Benoteter Programmentwurf – Simulationstechnik 2023

Allgemeine Hinweise:

- *Die Bearbeitung darf alleine oder zu zweit erfolgen; Sie dürfen keine Vorgehensweisen, Ergebnisse oder Simulationscode mit anderen Studierenden austauschen.*
- *Abgabetermin: spätestens Sonntag, 09.04.2023*
- *Geben Sie alle Elemente Ihrer Abgabe als 1 ZIP-Datei per Email beim Dozenten ab. Falls Sie zu zweit am Programmentwurf arbeiten, bitte nur 1 Abgabe.*
- *Bestandteile der Abgabe:*
  - *Matlab-Code und/oder Simulink-Diagramme zu jeder Aufgabe: Diese sollen beim Dozenten lauffähig sein, um die Ergebnisse reproduzieren zu können; geben Sie die Matlab-Version an, die Sie genutzt haben (z.B. R2021b, R2022a, siehe Command Window ganz oben links, oder Befehl* `ver` *eingeben).*
  - *Kurze formlose Dokumentation mit den wesentlichen Ergebnissen, Screenshots der Abbildungen, Aussagen sowie kurzen Hinweisen, wo im Code/in den Diagrammen die weiteren Ergebnisse zu finden sind, welche nicht im Dokument sind. Bitte auch Kurs Txx21 sowie Matrikelnummern der Beteiligten nennen. Diese Doku bitte als 1 PDF-Datei für beide Aufgaben zusammen erstellen. Falls PDF nicht geht, nehme ich zur Not auch WORD o.Ä.*
  - *Bezeichnen Sie Ihre Dateien mit "sprechenden" Namen, z.B.* `SIM2023_Matrikelnummer1_Matrikelnummer2_Doku.pdf` *oder* `SIM2023_Matrikelnummer1_Matrikelnummer2_Aufg1.m`*.*
- *Lesen Sie die Aufgaben vor Bearbeitung komplett durch.*

Aufgabe 1 *(55 Punkte)*

In dieser Aufgabe beschäftigen wir uns mit der Simulation eines Tempomats beim Auto, also einer Einrichtung, welche die gefahrene Fahrzeug-Geschwindigkeit regeln soll.

Hinweis: Die Aufgabe ist zur Ausführung mit Simulink + Matlab gedacht, und Teile des Simulationsmodells sind schon in Simulink vorgegeben. Sie können die ganze Aufgabe aber auch ohne Simulink lösen und einreichen, wenn Sie möchten. Der Dozent empfiehlt Simulink.

Die Geschwindigkeitsdynamik eines auf einer horizontalen Ebene fahrenden Fahrzeuges wird näherungsweise durch das folgende nichtlineare Modell beschrieben (alle Größen sind skalar, also nicht vektoriell):

$$m\,\dot{v}(t) = F_{\text{Motor}}(t) + F_{\text{Rollreibung}}(t) + F_{\text{Luftwiderstand}}(t) \tag{1}$$

$$F_{\text{Motor}}(t) = u(t) \tag{2}$$

$$F_{\text{Rollreibung}}(t) = -c_R\, m\, g\, \operatorname{sign}(v(t)) \tag{3}$$

$$F_{\text{Luftwiderstand}}(t) = -\frac{1}{2}\, c_W\, \rho\, A\, (v(t))^2\, \operatorname{sign}(v(t)) \tag{4}$$

mit

- der Zeit t
- dem Zustand Geschwindigkeit $x(t) = v(t)$ (Anfangswert $v(0) = 0$ m/s)
- dem Eingang (= Reglerstellgröße) Motorkraft $u(t) = F_{\text{Motor}}(t)$
- und den Parametern
  - Rollwiderstandsbeiwert: $c_R = 0.013$
  - Luftwiderstandsbeiwert: $c_W = 0.28$
  - Luftdichte: $\rho = 1.21$ kg/m^3
  - Referenzfläche für den Luftwiderstand: $A = 2$ m^2
  - Fahrzeugmasse: $m = 1.5$ t
  - Fallbeschleunigung: $g = 9.81$ m/s^2

Der gewünschte Sollwertverlauf $r(t)$ für die Geschwindigkeit wird während der Aufgabe erstellt, siehe unten. Die Regelung wird von einem PI-Regler mit der Gleichung

$$U(s) = K_P\left(1 + \frac{1}{T_I\, s}\right) \cdot (R(S) - V(S)) = \left(K_P + \frac{K_I}{s}\right) \cdot (R(S) - V(S))$$

im Laplace-Bereich übernommen. Regelkreisstruktur und Regler sind schon implementiert und eingestellt; sie sollen in dieser Aufgabe nicht verändert werden.

a) Warum ist das Modell der Geschwindigkeitsdynamik nichtlinear? *(1 Teilpunkt)*

b) Öffnen Sie das bereitgestellte Simulink-Diagramm `TempomatMitPI_model.slx` und das Matlab-Skript `TempomatMitPI.m`.
Hinweis: Das Simulink-Modell wurde mit Matlab R2021b erstellt, kann also mit dieser Version und neueren Matlab-Versionen geöffnet werden. Falls Sie damit nicht klarkommen, bitte Email an den Dozenten.

c) Implementieren Sie die oben gegebenen Gleichungen (1)–(4) der Geschwindigkeitsdynamik im fast leeren Block “Fahrzeug”. Der Block soll die Motorkraft u als Eingang und die Fahrzeug-Geschwindigkeit v als Ausgang besitzen. Nutzen Sie dabei die schon vorhandenen Ein-/Ausgangsports im Block sowie die im Matlab-Skript `TempomatMitPI.m` definierten Modellparameter; ergänzen Sie bei Bedarf.
Hinweis: Durch Doppelklick auf den Block kommt man hinein, mit ESC wieder heraus.

(20 Teilpunkte)

d) Implementieren Sie folgenden Sollwertverlauf im fast leeren Block “Sollwert Geschw.”.

$$r(t) = v_{\mathsf{soll}}(t) = \begin{cases} 0\ \mathrm{m/s} & \text{für} \quad 0\ \mathrm{s} \le t < 100\ \mathrm{s} \\ 40\ \mathrm{m/s} & \quad 100\ \mathrm{s} \le t < 1000\ \mathrm{s} \\ 20\ \mathrm{m/s} & \quad 1000\ \mathrm{s} \le t < 2000\ \mathrm{s} \\ 10\ \mathrm{m/s} & \quad 2000\ \mathrm{s} \le t < 2500\ \mathrm{s} \\ 40\ \mathrm{m/s} & \quad 2500\ \mathrm{s} \le t \end{cases}$$

Nutzen Sie dabei den schon vorhandenen Ausgangsport im Block. Sie können den Sollwertverlauf in Matlab definieren und in das Simulink-Diagramm überführen, oder direkt in Simulink definieren (siehe “Sources” in der Simulink-Basis-Modellbibliothek).
Hinweis: Durch Doppelklick auf den Block kommt man hinein, mit ESC wieder heraus.

(10 Teilpunkte)

e) Stellen Sie im Simulink-Modell folgende Simulationsparameter ein:

- Solver-Typ “Fixed-Step”
- Solver Ihrer Wahl
- Simulationsschrittweite 0.01 s oder kleiner
- Endzeit für die Simulation 4000 s *(4 Teilpunkte)*

f) Simulieren Sie das Verhalten des Tempomats, und stellen Sie die Zeitverläufe von Fahrzeuggeschwindigkeit und Sollgeschwindigkeit in einem Diagramm sowie die Motorkraft in einem zweiten Diagramm dar, entweder direkt in Simulink oder in Matlab. *(15 Teilpunkte)*

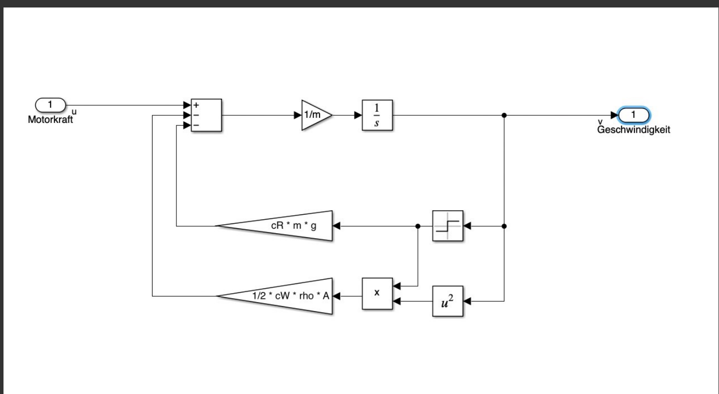
g) Das Auto wird überladen, und die Fahrzeugmasse beträgt nun $m = 2$ t. Wie ändert sich das Verhalten des Tempomats, d.h. wie sehen die in der vorigen Teilaufgabe genannten Zeitverläufe nun aus? Erklären Sie die Unterschiede kurz. *(5 Teilpunkte)*

a) Das Geschwindigkeitsdynamikmodell ist nicht linear, weil sowohl in (3) und (4) sign(v(t)) vorhanden ist. Außerdem ist v(t) in (4) quadriert.

b)

```
Command Window
>> ver
-----------------------------------------------------------------------------------------------------
MATLAB Version: 9.13.0.2126072 (R2022b) Update 3
MATLAB License Number: 886933
Operating System: macOS  Version: 13.2.1 Build: 22D68
Java Version: Java 1.8.0_202-b08 with Oracle Corporation Java HotSpot(TM) 64-Bit Server VM mixed mode
-----------------------------------------------------------------------------------------------------
MATLAB                                                Version 9.13        (R2022b)
Simulink                                              Version 10.6        (R2022b)
Control System Toolbox                                Version 10.12       (R2022b)
>>
```

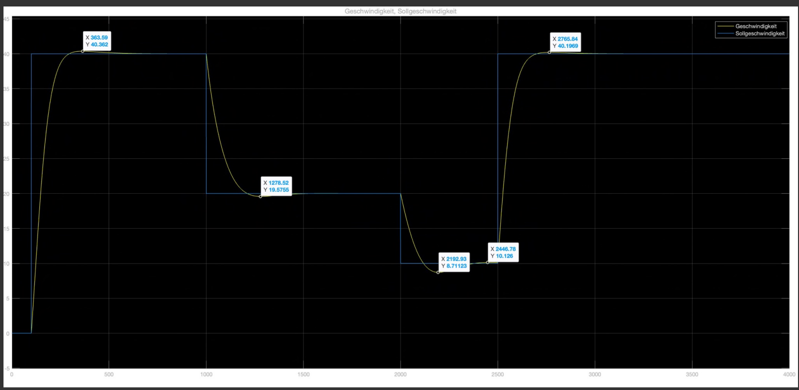
c)



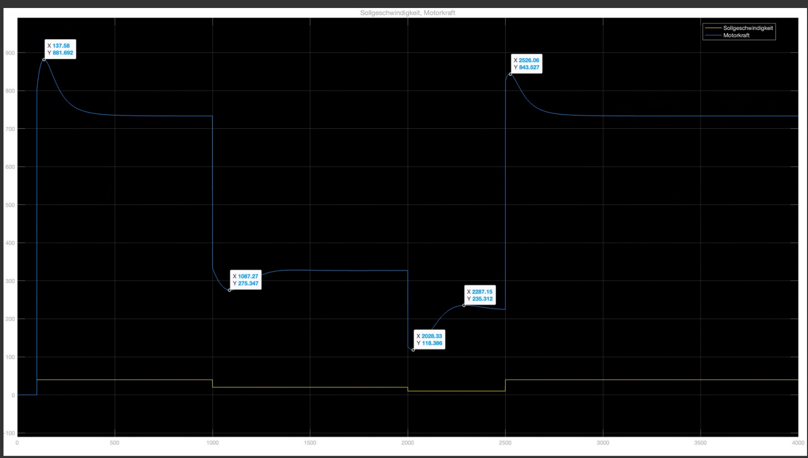
d)

```
function y = fcn(u)
    if u < 100
        y = 0;
    elseif u < 1000
        y = 40;
    elseif u < 2000
        y = 20;
    elseif u < 2500
        y = 20;
    else
        y = 40;
    end
end
```

f)

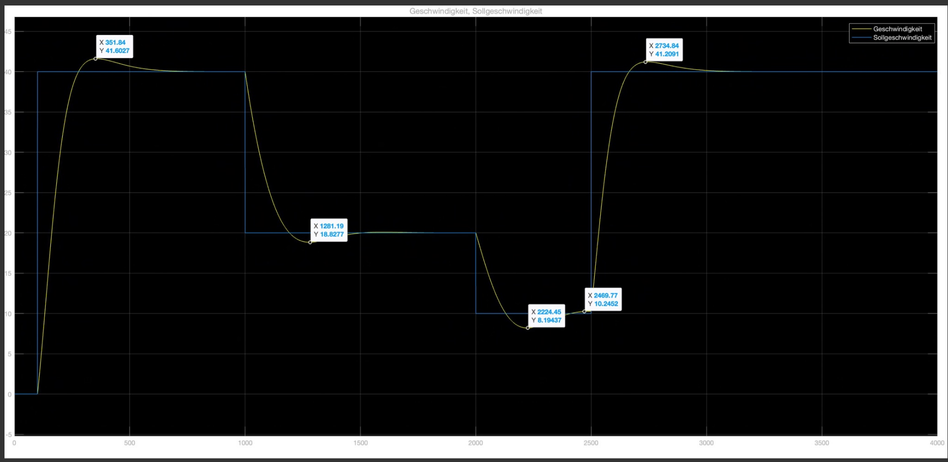


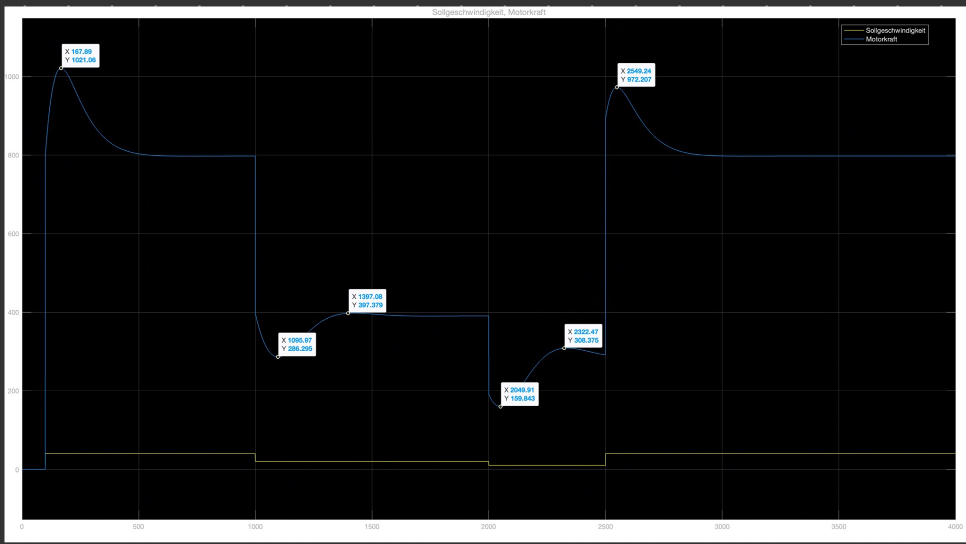
Geschwindigkeit, Sollgeschwindigkeit
Geschwindigkeit
Sollgeschwindigkeit
X 363.59
Y 40.362
X 1278.52
Y 19.5755
X 2192.93
Y 8.71123
X 2446.78
Y 10.126
X 2765.84
Y 40.1969



Sollgeschwindigkeit, Motorkraft
Sollgeschwindigkeit
Motorkraft
X 137.58
Y 881.692
X 1087.27
Y 275.347
X 2028.33
Y 118.386
X 2287.15
Y 235.312
X 2526.06
Y 843.527

g)

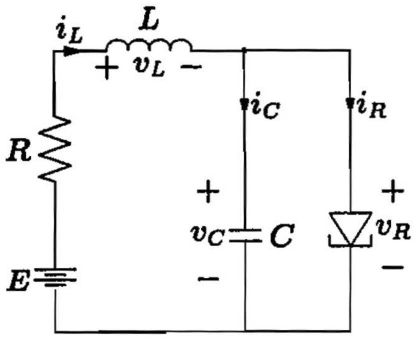




Wie man in den Diagrammen sehen kann, schwingt das System mehr bei mehr Gewicht. z. B. kurz nach $t \geq 100$ ist der Hochpunkt bei $m = 2000\,kg$ um ca. $1{,}3\,\frac{m}{s}$ und $139{,}4\,N$ größer als bei $m = 1500\,kg$

Aufgabe 2 *(60 Punkte)*

In dieser Aufgabe beschäftigen wir uns mit der Simulation und Analyse eines elektrischen Schwingkreises mit Tunneldiode, siehe nachfolgende Abbildung (Chua/Desoer/Kuh). Solch ein Schwingkreis kann zur Erzeugung sehr schnell einschwingenden bistabilen Verhaltens verwendet werden, wie z.B. bei Speicherbausteinen.



Hinweis: Die Aufgabe ist zur Ausführung mit Matlab gedacht. Sie können die ganze Aufgabe aber auch mit Simulink oder Simulink+Matlab lösen und einreichen, wenn Sie möchten. Der Dozent empfiehlt Matlab.

Das dynamische Verhalten des gezeigten Schaltkreises wird näherungsweise durch das folgende nichtlineare Modell beschrieben (alle Größen sind skalar, also nicht vektoriell):

$$\frac{\mathrm{d}}{\mathrm{d}t} v_C(t) = \frac{1}{C} \cdot \left(-h(v_C(t)) + i_L\right) \tag{5}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} i_L(t) = \frac{1}{L} \cdot \left(-v_C(t) - R\, i_L + u\right) \tag{6}$$

$$h(v_C) = \left(17.76 \cdot v_C - 103.79 \cdot v_C^2 + 229.62 \cdot v_C^3 - 226.31 \cdot v_C^4 + 83.72 \cdot v_C^5\right) \cdot 10^{-3} \tag{7}$$

mit

- der Zeit t
- den Zuständen
  - Kondensatorspannung $v_C(t)$
  - Gesamtstrom $i_L(t)$ (der Punkt bedeutet hier nicht die zeitliche Ableitung)
- dem konstanten Eingang $u = E = 1.2$ V
- und den Parametern
  - Widerstand $R = 1500\ \Omega$
  - Kapazität $C = 2 \cdot 10^{-12}$ F
  - Induktivität $L = 5\ \mu$H

Die physikalische Einheit der Strom-Spannungs-Kennlinie $h(\cdot)$ der Tunneldiode ist A (wie bei einer Stromstärke); die Einheiten der einzelnen Koeffizienten wurden der besseren Übersichtlichkeit wegen weggelassen.

a) Schreiben Sie das oben gegebene Differentialgleichungssystem (5)–(6) in der Standardform

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}(t) = \boldsymbol{f}(\boldsymbol{x}(t), u(t)) \qquad \text{mit Zustandsvektor} \quad \boldsymbol{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} v_C(t) \\ i_L(t) \end{bmatrix} \tag{8}$$

Nutzen Sie weiterhin die Abkürzung für die Funktion $h(\cdot)$, sie soll nicht ausgeschrieben werden. *(3 Teilpunkte)*

b) Schreiben Sie ein Matlab-Skript, mit dem der Zeitverlauf des Zustandsvektors $\boldsymbol{x}(t)$ simuliert wird, und zwar für etwa 200 verschiedene, ungefähr gleichmäßig verteilte Kombinationen von Anfangsbedingungen im Bereich $v_C \in [-0.5,\ 1.5]$ V und $i_L \in [-0.5,\ 1.5]$ mA.
Hinweis: Da das dynamische Verhalten sich innerhalb weniger Nanosekunden abspielt, wählen Sie die Endzeit für die Simulation im Bereich 20 ns bis 30 ns, mit entsprechend feiner Simulationsschrittweite. *(20 Teilpunkte)*

c) Simulieren Sie das Verhalten des Systems, und zeichnen Sie Diagramme der Simulationsergebnisse:

- Zeitverlauf von $v_C(t)$
- Zeitverlauf von $i_L(t)$
- Phasenportrait $i_L(t)$ (vertikal) über $v_C(t)$ (horizontal)

Markieren Sie im Phasenportrait automatisiert die Start- und Endwerte jeder Trajektorie.

(15 Teilpunkte)

d) Wieviele Ruhelagen erkennen Sie im Phasenportrait? Lesen Sie die Werte der Ruhelagen ab, ggf. durch Hineinzoomen. *(4 Teilpunkte)*

e) Welche Bedingung muss für die Ruhelagen des Systems gelten, wenn Sie diese aus den Differentialgleichungen (5)–(7) berechnen? Zeigen Sie, dass die Bedingung in der Form

$$h(v_C) = g(v_C)$$

aufgeschrieben werden kann (mit dem $h(v_C)$ aus (7)), und geben Sie $g(v_C)$ an.
Sie brauchen diese Gleichung höherer Ordnung nicht direkt lösen. *(4 Teilpunkte)*

f) Zeichnen Sie die beiden Funktionen $h(v_C)$ und $g(v_C)$ im Bereich $v_C \in [-0.1,\ 1.1]$ V zusammen in ein Diagramm, und bestimmen Sie damit alle Ruhelagen. *(4 Teilpunkte)*

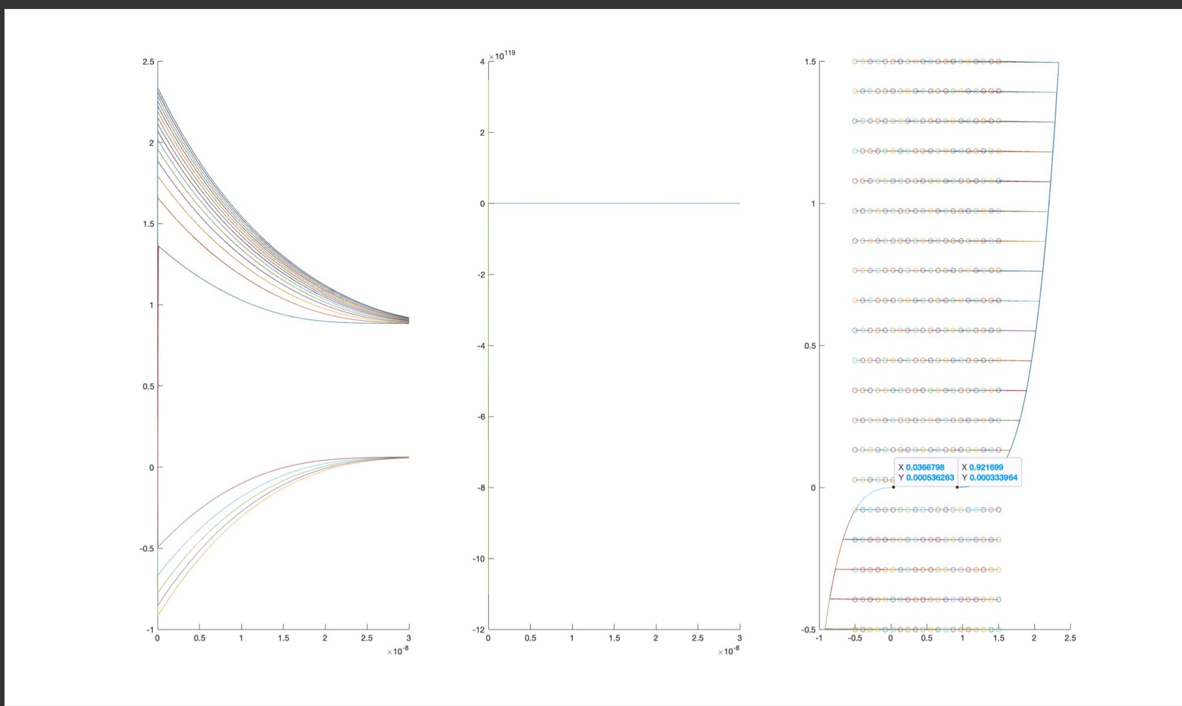
g) Berechnen Sie die $\boldsymbol{A}$-Matrix des linearisierten Systems von (8) für diejenige Ruhelage, für die $v_C^* > 0.5$ V gilt (mit den Werten aus Teilaufgabe d) oder Teilaufgabe f)). Berechnen Sie die Eigenwerte der $\boldsymbol{A}$-Matrix, und geben Sie den Typ (die Klassifikation) der Ruhelage an. *(10 Teilpunkte)* a)

$$\dot{\vec{x}}(t) = \begin{pmatrix} \dot{v}_C(t) \\ \dot{i}_L(t) \end{pmatrix} = \begin{pmatrix} \frac{1}{C} \cdot (-h(v_C(t)) + i_L(t)) \\ \frac{1}{L} \cdot (-v_C(t) - R\, i_L(t) + u) \end{pmatrix}$$

$$= \begin{pmatrix} \frac{1}{C}(-h(x_1(t)) + x_2(t)) \\ \frac{1}{L}(-x_1(t) - R\, x_2(t) + u) \end{pmatrix} = \vec{f}(\vec{x}(t), u)$$

b) s. SIM2023_6028535_Aufgabe2.m

c)



d) sie befinden sich ungefähr bei den oben markierten Punkten. Genauer gesagt sind das $\approx (0{,}06|0)$ und $\approx (0{,}89|0)$

e) Für die Ruhelage muss $\dot{\vec{x}}(t) = 0$ gelten.

$$\begin{cases} (1) & 0 = \frac{1}{C}\left(-h(v_C(t)) + i_L(t)\right) \\ (2) & 0 = \frac{1}{L}\left(-v_C(t) - R\, i_L(t) + u\right) \end{cases}$$

$h(v_C(t))$

aus (1):

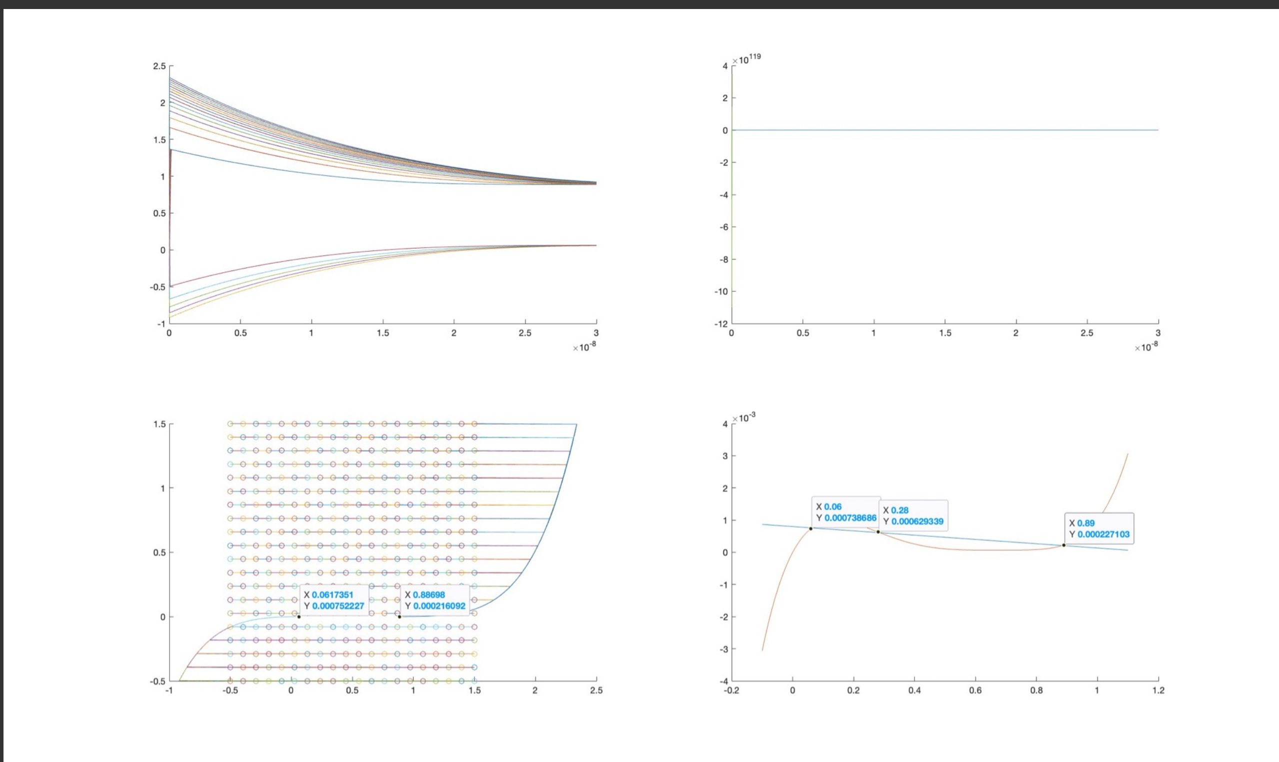
(1a) $h(v_C(t)) = i_L(t)$

aus (2):

(2a) $i_L(t) = \dfrac{u - v_C(t)}{R}$

(2a) in (1a):

$$h(v_C(t)) = \frac{u - v_C(t)}{R} = g(v_C(t))$$



f)

$$h(v_c(t)) = g(v_c(t)) \mid TR$$

$V_{c1} = 0{,}0626$

$V_{c2} = 0{,}2854$

$V_{c3} = 0{,}8844$

g)

$v_c^* = v_{c3} = 0{,}8844$

$$i_L^* = \frac{u - v_c^*}{R} = 2{,}104 \cdot 10^{-4}$$

$$h(v_c(t)) = \Big(\frac{444}{25} - \frac{10379}{50} v_c(t) + \frac{34443}{50} v_c^2(t) \ldots$$

$$\ldots - \frac{22631}{25} v_c^3(t) + \frac{2093}{5} v_c^4(t)\Big) \cdot 10^{-3}$$

$$A = \frac{\partial \vec{f}(\vec{x}(t), u)}{\partial x}\Big|_* = \begin{pmatrix} \frac{1}{C}(-h(v_c^*)) & \frac{1}{C} \\ -\frac{1}{L} & -\frac{R}{L} \end{pmatrix}$$

$$= \begin{pmatrix} 1{,}4364 \cdot 10^9 & 5 \cdot 10^{11} \\ -2 \cdot 10^5 & -3 \cdot 10^8 \end{pmatrix}$$

Eigenwerte:

$$\lambda_1 = 1{,}433 \cdot 10^9 \vee \lambda_2 = -2{,}9997 \cdot 10^{10}$$

Die Ruhelage ist vom Typ her ein instabiler Sattel.