



## Versuchsanleitung - Praktikum NLR 2

---

# Exakte Eingangs/Ausgangs-Linearisierung für Mehrgrößensysteme

Verantwortlicher Hochschullehrer: Prof. Dr.-Ing. Johann Reger

Versuchsbetreuer: M.Sc. Matti Noack

Name, Vorname	Matrikelnummer	Datum
Gruppenmitglieder		
Note	Unterschrift	

# Versuchsanleitung - Praktikum NLR 2

---

## Voraussetzungen

Voraussetzungen für die Vorbereitung und Durchführung des Versuchs sind folgende Lehrinhalte der Vorlesung *Nichtlineare Regelungssysteme 2*:

- Definition des Relativgrades für SISO-Systeme
- Definition des Vektorrelativgrades für MIMO-Systeme
- Exakte Eingangs/Ausgangs-Linearisierung für MIMO-Systeme

Zusätzlich wird auf Grundlagen aus den Lehrveranstaltungen *Nichtlineare Regelungssysteme 1* und *Regelungs- und Systemtechnik 3* zurückgegriffen. Dies betrifft:

- Stabilität der Ruhelage, Kriterien
- Entwurf Linear-Quadratisch optimaler Regler (LQR-Verfahren) für lineare MIMO-Systeme

## Lernziele

- Anwendung der exakten Eingangs/Ausgangs-Linearisierung für nichtlineare MIMO-Systeme
- Ermittlung einer dynamischen Erweiterung
- Bestimmung der Solltrajektorie für eine Vorsteuerung
- Entwurf eines Folgereglers für die entlang der Solltrajektorie linearisierten Strecke

## Anmerkung zur verwendeten Software

Sämtliche Untersuchungen sollen dabei mit Hilfe der Software MATLAB der Firma [The MathWorks](#) durchgeführt werden. Falls Ihnen für die Vorbereitungsaufgaben kein privater Zugang zur Software MATLAB möglich ist, so besteht die Möglichkeit in den Rechenkabinetten des Universitätsrechenzentrums die Software zu nutzen. Alternativ kann eine vergleichsweise kostengünstige Studentenversion von MATLAB erworben werden.

Für die Realisierung der Vorbereitungsaufgabe sowie der eigentlichen Versuchsdurchführung werden simulative Betrachtungen mit Hilfe der [Simulink](#) Toolbox vorgenommen. Zusätzlich wird auf die [Symbolic](#) Toolbox zurückgegriffen, um durch symbolische Programmierung den entsprechenden Regelungsalgorithmus zu bestimmen. Als Arbeitsumgebung wird empfohlen, den [Live Script Editor](#) in MATLAB zu nutzen, wenn die nötige Version vorhanden ist. Dieser ermöglicht übersichtliche symbolische Berechnungen und erleichterte Dokumentation, wie sie aus alternativen symbolischen Rechenprogrammen (z.B. Maple, Wolfram Mathematica, Maxima, etc.) bekannt sind.

## 1 Versuchsbeschreibung

Im Versuch wird die exakte Eingangs/Ausgangs-Linearisierung (E/A-Linearisierung) für nichtlineare Mehrgrößensysteme in quadratischer<sup>1</sup> eingangsafliner Darstellung betrachtet:

$$\Sigma_y : \begin{cases} \dot{x} &= f(x) + \sum_{j=1}^p g_j(x)u_j \\ y_1 &= h_1(x) \\ &\vdots \\ y_p &= h_p(x) \end{cases} \quad (1)$$

Die dabei auftretenden Funktionen seien auf einer offenen Teilmenge  $\mathcal{X} \subset \mathbb{R}^n$  erklärt. Zudem sei eingenommen, dass alle Funktionen  $f : \mathcal{X} \rightarrow \mathbb{R}^n$ ,  $g_i : \mathcal{X} \rightarrow \mathbb{R}^n$  und  $h_i : \mathcal{X} \rightarrow \mathbb{R}$  hinreichend oft differenzierbar sind.

Für diese Systemklasse lässt sich bei Erfüllung gewisser Voraussetzungen ein Regelgesetz finden, welches dem System in neuen Koordinaten ohne Näherung eine lineare Eingangs-Ausgangsdynamik verleiht. Als Resultat können für die so bestimmten äquivalenten Systemdarstellungen nun sämtliche Entwurfsmethoden für linearen Regelungssysteme verwendet werden, wie z.B. der LQR-Entwurf oder die  $\mathcal{H}_\infty$ -optimale Regelung. Darüber hinaus ist der Vorsteuerungs- und Folgeregelungsentwurf für diese linearen zeitinvarianten Systeme meist deutlich einfacher als beim direkten Entwurf mit einer nichtlinearen Dynamik.

In diesem Praktikum soll anhand zweier Beispiele die generelle Vorgehensweise bei der exakten E/A-Linearisierung vertieft werden.

Zur Vorbereitung auf das Praktikum lösen Sie die folgende Aufgabe bitte **vor** dem Praktikumstermin und nutzen Sie zur Bearbeitung MATLAB mit der Symbolic toolbox, um die Rechnungen sowie Simulationen durchzuführen.

Es werden sowohl für die Vorbereitungsaufgabe (im Vorhinein) sowie für die Versuchsdurchführung **Script-Templates** zur Verfügung gestellt, die neben den gegebenen Aufgaben den groben Ablauf der Bearbeitung aufzeigen und Funktionen wie auch Strukturen in MATLAB vorgeben sollen. Diese liegen sowohl als Live Script wie auch als standard m-file vor. Zusätzlich gibt es vorgefertigte Simulationsmodelle in Simulink, die nur noch um die entsprechenden Regelungsfunktionen ergänzt werden müssen.

---

<sup>1</sup>Quadratische Systeme verfügen über die gleiche Anzahl an Eingangs- und Ausgangsvariablen.

# Versuchsanleitung - Praktikum NLR 2

## 2 Versuchsvorbereitung

In der Vorbereitung soll ein vereinfachtes Modell einer permanentenregten Synchronmaschine betrachtet werden. Die Systemdarstellung des Motors ist mit

$$\begin{aligned}\dot{x}_1 &= -x_1 + x_3 \cos(x_4) + u_1 \\ \dot{x}_2 &= -x_2 + x_3 \sin(x_4) + u_2 \\ \dot{x}_3 &= -x_3 + x_1 \cos(x_4) - x_2 \sin(x_4) \\ \dot{x}_4 &= x_3\end{aligned}\tag{2}$$

gegeben<sup>2</sup>. Hierbei beschreiben die Zustände  $x_1$  und  $x_2$  die Ströme im Motor,  $x_3$  die Winkelgeschwindigkeit und  $x_4$  die Winkelposition des Rotors. Die Ausgänge seien wie folgt definiert:

$$\begin{aligned}y_1 &= x_1 \sin(x_4) + x_2 \cos(x_4) \\ y_2 &= x_4\end{aligned}\tag{3}$$

Im folgenden soll eine exakte E/A-Linearisierung für das beschriebene System entworfen werden. Das so bestimmte lineare System soll dann mit einem linearen Zustandsregler in neuen Koordinaten geregelt werden. Verwenden Sie dazu die **Programmiervorlage** `synch_mach_calc` (mlx- bzw. m-File).

### Aufgabe 2.1 - Systemdarstellung

Überführen Sie das Systemmodell (2) und (3) in eine Darstellung  $\Sigma_y$  gemäß (1). Ermitteln Sie die Funktionen  $f$ ,  $g$  sowie  $h$  und implementieren Sie diese als symbolische Strukturen.

### Aufgabe 2.2 - Relativgrad

Bestimmen Sie den Vektorrelativgrad  $r = (r_1 \ r_2 \ \dots \ r_p)$  und Summenrelativgrad  $r_1 + \dots + r_p$  des Systems bzgl. der Ausgänge in (3). Realisieren Sie dazu eine MATLAB Funktion zur Bestimmung von symbolischen Lie-Ableitungen nach der iterative Definition aus der Vorlesung. Ist das System mit diesen Ausgängen exakt E/A-linearisierbar? Welche Bedingungen müssen hierfür erfüllt sein?

### Aufgabe 2.3 - Entkopplung

Stellen Sie das System ausgehend von der vorherigen Aufgabe in der Form

$$\begin{pmatrix} y_1^{(r_1)} \\ y_2^{(r_2)} \\ \vdots \\ y_p^{(r_p)} \end{pmatrix} = \underbrace{\begin{pmatrix} \mathcal{L}_f^{r_1} h_1(x) \\ \mathcal{L}_f^{r_2} h_2(x) \\ \vdots \\ \mathcal{L}_f^{r_p} h_p(x) \end{pmatrix}}_{=: \beta(x)} + \underbrace{\begin{pmatrix} \mathcal{L}_{g_1} \mathcal{L}_f^{r_1-1} h_1(x) & \dots & \mathcal{L}_{g_p} \mathcal{L}_f^{r_1-1} h_1(x) \\ \vdots & \ddots & \vdots \\ \mathcal{L}_{g_1} \mathcal{L}_f^{r_p-1} h_p(x) & \dots & \mathcal{L}_{g_p} \mathcal{L}_f^{r_p-1} h_p(x) \end{pmatrix}}_{=: \alpha(x)} \underbrace{\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_p \end{pmatrix}}_u\tag{4}$$

<sup>2</sup>Für eine detaillierte Modellbildung sei auf [KKM03] verwiesen. Hierbei wurden folgende Annahmen für die Parameter getroffen:  $R = n_p = i_f = 1$ ,  $L_{m1} = \frac{2}{\sqrt{6}}$ ,  $L_a = \frac{2}{3}$ ,  $L_g = 0$

# Versuchsanleitung - Praktikum NLR 2

---

dar und bestimmen Sie die Funktionen  $\alpha$  und  $\beta$ . Welche Aussage können Sie über den Rang der Matrix  $\alpha(x)$  treffen?

## Aufgabe 2.4 - Koordinatentransformation

Transformieren Sie nun das System mit Hilfe des Regelgesetzes

$$u = \alpha(x)^{-1} (w - \beta(x)) \quad (5)$$

mit dem neuen Eingang  $w$  in ein lineares zeitinvariantes System der Darstellung

$$\begin{aligned} \dot{z} &= A z + B w \\ y &= C z \end{aligned} \quad (6)$$

mit Matrizen  $A$ ,  $B$  und  $C$  entsprechender Dimension. Bestimmen Sie dazu zunächst die nichtlineare Koordinatentransformation

$$z = \Phi(x) = \begin{pmatrix} h_1(x) \\ \mathcal{L}_f h_1(x) \\ \vdots \\ \mathcal{L}_f^{r_1-1} h_1(x) \\ \vdots \\ h_p(x) \\ \vdots \\ \mathcal{L}_f^{r_p-1} h_p(x) \end{pmatrix} \quad (7)$$

sowie die inverse Transformation  $x = \Phi^{-1}(z)$ .

*Hinweis:* Nutzen Sie die `solve` Funktion zur Lösung des entsprechenden nichtlinearen Gleichungssystems und `jacobian` zum Prüfen der Lösbarkeitsbedingung ([Satz über Implizite Funktionen](#)).

Welche Struktur weisen  $A$ ,  $B$  und  $C$  auf? Ist das transformierte System (6) steuerbar?

## Aufgabe 2.5 - Linearer Regelungsentwurf

Entwerfen Sie für das System (6) nun einen Zustandsregler der Form  $w = K z$  mit  $K \in \mathbb{R}^{2 \times 4}$ , so dass die Eigenwerte von  $A + B K$  alle bei  $-1$  liegen.

## Aufgabe 2.6 - Systemimplementierung

Implementieren Sie den geschlossenen Regelkreis bestehend aus nichtlinearer Strecke, zustandslinearisierender Rückführung und Zustandsregler in MATLAB Simulink und illustrieren Sie den Systemgrößenverlauf für eine konstante Ausgangssollwertvorgabe.

# Versuchsanleitung - Praktikum NLR 2

## Aufgabe 2.7 - Trajektorienplanung

Schreiben bzw. Erzeugen Sie eine MATLAB Funktion zur Vorbereitung einer Regelung für den Betriebspunktwechsel über eine Referenztrajektorie mit den folgenden Vorgaben:

*Eingänge:* Anfangsbetriebspunkt  $y_0$ ; Sollbetriebspunkt  $y_f$ ; Endzeit  $T_f$ ; aktuelle Zeit  $t$

*Ausgänge:* flache Solltrajektorie  $z^*(t)$ ; Vorsteuerung  $w_i(t) = y_i^{(r_i)}$

Nutzen Sie eine polynomiale Funktion zur Erzeugung des Referenzsignals. Wie sieht die Reglerarchitektur entsprechend für das Lösen des Folgeregelungsproblems aus? Skizzieren Sie das Blockschaltbild als Erweiterung des vorgegebenen Simulink-Modells. [siehe Vorlesung: 5.2 Folgeregelung]

## Aufgabe 2.8 - Recherche und Wiederholung

Machen Sie sich zur Manipulation von Blockmatrixstrukturen und MIMO Systemen mit der Funktionsweise von `kron`, `blkdiag` und `lqr` vertraut. Lösen Sie die folgenden Aufgaben:

a) Input:  $N \in \mathbb{N}$ ,

$$A = \begin{pmatrix} 8 & 1 & 6 \\ 3 & 5 & 7 \\ 4 & 9 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 3 \\ 1 \\ 2 \end{pmatrix}, \quad C = (1 \quad 4 \quad 6).$$

Output:

$$\bar{A} = \begin{bmatrix} A & BC & 0 & \cdots & 0 \\ C^T B^T & A & BC & \ddots & \vdots \\ 0 & C^T B^T & A & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & BC \\ 0 & \cdots & 0 & C^T B^T & A \end{bmatrix} \in \mathbb{R}^{3N \times 3N}, \quad \bar{B} = \begin{bmatrix} B & C^T \\ 2B & 2C^T \\ \vdots & \vdots \\ NB & NC^T \end{bmatrix} \in \mathbb{R}^{3N \times 2}.$$

Berechnen Sie für  $N \in \{0, 1, 5, 20\}$  aus dem Input den Output, ohne Schleifen zu benutzen.

b) Berechnen Sie für die Matrizen  $\bar{A}$  und  $\bar{B}$  aus a) mit  $N = 5$  eine optimale Verstärkung  $K_{LQR}$  mittels der `lqr` Funktion. Beziehen Sie dabei die Gewichtung der Zustandsabweichung nicht mit in die Kostenfunktion ein und gewichten Sie den Stellaufwand mit einer Einheitsmatrix.

Informieren Sie sich über die sogenannte **dynamische Erweiterung** (*dynamic extension*) im Zusammenhang mit der exakten E/A-Linearisierung. Als mögliche Quelle können hier u.a. [Isi95, Sei14] dienen.

# Versuchsanleitung - Praktikum NLR 2

## 3 Versuchsdurchführung

In diesem Versuch soll das Modell eines Portalkranes betrachtet werden. Abbildung 1 enthält eine schematische Darstellung des Aufbaus.

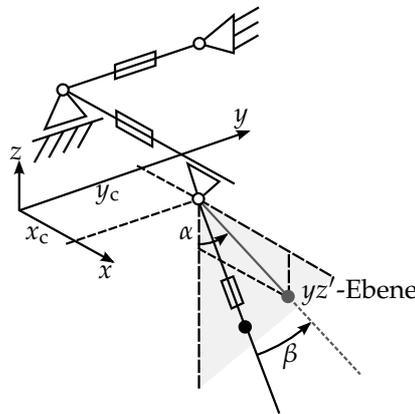


Abbildung 1: Schematische Darstellung eines Portalkranes

Die Laufkatze kann in  $x$  sowie in  $y$  Richtung bewegt werden. Weiterhin kann die Seillänge  $l$  von der Laufkatze zur angehängten Last gesteuert werden. Die Winkel  $\alpha$  und  $\beta$  beschreiben die Auslenkung des Seils.

Über die Lagrange-Gleichungen zweiter Art kann das folgende dynamische System des Portalkranes gewonnen werden:

$$\begin{aligned}\ddot{x}_c &= u_1 \\ \ddot{y}_c &= u_2 \\ \ddot{l} &= u_3 \\ \ddot{\alpha} &= \frac{1}{l \cos(\beta)} (2 \dot{\alpha} \dot{\beta} \sin(\beta) l - 2 \dot{l} \dot{\alpha} \cos(\beta) - \sin(\alpha) g_0 - \cos(\alpha) u_2) \\ \ddot{\beta} &= \frac{1}{l} (-\cos(\alpha) \sin(\beta) g_0 - \sin(\beta) \cos(\beta) \dot{\alpha}^2 l - 2 \dot{l} \dot{\beta} + \sin(\beta) \sin(\alpha) u_2 - \cos(\beta) u_1)\end{aligned}\tag{8}$$

Bei der Herleitung des Modells wird angenommen, dass die Beschleunigung der Laufkatze und der Masse am Seil direkt aufgeprägt werden kann und die Bewegung der Last keine Rückwirkung auf die Laufkatze hat. In einer realen Anlage kann dieses Verhalten durch eine unterlagerte Geschwindigkeitsregelung der Achsen und der Winde umgesetzt werden.

Außer dem Parameter  $g_0$ , welcher die Erdbeschleunigung beschreibt, gibt es in dem Modell keine weiteren freien Parameter wie beispielsweise die Masse der Last oder der Laufkatze. Dies rührt von der unterlagerten Regelung der jeweiligen Achsen her. Weiterhin kann beobachtet werden, dass das System (8) nur für  $l > 0$  Gültigkeit besitzt.

Verwenden Sie zur Bearbeitung der folgenden Durchführungsaufgaben das **Template** `crane_calc` (m1x- bzw. m-File) sowie die Simulink-Modelle zur nominellen Regelung `crane_md1.slx` und mit Vorsteuerung `crane_md1_feedforward.slx`. Dort finden Sie wichtige Hinweise zur Konstruktion des gesuchten flachen Ausgangs und der generellen Systemstruktur.

# Versuchsanleitung - Praktikum NLR 2

## Aufgabe 3.1 - Systemdarstellung

Das System (8) soll exakt linearisiert werden. Als Ausgang werde hier die Position der Last

$$y_l = \begin{pmatrix} x_c + l \sin(\beta) \\ y_c + l \sin(\alpha) \cos(\beta) \\ -l \cos(\alpha) \cos(\beta) \end{pmatrix} \quad (9)$$

verwendet. Bestimmen Sie für das Systems (8) mit dem Ausgang (9) wie in der Vorbereitung zunächst die Funktionen  $f$ ,  $g$  und  $h$  gemäß Darstellung (1) und implementieren Sie diese symbolisch.

## Aufgabe 3.2 - Relativgrad

Bestimmen Sie den Vektorrelativgrad des Ausganges (9) bzgl. der Eingänge  $u_1$ ,  $u_2$  und  $u_3$ . Nutzen Sie hierfür Ihre MATLAB-Funktion aus der Vorbereitung zur Berechnung von Lie-Ableitungen. Ist das System für den gewählten Ausgang exakt E/A-linearisierbar?

## Aufgabe 3.3 - Dynamische Erweiterung zur Entkopplung

Aufgrund der Tatsache, dass der Summenrelativgrad nicht der Systemordnung entspricht, soll für das System das Konzept der dynamischen Erweiterung angewandt werden. Die Idee ist hierbei, dass an beliebigen Eingängen des Originalsystemes Integratoren vorgeschaltet werden und somit der Vektorrelativgrad angepasst werden kann, um eine exakte Linearisierung zu ermöglichen. Die prinzipielle Struktur der Vorgehensweise ist in Abbildung 2 dargestellt.

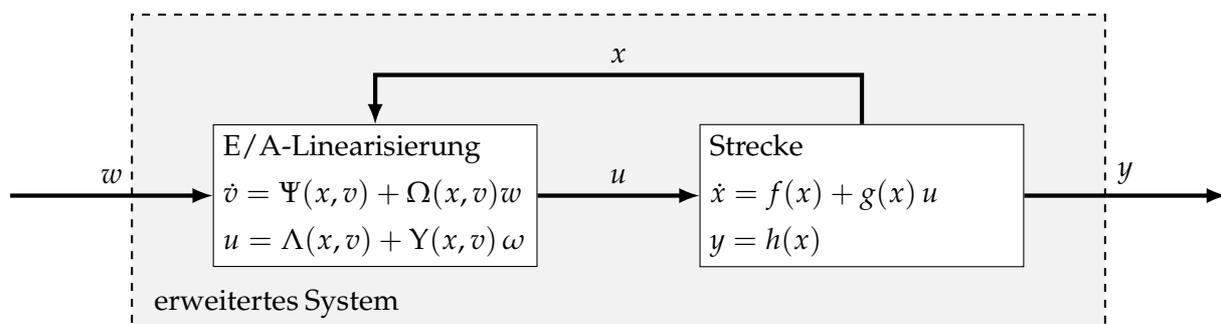


Abbildung 2: Struktur der dynamischen Erweiterung [Isi95]

In dem konkreten Beispiel soll dies mit dem dritten Eingang  $u_3$  erreicht werden. Ziel ist es dabei, dass das System einen Vektorrelativgrad von  $r = (4 \ 4 \ 2)$  bezüglich des Einganges  $u = (u_1 \ u_2 \ v)$  aufweist. Dabei wurde ein neuer Eingang  $v$  eingeführt, welcher als ausreichend oft differenzierbar angenommen wird. Zu diesem Zweck bestimmen Sie  $u_3(x, v, u_1, u_2)$  so, dass gilt  $\dot{y}_{l,3} = v$ . Es werden nun Integratoren vor den neuen Eingang geschaltet. Konkret bedeutet dies, dass das System um die Zustände  $v$ ,  $\dot{v}$  usw. erweitert wird.

Bestimmen Sie nun wie viele Integratoren nötig sind, so dass der Summenrelativgrad der neuen Systemordnung im erweiterten System entspricht.

# Versuchsanleitung - Praktikum NLR 2

---

## Aufgabe 3.4 - Koordinatentransformation

Berechnen Sie nun für das erweiterte System die exakte E/A-Linearisierung! Bestimmen Sie die Funktionen  $\Lambda(x, v, \dot{v}, \dots)$ ,  $Y(x, v, \dot{v}, \dots)$  und die Transformation  $z = \Phi(x, v, \dot{v}, \dots)$ .

Beachten Sie: Die Vorgehensweise ist wie in der Aufgabe der Vorbereitung, jedoch beinhalten  $\Lambda$  und  $Y$  nun noch die Zustände der zusätzlichen Integratoren!

## Aufgabe 3.5 - Optimaler linearer Regelungsentwurf

Entwerfen Sie nun einen LQR-Regler für das linearisierte System in der Form  $w = Kz$ . Nutzen Sie hierbei die in MATLAB enthaltene Funktion `lqr`. Die Gewichtungsmatrizen sind wie folgt zu wählen:

$$Q = \begin{pmatrix} Q^* & 0 & 0 \\ 0 & Q^* & 0 \\ 0 & 0 & Q^* \end{pmatrix} \quad \text{mit} \quad Q^* = \text{diag}(0.01 \quad 0.1 \quad 1 \quad 10), \quad R = \text{diag}(1 \quad 1 \quad 1).$$

## Aufgabe 3.6 - Realisierung des erweiterten Systems

Implementieren Sie ihre berechneten Funktionen sowie die lineare Zustandsrückführung des flachen Zustandes in das bereitgestellte Simulink-Modell, das bereits ein Simulationsmodell des Portalkrans enthält `crane_md1.slx`. Fügen Sie außerdem die benötigte Anzahl an Integratoren in den Block der exakten Linearisierung ein (*Extension* Block).

Simulieren Sie nun das Gesamtsystem bestehend aus Portalkranstrecke und dem E/A-linearisierenden Regelgesetz unter der Vorgabe einer konstanten Referenzposition. Untersuchen Sie den Einfluss von externen Störungen auf das System.

## Aufgabe 3.7 - Vorsteuerungsentwurf

Nun soll für das linearisierte System noch eine Vorsteuerung entworfen werden. Wie oft muss die Trajektorie  $z = z^*(t)$  hierfür differenzierbar sein? Bestimmen Sie ein Polynom, welches das System in einer Zeit von  $T = 10$  von einem Anfangszustand  $y_I(0)$  in einen gewünschten Endzustand  $y_I(T)$  überführt. Implementieren Sie anschließend die Funktion im dafür vorgesehenen Simulink-Block des Modells `crane_md1_feedforward.slx`.

## Literatur

[Isi95] Alberto Isidori. *Nonlinear Control Systems*. Springer, 1995.

[KKM03] Farshad Khorrami, Prashanth Krishnamurthy, and Hemant Melkote. *Modeling and Adaptive Nonlinear Control of Electric Motors*. Springer, 2003.

[Sei14] Robert Seifried. *Dynamics of Underactuated Multibody Systems*. Springer, 2014.