Grundlagen der Robotik

7. Inverse Kinematik, Trajektoriengenerierung, Systeme

Prof. Sven Behnke

UNIVERSITÄT BONN

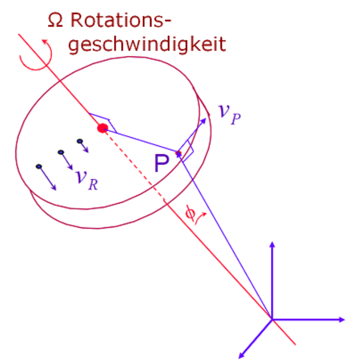
AIS

Letzte Vorlesung

- Basis-Jacobimatrix $\begin{pmatrix} v \\ \omega \end{pmatrix}_{(6x1)} = J_0(q)_{(6xn)}\,\dot{q}_{(nx1)}$
- Abbildung auf andere Positions- und Orientierungsrepräsentationen

$$J = \begin{pmatrix} J_{XP} \\ \hline J_{XR} \end{pmatrix} = \left(\begin{array}{c|c} E_P & 0 \\ \hline 0 & E_R \end{array}\right)\begin{pmatrix} J_v \\ \hline J_w \end{pmatrix}$$

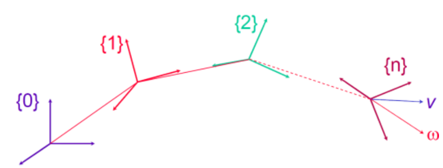
- Rotation erzeugt Lineargeschwindigkeit
Berechnung durch Kreuzprodukt: $v_P = \Omega \times P$



- Geschwindigkeitspropagierung

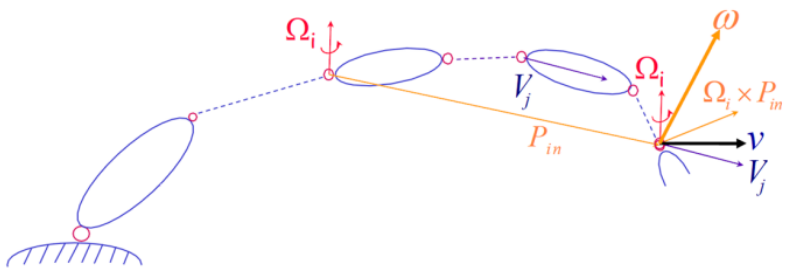
$${}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R.{}^{i}\omega_i + \dot{\theta}_{i+1}.{}^{i+1}Z_{i+1}$$

$${}^{i+1}v_{i+1} = {}^{i+1}_{i}R.({}^{i}v_i + {}^{i}\omega_i \times {}^{i}P_{i+1}) + \dot{d}_{i+1}.{}^{i+1}Z_{i+1}$$



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

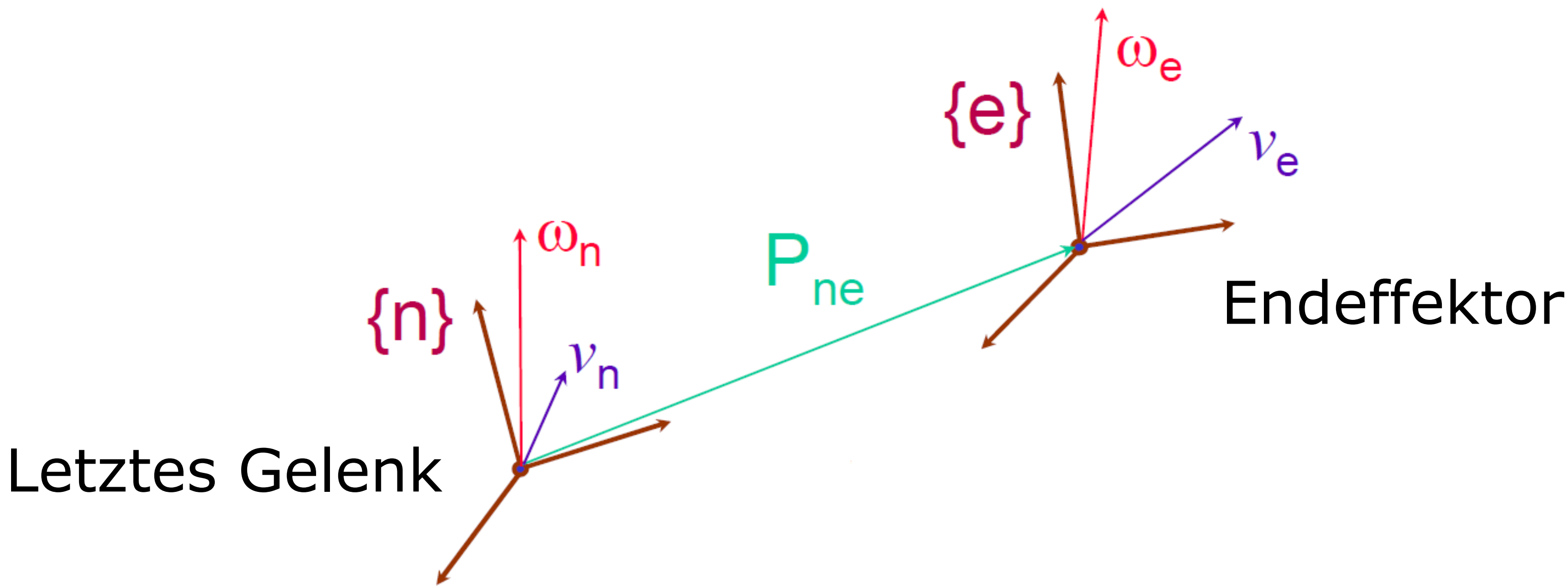
- Explizite Form



Einflüsse auf Endeffektor:	Lineargelenk	Drehgelenk
Lineargeschw.	V_j	$\Omega_i \times P_{in}$
Winkelgeschw.	—	Ω_i

- Kinematische Singularität $\det(J) = 0$

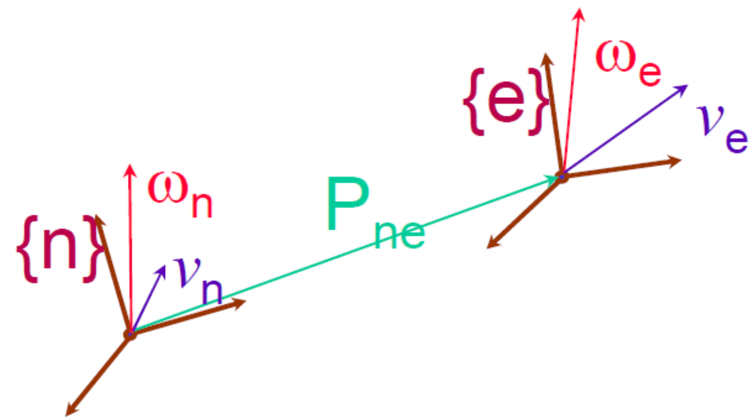
Jacobimatrix des Endeffektors



$$v_e = v_n + \omega_n \times P_{ne}$$

$$\begin{cases} v_e = v_n - P_{ne} \times \omega_n \\ \omega_e = \omega_n \end{cases} \qquad \begin{pmatrix} v_e \\ \omega_e \end{pmatrix} = \left(\begin{array}{c|c} \mathrm{I} & -\hat{P}_{ne} \\ \hline \mathrm{O} & \mathrm{I} \end{array}\right) \begin{pmatrix} v_n \\ \omega_n \end{pmatrix}$$

Framewechsel für Kreuzprodukt-Operator



- Wir wollen Endeffektor-Jakobimatrix in Frame {0}

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

- Wir brauchen Kreuzprodukt-Operator in Frame {0}

$$^0P \times {}^0\omega = {}^0_nR.({}^nP \times {}^n\omega)$$

$$^0\hat{P}.{}^0\omega = {}^0_nR.({}^n\hat{P}.{}^n\omega) = {}^0_nR.({}^n\hat{P}.{}^0_nR^T.{}^0\omega)$$

$$\boxed{{}^0\hat{P} = {}^0_nR \; {}^n\hat{P} \; {}^0_nR^T}$$

- Ausgehend von ^nJ_n: $^0J_e = \begin{pmatrix} {}^0_nR & -{}^0_nR\,{}^n\hat{P}_{ne}\,{}^0_nR^T \\ 0 & {}^0_nR \end{pmatrix} {}^nJ_n$

Inverse Kinematik mit inverser Jacobimatrix

- Jacobimatrix J linearisiert Beziehung zwischen Änderungen der Gelenkwinkel δθ und Änderungen der Endeffektorpose δx an der Stelle θ:

$$\delta x = J(\theta)\delta\theta$$

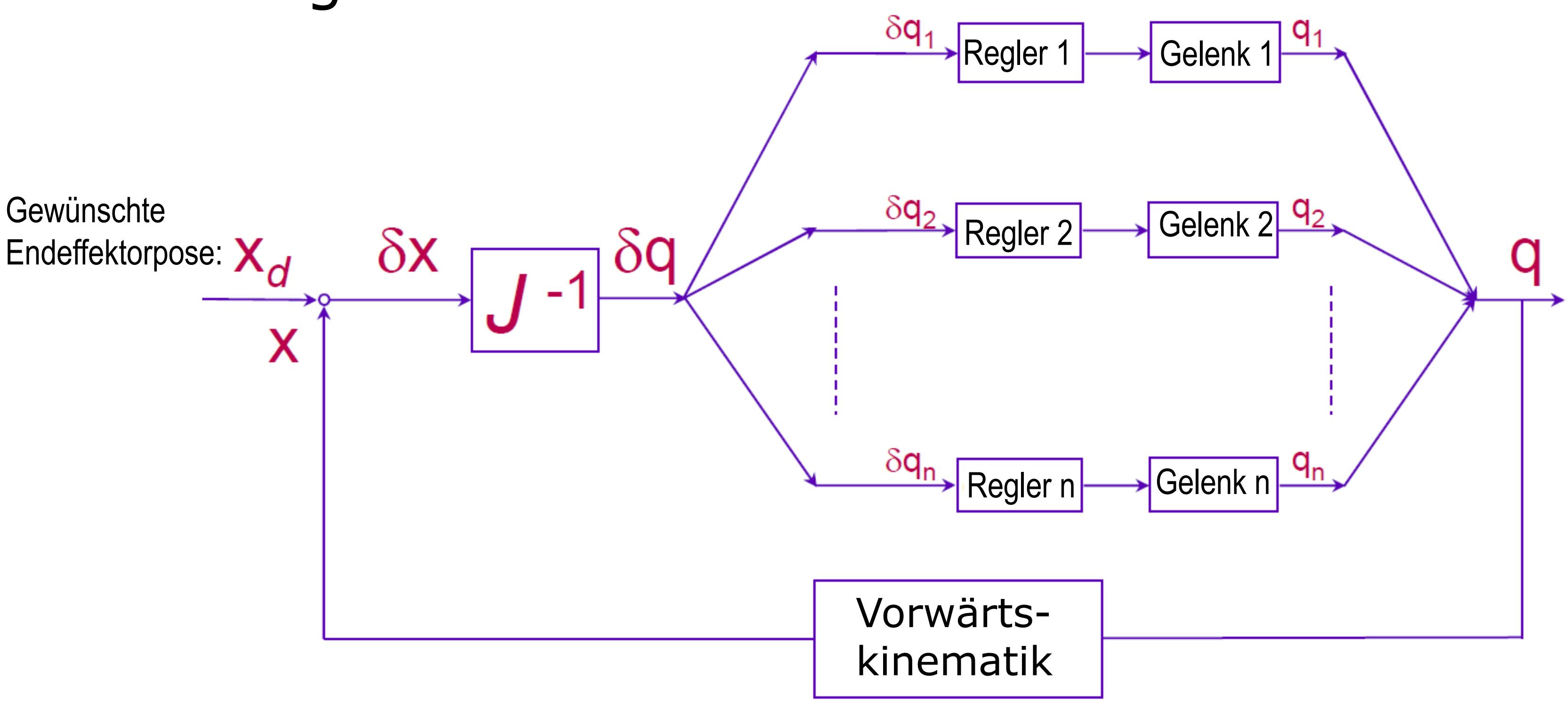
- Wenn J invertierbar (keine Singularität):

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

- Ausgehend von Gelenkstellung θ:
  - Vorwärtskinematik gibt Endeffektorpose: $x = f(\theta)$
  - Differenz zu gewünschter Pose x_d: $\delta x = x_d - x$
  - Notwendige Gelenkwinkeländerung: $\delta\theta = J^{-1}\delta x$
  - Resultierender Gelenkwinkel: $\theta^+ = \theta + \delta\theta$

Reglung der Endeffektorpose

- Rückführung auf Positionsregelung der Einzelgelenke



- Funktioniert bei langsamen Bewegungen
- Keine Berücksichtigung der Dynamik!

Mehrdeutigkeit inverser Kinematik

- Vorwärtskinematik bildet Gelenkwinkel immer eindeutig auf Endeffektorpose ab
- Eine Endeffektorpose kann aus mehreren Gelenkwinkelstellungen resultieren
- Bei redundanten Armen gibt es eine ganze Mannigfaltigkeit von Gelenkstellungen, welche die gleiche Endeffektorpose erzeugen: den Nullraum

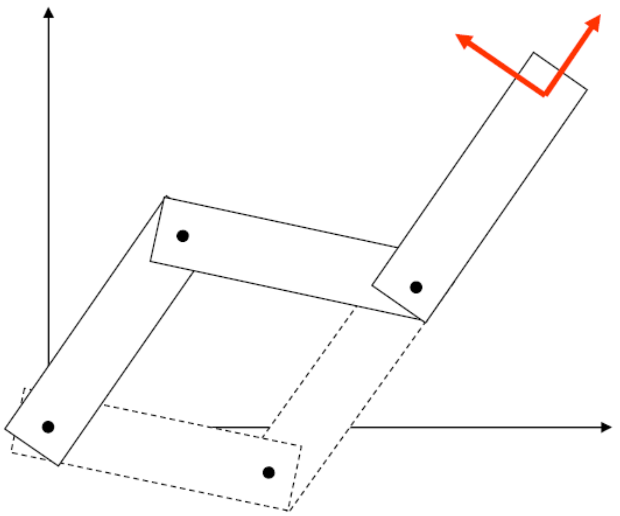
=> Benutze Pseudo-Inverse

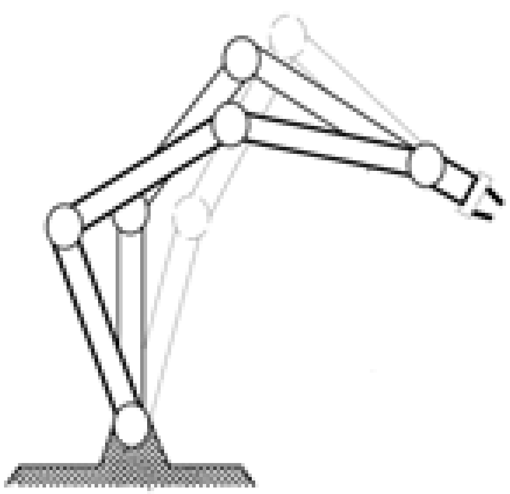
$$J^+ = (J^T J)^{-1} J^T$$

Linksinverse: $J^+ J = I$
(Spalten von J
unabhängig)

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

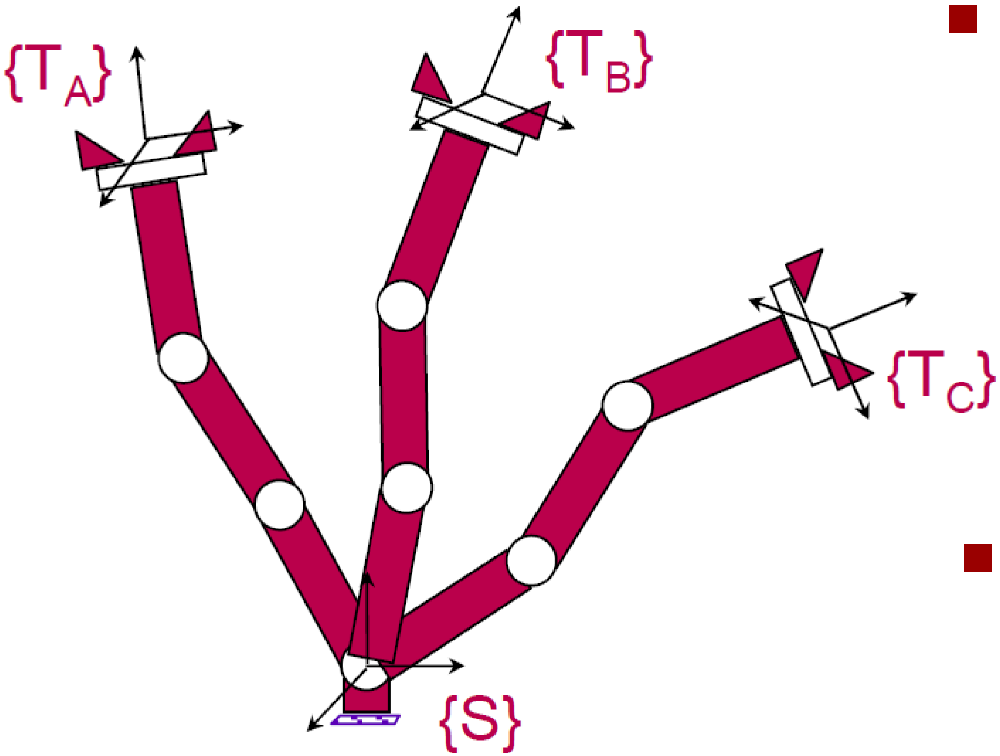
=> Optimiere Kostenfunktion im Nullraum,
z.B. Abstand zu Gelenklimits,
Drehmoment, …





Erzeugung von Trajektorien

- Ziel: Bewegung des Endeffektors von einer Anpangspose $\{T_A\}$ zu einer Endpose $\{T_C\}$
- Möglicherweise Zwischenposen $\{T_B\}$



- Nebenbedingungen
  - Räumlich (Hindernisse)
  - Zeitlich (Synchronisation)
  - Glattheit (Energie)

- Ausgabe: Trajektorie, d.h. Position, Geschwindigkeit, Beschleunigung aller Gelenke als Funktion der Zeit

Ansätze zur Trajektorienerzeugung

- Im Gelenkraum
  - Löse inverse Kinematik an allen gegebenen Pfadpunkten
  - Interpoliere im Gelenkraum

  => wenig Rechenaufwand

  => keine Probleme mit Singularitäten

  => kann keiner geraden Line folgen

- Im Arbeitsraum
  - Interpoliere die Koordinaten des Endeffektors
  - Löse inverse Kinematik für jeden Zeitschritt

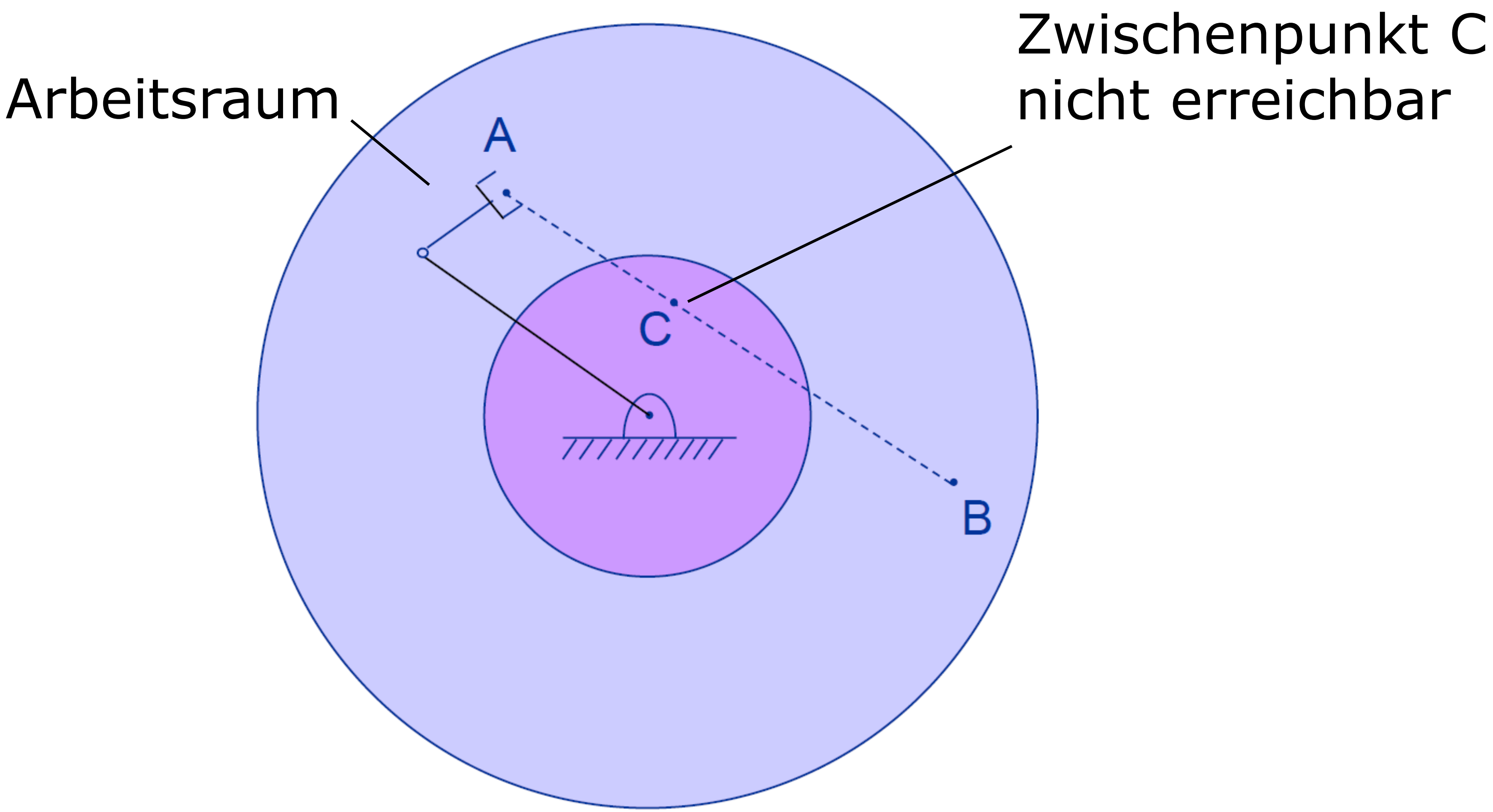
  => man kann genaue Trajektorie erzeugen

  => hoher Rechenaufwand

  => Probleme mit Erreichbarkeit, Singularitäten, mehrdeutigen Lösungen

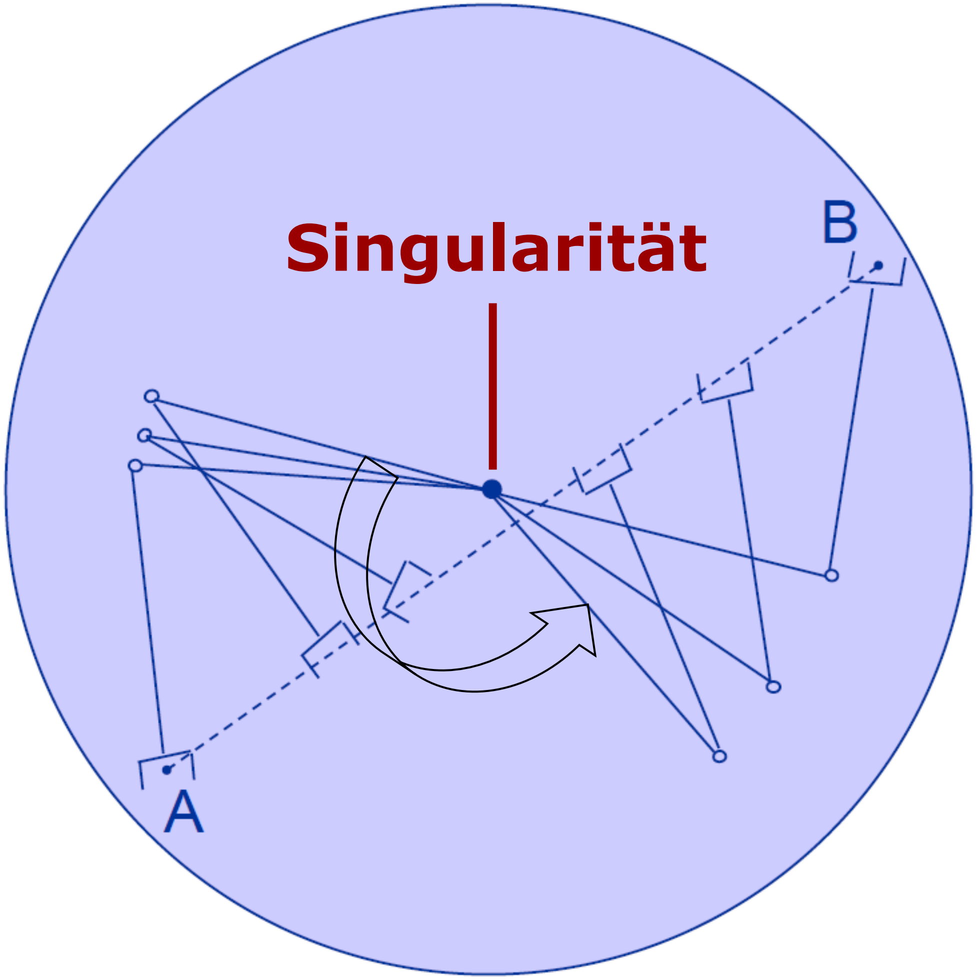
Beispiel für Nichterreichbarkeit

- Planarer Roboter mit zwei Rotationsgelenken



Beispiel für Singularität

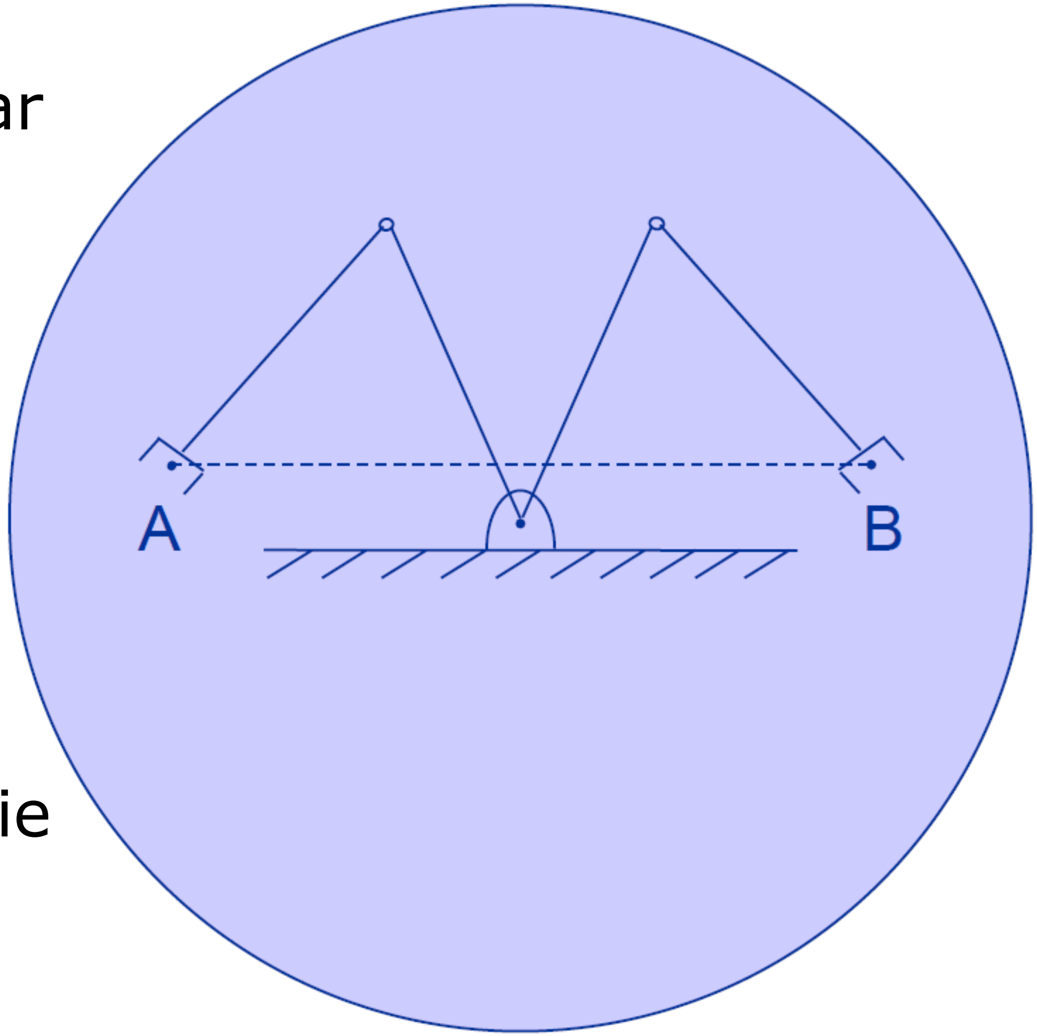
- Bei Annäherung an die Singularität nähern sich die erforderlichen Gelenkgeschwindigkeiten ∞, was Abweichungen vom Pfad erzeugt



Beispiel für nicht verbundene Bereiche des Gelenkraums

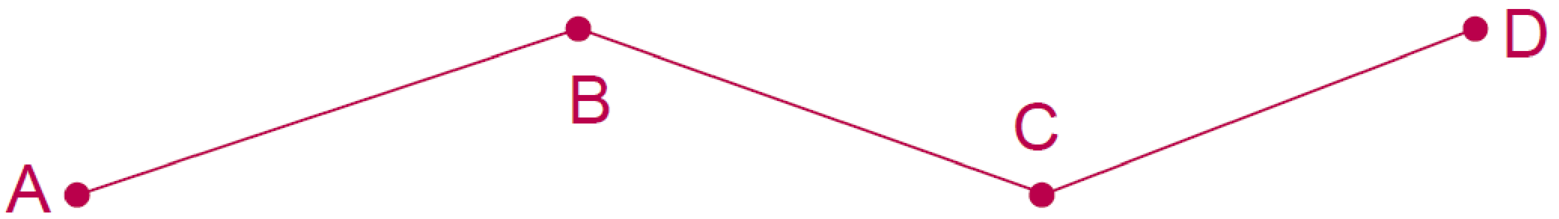
- Obwohl Start- und Zielpunkt erreichbar sind, befinden sie sich in nicht verbundenen Bereichen des Gelenkraums

- Die Punkte in der Mitte der Trajektorie sind von unten erreichbar

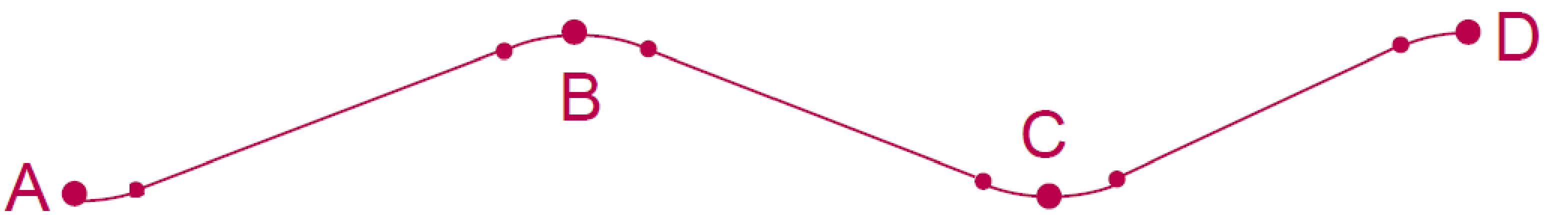


Interpolationsmethoden

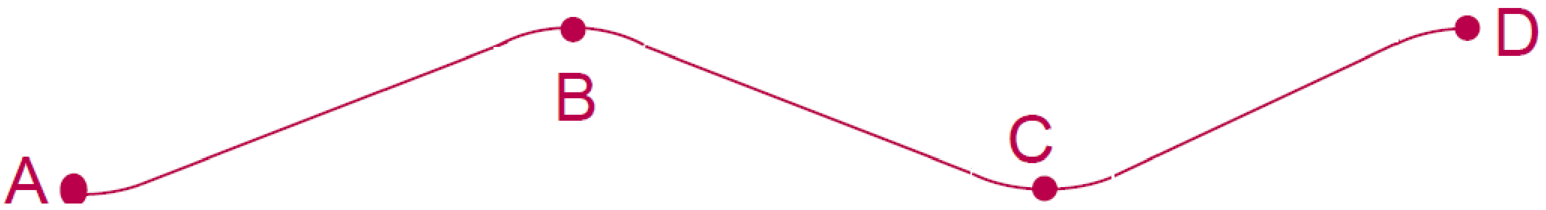
- Lineare Interpolation => Unstetigkeit



- Linear mit glatten Übergängen



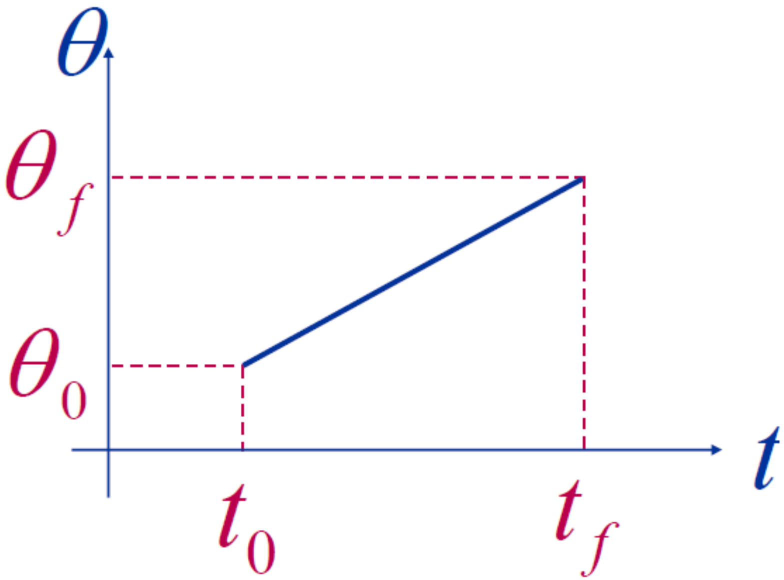
- Kubische Polynome (Splines)



- Polynome höherer Ordnung

Lineare Interpolation

- Zwei Koeffizienten: $\theta(t) = a_0 + a_1 t$



- Randbedingungen: $\theta(t_0) = \theta_0$

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

=> Unstetigkeit in der Geschwindigkeit

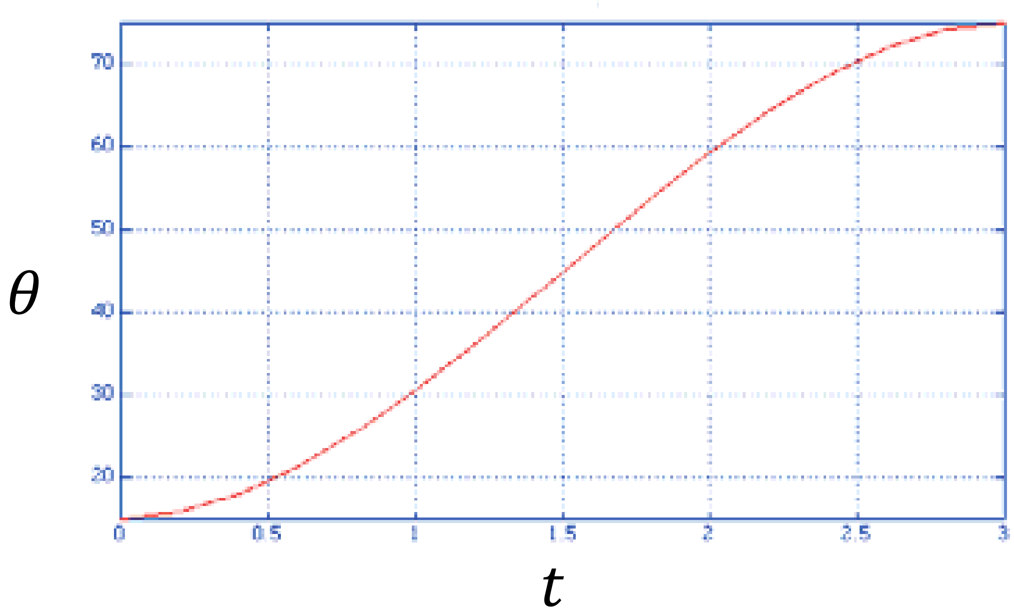
Einzelnes Kubisches Polynom

- Koeffizienten a_i bestimmen Form:

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

- Randbedingungen:

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

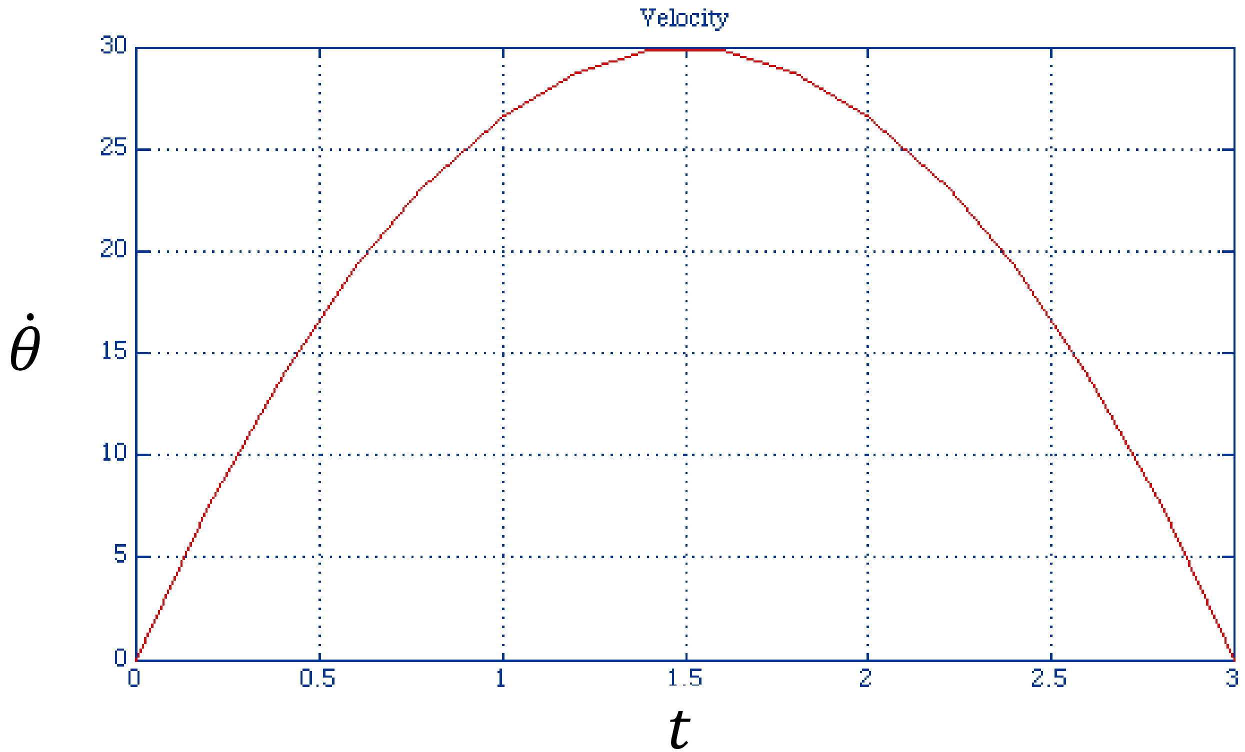


Einzelnes Kubisches Polynom

- Geschwindigkeit:

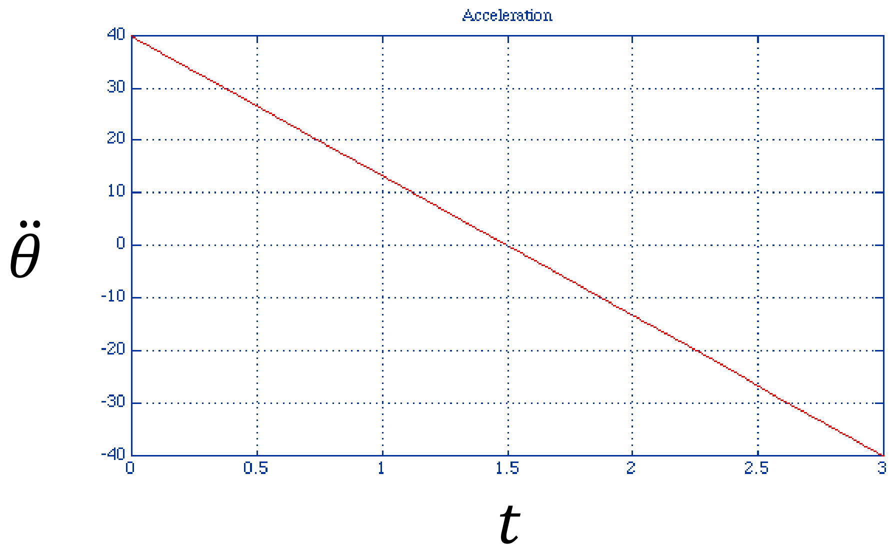
$$\dot{\theta}(t) = a_1 + 2a_2 t + 3a_3 t^2$$

- Randbedingungen: $\dot{\theta}(0) = 0 \; ; \; \dot{\theta}(t_f) = 0$



Einzelnes Kubisches Polynom

- Beschleunigung: $\ddot{\theta}(t) = 2a_2 + 6a_3 t$
- Ruck (konstant): $\dddot{\theta}(t) = 6a_3$



- Lösung:

$$\theta(t) = \theta_0 + \frac{3}{t_f^2}(\theta_f - \theta_0)t^2 + \left(-\frac{2}{t_f^3}\right)(\theta_f - \theta_0)t^3$$

Mit Geschwindigkeiten am Rand

- Randbedingungen: $$\dot{\theta}(0) = \dot{\theta}_0$$ $$\dot{\theta}(t_f) = \dot{\theta}_f$$

- Lösung:

$$a_0 = \theta_0$$

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

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

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

Bestimmung der Geschwindigkeiten an Zwischenpunkten

- Aus Geschwindigkeiten im Arbeitsraum

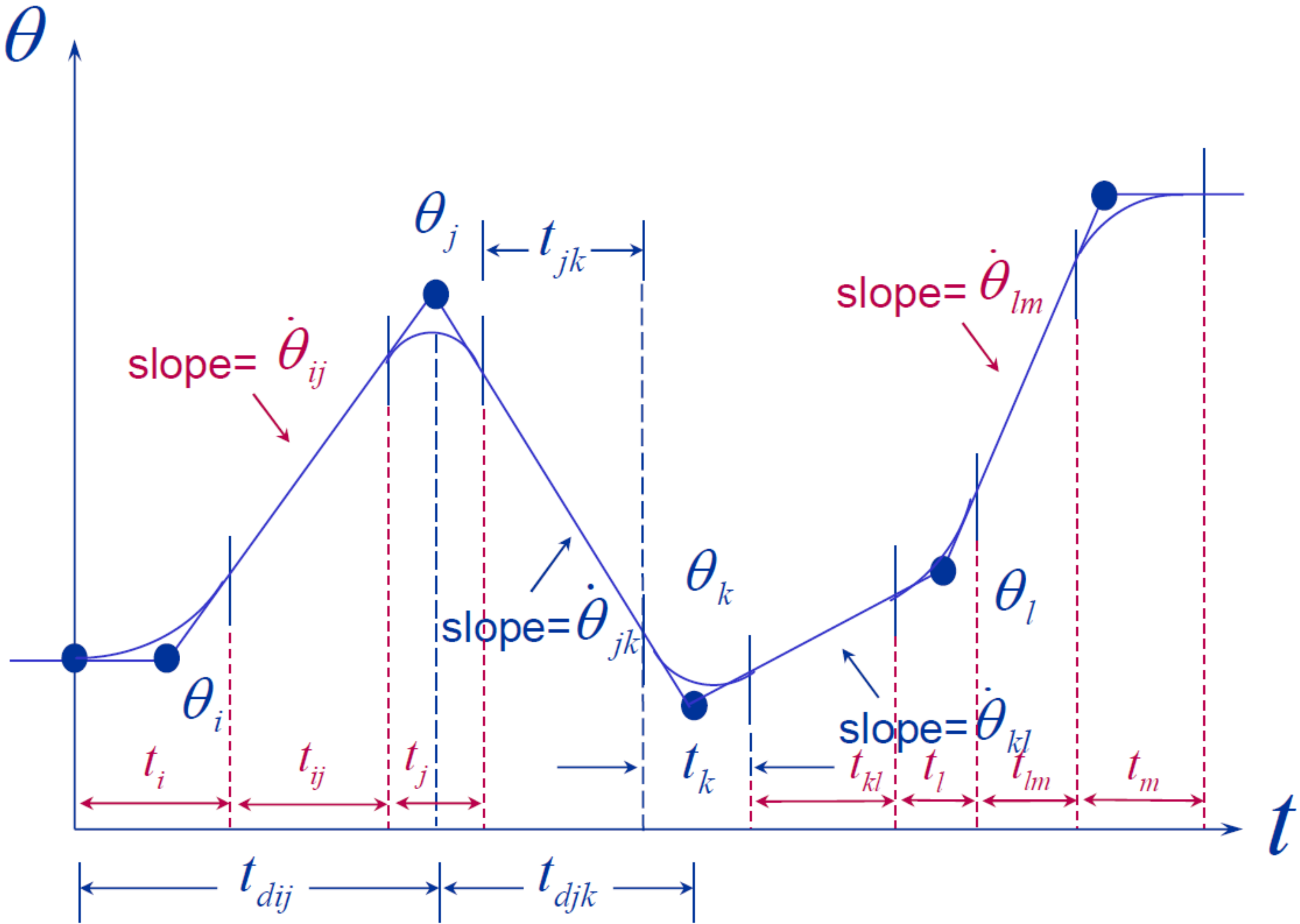
$$\dot{\theta} = J^{-1} \begin{pmatrix} \mathbf{v} \\ \omega \end{pmatrix}$$

- Heuristisch, z.B. Durchschnitt beider Seiten

- Kontinuitätsbedingung
  - Geschwindigkeit: $\dot{\theta}_1(t_f) = \dot{\theta}_2(0)$
  - Beschleunigung: $\ddot{\theta}_1(t_f) = \ddot{\theta}_2(0)$

Lineare Interpolation mit glatten Übergängen

- Konstante Beschleunigung an Übergängen $\ddot{\theta}(t) = a$



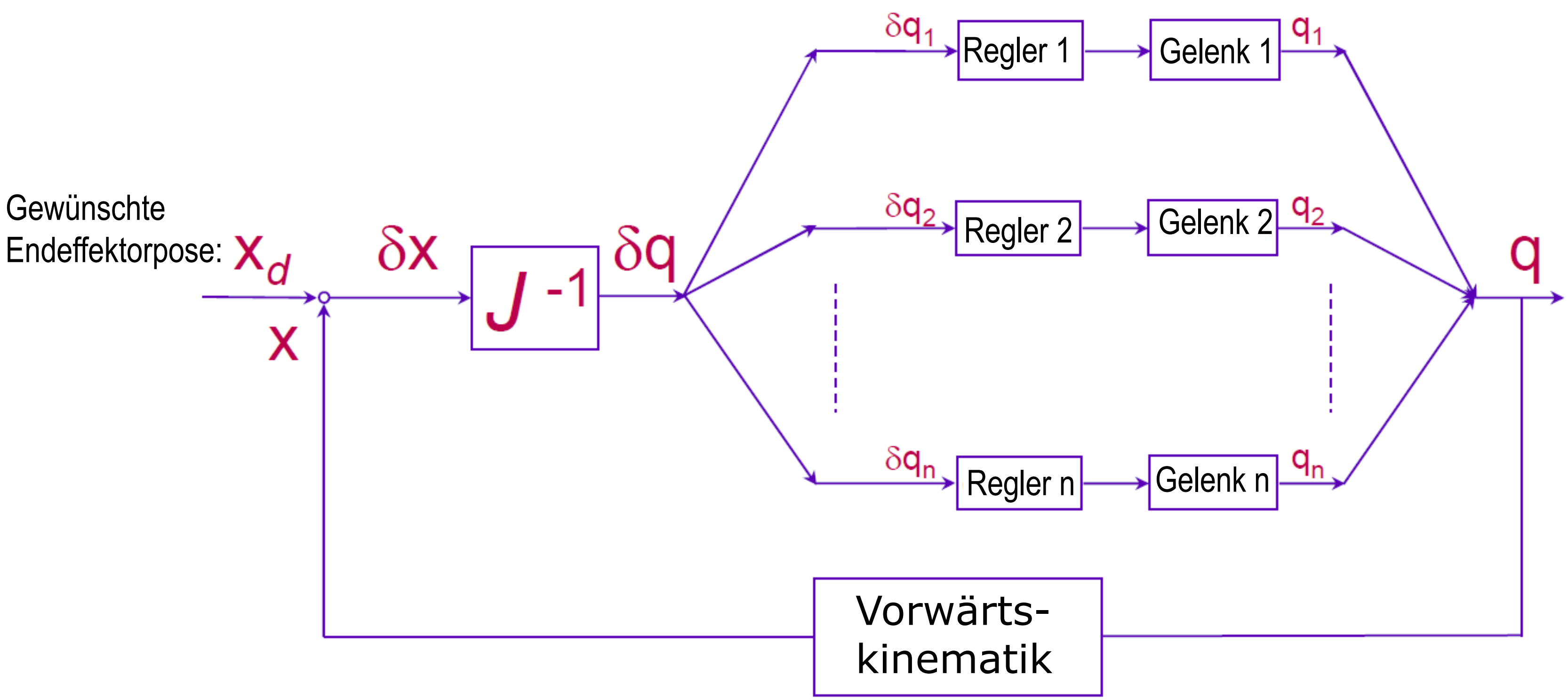
Grundlagen der Robotik

Systeme

Prof. Sven Behnke

Erinnerung

- Rückführung Reglung der Endeffektorpose auf Positionsregelung der Einzelgelenke



=> Wir brauchen Regler!

Regelungstechnik-Grundlagen

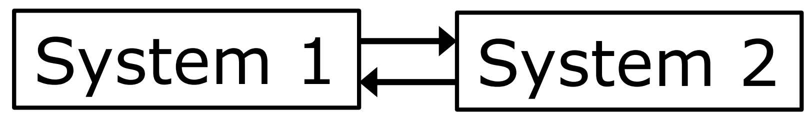
- In eingebetteten Systemen werden physikalische Größen, wie Geschwindigkeit, Position, etc. durch Computer geregelt.
- Häufiges Ziel: Geregelte Größe soll konstant bleiben
- => Zurückweisung von Störungen nötig
- => Wir brauchen ein Modell des geregelten Systems
- Generelles Prinzip: **Feedback**

Was ist Feedback?

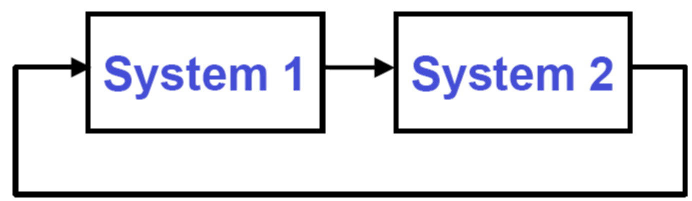
- Rückkopplung des Zustands eines Systems zur Steuerungseinheit

- Feedback = gegenseitige Beeinflussung von zwei (oder mehr) Systemen
  - System 1 beeinflusst System 2
  - System 2 beeinflusst System 1
  - Ursache und Wirkung können nicht einfach unterschieden werden; Systeme sind wechselseitig abhängig

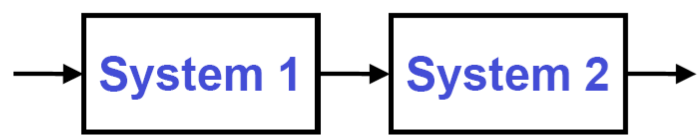
- Rückkopplung kommt überall in der Natur und in technischen Systemen vor



- Terminologie:



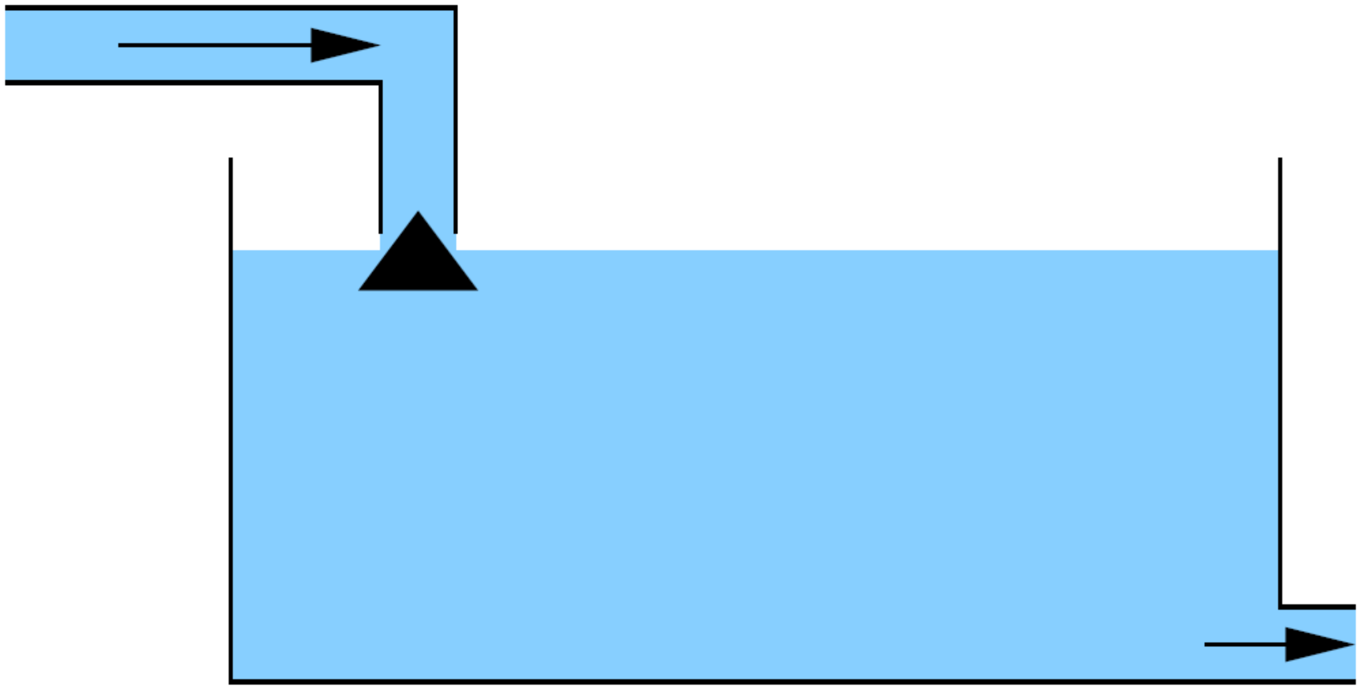
Closed Loop



Open Loop

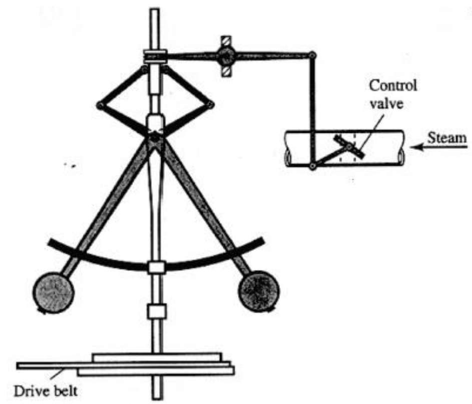
Feedback-Regelungssystem

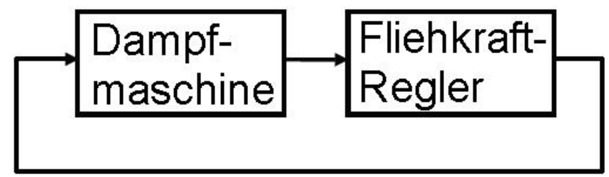
- Feedback-Prinzip ist seit Antike bekannt
- Beispiel: Füllstandsregelung

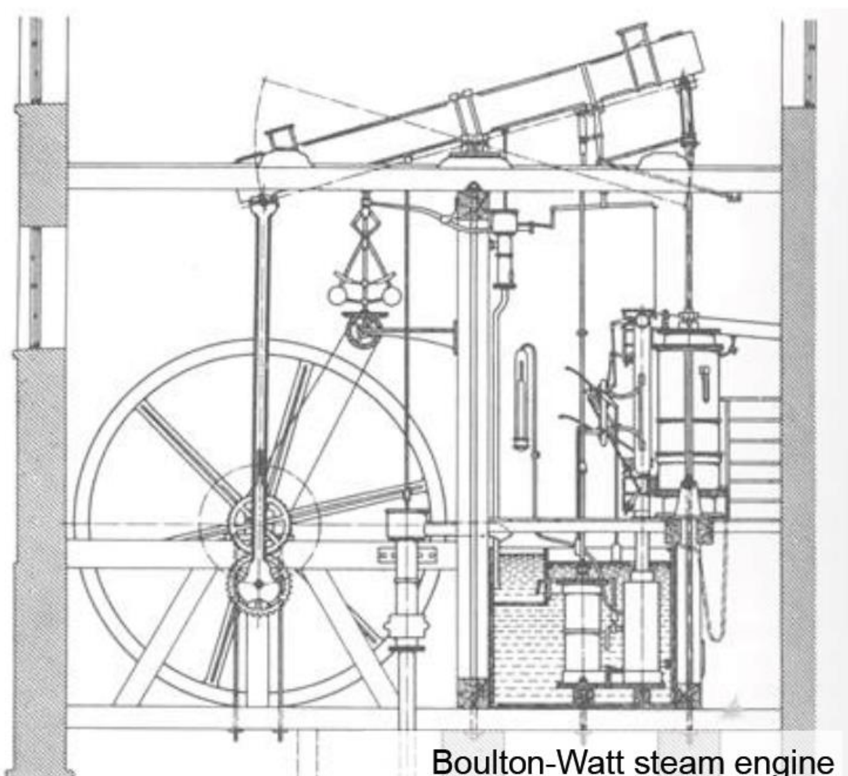


Beispiel: Fliehkraftregler

- 1788 von James Watt entwickelt
- Reduziert Effekt schwankender Belastung
- Wichtiger Schritt zur industriellen Revolution



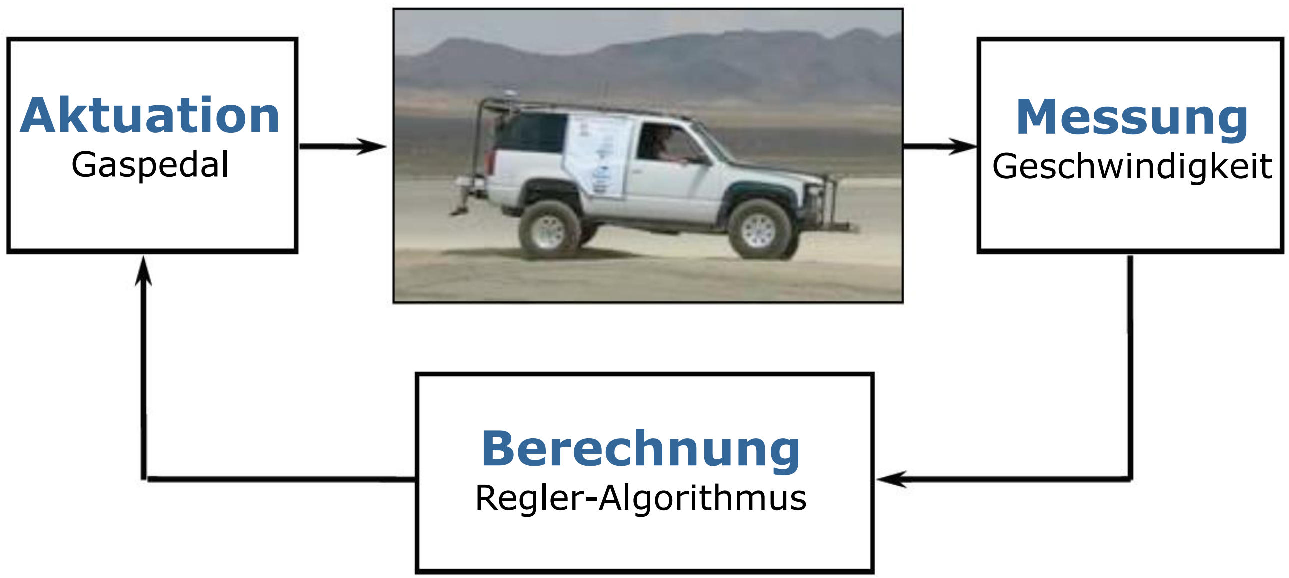



Boulton-Watt steam engine


Courtesy Eric Klavins, U. Washington (2008)

Regelung = Messung + Berechnung + Aktuation

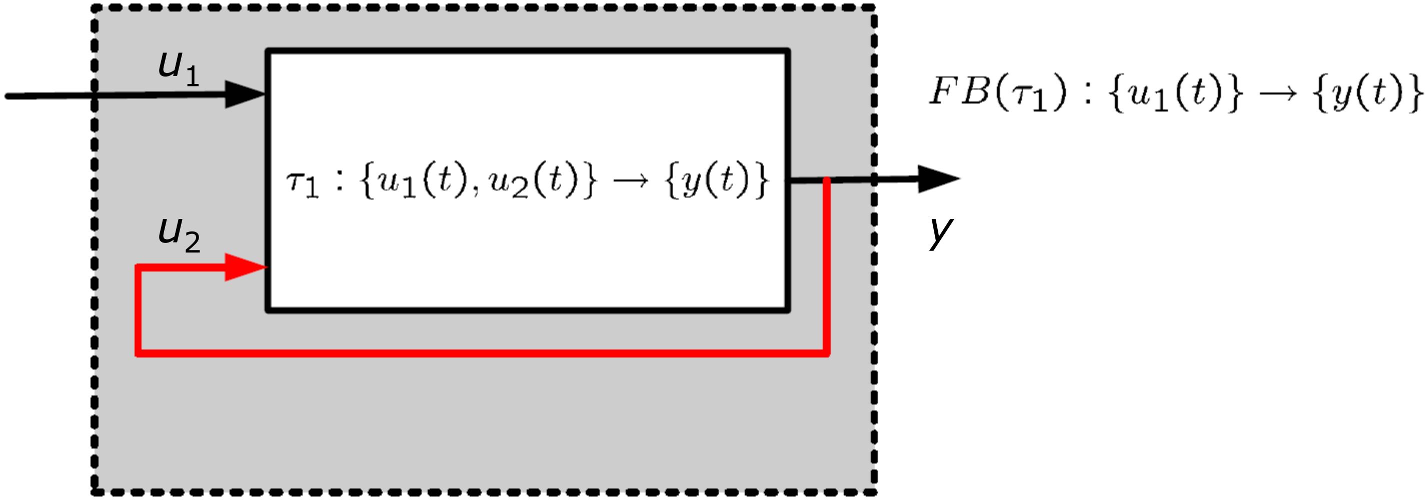
$\underbrace{\text{Messung + Berechnung + Aktuation}}_{\text{In Rückkopplungsschleife}}$



Ziele:

- **Statische Performanz**: System hält erwünschten Zustand (z.B. erlaubte Geschwindigkeit)
- **Dynamische Performanz**: System kann schnell auf Veränderungen reagieren (z.B. neue Zielgeschwindigkeit)
- **Robustheit**: System toleriert Änderungen in der Dynamik (Masse, Anstieg der Straße, …)

Rückkopplung



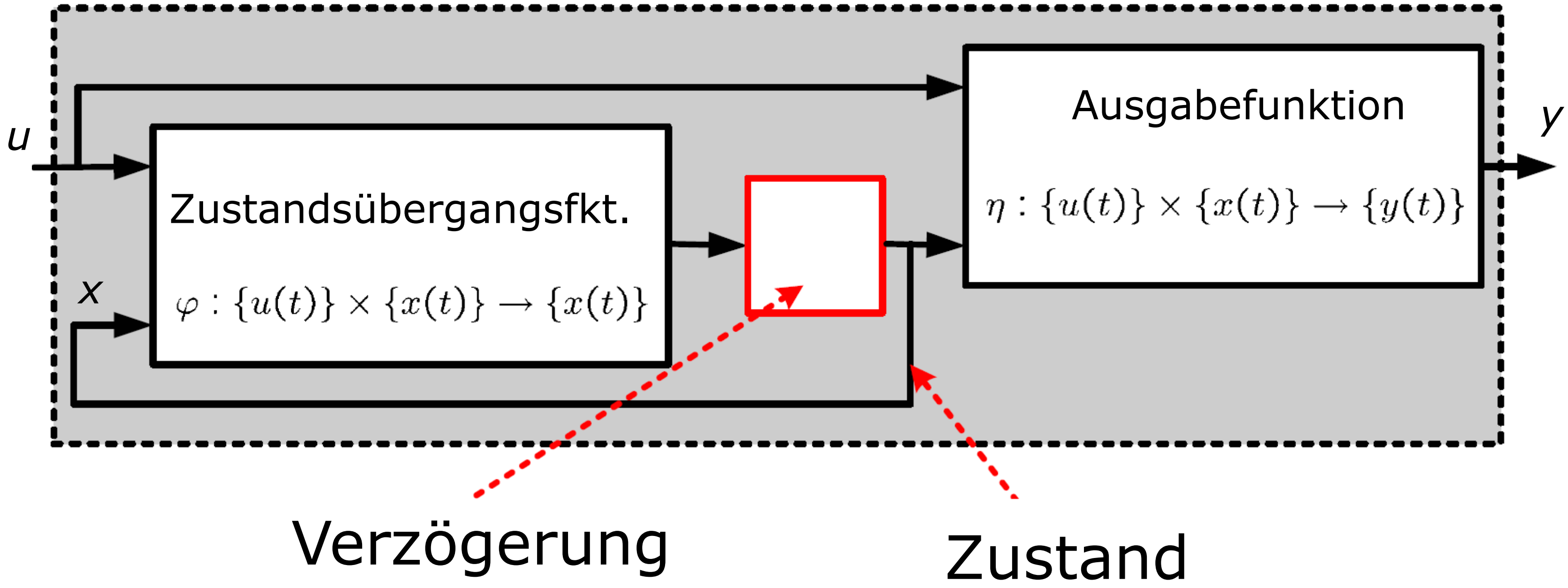
$FB(u_1(t))$ ist die Menge aller Werte, für die gilt:

$$y(t) = u_2(t) \quad \forall t \in T$$

d.h. $FB(u_1(t))$ ist die Lösung eines Gleichungssystems

Systeme mit Internem Zustand

- Modell eines dynamischen Systems



- **Zustandsübergangsfunktion** beschreibt Dynamik
- **Ausgabefunktion** ist nicht dynamisch

Klassen Dynamischer Systeme

- Systeme mit internen Zustand werden als dynamische Systeme bezeichnet

- Unterscheidung nach Zeitbasis:
  - **Zeitdiskret**
    - Zustand ändert sich regelmäßig
  - **Zeitkontinuierlich**
    - Zustand ändert sich kontinuierlich
  - **Ereignisbasierte Systeme**
    - Zustand ändert sich nur bei Ereignissen

Beispiel für Zeitdiskretes Sytem

Entwicklung einer Population

- Zeitbasis: $T = 2012, 2013, \ldots$
- Eingabe: $geburtsrate_{alter}[t]$, $sterblichkeit_{alter}[t]$, $alter \in [0,99]$
- Ausgabe: $population[t]$
- Zustand: Altersverteilung $personen_{alter}[t]$, $alter \in [0,99]$
- Zustandsübergang:

$$personen_0\,[\mathrm{t}] := \sum_{a=16}^{a=40} geburtsrate_a[t] \cdot personen_a[t\text{-}1]$$

$$personen_a\,[\mathrm{t}] := personen_{a\text{-}1}[t\text{-}1] \cdot (1\text{-}sterblichkeit_a[t]),\ a \in [1,99]$$

- Ausgabewert: $population[t] := \sum_{a=0}^{a=99} personen_a[t]$

Zeitdiskrete Systeme

Ein zeitdiskretes System ist ein dynamisches System mit:

- Zeitbasis isomorph zu natürlichen Zahlen

$$T = t_c \cdot \mathbb{N}$$

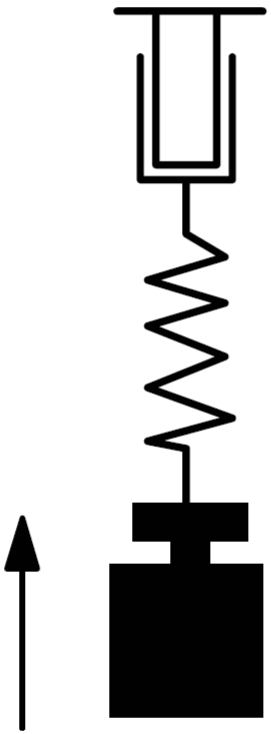
- Zustandsraum X ist beliebige Menge
- Eingabe U aus beliebiger Menge
- Eingabesignale $u[n]$ sind diskrete Folgen
- Ausgabe Y ist beliebige Menge
- Ausgabesignale $y[n]$ sind diskrete Folgen
- Zustandsübergangsfunktion als **Tabelle**:

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

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

Zeitkontinuierliche Systeme

Beschreibung durch Differentialgleichungen



$$\dot{v} = a$$

$$\dot{y} = v$$

$$y = \frac{1}{k}(ma + cv + F_{ext})$$

Feder-
konstante

Beschleuni-
gungskraft

Dämpfung

Externe
Kräfte

Zeitkontinuierliche Systeme

- $T = \mathbb{R}$.
- X reellwertiger Vektorraum $\mathbb{R}^n$ (Zustandsraum)
- U reellwertiger Vektorraum $\mathbb{R}^m$ (Eingaberaum)
- Eingabesignale $u(t)$ sind stückweise stetig
- Y reellwertiger Vektorraum $\mathbb{R}^p$ (Ausgaberaum)
- Zustandsübergang $\varphi: X \times U \longrightarrow X$

  beschrieben durch **Differentialgleichungen**,
  z.B. $\dot{x} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$
- Ausgabefunktion:

  $\eta : X \rightarrow Y$ oder $\eta : X \times \{u(t)\} \rightarrow Y$

Systemeigenschaften

- Unterschiedliche Systeme können gleiche Eigenschaften haben, was es erlaubt ein gemeinsames Modell zu nutzen

- Wichtige Systemeigenschaften:
  - Zustandslos oder mit Zustand
  - Kausalität
  - Linearität
  - Zeitinvarianz
  - Stabilität

Zustandslose Systeme vs. Systeme mit Zustand

- Zustandsloses System:

  Ausgabe $y(t)$ zum Zeitpunkt t hängt nur von der aktuellen Eingabe $u(t)$ ab.

- System mit Zustand:

  Ausgabe $y(t)$ zum Zeitpunkt t hängt auch von Eingaben $u(t')$ zu anderen Zeitpunkten t' ab.

Kausale Systeme

- Kausalität bezeichnet die Eigenschaft, dass eine Reaktion zeitlich nicht vor der Ursache $u(t)$ erfolgen kann.

- Formal: Wenn zwei Eingabesignale $u_1(t), u_2(t)$ bis zum Zeitpunkt τ identisch sind, dann muss die Ausgabe auch bis zu diesem Zeitpunkt identisch sein.

- Ein System ist kausal wenn:

$$u_1(t) = u_2(t) \forall t \leq \tau \quad \Rightarrow \quad \mathcal{T}(u_1(t)) = \mathcal{T}(u_2(t)) \forall t \leq \tau$$

Kausale Signale

- Die Eigenschaft Kausalität wird auch für Signale verwendet

- Ein Signal ist kausal wenn

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

- Ein nicht-kausales Signal $f(t)$ kann kausal gemacht werden, wenn man es mit der Stufenfunktion $s(t)$ multipliziert.

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

$s(t)$=0 für $t < 0$
$s(t)$=1 für $t >= 0$

Kausalität: Beispiele

- Ist die Systemfunktion $\mathcal{T} : y(t) = u(t-1)^2$ kausal?
- Ist die Systemfunktion $\mathcal{T} : y(t) = u(t+1) + t^2$ kausal?
- Ist das Signal $f(t) = e^{-t}$ kausal?
- Ist das Signal $f(t) = s(t)e^{-t}$ kausal?

Lineare Systeme

- Zwei Eigenschaften müssen erfüllt sein:
  - Skalierung der Eingabe mit einem Faktor k skaliert die Ausgabe mit dem Faktor k **(Skalierungseigenschaft)**
  - Es macht keinen Unterschied, ob zwei Signale vor oder nach Durchgang durch das System addiert werden **(Superpositionseigenschaft)**
- Ein System ist linear wenn:

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

Beispiel: Lineares System

- Ein System berechnet seine Ausgabe $y[n]$ als Summe der letzten beiden Eingaben $u[n]$ und $u[n-1]$:

$$y[n] = u[n] + u[n-1]$$

- Um Linearität zu zeigen, müssen wir zeigen:

$$\mathcal{T}\{\sum_{i=1}^{K} k_i u_i[n]\} = \sum_{i=1}^{K} k_i \mathcal{T}\{u_i[n]\}$$

oder $\quad c\mathcal{T}\{u[n]\} = \mathcal{T}\{cu[n]\}\quad$ (Skalierung)

und

$$\mathcal{T}\{u_1[n] + u_2[n]\} = \mathcal{T}\{u_1[n]\} + \mathcal{T}\{u_2[n]\}$$

(Superposition)

Beispiel: Lineares System

- Für unser Beispielsystem $y[n] = u[n] + u[n-1]$ gilt:
  - Skalierung

$$
\begin{aligned}
c\mathcal{T}\{u[n]\} &= c(u[n] + u[n-1]) \\
&= cu[n] + cu[n-1] \\
&= \mathcal{T}\{cu[n]\}
\end{aligned}
$$

  - Superposition

$$
\begin{aligned}
\mathcal{T}\{u_1[n]\} + \mathcal{T}\{u_2[n]\} &= (u_1[n] + u_1[n-1]) + (u_2[n] + u_2[n-1]) \\
&= (u_1[n] + u_2[n]) + (u_1[n-1] + u_2[n-1]) \\
&= \mathcal{T}\{u_1[n] + u_2[n]\}
\end{aligned}
$$

Also ist es linear!

Zeitinvariante Systeme

- Man spricht von Zeitinvarianz, wenn das Systemverhalten nicht von der Zeit abhängt
- Zeitverschiebung der Eingabe resultiert in identischer Ausgabe, die nur um die gleiche Zeit verschoben ist
- Für zeitinvariante Systeme gilt:

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

Signal-Dekomposition

Die Superpositionseigenschaft linearer Systeme erlaubt es, komplexe Probleme auf einfachere Probleme zurückzuführen

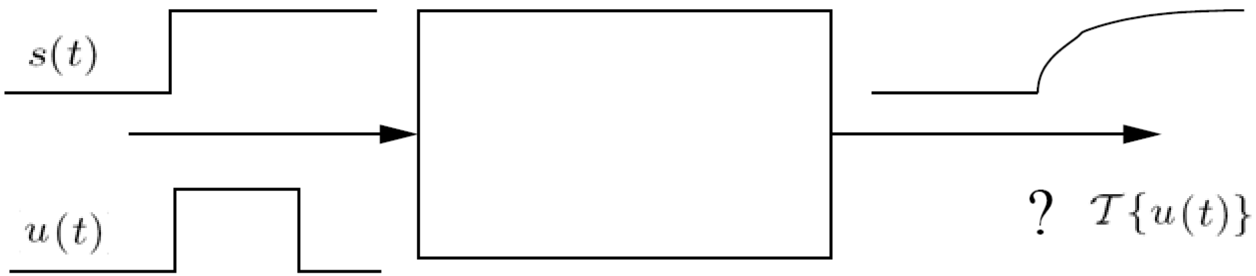
Prinzip:

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

Beispiel: Signaldekomposition

- Gegeben: Ausgabe für Stufenfunktion:

$$\mathcal{T}\{s(t)\} = s(t)(1 - e^{-t})$$



- Gesucht: Ausgabe für Signal u(t)
- Idee: zerlege Signal u(t) in Stufenfunktionen

Stabilität

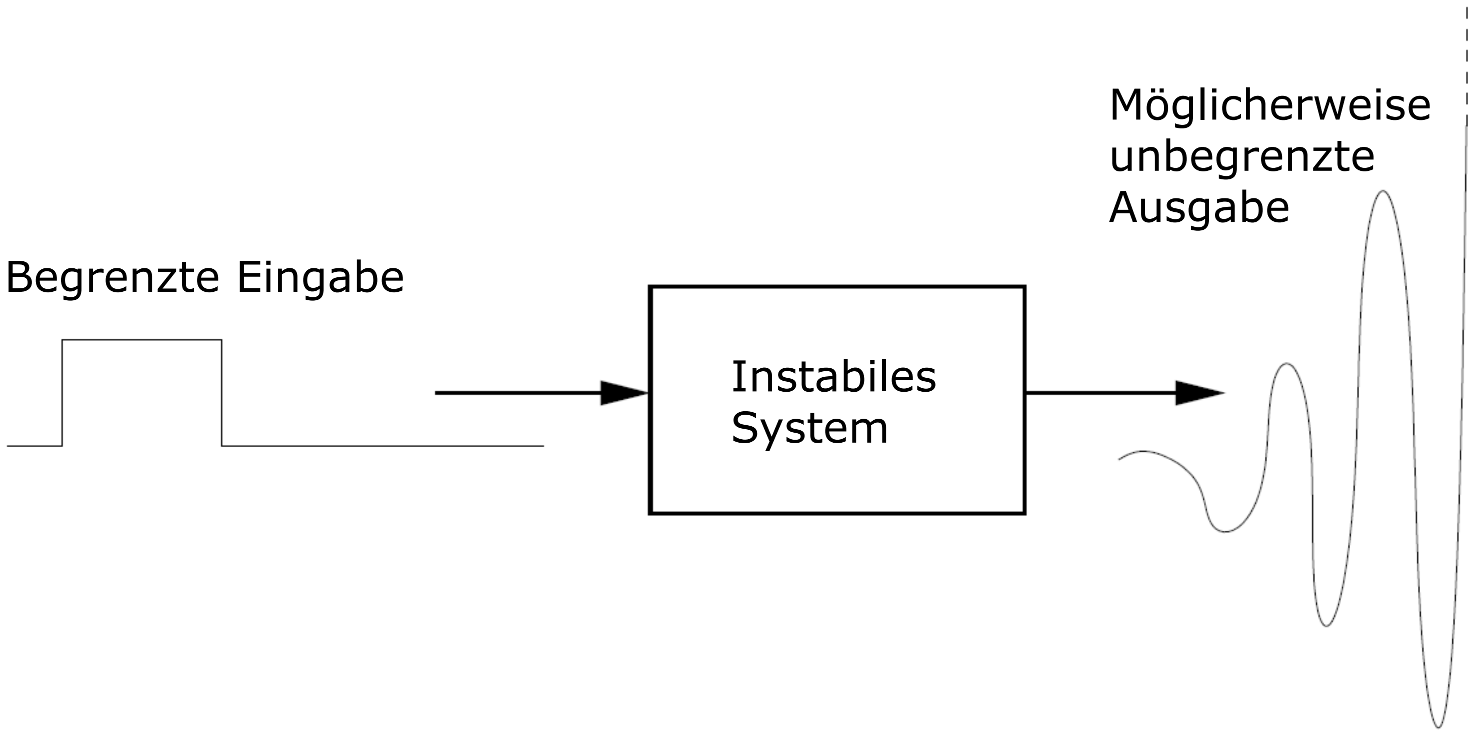
- Ein System wird stabil genannt, wenn für beliebige Eingabesignale $u(t)$, die einen Schwellwert M nicht überschreiten, man einen Schwellwert N finden kann, den die Ausgabe $y(t) = \mathcal{T}\{u(t)\}$ nicht übersteigt.

- Ein System ist stabil wenn:

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

Instabile Systeme

- Das Fehlen von Stabilität in physikalischen Systemen führt häufig zu deren Zerstörung … ☹



Beispiel: Stabilität

- Ein System berechnet seine Ausgabe $y[n]$ als Summe der letzten beiden Eingaben $u[n]$ und $u[n-1]$: $$y[n] = u[n] + u[n-1]$$
- Frage: Ist das System stabil?
- Behauptung: Das System ist stabil.
- Um Stabilität zu zeigen, müssen wir zeigen

$$|u[n]| < M < \infty \Rightarrow |\mathcal{T}\{u[n]\}| < N < \infty$$

- Sei $u[n]$ eine Eingabe mit $|u[n]| < M$;

$$|\mathcal{T}\{u[n]\}| = |u[n] + u[n-1]| \leq |M + M| = |2M|$$

- Die Ausgabe ist begrenzt durch $N = 2M$

=> Stabilität.

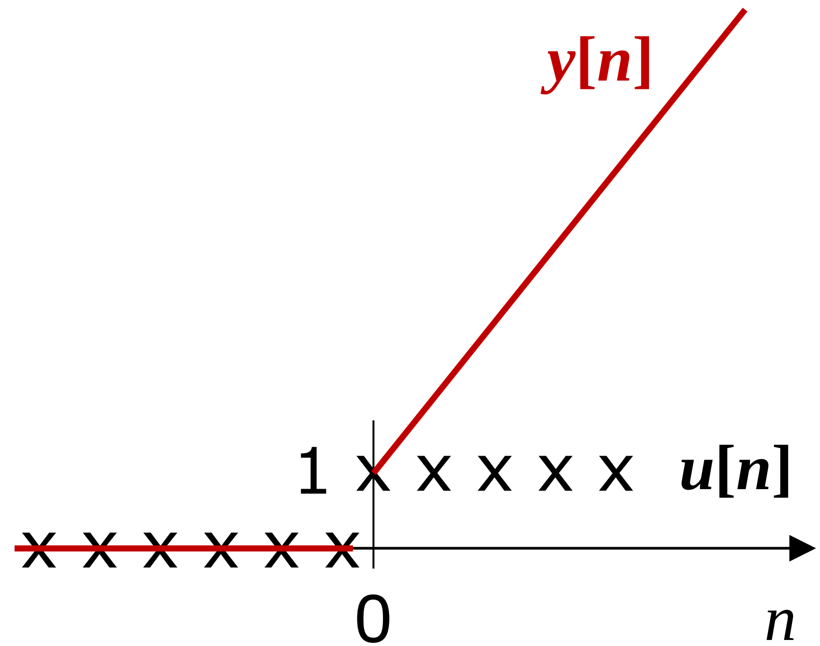
Beispiel: Instabilität

- Ein System berechnet seine Ausgabe $y[n]$ als Summe von Eingabe $u[n]$ und voriger Ausgabe $y[n-1]$:

$$y[n] \; = \; u[n] \; + \; y[n-1]$$

- Wie sieht die Ausgabe für folgende Eingabe aus:
  $u[n] \; = \; 0, \text{ für } n < 0$
  $u[n] \; = \; 1, \text{ für } n \geq 0 \quad ?$

- $y[n] = 0, \text{ für } n < 0$
  $y[n] = n + 1, \text{ für } n \geq 0$



Ausgabe wächst unbegrenzt => System instabil!

Lineare zeitinvariante Systeme (LTI-Systeme)

- **Linearität:** erlaubt, das Superpositionsprinzip anzuwenden
- **Zeitinvarianz:** erlaubt, Signale zeitlich zu verschieben
- Systeme, die beide Eigenschaften haben, werden lineare zeitinvariante Systeme genannt (linear time-invariant (LTI))
- Können zeitdiskret oder kontinuierlich sein