



Flachheitsbasierte Regelung des RoPa30 unter Berücksichtigung von Totzeiten

Jennifer Wickert
Festo – Robotics System Design, Dept. BC-DS

Rüter Str. 82
D-73734 Esslingen, Germany
E-Mail: jennifer.wickert@festo.com

31. August 2023

Sperrvermerk

Die nachfolgende Masterarbeit enthält vertrauliche Daten und Informationen der Firma Festo SE & Co. KG. Veröffentlichungen oder Vervielfältigungen - auch nur auszugsweise - sind ohne ausdrückliche Genehmigung des Unternehmens nicht gestattet.

Die Masterarbeit ist nur den Korrektoren sowie den Mitgliedern des Prüfungsausschusses zugänglich zu machen.

Eidesstattliche Erklärung

Hiermit erkläre ich,

1. dass ich meine Masterarbeit selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel verwendet habe.
2. dass ich die Übernahme wörtlicher Zitate aus der Literatur sowie die Verwendung der Gedanken anderer Autoren an den entsprechenden Stellen innerhalb der Arbeit gekennzeichnet habe.
3. dass ich die Masterarbeit bei keiner anderen Prüfung vorgelegt habe.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Stuttgart, den 31. August 2023_____

Abstract

in classFesto.cls

Kurzfassung

in classFesto.cls

Inhaltsverzeichnis

List of Figures	III
List of Tables	V
1. Einführung und Motivation	1
2. Antriebsmodell	2
2.1. Mechanisches Modell	4
2.2. Pneumatisches Modell	5
2.3. Ventilmodell	5
3. Theorie	7
3.1. Differentielle Flachheit	7
3.1.1. Differentielle Flachheit für RoPa30	7
3.1.2. Inversion des Ventilmodells	9
3.2. Delta Flachheit	9
3.2.1. Definition Delta Flatness für nichtlineare Systeme	9
3.2.2. Delta Flachheit für RoPa30	10
Prädiktionsalgorithmus	13
4. Simulationsmodell	21
4.1. Simulationsmodell des Schwenkantriebs mit FOSIL	21
5. Flachheitsbasierte Regelung des RoPa30 Antriebs	23
5.1. Simulationsergebnisse	23
5.1.1. Test inverses Modell	24
5.2. Regelergebnisse	25
5.2.1. Regelergebnisse	26
5.2.2. Test auf Robustheit des Reglers	27
6. Flachheitsbasierte Regelung des RoPa30 Antriebs unter Berücksichtigung von Totzeiten	30
6.1. Reglerergebnisse	31
6.1.1. Mit Totzeit, mit Prädiktion	31
6.1.2. Vergleich mit und ohne Prädiktion	34
6.1.3. Robustheit des δ -flachen Reglers	38
7. Test am Versuchsstand	39

8. Zusammenfassung und Diskussion	40
A. Beispiel für Anhang	41

Abbildungsverzeichnis

2.1. Pneumatischer Schwenkflügelantrieb, durch vier Ventile angesteuert.	2
2.2. Topologie eines pneumatischen Gelenks bestehend aus Schwenkantrieb (Swivel Drive), Piezoventilemodul (VEAE-Modul), Main Board, Positionsenncoder und Safety Board/ Platine.	3
2.3. Safety Board, Encoder (links) und VEAE-Modul (rechts).	4
4.1. Modell des Schwenkantriebs mit der FOSIL-Toolbox. [aus FOSIL-Tutorial, 20.06.2023]	21
5.1. Simulationsmodell des flachheitsbasierten Reglers für den RoPA30-Antrieb.	24
5.2. Partielle Ableitung $\frac{\partial \tau_f}{\partial \dot{q}}$ des Reibmoments.	25
5.3. Gewünschter Positionsverlauf φ_d (rad). [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_1]	26
5.4. Solldruckverlauf p_2, p_4 (Pa). [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_1]	27
5.5. Oben Positions- und unten Geschwindigkeitsfehler. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_1]	28
5.6. Druckfehler in Kammer 2 oben; in Kammer 4 unten. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_1]	28
5.7. Gewünschter Massenstromverlauf (-) vs. simulativer Massenstrom (-). [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_1]	29
6.1. Simulationsmodell des δ -flachen Reglers für den RoPA30-Antrieb mit der Delta-Schreibweise $\delta^\tau f := f(t - \tau)$	30
6.2. Positionsfehler $e_\varphi = \varphi_d - \varphi$ und Geschwindigkeitsfehler bei Totzeiten und Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]	32
6.3. Druckfehler $e_p = p_d - p$ bei Totzeiten und Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]	32
6.4. Simulativer Massenstrom in orange, sowie Sollmassenstrom in blau bei Totzeiten und Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]	33
6.5. Mitteldruckverlauf bei Totzeiten und Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]	34
6.6. Prädiktions- und Delay-Fehler im Winkel. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]	35
6.7. Prädiktions- und Delay-Fehler im Druck. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]	35

6.8. Soll- und Ist-Position ohne Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_3]	36
6.9. Soll- und Ist-Massenstrom ohne Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_3]	37
6.10. Soll- und Ist-Druck ohne Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_3]	37

Tabellenverzeichnis

4.1. Parameterwerte des RoPA30-Schwenkantriebs.	22
4.2. Datensets für den flachen Regler für den RoPA30-Schwenkantrieb.	22

1. Einführung und Motivation

....

2. Antriebsmodell

Ein Beispiel für einen FESTO-Collabor-Robot (Cobot) ist der "ROPA-650-3" ("Clarissa"), welcher aus fünf pneumatischen Gelenken (RoPA = rotational pneumatic actuator) besteht. Diese werden jeweils durch einen Schwenkantrieb, wie in Abbildung 2.1, gesteuert. Der Schwenkantrieb besteht aus den Kammern ② und ④, zu welchen Druck entweder

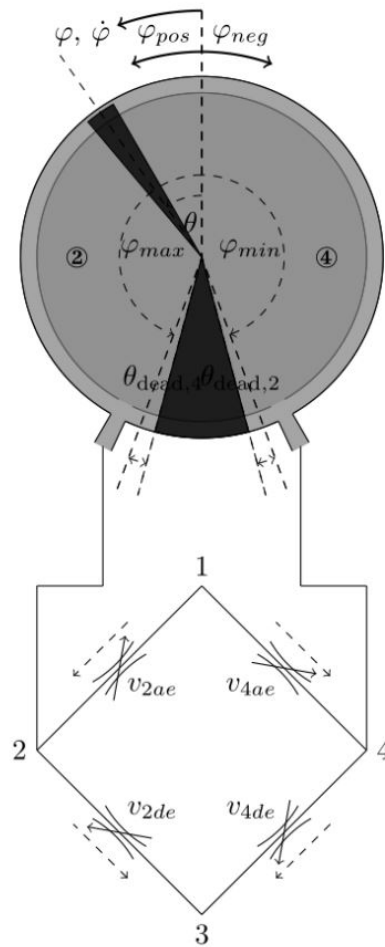


Abbildung 2.1.: Pneumatischer Schwenkflügelantrieb, durch vier Ventile angesteuert.

zu- oder abgeführt wird. Damit sich also z.B. der Kolben in positive φ -Richtung bewegt, wird Kammer ④ Druck in Form eines Massenstroms \dot{m}_4 zugeführt, während Kammer

② Druck abgeführt wird (negativer Massenstrom \dot{m}_2). Dieser Massenstrom wird jeweils durch zwei Piezoventile generiert, d.h. jedes pneumatische Gelenk besitzt insgesamt vier Piezoventile. In dieser Arbeit wird der RoPA30-Antrieb untersucht.

An jedem Antrieb sind Sensoren verbaut, welche die Drücke p_2, p_4 , sowie die Kolben-Position φ messen. Die Position wird hierbei mittels des EnDat Encoders bestimmt. Wie in Abbildung 2.2 skizziert, wird die Positionsmessung und zusammen mit den Druckmessungen an das Safety Board kommuniziert. Dieses befindet sich beim Schwenkantrieb,

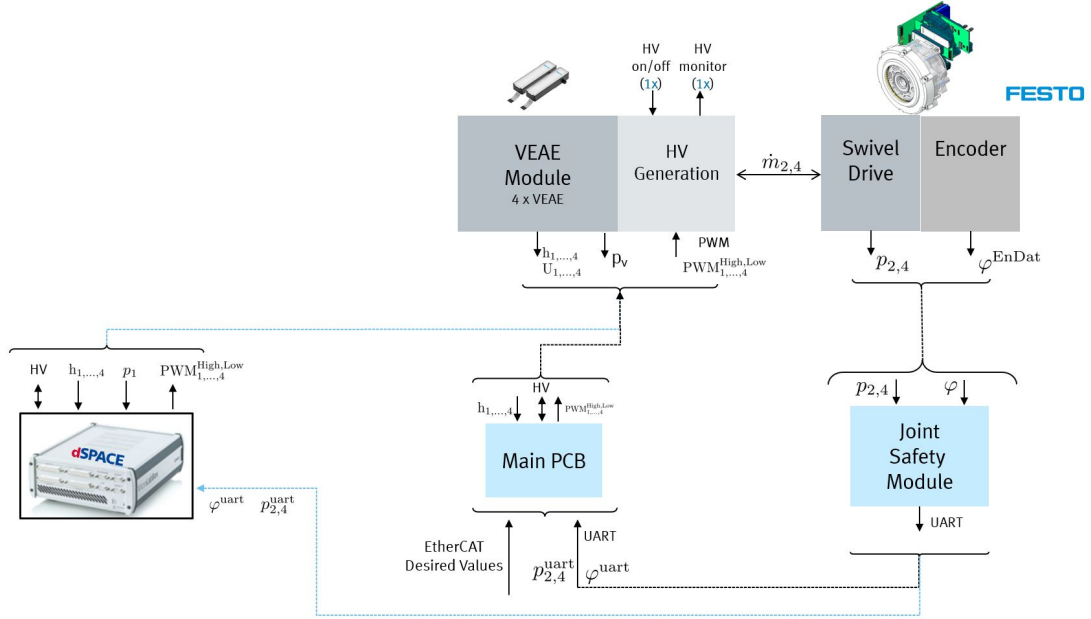


Abbildung 2.2.: Topologie eines pneumatischen Gelenks bestehend aus Schwenkantrieb (Swivel Drive), Piezoventilmodul (VEAE-Modul), Main Board, Positionsenncoder und Safety Board/ Platine.

wie in Abbildung 2.3, und überprüft die Messungen auf zulässige Werte, sowie eventuelle Defekte in der Elektronik. Falls Defekte erkannt werden, wird der Roboter in einen sicheren Zustand überführt, d.h. die Bremsen werden gezogen und Drücke wieder auf Umgebungsdruck abgelassen. Hierdurch wird eine sichere Mensch-Roboter-Kollaboration ermöglicht.

Werden keine Beanstandungen vom Safety-Modul gefunden, so werden die Messwerte über UART an das Main Board kommuniziert. UART (Universal-Asynchron-Receiver-Transmitter) ist eine serielle Schnittstelle zur Kommunikation zwischen zwei Boards.

Das Main Board beinhaltet die kaskadierte Hub-Regelung der Piezoventile im VEAE-Modul, welches ebenfalls in Abbildung 2.3 zu sehen ist. Der Hub h beschreibt den Öffnungsgrad der Piezoventile, da diese je nach anliegender Spannung unterschiedlich stark gebogen werden. Anhand der gemessenen Hub- h_i , sowie Spannungswerte U_i des

VEAE-Moduls, wird ein PWM-Signal $U_i^{PWM} = 3.3V$ mithilfe des Reglers im Main-board erzeugt. Dieses wird nun durch einen Hochsetzsteller zur Hochvolterzeugung auf die Ventil-Nennspannung von 300V erhöht.

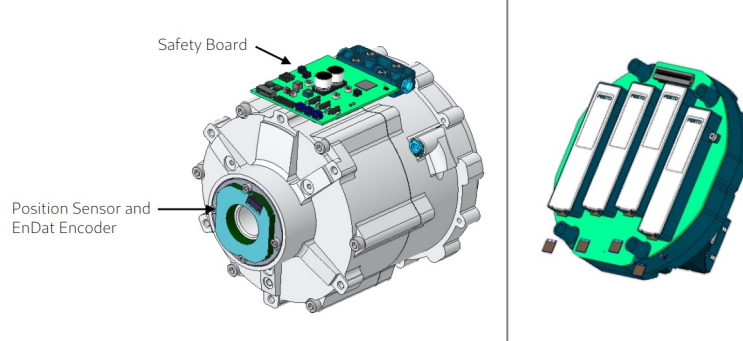


Abbildung 2.3.: Safety Board, Encoder (links) und VEAE-Modul (rechts).

Die Taktfrequenz der internen kaskadierten Regelung beträgt 16kHz, und ist somit in ausreichendem Maße größer als die Echtzeit-Taktfrequenz 1kHz des Main Boards. Letztendlich wird aus dem Ventilhub h_1, \dots, h_4 der Massenstrom \dot{m}_2, \dot{m}_4 für beide Kammern bestimmt und auf den Schwenkantrieb gegeben.

2.1. Mechanisches Modell

Mit den in Abbildung 2.1 eingezeichneten Größen, liefert die Drehimpulserhaltung die Differentialgleichung 2. Ordnung

$$J\ddot{q} = \tau_p - \tau_f. \quad (2.1)$$

Hierbei beinhaltet das Reibungsmoment τ_f ein Coulomb-, sowie viskoses Reibungsmodell

$$\begin{aligned} \tau_f &= \tau_{\text{coulomb}} + \tau_{\text{viscous}} \\ &= f_c \tanh \frac{\dot{q}}{\dot{q}_c} + f_v \dot{q}. \end{aligned} \quad (2.2)$$

\dot{q}_c ist ein Faktor mit der Einheit einer Winkelgeschwindigkeit, welcher die *tanh*-Funktion glättet, sodass keine numerischen Probleme auftauchen.

Das (Druck-)Antriebsmoment τ_p hingegen resultiert aus der Druckdifferenz in den Kammern

$$\tau_p = A r_m (p_4 - p_2), \quad (2.3)$$

wobei A der Querschnitt und r_m den effektiven Radius des Schwenkantriebs bzw. der Kammern.

2.2. Pneumatisches Modell

Für den Druck wird ein polytropher Prozess [Brun, 1999]

$$\dot{p}_i = \frac{n}{V_i(q)} \left(RT\dot{m}_i - \dot{V}_i(q)p_i \right) \quad \text{für } i \in \{2, 4\} \quad (2.4)$$

angenommen. Hierbei sind n der polytrope Koeffizient, R die spezifische Gaskonstante, sowie T die Umgebungstemperatur.

Die Volumina werden entsprechend Abbildung 2.1 wie folgt berechnet

$$V_2 = Ar_m(q_{\max} - q) \quad (2.5a)$$

$$V_4 = Ar_m(-q_{\min} + q) \quad (2.5b)$$

$$\dot{V}_2 = -Ar_m\dot{q} \quad (2.5c)$$

$$\dot{V}_4 = Ar_m\dot{q} \quad (2.5d)$$

mit den Endanschlägen bei q_{\min} und q_{\max} .

2.3. Ventilmodell

Die vier Ventile werden abhängig von den (normierten) Öffnungsvariablen der Belüftungs- (ae) und Entlüftungsventile (de) $v_{2,ae}, v_{2,de}, v_{4,ae}, v_{4,de}$ geöffnet. Diese entsprechen der auf das Intervall $[0, 1] \subset \mathbb{R}$ normierten Hubvariable, es gilt also z.B. für das Belüftungsventil von Kammer 2:

- wenn $v_{2,ae} = 0$, dann ist das Ventil vollständig geschlossen
- wenn $v_{2,ae} = 1$, dann ist das Ventil vollständig geöffnet.

Das Öffnungssignal lässt sich aus dem Steuer/Reglereingang $\mathbf{v} = [v_2, v_4]^T \in [-1, 1] \times [-1, 1]$ bestimmen:

$$v_{i,ae} = \begin{cases} v_i & , v_i > 0 \\ 0 & , v_i \leq 0 \end{cases} \quad (2.6)$$

$$v_{i,de} = \begin{cases} -v_i & , v_i < 0 \\ 0 & , v_i \geq 0 \end{cases} \quad (2.7)$$

für die Kammern $i \in \{2, 4\}$.

Der pneumatische Leitwert C ist umgekehrt proportional zum pneumatischen Widerstand, welcher gegen das Ventilöffnen wirkt. Die charakteristische Kurve von $C(v, \Delta p)$ ist in der Regel nichtlinear und wird in Abhängigkeit der Ventilöffnung und Druckdifferenz experimentell ermittelt. Ein typischer Verlauf von C ist in [?] dargestellt. In dieser

Arbeit wird eine Modellreduktion durchgeführt, wodurch die Druckdifferenz verschwindet und es einen eindeutigen Zusammenhang zwischen Leitwert und Ventilöffnen ergibt. Somit gilt für jede Kammer i

$$\dot{m}_i = C_{\max} C_{\text{rel}}(v_i) p_{\text{prim},i} \rho \Psi\left(b, \frac{p_{\text{sec},i}}{p_{\text{prim},i}}\right) \quad (2.8)$$

mit ρ der Luftdichte, p_{prim} der Luftdruck vor dem Ventil, p_{sec} der Luftdruck nach dem Ventil, kritischem Druckverhältnis b , $C = C_{\max} C_{\text{rel}}(v_i)$ und der Flussfunktion Ψ .

Da die Ventile, welche für FESTO verwendet werden, bereits einen internen Hubregler haben, ist die Kurve des pneumatischen Leitwerts hier linear bzgl. dem Hub. [?]

Der Massenstrom in der Kammer ist positiv, wenn es einen Massenzufluss gibt, und negativ, wenn es einen Massenabfluss gibt.

3. Theorie

3.1. Differentielle Flachheit

(Differentielle) Flachheit beschreibt die Steuerbarkeit für nichtlineare Systeme. Für flache Systeme kann ein Ausgang gefunden werden, sodass alle Zustände, sowie Eingänge in Abhängigkeit dieses Ausgangs und seiner Zeitableitungen dargestellt werden können. In [Marx, 2015] wird eine Anleitung zur Bestimmung der differentiellen Flachheit gegeben, sowie zur Trajektorienfolgeregelung. Aus diesem Grund werden hier nur die für diese Arbeit relevanten Ergebnisse angegeben.

3.1.1. Differentielle Flachheit für RoPa30

Es werden der Eingang

$$\mathbf{u} = \begin{bmatrix} \dot{m}_2 \\ \dot{m}_4 \end{bmatrix} \quad (3.1)$$

und der Zustand

$$\mathbf{x} = \begin{bmatrix} q \\ \dot{q} \\ p_2 \\ p_4 \end{bmatrix} \quad (3.2)$$

mit $q = \varphi$ definiert. Die Antriebsdynamik in Zustandsraumdarstellung ergibt sich dann aus den Gleichungen (2.1), (2.2), (2.3) zu

$$\dot{\mathbf{x}} = \begin{bmatrix} x_2 \\ \frac{1}{J} \left(Ar_m(x_4 - x_3) - \tau_f(x_2) \right) \\ -\frac{nx_3\dot{V}_2(x_2)}{V_2(x_1)} \\ -\frac{nx_4\dot{V}_4(x_2)}{V_4(x_1)} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{nRT}{V_2(x_1)} \\ 0 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{nRT}{V_4(x_1)} \end{bmatrix} u_2 \quad (3.3)$$

Der flache Ausgang wird gewählt zu

$$\mathbf{y}(t) = \begin{bmatrix} q(t) \\ p_m(t) \end{bmatrix} \quad (3.4)$$

mit der Pistonposition q [rad] und dem Mitteldruck

$$p_m(t) = \frac{p_2(t) + p_4(t)}{2}. \quad (3.5)$$

Jeder Zustand, sowie Eingang lässt sich durch den flachen Ausgang beschreiben:

$$z_1(t) = y_1(t) \quad (3.6)$$

$$z_2(t) = \dot{y}_1(t) \quad (3.7)$$

$$z_3(t) = y_2(t) - \frac{J\ddot{y}_1(t) + \tau_f(\dot{y}_1(t))}{2Ar_m} \quad (3.8)$$

$$z_4(t) = y_2(t) + \frac{J\ddot{y}_1(t) + \tau_f(\dot{y}_1(t))}{2Ar_m} \quad (3.9)$$

$$\begin{aligned} z_5(t) = & \frac{\dot{V}_2(\dot{y}_1(t))}{2Ar_m RT} \left(2Ar_m y_2(t) - J\ddot{y}_1(t) - \tau_f(\dot{y}_1(t)) \right) \\ & + \frac{V_2(y_1(t))}{2Ar_m nRT} \left(2Ar_m \dot{y}_2(t) - J\ddot{y}_1(t) - \frac{d}{dt} \left(\tau_f(\dot{y}_1(t)) \right) \right) \end{aligned} \quad (3.10)$$

$$\begin{aligned} z_6(t) = & \frac{\dot{V}_4(\dot{y}_1(t))}{2Ar_m RT} \left(2Ar_m y_2(t) + J\ddot{y}_1(t) + \tau_f(\dot{y}_1(t)) \right) \\ & + \frac{V_4(y_1(t))}{2Ar_m nRT} \left(2Ar_m \dot{y}_2(t) + J\ddot{y}_1(t) + \frac{d}{dt} \left(\tau_f(\dot{y}_1(t)) \right) \right) \end{aligned} \quad (3.11)$$

Zur Folgeregelung wird das System zunächst auf Brunovsky-Normalform

$$\begin{aligned} \dot{y}_{11} &:= \dot{y}_1 = y_{12} \\ \dot{y}_{12} &:= \ddot{y}_1 = y_{13} \\ \dot{y}_{13} &:= \ddot{\ddot{y}}_1 = \nu_1(t) \\ \dot{y}_2 &= \nu_2(t) \end{aligned}$$

mit der neuen Stellgröße ν überführt. Dann lautet der Eingang \mathbf{u} für die Folgeregelung

$$\begin{aligned} \dot{m}_2 = & \frac{\dot{V}_2(\dot{y}_1(t))}{2Ar_m RT} \left(2Ar_m y_2(t) - J\ddot{y}_1(t) - \tau_f(\dot{y}_1(t)) \right) \\ & + \frac{V_2(y_1(t))}{2Ar_m nRT} \left(2Ar_m \nu_2(t) - J\nu_1(t) - \frac{\partial \tau_f(\dot{y}_1(t))}{\partial \dot{y}_1} \dot{y}_1(t) \right) \\ \dot{m}_4 = & \frac{\dot{V}_4(\dot{y}_1(t))}{2Ar_m RT} \left(2Ar_m y_2(t) + J\ddot{y}_1(t) + \tau_f(\dot{y}_1(t)) \right) \\ & + \frac{V_4(y_1(t))}{2Ar_m nRT} \left(2Ar_m \nu_2(t) + J\nu_1(t) + \frac{\partial \tau_f(\dot{y}_1(t))}{\partial \dot{y}_1} \dot{y}_1(t) \right). \end{aligned} \quad (3.12)$$

Der Positionsregler wird als PT3-Glied und Druckregler als PT1-Glied gewählt, um den Folgefehler zu stabilisieren

$$\nu_1(t) = \ddot{y}_{1,des} + c_{12}(\ddot{y}_{1,des} - \ddot{y}_1) + c_{11}(\dot{y}_{1,des} - \dot{y}_1) + c_{10}(y_{1,des} - y_1) \quad (3.13)$$

$$\nu_2(t) = \dot{y}_{2,des} + c_{20}(y_{2,des} - y_2). \quad (3.14)$$

3.1.2. Inversion des Ventilmodels

Aus dem Massenstrom wird nun mithilfe der Gleichung (2.8) die Ventil-Inversion bestimmt [?]:

$$C_{rel,i} = \frac{\dot{m}_i}{C_{max} p_{prim,i} \rho \Psi\left(b, \frac{p_{sec,i}}{p_{prim,i}}\right)} \quad (3.15)$$

$$v_i = C_{rel}^{-1}(C_{rel,i}) \quad (3.16)$$

für $i \in \{2, 4\}$.

Die Gleichungen (3.12) und (3.15) beschreiben die Inversion der Strecke, d.h. die Berechnung des Massenstroms aus den Ventilöffnungen.

3.2. Delta Flachheit

Für komplexe Systeme mit Totzeiten funktionieren Standard-Regelverfahren i.d.R. nicht wie gewünscht bzw. führen zu Ungenauigkeiten. Aus diesem Grund wird die "δ-freeness" [Mounier, 1995] für lineare Totzeit-Systeme, zusammen mit der Flachheitsdefinition [Fliess, 1995], zur "δ-flatnes" für nichtlineare Systeme in erweitert [Mounier, 1998].

3.2.1. Definition Delta Flatness für nichtlineare Systeme

Es sei ein beliebiges MIMO System mit Systemvariablen $\mathbf{z} \in \mathbb{R}^c$ (kann ein beliebiger mix aus inputs, zuständen,... sein)

$$\begin{aligned} \mathbf{F}(\mathbf{z}, \dot{\mathbf{z}}(t - \sum_{i=1}^r \tau_i s_{1i}), \ddot{\mathbf{z}}(t - \sum_{i=1}^r \tau_i s_{2i}), \dots, \mathbf{z}^{(a)}(t - \sum_{i=1}^r \tau_i s_{ai})) = \mathbf{0} \\ \Leftrightarrow: \\ \mathbf{F}(\mathbf{z}, \delta^{z_1} \dot{\mathbf{z}}, \delta^{z_2} \ddot{\mathbf{z}}, \dots, \delta^{z_a} \mathbf{z}^{(a)}) = \mathbf{0} \end{aligned} \quad (3.17)$$

gegeben.

Das nichtlineare System $\mathbf{F} \in \mathbb{R}^N$ mit N der Anzahl der nichtlinearen DGLs hängt also von beliebigen Ableitungen bis zur höchsten Ableitung in der Systemvariablen $\mathbf{z}^{(a)}$ ab, sowie von Zeitverschiebungen τ_i , $i \in \{1, 2, \dots, r\}$ mit r der Anzahl aller verschiedenen Zeitverschiebungen im System ab. Hierbei sind $s_{ki} \geq 0$ mit $(k, i) \in \{1, \dots, a\} \times \{1, \dots, r\}$ die Verschiebefaktoren der i -ten Zeitverzögerung zugehörig zur k -ten Ableitung von \mathbf{z} . Der zugehörige δ -Verschiebeoperator ist definiert als

$$\delta_i^{s_{ki}} z^{(k)}(t) = z^{(k)}(t - \tau_i s_{ki}) \quad \forall (k, i) \in \{1, \dots, a\} \times \{1, \dots, r\}. \quad (3.18)$$

Ein System (3.17) heißt δ -Flach, wenn ein δ -flacher Ausgang $\mathbf{y} \in \mathbb{R}^m$ existiert, sodass

- alle δ -flachen Ausgänge können in Abhängigkeit der Systemvariablen angegeben werden:

$$y_j = P_j(\mathbf{z}, \delta^{z_1} \delta^{-\tilde{z}_1} \dot{\mathbf{z}}, \delta^{z_2} \delta^{-\tilde{z}_2} \ddot{\mathbf{z}}, \dots, \delta^{z_{\hat{a}}} \delta^{-\tilde{z}_{\hat{a}}} \mathbf{z}^{(\hat{a})}) \quad \forall j \in \{1, \dots, m\} \quad (3.19)$$

mit $\delta^{-\tilde{z}_a}$ positive Zeitverschiebungen, bzw. Vorhersagen in die Zukunft. Je nach Wahl der flachen Ausgänge kann sich die höchste Ableitung verändern, weswegen $\mathbf{z}^{(\hat{a})}$ anstatt $\mathbf{z}^{(a)}$ verwendet wird.

- die fiktiven Ausgänge sind unabhängig bzgl. Zeitverschiebungen, sowie Ableitungen, d.h.

$$Q(\mathbf{y}, \delta^{z_1} \delta^{-\tilde{z}_1} \dot{\mathbf{y}}, \delta^{z_2} \delta^{-\tilde{z}_2} \ddot{\mathbf{y}}, \dots, \delta^{z_{\hat{a}}} \delta^{-\tilde{z}_{\hat{a}}} \mathbf{y}^{(\hat{a})}) = 0 \quad (3.20)$$

wird von keinem \mathbf{y} mit $\hat{a} \geq 0$ der höchsten Ableitung in \mathbf{y} erfüllt.

- jede Systemvariable (inklusive deren Ableitungen) kann in Abhängigkeit vom fiktiven Ausgang, dessen Ableitungen, sowie positiven und negativen Zeitverschiebungen dargestellt werden:

$$z_i = R_i(\mathbf{y}, \delta^{z_1} \delta^{-\tilde{z}_1} \dot{\mathbf{y}}, \delta^{z_2} \delta^{-\tilde{z}_2} \ddot{\mathbf{y}}, \dots, \delta^{z_{\hat{a}}} \delta^{-\tilde{z}_{\hat{a}}} \mathbf{y}^{(\hat{a})}) \quad \forall i \in \{1, \dots, c\}. \quad (3.21)$$

3.2.2. Delta Flachheit für RoPa30

Am Antrieb werden die Drücke und Position gemessen, welche durch das "Safety-Modul" (siehe Abbildung 6.1) auf zulässige Werte und Fehler überprüft und anschließend kommuniziert werden, wodurch diese Werte verzögert in das System zurückgegeben werden. Da die Geschwindigkeit und Beschleunigung direkt aus der Position bestimmt werden, werden diese um dieselbe Zeit verzögert. Zudem werden die Stellgrößen (Hub v) ebenfalls Zeitverzögert verwendet, da Synchronisationswartezeiten vorhanden sind.

Es sind also zum Zeitpunkt t nur die vergangenen Zustände $(x_1(t - \tilde{\tau}_q), x_2(t - \tilde{\tau}_q), x_3(t - \tilde{\tau}_p), x_4(t - \tilde{\tau}_p), \mathbf{u}(t - \tau_v))$ bekannt. Der momentan im Roboter implementierte Regler berücksichtigt diese Totzeiten nicht, weswegen der Massenstrom lediglich auf vergangene Werte reagiert. Um diesen Fehler zu korrigieren, müssen die Zustände zum Zeitpunkt t vorhergesagt werden.

Zur einfacheren Schreibweise und Berechnungen werden die resultierenden Gesamt-Totzeiten verwendet: (siehe Abbildung 6.1)

$$\tau_q = \tau_v + \tilde{\tau}_q \quad (3.22)$$

$$\tau_p = \tau_v + \tilde{\tau}_p \quad (3.23)$$

$$(3.24)$$

Der Ausgang des Safety-Moduls

$$q_{\text{saf}}(t) = q(t - \tau_q) \quad (3.25)$$

$$p_{2,\text{saf}}(t) = p_2(t - \tau_p) \quad (3.26)$$

$$p_{4,\text{saf}}(t) = p_4(t - \tau_p) \quad (3.27)$$

ist verschoben um die Zeitkonstanten $\tau_q, \tau_p \geq 0$. Hieraus ergeben sich durch Differentiation und Umformungen die weiteren Systemgrößen

$$\dot{q}_{\text{saf}}(t) = \dot{q}(t - \tau_q) \quad (3.28)$$

$$\ddot{q}_{\text{saf}}(t) = \ddot{q}(t - \tau_q) \quad (3.29)$$

$$p_{m,\text{saf}}(t) = p_m(t - \tau_p). \quad (3.30)$$

Wähle den Zustandsvektor \mathbf{x}_{saf}

$$\mathbf{x}_{\text{saf}} = \begin{bmatrix} q_{\text{saf}} \\ \dot{q}_{\text{saf}} \\ p_{2,\text{saf}} \\ p_{4,\text{saf}} \end{bmatrix}. \quad (3.31)$$

Dann lautet die aus Gleichung (3.3) transformierte Systemdynamik als $\dot{\mathbf{x}}_{\text{saf}} = \mathbf{f}_{\text{saf}}(\mathbf{x}_{\text{saf}}, \mathbf{u})$

$$\begin{aligned} \dot{\mathbf{x}}_{\text{saf}}(t) = & \begin{bmatrix} x_{2,\text{saf}}(t) \\ \frac{1}{J} \left(\tilde{A}r_m(\tilde{x}_{4,\text{saf}}(t + \tau_p - \tau_q) - \tilde{x}_{3,\text{saf}}(t + \tau_p - \tau_q)) - \tau_f(x_{2,\text{saf}}(t)) \right) \\ - \frac{n\tilde{x}_{3,\text{saf}}(t)\dot{V}_2(x_{2,\text{saf}}(t + \tau_q - \tau_p))}{V_2(x_{1,\text{saf}}(t + \tau_q - \tau_p))} \\ - \frac{n\tilde{x}_{4,\text{saf}}(t)\dot{V}_4(x_{2,\text{saf}}(t + \tau_q - \tau_p))}{V_4(x_{1,\text{saf}}(t + \tau_q - \tau_p))} \end{bmatrix} \\ & + \begin{bmatrix} 0 \\ 0 \\ \frac{n\tilde{R}T}{V_2(x_{1,\text{saf}}(t + \tau_q - \tau_p))} \\ 0 \end{bmatrix} u_1(t - \tau_p) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{n\tilde{R}T}{V_4(x_{1,\text{saf}}(t + \tau_q - \tau_p))} \end{bmatrix} u_2(t - \tau_p). \end{aligned} \quad (3.32)$$

Hierbei sind \tilde{A} , \tilde{x}_3 , \tilde{x}_4 , \tilde{R} so gewählt, dass der Druck auf 1-7 bar normiert ist. Damit das System nicht akausal wird, müssen sowohl

$$\tau_p - \tau_q \leq 0, \quad (3.33)$$

als auch

$$\tau_q - \tau_p \leq 0. \quad (3.34)$$

Damit beide Aussagen wahr sind, muss

$$\tau_q = \tau_p. \quad (3.35)$$

Der δ -flache Ausgang wird als

$$\mathbf{y}(t) = \begin{bmatrix} q_{\text{saf}}(t) \\ p_{\text{m,saf}}(t) \end{bmatrix} \quad (3.36)$$

gewählt.

Definiere die Systemvariablen zu

$$\mathbf{z}(t) = \begin{bmatrix} q_{\text{saf}}(t) & \dot{q}_{\text{saf}}(t) & p_{2,\text{saf}}(t) & p_{4,\text{saf}}(t) & \dot{m}_2(t) & \dot{m}_4(t) \end{bmatrix}^T. \quad (3.37)$$

Nun wird das System auf δ -Flachheit untersucht:

- offensichtlich lässt sich \mathbf{y} durch \mathbf{z} darstellen
- durch Umformungen und Zeitverschiebungen ergibt sich

$$z_1(t) = y_1(t) \quad (3.38)$$

$$z_2(t) = \dot{y}_1(t) \quad (3.39)$$

$$z_3(t) = y_2(t) - \frac{J\ddot{y}_1(t + \tau_q - \tau_p) + \tau_f(\dot{y}_1(t + \tau_q - \tau_p))}{2Ar_m} \quad (3.40)$$

$$z_4(t) = y_2(t) + \frac{J\ddot{y}_1(t + \tau_q - \tau_p) + \tau_f(\dot{y}_1(t + \tau_q - \tau_p))}{2Ar_m} \quad (3.41)$$

$$\begin{aligned} z_5(t) = & \frac{\dot{V}_2(\dot{y}_1(t + \tau_q))}{2Ar_m RT} \left(2Ar_m y_2(t + \tau_p) - J\ddot{y}_1(t + \tau_q) - \tau_f(\dot{y}_1(t + \tau_q)) \right) \\ & + \frac{V_2(y_1(t + \tau_q))}{2Ar_m n RT} \left(2Ar_m \dot{y}_2(t + \tau_p) - J\ddot{y}_1(t + \tau_q) - \frac{d}{dt} \left(\tau_f(\dot{y}_1(t + \tau_q)) \right) \right) \end{aligned} \quad (3.42)$$

$$\begin{aligned} z_6(t) = & \frac{\dot{V}_4(\dot{y}_1(t + \tau_q))}{2Ar_m RT} \left(2Ar_m y_2(t + \tau_p) + J\ddot{y}_1(t + \tau_q) + \tau_f(\dot{y}_1(t + \tau_q)) \right) \\ & + \frac{V_4(y_1(t + \tau_q))}{2Ar_m n RT} \left(2Ar_m \dot{y}_2(t + \tau_p) + J\ddot{y}_1(t + \tau_q) + \frac{d}{dt} \left(\tau_f(\dot{y}_1(t + \tau_q)) \right) \right) \end{aligned} \quad (3.43)$$

Also lässt sich auch jede Systemvariable in Abhängigkeit der δ -flachen Ausgänge darstellen. Daraus folgt nach [Fliess et al. 1995] die Unabhängigkeit von \mathbf{y} bzgl. Zeitverschiebungen und Ableitungen.

Somit ist gezeigt, dass \mathbf{y} ein δ -flacher Ausgang ist.

Zum Folgen von Referenztrajektorien muss das System zunächst in Brunovsky-Normalform vorliegen. Hierfür muss

$$\begin{aligned}\dot{y}_{11} &= \dot{y}_1 = y_{12} \\ \dot{y}_{12} &= \ddot{y}_1 = y_{13} \\ \dot{y}_{13}(t + \tau_q + \tau_p) &= \ddot{\ddot{y}}_1(t + \tau_q + \tau_p) = \nu_1(t) \\ \dot{y}_2(t + \tau_p) &= \nu_2(t)\end{aligned}$$

mit dem neuen Eingang $\boldsymbol{\nu} \in \mathbb{R}^2$. Analog zum Fall ohne Totzeit ergeben sich zwei unabhängige Integratorketten. Dies eingesetzt in (3.38) ergibt die Eingänge

$$\begin{aligned}\dot{m}_2 &= \frac{\dot{V}_2(\dot{y}_1(t + \tau_q))}{2Ar_m RT} \left(2Ar_m y_2(t + \tau_p) - J\ddot{y}_1(t + \tau_q) - \tau_f(\dot{y}_1(t + \tau_q)) \right) \\ &\quad + \frac{V_2(y_1(t + \tau_q))}{2Ar_m nRT} \left(2Ar_m \nu_2(t + \tau_p) - J\nu_1(t + \tau_q) - \frac{d}{dt} \left(\tau_f(\dot{y}_1(t + \tau_q)) \right) \right) \\ \dot{m}_4 &= \frac{\dot{V}_4(\dot{y}_1(t + \tau_q))}{2Ar_m RT} \left(2Ar_m y_2(t + \tau_p) + J\ddot{y}_1(t + \tau_q) + \tau_f(\dot{y}_1(t + \tau_q)) \right) \\ &\quad + \frac{V_4(y_1(t + \tau_q))}{2Ar_m nRT} \left(2Ar_m \nu_2(t + \tau_p) + J\nu_1(t + \tau_q) + \frac{d}{dt} \left(\tau_f(\dot{y}_1(t + \tau_q)) \right) \right).\end{aligned}\quad (3.44)$$

Der Regler wird wie zuvor als PT3- bzw. PT1-Glied gewählt, um den Folgefehler zu stabilisieren

$$\nu_1(t) = \ddot{\ddot{y}}_{1,d}(t) + c_{12}(\ddot{\ddot{y}}_{1,d}(t) - \ddot{\ddot{y}}_1(t)) + c_{11}(\dot{\ddot{y}}_{1,d}(t) - \dot{\ddot{y}}_1(t)) + c_{10}(y_{1,d}(t) - y_1(t)) \quad (3.45)$$

$$\nu_2(t) = \dot{y}_{2,d}(t) + c_{20}(y_{2,d}(t) - y_2(t)) \quad (3.46)$$

mit den Referenztrajektorien für den Ausgang des Safetymoduls, d.h. zeitverzögerte Systemvariablen.

Die von dem Safety-Modul ausgegebenen Werte zum Zeitpunkt t sind noch bekannt, jedoch nicht zu den Zeitpunkten $t + \tau_p, t + \tau_q$, weswegen diese mithilfe einer Prädiktion bestimmt werden.

Mit Gleichung (3.44) reagiert der Massenstrom nun auf wieder in die Gegenwart prädierte Werte, und nicht mehr auf vergangene Werte. Somit werden die Totzeiten zusätzlich zum Systemverhalten invertiert.

Prädiktionsalgorithmus

Es gibt verschiedene Ansätze zur Prädiktion von $y_1(t + \tau_q), \dot{y}_1(t + \tau_q), \ddot{y}_1(t + \tau_q), y_2(t + \tau_p)$, welche für die Berechnung des gewünschten Massenstroms in (3.44) benötigt wird. Hier wird zunächst ein Algorithmus nach [Mounier, 1998] angewandt, welcher die linearisierte Systemdynamik verwendet.

Die Systemdynamik (3.32) $\dot{\mathbf{x}}_{\text{saf}}(t) = \mathbf{f}_{\text{saf}}(\mathbf{x}_{\text{saf}}(t), \mathbf{x}_{\text{saf}}(t + \tau_p - \tau_q), \mathbf{x}_{\text{saf}}(t + \tau_q - \tau_p), \mathbf{u}(t - \tau_p))$

wird um die Referenztrajektorie linearisiert. Hierzu wird zunächst die Taylorentwicklung um die Referenztrajektorie gebildet.

$$\begin{aligned}
\dot{\mathbf{x}}_{\text{saf}}(t) &= \mathbf{f}_{\text{saf}}(\mathbf{x}_{\text{saf,d}}(t), \mathbf{x}_{\text{saf,d}}(t + \tau_p - \tau_q), \mathbf{x}_{\text{saf,d}}(t + \tau_q - \tau_p), \mathbf{u}_d(t - \tau_p)) \\
&+ \left. \frac{\partial \mathbf{f}_{\text{saf}}}{\partial \mathbf{x}_{\text{saf}}(t)} \right|_{(\mathbf{x}_{\text{saf,d}}, \mathbf{u}_d)} (\mathbf{x}_{\text{saf}}(t) - \mathbf{x}_{\text{saf,d}}(t)) \\
&+ \left. \frac{\partial \mathbf{f}_{\text{saf}}}{\partial \mathbf{x}_{\text{saf}}(t + \tau_p - \tau_q)} \right|_{(\mathbf{x}_{\text{saf,d}}, \mathbf{u}_d)} (\mathbf{x}_{\text{saf}}(t + \tau_p - \tau_q) - \mathbf{x}_{\text{saf,d}}(t + \tau_p - \tau_q)) \\
&+ \left. \frac{\partial \mathbf{f}_{\text{saf}}}{\partial \mathbf{x}_{\text{saf}}(t + \tau_q - \tau_p)} \right|_{(\mathbf{x}_{\text{saf,d}}, \mathbf{u}_d)} (\mathbf{x}_{\text{saf}}(t + \tau_q - \tau_p) - \mathbf{x}_{\text{saf,d}}(t + \tau_q - \tau_p)) \\
&+ \left. \frac{\partial \mathbf{f}_{\text{saf}}}{\partial u_1(t - \tau_p)} \right|_{(\mathbf{x}_{\text{saf,d}}, \mathbf{u}_d)} (u_1(t - \tau_p) - u_{1,d}(t - \tau_p)) \\
&+ \left. \frac{\partial \mathbf{f}_{\text{saf}}}{\partial u_2(t - \tau_p)} \right|_{(\mathbf{x}_{\text{saf,d}}, \mathbf{u}_d)} (u_2(t - \tau_p) - u_{2,d}(t - \tau_p)) \\
&+ H.O.T.
\end{aligned} \tag{3.47}$$

Diese werden nun mit $\Delta \mathbf{x} := \mathbf{x}_{\text{saf}} - \mathbf{x}_{\text{saf,d}}$ und $\Delta \mathbf{u} := \mathbf{u} - \mathbf{u}_d$ zum linearen System

$$\Delta \dot{\mathbf{x}}(t) = \mathbf{A}_0 \Delta \mathbf{x}(t) + \mathbf{A}_1 \Delta \mathbf{x}(t + \tau_p - \tau_q) + \mathbf{A}_2 \Delta \mathbf{x}(t + \tau_q - \tau_p) + \mathbf{B}_1 \Delta u_1(t - \tau_p) + \mathbf{B}_2 \Delta u_2(t - \tau_p) \tag{3.48}$$

zusammengefasst.

Die Matrizen des linearisierten Systems werden entsprechend (3.47) zu

$$\mathbf{A}_0 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{1}{J} \frac{\partial \tau_f(x_{2,\text{saf,d}}(t))}{\partial x_{2,\text{saf}}(t)} & 0 & 0 \\ 0 & 0 & -\frac{n\dot{V}_2(x_{2,\text{saf,d}}(t+\tau_q-\tau_p))}{V_2(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} & 0 \\ 0 & 0 & 0 & -\frac{n\dot{V}_4(x_{2,\text{saf,d}}(t+\tau_q-\tau_p))}{V_4(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} \end{bmatrix} \quad (3.49a)$$

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{Ar_m}{J} & \frac{Ar_m}{J} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.49b)$$

$$\begin{aligned} \mathbf{A}_2 &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \left(-nx_{3,\text{saf,d}}(t)\dot{V}_2(x_{2,\text{saf,d}}(t+\tau_q-\tau_p)) + nRTu_{1,d}(t-\tau_p) \right) \frac{\partial(1/V_2(x_{1,\text{saf,d}}(t+\tau_q-\tau_p)))}{\partial x_{1,\text{saf}}(t+\tau_q-\tau_p)} & \frac{nx_{3,\text{saf,d}}(t)Ar_m}{V_2(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} \\ \left(-nx_{4,\text{saf,d}}(t)\dot{V}_4(x_{2,\text{saf,d}}(t+\tau_q-\tau_p)) + nRTu_{2,d}(t-\tau_p) \right) \frac{\partial(1/V_4(x_{1,\text{saf,d}}(t+\tau_q-\tau_p)))}{\partial x_{1,\text{saf}}(t+\tau_q-\tau_p)} & -\frac{nx_{4,\text{saf,d}}(t)Ar_m}{V_4(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \left(-nx_{3,\text{saf,d}}(t)\dot{V}_2(x_{2,\text{saf,d}}(t+\tau_q-\tau_p)) + nRTu_{1,d}(t-\tau_p) \right) \frac{Ar_m}{V_2^2(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} & \frac{nx_{3,\text{saf,d}}(t)Ar_m}{V_2(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} \\ \left(-nx_{4,\text{saf,d}}(t)\dot{V}_4(x_{2,\text{saf,d}}(t+\tau_q-\tau_p)) + nRTu_{2,d}(t-\tau_p) \right) \frac{-Ar_m}{V_4^2(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} & -\frac{nx_{4,\text{saf,d}}(t)Ar_m}{V_4(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} \end{bmatrix} \end{aligned} \quad (3.49c)$$

$$\mathbf{B}_1 = \begin{bmatrix} 0 \\ 0 \\ \frac{nRT}{V_2(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} \\ 0 \end{bmatrix} \quad (3.49d)$$

$$\mathbf{B}_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{nRT}{V_4(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))} \end{bmatrix} \quad (3.49e)$$

bestimmt.

Bei dem Prädiktionsalgorithmus handelt es sich um einen diskreten Algorithmus, welcher die Lösung des linearen Systems in jedem Zeitschritt $t_k = k\Delta t$ bestimmt. Hierfür wird die Approximation getroffen, dass der Eingang u_1, u_2 , sowie die Matrizen $\mathbf{A}_0, \mathbf{A}_1, \mathbf{A}_2, \mathbf{B}_1, \mathbf{B}_2$

in jedem Iterationsschritt stückweise konstant sind. Dann lautet die iterative Lösung

$$\begin{aligned}\hat{\Delta}x(t_{k+1}) = e^{\mathbf{A}_{0k}\Delta t}\hat{\Delta}x(t_k) + \mathbf{A}_{0k}^{-1}(e^{\mathbf{A}_{0k}\Delta t} - \mathbf{I}_{4 \times 4})\{ & \mathbf{A}_{1k}\hat{\Delta}x(t_k + \tau_p - \tau_q) \\ & + \mathbf{A}_{2k}\hat{\Delta}x(t_k + \tau_q - \tau_p) \\ & + \mathbf{B}_{1k}\Delta u_1(t_k - \tau_p) \\ & + \mathbf{B}_{2k}\Delta u_2(t_k - \tau_p)\}. \quad (3.50)\end{aligned}$$

Hier kennzeichnet $\hat{\cdot}$ die Prädiktion der Zustandsdifferenzen. Die Iteration wird so lange fortgeführt, bis die Werte $y_1(t + \tau_q)$, $\dot{y}_1(t + \tau_q)$, $\ddot{y}_1(t + \tau_q)$, $y_2(t + \tau_p)$ erreicht werden.

Aus (3.50) kann nun die Prädiktion

$$\begin{bmatrix} \hat{y}_1(t + \tau_q) \\ \hat{\dot{y}}_1(t + \tau_q) \\ \hat{\ddot{y}}_1(t + \tau_q) \\ \hat{y}_2(t + \tau_p) \end{bmatrix} = \begin{bmatrix} \hat{x}_{1,\text{saf}}(t_k + \tau_q) \\ \hat{x}_{2,\text{saf}}(t_k + \tau_q) \\ \frac{1}{J} \left(Ar_m(\hat{x}_{4,\text{saf}}(t + \tau_p) - \hat{x}_{3,\text{saf}}(t + \tau_p)) - \tau_f(\hat{x}_{2,\text{saf}}(t + \tau_q)) \right) \\ \frac{\hat{x}_3(t_k + \tau_p) + \hat{x}_4(t_k + \tau_p)}{2} \end{bmatrix} \quad (3.51)$$

mit

$$\hat{x}_{saf} = \hat{\Delta}x_{saf} + \mathbf{x}_{\text{saf},d} \quad (3.52)$$

bestimmt werden.

Auch wenn das System akausal ist, wenn $\tau_q \neq \tau_p$, kann eine Prädiktion erfolgen, da die Zeitverschiebungen $t + \tau_q - \tau_p$ und $t + \tau_p - \tau_q$ der linearisierten Matrizen entkoppelt sind. Dies lässt sich am besten erkennen, wenn Gleichung (3.50) in Blockmatrizen geschrieben wird. Hierfür werden zunächst die Matrizen (3.49) in Blockdiagonalmatrizen umgeschrieben

$$\mathbf{A}_0 = \begin{bmatrix} \mathbf{A}_{0Q}(x_{2,\text{saf},d}(\textcolor{teal}{t})) & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{0P}(x_{1,\text{saf},d}(\textcolor{red}{t} + \tau_q - \tau_p), x_{2,\text{saf},d}(\textcolor{red}{t} + \tau_q - \tau_p)) \end{bmatrix} \quad (3.53a)$$

$$\mathbf{A}_1 = \begin{bmatrix} \mathbf{0} & \mathbf{A}_{1P} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (3.53b)$$

$$\mathbf{A}_2 = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{A}_{2Q}(x_{1,\text{saf},d}(\textcolor{red}{t} + \tau_q - \tau_p), x_{2,\text{saf},d}(\textcolor{red}{t} + \tau_q - \tau_p), x_{3,\text{saf},d}(\textcolor{teal}{t}), x_{4,\text{saf},d}(\textcolor{teal}{t}), u_{1,d}(\textcolor{brown}{t} - \tau_p), u_{2,d}(\textcolor{brown}{t} - \tau_p)) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (3.53c)$$

$$\mathbf{B}_1 = \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_{1P}(x_{1,\text{saf},d}(\textcolor{red}{t} + \tau_q - \tau_p)) \end{bmatrix} \quad (3.53d)$$

$$\mathbf{B}_2 = \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_{2P}(x_{1,\text{saf},d}(\textcolor{red}{t} + \tau_q - \tau_p)) \end{bmatrix} \quad (3.53e)$$

wobei die Indizes P, Q kennzeichnen, ob die zugehörige Blockmatrix durch den Druck (P), oder die Position/ Geschwindigkeit (Q) beeinflusst wird.

Die Lösung des linearen Systems (3.50) lässt sich nun mit $\Delta Q = [\Delta x_{1,saf}, \Delta x_{1,saf}]^T$, $\Delta P = [\Delta x_{3,saf}, \Delta x_{4,saf}]^T$ umschreiben in

$$\begin{aligned} \begin{bmatrix} \hat{\Delta Q}(t_{k+1}) \\ \hat{\Delta P}(t_{k+1}) \end{bmatrix} &= \begin{bmatrix} e^{\mathbf{A}_{0Q,k}\Delta t} & \mathbf{0} \\ \mathbf{0} & e^{\mathbf{A}_{0P,k}\Delta t} \end{bmatrix} \begin{bmatrix} \hat{\Delta Q}(t_k) \\ \hat{\Delta P}(t_k) \end{bmatrix} + \begin{bmatrix} \mathbf{A}_{0Q}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{0P}^{-1} \end{bmatrix} \begin{bmatrix} e^{\mathbf{A}_{0Q,k}\Delta t} - \mathbf{I}_{2 \times 2} & \mathbf{0} \\ \mathbf{0} & e^{\mathbf{A}_{0P,k}\Delta t} - \mathbf{I}_{2 \times 2} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{0} & \mathbf{A}_{1P} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \hat{\Delta Q}(t_k + \tau_p - \tau_q) \\ \hat{\Delta P}(t_k + \tau_p - \tau_q) \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{A}_{2Q} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \hat{\Delta Q}(t_k + \tau_q - \tau_p) \\ \hat{\Delta P}(t_k + \tau_q - \tau_p) \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_{1P} \end{bmatrix} \Delta u_1(t_k - \tau_p) \\ &+ \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_{2P} \end{bmatrix} \Delta u_2(t_k - \tau_p) \end{aligned} \quad (3.54a)$$

$$\begin{aligned} &= \begin{bmatrix} e^{\mathbf{A}_{0Q,k}\Delta t} \hat{\Delta Q}(t_k) \\ e^{\mathbf{A}_{0P,k}\Delta t} \hat{\Delta P}(t_k) \end{bmatrix} + \begin{bmatrix} \mathbf{A}_{0Q}^{-1}(e^{\mathbf{A}_{0Q,k}\Delta t} - \mathbf{I}_{2 \times 2}) & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{0P}^{-1}(e^{\mathbf{A}_{0P,k}\Delta t} - \mathbf{I}_{2 \times 2}) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{A}_{1P} \hat{\Delta P}(t_k + \tau_p - \tau_q) \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{A}_{2Q} \hat{\Delta Q}(t_k + \tau_q - \tau_p) \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_{1P} \Delta u_1(t_k - \tau_p) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_{2P} \Delta u_2(t_k - \tau_p) \end{bmatrix} \end{aligned} \quad (3.54b)$$

$$\begin{aligned} &= \begin{bmatrix} e^{\mathbf{A}_{0Q,k}\Delta t} \hat{\Delta Q}(t_k) \\ e^{\mathbf{A}_{0P,k}\Delta t} \hat{\Delta P}(t_k) \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{A}_{0Q}^{-1}(e^{\mathbf{A}_{0Q,k}\Delta t} - \mathbf{I}_{2 \times 2}) \mathbf{A}_{1P} \hat{\Delta P}(t_k + \tau_p - \tau_q) \\ \mathbf{A}_{0P}^{-1}(e^{\mathbf{A}_{0P,k}\Delta t} - \mathbf{I}_{2 \times 2}) (\mathbf{A}_{2Q} \hat{\Delta Q}(t_k + \tau_q - \tau_p) + \mathbf{B}_{1P} \Delta u_1(t_k - \tau_p) + \mathbf{B}_{2P} \Delta u_2(t_k - \tau_p)) \end{bmatrix} \end{aligned} \quad (3.54c)$$

$$= \begin{bmatrix} fkt_1(x_{1,saf}(t_k), x_{2,saf}(t_k), p_{2,saf}(t_k + \tau_p - \tau_q), p_{4,saf}(t_k + \tau_p - \tau_q)) \\ fkt_2(x_{1,saf}(t_k + \tau_q - \tau_p), x_{2,saf}(t_k + \tau_q - \tau_p), p_{2,saf}(t_k), p_{4,saf}(t_k), u_1(t_k - \tau_p), u_2(t_k - \tau_p)) \end{bmatrix} \quad (3.54d)$$

Die hellgrün und olivgrün markierten Einträge in (3.53) und (3.54) sind zum Zeitpunkt t_k bekannte Größen, weswegen diese keine Probleme darstellen. Die roten und blauen Größen hingegen beschreiben zukünftige, d.h. unbekannte, Größen. Die Matrizen spielen hierbei keine Rolle, da diese nur von der gegebenen Referenztrajektorie abhängen, welche

bereits zu Beginn der Simulation für alle Zeitschritte bekannt ist. Es können jedoch immer nur entweder die **roten**, oder nur die **blauen** Größen unbekannt sein. D.h.

$$t_k + \tau_q - \tau_p \begin{cases} < t_k \text{ if } \tau_q < \tau_p \\ = t_k \text{ if } \tau_q = \tau_p \\ > t_k \text{ if } \tau_q > \tau_p \end{cases} \quad (3.55a)$$

$$t_k + \tau_p - \tau_q \begin{cases} > t_k \text{ if } \tau_q < \tau_p \\ = t_k \text{ if } \tau_q = \tau_p \\ < t_k \text{ if } \tau_q > \tau_p \end{cases} \quad (3.55b)$$

Es gilt entweder $t_k + \tau_q - \tau_p \leq t_k$, dann liegen die zu diesem Zeitpunkt zugehörigen Werte in der Vergangenheit oder in der Gegenwart und sind verfügbar, während die **blauen** Größen $t_k + \tau_p - \tau_q > 0$ prädiziert werden müssen, da diese in der Zukunft liegen.

Oder es gilt $t_k + \tau_q - \tau_p > t_k$, dann liegen die zugehörigen Größen in der Zukunft und müssen prädiziert werden, während jedoch die Größen zum Zeitpunkt $t_k + \tau_p - \tau_q \leq 0$ bekannt sind.

Ein Blick auf Gleichung (3.54d) zeigt, dass die erste Zeile lediglich von Größen zu den Zeitpunkten t_k und $t_k + \tau_p - \tau_q$ abhängt, während die zweite Zeile nur von Größen zu den Zeitpunkten t_k , $t_k - \tau_p$ und $t_k + \tau_q - \tau_p$ abhängt. D.h. die Zeitverschiebungen $t_k + \tau_q - \tau_p$ und $t_k + \tau_p - \tau_q$ sind entkoppelt. Die Zeitverzögerung im Druck wirkt nur auf die Druckdynamik, während die Zeitverzögerung in der Position nur auf die mechanische Bewegungsgleichung wirkt.

Weiterhin werden die Zeitverschiebungen um $t_k + \tau_q - \tau_p$ in (3.54d) nur auf $x_{1,\text{saf}}$ und $x_{2,\text{saf}}$ angewendet, während $t_k + \tau_p - \tau_q$ nur auf die Drücke $x_{3,\text{saf}}$ und $x_{4,\text{saf}}$ wirkt.

Mit den zuvor gemachten Überlegungen ist also entweder

- $t_k + \tau_q - \tau_p < t_k$ und $t_k + \tau_p - \tau_q > t_k$ ($\tau_q < \tau_p$):
Alle **rot**, **grün** und **olivgrün** markierten Größen in (3.54d) sind bekannt. Lediglich die Drücke $p_{2,\text{saf}}(t_k + \tau_p - \tau_q)$ und $p_{4,\text{saf}}(t_k + \tau_p - \tau_q)$ liegen in der Zukunft und müssen zunächst selbst prädiziert werden.

Das Vorhersagen der Drücke ist möglich, da im ersten Zeitschritt $\hat{\Delta}P(t_{k+1})$ mit der zweiten Zeile bestimmt wird, welche nur von vergangenen Zeitverschiebungen abhängt.

Wenn schließlich $\hat{\Delta}P(t_{k+1})$ berechnet wurde, so kann dies im nächsten Iterationsschritt weiterverwendet werden.

Da $\tau_q < \tau_p$, ist ebenso $q_{\text{saf}}(t_k + \tau_q - \tau_p)$ bis inklusive zum Zeitpunkt $q_{\text{saf}}(t_k)$ bekannt. (analog für \dot{q}_{saf}). Die Iteration ist beendet, wenn keine Positionswerte mehr bekannt sind, dann sind alle nötigen Druckwerte berechnet, welche zur Positionsprädiktion benötigt werden.

Pseudocode zum Zeitpunkt t :

Wähle die Iterationszeitkonstante Δt klein genug.

1: *gegeben* : $Q(0...t), P(0...t), u(0...t), x_{\text{saf},d}(0...end)$

```

2:  $\tau_\Delta = \tau_p - \tau_q (> 0)$ 
3:                                      $\triangleright$  Predict  $P$  first
4:  $\hat{\Delta}P(t + \Delta t) \leftarrow fkt_2(\Delta P(t), \Delta Q(t - \tau_\Delta \Delta t), \Delta u(t - \tau_p \Delta t))$ 
5:  $\hat{\Delta}P(t + 2\Delta t) \leftarrow fkt_2(\hat{\Delta}P(t + \Delta t), \Delta Q(t + (-\tau_\Delta + 1)\Delta t), \Delta u(t + (-\tau_p + 1)\Delta t))$ 
6:  $\hat{\Delta}P(t + 3\Delta t) \leftarrow fkt_2(\hat{\Delta}P(t + 2\Delta t), \Delta Q(t + (-\tau_\Delta + 2)\Delta t), \Delta u(t + (-\tau_p + 2)\Delta t))$ 
7:  $\vdots$ 
8:  $\hat{\Delta}P(t + \tau_\Delta \Delta t) \leftarrow fkt_2(\hat{\Delta}P(t + (\tau_\Delta - 1)\Delta t), \Delta Q(t - \Delta t), \Delta u(t + (-\tau_q - 1)\Delta t))$ 
9:                                      $\triangleright$  Predict  $Q$  next
10:  $\hat{\Delta}Q(t + \Delta t) \leftarrow fkt_1(\Delta Q(t), \hat{\Delta}P(t + \tau_\Delta \Delta t))$ 
11:                                      $\triangleright$  Predict  $P$  again
12:  $\hat{\Delta}P(t + (\tau_\Delta + 1)\Delta t) \leftarrow fkt_2(\hat{\Delta}P(t + \tau_\Delta \Delta t), \Delta Q(t), \Delta u(t - \tau_q \Delta t))$ 
13:                                      $\triangleright$  Predict  $Q$  again
14:  $\hat{\Delta}Q(t + 2\Delta t) \leftarrow fkt_1(\hat{\Delta}Q(t + \Delta t), \hat{\Delta}P(t + (\tau_\Delta + 1)\Delta t))$ 
15:                                      $\triangleright$  Predict  $P$  again
16:  $\hat{\Delta}P(t + (\tau_\Delta + 2)\Delta t) \leftarrow fkt_2(\hat{\Delta}P(t + (\tau_\Delta + 1)\Delta t), \hat{\Delta}Q(t + \Delta t), \Delta u(t + (-\tau_q + 1)\Delta t))$ 
17:                                      $\triangleright$  Predict  $Q$  again
18:  $\hat{\Delta}Q(t + 3\Delta t) \leftarrow fkt_1(\hat{\Delta}Q(t + 2\Delta t), \hat{\Delta}P(t + (\tau_\Delta + 2)\Delta t))$ 
19:                                      $\triangleright$  Repeat lines 16 and 18 over and over
20:  $\vdots$ 
21:                                      $\triangleright$  Predict final  $Q$  and  $P$ 
22:  $\hat{\Delta}P(t + \tau_p \Delta t) \leftarrow fkt_2(\hat{\Delta}P(t + (\tau_p - 1)\Delta t), \hat{\Delta}Q(t + (\tau_q - 1)\Delta t), \Delta u(t - \Delta t))$ 
23:  $\hat{\Delta}Q(t + \tau_q \Delta t) \leftarrow fkt_1(\hat{\Delta}Q(t + (\tau_q - 1)\Delta t), \hat{\Delta}P(t + (\tau_p - 1)\Delta t))$ 

```

In den Zeilen 4 bis 8 wird die Differenz der Druck- und Positionsverschiebungen kompensiert. Danach werden die Zeitschritte bis $P(t + \tau_\Delta + \tau_q) = P(t + \tau_p)$ und $Q(t + \tau_q)$ iterierend prädiziert. Es sind zu jedem Zeitschritt alle benötigten Werte bekannt.

- $t_k + \tau_q - \tau_p > t_k$ und $t_k + \tau_p - \tau_q < t_k$ ($\tau_q > \tau_p$): Dieser Fall funktioniert analog zum vorherigen Fall. Es werden lediglich zuerst die Positionen prädiziert, und anschließend die Drücke. Zudem ist die Definition der Differenzzeitverschiebung τ_Δ verändert, sodass $\tau_\Delta > 0$ immer gilt.

Pseudocode zum Zeitpunkt t :

```

1: gegeben :  $Q(0...t), P(0...t), u(0...t)$ 
2:  $\tau_\Delta = -\tau_p + \tau_q (> 0)$ 
3:                                      $\triangleright$  Predict  $Q$  first
4:  $\hat{\Delta}Q(t + \Delta t) \leftarrow fkt_1(\Delta Q(t), \Delta P(t - \tau_\Delta \Delta t))$ 
5:  $\hat{\Delta}Q(t + 2\Delta t) \leftarrow fkt_1(\hat{\Delta}Q(t + \Delta t), \Delta P(t + (-\tau_\Delta + 1)\Delta t))$ 
6:  $\hat{\Delta}Q(t + 3\Delta t) \leftarrow fkt_1(\hat{\Delta}Q(t + 2\Delta t), \Delta P(t + (-\tau_\Delta + 2)\Delta t))$ 
7:  $\vdots$ 
8:  $\hat{\Delta}Q(t + \tau_\Delta \Delta t) \leftarrow fkt_1(\hat{\Delta}Q(t + (\tau_\Delta - 1)\Delta t), \Delta P(t - \Delta t))$ 
9:                                      $\triangleright$  Predict  $P$  next
10:  $\hat{\Delta}P(t + \Delta t) \leftarrow fkt_2(\Delta P(t), \hat{\Delta}Q(t + \tau_\Delta \Delta t), \Delta u(t - \tau_p \Delta t))$ 

```

11: ▷ Predict Q again
 12: $\hat{\Delta}Q(t + (\tau_\Delta + 1)\Delta t) \leftarrow fkt_1(\hat{\Delta}Q(t + \tau_\Delta \Delta t), \Delta P(t))$
 13: ▷ Predict P again
 14: $\hat{\Delta}P(t + 2\Delta t) \leftarrow fkt_2(\hat{\Delta}P(t + \Delta t), \hat{\Delta}Q(t + (\tau_\Delta + 1)\Delta t), \Delta u(t + (-\tau_p + 1)\Delta t))$
 15: ▷ Predict Q again
 16: $\hat{\Delta}Q(t + (\tau_\Delta + 2)\Delta t) \leftarrow fkt_1(\hat{\Delta}Q(t + (\tau_\Delta + 1)\Delta t), \hat{\Delta}P(t + \Delta t))$
 17: ▷ Predict P again
 18: $\hat{\Delta}P(t + 3\Delta t) \leftarrow fkt_2(\hat{\Delta}P(t + 2\Delta t), \hat{\Delta}Q(t + (\tau_\Delta + 2)\Delta t), \Delta u(t + (-\tau_p + 2)\Delta t))$
 19: ▷ Repeat lines ?? and ?? over and over
 20: \vdots
 21: ▷ Predict final Q and P
 22: $\hat{\Delta}Q(t + \tau_q \Delta t) \leftarrow fkt_1(\hat{\Delta}Q(t + (\tau_q - 1)\Delta t), \hat{\Delta}P(t + (\tau_p - 1)\Delta t))$
 23: $\hat{\Delta}P(t + \tau_p \Delta t) \leftarrow fkt_2(\hat{\Delta}P(t + (\tau_p - 1)\Delta t), \hat{\Delta}Q(t + (\tau_q - 1)\Delta t), \Delta u(t - \Delta t))$

Von Zeile 4 bis 8 wird die Differenz zwischen der Druckverschiebung und der Positionsverschiebung kompensiert. Anschließend werden die Prädiktionen von Druck und Position so lange alterniert, bis $Q(t + \tau_q \Delta t)$ und $P(t + \tau_p \Delta t)$ erreicht sind.

- $t_k + \tau_q - \tau_p = t_k$ und $t_k + \tau_p - \tau_q = t_k$ ($\tau_q = \tau_p$):
Trivial.

Parameter	Zugehörigkeit	Wert	Einheit
Druckquelle p_s	Ventil	$7 \cdot 10^5$	Pa
Umgebungsdruck p_0	Ventil	$1 \cdot 10^5$	Pa
max. Leitwert C_{\max}	Ventil	$1.5588 \cdot 10^{-9}$	$\text{m}^3/\text{Pa/s}$
max. kritisches Druckverhältnis b_{\max}	Ventil	0.5	—
Druck-Totzone p_{tot}	Ventil	0	Pa
effektiver Zylinder-Querschnitt A	Antrieb	0.00167	m^2
effektiver Zylinder-Radius r_m	Antrieb	0.035	m
Totvolumen V_{tot}	Antrieb	$1.795 \cdot 10^{-5}$	m^3
min. Zylinderposition φ_{\min}	Antrieb	-2.3736	rad
max. Zylinderposition φ_{\max}	Antrieb	2.3736	rad
spezifische Gaskonstante R	Druckkammern	287	J/kg/K
Umgebungstemperatur T_0	Druckkammern	293	K
polytropischer Exponent κ	Druckkammern	1.25	—
Trägheitsmoment Zylinder J	Mechanik	$1.04 \cdot 10^{-4}$	kgm^2
Coulombreibungskoeffizient f_c	Mechanik	1.28	Nm
Koeffizient statische Reibung f_s	Mechanik	1.28	Nm
viskoser Reibungskoeffizient f_c	Mechanik	0.02	$\text{Nm}/(\text{rad/s})$
Parameter für glatte tanh-Funktion \dot{q}_c	Mechanik	0.1	rad/s

Tabelle 4.1.: Parameterwerte des RoPA30-Schwenkantriebs.

Datenset	Parameter	Wert	Einheit
Flacher Regler ohne Delays (\mathcal{D}_1)	Regelparameter T_1	0.039	s
	Regelparameter D_1	600	—
	Druckregler T_2	0.5	s
Flacher Regler mit Delays und Prädiktion (\mathcal{D}_2)	Regelparameter T_1	0.1	s
	Regelparameter D_1	2000	—
	Druckregler T_2	0.15	s
Flacher Regler mit Delays ohne Prädiktion (\mathcal{D}_3)	Regelparameter T_1	0.5	s
	Regelparameter D_1	39	—
	Druckregler T_2	0.5	s

Tabelle 4.2.: Datensets für den flachen Regler für den RoPA30-Schwenkantrieb.

5. Flachheitsbasierte Regelung des RoPa30 Antriebs

Hier noch ohne Totzeit, nur proof of concept.

5.1. Simulationsergebnisse

Zur Implementierung und Bestimmung der einzelnen Ventileingänge, wird zunächst der Massenstrom aufgespaltet:

$$\dot{m}_{i,ae} = \begin{cases} \dot{m}_i & , \dot{m}_i > 0 \\ 0 & , \dot{m}_i \leq 0 \end{cases} \quad (5.1)$$

$$\dot{m}_{i,de} = \begin{cases} \dot{m}_i & , \dot{m}_i < 0 \\ 0 & , \dot{m}_i \geq 0 \end{cases} \quad (5.2)$$

Dann folgt aus Gleichung (3.15)

$$C_{rel,i,ae,des} = \frac{\dot{m}_{i,ae,des}}{C_{max} p_{prim,i} \rho \Psi\left(b, \frac{p_{sec,i}}{p_{prim,i}}\right)} > 0 \quad (5.3)$$

$$C_{rel,i,de,des} = \frac{\dot{m}_{i,de,des}}{C_{max} p_{prim,i} \rho \Psi\left(b, \frac{p_{sec,i}}{p_{prim,i}}\right)} < 0. \quad (5.4)$$

Da in dieser Arbeit der pneumatische Leitwert als eindimensionale Funktion beschrieben wird, kann die lookup-Tabelle verwendet werden, um v_i aus C_i zu bestimmen. Somit ergeben sich die Öffnungs-Eingänge zu

$$v_{i,ae} = C_{rel}^{-1}(C_{rel,i,ae,des}) > 0 \quad (5.5)$$

$$v_{i,de} = C_{rel}^{-1}(C_{rel,i,de,des}) < 0. \quad (5.6)$$

Die Modellinversion wird auf die Strecke gegeben, wie in Abbildung 5.1 abgebildet. Das RoPA30-Modell beinhaltet einen Zustandsvariablenfilter, welcher die zugehörigen Geschwindigkeiten und Beschleunigungen aus der gemessenen Position bestimmt.

Die partielle Ableitung der Reibung, welche in (2.2) definiert wurde,

$$\frac{\partial \tau_f}{\partial \dot{q}} = f_v + 4 \frac{f_c e^{\frac{2\dot{q}}{\dot{q}_c}}}{\dot{q}_c (e^{\frac{2\dot{q}}{\dot{q}_c}} + 1)^2} \quad (5.7)$$

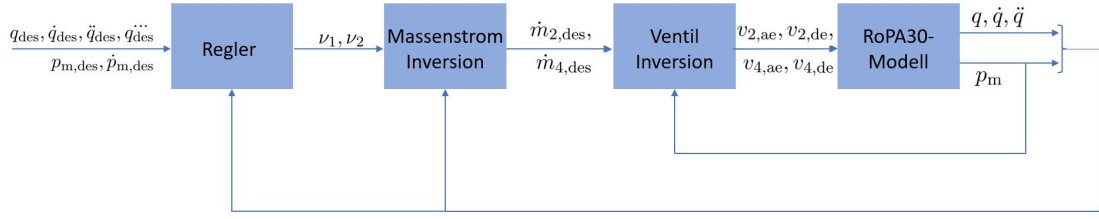


Abbildung 5.1.: Simulationsmodell des flachheitsbasierten Reglers für den RoPA30-Antrieb.

in Abbildung 5.2 führt für kleine Geschwindigkeiten aufgrund des Peaks zu numerischen Problemen. Weiterhin wird für große positive Geschwindigkeiten der Grenzwert

$$\lim_{\dot{q} \rightarrow \infty} \frac{\partial \tau_f}{\partial \dot{q}} = \lim_{a \rightarrow \infty} f_v + 4 \frac{f_c a}{\dot{q}_c (a + 1)^2} \quad (5.8a)$$

$$\stackrel{l'Hopital}{=} \lim_{a \rightarrow \infty} f_v + 4 \frac{f_c}{\dot{q}_c (2a + 2)} \quad (5.8b)$$

$$= f_v \quad (5.8c)$$

von Matlab nicht berechnet, sondern 'Nan' ausgegeben. Aus diesen Gründen wird diese als

$$\frac{\partial \tau_f}{\partial \dot{q}} = f_v \quad (5.9)$$

approximiert. Es wird also der Coulomb-Anteil in der Ableitung vernachlässigt. Damit ist die absolute Zeitableitung

$$\frac{d\tau_f}{dt} = \frac{\partial \tau_f}{\partial \dot{q}} \frac{\partial \dot{q}}{\partial t} = f_v \ddot{q}. \quad (5.10)$$

5.1.1. Test inverses Modell

Verwende Gleichung (3.6), um zu testen, ob die Invertierung im Simulinkmodell korrekt ist. Gebe hierfür eine realistische Trajektorie vor. Dies ist z.B. ein PT3-gefilterter Einheitssprung für q_{des} , \dot{q}_{des} , \ddot{q}_{des} , $\dot{\ddot{q}}_{des}$, sowie

$$p_{m,des}(t) = 3.5e5 + 0.5e5 \sin(t) \quad (5.11)$$

$$\dot{p}_{m,des}(t) = 0.5e5 \cos(t) \quad (5.12)$$

für den Mitteldruck. Hierbei ist noch kein Regler vorhanden.

Beachte insbesondere, dass die Anfangswerte im PT3-Block richtig gewählt sind.

Diese Referenzgrößen werden nun auf das System gegeben und mit den ausgegebenen Werten verglichen. Diese müssen mit den Referenztrajektorien übereinstimmen.

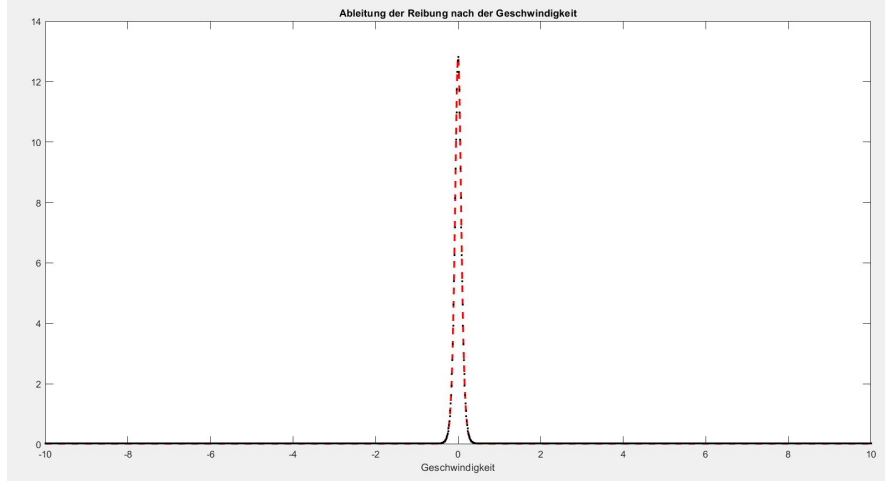


Abbildung 5.2.: Partielle Ableitung $\frac{\partial \tau_f}{\partial \dot{q}}$ des Reibmoments.

5.2. Regelergebnisse

Der Regler (3.13) wird für die Position als in Reihe geschaltetes PT1 und PT2-Glieder mit derselben Zeitkonstanten und gain $K = 1$ verwendet. Die Übertragungsfunktion lautet

$$\begin{aligned} G_{PT1} &= \frac{K_{11}}{1 + T_1 s} \\ G_{PT2} &= \frac{K_{12}}{T_1^2 s^2 + 2DT_1 s + 1} \end{aligned} \quad (5.13)$$

mit der Dämpfung D . Das charakteristische Polynom ergibt sich aus dem Nenner zu

$$p_1(s) = (1 + T_1 s)(T_1^2 s^2 + 2DT_1 s + 1) = T_1^3 s^3 + (1 + 2D)T_1^2 s^2 + (1 + 2D)T_1 s + 1 \quad (5.14)$$

Koeffizientenvergleich von (5.14) und (3.13) ergibt

$$\begin{aligned} c_{10} &= \frac{1}{T_1^3} \\ c_{11} &= \frac{2D + 1}{T_1^2} \\ c_{12} &= \frac{2D + 1}{T_1} \end{aligned} \quad (5.15)$$

Der Druckregler hingegen ist ein einfacher PT1-Regler, ebenfalls mit Verstärkung 1

$$G_{PT1} = \frac{K_2}{1 + T_2 s}. \quad (5.16)$$

Dann ergibt sich ebenfalls durch Koeffizientenvergleich

$$c_{20} = \frac{1}{T_2} \quad (5.17)$$

5.2.1. Regelergebnisse

Die Position wird als PT3-gesfiltertes Sinussignal vorgegeben:

$$\varphi(t) = \sin(t) \quad (5.18)$$

$$\dot{\varphi}(t) = \cos(t) \quad (5.19)$$

$$\ddot{\varphi}(t) = -\sin(t) \quad (5.20)$$

$$\ddot{\varphi}(t) = -\cos(t), \quad (5.21)$$

wie in Abbildung 5.3 zu sehen. Der Mitteldruck wird konstant als $p_m = 3.5 \cdot 10^5 \text{Pa}$ vor-

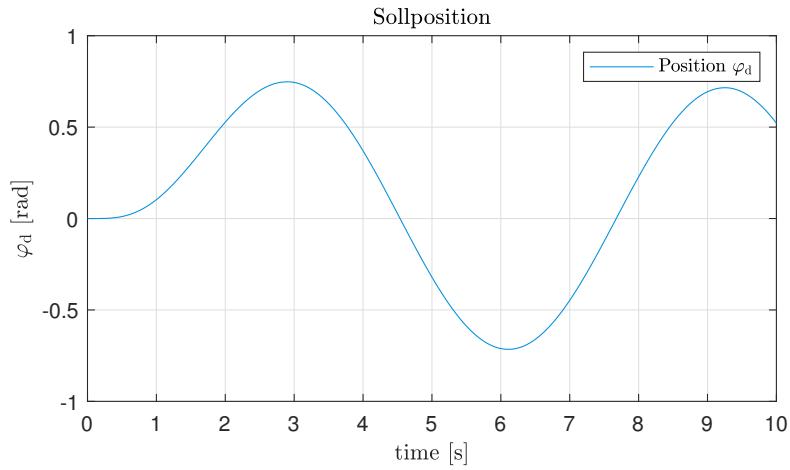


Abbildung 5.3.: Gewünschter Positionsverlauf φ_d (rad). [Aus ro-Pa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_1]

gegeben.

Hiermit ergibt sich der Sollruckverlauf in Abbildung 5.4.

Der Positionsfehler ist klein in der Größenordnung 10^{-4}rad , d.h. klein genug.

Je kleiner die Zeitkonstanten, oder je größer der Dämpfungsparameter, desto kleiner wird der Fehler.

Der Druckfehler ist in Abbildung 5.6 abgebildet. Er beträgt maximal $100 \text{Pa} = 0.001 \text{bar} = 1 \text{mbar}$, was vernachlässigbar klein ist. Der Mitteldruck folgt mit einer absoluten Genauigkeit von $5 \text{Pa} = 0.05 \text{mbar}$

Beim Massenstrom in Abbildung 5.7 ist ein Peak im simulativen Massenstrom zu sehen, wenn dieser die 0 durchläuft. Hierbei handelt es sich um die Ungenauigkeiten durch die Approximation der Reibeabweichung. Der Peak in Abbildung 5.2 wird jedes mal bei

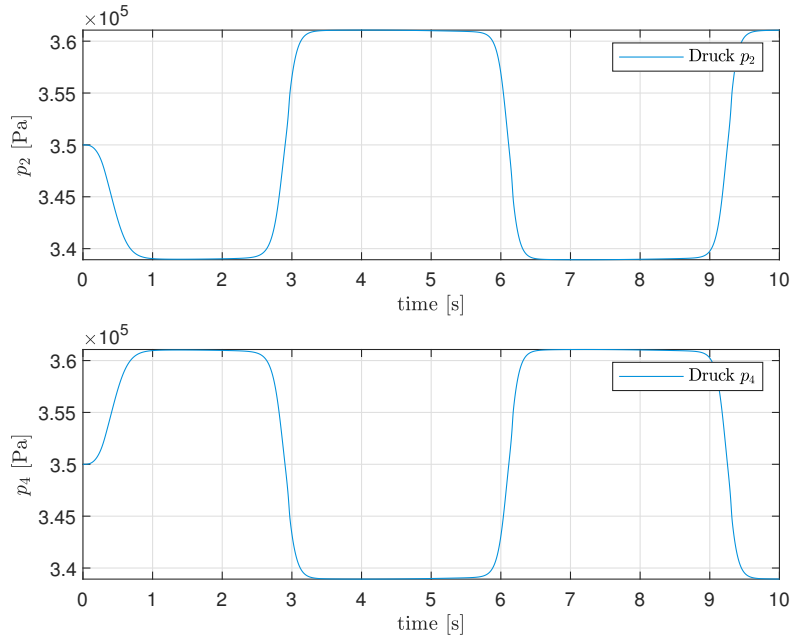


Abbildung 5.4.: Solldruckverlauf p_2, p_4 (Pa). [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_1]

$\dot{\varphi} = 0$ im Massenstrom benötigt, um der Sollposition und -druck zu folgen. Diese Peaks sind sichtbar, weil sie durch das reale System nicht ausgeglichen werden (im Vergleich zur nicht approximierten Reibungsableitung).

5.2.2. Test auf Robustheit des Reglers

Führe Ungenauigkeiten im Modell ein, um den Regler auf Robustheit zu testen, da nicht alle Parameter des realen Systems exakt bekannt sind.

- Im realen System wird nur die Position gemessen. Deswegen wird ein Zustandsvariablenfilter (Alternative: reduzierter Beobachter) auf die ausgegebene Position angewendet, anstatt \dot{q}, \ddot{q} aus dem FOSIL-Mechanik-Block zu verwenden.
- Ungenaue Parameter: Ändere die Coulomb- und viskose Reibung, Trägheit, polytropher Exponent, Totvolumen, effektiver Radius und Fläche um $\pm 20\%$

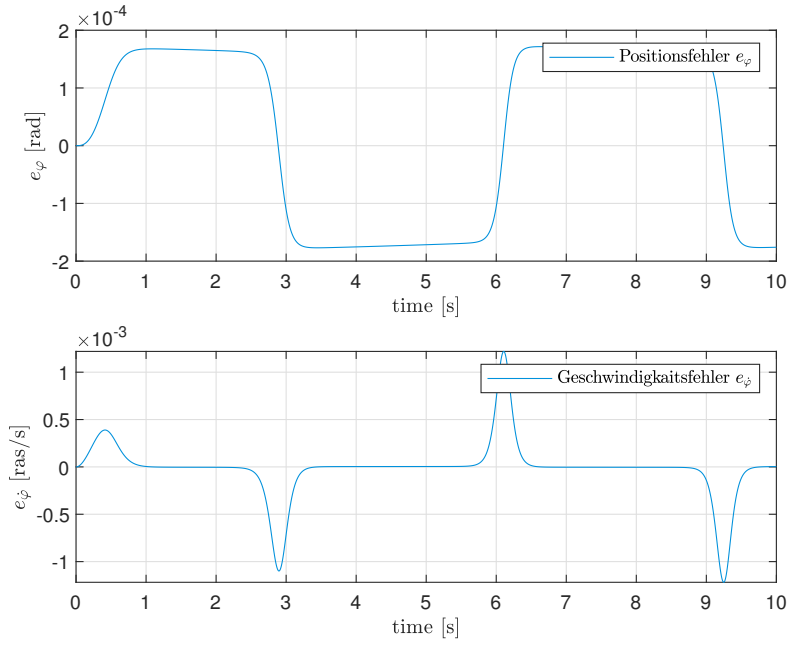


Abbildung 5.5.: Oben Positions- und unten Geschwindigkeitsfehler. [Aus ro-Pa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_1]

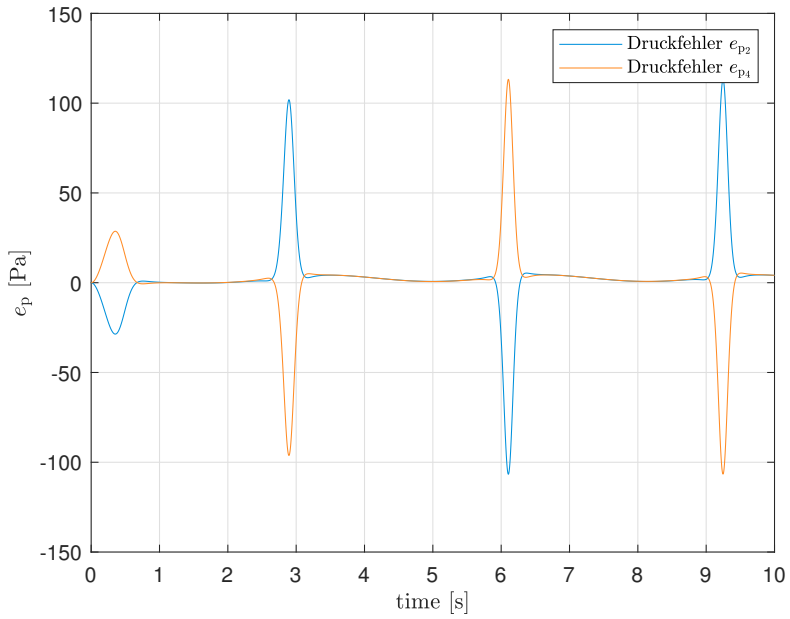


Abbildung 5.6.: Druckfehler in Kammer 2 oben; in Kammer 4 unten. [Aus ro-Pa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_1]

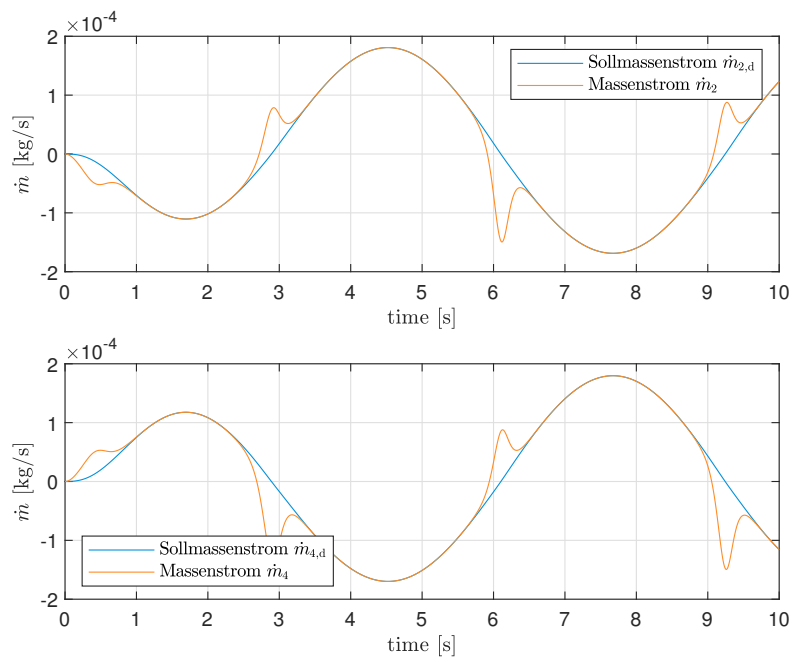


Abbildung 5.7.: Gewünschter Massenstromverlauf (-) vs. simulativer Massenstrom (-).
[Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_1]

6. Flachheitsbasierte Regelung des RoPa30 Antriebs unter Berücksichtigung von Totzeiten

Das Simulationsmodell 6.1 ist sehr ähnlich zum Modell ohne Totzeitberücksichtigung in Abbildung 5.1, bis auf die Zeitverzögerungen durch das Safety-Modul, sowie durch eine Stellgrößenverzögerung aufgrund von Synchronisationswartezzeiten. Zudem wird die "Rück"-Prädiktion der Zustände zum momentanen Zeitpunkt hinzugefügt, welche zur Regelung und Linearisierung verwendet wird.

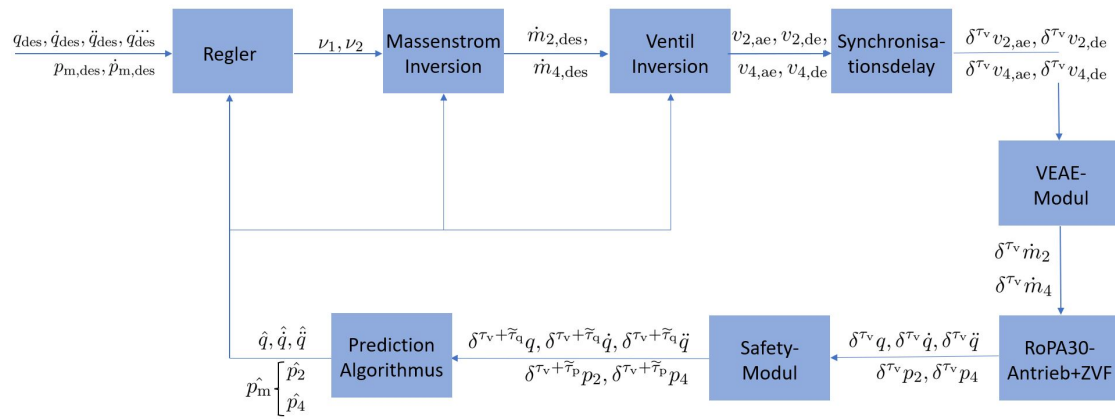


Abbildung 6.1.: Simulationsmodell des δ -flachen Reglers für den RoPA30-Antrieb mit der Delta-Schreibweise $\delta^{\tau} f := f(t - \tau)$.

Aufgrund der Blockdiagonalmatrixstruktur der Matrix \mathbf{A}_{0k} in (3.54) lässt sich die Inverse Matrix offline analytisch berechnen, was die online-Rechenzeit des Prädiktionsalgorithmus deutlich verkürzt. Für den Eintrag \mathbf{A}_{0Q}^{-1} muss jedoch aufgrund der Nullspalte (und

somit nicht vollem Rang) die Pseudoinverse bestimmt werden.

$$\mathbf{A}_0^{-1} = \begin{bmatrix} \mathbf{A}_{0\mathbf{Q}}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{0\mathbf{P}}^{-1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{J^2}{J^2+f_v^2} & -\frac{Jf_v}{J^2+f_v^2} & 0 & 0 \\ 0 & 0 & -\frac{V_2(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))}{n\dot{V}_2(x_{2,\text{saf,d}}(t+\tau_q-\tau_p))} & 0 \\ 0 & 0 & 0 & -\frac{V_4(x_{1,\text{saf,d}}(t+\tau_q-\tau_p))}{n\dot{V}_4(x_{2,\text{saf,d}}(t+\tau_q-\tau_p))} \end{bmatrix}, \quad (6.1)$$

wenn $x_{2,\text{saf,d}}(t+\tau_q-\tau_p) \neq 0$.

Für den Fall, dass $x_{2,\text{saf,d}}(t+\tau_q-\tau_p) = 0$, ist die Pseudoinverse von $\mathbf{A}_{0\mathbf{P}}$ die Nullmatrix.

6.1. Reglerergebnisse

Es wird derselbe Regler wie in Kapitel 5 verwendet. Lediglich die Positionszeitkonstante T_1 , sowie D_1 müssen vergrößert werden, um sicher zu gehen, dass die Position der Referenztrajektorie folgt, aber nicht den Peaks im Massenstrom. Zudem werden durch die Totzeit Schwingungen angeregt (siehe Abschnitt 6.1.2), welche gedämpft werden müssen. Je größer der Dämpfungsparameter, desto besser folgt das System der Referenztrajektorie.

Für einen Folgefehler der Ordnung 10^{-4}rad und 10^{-2}bar wird $D_1 = 2000$ gewählt (siehe Tabelle 4.2). Am Versuchsstand hingegen wird eine deutlich geringere Dämpfung gewählt werden müssen. Zudem ist der Regler nicht robust gegenüber Ungenauigkeiten im Inversionsmodell.

Aufgrund der Tatsache, dass die genauen Totzeiten im System nicht bekannt sind, werden für das Beispiel in diesem Kapitel Totzeiten von $\tau_p = 3 \cdot 10^{-3}\text{s}$ und $\tau_q = 2 \cdot 10^{-3}\text{s}$ angenommen.

6.1.1. Mit Totzeit, mit Prädiktion

Der Druck folgt annähernd der Solltrajektorie in Abbildung 5.4. Der relative Fehler besitzt die Größenordnung $1000/10^5 = 10^{-2}\text{Pa}$ und entspricht dem absoluten Fehler in bar von 10^{-2} , wie in Abbildung 6.3 abgebildet. Es ist jedoch zu bedenken, dass die einzelnen Kammerdrücke nicht geregelt werden, sondern lediglich der Mitteldruck.

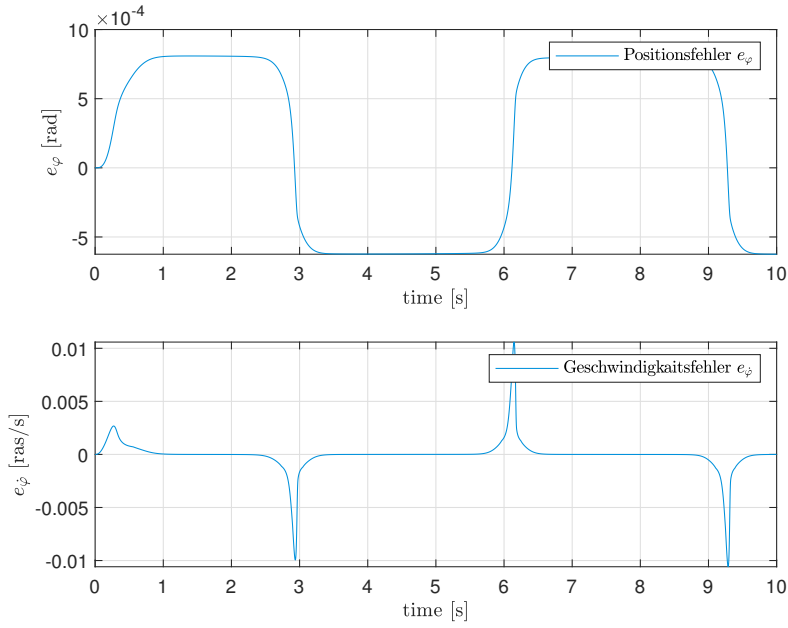


Abbildung 6.2.: Positionsfehler $e_\varphi = \varphi_d - \varphi$ und Geschwindigkeitsfehler bei Totzeiten und Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]

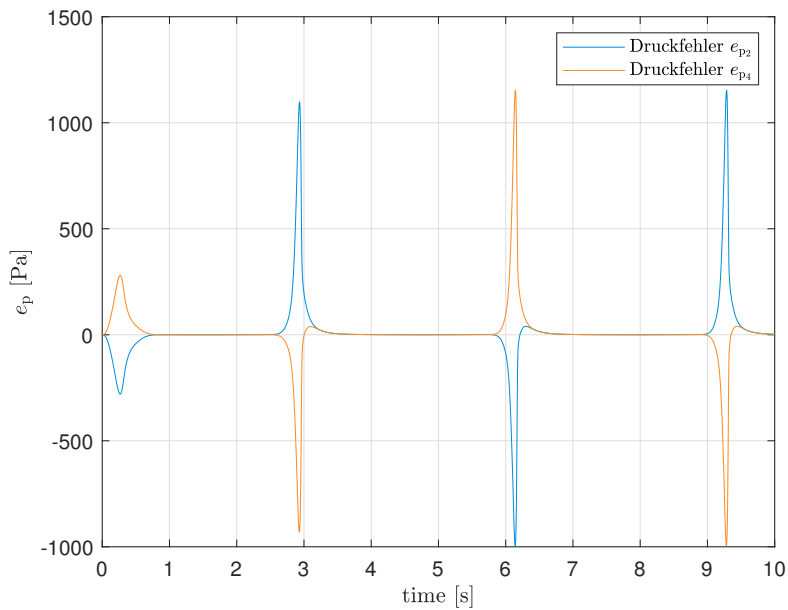


Abbildung 6.3.: Druckfehler $e_p = p_d - p$ bei Totzeiten und Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]

In Abbildung 6.4 ist der Massenstrom abgebildet. Verglichen werden die Sollmassen-

ströme mit der approximierten Reibungsableitung mit den simulativ berechneten Massenströmen. Die Abweichungen resultieren aus der Reibeableitungs-Approximation. Denn genau die Peaks um $\dot{\varphi} = 0$ im Massenstrom fehlen hierdurch, weswegen diese wieder ausgeglichen werden müssen.

Der Mitteldruck folgt mit einer absoluten Genauigkeit von ca. $100\text{Pa} = 10^{-3}\text{bar}$ dem

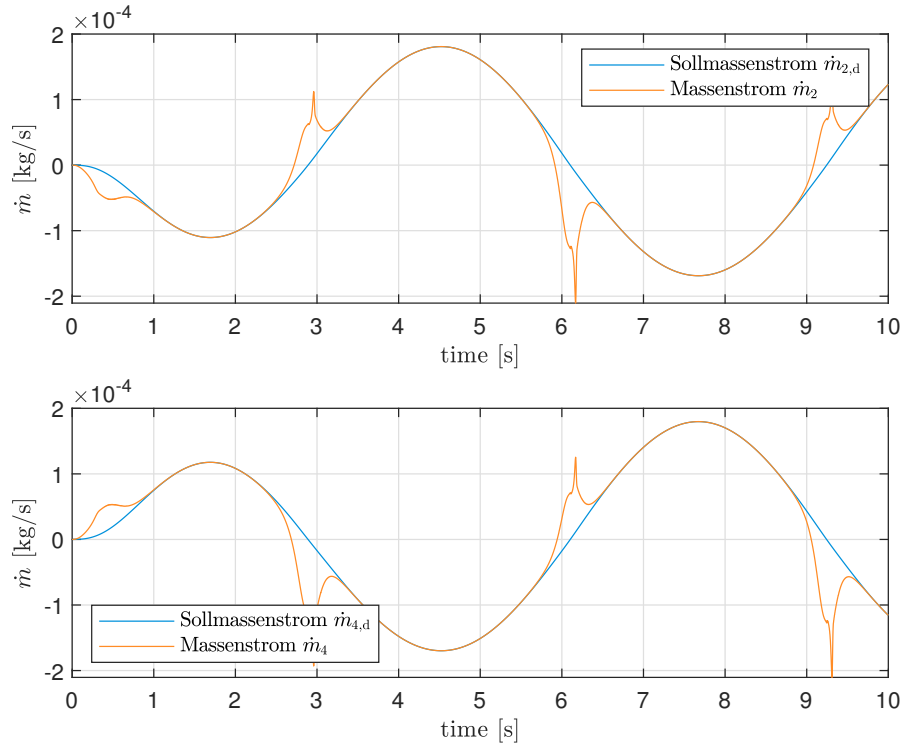


Abbildung 6.4.: Simulativer Massenstrom in orange, sowie Sollmassenstrom in blau bei Totzeiten und Prädiktion. [Aus roPa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]

Sollmitteldruck von 3.5 bar, siehe Abbildung 6.5.

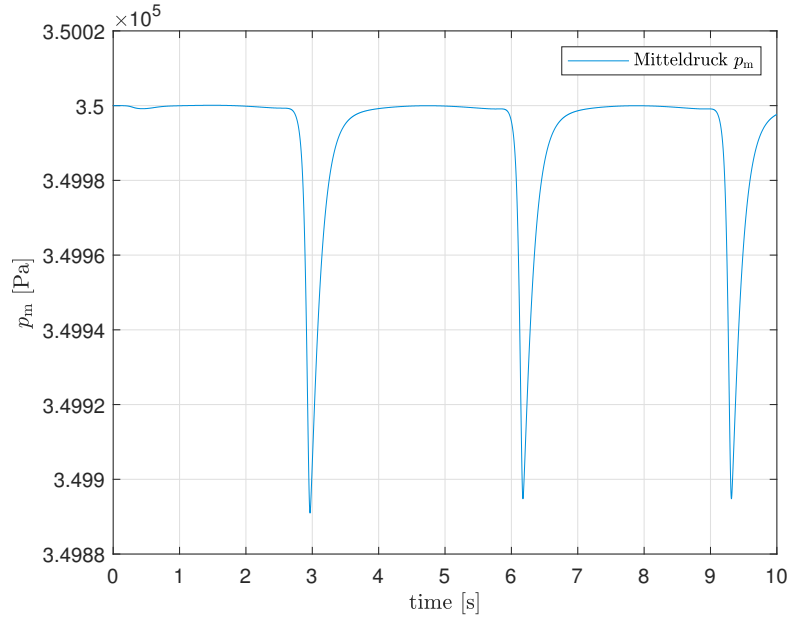


Abbildung 6.5.: Mitteldruckverlauf bei Totzeiten und Prädiktion. [Aus ro-Pa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]

6.1.2. Vergleich mit und ohne Prädiktion

Für diese Arbeit wird $\Delta t = T_s$ verwendet.

In Abbildung 6.6 ist der Positionsfehler durch die Totzeit $e_{\varphi,\text{delay}}(t) = \varphi(t) - \delta^{\tau_a}$, sowie der Fehler in der Position, welcher nach der Prädiktion noch vorhanden ist: $e_{\varphi,\text{pred}}(t) = \varphi(t) - \hat{\varphi}(t)$ (Variablen siehe Abb. 6.1). Es ist zu erwarten, dass der Prädiktionsfehler kleiner ist als der Delay-induzierte Fehler, was auch so ist ($e_{\varphi,\text{delay}} 10^{-3}$, $e_{\varphi,\text{pred}} 10^{-5}$). D.h. die Prädiktion verbessert die Positionsgenauigkeit.

Der Druck wird ebenfalls durch die Prädiktion verbessert, jedoch nur um einen Faktor von 10^{-1}Pa , siehe Abbildung 6.7.

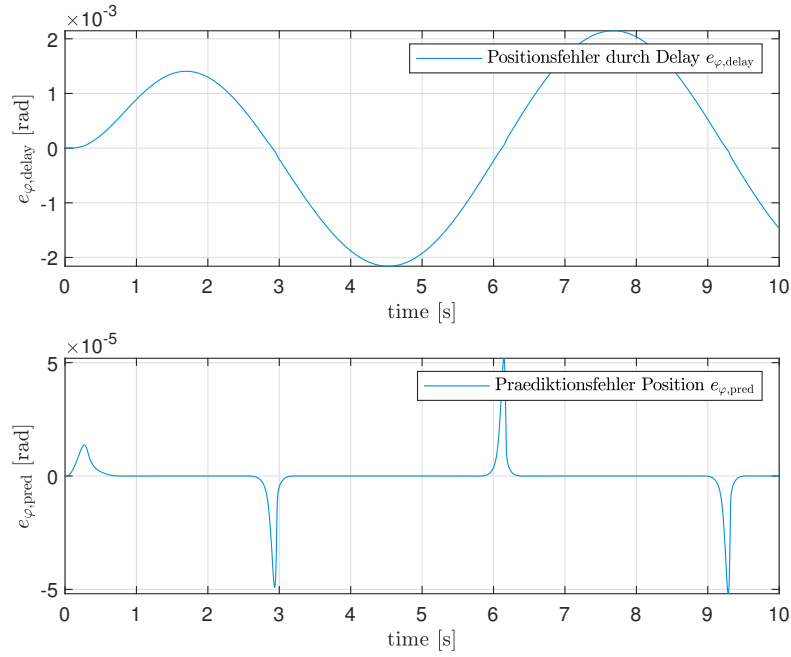


Abbildung 6.6.: Prädiktions- und Delay-Fehler im Winkel. [Aus ro-Pa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]

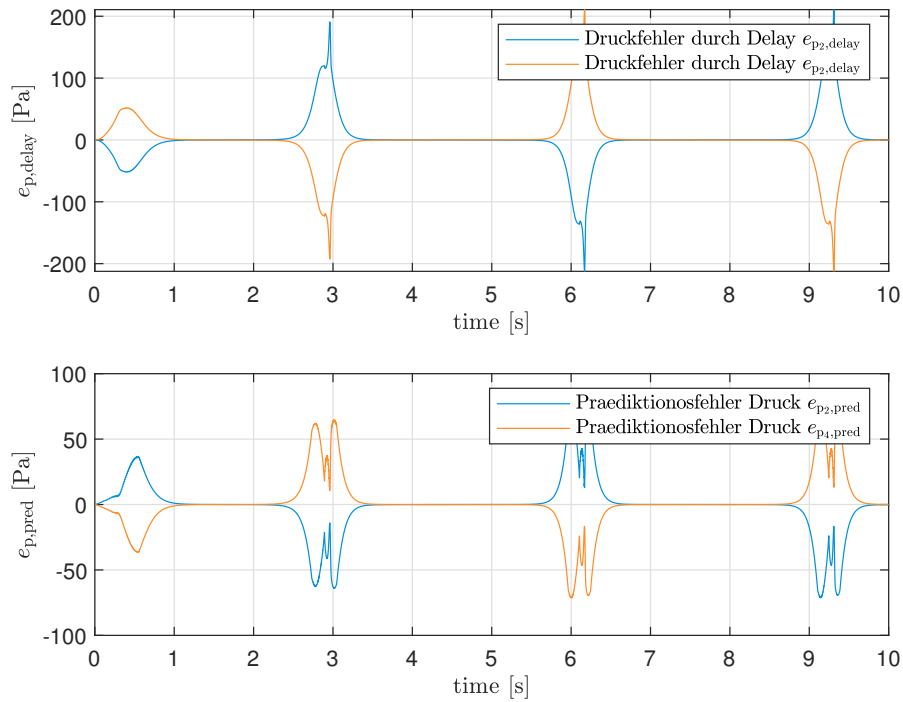


Abbildung 6.7.: Prädiktions- und Delay-Fehler im Druck. [Aus ro-Pa30_flatCtrl_delay_simplify.slx, erstellt am 26.06.2023, Datensatz \mathcal{D}_2]

Wenn die Totzeiten von oben auf das System wirken, jedoch keine Prädiktion vorgenommen wird, so ist die Reglerperformance deutlich schlechter. Für kleinere Zeitkonstanten T_1, T_2 , und somit genauerem Folgen der Referenztrajektorie, führen die Totzeiten zu nicht vernachlässigbaren Schwingungen in allen Systemgrößen. Die Zeitkonstanten müssen also größer gewählt werden, als in \mathcal{D}_2 . Hierdurch verschwindet das Rauschen, der Regler ist also robuster, jedoch ist nun die Genauigkeit der Position und Einzeldrücke deutlich ungenau.

In den Abbildungen 6.8, 6.9, 6.10 sind die Position, Druck und Massenstrom dargestellt, welche sich für groß genug Zeitkonstanten ergeben, für welche keine Schwingungen mehr auftreten.

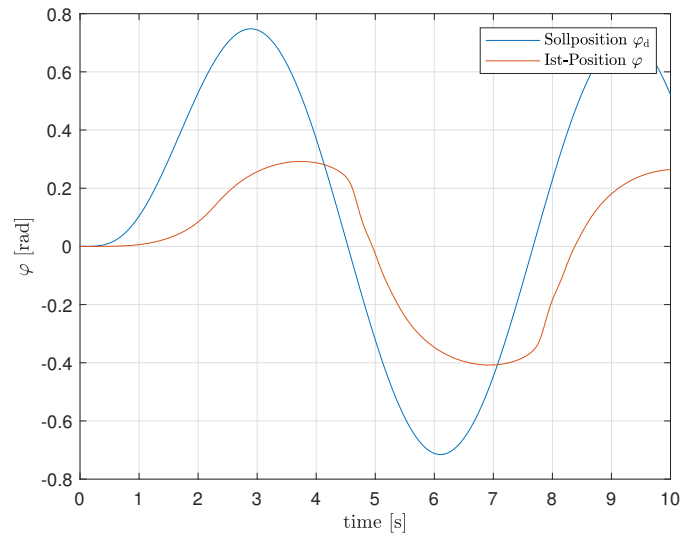


Abbildung 6.8.: Soll- und Ist-Position ohne Prädiktion. [Aus ro-Pa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_3]

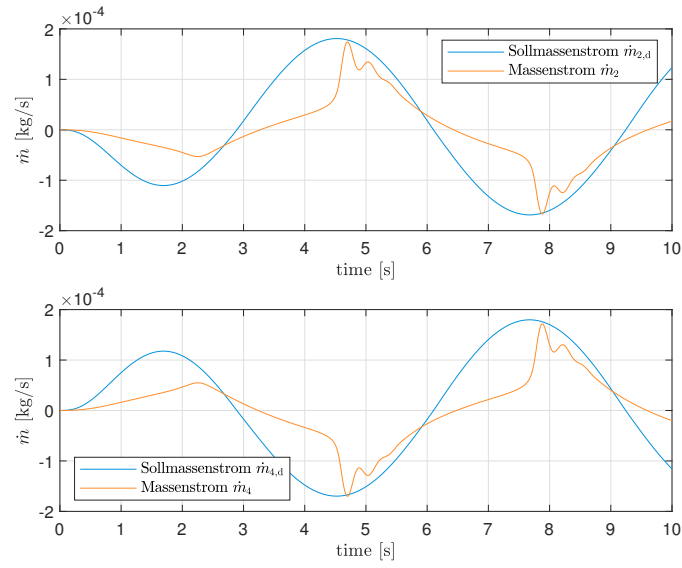


Abbildung 6.9.: Soll- und Ist-Massenstrom ohne Prädiktion. [Aus ro-Pa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_3]

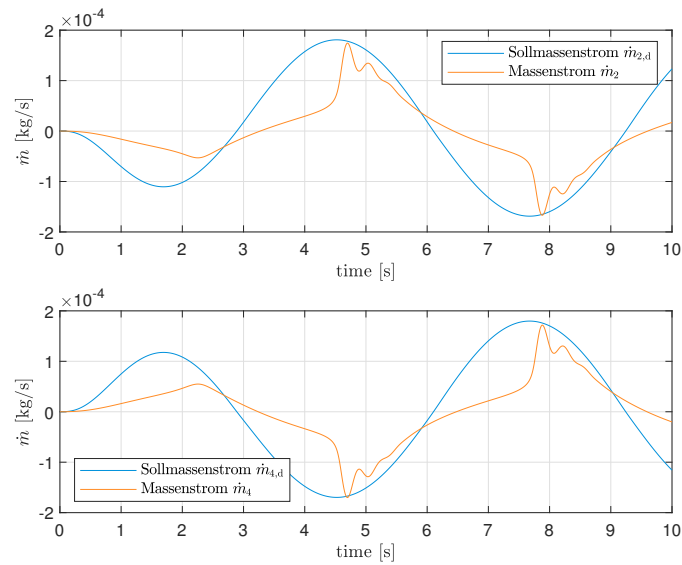


Abbildung 6.10.: Soll- und Ist-Druck ohne Prädiktion. [Aus ro-Pa30_flatCtrl_delay_simplify.slx, erstellt am 28.06.2023, Datensatz \mathcal{D}_3]

6.1.3. Robustheit des δ -flachen Reglers

Die Approximation der Reibungsableitung macht den Massenstrom ungenauer an den Stellen mit $\dot{\varphi}_d$, jedoch wird hierdurch der Regler robuster, da dieser nicht auf jeden Peak des Massenstroms reagiert.

7. Test am Versuchsstand

In dieser Arbeit wurde der flache Regler bei einem Taktzyklus von 1kHz an einem RoPa30-Aktuator angewendet und bei passender Wahl der Regelparameter ergibt sich eine akzeptable/gute Regelpformance - ähnlich zum PID-Regler.

8. Zusammenfassung und Diskussion

Ausblick:

Baue Totzeit am realen System mit ein.

A. Beispiel für Anhang

Dies ist ein Beispiel für ein Anhang

Literaturverzeichnis

- [Brun, 1999] Brun, X. e. a. (1999). Control of an electropneumatic actuator: comparison between some linear and non-linear control laws. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 213(5), 387–406.
- [Fliess, 1995] Fliess, M. e. a. (1995). Flatness and defect of non-linear systems: Introductory theory and examples. *International Journal of Control*, 61, 1327–1361.
- [Marx, 2015] Marx, C. (2015). Nachgiebigkeitsregelung von pneumatischen Aktoren. Master thesis, Esslingen University, Festo AG & Co. KG.
- [Mounier, 1998] Mounier, H., . R. J. (1998). Flatness-based control of nonlinear delay systems: A chemical reactor example. *International Journal of Control*, 71(5), 871–890.
- [Mounier, 1995] Mounier, H. (1995). *Structural properties of linear time-delayed systems: Theoretical and practical aspects*. PhD thesis, Université Paris-Sud, Orsay.