

Regelungstechnik 2

Zustandsraumdarstellung

$$\dot{\vec{x}} = \mathbf{A} \cdot \vec{x} + \mathbf{B} \cdot \vec{u}(t), \quad \text{Zustandsgleichung}$$

$$\vec{y}(t) = \mathbf{C} \cdot \vec{x} + \mathbf{D} \cdot \vec{u}(t), \quad \text{Ausgangsgleichung}$$

A	$n \times n$	Systemmatrix
B	$n \times m$	Eingangsmatrix
C	$r \times n$	Ausgangsmatrix
D	$r \times m$	Durchgangsmatrix

Mit **n** = # **Zustände** (Systemordnung), **m** = # Systemeingänge und **r** = # Systemausgänge

Beispiel H.O.

$$\dot{y} = -\frac{d}{m}\dot{y} - \frac{c}{m}y(t) + \frac{1}{m}f(t), \quad \text{DGL von H.O.}$$

Definiere Zustände:

$$x_1 = y(t), \quad x_2 = \dot{y} \rightarrow \dot{x}_1 = \dot{y} = x_2, \quad \dot{x}_2 = \ddot{y} = x_3$$

$$\dot{x}_1 = 1 \cdot x_2$$

$$\dot{x}_2 = \frac{d}{m}x_2 - \frac{c}{m}x_1 + \frac{1}{m}f(t)$$

Vektoriell:

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad \text{Der Zustandsvektor}$$

$$\dot{\vec{x}} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{c}{m} & \frac{d}{m} \end{pmatrix} \cdot \vec{x} + \begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix} \cdot f(t),$$

$$\begin{aligned} \text{Zustandsgleichung} \\ &= \mathbf{A} \cdot \vec{x} + \mathbf{B} \cdot f(t), \quad f(t) = u(t), \text{ in diesem Fall} \\ \mathbf{y}(t) &= (1 \ 0) \cdot \vec{x} + 0 \cdot f(t) \end{aligned}$$

DGL zu ZSR-Darstellung

$$1 \cdot y^{(n)} + a_{n-1} \cdot y^{(n-1)} + \dots + a_1 \cdot \dot{y} + a_0 \cdot y = b_n \cdot u^{(n)} + b_{n-1} \cdot u^{(n-1)} + \dots + b_1 \cdot \dot{u} + b_0 \cdot u(t)$$

$$\mathbf{A}_R = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \dots & 1 \\ -a_0 & -a_1 & \dots & \dots & -a_{n-1} \end{pmatrix} \quad \mathbf{B}_R = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}$$

$$\mathbf{C}_R = (b_0 - b_n a_0 \quad b_1 - b_n a_1 \quad \dots \quad b_{n-1} - b_n a_{n-1}) \quad \mathbf{D}_R = b_n$$

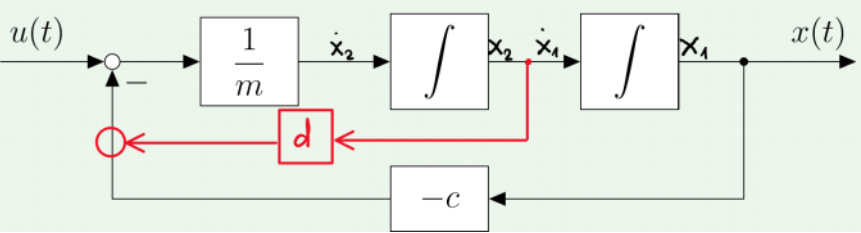
Regelungsnormalform

Übertragungsfunktion aus ZSR-Darstellung

Aus Regelungsnormalform lässt sich G(s) direkt hinschreiben. ↑
Sonst:

$$G(s) = \mathbf{C} \cdot (s \cdot \mathbb{I}_n - \mathbf{A})^{-1} \cdot \mathbf{B} + \mathbf{D}$$

ZSR-Darstellung aus Blockschaltbild



$$\dot{x}_1 = x_2, \quad \dot{x}_2 = \frac{1}{m} \cdot (u(t) - (-c) \cdot x_1 - d \cdot x_2)$$

$$\dot{\vec{x}} = \begin{pmatrix} 0 & 1 \\ \frac{c}{m} & -\frac{d}{m} \end{pmatrix} \cdot \vec{x} + \begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix} u(t), \quad y(t) = (1 \ 0) \vec{x}$$

Polstellen eines LTI

Die Polstellen sind gegeben durch die Eigenwerte der Systemmatrix A

Veranschaulicht durch das **Charakteristische Polynom**:

$$P(\lambda) = \det(\lambda \cdot \mathbb{I}_n - \mathbf{A}) \triangleq 0$$

Bsp. H.O mit viskoser Dämpfung

$$P(\lambda) = \det \begin{pmatrix} \lambda - 0 & 0 - 1 \\ 0 - \frac{c}{m} & \lambda - \left(-\frac{d}{m}\right) \end{pmatrix} = \lambda^2 + \frac{d}{m} \cdot \lambda - \frac{c}{m} \triangleq 0, \quad \lambda_{1,2} = \dots$$

Ähnlichkeitstransformation

Neuer Zustandsvektor: $\tilde{x} = \mathbf{T} \cdot x$, mit T invertierbar

$$\tilde{\dot{x}} = \mathbf{T} \cdot \dot{x} \rightarrow \dot{\tilde{x}} = \mathbf{T}^{-1} \tilde{\dot{x}}$$

Die Zustandsgleichung für das neue $\tilde{x}$ lautet somit:

$$\tilde{\dot{x}} = \mathbf{T} \cdot \mathbf{A} \cdot \mathbf{T}^{-1} \cdot \tilde{x} + \mathbf{T} \cdot \mathbf{B} \cdot u, \quad y = \mathbf{C} \cdot \mathbf{T}^{-1} \cdot \tilde{x} + \mathbf{D} \cdot u$$

Somit ist $\tilde{\mathbf{A}} = \mathbf{T} \cdot \mathbf{A} \cdot \mathbf{T}^{-1}$, $\tilde{\mathbf{B}} = \mathbf{T} \cdot \mathbf{B}$, $\tilde{\mathbf{C}} = \mathbf{C} \cdot \mathbf{T}^{-1}$, $\tilde{\mathbf{D}} = \mathbf{D}$

Spezialfall: T ist skalar

$$\text{Dann ist } \tilde{\mathbf{A}} = \mathbf{A}, \quad \tilde{\mathbf{B}} = \mathbf{T} \cdot \mathbf{B}, \quad \tilde{\mathbf{C}} = \frac{1}{\mathbf{T}} \cdot \mathbf{C}$$

Bsp. H.O. mit T = m:

$$\tilde{\mathbf{A}} = \begin{pmatrix} 0 & 1 \\ -\frac{c}{m} & -\frac{d}{m} \end{pmatrix}, \quad \tilde{\mathbf{B}} = m \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

$$\tilde{\mathbf{C}} = \frac{1}{m} \cdot (1 \ 0) = \begin{pmatrix} \frac{1}{m} & 0 \end{pmatrix}$$

Das entspricht der **Regelungsnormalform**. Die Ähnlichkeitstransformation kann also zur Umwandlung genutzt werden.

Normalformen

Kanonische Normalform

Wenn alle Polstellen des Systems (=Eigenwerte der **A** Matrix) reell sind, so sind auch die Eigenvektoren reell und diese können spaltenweise zu einer Matrix zusammengefasst werden:

$$\mathbf{V} = (v_1, v_2, \dots, v_n) = \mathbf{T}^{-1}$$

In der Darstellungsart ist die Matrix nun gerade die inverse Transformationsmatrix und macht die neue **A** Matrix diagonal.

$$\tilde{\mathbf{A}} = \mathbf{V}^{-1} \mathbf{A} \mathbf{V} = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \lambda_n \end{pmatrix}$$

Matlab: canon

Betrachtet man dazu wieder das System der *n* Differentialgleichungen, so sind diese gerade entkoppelt!
Hinweis: Neben der Bedingung der reellen Eigenwerte, müssen die Eigenvektoren zudem linear unabhängig sein!

Handwritten notes: $\tilde{\dot{x}}_1 = \lambda_1 x_1 + b_1 u(t)$, $\tilde{\dot{x}}_2 = \lambda_2 x_2 + \dots$, C_i Matrix vollbasetzt

$$\mathbf{A}_R = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \dots & 1 \\ -a_0 & -a_1 & \dots & \dots & -a_{n-1} \end{pmatrix} \quad \mathbf{B}_R = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}$$

$$\mathbf{C}_R = (b_0 - b_n a_0 \quad b_1 - b_n a_1 \quad \dots \quad b_{n-1} - b_n a_{n-1}) \quad \mathbf{D}_R = b_n$$

Regelungsnormalform (RNF)

$$\mathbf{A}_B = \begin{pmatrix} 0 & \dots & \dots & 0 & -a_0 \\ 1 & 0 & \dots & 0 & -a_1 \\ 0 & 1 & 0 & \dots & -a_2 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & -a_{n-1} \end{pmatrix} \quad \mathbf{B}_B = \begin{pmatrix} b_0 - b_n a_0 \\ b_1 - b_n a_1 \\ \vdots \\ b_{n-1} - b_n a_{n-1} \end{pmatrix}$$

$$\mathbf{C}_B = (0 \quad \dots \quad 0 \ 1) \quad \mathbf{D}_B = b_n$$

Beobachtungsnormalform (BBNF)

Umrechnung BBNF zu RNF

$$\mathbf{A}_B = \mathbf{A}_R^T, \quad \mathbf{B}_B = \mathbf{C}_R^T, \quad \mathbf{C}_B = \mathbf{B}_R^T$$

Steuerbarkeit

Kann ich durch Eingaben jeden Punkt im Zustandsraum ansteuern/erreichen?

Mathematisch: Wenn für jeden Anfangszustand x(t₀) eine Eingabefunktion u(t) existiert, die das System in endlicher Zeit zu einem beliebigen Endzustand x(t_{end}) führt.

Steuerbarkeitsmatrix

Um zu überprüfen ob das System steuerbar ist wird die Steuerbarkeitsmatrix aufgestellt und auf **vollen Rang** geprüft.

$$\mathbf{Q}_s = (\mathbf{B} \quad \mathbf{A}\mathbf{B} \quad \mathbf{A}^2\mathbf{B} \quad \dots \quad \mathbf{A}^{(n-1)}\mathbf{B}), \quad \text{rang}(\mathbf{Q}_s) = n$$

Bei quadratischen Matrizen ($\mathbf{B} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, einen Eingang) $\Rightarrow \det(\mathbf{Q}_s) \neq 0$

Bsp. DC-Motor

$$\mathbf{A} = \begin{pmatrix} 0 & k_{m/J} \\ -k_{m/L} & -R/L \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 \\ 1/L \end{pmatrix}, \quad \mathbf{C} = (1 \ 0)$$

Mit x₁ = Drehzahl und x₂ = Strom

$$\mathbf{Q}_s = (\mathbf{B} \quad \mathbf{A}\mathbf{B}) = \begin{pmatrix} 0 & km/JL \\ 1/L & -R/L^2 \end{pmatrix}, \quad \det(\mathbf{Q}_s) = -\frac{km}{L^2 J^2} \neq 0$$

Der **Rang** ist **voll** --> Das System **ist steuerbar**

Beobachtbarkeit

Kann ich am Ausgang messen was im Zustandsraum passiert?

Mathematisch: Wenn der Ausgangszustand x(t₀) aus den bekannten Größen y(t) und u(t) im Intervall t₀ < t < t_{end} bestimmt werden kann.

Beobachtbarkeitsmatrix

Um zu überprüfen ob das System beobachtbar ist wird die Beobachtbarkeitsmatrix aufgestellt und auf **vollen Rang** geprüft.

$$\mathbf{Q}_B = \begin{pmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A} \\ \vdots \\ \mathbf{C}\mathbf{A}^{(n-1)} \end{pmatrix}, \quad \text{rang}(\mathbf{Q}_B) = n$$

Bei quadratischen Matrizen ($\mathbf{C} = (1 \ 0)$, einen Ausgang) $\Rightarrow \det(\mathbf{Q}_B) \neq 0$

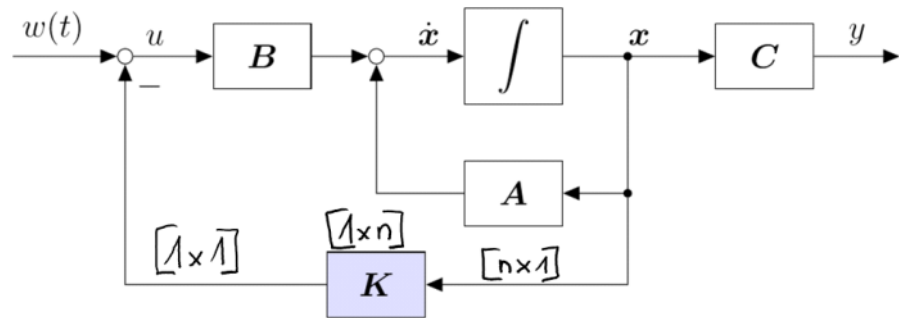
Bsp. DC-Motor

$$\mathbf{Q}_B = \begin{pmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -km/L \end{pmatrix}, \quad \text{diagonal} \rightarrow \text{voller Rang}$$

Der **Rang** ist **voll** --> Das System **ist beobachtbar**

Zustandsregelung (ZSR)

Bei der Zustandsregelung wird die Systemdynamik in dem Zustandsvektor abgebildet und als Ganzes für die Regelung verwendet.



Die Zustandsgleichung lautet nun:

$\dot{x} = (A - B \cdot K) \cdot \tilde{x} + B \cdot w(t) = \tilde{A} \cdot \tilde{x} + B \cdot w(t)$

($\tilde{A}$ ist nicht die gleiche wie von der Transformation)

$K = (k_1 \quad k_2 \quad \dots \quad k_n)$

$\Rightarrow \tilde{A} = (A - B \cdot K) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 - k_1 & \dots & -a_{n-1} - k_n \end{pmatrix}$

$\Rightarrow P(\lambda) = \lambda^n + (-a_{n-1} - k_n)\lambda^{n-1} + \dots + (-a_1 - k_2)\lambda + (-a_0 - k_1)$

In jedem Summand kommt genau ein Koeffizient aus **K** vor. --> Das char. Polynom kann beliebig gestaltet werden.

--> **Die Eigenwerte (Polstellen) könne frei gewählt werden.**

Berechnung von K aus RNF

- 1. Erstelle die RNF des Systems
- 2. Wähle n Polstellen des *geschlossenen* Regelkreises, $\lambda_1 - \lambda_n$, und formuliere das Wunschpolynom

$P_{wunsch}(\lambda) = (\lambda - \lambda_1) \cdot \dots \cdot (\lambda - \lambda_n)$
 $= \lambda^n + p_{n-1} \cdot \lambda^{(n-1)} + \dots + p_1 \lambda + p_0$

- 3. Mache einen Koeffizientenvergleich

$p_0 = a_0 + k_1 \Rightarrow k_1 = p_0 - a_0$

$p_1 = a_1 + k_2 \Rightarrow k_2 = p_1 - a_1$

$\vdots$

$p_{n-1} = a_{n-1} + k_n \Rightarrow k_n = p_{n-1} - a_{n-1}$

Bsp. H.O.

Mit m = 1, c = 100, d = 1, Mit cl-Polstellen bei -50 und aperiodisch

$A = \begin{pmatrix} 0 & 1 \\ -\frac{c}{m} & -\frac{d}{m} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -100 & -1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -a_0 & -a_1 \end{pmatrix},$

$B = \begin{pmatrix} 0 \\ 1/m \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad C = (1 \quad 0)$

$P(\lambda) = \lambda^2 + (1 + k_2)\lambda + (100 + k_1)$

$P_{wunsch}(\lambda) = (\lambda + 50)^2 = \lambda^2 + 100\lambda + 2500$

$k_1 = 2500 - 100 = 2400$

$k_2 = 100 - 1 = 99$

$K = (2400 \quad 99)$

Berechnung von K allgemein

- Berechne $\tilde{A} = A - BK$
 - Mit $K = (k_1, k_2, \dots, k_n)$
- Berechne charakteristisches Polynom
 - $P(\lambda) = |\lambda \cdot \mathbb{I}_n - \tilde{A}|$
- Schreibe $P_{wunsch}(\lambda)$ auf
- Koeffizientenvergleich

Bsp. System 2.Ordnung

$A = \begin{pmatrix} -100 & 100 \\ -100 & 99 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad C = (1 \quad -1)$

• $\tilde{A} = A - BK = \begin{pmatrix} -100 - k_1 & 100 - k_2 \\ -100 - k_1 & 99 - k_2 \end{pmatrix}$

• $P(\lambda) = \left| \begin{pmatrix} \lambda + 100 + k_1 & -100 + k_2 \\ 100 + k_1 & \lambda - 99 + k_2 \end{pmatrix} \right|$
 $= \lambda^2 + (k_1 + k_2 + 1)\lambda + k_1 + 100$

• $P_w(\lambda) = \lambda^2 + 100\lambda + 2500$

• $k_1 = 2400, \quad k_2 = -2301$

Statische Verstärkung

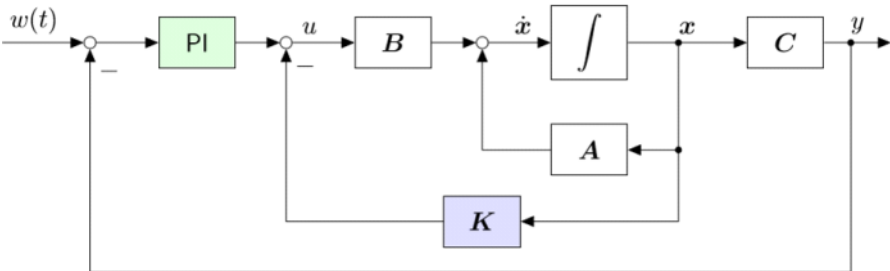
$dc = y_{statisch} = -C \cdot \tilde{A}^{-1} \cdot B + D = C \cdot (BK - A)^{-1} \cdot B + D$

Vorsteuerung / statischer Vorfilter

$V = 1 / dc$

Regler mit I-Anteil

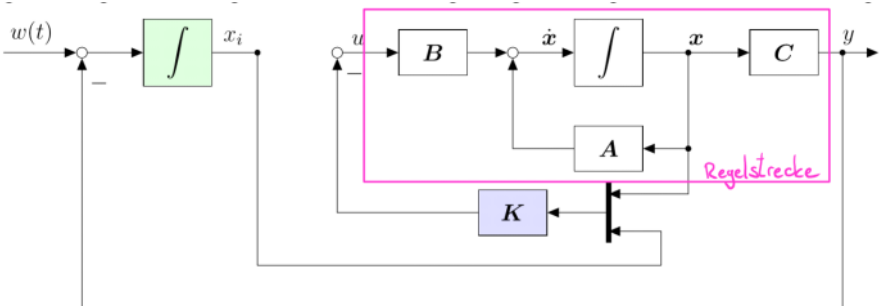
Überlagerter I / PI-Regler



Besser als Vorfilter als statischer Vorfilter

Auslegung wie gehabt (Kompensation, WOK,...)

I-Anteil als zusätzlicher Zustand



$\bar{x} = \begin{pmatrix} x \\ x_i \end{pmatrix}, \quad \bar{A} = \begin{pmatrix} A & 0_n \\ -C & 0 \end{pmatrix}, \quad \bar{B} = \begin{pmatrix} B \\ 0 \end{pmatrix}, \quad \bar{C} = (C \quad 0)$

Bsp. System 2.Ordnung (siehe oben)

$\bar{A} = \begin{pmatrix} -100 & 100 & 0 \\ -100 & 99 & 0 \\ -1 & 1 & 0 \end{pmatrix}, \quad \bar{B} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \quad \bar{C} = (1 \quad -1 \quad 0)$

Mit Gütekriterium

Das Gütekriterium erweitert den stabilen Regler zu einem optimalen Regler.

Dafür wird meist ein quadratisches Gütekriterium eingefügt.

$J = \int_0^\infty (x^T Q x + u^T R u) dt$

Es wird die bestmögliche Rückführung $u(t) = -Kx(t)$, d.h. möglichst kleines J gesucht.

Dies kann analytisch gemacht werden --> MATLAB

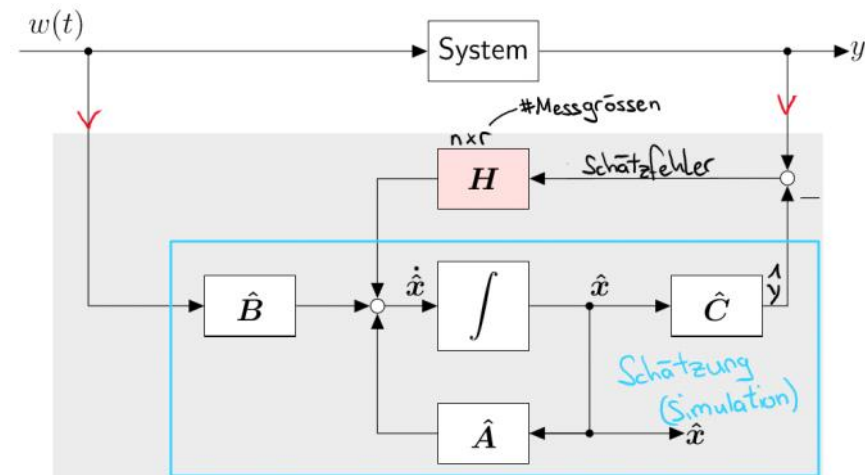
`[K_lqr, P_lqr, eigVals] = lqr(A, B, Q, R);`

Oder `[P, eigVals, K] = care(A, B, Q, R);`

Zustandsbeobachter (ZSB)

Bei der Zustandsregelung wurde angenommen, dass alle Zustände als Messgrößen vorhanden sind. --> Ist häufig **nicht** der Fall.

Luenberger Beobachter



Ansatz: Schätzen der fehlenden Systemeingänge anhand den vorhandenen Messungen y und dem Systemmodell.

$$\dot{\hat{x}} = H(y - \hat{C} \cdot \hat{x}) + \hat{A} \cdot \hat{x} + \hat{B} \cdot w(t) = (\hat{A} - H\hat{C}) \cdot \hat{x} + \hat{B} \cdot w(t)$$

Hier bei nimmt man an dass $\hat{A} = A, \hat{B} = B, \hat{C} = C$.

Daraus ergibt sich, dass der **Beobachtungsfehler** $e = x - \hat{x}$, gegen **Null** geht wenn die **Eigenwerte** der Matrix $F = A - HC$ stabil sind (**negativen Realteil** haben).

Berechnung der H-Matrix

Die H-Matrix berechnet sich **analog** zur K-Matrix, **aber mit der BBNF**

- $\tilde{A} = A - HC = \begin{pmatrix} 0 & 0 & -a_0 \\ 1 & 0 & \vdots \\ 0 & 1 & -a_{n-1} \end{pmatrix} - \begin{pmatrix} h_1 \\ \vdots \\ h_n \end{pmatrix} \begin{pmatrix} 0 & \dots & 1 \end{pmatrix}$
- $= \begin{pmatrix} 0 & 0 & -a_0 - h_1 \\ 1 & 0 & \vdots \\ 0 & 1 & -a_{n-1} - h_n \end{pmatrix}$
- $P(\lambda) = |(\lambda \cdot \mathbb{I}_n - \tilde{A})|$
- $P_{\text{wünsch}}(\lambda)$ aufstellen
- Koeffizienten Vergleich
 - $h_1 = p_0 - a_0, \dots$

Matlab:

$H = \text{place}(A', B', P)'$

Vergleich zu ZSR

$$A = A^T, \quad B = C^T, \quad H = K^T$$

Separationstheorem

Wenn die Regelstrecke steuerbar und beobachtbar ist, so können die Pole für die ZSR und die für den ZSB unabhängig voneinander gewählt werden. Die Pole des Gesamtsystems setzen sich aus diesen beiden zusammen.

Zeitdiskrete Systeme

Zeitkontinuierliches System (Differentialgleichung):

$$a_n \cdot y^{(n)} + a_{n-1} \cdot y^{(n-1)} + \dots + a_0$$
$$= b_q \cdot u^{(q)} + b_{q-1} \cdot u^{(q-1)} + \dots + b_0$$

Zeitdiskretes System (Differenzengleichung):

$$a_n \cdot y_{k+n} + a_{n-1} \cdot y_{k+n-1} + \dots + a_0 y_k$$
$$= b_q \cdot u_{k+q} + b_{q-1} \cdot u_{k+q-1} + \dots + b_0 \cdot u_k$$

Bzw. in Signalverarbeitungsschreibweise

$$a_n \cdot y_k + a_{n-1} \cdot y_{k-1} + \dots + a_0 y_{k-n}$$
$$= b_q \cdot u_k + b_{q-1} \cdot u_{k-1} + \dots + b_0 \cdot u_{k-n}$$

RC-Glied

$\tau \cdot \dot{y} = -y(t) + u(t),$ $\tau = R \cdot C$	$y_k = -\frac{\tau}{\tau + \Delta t} \cdot y_{k-1} + \frac{\Delta t}{\tau + \Delta t} \cdot u_k$
---	--

Zustandsraummodell

$$x_{k+1} = A_d \cdot \vec{x}_k + B_d \cdot u_k$$
$$y_k = C_d \cdot \vec{x}_k + D_d \cdot u_k$$

Gewichtsfolge (Impulsantwort)

Die Gewichtsantwort g_k ist die Antwort eines Systems auf einen Impuls der Form:

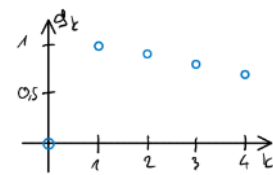
$$u_k = \begin{cases} 1 & \text{für } k = 0 \\ 0 & \text{für } k > 0 \end{cases}$$



Beispiel: $y_k = 0,9 \cdot y_{k-1} + u_{k-1}$

$y_1 = 0,9 \cdot 0 + 1 = 0,9$
 $y_2 = 0,9 \cdot 0,9 + 0 = 0,81$
 $y_3 = 0,9 \cdot 0,81 + 0 = 0,729$

$y_k = \begin{cases} 0, & k = 0 \\ 0,9^{k-1}, & k > 0 \end{cases}$



z-Transformation

Definition:

$$F(z) = Z\{f_k\} = \sum_{k=0}^{\infty} f_k z^{-k}$$

Geom. Reihe: $\sum_{k=0}^{\infty} q^k = \frac{1}{1-q}$	$\{1, 0.5, 0.25, \dots\} = 0.5^k$ $Z\{0.5^k\} = \sum_{k=0}^{\infty} 0.5^k \cdot z^{-k} = \frac{1}{1 - \frac{0.5}{z}}$ $= \frac{z}{z - 0.5}$
---	---

z-Übertragungsfunktion

$$G(z) = \frac{Y(z)}{U(z)} = \frac{Z\{y_k\}}{Z\{u_k\}}$$

Aus Differenzengleichung:

$$a_n \cdot y(z) \cdot z^n + a_{n-1} \cdot y(z) \cdot z^{n-1} + \dots + a_0 \cdot y(z)$$
$$= b_q \cdot u(z) \cdot z^q + b_{q-1} \cdot u(z) \cdot z^{q-1} + \dots + b_0 \cdot u(z)$$

$$G(z) = \frac{Y(z)}{U(z)} = \frac{b_n z^n + b_{n-1} z^{n-1} + \dots + b_1 z^1 + b_0}{a_n z^n + a_{n-1} z^{n-1} + \dots + a_1 z^1 + a_0}$$

Aus zeitdiskretem Zustandsmodell

$$G(z) = C \cdot (z \cdot I - A)^{-1} \cdot B + D$$

Stabilität, Polstellen

Stabil wenn alle Polstellen $|p_i| < 1$, $\forall i$, also innerhalb des Einheitskreis.

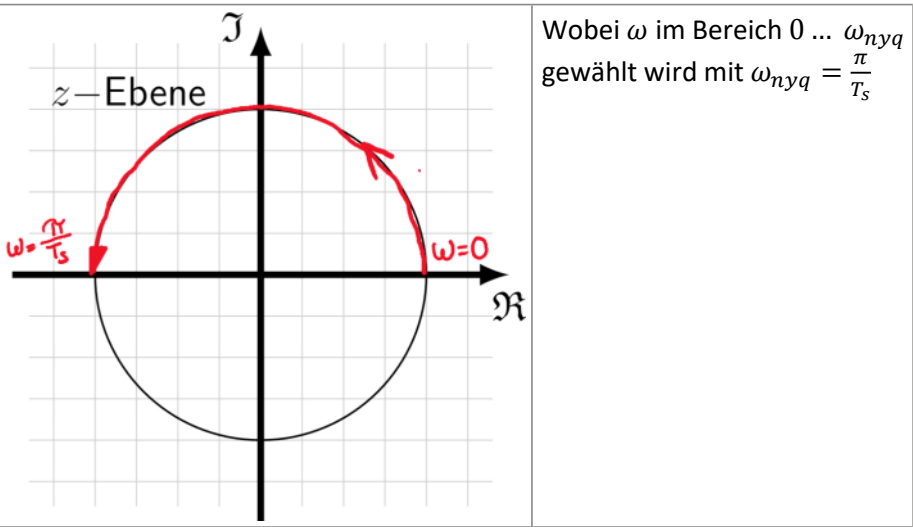
Grenzstabil wenn auf dem Einheitskreis

Die Eigenwerte von A_d sind die Polstellen

Nullstellen berechnen sich aus dem Zählerpolynom wie gehabt

Visualisierung in Bodeplot

$$G(z) \rightarrow G(\omega), \quad \text{mit } z = e^{j\omega T_s}$$



ZOH - Transformation

$$G(z) = \frac{z-1}{z} \cdot Z \left\{ \mathcal{L}^{-1} \left\{ \frac{G(s)}{s} \right\} \right\}$$

Bsp. Integrator ($G(s) = \frac{1}{s}$)

$$\mathcal{L}^{-1} \left\{ \frac{1}{s} \right\} = t \rightarrow k \cdot T_s$$
$$G(z) = \frac{z-1}{z} \cdot Z\{k \cdot T_s\} = \frac{z-1}{z} \cdot T_s \frac{z}{(z-1)^2} = \frac{T_s}{z-1}$$

Differenzengleichung:

$$y_{k+1} - 1 \cdot y_k = T_s \cdot u_k \Rightarrow y_{k+1} = y_k + T_s \cdot u_k$$

Bilineare Transformation (Tustin)

$$s = \frac{2}{T_s} \cdot \frac{z-1}{z+1}$$

Bsp. Integrator

$$G(z) = \frac{T_s}{2} \frac{z+1}{z-1}$$

$$y_{k+1} = y_k + \frac{T_s}{2} (u_{k+1} + u_k)$$

Euler-Vorwärts-Transformation

$$s = \frac{z-1}{T_s}$$

Bsp. Integrator

$$G(z) = \frac{T_s}{z-1}$$

$$y_{k+1} = y_k + T_s \cdot u_k$$

Euler-Rückwärts Transformation

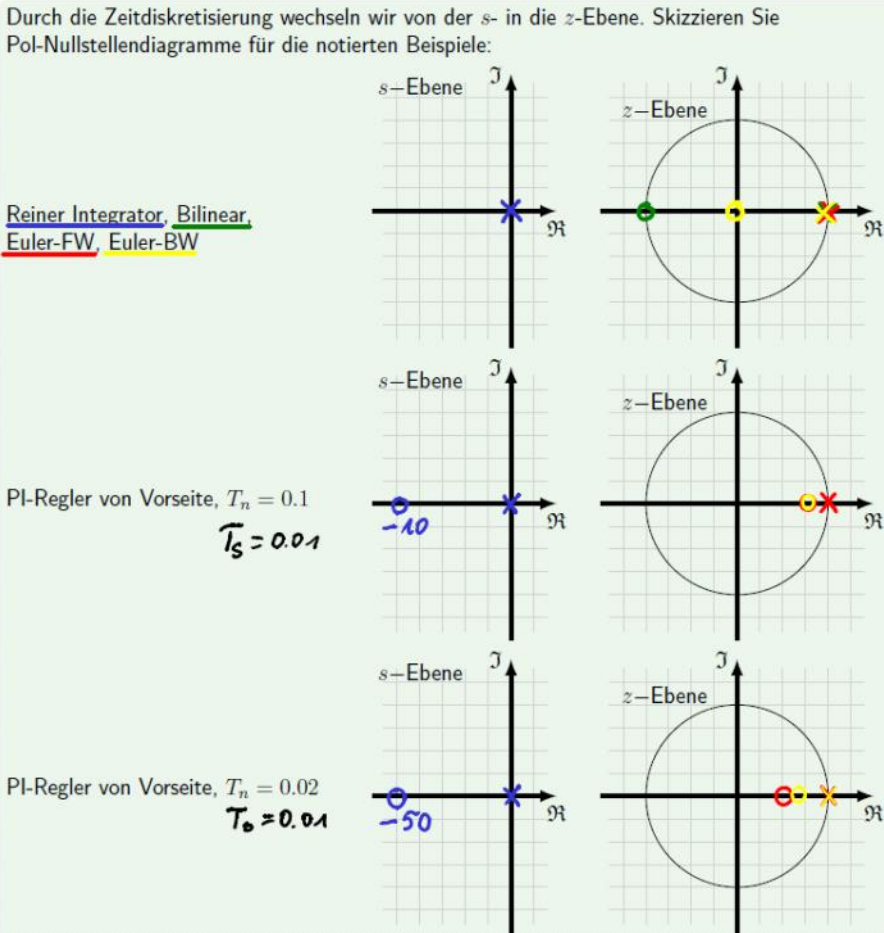
$$s = \frac{1}{T_s} \frac{z-1}{z}$$

Bsp. Integrator

$$G(z) = \frac{T_s \cdot z}{z-1}$$

$$y_{k+1} = y_k + T_s \cdot u_{k+1}$$

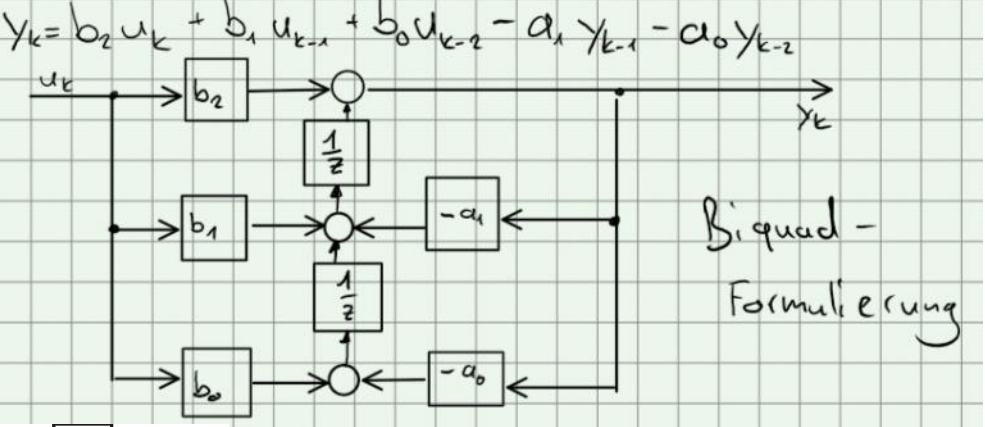
s-Ebene und z-Ebene



Blockschaltbilder aus Differenzengleichung

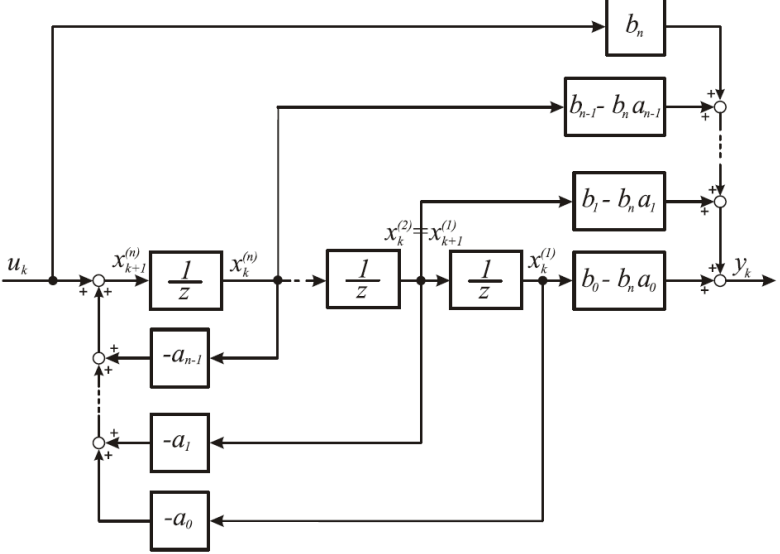
Biquad Formulierung

Nur 2. Ordnung

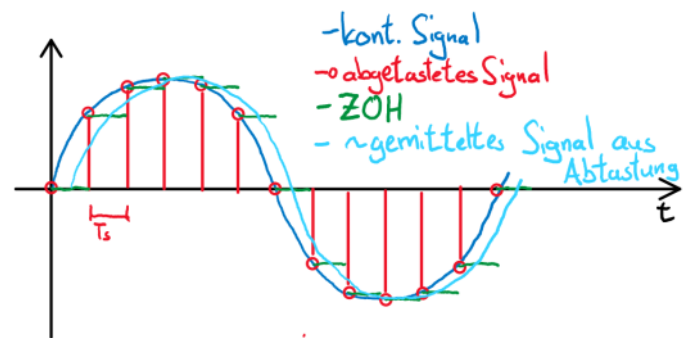


Allgemeine Formulierung

n-ter Ordnung



Abtastregelung Approximation der ZOH-Transformation



Die ZOH-Transformation kann im zeitkontinuierlichen durch ein nachgeschaltetes **Totzeitsystem** mit der **Totzeit** $T_t = T_s / 2$ **approximiert** werden. ($e^{-j\omega \frac{T_s}{2}}$)

D.h. das Abtasten einer Regelstrecke führt zu einer Zeitverzögerung von $T_s / 2$ bzw. einem Phasenverlust von $\varphi_{Verl.}(\omega) = -\omega \cdot \frac{T_s}{2}$

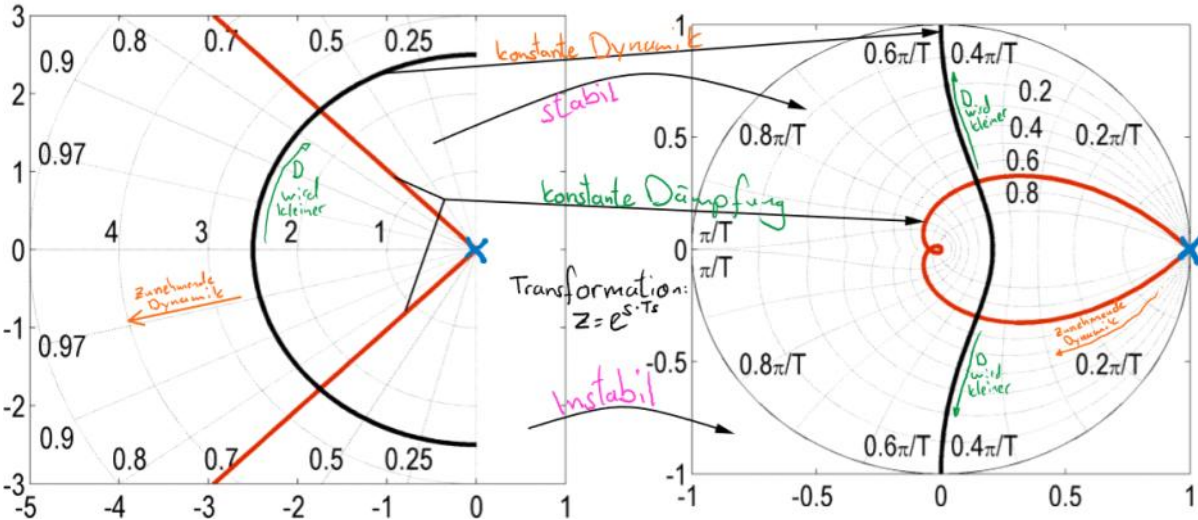
Wahl der Abtastzeit

Ein gewisser Phasenverlust ist tolerierbar. (5-7°)

$$\Rightarrow \varphi_{Verl.}(\omega_B) = 5 \dots 7 \cdot \pi / 180 = \frac{\omega_B \cdot T_s}{2}$$

$$\Rightarrow T_s = \frac{1}{4 \dots 6 \omega_B}, \quad \omega_B \text{ in } [\frac{rad}{s}]$$

s-Ebene und z-Ebene



Transformation: $z = e^{s \cdot T_s}$

Zeitkontinuierlich:

Zeitdiskret:

Integrator:

Pol bei $s = 0$

Pol bei $z = 1$

Trennung stabil/instabil:

Imaginäre Achse

Einheitskreis

Werte konstanter Dämpfung:

Ursprungsgerade

Herzförmige Linien

Ansteigende Dynamik:

Konzentrische Kreise vom Ursprung

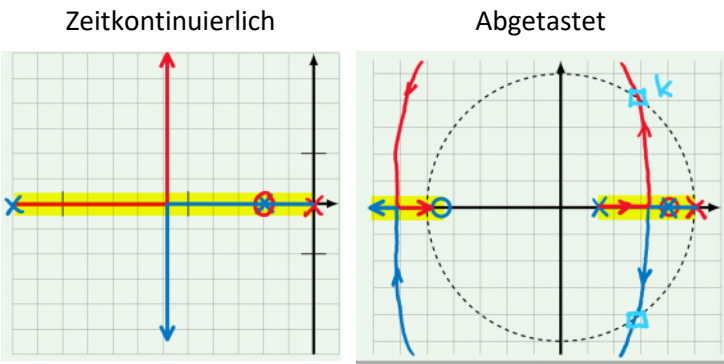
Wellenförmig von $z = 1$ aus

WOK

Die WOK ist genau gleich wie bei zeitkontinuierlichen Systemen. Dies ist möglich, da sie auf der Rückkopplung gebrochen-rationalen Funktionen basiert.

ABER, Abtastsysteme werden bei Rückkoppelung bei **Erhöhung** der Reglerverstärkung **k stets** irgendwann **instabil**.

Bsp. RCRC



Linearisierung

DGL n-ter Ordnung bestehen aus n DGL 1. Ordnung

$\dot{x}_1 = f_1(\vec{x}, \vec{u}, t)$
 $\dot{x}_2 = f_2(\vec{x}, \vec{u}, t),$
 $\vdots$
 $\dot{x}_n = f_n(\vec{x}, \vec{u}, t)$
mit $\vec{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$

Wobei die f_i im allgemeinen nichtlineare, zeitabhängige Funktionen sind.

1 Arbeitspunkt bestimmen

Die Linearisierung erfolgt stets an einem Arbeitspunkt (AP) und ist statisch
Ergo alle **Zeitableitungen** sind **=0**.

$\dot{\vec{x}} = \vec{0} \rightarrow \vec{f} = \vec{0}$
Bedingungen für $\vec{x}_0, \vec{u}_0$. Berechne alle $x_{i,0}$ und $u_{i,0}$ aus den gegebenen

2 Δ-Größen bestimmen

$\vec{x} = \vec{x}_0 + \Delta \vec{x}, \quad \vec{u} = u_0 + \Delta \vec{u}$
 $\Rightarrow 0 \dot{\vec{x}} = \Delta \vec{f}, \quad \dot{\vec{u}} = \Delta \vec{u}$

3 Lineare Approximation, Ableitungen bilden, Jacobi-Matrix

Bei der Linearisierung wird die nichtlineare Funktion um einen Arbeitspunkt (AP) mittels Taylorreihe angenähert – dabei bricht man nach dem linearen Term ab. Es gibt n partielle Ableitungen pro f_i nach den Zuständen, sowie q Ableitungen nach den Eingängen (In RT2 q = 1).

$\frac{d\vec{f}}{d\vec{x}} = \tilde{A} = \begin{pmatrix} \frac{df_1}{dx_1} & \frac{df_1}{dx_2} & \dots & \frac{df_1}{dx_n} \\ \frac{df_2}{dx_1} & \frac{df_2}{dx_2} & \dots & \frac{df_2}{dx_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{df_n}{dx_1} & \frac{df_n}{dx_2} & \dots & \frac{df_n}{dx_n} \end{pmatrix}_{\substack{\vec{x}=\vec{x}_0 \\ \vec{u}=\vec{u}_0}}, \quad \frac{d\vec{f}}{d\vec{u}} = \tilde{B} = \begin{pmatrix} \frac{df_1}{du_1} & \frac{df_1}{du_2} & \dots & \frac{df_1}{du_q} \\ \frac{df_2}{du_1} & \frac{df_2}{du_2} & \dots & \frac{df_2}{du_q} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{df_q}{du_1} & \frac{df_q}{du_2} & \dots & \frac{df_q}{du_q} \end{pmatrix}_{\substack{\vec{x}=\vec{x}_0 \\ \vec{u}=\vec{u}_0}}$

Ergibt die $\tilde{A}$ und $\tilde{B}$ Matrix der Zustandsraumdarstellung.

4 Linearisierte DGL aufstellen

Einsetzen:
 $\Delta \dot{\vec{x}} = \tilde{A} \cdot \Delta \vec{x} + \tilde{B} \cdot \Delta \vec{u}$

$\vec{y} = C \cdot \Delta \vec{x} + D \cdot \Delta \vec{u}$
C und D bestimmen wie in normalem Zustandsraummodell

Bsp. Schwebend Kugel

DGL und Zustandsvariablendarstellung

1. $m \cdot \ddot{x} = mg - \frac{10^{-6} \cdot \dot{x}^2}{(x+0.001)^2}$
 $\ddot{x} = g - \frac{10^{-6} \cdot \dot{x}^2}{m(x+0.001)^2}$
ZVD: $x_1 = x$
 $x_2 = \dot{x} = \dot{x}_1 \rightarrow \dot{x}_2 = \ddot{x}$
 $\dot{x}_1 = x_2 = f_1$
 $\dot{x}_2 = g - \frac{10^{-6} \cdot \dot{x}^2}{m(x+0.001)^2} = f_2$

1 Arbeitspunkt bestimmen

$\dot{x}_1 = 0 \quad x_2 = 0$
 $\dot{x}_2 = 0 \Leftrightarrow g = \frac{10^{-6} \cdot \dot{x}^2}{m(x+0.001)^2} \Leftrightarrow 10 \cdot 0.05 \cdot (x_1+0.001)^2 = 10^{-6} \cdot \dot{x}^2$
 $x_{1,0} = 0.003 \rightarrow 0.5 \cdot 4^2 \cdot 10^{-6} = 10^{-6} \cdot \dot{x}^2 \rightarrow \dot{x}_0 = \pm \sqrt{8} = \pm 2\sqrt{2}$

2 Δ-Größen bestimmen

$\vec{x} = \vec{x}_0 + \Delta \vec{x} \quad \dot{x} = \dot{x}_0 + \Delta \dot{x}$

3 Lineare Approximation, Ableitungen bilden, Jacobi-Matrix

$\tilde{J} = \tilde{A} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 10^{-6} \cdot \dot{x}^2 \cdot \frac{2}{m(x+0.001)^3} \cdot 2 & 0 \end{pmatrix}_{\substack{\vec{x}=\vec{x}_0 \\ \dot{x}=\dot{x}_0}}$
 $= \begin{pmatrix} 0 & 1 \\ 10^{-6} \cdot 8 \cdot 20 \cdot 4^2 \cdot 10^{-9} \cdot 2 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 5000 & 0 \end{pmatrix}$
 $\tilde{B} = \begin{pmatrix} \frac{\partial f_1}{\partial u} \\ \frac{\partial f_2}{\partial u} \end{pmatrix} = \begin{pmatrix} 0 \\ -\frac{10^{-6} \cdot 2 \cdot \dot{x}}{m(x+0.001)^2} \end{pmatrix}_{\substack{x=x_0 \\ \dot{x}=\dot{x}_0}} = \begin{pmatrix} 0 \\ -\frac{20 \cdot 10^{-6} \cdot 2 \cdot 2\sqrt{2}}{4 \cdot 10^{-6}} \end{pmatrix} = \begin{pmatrix} 0 \\ -5\sqrt{2} \end{pmatrix}$

4 Linearisierte DGL aufstellen

$\Delta \dot{\vec{x}} = \tilde{A} \Delta \vec{x} + \tilde{B} \Delta \dot{x}$
Messung der Position (x) $\rightarrow C = \begin{pmatrix} 1 & 0 \end{pmatrix} \quad y = C \Delta x$