**Gegeben:** Impulsantwort  $h[n]$  und Eingabe  $u[n] = c \cdot e^{j\omega nT}$

**Ausgabe:** Berechnung durch Konvolution

$$y[n] = \sum_{\nu=-\infty}^{\infty} u[n-\nu]h[\nu] = c \sum_{\nu=-\infty}^{\infty} e^{j\omega(n-\nu)T} h[\nu]$$

$$y[n] = \underbrace{ce^{j\omega nT}}_{u[n]} \underbrace{\sum_{\nu=-\infty}^{\infty} e^{-j\nu\omega T} h[\nu]}_{H(e^{j\omega T})}$$

**Definition:**  $H(e^{j\omega T}) = \frac{y[n]}{u[n]} \Big|_{u[n]=e^{j\omega nT}} \quad |\omega| \leq \omega_0/2 \quad \omega_0 = 2\pi/T$

$$H(j\omega) = \frac{y[n]}{u[n]} \Big|_{u[n]=c e^{j\omega nT}} = \frac{Y(j\omega)}{U(j\omega)}$$

21

### Zeitdiskrete Fourier-Transformation

→ **Ersatz:**  $\Omega = \omega T$  und  $d\omega = d\Omega/T$

**Vorwärts-Transformation:**  $F(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} f[n]e^{-jn\Omega}$

**Rückwärts-Transformation:**  $f[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} F(e^{j\Omega n})e^{j\Omega n} d\Omega$

**Konvergenz**  $\Leftrightarrow \sum_{-\infty}^{\infty} |f[n]| < \infty$

## Z-Transformation

**Problem:** Konvergenz der Fourier-Transformation  $\Leftrightarrow \sum_{-\infty}^{\infty} |f[n]| < \infty$

**Idee:** Multipliziere  $f[n]$  mit geometrischer Reihe  $r^{-n}$

$\Rightarrow$  Konvergenz gilt dann:  $\sum_{n=-\infty}^{\infty} |f[n] r^{-n}| < M < \infty$

**Definition:** Schreibe  $z \in \mathbb{C}$  anstatt  $r \cdot e^{j\Omega}$

$$F(z) = \sum_{n=-\infty}^{\infty} f[n] z^{-n}$$

**Korrespondenz:**

$$\begin{aligned} f[n] r^{-n} &\rightsquigarrow \sum_{n=-\infty}^{\infty} f[n] r^{-n} e^{-jn\Omega} \\ &= \sum_{n=-\infty}^{\infty} f[n] (r e^{j\Omega})^{-n} \end{aligned}$$

## Inverse Z-Transformation:

$$f[n] = \frac{1}{2\pi j} \oint_C F(z) z^{n-1} dz$$

## Transferfunktion:

$\rightarrow$  Z-Transformation der Impulsantwort charakterisiert zeitdiskrete Systeme

$$h[n] \rightsquigarrow H(z) = \frac{Y(z)}{X(z)}$$

## Behaupfte Korrespondenzen

Zeitdiskretes Signal	Frequenzdarstellung
-------------------------	---------------------

$f[n]$	$F(z)$
$\delta[n]$	$1 \quad z \in \mathbb{R}$
$\delta[n-m]$	$z^{-m} \quad z \neq 0 \quad \forall m > 0 \text{ or } z \neq \infty \quad \forall m < 0$
$s[n]$	$\frac{1}{1-z^{-1}} \quad  z  > 1$
$s[n] a^n$	$\frac{1}{1-a z^{-1}} \quad  z  >  a $
$s[n] n a^n$	$\frac{a z^{-1}}{(1-a z^{-1})^2} \quad  z  >  a $
$s[n] n^2 a^n$	$\frac{a z^{-1} + a^2 z^{-2}}{(1-a z^{-1})^3}$

### Linearität:

$$k_1 f_1[n] + k_2 f_2[n] \rightsquigarrow k_1 F_1(z) + k_2 F_2(z)$$

### Zeitverschiebung:

$$f[n-k] \rightsquigarrow F(z) z^{-k}$$

### Konvolution:

$$f_1[n] * f_2[n] \rightsquigarrow F_1(z) F_2(z)$$

## Zeitverschiebung

$$\begin{aligned}
 f[n-k] &\circlearrowleft \bullet \sum_{n=-\infty}^{\infty} f[n-k] z^{-n} \\
 &= \sum_{m=-\infty}^{\infty} f[m] z^{-m-k} = z^{-k} \sum_{m=-\infty}^{\infty} f[m] z^{-m} \\
 &= z^{-k} F(z)
 \end{aligned}$$

Jeder Zeitschritt Verschiebung  
entspricht Vorfaktor  $z^{-1}$

$\hat{z}$

## Relationale Z-Transformation

→ Lineare Differentialgleichungen entsprechen rationalen Z-Transformationen

Basiert auf:  $f[n-k] \circlearrowleft \bullet z^{-k} F(z)$

Gegeben: Lineare Gleichungssysteme der Form:

$$y[n] = \sum_{k=1}^N a_k y[n-k] + \sum_{k=0}^M b_k u[n-k]$$

→ Z-Transformation:

$$Y(z) = \sum_{k=1}^N a_k z^{-k} Y(z) + \sum_{k=0}^M b_k z^{-k} U(z)$$

$$Y(z) = \sum_{k=1}^N a_k z^{-k} Y(z) + \sum_{k=0}^M b_k z^{-k} U(z)$$

$$Y(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}} U(z)$$

## Transferfunktion:

$$H(z) = \frac{Y(z)}{U(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}} = \frac{N(z^{-1})}{D(z^{-1})}$$

## Alternative Darstellung

$$H(z) = \frac{z^N}{z^M} \frac{N'(z)}{D'(z)} = z^{N-M} \frac{N'(z)}{D'(z)}$$

$$\begin{aligned}
 N(z^{-1}) &= \sum_{k=0}^M b_k z^{-k} = \frac{1}{z^M} \sum_{k=0}^M b_{M-k} z^k = \frac{1}{z^M} N'(z) \\
 D(z^{-1}) &= 1 - \sum_{k=1}^N a_k z^{-k} = 1 - \frac{1}{z^N} \sum_{k=0}^{N-1} a_{N-k} z^k = \frac{1}{z^N} D'(z)
 \end{aligned}$$

## Produktdarstellung

$$H(z) = A z^{N-M} \frac{\prod_{k=1}^M (z - z_{0,k})}{\prod_{k=1}^N (z - z_{\infty,k})} = A \frac{\prod_{k=1}^M (1 - z_{0,k} z^{-1})}{\prod_{k=1}^N (1 - z_{\infty,k} z^{-1})}$$

$z_0$ : Nullstellen

$z_\infty$ : Polstellen

## Darstellung als Partialbruch

**Bedingung:**  $N >= M$

$$H(z) = K_0 + \sum_{i=1}^N \frac{K_i}{1 - z_{\infty,i} z^{-1}}$$

**Berechnung:**  $K_0 = \lim_{z \rightarrow \infty} H(z)$

$$K_i = \lim_{z \rightarrow z_{\infty,i}} H(z)(1 - z_{\infty,i} z^{-1})$$

**Rücktransformation:**

→ Terme einzeln rücktransformieren

$$\frac{K_i}{1 - z_{\infty,i} z^{-1}} \rightsquigarrow s[n] K_i z_{\infty,i}^n$$

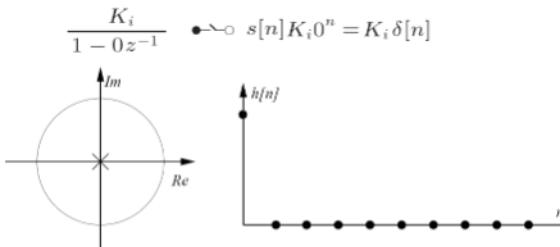
Rücktransformation rationaler Transferfunktionen

$H(z)$  ist Linearkombination komplexer geometrischer Reihen

$$c z_{\infty,i}^n$$

**Spezialfälle:**

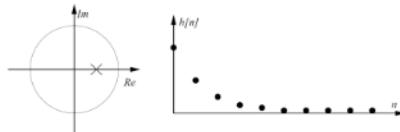
1.  $z_{\infty,i} = 0$  :



- Polstelle bei Null korrespondiert zu Delta-Impuls

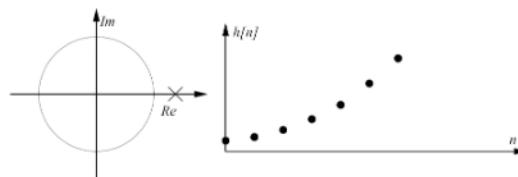
2.  $0 < z_{\infty,i} < 1$  :

- $0 < z_{\infty,i} < 1$  hat monoton fallende geometrische Reihe als Impulsantwort



3.  $z_{\infty,i} \in (1, \infty)$  :

- $1 < z_{\infty,i}$  hat steigende geometrische Reihe als Impulsantwort

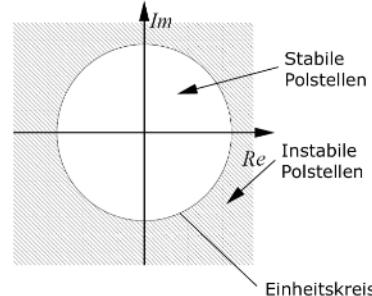


⇒ instabil

## Stabilität zeitdiskreter Systeme

$H(z)$  ist stabil, wenn alle Partialbrüche stabil

Es muss gelten:  $|z_{\infty,i}| < 1$



## Lineare Digitale Filter

→ Realisierung zeitdiskreter linearer Systeme mit digitalen Schaltkreisen

Anwendung:

- Signalverarbeitung
- Regelungstechnik

Eingabe: Digitales Signal

$$u[n] \circlearrowright U(z)$$

Ausgabe: Digitales Signal

$$y[n] \circlearrowright Y(z) = U(z)H(z)$$

## Klassen Linearer Filter

**FIR-Filter:** (finite impulse response)

- Immer stabil
- Höherer Implementierungsaufwand

**IIR-Filter:** (infinite impulse response)

- Möglicherweise instabil
- Geringer Implementierungsaufwand

## FIR-Filter

→ auch: Moving-Average-Filter

### Definition

Lineare Differenzengleichungen der Form:

$$y[n] = \sum_{k=1}^N a_k y[n-k] + \sum_{k=0}^M b_k u[n-k]$$

mit  $a_i = 0$ :  $y[n] = b_0 u[n] + b_1 u[n-1] + \dots + b_M u[n-M]$

→ unabhängig von früheren Ausgaben

**Transferfunktion:**  $H(z) = \sum_{k=0}^M b_k z^{-k}$

→ keine Polstellen  $\Rightarrow$  immer stabil

## IIR-Filter

→ auch: Auto-Regressive Moving Average-Filter

Lineare Differenzengleichungen mit  $a_i \neq 0$

$$y[n] = \sum_{k=1}^N a_k y[n-k] + \sum_{k=0}^M b_k u[n-k]$$

Stabilität nur, wenn der Betrag der Polstellen < 1

Stabiler Tiefpass:  $y[n] = 0.01 u[n] + 0.99 y[n-1]$

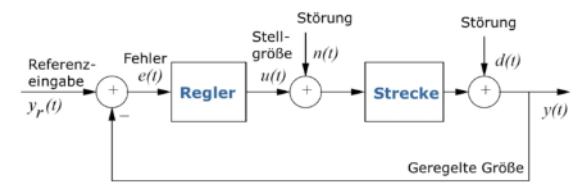
**Instabil:**  $y[n] = 0.1 u[n] + 1.01 y[n-1]$

↑  
Verstärkungsfaktor muss < 1

## 15. Performance von Reglern

### Ziele von Reglern im Frequenzraum

Folgeverhalten:  $H_{CL}(s) = \frac{Y(s)}{Y_r(s)} \rightarrow 1$



### Rückweisung von Störung:

Störung nach der Strecke:  $H_D(s) = \frac{Y(s)}{D(s)} \rightarrow 0$

Störung vor der Strecke:  $H_N(s) = \frac{Y(s)}{N(s)} \rightarrow 0$

### PT<sub>1</sub>-Strecke

→ Strecke im Frequenzraum

Häufige Transferfunktion der geregelten Strecke

$$H_{PT_1}(s) = \frac{K_S}{1 + sT_1}$$

Verstärkungsfaktor  
Zeitkonstante

→ nur eine Polstelle mit negativem  $s$  ( $s_P = -1/T_1$ )  
⇒ stabil

Korrespondenz:  $s(t) e^{-at} \circlearrowleft \frac{1}{s+a}$

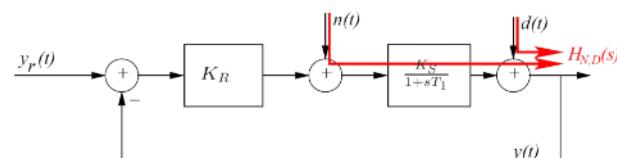
Stationäres Verhalten für konstante Stellgröße  $u(t)$ :

Integral der Impulsantwort =  $K_S$

=> Endwert =  $K_S u(t)$

### Proportional-Regler für PT<sub>1</sub>-Strecke

Einfachster Regler:  $H_R = K_R$



- Gesucht:
- Folgeverhalten:  $H_{CL}(s)$  soll = 1
  - Rückweisung von Störungen:  $H_N(s), H_D(s)$

### Folgeverhalten:

Open-Loop Transferfunktion:  $H_{OL} = H_R H_{PT_1} = \frac{K_S K_R}{1 + sT_1}$

Closed-Loop Transferfunktion durch Superposition:

$$H_{CL} = \frac{H_{OL}}{1 + H_{OL}} = \frac{K_S K_R}{(1 + K_S K_R) + sT_1} = y_r(t) \frac{K_S K_R}{1 + K_S K_R}$$

mit  $s=0$  da konstante Stellgröße  $u(t)$

$H_{CL} \neq 1 \Rightarrow$  kein ideales Folgeverhalten

## Rückweisung von Störungen der Stellgröße

Transferfunktion für Störung  $n(t)$  der Stellgröße  $u(t)$ :

$$H_N = \frac{H_S}{1 + H_R H_S} = \frac{K_S}{1 + K_R K_S + sT_1}$$

Für  $s=0$ :  $y(t) = n(t) \frac{K_S}{1 + K_R K_S}$

$H_N \neq 0 \Rightarrow$  dauerhafter Fehler

→ wird beliebig klein, wenn  $K_R \rightarrow \infty$

## Rückweisung von Störungen der Regelgröße

Transferfunktion für Störung  $d(t)$  der Ausgabe  $y(t)$

$$H_N = \frac{H_S}{1 + H_R H_S} = \frac{K_S}{1 + K_R K_S + sT_1}$$

Für  $s=0$ :  $y(t) = n(t) \frac{K_S}{1 + K_R K_S}$

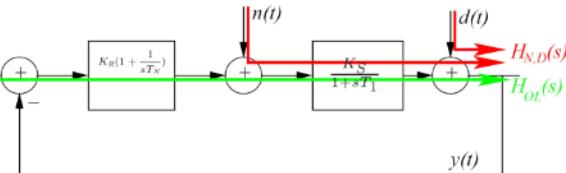
$H_D \neq 0 \Rightarrow$  dauerhafter Fehler

→ wird beliebig klein, wenn  $K_R \rightarrow \infty$

## Proportional-Integral-Regler für PT<sub>1</sub>-Strecke

Transferfunktion:  $H_R(s) = K_R \left(1 + \frac{1}{sT_N}\right)$

↑ Integrationszeit



Impulsantwort:  $s(t) \circ - \bullet \frac{1}{s}$

Folgeverhalten:

$$\begin{aligned} H_{CL}(s) &= \frac{K_R K_S (1 + sT_N)}{sT_N (1 + sT_1) + K_R K_S (1 + sT_N)} \\ &= \frac{K_R K_S + s K_R K_S T_N}{K_R K_S + sT_N (K_R K_S + 1) + s^2 T_N T_1} \end{aligned}$$

Für  $s=0$ :  $H_{CL}(0) = \frac{K_R K_S}{K_R K_S} = 1$

→ kein Fehler

## Rückweisung von Störungen der Stellgröße

$$\begin{aligned} H_{CL}(s) &= \frac{K_R K_S (1 + sT_N)}{sT_N (1 + sT_1) + K_R K_S (1 + sT_N)} \\ &= \frac{K_R K_S + s K_R K_S T_N}{K_R K_S + sT_N (K_R K_S + 1) + s^2 T_N T_1} \end{aligned}$$

Für  $s=0$ :  $H_N(0) = \frac{0}{K_R K_S} = 0$

## Rückweisung von Störungen der Regelgröße

$$\begin{aligned} H_D(s) &= \frac{sT_N K_S (1 + sT_1)}{sT_N (1 + sT_1) + K_R K_S (1 + sT_N)} \\ &= \frac{s^2 K_S T_N T_1 + sT_N K_S}{K_R K_S + sT_N (K_R K_S + 1) + s^2 T_N T_1} \end{aligned}$$

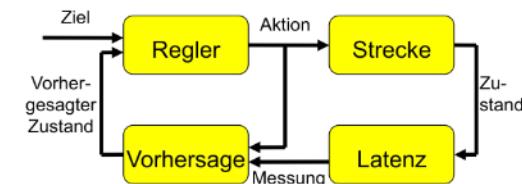
Für  $s=0$ :  $H_D(0) = \frac{0}{K_R K_S} = 0$

## 16. Verhaltenssteuerung

### Problem: Totzeit

→ Regelkreis besitzt Latenz 100...150 ms

→ Exakt und langsam oder ungenau und schnell



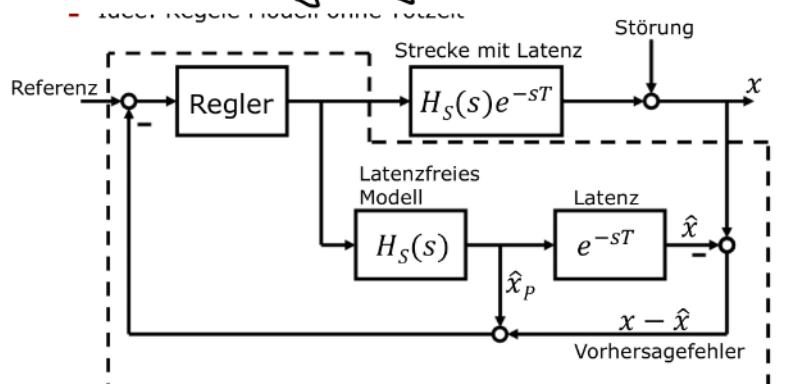
Lösung: Vorhersage des Bewegungszustands für die Latenzzeit

→ **Neuronales Netz** trainiert mit aufgenommenen Daten.  
→ wird gespeist mit Bewegungsmessungen und letzten Aktionen

Ergebnis: Verhaltenssteuerung ohne Verzögerung  
→ schnell und genau

### Smith-Prediktor:

Idee: Modellregelung ohne Totzeit



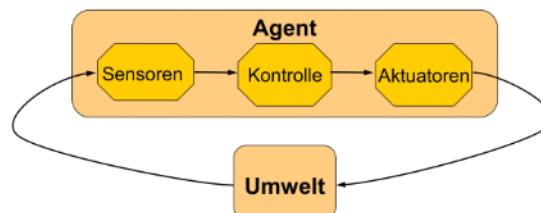
## Autonome Mobile Roboter

Eigenschaften:

- vollständiger Agent
- Autonomie
- selbsterhaltend
- Körperllichkeit
- situiert

### Teilprobleme:

- Wahrnehmung der Umwelt
- Verhaltenssteuerung
- Beeinflussung der Umwelt



# Architekturen zur Verhaltenssteuerung

## Deliberativ

- sense-plan-act
- langsam, komplex

## Reaktiv

- sense-act
- schnell, primitiv

## Hybrid

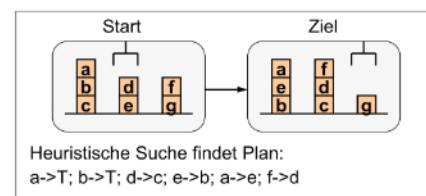
- deliberative und reaktive Schicht
- eventuell Zwischenschicht

## Verhaltensbasiert

- mehrere einfache Verhalten
- Aktivierung situationsabhängig

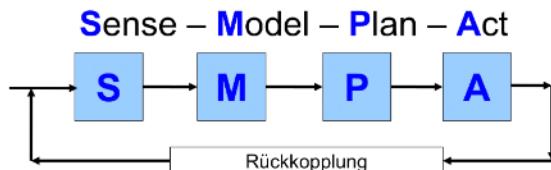
## Deliberative Verhaltenssteuerung

- Erstellung eines kompletten Weltmodells
- Planen von Start zum Ziel
- Ausführung des Plans
- Beispiele: Blockswelt, Shakey



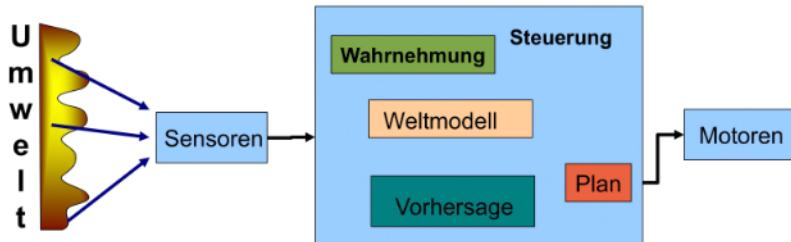
## SMPA-Architektur

→ Steuerung in vier funktionale Blöcke unterteilt



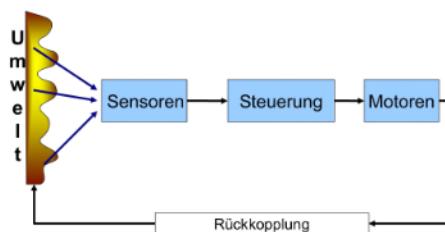
## Verhaltensplanung

- Nutzt internes Modell von Umwelt um weitere Aktionen zu bestimmen
- Plan = Aktionsfolge zur Erreichung des Ziels
- Vorhersage zukünftiger Zustände nötig



## Reaktive Verhaltenssteuerung

- Kein Weltmodell
- Kein Plan
- Direkte Kopplung von Sensoren und Aktuatoren

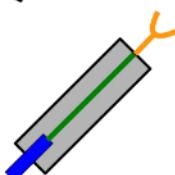


→ Direkte Kopplung

## Braitenberg-Vehicle

### Typ 1:

Eigenschaften: • 1 Sensor, 1 Motor

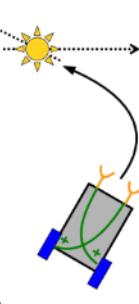
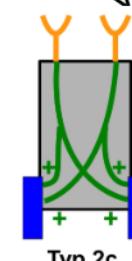
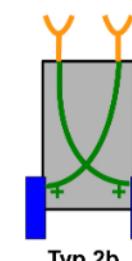
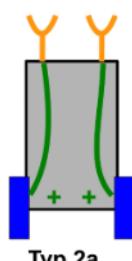


- gemessene Daten des Sensors verstärken den Motor
- in realen Systemen: links-, rechts-Abweichungen durch Störungen

### Typ 2:

Eigenschaften: • 2 Sensoren, 2 Motoren

- Sensoren verstärken Motoren
- unterschiedliche Verbindungen



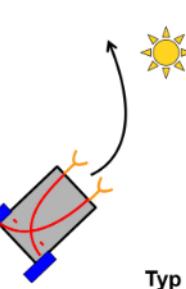
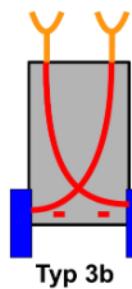
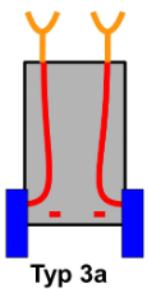
Typ 2a

Typ 2b

### Typ 3:

Eigenschaften: • 2 Sensoren, 2 Motoren

- Sensoren dämpfen Motoren



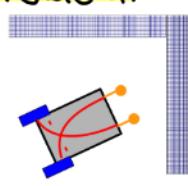
Typ 3a

Typ 3b

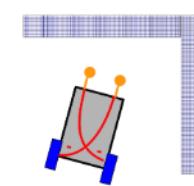
### Typ 3b:

→ kann mit einem Proximity-Sensor zur Hindernisvermeidung genutzt werden

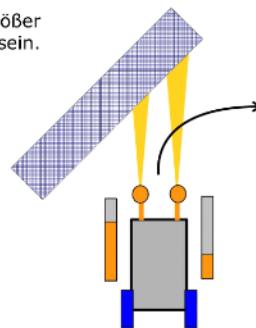
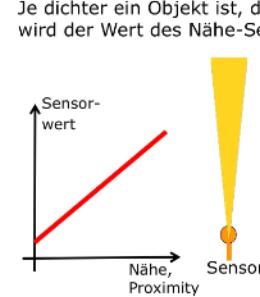
Problem:



nach links drehen



nach rechts drehen



## Weitere Braatenberg Vehikel

Typen 1, 2, 3, 4:  
Reaktive Kontrolle

Typ 5:  
Interne Zustände, Ketten logischer Elemente.

Typ 6:  
Entwurf von Typ 5 Strukturen durch Evolution.

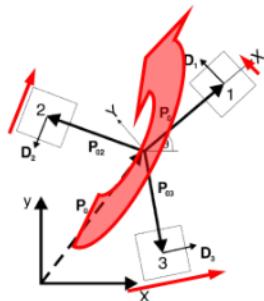
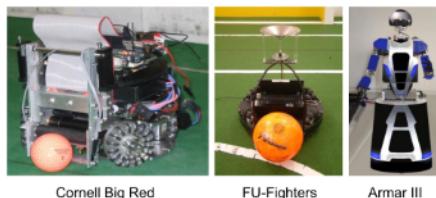
Typ 7:  
Lernen der internen Struktur (Mnemotrix).

Typ 8:  
Visuelles System und Computer Sehen

Typen 9, 10, 11, 12, 13, 14:  
Höhere kognitive Funktionen (Shapes, Getting Ideas,  
Rules and Regularities, Trains of Thought, Foresight,  
Egotism and Optimism) 48

## Omnidirektionale Fahrwerke

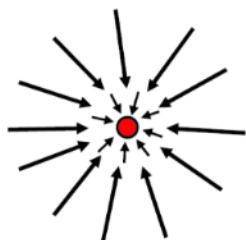
- Geschwindigkeit ( $x, y, \theta$ ) verteilt auf 3 oder mehr Räder
- gute Manövriertbarkeit
- Vorteile in begrenzten Situationen und dynamischen Situationen



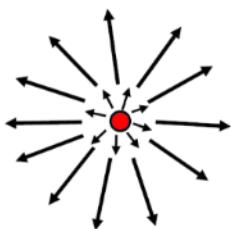
## Potentialfelder

### Potentialfeld-Typen:

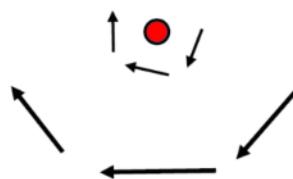
Anziehend:



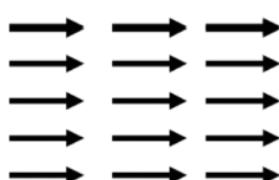
Abstoßend



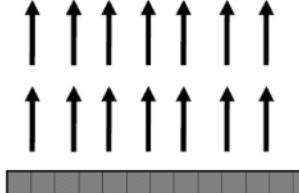
tangential:



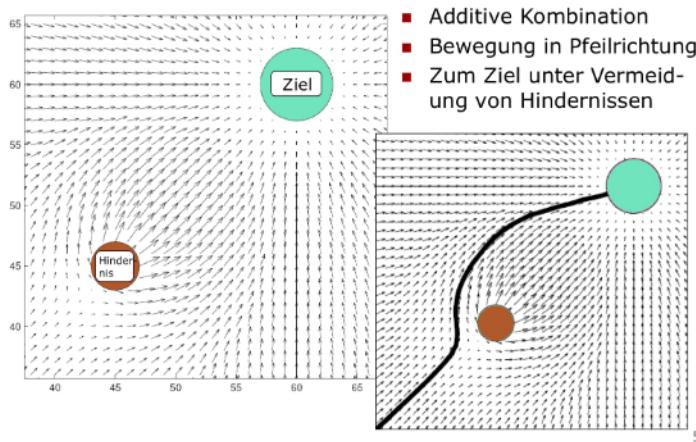
Uniform:



Normal zur Wand:

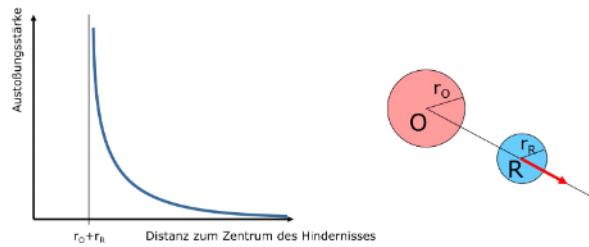


## Kombinierte Potentialfelder:



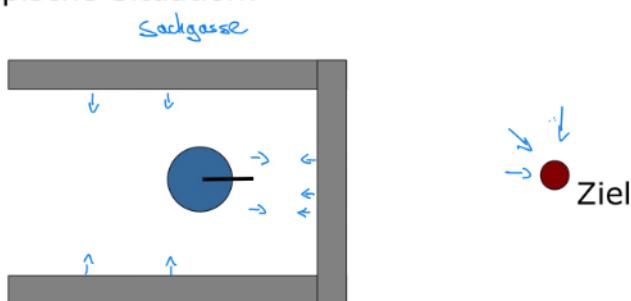
## Beispiel: Abstoßendes Potential

- Richtung: Ausgehend vom Zentrum des Hindernisses
- Stärke: Zunehmend mit Nähe, z.B.  $1/(dist(O, R) - (r_O + r_R))$



## Problem: Lokale Minima

- Anziehende und abstoßende Potentiale löschen sich aus
- Roboter findet nicht heraus
- Typische Situation:

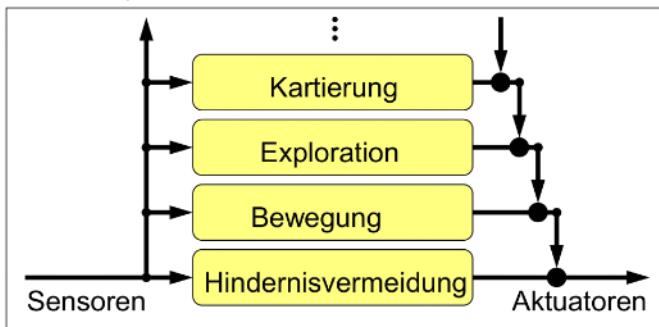


## Subsumptionsarchitektur

→ Verhaltensbasiert, endliche Automaten, Nachrichten

→ Kompetenzschichten mit zunehmender Komplexität

**Subsumption:** überstimmen tieferer Schichten



## Dual Dynamics

→ verhaltensbasiert

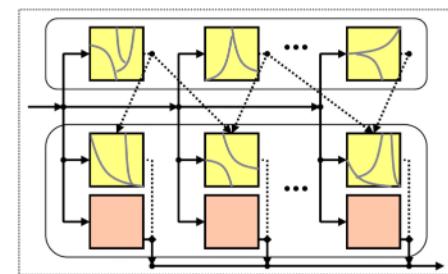
→ dynamische Systeme

→ besteht aus 2 Verhaltensarten

- elementar
- komplex

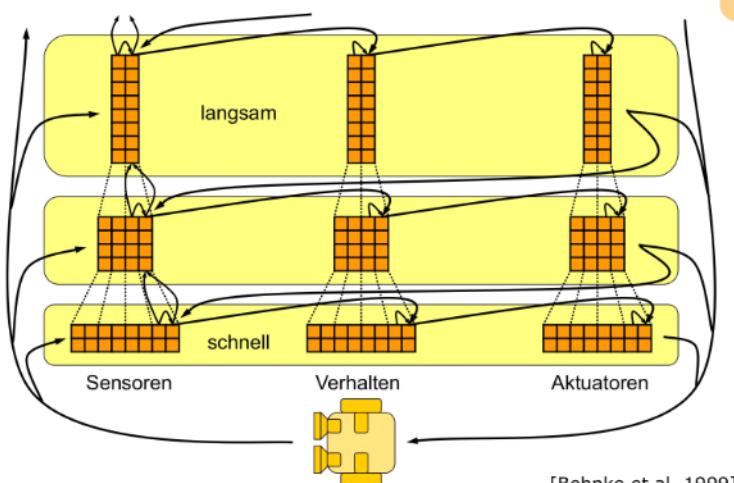
Elementare Verhalten haben zwei Module:

- Aktivierungsdynamik: Soll ich jetzt etwas tun?
- Zieldynamik: Was soll ich tun?



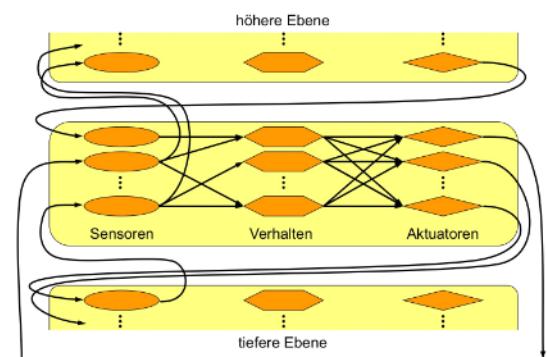
Herbert Jaeger 1995

## Hierarchie reaktiver Verhalten



Behnke et al. 1999

## Aufbau einer Ebene:



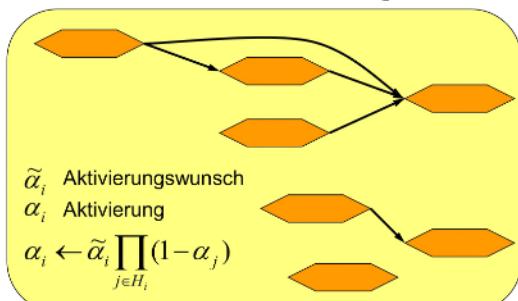
## Hemmung von Verhalten

→ mehrere Verhalten können gleichzeitig aktiv sein.

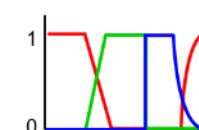
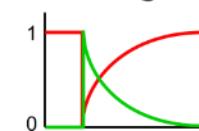
Diese können sich gegenseitig hemmen

→ multiplikative Hemmung

Aktuator-dynamik: gewichtetes Mittel der Änderungswünsche

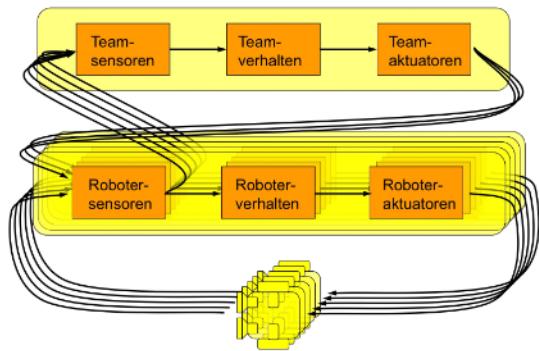


Hemmung graph:

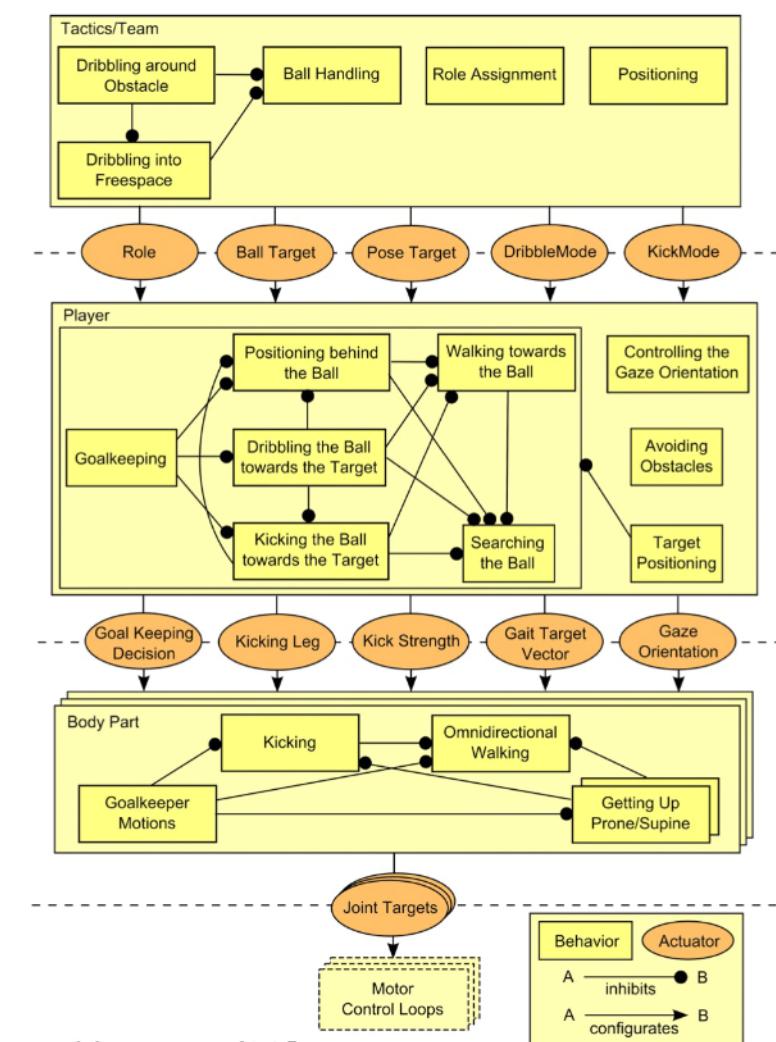


## Teamebene

→ übergeordnete Schicht, die über gemeinsames Verhalten von Robotern entscheidet



## Verhaltenshierarchie



→ Taktik / Teamverhalten

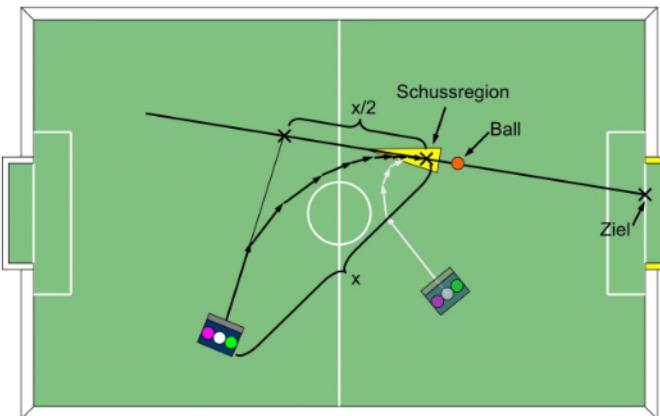
→ Fußballverhalten

→ Grundlegende Fähigkeiten

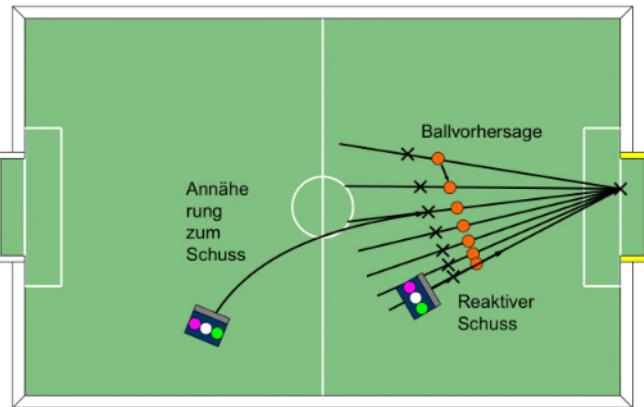
# Beispiele

## Fussball-Roboter:

### Annäherung zum Schuss



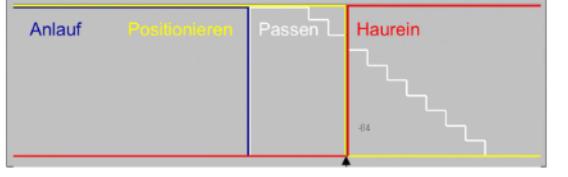
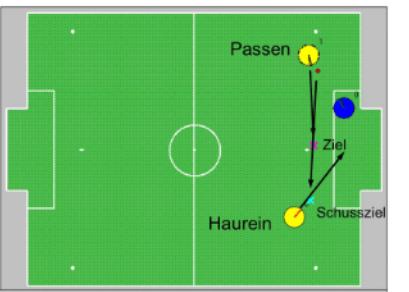
### Schuss eines rollenden Balls



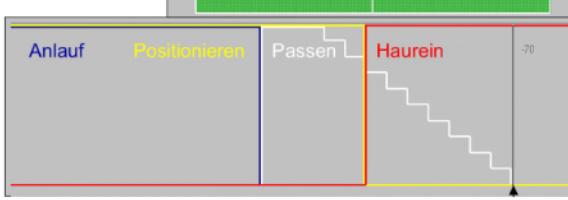
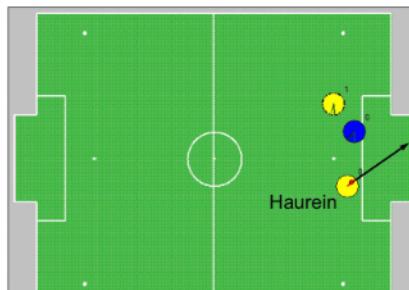
### Passen



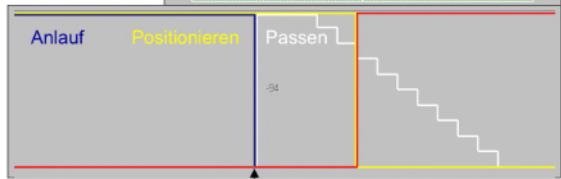
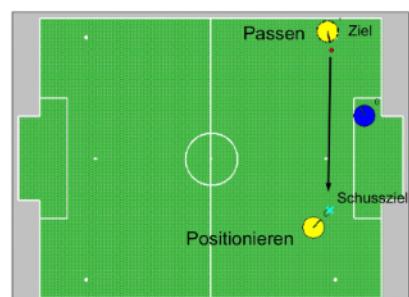
### Passen



### Passen



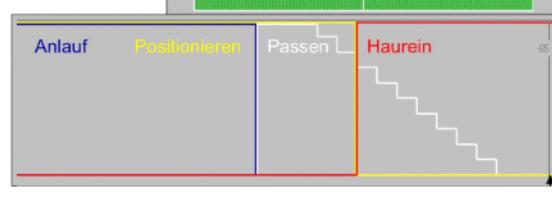
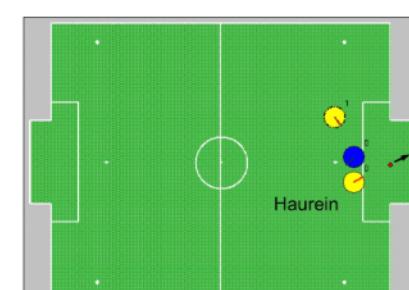
### Passen



### Passen

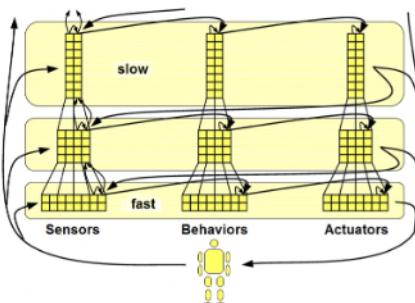


### Passen



# Hierarchisch-Reaktive Verhaltenssteuerung für humanoide Roboter

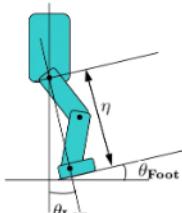
- Hierarchisch, reaktiv
  - Zeithierarchie (kHz, 83Hz, 41.5Hz, 20.75Hz)
  - Agentenhierarchie (Einzelgelenk, Körperteil, Roboter, Team)
  - Komplexitätsreduktion durch Interaktionsbeschrankungen



- Kinematische Steuerung von Körperteilen:

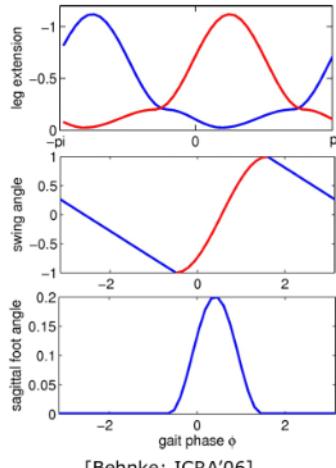
- Beinwinkel
- Fußwinkel
- Beinlänge

[Behnke, Stückler: IJHR'08]



## Omnidirektionales Gehen

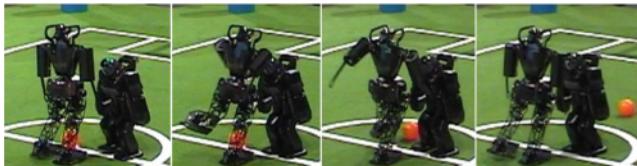
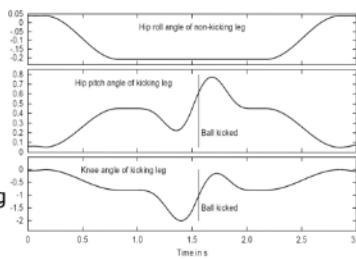
- Kombination von:
  - Vorwärts/rückwärts Gehen
  - Seitliches Gehen
  - Drehen auf der Stelle
- Hauptbestandteile:
  - Gewichtsverlagerung
  - Beinverkürzung
  - Bewegung in Laufrichtung



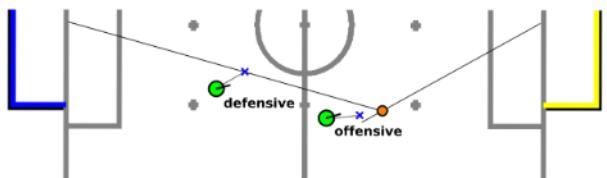
[Behnke: ICRA'06]

## Parametrisierbarer Schuss

- Gewichtsverlagerung
- Ausholen
- Schneller Schwung nach vorn
- Abbremsen
- Zurück in Stand
- Schussstärke, -richtung und Ballposition parametrisierbar



## Positionierung der Spieler



## Hybride Verhaltenssteuerung

→ das Beste aus reaktivem und deliberativen Verhalten soll vereinigt werden

### Drei-Schichten-Architektur:

- **Deliberative Schicht (Plänen)**

- Repräsentation der Zukunft
- ziel-getrieben, langsam

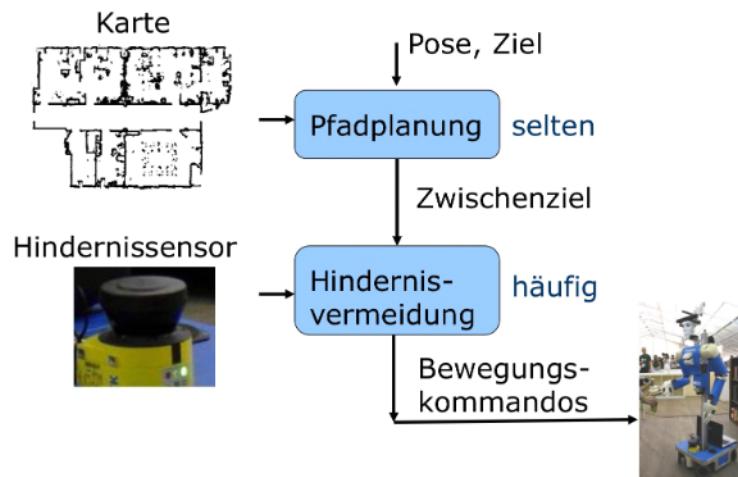
- **Zwischenschicht (Sequenzer)**

- Repräsentation der Vergangenheit
- Abstraktion
- Fehlererkennung, Alternativverhalten

- **Reaktive Schicht (Controller)**

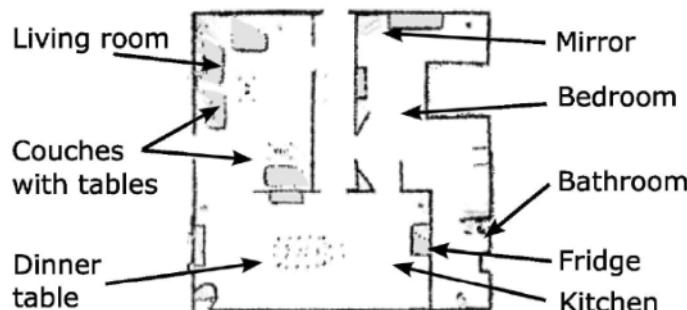
- kein Zustand, schnell, Sensor-getrieben

### Zweischicht-Architektur zur Navigation

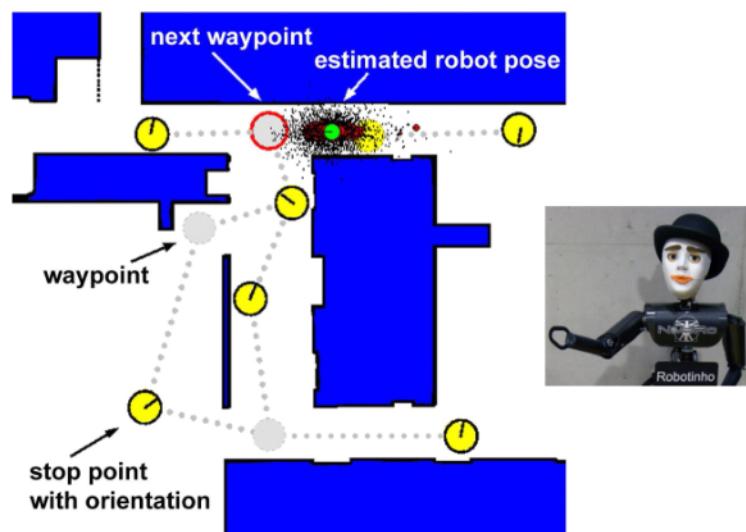


# 17. Pfadplanung

## Kartierung der Umgebung



## Navigation

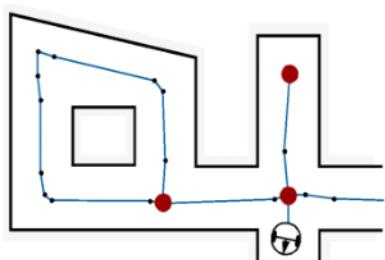


## Pfadplanungsarten

### Routengraphen

Besteht aus Knoten und Kanten.

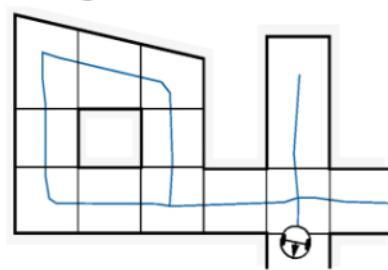
→ Platzierung der Knoten an ausgezeichneten Stellen



### Zelldekomposition

Unterscheidung von freien und belegten Zellen

→ Zellgrenzen willkürlich oder von Karte abgeleitet

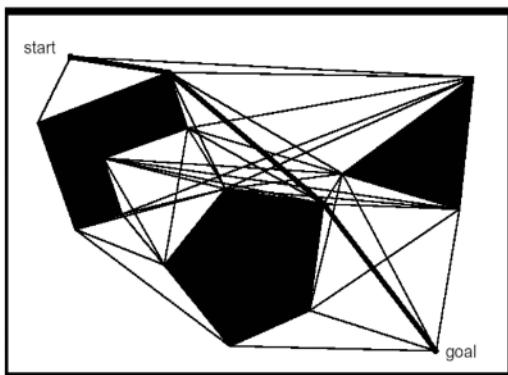


## Sichtbarkeitsgraphen

- Knoten:**
- Ecken von Hindernis-Polygonen
  - Start/Ziel
- Hindernisse müssen um den Radius des Roboters erweitert werden

**Kanten:** gerade Verbindungen zwischen Knoten

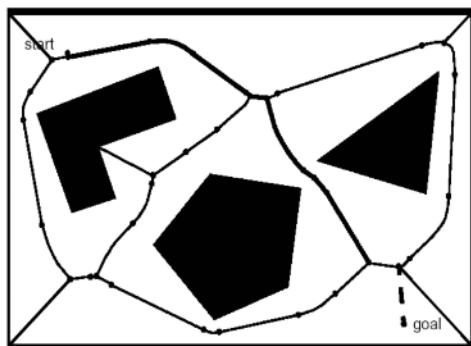
Finden des kürzesten Pfades: **Graphensuche**



## Voronoi-Routengraph

Knoten und Kanten sind zu den nächsten Hindernissen gleich weit entfernt

→ Pfad maximiert Abstand zu Hindernissen

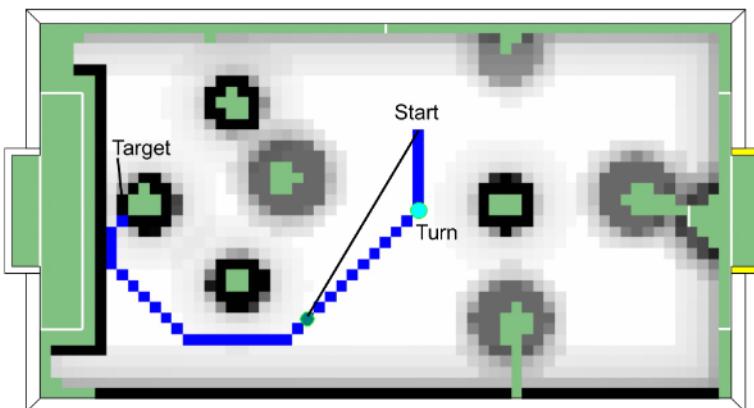


## Zelldekomposition mit Kosten

Hindernisse: Roboter, Feldgrenzen, Torraum

→ Pfad durch **heuristische Suche A\*** bestimmt

8-Nachbarschafts-Graph



## Heuristische Suche:

```
PlanPath(target, obstacles, N){ // Gegeben Positionen von Ziel und
    grid = eval = ClearGrid(); // Hindernissen, Nachbarschaftsgraph
    q = InitPriorityQueue(); // Startknoten einfügen
    while (!q.empty()){
        p = q.pop(); // nächster zu expandierender Knoten
        if (target == p) // Sind wir fertig?
            return previous; // Ausgabe des Pfades
        for(n ∈ N(p)){ // für alle Nachbarn
            if(eval(n) == 0) // Kosten bisher
                best = ∞;
            else
                best = eval(n); // neue Kosten
            new = p.cost + grid(p)·n.l0 + getGrid(n, obstacles, grid)·n.l1;
            if(new < best){ // Vergleich mit alten Kosten
                eval(n) = new; // Hinzufügen zur Prioritätsschlange
                q.push([n, new, getHeuristic(n, target)]);
                previous(n) = p; // Vorgänger merken
            }
        }
    }
}
```

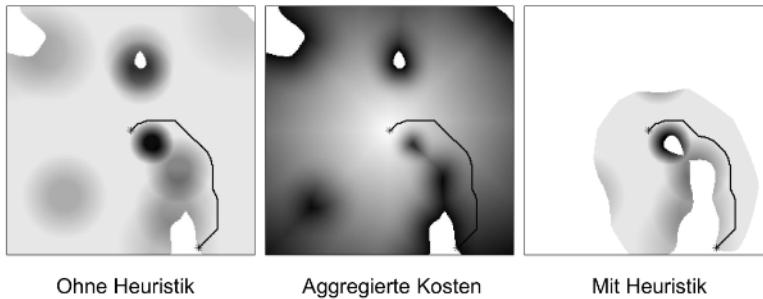
10

## Effekt der Heuristischen Suche

Heuristik schätzt Kosten, die mindestens noch bis zum Ziel anfallen

→ Euklidischer Abstand zum Ziel multipliziert mit Hindernisfreien Kosten

⇒ Reduziert Expansion des Suchraums



Ohne Heuristik

Aggregierte Kosten

Mit Heuristik

## Beispiele:

### Pfadplanung in Büroumgebung Lokale Navigationsplanung

- Omnidirektionales Fahrwerk mit vier lenkbaren Radpaaren
- Zellkosten steigen in der Nähe der Wände
- A\* findet kostengünstigsten Pfad
- Ausrichtung in Engstellen
- Neuplanung wenn Pfad blockiert



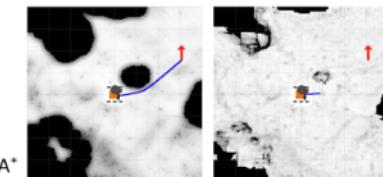
- Lokales Geländemodell aus Rundum-RGB-D-Kamera



- Zellkosten und geplanter Pfad



RoboCup 2013

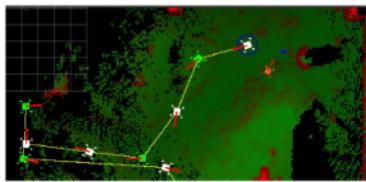
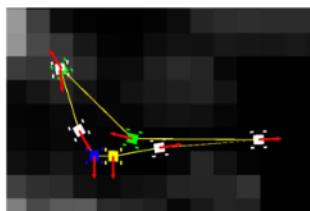


A\*

Rollout

## Missionsplanung

- Spezifikation von Wegpunkten relativ zu initialer Karte
- Scan- und Objektsuchposen
- Interaktive Verfeinerung bei Checkpunkten

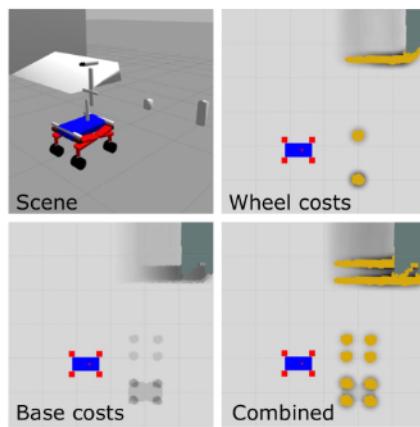


- Posen sind Ziele für allozentrische Pfadplanung
- Abarbeitung braucht Lokalisierung!

## Berücksichtigung der Roboterform

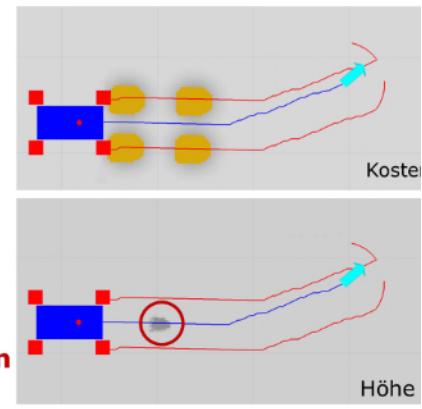
- Kosten für individuelle Radpaare abgeleitet aus lokalen Höhenunterschieden
- Kosten für die Basis
- Nicht-lineare Kombination ergibt 3D ( $x, y, \theta$ ) Kosten

[Klamt and Behnke, 2017]



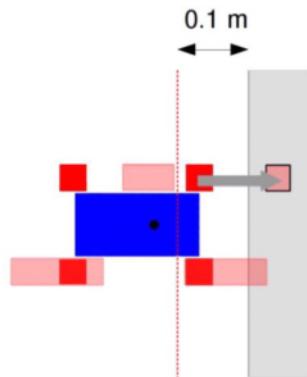
- 16 Fahrrichtungen
  - Orientierungsänderungen
- => Hindernis zwischen den Rädern

[Klamt and Behnke, IROS 2017]



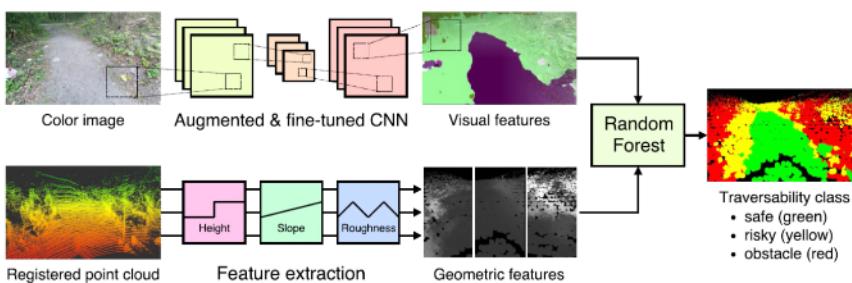
## Überwindung von Höhenunterschieden

- Wenn nicht befahrbarer Höhenunterschied vor einem Radpaar
- Aufsetzpunkt nach dem Höhenunterschied muss befahrbar sein
- Die drei anderen Beine müssen fahrbar sein

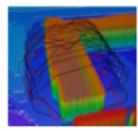
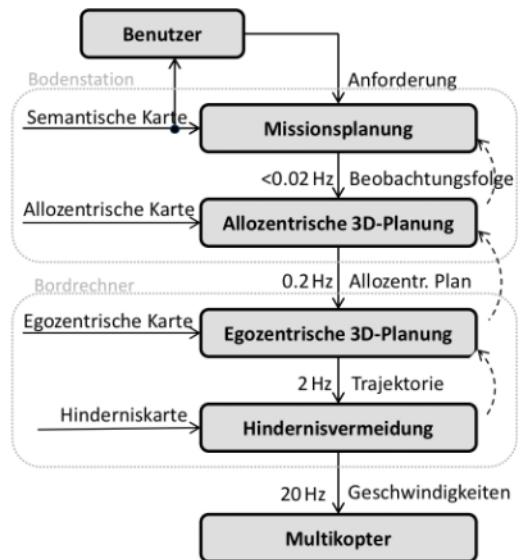


[Klamt and Behnke, IROS 2017]

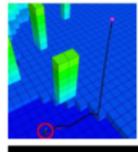
## Terrain-klassifikation



# Navigationsarchitektur



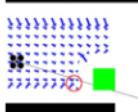
**Missionsplanung**



**Allozentrische Planung**



**Egozentrische Planung**



**Hindernisvermeidung**