

Hans-Werner Philippsen

Einstieg in die Regelungstechnik mit Python



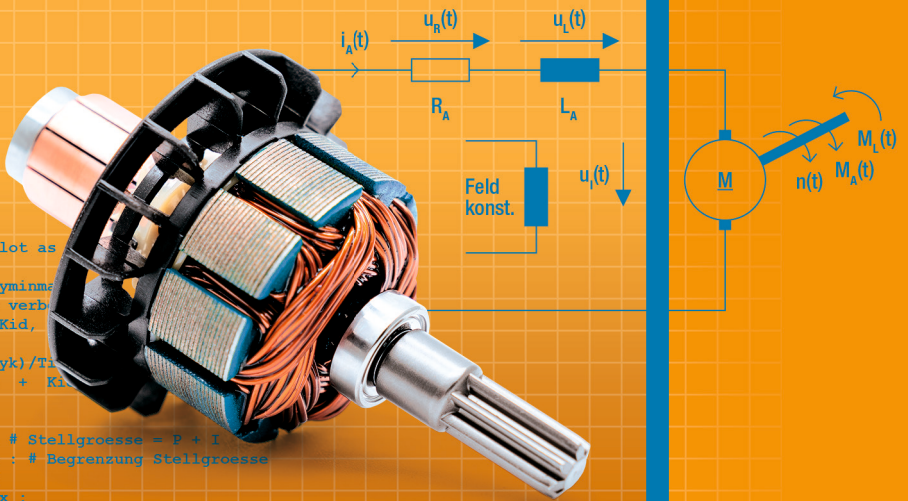
```
import numpy as np
import control as ct
import matplotlib.pyplot as plt

def PI_AWR_Regler(ek, y_minmax):
    """ PI-Regler mit verborgener Aktor- und Feldkonstante """
    global y_minmax, yk, Kid, Kf, Kt, Kd, Ks, Kp, Ki, Kd, Ks, Kp, Ki
    global y_minmax, yk, Kid, Kf, Kt, Kd, Ks, Kp, Ki
    Intkorr = 0
    yk = yk - yk / Td
    yi = yil + Kd * ek + Ki * ek
    ekl = ek
    yil = yi
    ykk = Kp * ek + yi # Stellgroesse = P + I
    if ykk < y_minmax: # Begrenzung Stellgroesse
        yk = y_minmax
    elif ykk > y_minmax:
        yk = y_minmax
    else:
        yk = ykk # y ohne Begrenzung

    return yk

# Sim mit forced response und digitalem PI-Regler
n = 500 # Anzahl Schritte
T0 = 0.01 # Abtastzeit
xx = np.zeros(n) # Regelgrößenvektor
u = np.zeros(n)
yyi = np.zeros(n) # y ohne Begrenzung
tk = np.arange(0, T0 * n, T0) # k-Achse

DCM = control.tf([101, 11.2, -1]) # Motormodell 3. Ordnung
sp = 2 # Typ 2
kp = 1 # Reglerparameter
ki = 0.1 # Reglerparameter
kint = 0.1 # Reglerparameter
kint = 0.1 # Reglerparameter
kint = 0.1 # Reglerparameter
kint = 0.1 # Reglerparameter
```



4., überarbeitete Auflage

HANSER

Philippsen

Einstieg in die Regelungstechnik mit Python



Blieben Sie auf dem Laufenden!

Hanser Newsletter informieren Sie regelmäßig über neue Bücher und Termine aus den verschiedenen Bereichen der Technik. Profitieren Sie auch von Gewinnspielen und exklusiven Leseproben. Gleich anmelden unter

www.hanser-fachbuch.de/newsletter

Hans-Werner Philippsen

Einstieg in die Regelungstechnik mit Python

4., überarbeitete Auflage

HANSER

Prof. Dr.-Ing. Hans-Werner Philippsen

Hochschule Bremen, Fakultät Elektrotechnik und Informatik



Bibliografische Information der Deutschen Nationalbibliothek:

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Dieses Werk ist urheberrechtlich geschützt.

Alle Rechte, auch die der Übersetzung, des Nachdruckes und der Vervielfältigung des Buches, oder Teilen daraus, vorbehalten. Kein Teil des Werkes darf ohne schriftliche Genehmigung des Verlages in irgendeiner Form (Fotokopie, Mikrofilm oder ein anderes Verfahren), auch nicht für Zwecke der Unterrichtsgestaltung – mit Ausnahme der in den §§53, 54 URG genannten Sonderfälle – reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

© 2022 Carl Hanser Verlag München, www.hanser-fachbuch.de

Lektorat: Sylvia Hasselbach

Copy editing: Sandra Gottmann, Wasserburg

Coverconcept: Marc Müller-Bremer, München, www.rebranding.de

Coverrealisierung: Max Kostopoulos

Titelmotiv: © shutterstock.com/Sugrit Jiranarak

Satz: Eberl & Koesel Studio GmbH, Altusried-Krugzell

Druck und Bindung: Hubert & Co. GmbH & Co. KG BuchPartner, Göttingen

Printed in Germany

Print-ISBN: 978-3-446-47282-2

E-Book-ISBN: 978-3-446-47433-8

Vorwort zur 4. Auflage

Die vierte Auflage des Fachbuchs *Einstieg in die Regelungstechnik* wurde überarbeitet und erweitert. Hinzugefügt wurden die Themen Regelung von integralwirkenden Strecken, von Strecken mit Totzeit, Modellreduktion und die numerische Optimierung. Besonders hervorzuheben ist die durchgängige Nutzung der kostenlosen Programmiersprache Python. Mit Hilfe von Python und den freien Paketen NumPy, SciPy und Control lassen sich Regelkreise vollständig berechnen und entwerfen.



web###

Alle Beispielprogramme, die unter der URL www.einstieg-rt.de zum Download bereitstehen, sind im Buch mit dem Kürzel `web###` gekennzeichnet. `###` ist hier der Platzhalter für die jeweilige Listingnummer im Buch. `Web201` z. B. bezeichnet das erste Listing in Kapitel 2. Durch die Eingabe von `einstieg-rt.de/web###` gelangen Sie direkt zum Code.

Leser des E-Books klicken in der PDF-Datei auf den Hinweis `web###` und werden direkt zum Code weitergeleitet.

Moderne Regler sind digital und können u. a. auf der Grundlage von Python auf einem PC oder auf dem Mikrorechner Raspberry Pi oder auf einem Microcontroller realisiert werden. Mit Hilfe der Programmiersprache MicroPython, die für die direkte Ausführung von Python-Code auf Microcontrollern konzipiert wurde, können regelungstechnische Projekte realisiert werden, ohne eine neue Programmiersprache lernen zu müssen. Python hilft zudem bei der Erarbeitung des Stoffes und bietet auch für andere Fachgebiete eine Fülle von Paketen wie z. B. für die Bildverarbeitung, KI und Kommunikationstechnik.

Das Buch wendet sich an Bachelor-Studierende ingenieurwissenschaftlicher Studiengänge, die das Fach Regelungstechnik als Pflicht- oder Wahlmodul studieren. Darüber hinaus wurde das Buch für den praktisch tätigen Ingenieur konzipiert. Ziel ist es, einen praxisnahen und einfachen Einstieg in die Regelungstechnik zu ermöglichen. Aus diesem Grunde wurden weitere Beispiele hinzugefügt, Zeichnungen erneuert und an einigen Stellen der Text überarbeitet sowie Fehler korrigiert.

Die im Rahmen dieses Buches häufige Verwendung von Wirkungsplänen dient der vereinfachten Darstellung abstrakter Zusammenhänge.

Die Regelungstechnik ist ein Fachgebiet mit einer sehr langen Geschichte. Seit den ersten industriellen Anwendungen im 18. Jahrhundert und den ersten theoretischen Arbeiten Mitte des 19. Jahrhunderts kann bis heute eine reichhaltige Entwicklung festgestellt wer-

den. Aus diesem Grunde haben Fachbücher der Regelungstechnik oft einen erheblichen Umfang und sind durch einen großen mathematischen Anteil geprägt. Ein Lehrbuch der Regelungstechnik sollte m.E. die Studierenden in die Lage versetzen, mit möglichst einfachen Mitteln und Software-Unterstützung ausgehend vom Prozess zum eingestellten Regler zu kommen.

Viel Erfolg bei der Erarbeitung des Fachgebietes und der Nutzung der Python-Beispiele!

Bedanken möchte ich mich bei Frau Sylvia Hasselbach für Übernahme des Lektorats und die sehr gute Zusammenarbeit. Der gleiche Dank auch an Frau Kristin Rothe und Frau Irene Weilhart vom Hanser-Verlag für die Arbeit am Text und den vielen Bildern. Meinem Sohn Tom danke ich sehr für die Realisierung des Internet-Auftritts.

Bremen, im April 2022

Hans-Werner Philippsen

Inhalt

1	Einführung	1
1.1	Das Prinzip der Steuerung	2
1.2	Das Prinzip der Regelung	2
1.3	Signale und Komponenten des Regelkreises	3
1.4	Klassifikation von Regelungen	4
1.5	Regelungsstrategien	5
1.6	„Beste“ Einstellung eines Reglers	6
1.7	Statisches Übertragungsverhalten	10
1.8	Dynamisches Verhalten linearer Übertragungsglieder	16
1.9	Besondere Führungsgrößen	21
1.10	Vorgehensmodell Reglerentwurf	22
2	Einführung Python und Pakete	27
2.1	Python	27
2.2	Matplotlib	32
2.3	NumPy	35
2.4	SciPy	37
2.5	Python Control Systems Library	40
2.6	MicroPython	41
3	Mathematische Modelle	47
3.1	Lösung der Differenzialgleichung mithilfe eines Simulators	47
3.2	Lösung der Differenzialgleichung mithilfe der Laplace-Transformation	48
3.2.1	Regeln für das Rechnen mit Übertragungsfunktionen	53
3.2.2	Anfangswert- und Endwertsätze	55
3.3	Frequenzgang	59
3.3.1	Berechnung und Messung des Frequenzgangs	59
3.3.2	Darstellung des Frequenzgangs in der Ortskurve	62
3.3.3	Darstellung des Frequenzgangs im Bode-Diagramm	64
3.4	Python-Beispiel	66
3.5	Mathematische Signalmodelle	68
3.6	Elementare Übertragungsglieder	69
3.6.1	Proportionalglied (P-Glied)	69
3.6.2	Integrierendes Glied (I-Glied)	71
3.6.3	Differenzierendes Glied (D-Glied)	73
3.6.4	Totzeitglied (T_t -Glied)	74
3.7	Wirkungsplan-Elemente der Elektrotechnik	75

4 Regelstrecken	79
4.1 Verzögerungsglied 1. Ordnung (P-T ₁ -Glied)	79
4.2 Verzögerungsglied n-ter Ordnung (P-T _n -Glied)	81
4.3 Verzögerungsglied 2. Ordnung (P-T ₂ -Glied)	85
4.4 Totzeit-Strecke mit Verzögerung	92
4.5 Integrale Strecke mit Verzögerung (I-T ₁ -Glied)	95
4.6 Lead-Lag-Glied	98
4.7 Das allgemeine rationale Übertragungsglied	100
4.8 Strecke ohne Ausgleich: Beispiel Fahrzeuglenkung	102
4.9 Strecke mit Ausgleich: Beispiel DC-Motor	107
4.10 Nicht lineare Strecke mit Ausgleich: Beispiel Fahrzeug	115
4.11 Nicht lineare Strecke mit Ausgleich: Beispiel Tiefsetzsteller	119
5 PID-Regler	123
5.1 PI-Regler	123
5.2 PD-Regler	128
5.3 PD-T ₁ -Regler	129
5.4 PID-Regler	132
5.5 PID-T ₁ -Regler	133
5.6 Industrielle PID-Regler	135
5.6.1 Kompaktregler	135
5.6.2 Regler-Software-Baustein	138
5.6.3 Servo-Regler	140
6 Reglerentwurf	143
6.1 Stationäres Regelkreisverhalten	144
6.2 Dynamisches Regelkreisverhalten	150
6.3 Stabilität	152
6.4 Frequenzkennlinienverfahren	161
6.5 Kompensationsreglerentwurf	164
6.6 Entwurfsverfahren für PID-Regler	165
6.6.1 PID-Kompensationsregler	165
6.6.2 Das Betragsoptimum	169
6.6.3 Einstellregel nach Ziegler und Nichols	173
6.6.4 T-Summen-Einstellregel	174
6.6.5 Einstellregel von Chien, Hrones und Reswick	176
6.7 Regelung von I-T ₁ -Strecken	179
6.8 Regelung von Strecken mit Totzeit	191
6.9 Numerische Optimierung der Reglerparameter	196
6.9.1 Optimierung mithilfe von evolutionären Algorithmen	197
6.9.2 Numerische Optimierung für den robusten Entwurf	200

7	Besondere Strukturen der Regelungstechnik	207
7.1	Störgrößenaufschaltung	207
7.1.1	Statische Störgrößenaufschaltung	208
7.1.2	Dynamische Störgrößenaufschaltung	209
7.2	Kaskadenregler	210
7.3	Regelkreis mit Vorfilter und Vorsteuerung	213
7.4	Regler in der Rückführung	216
7.5	Entkopplung von Mehrgrößensystemen	217
7.6	Schaltende Regler	220
7.7	Zustandsraumbeschreibung	226
7.7.1	Einführung Zustandsraum	226
7.7.2	Zustandsraummodell DC-Motor	233
7.7.3	Ähnlichkeitstransformation	235
7.8	Zustandsregler	239
7.8.1	Struktur eines Zustandsreglers	239
7.8.2	Entwurf Zustandsregler	241
7.8.3	Beispiel Portalkran	248
7.9	Beobachter	256
7.10	Modellreduktion	260
8	Digitale Regelungen	265
8.1	Analog-Digital-Wandlung	268
8.2	Signale des digitalen Regelkreises	271
8.3	Mathematische Beschreibung des Abtastvorgangs	272
8.3.1	Abtasttheorem von Shannon	273
8.3.2	Wahl der Abtastzeit	274
8.3.3	Abtast-Halteglied	274
8.4	Diskretisierung linearer Systeme	278
8.4.1	Diskretisierung P-T ₁ -Glied	278
8.4.2	Diskretisierung I-Glied	281
8.5	Diskretisierte Grundregler	282
8.5.1	Diskretisierter P-Regler	284
8.5.2	Diskretisierter I-Regler	284
8.5.3	Diskretisierter PI-Regler	285
8.5.4	Diskretisierter D-T ₁ -Regler	285
8.6	Regleralgorithmen	286
8.7	Z-Transformation	289
8.8	Stabilität zeitdiskreter Systeme	299
8.9	Bilineare Transformation	301
8.10	Entwurf digitaler Regler	306
8.11	Motion Control	310

9	Beispiele und Aufgaben	325
9.1	Beispiele	325
9.1.1	Temperaturregelstrecke	325
9.1.2	Drehzahlregelung Elektromotor	333
9.1.3	Pendel	338
9.1.4	Dreifachkaskade	343
9.1.5	Querdynamikregelung	352
9.2	Aufgaben	360
9.2.1	Rechnen mit Wirkungsplänen und Übertragungsfunktionen	360
9.2.2	Elektrische Netzwerke	361
9.2.3	Bestimmung von Kenngrößen	361
9.2.4	Berechnung der Sprungantwort	362
9.2.5	Verhalten von PID-Reglern	362
9.2.6	Berechnung von Regelkreisen	362
9.2.7	Reglerentwurf	363
9.2.8	Diskretisierung von Systemen	363
9.2.9	Modellbildung und Reglerentwurf	364
9.3	Lösungen der Übungsaufgaben	364
10	Anhang	369
10.1	Laplace-Korrespondenztabelle	369
10.2	Berechnung der Sprungantwort eines $P-T_2$ -Gliedes	370
10.3	Bestimmung der Überschwingweite eines $P-T_2$ -Gliedes	373
10.4	Reglerentwurf gemäß Betragsoptimum	375
10.5	Installation der Software	377
10.6	Jupyter Notebooks	379
10.7	Input Output Systems ioSys	381
10.8	Formelzeichen	387
10.9	Abkürzungen	390
10.10	Literatur	391
	Index	393

1

Einführung

Der Beginn der Regelungstechnik wird stets mit dem ersten Einsatz eines Fliehkraftreglers zur Drehzahlregelung einer rotierenden Dampfmaschine durch James Watt im Jahre 1788 datiert (Bild 1.1). Auch wenn bereits seit Mitte des 18. Jahrhunderts Fliehkraftregler in Getreidemöhlen eingesetzt wurden, ist die erste Anwendung für Dampfmaschinen als Startpunkt der modernen Regelungstechnik anzusehen. Mit den geregelten Dampfmaschinen von Boulton und Watt, die eine zu ihrer Zeit außerordentlich große Leistung zuverlässig bereitstellen konnten, begann die industrielle Revolution. Die industrielle Nutzung der Regelungstechnik führte schließlich zu einer wissenschaftlichen Betrachtung des Regelungsproblems und einer weitgehend ausgearbeiteten Regelungstheorie [Kri95].

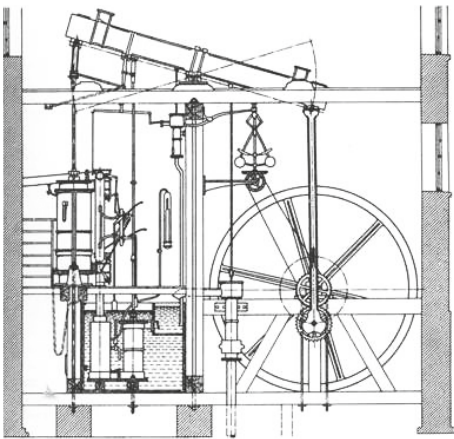


Bild 1.1: Dampfmaschine mit Fliehkraftregler

Die Regelungstechnik stellt aufgrund der langjährigen und intensiven Forschung sowie unzähliger praktischer Anwendungen ein etabliertes Fachgebiet dar. Die Regelungstheorie bietet umfassende mathematische Werkzeuge für die Beschreibung und den Entwurf von Regelsystemen, die im Rahmen dieses Buches in ausgewählter Form dargestellt werden. Die Grundprinzipien und zentralen Begriffe stehen am Anfang der weiteren Einführung.

Um den Begriff Regelung zu erklären, ist es sinnvoll, erst einmal darzulegen, was Regelung nicht ist.

■ 1.1 Das Prinzip der Steuerung

Kennzeichen einer Steuerung ist der offene, in einer Richtung verlaufende **Wirkungsablauf**. Es gibt keine Rückmeldung über den tatsächlichen Zustand des zu steuernden Prozesses. Die Einstellung der Temperatur einer Mischung von Kalt- und Warmwasser kann mithilfe eines Ventils, z. B. des Warmwasserventils, wie in Bild 1.2 gezeigt, erfolgen. Sind die Durchflüsse und Temperaturen von Kalt- und Warmwasser konstant, dann stellt sich eine konstante Temperatur der Mischung ein. In diesem Fall kann die Temperatur durch das Ventil erfolgreich gesteuert werden. Sollte sich eine Größe ändern, z. B. die Warmwassertemperatur, dann liegt eine Störung des Prozesses vor und die Temperatur der Mischung ändert sich.

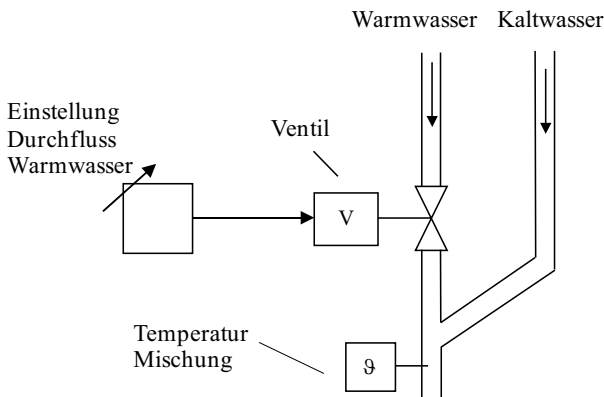


Bild 1.2 Steuerung der Wassertemperatur

Störungen werden nicht automatisch ausgeglichen, das ist das Hauptproblem einer Steuerung! Eine Steuerung erhält keine kontinuierliche Information über den Prozesszustand.

■ 1.2 Das Prinzip der Regelung

Kennzeichen einer Regelung ist der geschlossene Wirkungsablauf. Es erfolgt eine ständige Rückmeldung über den tatsächlichen Zustand des zu regelnden Prozesses. Die entstehende Struktur ist kreisförmig, daher auch die Bezeichnung Regelkreis. In der Temperaturregelung gemäß Bild 1.3 wird die Temperatur kontinuierlich gemessen und dem Vergleichler zugeführt. Weicht der Istwert vom Sollwert ab, so wird der Regler durch ein Steuersignal die Ventilstellung verändern und damit den Warmwasserdurchfluss geeignet anpassen. Ursächlich für eine Abweichung vom Sollwert sind die Störgrößen. Grundaufgabe einer Regelung ist der Ausgleich von Störungen und, wie später genauer beschrieben wird, die Beeinflussung des Führungsverhaltens der Strecke. Die Unterscheidung Steuerung und Regelung ist bei modernen Steuerungssystemen nicht immer einfach, da auch in Steuerungssystemen Rückmeldungen erfolgen können.

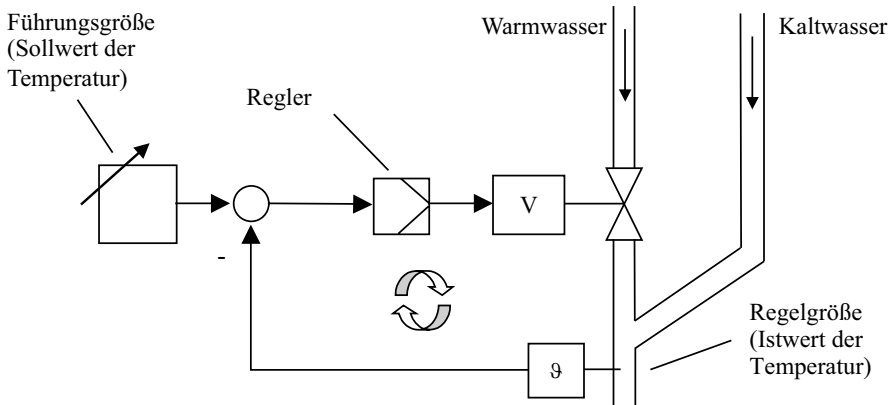


Bild 1.3 Regelung der Wassertemperatur

Diese Rückmeldungen sind z. B. Endschalter „AUS“ oder „Temperatur erreicht“, d. h., es handelt sich meist um binäre Signale, die im Rahmen einer Ablaufsteuerung verwendet werden. Merkmale einer Regelung sind jedoch die kontinuierliche Messung der Regelgröße, der ständige Vergleich mit der Führungsgröße und die permanente Anpassung der Stellgröße. Ein Regler hat die Aufgabe, den Zustand *Führungsgröße* = *Regelgröße* herzustellen, wobei dieser Zustand im Fall von Störungen und im Fall von Führungsgrößenänderungen möglichst schnell realisiert werden sollte.

■ 1.3 Signale und Komponenten des Regelkreises

Die Bezeichnungen der Signale und Komponenten des Grundregelkreises gemäß Bild 1.4 sind genormt (DIN IEC 60050-351 [DIN09], ehemals DIN 19226 [DIN94]) und sollten möglichst normgerecht verwendet werden.

Führungsgröße $w(t)$

Die Führungsgröße wird von außen zugeführt und stellt die Zielgröße dar, der die Regelgröße in endlicher Zeit folgen soll. Die Führungsgröße hat einen vorgegebenen **Sollwert**.

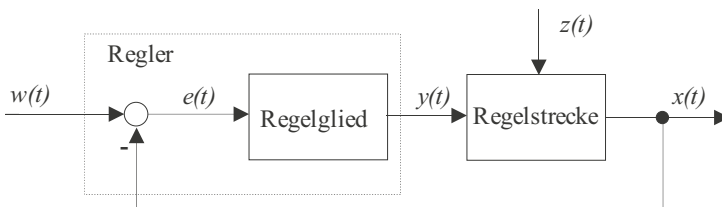


Bild 1.4 Grundregelkreis

Regelgröße $x(t)$

Die Regelgröße $x(t)$ ist die Prozessgröße, die zum Zwecke des Regelns erfasst und der Regeleinrichtung zugeführt wird. Die Regelgröße wird von einem Messglied bzw. Sensor erfasst. Der **Istwert** der Regelgröße soll dem Sollwert folgen.

Regeldifferenz $e(t)$, Regelabweichung $x_w(t)$

Die Soll-Istwert-Abweichung kann als Regeldifferenz $e(t) = w(t) - x(t)$

oder als Regelabweichung $x_w(t) = x(t) - w(t)$

definiert werden. Die Regeldifferenz wird auch als Regelfehler bezeichnet.

Störgröße $z(t)$

Die Störgröße wirkt auf den zu regelnden Prozess und führt zu einer Veränderung der Regelgröße.

Stellgröße $y(t)$

Die Stellgröße überträgt die steuernde Wirkung des Reglers auf die Regelstrecke, wobei die Stellgröße Eingangsgröße der Strecke und Ausgangsgröße des Reglers ist. Die Stellgröße wirkt mithilfe des Stellgliedes bzw. des Aktors auf die Strecke ein.

Regler

Der Regler umfasst das Regelglied und das Vergleichsglied. Der Regler bildet die Stellgröße in Abhängigkeit von der Regeldifferenz, um die Regelgröße der Führungsgröße anzupassen. Regler werden mechanisch, elektronisch oder softwaretechnisch realisiert.

Regeleinrichtung

Die Regeleinrichtung beinhaltet das Regelglied, die Vergleichsstelle und gegebenenfalls die Signalanpassung für das Stellglied.

Regelstrecke

Die Regelstrecke (Strecke) ist der zu beeinflussende Prozess oder das System. Der Regelstrecke wird meist das **Stellglied** (z. B. Ventil) zugeordnet.

■ 1.4 Klassifikation von Regelungen

Regelungen sind nach vielfältigen Kriterien einteilbar, z.B. nach der Art der Führungsgröße: **Festwertregelungen** sind durch eine konstante Führungsgröße gekennzeichnet. Die Temperaturregelung einer Gefrierbox ist ein Beispiel für eine Festwertregelung. Der Regler soll in diesem Fall nur die Störgrößen ausregeln. Bei einer **Folgeregelung** ist die Führungsgröße eine Funktion irgendeiner anderen Größe. Die Regelung muss dafür sorgen, dass die Regelgröße auch unter Störeinflüssen der Führungsgröße möglichst gut folgt. Die Regelung der Position der Laufkatze eines Kranes folgt den Führungssignalen des Kranführers. Die **zeitgeführte Regelung** (auch **Zeitplanregelung**) ist eine Folgeregelung, bei der die Führungsgröße nach einem Zeitplan vorgegeben wird. Andere Kriterien zur Einteilung von Reglern sind z.B. die Regelgröße (Temperaturregler, Drehzahlregler ...) oder das Zeitverhalten (PI-Regler). Darüber hinaus gibt es die Bezeichnungen: Digitalregler, Zweipunktregler, Zustandsregler, Mehrgrößenregler (d. h. Regelung von Strecken mit

mehreren Ein- und Ausgängen), Fuzzy-Regler, neuronale Regler ..., die zum Teil im Rahmen dieses Buches noch erklärt werden.

■ 1.5 Regelungsstrategien

Ein Regler erhält als Eingangsinformation üblicherweise die Regeldifferenz und passt die Stellgröße in Abhängigkeit vom Wert und Zeitverhalten von $e(t)$ an. Die Anpassung von $y(t)$ kann qualitativ mithilfe der folgenden Regelstrategien beispielhaft beschrieben werden:

1. Je größer die Regeldifferenz, desto größer die Gegenreaktion.
2. Je länger die Regeldifferenz vorliegt, desto größer die Gegenreaktion.
3. Je schneller die Änderung der Regeldifferenz erfolgt, desto größer die Gegenreaktion.

Die Gegenreaktion wird durch die Stellgröße auf die Regelstrecke ausgeübt. Das Autofahren erfordert die Anwendung aller drei Regelstrategien, wenn auch unbewusst. Beim Spurwechsel verändern wir den Lenkeinschlag α in Abhängigkeit von der Abweichung zum Ziel. Im Bild 1.5 ist α_2 größer als α_1 , da die angefahrene Spur weiter entfernt ist. Es liegt in diesem Fall eine große Regeldifferenz vor. Stellen wir bei der Kurvenfahrt eine anhaltende Abweichung von der Ideallinie fest, dann vergrößern wir den Lenkeinschlag. Ein plötzlicher Seitenwind erzwingt eine schnelle und heftige Gegenreaktion. Diese Regelstrategien werden in Abhängigkeit von der Geschwindigkeit ausgeführt. Bei hoher Geschwindigkeit ist ein Lenkeinschlag, der für das Einparken günstig ist, beim Spurwechsel sehr gefährlich. Die Regelstrategien müssen hinsichtlich der Intensität an die Geschwindigkeit angepasst werden, d. h., es findet eine Adaption statt. Als menschlicher Regler haben wir nach einer Lernphase ein Gefühl für das richtige Verhalten erworben. Ohne mathematische Berechnungen sind wir in der Lage, die Regelstrategien und deren Adaption sicher anzuwenden. Der Mensch ist als Regler sehr leistungsstark, sehr komplexe und komplizierte Prozesse, wie z. B. der Hochofenprozess, sind beherrschbar.

Es gibt im Rahmen der Realisierung technischer Regelungen Ansätze, den menschlichen Lernvorgang oder die erworbenen Regelstrategien künstlich nachzubilden. Die zwar recht grobe Nachbildung der Hirnfunktionen mithilfe von künstlichen neuronalen Netzen (KNN) ermöglicht die Umsetzung des Lernvorgangs auf dem Computer [VDI95]. Die Vorgabe von unterschiedlichen Situationen (Mustern) und den resultierenden Folgen bildet die Grundlage für das Training der KNN.

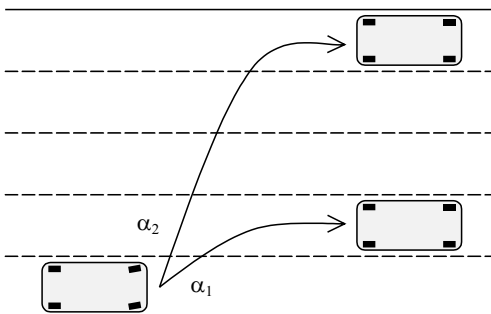


Bild 1.5 Beispiel Fahrzeuglenkung:
Lenkwinkel $\alpha_2 > \alpha_1$ (Regelstrategie 1)

In der realen Welt ist das Lernen ein durchaus riskanter Vorgang, man denke an die ersten Versuche, das Fahrradfahren zu erlernen. Beim Lernen ist daher die Unterstützung durch einen erfahrenen Prozessbediener hilfreich. Gefährliche oder sinnlose Trainingsmuster sind mithilfe einer Anleitung vermeidbar.

Das Lernen wird durch die Erklärung des Vorgangs unterstützt, d. h., sprachlich formulierte Regeln bzw. Regelstrategien werden dem Lernenden vor der Durchführung der eigenen Versuche mitgeteilt. Die softwaretechnische Nachbildung dieser sprachlich formulierten Regeln (z. B.: wenn Fahrrad nach links kippt, dann Lenker nach links) erfolgt durch den Einsatz der Fuzzy-Logik. Da ein Rechner mit *harter* Logik rechnet ($>$, $<$, $=$, \neq), muss ein Software-Paket für die Realisierung der *weichen* Fuzzy-Logik eingesetzt werden. Oft ist es vorteilhaft, die grundsätzliche Funktionsweise, d. h. die verwendeten Regelstrategien, mithilfe von Fuzzy-Regeln zu beschreiben und danach die herkömmliche Berechnung eines Reglers vorzunehmen. Die oben aufgeführten qualitativen Regelstrategien können quantitativ so formuliert werden:

1. $y(t) = K_p e(t)$
2. $y(t) = K_I \int e(t) dt$
3. $y(t) = K_D \frac{de(t)}{dt}$

Bei der Regelstrategie 1 wird die Stellgröße $y(t)$ proportional aus $e(t)$ gebildet, bei der Strategie 2 wird $e(t)$ integriert und im Fall 3 wird $e(t)$ differenziert. Werden alle Regelstrategien kombiniert, erhält man den bewährten PID-Regler. Mit dem PID-Regler werden die meisten Regelungsaufgaben in der Praxis gelöst. Die Formel lautet:

$$y(t) = K_p e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{de(t)}{dt}$$

In Kapitel 5 wird ausführlich auf diesen Reglertyp und die Konstanten K_p , K_I und K_D eingegangen.

■ 1.6 „Beste“ Einstellung eines Reglers

Nachdem die grundlegenden Regelstrategien und eine mögliche Realisierung in Form eines PID-Reglers vorgestellt wurden, stellt sich die Frage, welches Verhalten ein Regelkreis aufweisen sollte.

Diese Frage lässt sich exakt nur für die spezielle Aufgabenstellung beantworten, wobei meist folgende Vorgaben gelten: Einem Sprung der Führungsgröße $w(t)$ sollte die Regelgröße $x(t)$ möglichst schnell, ohne großes Überschwingen folgen und damit den Regelfehler $e(t)$ zum Verschwinden bringen. Störungen sind schnell auszuregeln. Eine Regelgröße reagiert auf eine sprunghafte Änderung von $w(t)$ (Sprung von 0 auf 1 zum Zeitpunkt $t = 0$) mit einem Ausgleichsvorgang, z. B. in der Art von Bild 1.6. In Abhängigkeit von den prak-

tischen Anforderungen wird ein Toleranzbereich definiert (z. B. $\pm 2\%$), innerhalb dessen sich die Regelgröße nach der **Ausregelzeit** T_{aus} befinden soll. Die **Anregelzeit** T_{an} gibt den Zeitraum an, der mit dem Führungsgrößenprung beginnt und mit dem Erreichen des Toleranzbereiches endet. Nach T_{an} kann ein Überschwingen erfolgen, wobei die **maximale Überschwingweite** x_m die maximale Abweichung vom Sollwert angibt.

$$x_m = \frac{x_{\max} - x_{\infty}}{x_{\infty}}$$

Ist x_m innerhalb des Toleranzbereiches, so ist $T_{aus} = T_{an}$. Erreicht die Regelgröße nicht den Sollwert, so spricht man von einer **bleibenden Sollwertabweichung** bzw. von einer **bleibenden Regeldifferenz** $e(\infty)$. Die oben genannten Begriffe gelten auch für die Betrachtung und Beurteilung von Störgrößenprüngen. Das nachfolgende Bild 1.7 zeigt das Verhalten der Regelgröße nach einem Sprung einer Störgröße, die innerhalb der Strecke wirkt.

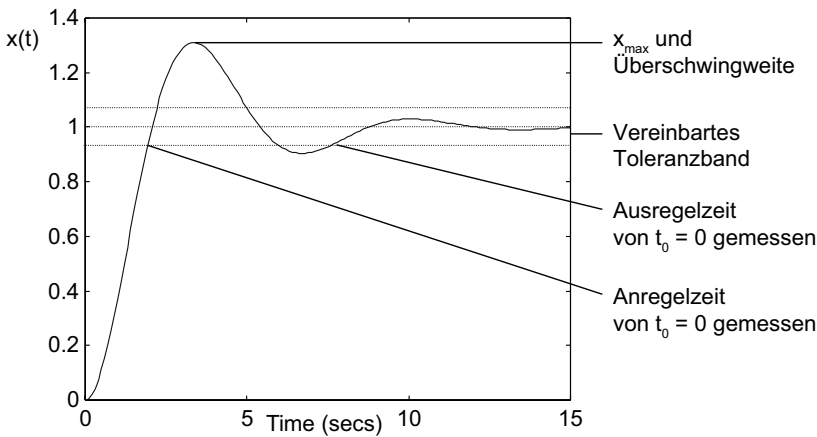


Bild 1.6 Kennzahlen des Führungsverhaltens

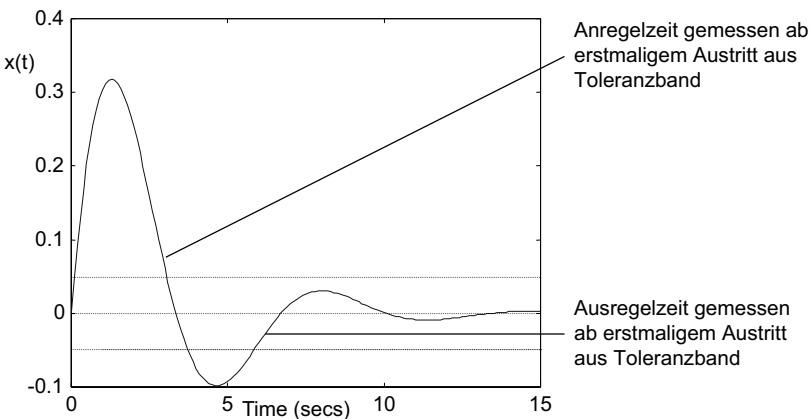


Bild 1.7 Kennzahlen des Störverhaltens

Eine kurze Ausregelzeit ist nicht immer realisierbar und kostet mehr Stellenergie als im Fall einer langsamen Ausregelzeit. Der Einsatz von Reglern ist für den Ingenieur immer eine problematische (manchmal gefährliche) Aufgabe. Regler können, wenn falsch ausgelegt, ein **instabiles Regelkreisverhalten** auslösen; siehe Verlauf einer Regelgröße gemäß Bild 1.8. Ein reales System wird im Fall einer derartig aufklingenden Schwingung möglicherweise beschädigt oder zerstört. Instabile Regelkreise oder Systeme reagieren auf eine beschränkte Eingangsgröße (z. B. $w(t) = 1$) mit einer unbeschränkten Ausgangsgröße (z. B. $x(t) \rightarrow \infty$).

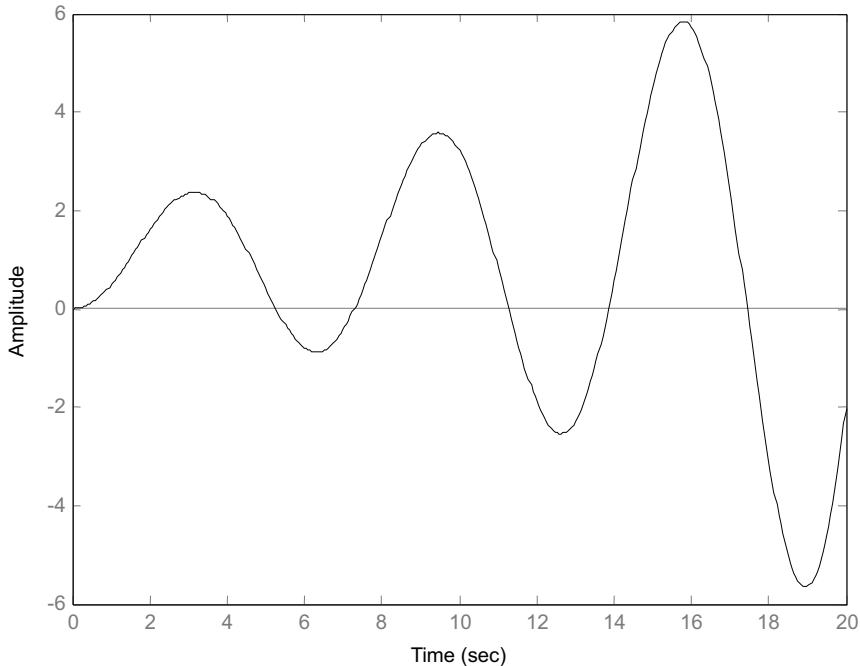


Bild 1.8 Instabiler Regelkreis

Es lassen sich folgende Grundanforderungen an einen Regler stellen:

- Der Regelkreis muss stabil sein.
- Die Regelgröße soll gewünschtes Führungsverhalten und Störverhalten aufweisen.

Damit ein Regler diese Grundanforderungen erfüllen kann, muss er richtig entworfen bzw. eingestellt sein. Der *richtige* Regler mit den *richtigen* Reglerparametern kann nach den folgenden Methoden bestimmt werden:

- Versuch-und-Irrtum-Methode, d. h. probieren, bis Anforderungen erfüllt sind,
- mithilfe von Einstellregeln, nach sorgfältig durchgeführtem Streckentest,
- unter Verwendung eines mathematischen Modells, wobei das Modell messtechnisch oder aufgrund einer physikalischen Modellbildung gewonnen wird.

Leider ist in der Praxis die problematische Methode 1 weitverbreitet, mit der Folge, dass sehr viele Regelkreise ungenügend eingestellt sind. Der Einstellung der *richtigen* Reglerparameter geht die Auswahl eines geeigneten Reglertyps bzw. die Strukturierung des Reglers voraus. Der PID-Regler eignet sich nicht in allen Anwendungsfällen, oft schadet der differenzierende Anteil mehr, als er Nutzen bringt. Die Auswahl des Reglertyps und die Bestimmung der Reglerparameter wird Reglerentwurf genannt.

Reglerentwurfsverfahren liegen in einer nicht mehr zu überblickenden Vielfalt vor. Mit diesem Buch wird nicht die Absicht verfolgt, möglichst viele Entwurfsverfahren vorzustellen.

Auf den ersten Seiten wurde auf die Notwendigkeit der Modellbildung für einen erfolgreichen Reglerentwurf eingegangen. Die Modellbildung schafft ein Abbild der Wirklichkeit, wobei das Modell zunächst die zu regelnde Strecke betrifft. Ein Modell der Umgebung, d. h. der auf den Regelkreis einwirkenden Größen, vervollständigt die Modellbildung. Dabei handelt es sich um die Führungsgröße(n) und die Störgröße(n), die auch als Signale bezeichnet werden. Auf mathematische Signalmodelle wird in Abschnitt 3.5 eingegangen.

Im Folgenden steht die Modellierung von Regelstrecken bzw. von dynamischen Systemen im Mittelpunkt der Betrachtungen. Bestes Resultat einer Modellbildung stellt ein vollständiges mathematisches Modell dar, das sich sehr gut für den Entwurf eines Reglers eignet. Python kann mithilfe der Zusatzpakete eine Basis für die Berechnung dieser Modelle bieten. Manchmal gelingt nur die grobe Modellierung der Regelstrecke. Trotzdem kann aufgrund einer groben Klassifikation der Strecke eine sinnvolle Auswahl des Reglertyps erfolgen. Ein Modell kann auf der Grundlage der Physik, Chemie, Biologie ... der Strecke erstellt werden oder auf der Grundlage von Messungen. Im ersten Fall spricht man von theoretischer Modellbildung, im zweiten von experimenteller Modellbildung. Im Rahmen einer Modellbildung hat sich die Anfertigung eines sogenannten Wirkungsplanes bewährt. Diese grafische Darstellung erleichtert das Verständnis der Vorgänge innerhalb der Strecke und des Regelkreises.

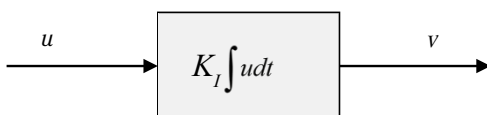
Wirkungsplan

Ausgehend von einem Anlagenplan, Schaltplan oder Rohrleitungs- und Instrumentenfließbild (RI-Fließbild) lassen sich die für die Regelungstechnik relevanten Wirkungen mit einem Wirkungsplan darstellen. Ein Wirkungsplan wird oft auch als **Signalflussplan** oder **Blockschaltplan** bezeichnet. Sind mathematische Zusammenhänge bekannt, sind diese vollständig mit einem Wirkungsplan darstellbar. Ein Wirkungsplan hat vier Elemente.

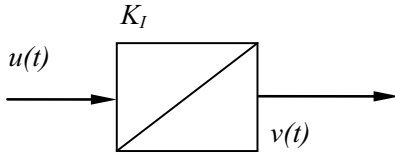
Die **Wirkungslinie** stellt den Weg/Wirkungsrichtung einer Größe dar.



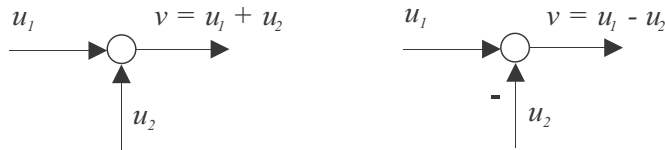
Der **Block** stellt ein System dar, wobei mehrere Ein- und Ausgangsgrößen möglich sind. Jeder Block im Wirkungsplan ist rückwirkungsfrei, d. h., die Ausgangsgröße hat keinen Einfluss auf die Eingangsgröße.



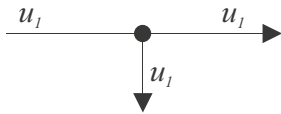
Teilweise wurden für häufig vorkommende Komponenten besondere Symbole definiert, wie zum Beispiel für den Integrierer.



Die **Addition** bildet die Summe mehrerer Größen (Vorzeichen beachten!).



Die **Verzweigung** ist eine Stelle, von der aus ein und dieselbe Größe anderen Blöcken oder Additionen zur Verfügung gestellt wird.



Die Größe teilt sich nicht in Anteile auf, sondern steht den folgenden Blöcken oder Additionen in vollem Umfang zur Verfügung. Die Verzweigungen eines Wirkungsplanes unterscheiden sich daher von z. B. elektrotechnischen Schaltplänen deutlich.

■ 1.7 Statisches Übertragungsverhalten

Das Übertragungsverhalten eines stabilen Systems wird in zwei Kategorien eingeteilt: in das statische sowie in das dynamische Übertragungsverhalten. Beim statischen Übertragungsverhalten wird die Wirkung einer konstanten Eingangsgröße auf den Ausgang untersucht, wobei nur der korrespondierende eingeschwungene Ausgangswert von Bedeutung ist. Die Aufnahme der Kennlinie des statischen Übertragungsverhaltens erfordert die Aufschaltung unterschiedlicher konstanter Eingangsgrößen. Das System antwortet mit einem Übergangsvorgang, z. B. gemäß Bild 1.9, und schwingt auf einen konstanten Wert ein, der in die Kennlinie eingetragen wird. Falls das System nicht *einschwingt*, dann handelt es sich entweder um ein instabiles System oder starke Störungen wirken auf das System ein.

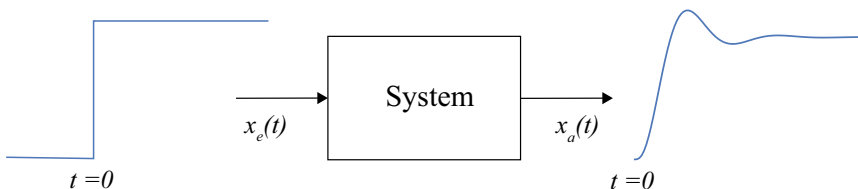


Bild 1.9 Übergangsvorgang

Mit der statischen Kennlinie kann die wichtige Frage geklärt werden, ob eine Strecke linear oder nicht linear ist. Die gebräuchlichsten und die in diesem Buch vorgestellten Regler-Entwurfsverfahren setzen die Linearität der Strecke voraus. Die Linearität eines Systems wird mit dem **Superpositionsgesetz** bzw. mit dem **Überlagerungsprinzip** geprüft:

Bewirkt das Eingangssignal x_{e1} das Ausgangssignal x_{a1} und ebenso x_{e2} x_{a2} , dann hat das überlagerte Eingangssignal $x_{e1} + x_{e2}$ das Ausgangssignal $x_{a1} + x_{a2}$ zur Folge.

Im Rahmen des Reglerentwurfs hat die Aufnahme der statischen Kennlinie eine wichtige Bedeutung. Neben der Verstärkung und Linearität sind folgende Fragen zu klären:

- Ist das Einschwingverhalten (Einschwingzeit, Überschwingen) unabhängig vom Arbeitsbereich?
- Wie groß sind die Störungen?
- Sind die Störungen im gesamten Kennlinienbereich gleich wirksam?
- Gibt es Resonanzbereiche (siehe Motorbeispiel: bestimmte Drehzahlen scheinen Resonanzen des mechanischen Aufbaus anzuregen)?



Beispiel 1.1: Elektromotor

Das statische Übertragungsverhalten (die statische Kennlinie der Wirkung der Steuerspannung auf die Drehzahl) eines Gleichstrom-Elektromotors (Bild 1.10) wird mithilfe eines treppenförmigen Eingangssignals aufgenommen (Bild 1.11).

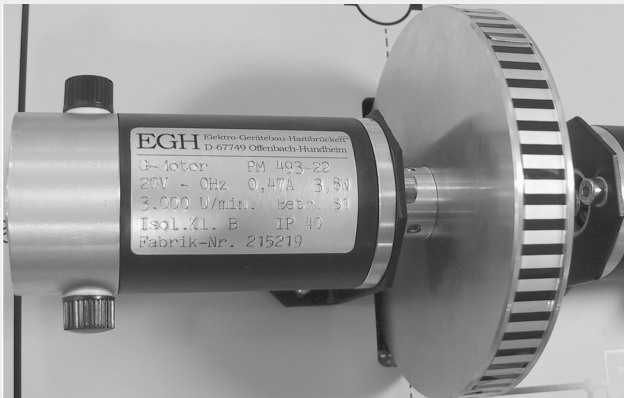


Bild 1.10 Gleichstrom-Elektromotor

Die Steuerspannung erhält in diesem Fall die Sprunghöhe von jeweils 1 V, wobei nach acht Sekunden ein neuer Sprung erfolgt. Das treppenförmige Signal hat im Gegensatz zu einer Rampenfunktion den Vorteil, dass das Einschwingverhalten der Drehzahl genau beurteilt werden kann. Die Drehzahl ist nach ca. sechs Sekunden praktisch eingeschwungen, wobei im unteren Drehzahlbereich deutliche Störungen zu beobachten sind (Bild 1.11).

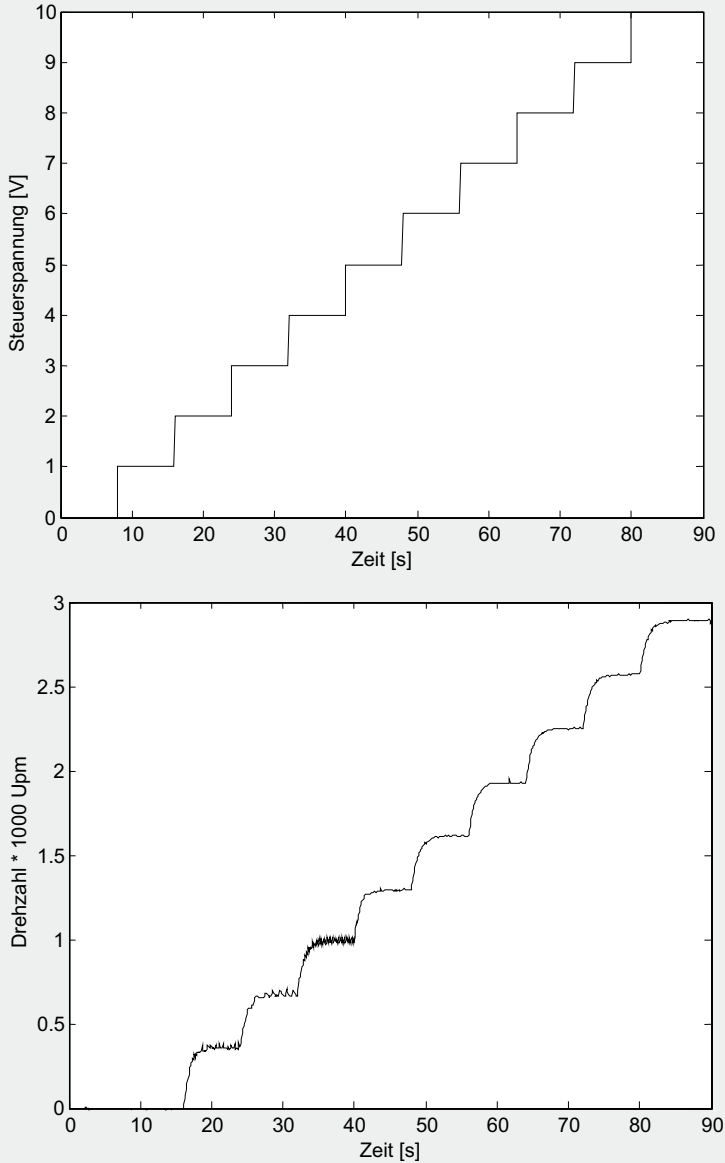
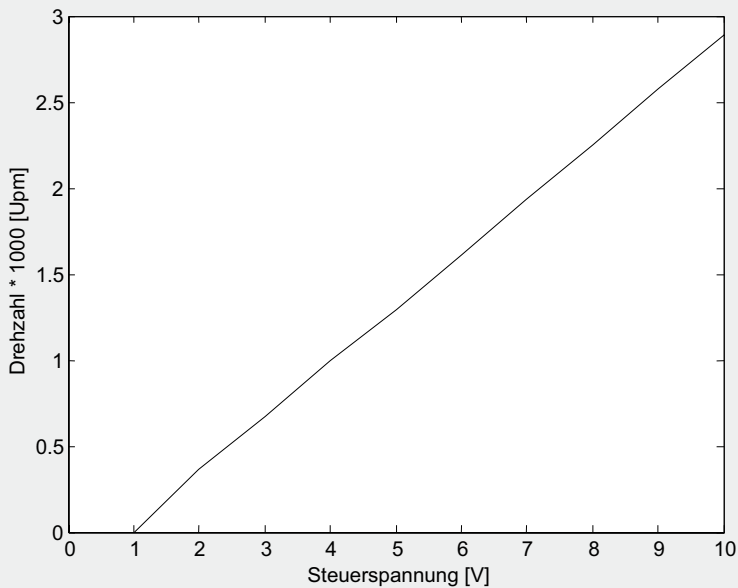


Bild 1.11 Eingangs- und Ausgangssignal für die Aufnahme der statischen Kennlinie Elektromotor

Der Motor benötigt eine Mindestspannung > 1 V, um die Haftreibung überwinden und eine Drehbewegung ausführen zu können. Für die statische Kennlinie wird der Endwert nach dem Einschwingen benötigt, der aufgrund der Störungen berechnet werden muss. In diesem Versuch wurde das Messsignal mit $T_0 = 0,1$ Sekunden abgetastet und der Mittelwert der letzten zehn Werte bestimmt. Der Mittelwert der letzten Sekunde vor dem nächsten Sprung wurde in die Tabelle 1.1 eingetragen.

Tabelle 1.1 Steuer- und Messwerte

Steuerspannung [V]	Drehzahl [1000 1/min]
1	0
2	0,3598
3	0,6736
4	0,9972
5	1,2943
6	1,6133
7	1,9308
8	2,2527
9	2,5726
10	2,8902

**Bild 1.12** Statische Kennlinie Elektromotor

Die abgebildete statische Kennlinie (Bild 1.12) enthält die Messpunkte, die mit einzelnen Geraden verbunden sind. Sinnvoll kann auch die Verwendung einer Ausgleichsgeraden sein oder im Fall stark nicht linearer Kennlinien die Approximation mithilfe von Funktionen höherer Ordnung. In Abschnitt 2.4 wird eine Python-Lösung für die Interpolation und Approximation einer Kennlinie vorgestellt. Reale Systeme sind nur abschnittsweise linear und mit einer Geraden als statische Kennlinie zu beschreiben. Die Begrenzung der Ausgangsgröße (in die-

sem Fall ca. 2900 1/min) kennzeichnet jedes existierende System und damit wird das Superpositionsgesetz streng genommen nie erfüllt. Die statische Kennlinie zeigt den Stellbereich, den Bereich, in dem die Stellgröße verändert werden kann (hier von 0 bis 10 Volt). Der wirksame Stellbereich kann kleiner sein, wie das Motorbeispiel zeigt. Die Regelgröße hat einen Regelbereich, der im Beispiel von 0 bis 2890,2 1/min liegt.

Die statische Kennlinie des Elektromotors ist von der mechanischen Belastung des Motors abhängig. Für unterschiedliche Belastungen (vom Leerlauf bis Vollast) können mehrere statische Kennlinien aufgenommen werden. Zusammengefasst ergeben diese ein Kennlinienfeld.

Statische Kennlinien linearer und nicht linearer Übertragungsglieder

Ein einfaches lineares Übertragungsglied stellt ein Winkelgeber dar (Bild 1.13), der mittels Potentiometer realisiert wird. In Abhängigkeit von der Schleiferstellung mit dem Winkel α ist eine Spannung U_2 messbar:

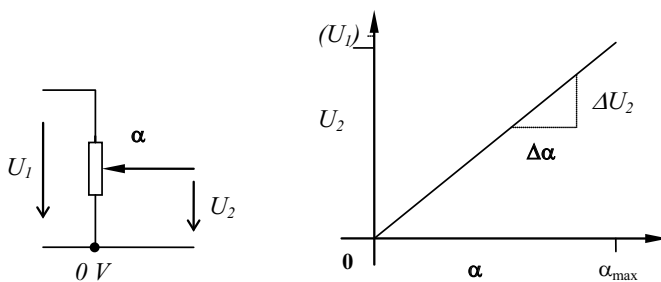


Bild 1.13 Einfacher Winkelgeber

Aus der Steigung der Kennlinie wird der Proportionalbeiwert K_p gebildet:

$$K_p = \frac{\Delta U_2}{\Delta \alpha} = \frac{\Delta x_a}{\Delta x_e}$$

Der Proportionalbeiwert kann auch als Verstärkung bezeichnet werden! Die Übertragungsfunktion $x_a = f(x_e)$ des Winkelgebers ist:

$$U_2 = K_p \alpha$$

Die Kennlinie gibt das statische Verhalten des Winkelgebers wieder, wobei für langsame Änderungen gilt:

$$u_2(t) = K_p \alpha(t)$$

Der Winkelgeber ist ein Beispiel für ein elektromechanisches P-Glied; siehe Abschnitt 3.6.1. Der Proportionalbeiwert kann eine Dimension aufweisen, beim Winkelgeber [V/rad]. Im Fall des Gleichstrommotors beträgt K_p im Arbeitsbereich 500 bis 2500 1/min $K_p \approx 271$

1/min/V. Die mathematische Behandlung von Regelkreisen ist im Fall von nicht linearen Übertragungsgliedern (Bild 1.14) meist sehr schwierig und auch der Reglerentwurf stellt ein Problem dar. Trotzdem lassen sich Regler über einen einfachen Umweg für nicht lineare Systeme entwerfen, wobei zunächst das nicht lineare System in einem bekannten Arbeitspunkt linearisiert wird. Für den überwiegenden Teil der regelungstechnischen Anwendungen ist dieses Vorgehen, d. h., den Reglerentwurf mithilfe eines linearisierten Streckenmodells durchzuführen, ausreichend genau und stabil. Darüber hinaus können Regler für mehrere Arbeitspunkte entworfen und jeweils für einen bestimmten Arbeitspunkt *eingeschaltet* werden. Auch ist eine Anfahrprozedur gebräuchlich, die eine nicht lineare Strecke in den Betriebspunkt steuert und dann den Regler zuschaltet.

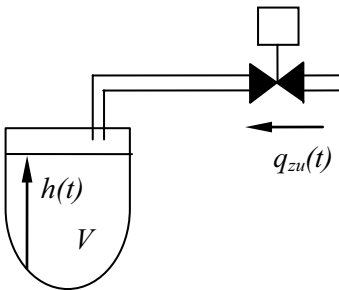


Bild 1.14 Nicht lineares System

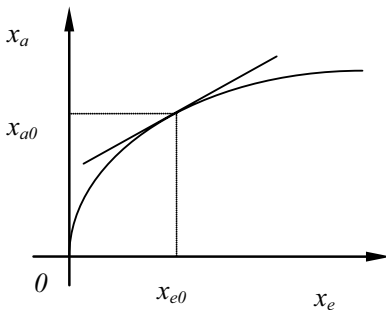


Bild 1.15 Linearisierung

Im Arbeitspunkt, siehe Bild 1.15, wird eine Tangente angelegt. Die Steigung der Tangente entspricht dem Proportionalbeiwert des linearisierten Modells. Zu beachten ist der Linearisierungsfehler, d. h. die Abweichung vom wahren Wert. Der maximal tolerierbare Linearisierungsfehler ist jedoch nur von der konkreten Aufgabenstellung ableitbar! Liegt die nicht lineare Kennlinie formelmäßig als Übertragungsfunktion $x_a = f(x_e)$ vor, kann mithilfe des Taylor-Reihenansatzes im Arbeitspunkt x_{a0} bzw. x_{e0} linearisiert werden:

$$x_a = x_{a0} + \Delta x_a$$

$$\Delta x_e = x_e - x_{e0}$$

$$x_a = x_{a0} + \left. \frac{df}{dx_e} \right|_{x_{e0}} \cdot \frac{\Delta x_e}{1!} + \left. \frac{d^2 f}{dx_e^2} \right|_{x_{e0}} \cdot \frac{\Delta x_e^2}{2!} + \left. \frac{d^3 f}{dx_e^3} \right|_{x_{e0}} \cdot \frac{\Delta x_e^3}{3!} \dots$$

Bei *kleinen* Kennlinienkrümmungen wird die Reihe nach der ersten Differentiation abgebrochen und mit der Tangente im Arbeitspunkt gearbeitet. Der Linearisierungsfehler kann mithilfe des sogenannten Lagrange-Restgliedes abgeschätzt werden oder ist für bestimmte Werte durch Einsetzen in die Formeln direkt bestimmbar.

Im Hinblick auf die Verbesserung der Qualität einer Regelung, sollte ein nicht lineares System – wenn möglich – durch das Vorschalten einer ausgleichenden Kennlinie linearisiert werden. Das Grundprinzip verdeutlicht Bild 1.16. Die nicht lineare statische Kennlinie des bereits vorgestellten Elektromotors erfährt im Anfangsbereich eine Linearisierung, indem eine sogenannte Vorlastkennlinie dem System hinzugefügt wird.

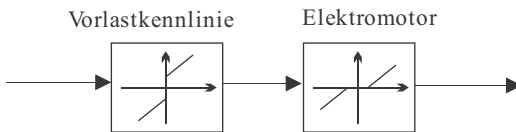


Bild 1.16 Motor mit Vorlastkennlinie

Die Vorlastkennlinie hebt bzw. senkt die Steuerspannung um die Mindestspannung an. Das Gesamtsystem verhält sich im Eingangssignalebereich ± 9 Volt weitgehend linear, insbesondere um den Nullpunkt. Dieses Verhalten ist z. B. für das langsame Anfahren und das Halten einer Position günstig. Eine weitere Vorschaltkennlinie kann die inverse statische Kennlinie sein. Voraussetzung für den Einsatz von Vorschaltkennlinien ist die Möglichkeit der Aufteilung eines Systems in einen linearen und einen nicht linearen Teil gemäß Bild 1.17.

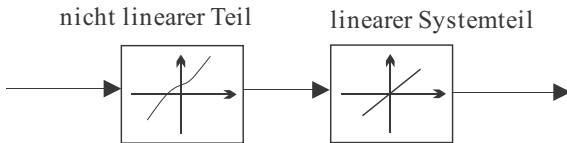


Bild 1.17 Linearisierung einer nicht linearen Strecke

Der nicht lineare Systemteil wird mithilfe der Umkehrfunktion, d. h. der gespiegelten (inversen) Vorschaltkennlinie, linearisiert. Eine quadratische Kennlinie z. B. wird mit der Radizierfunktion linearisiert. Die Linearisierung kann auch durch Regelung erfolgen. Grundaufgabe eines Reglers ist die Herstellung des Zustandes $w = x$, d. h., ein korrekt entworfener Regler stellt ein lineares statisches Regelkreisverhalten her. Die Nutzung von Vorschaltkennlinien erleichtert jedoch den Entwurf und verbessert das dynamische Verhalten deutlich.

■ 1.8 Dynamisches Verhalten linearer Übertragungsglieder

Mit dem Begriff dynamisches Verhalten ist das Zeitverhalten eines Übertragungsgliedes gemeint, das anschaulich mithilfe einer Sprungantwort dargestellt werden kann. Auf einen Sprung der Eingangsgröße $x_e(t) = \Delta x_e \sigma(t)$ reagiert das dynamische Übertragungsglied z. B. mit der in Bild 1.18 gezeigten Sprungantwort $x_a(t)$.

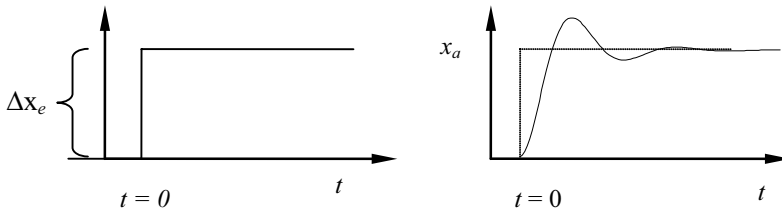


Bild 1.18 Sprungantwort eines dynamischen Systems

Die Sprunghöhe Δx_e ist von der jeweiligen Anwendung abhängig und weist gegebenenfalls eine Dimension auf. Mit der Sigma-Funktion $\sigma(t)$ wird der Einheitssprung bezeichnet. Die Sprungantwort des Systems kann durch Quotientenbildung auf die Sprunghöhe der Eingangsgröße bezogen werden, dann entsteht die **Übergangsfunktion** $h(t)$:

$$h(t) = \frac{x_a(t)}{\Delta x_e}$$

Wird die Sprungantwort in einem Arbeitspunkt von x_a aufgenommen, dann muss die Formel wie folgt erweitert werden:

$$h(t) = \frac{x_a(t) - x_{Ap}}{\Delta x_e}$$

Die Übergangsfunktion entspricht der Sprungantwort eines Systems mit Einheitssprung-erregung. Die Sprungantwort bietet eine sehr gute Grundlage für die Beurteilung der Dynamik eines Systems.

Das dynamische Verhalten wird durch die Energiespeicher des Systems bestimmt. Ein einfaches elektrotechnisches Beispiel zeigt Bild 1.19 mit einer Schaltung bestehend aus einem aufgeladenen Kondensator mit der Ladespannung U_0 und einem ohmschen Widerstand. Der Strom wird durch den Widerstand R begrenzt und kann aus der Spannung am Widerstand und R berechnet werden.

$$i(t) = \frac{u_R(t)}{R}$$

$$i(t) = -C \dot{u}_C(t)$$

$$\dot{u}_C(t) = -\frac{1}{RC} u_R(t)$$

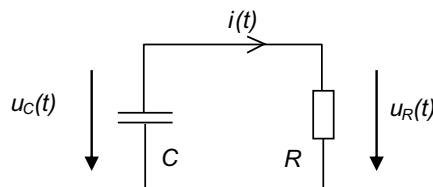


Bild 1.19 Entladung eines Kondensators

Der Strom, der ein Entladestrom ist (negatives Vorzeichen), wird durch die Änderung der Ladespannung bestimmt. Zusammengefasst entsteht die Differentialgleichung

$$\dot{u}_c(t) = -\frac{1}{RC}u_c(t)$$

da die Kondensatorspannung und die Spannung am Widerstand gleich sind. Die Darstellung im Wirkungsplan hat die in Bild 1.20 gezeigte Struktur:

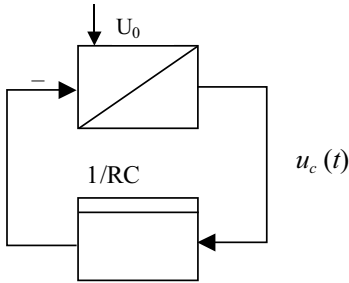


Bild 1.20 Wirkungsplan RC-Glied

Die Ableitung der Kondensatorspannung wird durch Integration nach u_c überführt, die Differentialgleichung wird somit gelöst. Der Anfangswert des Integrators ist die Ladespannung U_0 . Die Dynamik des Entladevorgangs wird durch die Zeitkonstante $T = RC$ bestimmt. In Bild 1.21 ist die Kondensatorspannung in Abhängigkeit von der Zeit dargestellt. Die Spannung nimmt zunächst annähernd linear ab und danach vollzieht sich der Entladevorgang zunehmend langsamer.

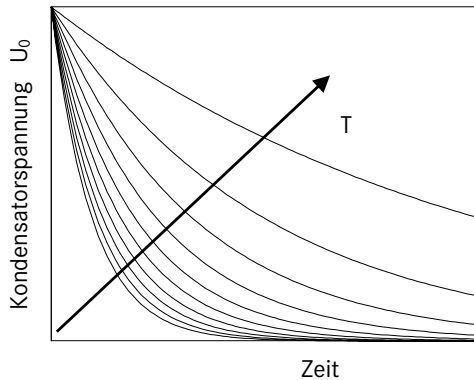


Bild 1.21 Entladevorgänge

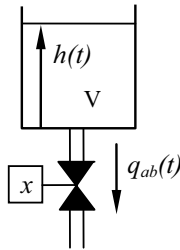
Die Simulationen wurden für zehn verschiedene Zeitkonstanten durchgeführt. Eine kleine Zeitkonstante korrespondiert mit einem schnellen Entladevorgang. Verantwortlich dafür kann ein kleiner Energiespeicher (Kondensator) oder kleiner Widerstand sein.

Ein nicht lineares Beispiel für die Abhängigkeit der Systemdynamik von einem Energiespeicher stellt ein Behälter mit Abfluss dar (Bild 1.22). Der gefüllte Behälter mit der Grundfläche A und dem Pegel h kann mittels Abflussventil (Querschnitt K) entleert werden. Die Ausströmgeschwindigkeit ist gemäß Torricelli

$$v = \sqrt{2gh}$$

Die Änderung des Flüssigkeitsvolumens ergibt sich aus dem Produkt aus Geschwindigkeit und Querschnittsfläche.

Grundfläche A
 Volumen V
 Höhe h
 Ventilstellung x



Die weitere Modellbildung ergibt:

$$\dot{V} = -K\sqrt{2g}x\sqrt{h}$$

$$V = Ah$$

$$\dot{h} = \frac{-K\sqrt{2g}}{A}x\sqrt{h}$$

Bild 1.22 Behälter mit Abfluss

Der Wirkungsplan des Behälters mit Abfluss wird in Bild 1.23 in Simulink-Darstellung gezeigt. Die Dreiecke stellen P-Glieder dar.

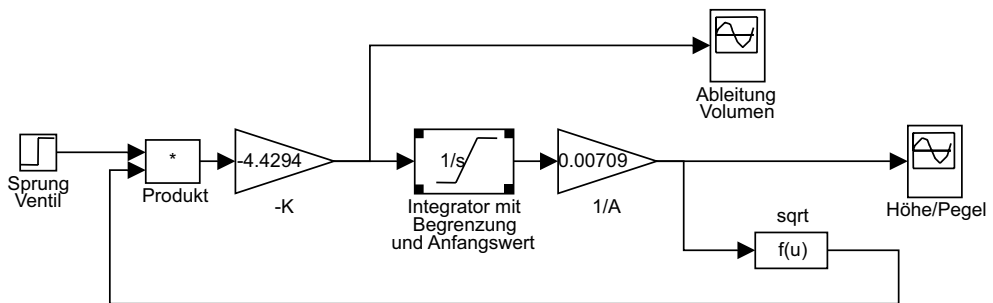


Bild 1.23 Wirkungsplan in Simulink-Darstellung

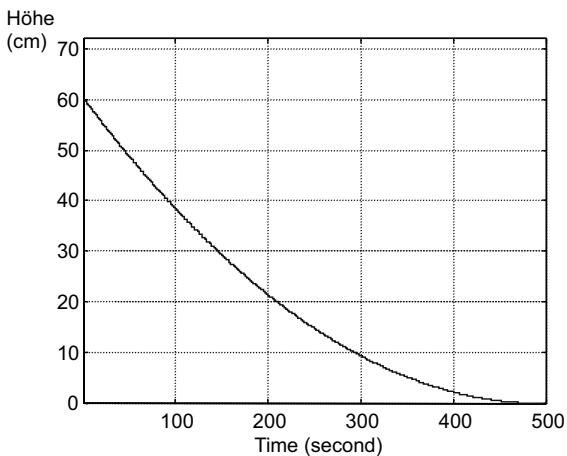


Bild 1.24 Pegel im Fall konstant geöffneten Auslaufventils

Das Diagramm in Bild 1.24 zeigt die Änderung des Pegels im Fall eines konstant geöffneten Auslaufventils. Der Behälter hatte im Zeitpunkt t_0 eine Füllhöhe von 60 cm.

Beschreibung des dynamischen Verhaltens durch Differenzialgleichungen

Um ein genaues mathematisches Modell eines dynamischen Übertragungsgliedes zu erhalten, muss eine Differenzialgleichung (DGL) anhand der physikalischen, chemischen oder biologischen Systembeschreibung aufgestellt werden. Am Beispiel des Reihenschwingkreises in Bild 1.25 wird das Aufstellen einer Differenzialgleichung nochmals gezeigt.

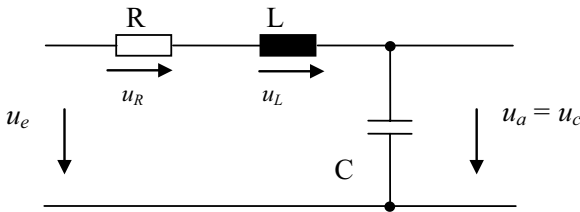


Bild 1.25 Reihenschwingkreis

Die Eingangsgröße ist die Spannung u_e und die Ausgangsgröße ist die Spannung u_a . Das Eingangs-Ausgangs-Verhalten erhält man mithilfe des 2. Kirchhoffschen Satzes (die Spannungssumme einer Masche ist gleich 0):

$$u_e = u_R + u_L + u_a$$

Es gilt:

$$u_R = iR$$

$$u_L = L \frac{di}{dt}$$

$$i = C \frac{du_a}{dt}$$

$$u_e = u_a + CR \dot{u}_a + LC \ddot{u}_a$$

Da beim Reihenschwingkreis zwei Ableitungen nach der Zeit erfolgen bzw. Energiespeicher vorliegen, spricht man von einem System 2. Ordnung. Für Systeme höherer Ordnung ergibt sich eine DGL in allgemeiner Form:

$$x_a^{(n)} + a_{n-1} x_a^{(n-1)} + a_{n-2} x_a^{(n-2)} + \dots + a_1 x_a + a_0 x_a = b_{n-1} x_e^{(n-1)} + b_{n-2} x_e^{(n-2)} + \dots + b_1 x_e + b_0 x_e$$

Für die Lösung von Differenzialgleichungen gibt es eine Reihe analytischer und numerischer Verfahren, die auch im Rahmen von Python-Paketen zugänglich sind. Bevor auf diese Verfahren eingegangen wird, erfolgt eine kurze Einführung in Python und die verwendeten Pakete.

■ 1.9 Besondere Führungsgrößen

Das dynamische Verhalten von Regelkreisen wurde bisher mithilfe von sprungförmigen Führungsgrößen untersucht. Der praktische Einsatz von Regelungen erfordert meist eine andere Form von Führungsgrößen. Soll beispielsweise im Rahmen einer Positionsregelung eine neue Position angefahren werden, erfolgt die Vorgabe eines neuen Sollwertes in Form einer Rampe oder Fahrkurve und nicht sprungförmig. Ein plötzliches Umschalten von $w(t)$ hat eine sehr hohe Belastung des Motors und der Arbeitsmaschine und einen hohen Energieverbrauch zur Folge. Aus diesem Grunde bieten industrielle Regler die Möglichkeit, den zeitlichen Verlauf der Führungsgröße zu beeinflussen. Somit wird aus einem Festwertregler ein Folgeregler. Die einfachste Vorgabe von $w(t)$ stellt die Rampe dar. Eine Drehzahlregelung würde z. B. in Abhängigkeit der Steigung m der Funktion $w(t) = mt$ bis zum Erreichen der Zieldrehzahl folgen. In dieser Phase ist die Beschleunigung begrenzt und konstant. Die Behandlung von Positionsregelungen umfasst die folgenden grundlegenden physikalischen Zusammenhänge von Ort, Geschwindigkeit, Beschleunigung und der Ruckfunktion:

$$v(t) = \frac{dx(t)}{dt} \quad a(t) = \frac{dv(t)}{dt} \quad r(t) = \frac{da(t)}{dt}$$

Je nachdem, ob die Beschleunigung oder auch der Ruck begrenzt sein sollen, erfolgt die Wahl des Führungsgrößenverlaufs. Die Fahrkurve bietet als Sollkurve für die Position/den Weg eine Begrenzung der Beschleunigung, wobei der Geschwindigkeitsverlauf ein Trapezprofil aufweist.

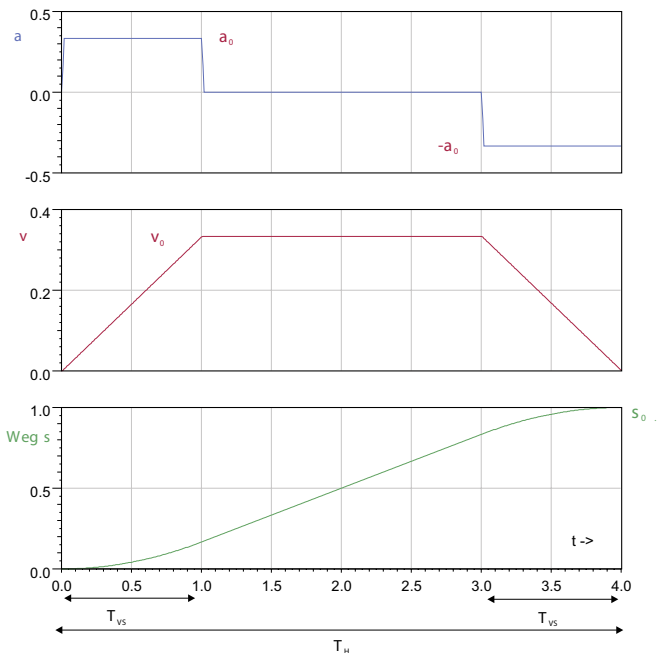


Bild 1.26 Fahrkurve und Trapezprofil

In diesem Zusammenhang wird auch von Bewegungsprofilen gesprochen, die den Regelungen als Führungsgröße zur Verfügung gestellt werden. Wir sehen im Geschwindigkeitsverlauf (Bild 1.26) die oben besprochene Rampenfunktion, die die Beschleunigungs- und Bremsphase bestimmt. Weiterhin liegt eine Fahrt mit konstanter Geschwindigkeit vor. Das Bewegungsprofil wird in Abhängigkeit der maximalen Geschwindigkeit und Beschleunigung des Motors gewählt. Eine Regelung versucht den vorgegebenen Profilen zu folgen, wobei darüber hinaus Störungen ausgeregelt werden. Die Form dieser Profile kann weiterhin durch trigonometrische Funktionen bestimmt werden, insbesondere wenn der Ruck begrenzt werden soll. Das \sin^2 -Geschwindigkeitsprofil weist die genannten Eigenschaften auf. In Abschnitt 8.11 wird auf die Herleitung und Programmierung von Bewegungsprofilen nochmals eingegangen.

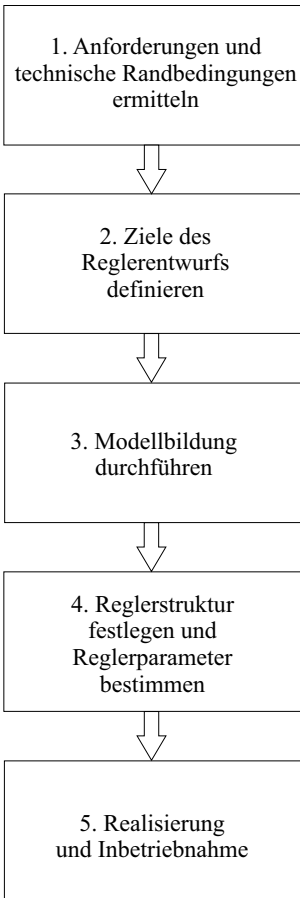
■ 1.10 Vorgehensmodell Reglerentwurf

Die Realisierung eines Regelkreises umfasst viele Arbeitsschritte, die im Folgenden mithilfe eines Vorgehensmodells strukturiert werden. Bild 1.27 zeigt ein derartiges Modell mit fünf nacheinander abzuarbeitenden Phasen.

Die Arbeitsinhalte der einzelnen Phasen sind in der Tabelle 1.2 kurz skizziert. Viele Begriffe und Abkürzungen sind wahrscheinlich noch unbekannt, sie werden jedoch in den weiteren Kapiteln ausführlich erläutert. Das Vorgehensmodell kann in Verbindung mit den Punkten der Tabelle 1.2 als Checkliste verwendet werden. Die Bearbeitung oder Klärung der einzelnen Punkte führt mit großer Sicherheit zum Ziel. Das gilt nicht nur für die praktische Arbeit, sondern insbesondere für das erfolgreiche Studium des Fachgebietes. Die vorgeschlagene Arbeitsweise ist daher auch eine gute Grundlage für die erfolgreiche Bearbeitung der Laborpraktika.

Beim Reglerentwurf geht es zunächst darum, die genaue Aufgabenstellung und technischen Randbedingungen eines Reglerentwurfs zu klären. Wird diese Phase 1 nicht sorgfältig bearbeitet, sind Probleme sehr wahrscheinlich. Ohne Festlegung der Genauigkeit der Regelung können der Sensor für die Regelgröße und auch der Regler nicht sinnvoll ausgelegt werden. Die Präzisierung der Anforderungen im Hinblick auf den Reglerentwurf erfolgt in Phase 2, die die Ziele, aber auch die Testfälle beschreibt. Auf die Notwendigkeit einer Modellbildung wurde bereits hingewiesen, wobei in der Phase 3 Art und Umfang der Modellbildung in Abhängigkeit von den Randbedingungen und den Entwurfszielen geschehen. Im Rahmen einer Modellbildung wird z. B. ein mathematisches Modell erarbeitet oder lediglich eine grobe Klassifikation vorgenommen. Die Phase 4 enthält die eigentliche Entwurfsarbeit und Phase 5 die Realisierung. Der Entwurf kann auf der Grundlage umfangreicher mathematischer Verfahren oder im günstigen Fall mithilfe einer Einstellregel (Faustformel) erfolgen.

Ein Vorgehensmodell hilft als Richtlinie effektiv bei der Vermeidung von Fehlern und der Verbesserung der Qualität. Eine Checkliste kann eine Grundlage eines Reviews sein, um nach durchgeführter Inbetriebnahme die Qualität des Entwurfs und der Realisierung darzustellen. Das um eine Checkliste gemäß Tabelle 1.2 erweiterte Vorgehensmodell ist darüber hinaus ein guter Wissensspeicher des Anwendungswissens der Projekt- und Inbetriebnahme-Ingenieure.

**Bild 1.27** Vereinfachtes Vorgehensmodell**Tabelle 1.2** Detaillierung und Checkliste Vorgehensmodell Reglerentwurf**1. Anforderungen und technische Randbedingungen klären**

- Regelgröße festlegen mit: Messort, Messverfahren, Messbereich, Messgenauigkeit
- Aufgabe des Reglers: Festwertregler, Folgeregler, Verhältnisregler ...
- Anforderung an die Genauigkeit der Regelung
- Auslegung der Sensorik und Aktorik kontrollieren oder durchführen
- Art und Umfang der Störungen, Messbarkeit der Störungen
- Stellbereich, maximale Stellenergie, Stellzeit
- sicherheitstechnische Grenzwerte, Verkopplung mit anderen Systemen
- Anfahrprozedur erforderlich
- Leistungsfähigkeit des Automatisierungsgerätes abschätzen
- Anforderungsszenarien und Testbedingungen definieren

Tabelle 1.2 Detaillierung und Checkliste Vorgehensmodell Reglerentwurf (*Fortsetzung*)**2. Ziele des Reglerentwurfs definieren**

- Führungsverhalten und/oder Störverhalten im Vordergrund?
- bleibende Regelabweichung und Toleranzbereich
- An- und Ausregelzeit sowie Überschwingweite
- Regelgüte (Integralkriterien)
- zeitoptimal oder energieoptimal?
- Entkopplung von Systemteilen erforderlich
- Testkriterien und Testfälle definieren

3. Modellbildung durchführen

- Die Modellbildung wird in Abhängigkeit von den Randbedingungen und Zielen durchgeführt. Innerhalb der Phase 3 kann die Modellbildung theoretisch oder auf Basis einer Modellanlage erfolgen.
- Ermittlung der statischen Kennlinie
- P-T₁, P-T₂ oder P-T_n
- P-T_x mit Totzeit oder integralwirkende Strecke
- G(s) allgemeine Übertragungsfunktion (stabil/instabil)
- Mehrgrößensystem
- Stark nichtlineare Strecke
- Testfälle definieren

4. Reglerstruktur festlegen und Reglerparameter bestimmen

- Auswahl des Reglertyps bzw. Reglerstruktur in Abhängigkeit von 1 bis 3:
- P-Regler, PI-Regler, PD-T₁-Regler oder PID-T₁-Regler
- Kaskadenregler
- Regler in der Rückführung
- Regler mit Störgrößenaufschaltung
- Entkopplungsstruktur
- Entwurfsverfahren auf Grundlage des Frequenzbereichs oder Übertragungsfunktion
- Mehrzielige simulationsbasierte Optimierung
- Auswahl des Automatisierungsgerätes und Bussystems
- Testfälle Reglerfunktion definieren

5. Realisierung und Inbetriebnahme

- Erstellung der EMSR-Stellenpläne und Blätter
- Bestellung der Komponenten, die Überwachung von Lieferung und Montage
- Konfiguration oder Programmierung des Reglers, inklusive aller Steuer- und Bedienfunktionen
- Inbetriebnahme

Da in der praktischen Arbeit Fehler, Rückschläge und Änderungen zu erwarten sind, muss die starre Abfolge der Arbeitsschritte durch Prüfpunkte und Rücksprünge gemäß Bild 1.28 rekursiv erweitert werden.

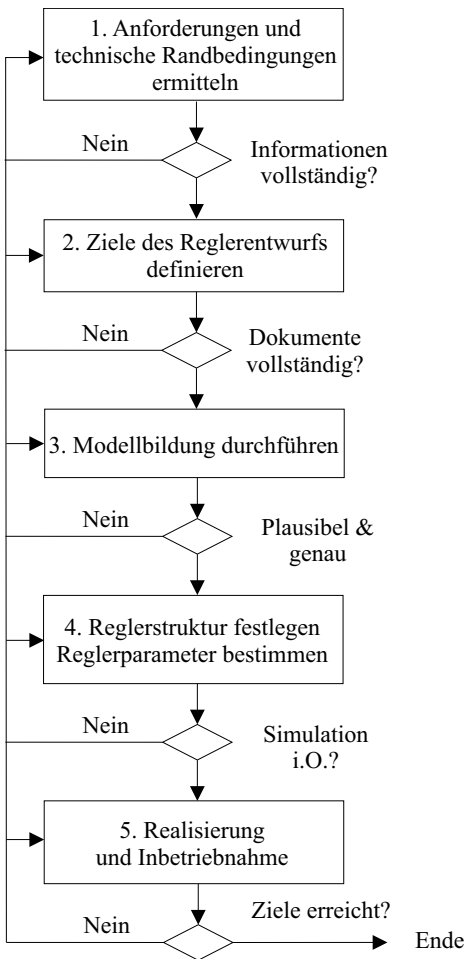


Bild 1.28 Vorgehensmodell rekursiv

Anspruchsvolle Reglerentwürfe erfordern das mehrmalige Durcharbeiten der einzelnen Phasen, bis ein akzeptabler Kompromiss zwischen den oft widersprüchlichen Teilzielen gefunden werden kann. Können wesentliche technische Randbedingungen erst während der Inbetriebnahme einer Anlage geklärt werden, was in der Praxis nicht selten ist, so muss der Inbetriebnahme-Ingenieur vor Ort den Reglerentwurf mit einfachen Methoden durchführen. Sehr wichtig ist es, frühzeitig technische Probleme zu erkennen, die die Qualität des Reglerentwurfs stark beeinträchtigen können. Dazu gehören vor allem falsch ausgelegte Stellglieder und Messbereiche der Sensoren.

Bevor eine erweiterte Diskussion der Phasen 1 und 2 des Vorgehensmodells möglich ist, erfolgt die Darstellung der theoretischen Grundlagen der Regelungstechnik. Kapitel 3 ent-

hält die Grundlagen der Modellierung linearer Systeme sowie die wesentlichen Streckentypen. Es wird somit auf die Grundlagen der Phase 3 eingegangen. Die Tabelle 1.2 umfasst die wichtigsten Begriffe und Arbeitsschritte der Regelungstechnik. Nach Bearbeitung der einzelnen Kapitel und Abschnitte sollte der Lernfortschritt mithilfe dieser Tabelle überprüft werden.

2

Einführung Python und Pakete

■ 2.1 Python

Im Rahmen der Auslegung von regelungstechnischen Systemen sind zum Teil aufwendige Berechnungen erforderlich. Seit Jahrzehnten ist dafür eine Unterstützung durch Software-Pakete üblich, z. B. Matlab [Mat18], octave [Oct18] oder WinFact [Kah16]. Darüber hinaus helfen regelungstechnische Software-Pakete bei der Erarbeitung des Fachgebietes. Moderne Regler sind selbst Software, wobei diese dann Teil eines Rechnersystems bzw. einer SPS ist. Auch aus diesem Grund ist die Einarbeitung in eine regelungstechnische Software und in eine höhere Programmiersprache sinnvoll und wichtig.

Bei den Berechnungen handelt es sich z. B. um die Nullstellensuche, die Lösung von Gleichungssystemen, Frequenzgangberechnungen und um die numerische Lösung von Differentialgleichungen. Für diese Aufgaben gibt es professionelle, und damit kostenpflichtige, Software-Pakete. Als kostenfreie und hochwertige Alternative bietet sich die Programmiersprache Python an, wobei die Erweiterungen NumPy, SciPy, Matplotlib und Control regelungstechnische Berechnungen in vollem Umfang ermöglichen. Weiterhin sind vielfältige Erweiterungen möglich, z. B. Bildverarbeitung, OPC UA, Anbindung von Datenbanken ...

Python ist weit verbreitet und für Windows, Linux und iOS erhältlich (www.python.org). Weiterhin ist Python fester Bestandteil der meisten Linux-Distributionen für den Mikrorechner Raspberry Pi. Dadurch existiert eine kostengünstige Hardware für regelungstechnische Anwendungen.

Als Interpretersprache bietet Python für unsere Zwecke weitere Vorteile. Diese Eigenschaft verbindet Python mit Matlab. Befehle können direkt auf einer Konsole eingegeben und die Ergebnisse z. B. als Zahlen oder Plot ausgegeben werden. Die Anaconda-Distribution von Python enthält die wissenschaftliche Entwicklungsumgebung Spyder, die eine große Ähnlichkeit mit Matlab aufweist (Bild 2.1).

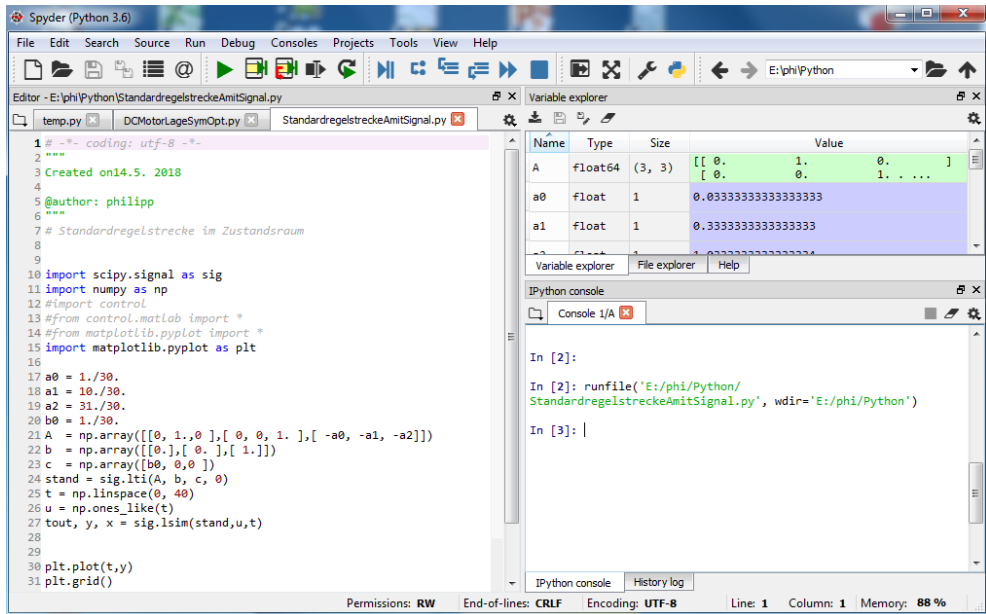


Bild 2.1 Spyder-Entwicklungsumgebung

Die Konsole, der Editor inklusive Debugger, der Variablen-Explorer und das Historylog unterstützen uns bei der Berechnung regelungstechnischer Systeme. Das interaktive Arbeiten wird sehr erleichtert, daher sollte Sie Spyder befähigen, parallel zur Erarbeitung des Stoffes Beispiele zu rechnen.

Von der Programmiersprache Python und den genannten Paketen benötigen wir nur eine kleine Untermenge. Wer Python im Detail kennen lernen möchte, sollte auf umfangreichere Darstellungen zurückgreifen [Kle18], [Kle19], [Woy18] und die Dokumentation im Netz nutzen. Die Beispielprogramme stehen unter einstieg-rt.de zum Download bereit.

Unmittelbar nützlich ist die IPython-Konsole, die zunächst wie ein Taschenrechner genutzt werden kann.

```

In [3]: 124/42
Out[3]: 2.9523809523809526

```

Die Eingabe unter In[*] wird sofort berechnet und unter Out[*] ausgegeben. Es können auch Variablen erzeugt werden, wobei der Typ im Rahmen der Zuweisung festgelegt wird:

```

In[4]: a=2.5
In[5]: b = 4
In[6]: type(a)
Out[6]: float
In[7]: type(b)

```



```

Out[7]: int

In[8]: b = a/b

In[9]: type(b)
Out[9]: float

In[10]: print(b)
0.625

In[11]: b
Out[11]: 0.625

```

Wir sehen, dass `b` zuerst den Typ `int` aufweist und nach der Division `a/b` den Typ `float` annimmt. Diese Eigenschaft mag für Nutzer anderer Sprachen ungewöhnlich sein, insbesondere dann, wenn eine Variable plötzlich komplex wird. Es ist also eine dynamische Typdeklaration möglich. Die wesentlichen Datentypen von Python sind:

<code>int</code>	ganze Zahl
<code>float</code>	Gleitkommazahl
<code>complex</code>	komplexe Zahl
<code>bool</code>	Wahrheitswert
<code>str</code>	Zeichenkette
<code>list</code>	Liste von unterschiedlichen Datentypen, die dynamisch modifizierbar sind
<code>tuple</code>	Tupel sind Listen, die sich nicht ändern lassen

Die Erzeugung einer List- oder Tupel-Variablen wird lediglich mittels der Klammerform gekennzeichnet. Mit `a = [1, 2, b, 42]` wird eine List-Variable erzeugt und mit `a = (1, 2, b, 42)` ein `tuple`. Die für uns wichtigen Datentypen `array` und `matrix` sind Bestandteile des NumPy-Paketes und werden in den entsprechenden Kapiteln behandelt. Eine weitere Besonderheit der Sprache Python ist der Verzicht von Zeichen für den Zeilenabschluss oder z. B. ein Ende-Zeichen für Schleifen und IF-Anweisungen. Mithilfe von Einrückungen und Tabs wird das Programm strukturiert. Soll z. B. eine Stellgröße begrenzt werden, würden das Python- und das IEC 61131-Programm einer SPS wie folgt aussehen:

Python	IEC 61131
<pre> if y > Max_Stell: y = Max_Stell elif y < Min_Stell: y = Min_Stell Nächste Anweisung </pre>	<pre> IF y > Max_Stell THEN y := Max_Stell; ELSIF y < Min_Stell THEN y := Min_Stell; END_IF Nächste Anweisung </pre>

Würde eine Einrückung fehlen, bliebe das beim IEC 61131-Programm ohne Folgen, Python würde die Fehlermeldung `IndentationError: expected an indented block` ausgeben. Als Strukturierungselement bietet Python die Möglichkeit zur Definition von Funktionen. Funktionen können als Unterprogramme aufgefasst werden, die wiederkehrende Berech-