

Technische Universität Dresden

Fakultät Elektrotechnik und Informationstechnik

Institut für Regelungs- und Steuerungstheorie

Studienarbeit

Über den Einfluss hochfrequenter mechanischer Oszillationen auf das Schaltverhalten supraleitender PID-Regler auf Quantenbasis

**Eine Fallstudie unter besonderer Berücksichtigung
stochastischer Einflüsse**

vorgelegt von: Julius Fiedler
geboren am: 13. Oktober 1996 in Dresden

Betreuer:	Betreuer 1
	Betreuer 2
Verantwortlicher Hochschullehrer:	Prof. Dr.-Ing. habil. Dipl.-Math. K. Röbenack
Tag der Einreichung:	2. Februar 2222

Bitte ersetzen Sie diese Seite vor dem Binden mit der Aufgabenstellung.

Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die von mir am heutigen Tage an der Fakultät Elektrotechnik und Informationstechnik eingereichte Studienarbeit zum Thema

Über den Einfluss hochfrequenter mechanischer Oszillationen auf das Schaltverhalten supraleitender PID-Regler auf Quantenbasis

selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Schriften entnommen sind, wurden als solche kenntlich gemacht.

Pirna, 1. Januar 2016

Julius Fiedler

Kurzfassung

An dieser Stelle fügen Sie bitte eine deutsche Kurzfassung ein.

Abstract

Please insert the English abstract here.

Inhaltsverzeichnis

Verzeichnis der Formelzeichen	VI
Abbildungsverzeichnis	VII
Tabellenverzeichnis	VIII
1 Notizen	IX
1.1 UDE	IX
1.1.1 8.7.20	IX
1.1.2 9.7.20	X
1.2 errors	XIII

Verzeichnis der Formelzeichen

Abbildungsverzeichnis

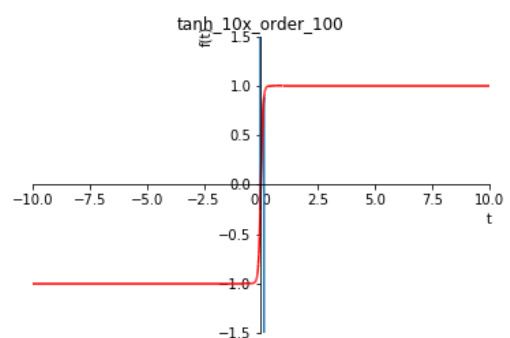
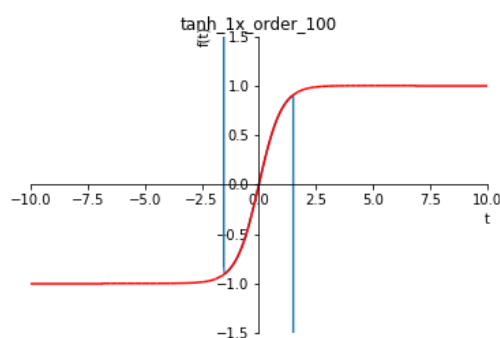
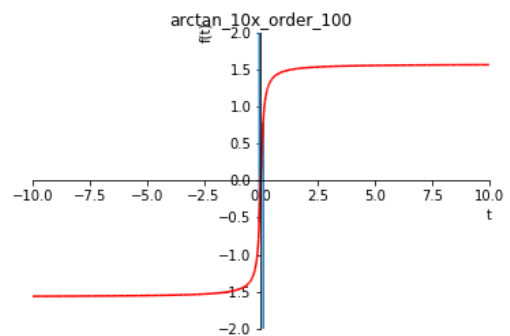
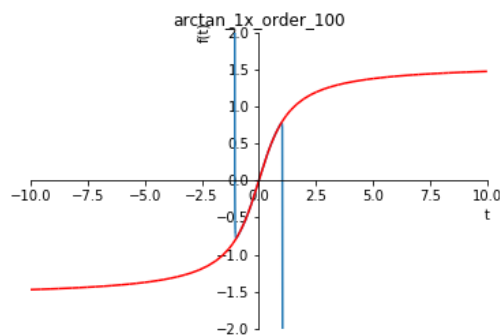
2	nur viskose Reibung	XIV
---	-------------------------------	-----

Tabellenverzeichnis

Kapitel 1

Notizen

1.1 UDE



1.1.1 8.7.20

überlegung NN in julia und sindy in python machen, export erledigt
trotzdem keine identifikation in python möglich, grund: zu wenig daten (datenfeld mit 31 daten Auflösung viel zu gering)

bei erhöhung der auflösung wird trainingszeit enorm groß NN approximiert den verlauf der ableitungen
sindy mal mit anderen anfangswerten trainieren

1.1.2 9.7.20

ude keine option für multiple trajectories / threshhold???

ude + sindy

definitorische Gleichungen bekannt

dt=0.1

sindy did not converge

$$du_1 = \cos(u_3) * p_1 \quad (1.1)$$

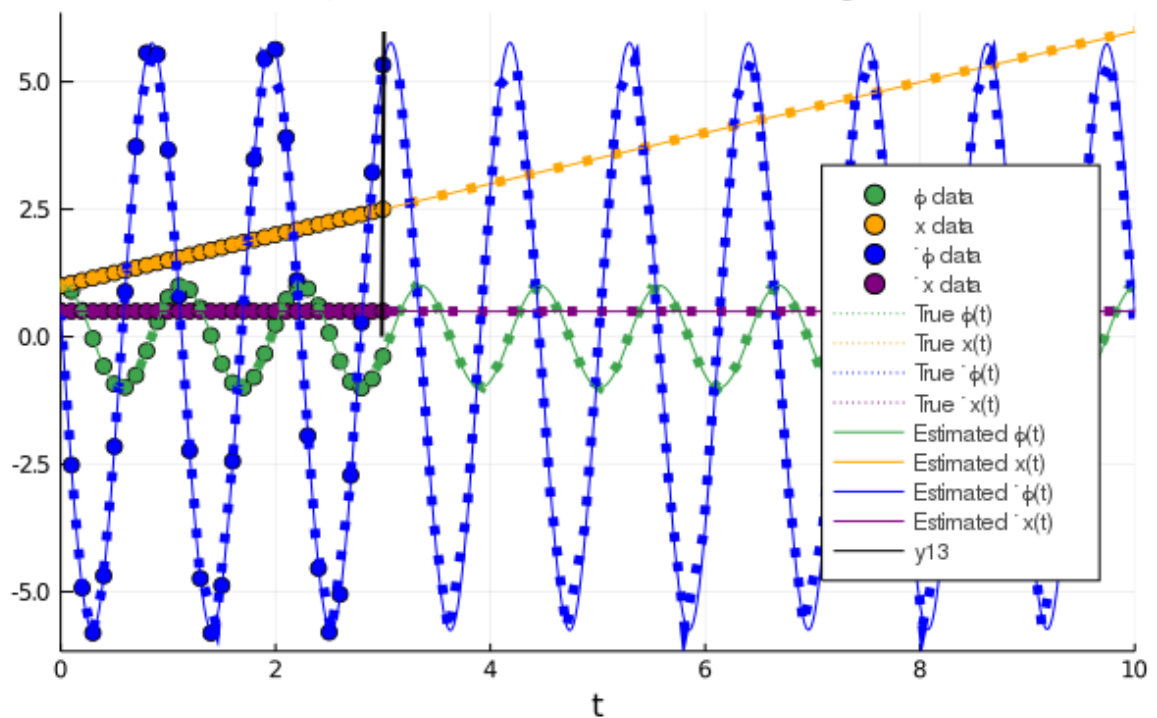
$$du_2 = \sin(u_3) * p_2 \quad (1.2)$$

$$du_3 = \sin(u_1) * p_3 + p_4 * u_1 \quad (1.3)$$

$$du_4 = \sin(u_3) * p_5 \quad (1.4)$$

parameters: Float32[0.050624948, 0.006272168, -20.755184, -13.720979, -0.01513608]

Extrapolated Fit From Short Training Data



im Vergleich: nur den Sindy Part in PySindy ausgelagert ergibt: (dieser Vergleich ist nicht sinnvoll? pysindy kennt ja die def. gl nicht)

$$\begin{aligned}
 \phi' &= 65170415.755x\dot{} + -72812822.138\sin(x\dot{}) + 2647179.524\cos(x\dot{}) \\
 x' &= 51018334.340x\dot{} + -57002163.483\sin(x\dot{}) + 2072882.755\cos(x\dot{}) \\
 \phi\dot{} &= -3.002\phi + -6.328x + 2682545588.568x\dot{} + -33.032\sin(\phi) \\
 &\quad + -1.358\sin(x) + -0.208\sin(\phi\dot{}) + -2997205703.337\sin(x\dot{}) \\
 &\quad + -0.020\cos(\phi) + -6.583\cos(x) + 109008747.059\cos(x\dot{}) \\
 x\dot{} &= 0.320x + -18300414.600x\dot{} + 20447317.167\sin(x\dot{}) + 0.344\cos(x) \\
 &\quad + -743815.199\cos(x\dot{})
 \end{aligned}$$

keine Gleichungen bekannt

dt=0.1

sindy did not converge

$$du_1 = p_1 * u_3 \quad (1.5)$$

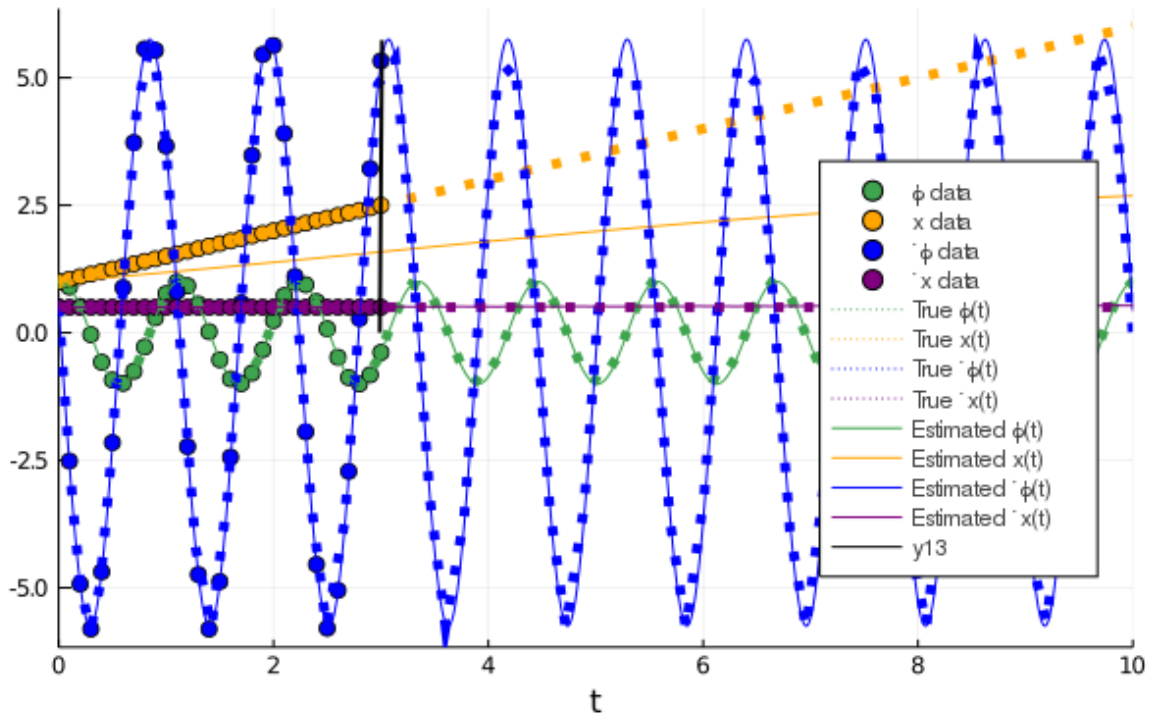
$$du_2 = \sin(u_2) * p_2 \quad (1.6)$$

$$du_3 = \sin(u_1) * p_3 + p_4 * u_1 \quad (1.7)$$

$$du_4 = \cos(u_1) * p_5 \quad (1.8)$$

parameters: Float32[0.997858, 0.20674846, -18.494001, -15.788721, 0.040644586]

Extrapolated Fit From Short Training Data



im Vergleich: nur den Sindy Part in PySindy ausgelagert ergibt:

$$\begin{aligned}
 \phi' &= 1.111\phi + 0.648x + 1.002\phi\dot{} + -503701656.178x\dot{} + -1.228\sin(\phi) \\
 &\quad + 562779966.638\sin(x\dot{}) + -0.050\cos(\phi) + 0.676\cos(x) + -20465608.922\cos(x\dot{}) \\
 x' &= 0.833\phi + -0.031x + 450669011.982x\dot{} + -0.967\sin(\phi) \\
 &\quad + -503527393.951\sin(x\dot{}) + 0.157\cos(\phi) + -0.051\cos(x) + 18310968.307\cos(x\dot{}) \\
 \phi\dot{} &= -2.551\phi + -4.834x + -3252316739.183x\dot{} + -33.509\sin(\phi) \\
 &\quad + -0.458\sin(x) + 3633781720.861\sin(x\dot{}) + -5.066\cos(x) + -132146405.440\cos(x\dot{}) \\
 x\dot{} &= 0.674\phi + -184783222.592x\dot{} + -0.759\sin(\phi) + 206455836.816\sin(x\dot{}) + \\
 &\quad - 7507657.694\cos(x\dot{})
 \end{aligned}$$

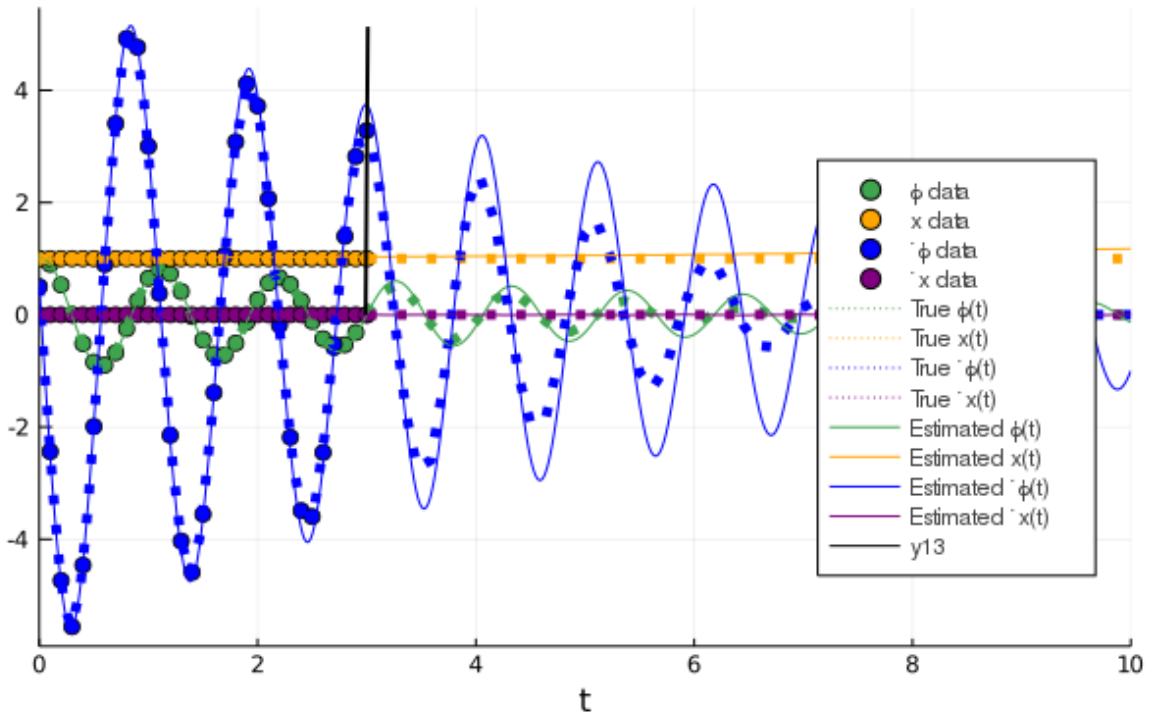
ude reibung

tanh wie erwartet problematisch, vmtl sinnvoll tanh vorzugeben
viskose+ haftreibung:

$$\begin{aligned}
 du_1 &= \sin(u_3) * p_1 \\
 du_2 &= \cos(u_1) * p_2 + \cos(u_3) * p_3 \\
 du_3 &= p_4 * u_3 \\
 du_4 &= \sin(u_3) * p_5
 \end{aligned}$$

parameters: Float32[-0.032978103, 0.020778582, 0.02180107, -0.30371103, -0.026243187]

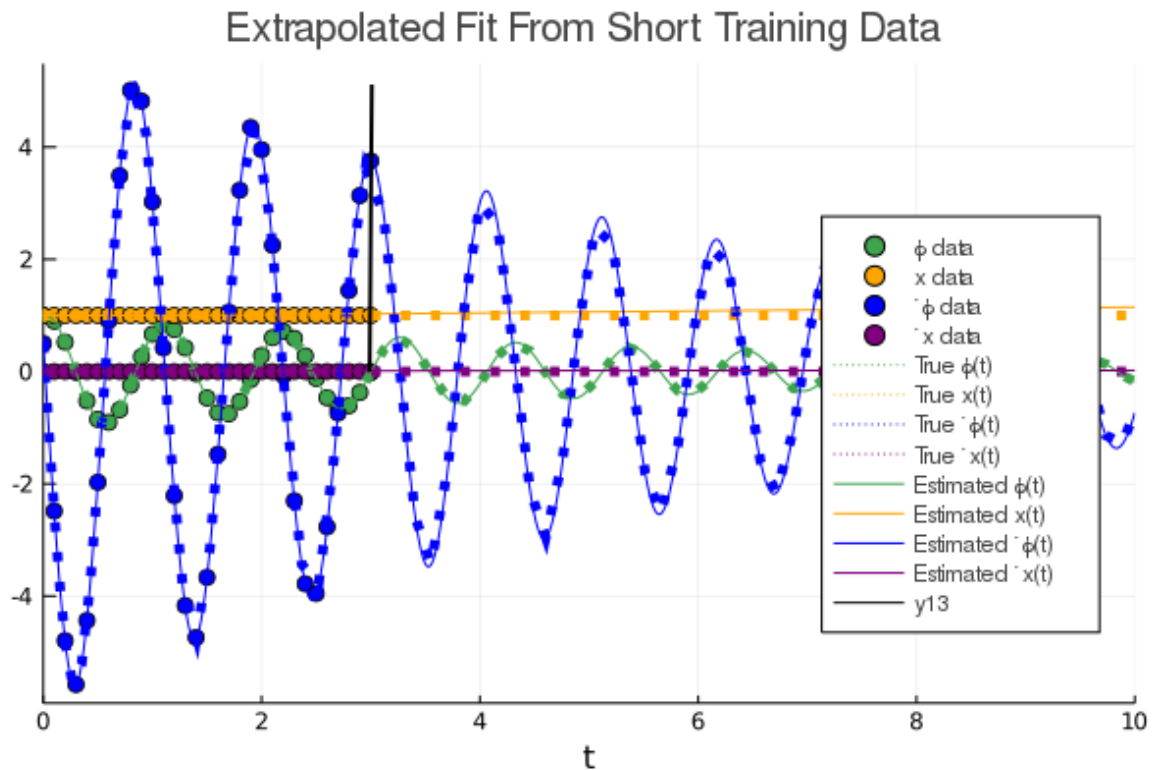
Extrapolated Fit From Short Training Data



nur viskose reibung, order 2:

$$d_1 = 0.3$$

$$d_{1,ident} = 0.2981498$$



Pysindy reibung

Ansatz: Sindy für Differenz von Realem Modell mit Reibung und theoretischem Modell verwenden, siehe Bsp:

keine Identifikation feststellbar

vermutung: aus Differenz geht keine exp funktion hervor, es fehlt die info über das vorherige System idee: prior system knowledge integrieren indem man das bekannte teilsystem aus library funktion zur verfügung stellt

1.2 errors

Ude:

`AssertionError : length(b) == length(variables(b))inunknown_sys :`

wenn λ so gewählt, dass ganze Zeilen rausfallen

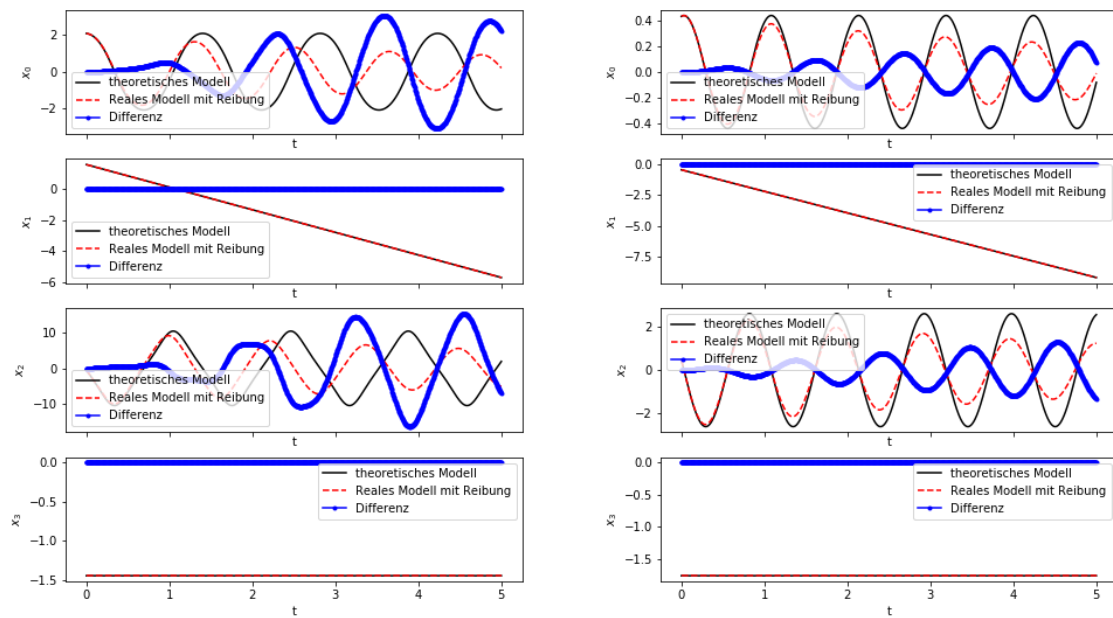


Abbildung 2 – nur viskose Reibung