

Prof. Dr. May-Britt Kallenrode

Fachbereich Physik

---

# Simple Simulation

Systemtheorie und SIMULINK

Osnabrück, 14. August 2008

# Vorbemerkung

The most incomprehensible thing about the world is that it is comprehensible  
A. Einstein

§ 1 Das Skript ist eine kurze Einführung in Grundkonzepte linearer, dynamischer und diskreter Systeme sowie deren Darstellung in SIMULINK.

§ 2 Natürliche Systeme wie die Atmosphäre oder die Ozeane bzw. das Klimasystem als eine Kopplung der Systeme Ozean und Atmosphäre sind dynamische Systeme. Sie werden formal durch Systeme partieller Differentialgleichungen beschrieben. Diese lassen sich – außer unter stark vereinfachenden Annahmen – in der Regel nicht analytisch lösen. Die Modellierung derartiger Systeme basiert daher auf numerischen Verfahren.

§ 3 Eine Alternative zu numerischen Verfahren bietet zumindest in einigen Fällen (insbesondere bei linearen DGLs) die Laplace Transformation der Differentialgleichung(en). Diese Transformation überführt das System von Differentialgleichungen in ein System algebraischer Gleichungen. Letzteres ist mit einfachen Methoden zu lösen. Damit lassen sich die Übertragungsfunktionen der Teilsysteme bzw. die Übertragungsfunktion des Gesamtsystems bestimmen. Diese Übertragungsfunktionen liefern die Systemantwort für beliebigen Input. Dieses Verfahren ist jedoch nur für statische Systeme geeignet, da nur in diesen eine Übertragungsfunktion sinnvoll definiert werden kann – ein dynamisches System dagegen würde eine zeitlich veränderliche Übertragungsfunktion aufweisen.

§ 4 SIMULINK ist eine Erweiterung von MATLAB u.a. zur Simulation dynamischer Systeme – es enthält aber ebenso Tools zur Behandlung linearer Systeme und zur Echtzeitsteuerung. SIMULINK stellt das System in der Form eines Blockdiagramms dar: die unterschiedlichen Teilsysteme können mit Hilfe ihrer Übertragungsfunktionen definiert werden – oder sie müssen dynamisch behandelt werden, d.h. ihre zeitliche Entwicklung wird berücksichtigt.

§ 5 Simulationen mit SIMULINK haben gegenüber einer konventionellen numerischen Lösung des Differentialgleichungssystems Vor- und Nachteile: der größte Nachteil ist der theoretische Überbau mit (Teil-)Systemen und Übertragungsfunktionen. Die Verwendung von SIMULINK erfordert also eine gewisse geistige Anfangsinvestition. Der Vorteil des Verfahrens ist die Einfachheit: auch für eine numerische Lösung des Differentialgleichungssystems müssten wir das Blockdiagramm erstellen und dann in jedem Block die DGL des entsprechenden Untersystems mit Hilfe von Euler, Runge–Kutta oder einem anderen Verfahren lösen. Mit SIMULINK müssen wir in die Blöcke nur die Übertragungsfunktion eintragen und sparen uns die numerische Codierung. Außerdem muss die Verknüpfung der Blöcke nicht in einen Programmcode eingearbeitet werden sondern erfolgt direkt – das Risiko von Codierungsfehlern wird verringert. Hat man einmal die Anfangsinvestition getätigt, so erlaubt das Verfahren eine schnelle Simulation. Daher wird es in der Industrie auch im ‘rapid prototyping’ angewendet.

§ 6 Skript und Vorlesung sollen in die hier erwähnten Konzepte einführen und diese an Hand von einfachen, aus der Physik wohlbekannten Beispielen illustrieren – da ich dabei Wert auf Im- und Exporte lege, werden einige Beispiele auch in erweiterter Anwendung diskutiert. Komplexere Probleme können im Rahmen von Projekten bearbeitet werden. Egal, welche Arbeits- und Lernstrategien Sie haben: die Verständnis- und Zwischenfragen sollten Sie ernst nehmen.

# Inhaltsübersicht

<b>I</b>	<b>Der Anfang</b>	<b>12</b>
<b>1</b>	<b>Einführung</b>	<b>13</b>
1.1	Ein einfaches Beispiel: Federpendel . . . . .	13
1.2	Eine erweiterte Form des einfachen Beispiels . . . . .	34
1.3	(Dynamische) Systeme . . . . .	39
1.4	Aufbau von Vorlesung und Skript . . . . .	47
<b>2</b>	<b>SIMULINK – Die Grundlagen</b>	<b>52</b>
2.1	Erste Schritte: Start von SIMULINK . . . . .	53
2.2	Der SIMULINK Library Browser . . . . .	56
2.3	Einfache Modelle: Federpendel und Verwandte(s) . . . . .	67
2.4	(Maskierte) Untersysteme . . . . .	79
2.5	Kommunikation mit dem Workspace . . . . .	90
2.6	Simulationskontrolle . . . . .	94
<b>II</b>	<b>Lineare zeitinvariante Systeme</b>	<b>111</b>
<b>3</b>	<b>Laplace Transformation</b>	<b>112</b>
3.1	Fourier Reihe . . . . .	113
3.2	Fourier Transformation . . . . .	121
3.3	Laplace Transformation . . . . .	140
<b>4</b>	<b>Systemtheorie im Vorbeigehen</b>	<b>147</b>
4.1	Grundbegriffe: Was ist ein System? . . . . .	148
4.2	Lösung der Systemgleichungen . . . . .	153
4.3	Übertragungsfunktion . . . . .	161
<b>5</b>	<b>Übertragungsfunktion vertieft: Filter</b>	<b>170</b>
5.1	Filter . . . . .	171
5.2	Analoge Filter im Zeitbereich . . . . .	173
5.3	Digitale Filter im Zeitbereich . . . . .	182
5.4	Filter im Spektralbereich . . . . .	191
5.5	Filter im Ortsbereich . . . . .	191
<b>III</b>	<b>Dynamische Systeme</b>	<b>194</b>
<b>6</b>	<b>Etwas Chaos</b>	<b>195</b>
6.1	Fraktale: geometrische Annäherung an das Chaos . . . . .	196

6.2	(Seltsame) Attraktoren und Abbildungen . . . . .	201
6.3	Variationen über ein einzelnes Pendel . . . . .	206
6.4	Gekoppelte Federpendel . . . . .	213
<b>7</b>	<b>Diffusions-Reaktions-Modelle</b>	<b>225</b>
7.1	Das Gleichungssystem . . . . .	225
7.2	Anwendungen . . . . .	226
7.3	Lösungsverfahren . . . . .	231
<b>IV</b>	<b>Diskrete Systeme</b>	<b>236</b>
<b>8</b>	<b>Etwas Spieltheorie</b>	<b>237</b>
8.1	Gefangene, Kühe und Stichlinge . . . . .	238
8.2	Klimafolgenabminderung spieltheoretisch . . . . .	246
8.3	Strategien auswählen mit SIMULINK . . . . .	259
<b>V</b>	<b>Anhang</b>	<b>263</b>
<b>9</b>	<b>Einige wichtige Begriffe und Konzepte</b>	<b>264</b>
9.1	Diffusions-Reaktions-Modell . . . . .	264
9.2	Emergenz . . . . .	265
9.3	Morphogenese, biologische . . . . .	267
9.4	Solitonen . . . . .	267
9.5	Synchronisation . . . . .	274
9.6	Schmetterlingseffekt . . . . .	278
<b>10</b>	<b>Nützliche Ergänzungen</b>	<b>280</b>
10.1	Formelzeichen und Abkürzungen . . . . .	280
10.2	Numerische Integration . . . . .	282
10.3	Code-Beispiele aus dem Text . . . . .	290
10.4	Nullstellenbestimmung: Newton-Raphson . . . . .	291
10.5	Etwas (historischer) Hintergrund . . . . .	292
10.6	Lösungen ausgewählter Aufgaben . . . . .	292

# Inhaltsverzeichnis

<b>I</b>	<b>Der Anfang</b>	<b>12</b>
<b>1</b>	<b>Einführung</b>	<b>13</b>
1.1	Ein einfaches Beispiel: Federpendel . . . . .	13
1.1.1	Die analytische Lösung . . . . .	14
	Freie Schwingung . . . . .	14
	Erzwungene Schwingung . . . . .	15
1.1.2	Die numerische Lösung . . . . .	16
	Freie Schwingung . . . . .	17
	Erzwungene Schwingung . . . . .	19
1.1.3	Und die Lösung in SIMULINK? . . . . .	20
	Die Idee . . . . .	20
	Freie Schwingung . . . . .	21
	Erzwungene Schwingung . . . . .	23
1.1.4	Alternative Darstellungen: Übertragungsfunktion und State Space . . . . .	25
	Übertragungsfunktion . . . . .	25
	Lösung der Systemgleichungen mit <b>State Space</b> . . . . .	26
	Zusammenfassung: Gewöhnliche lineare Differentialgleichung in SIMULINK . . . . .	27
1.1.5	Nichtlineare Erweiterung sorgt für Chaos: Fadenpendel . . . . .	28
1.1.6	Nichtlineares Beispiel in konventioneller Darstellung . . . . .	31
1.1.7	Geist ist geil: Im- und Exporte . . . . .	32
1.2	Eine erweiterte Form des einfachen Beispiels . . . . .	34
1.2.1	Die analytische Lösung . . . . .	35
1.2.2	Die numerische Lösung . . . . .	36
1.2.3	Und die Lösung in SIMULINK? . . . . .	36
1.3	(Dynamische) Systeme . . . . .	39
1.3.1	Was ist ein System? . . . . .	39
1.3.2	Struktur oder Aggregat? . . . . .	40
1.3.3	Selbstorganisation . . . . .	41
1.4	Aufbau von Vorlesung und Skript . . . . .	47
<b>2</b>	<b>SIMULINK – Die Grundlagen</b>	<b>52</b>
2.1	Erste Schritte: Start von SIMULINK . . . . .	53
2.2	Der SIMULINK Library Browser . . . . .	56
2.2.1	Blockgruppe Continuous . . . . .	57
	Derivative . . . . .	57
	Integrator . . . . .	58
	State Space . . . . .	60
	Transfer Function . . . . .	60

2.2.2	Blockgruppe Math Operations . . . . .	61
	Divide . . . . .	61
	Gain . . . . .	62
	Math Function . . . . .	63
	Sum . . . . .	63
	Trigonometric Function . . . . .	63
2.2.3	Blockgruppe Sinks . . . . .	64
2.2.4	Blockgruppe Sources . . . . .	65
2.2.5	Ausführen der Simulation . . . . .	66
2.3	Einfache Modelle: Federpendel und Verwandte(s) . . . . .	67
2.3.1	Federpendel: Die Simulation . . . . .	67
2.3.2	Test der Simulation . . . . .	70
2.3.3	Der Export der Daten . . . . .	71
2.3.4	Technische Anwendung: schwingender Container am Kran . . . . .	72
2.3.5	Etwas komplizierter: zwei gekoppelte Federpendel . . . . .	75
	Die Simulation . . . . .	75
	Gibt es Alternativen? . . . . .	76
	Einschub: Ist die Bewegung periodisch oder nicht? . . . . .	77
2.4	(Maskierte) Untersysteme . . . . .	79
2.4.1	Das Untersystem . . . . .	79
2.4.2	Übergabe der Modellparameter . . . . .	80
2.4.3	Maskierte Untersysteme . . . . .	81
	Parameter . . . . .	81
	Initialisierung . . . . .	82
	Dokumentation . . . . .	83
	Icon . . . . .	84
	Das zusammengebaute System . . . . .	84
	Und wohin mit den maskierten Blöcken? . . . . .	84
2.4.4	Bedingte und getriggerte Untersysteme . . . . .	85
2.4.5	Startende Rakete: einfaches zeitvariantes System mit Untersystem . . . . .	86
	Die Bewegungsgleichung . . . . .	87
	Das Modell . . . . .	88
2.5	Kommunikation mit dem Workspace . . . . .	90
2.5.1	GUI oder nicht-GUI? . . . . .	90
	Starten einer Simulation mit <code>sim</code> . . . . .	92
	Veränderung der Simulationsparameter mit <code>simset</code> . . . . .	93
2.6	Simulationskontrolle . . . . .	94
2.6.1	Solver . . . . .	94
	Feste oder variable Schrittweite? . . . . .	96
	Beispiele variable Schrittweite . . . . .	97
2.6.2	Nullstellendetektion . . . . .	98
	Springender Ball . . . . .	98
	Elastischer Ball . . . . .	99
	Diskontinuierliches System: Idealisierte (Zener-)Diode . . . . .	100
	Diskontinuierliches System: Schmitt-Trigger . . . . .	101
2.6.3	Datenim- und -export . . . . .	102
2.6.4	Optimierung . . . . .	103
2.6.5	Diagnostik . . . . .	103
2.6.6	Direkte Rückkopplung: algebraische Schleifen . . . . .	104
	Ein Beispiel mit Memory Block: zwei Wagen Huckepack . . . . .	106
2.6.7	Callbacks . . . . .	108

<b>II</b>	<b>Lineare zeitinvariante Systeme</b>	<b>111</b>
<b>3</b>	<b>Laplace Transformation</b>	<b>112</b>
3.1	Fourier Reihe . . . . .	113
3.1.1	Fourier Reihen anschaulich eingeführt . . . . .	114
	Bestimmung der Fourier Koeffizienten . . . . .	115
	Winkelfunktionen als Basen eines nicht-geometrischen Vektorraums . . . . .	118
	Fourier Reihe und nichtperiodische Funktion . . . . .	119
3.2	Fourier Transformation . . . . .	121
3.2.1	Kontinuierliche Fourier Transformation . . . . .	121
	Fourier Transformation Beispiele . . . . .	122
	Eigenschaften der kontinuierlichen Fourier Transformation . . . . .	125
	Fourier Transformation und harmonischer Oszillator . . . . .	128
3.2.2	Diskrete Fourier Transformation . . . . .	129
	Beispiele . . . . .	130
	Eigenschaften der diskreten Fourier Transformation . . . . .	131
3.2.3	Fast Fourier Transformation FFT . . . . .	132
3.2.4	FFT in MATLAB . . . . .	133
	Überprüfung mit selbsterzeugten Datensätzen . . . . .	134
	Fensterln mit Fourier . . . . .	135
	Verrauschter Fourier . . . . .	137
	Fourier und Faltung . . . . .	138
3.2.5	Fourier in SIMULINK . . . . .	139
3.3	Laplace Transformation . . . . .	140
3.3.1	Laplace Transformation und Lösung der DGLs . . . . .	143
	Gewöhnliche DGL 1. Ordnung: Auf- und Entladen eines Kondensators . . . . .	143
	Partielle Differentialgleichungen . . . . .	144
<b>4</b>	<b>Systemtheorie im Vorbeigehen</b>	<b>147</b>
4.1	Grundbegriffe: Was ist ein System? . . . . .	148
4.1.1	Ein- und Ausgangsgrößen, Zustandsvariable . . . . .	149
4.1.2	LTI-System . . . . .	150
4.1.3	Nicht-Eindeutigkeit der Zustandsvariablen . . . . .	152
4.2	Lösung der Systemgleichungen . . . . .	153
4.2.1	Wiederholung: Lösung im Zeitbereich . . . . .	153
	Homogener Anteil . . . . .	154
	Inhomogener Anteil . . . . .	154
	Gesamtlösung . . . . .	155
	Beispiel: Harmonischer Oszillator . . . . .	156
	Beispiel: Tiefpass . . . . .	157
4.2.2	Lösung mit Hilfe einer Laplace Transformation . . . . .	159
4.3	Übertragungsfunktion . . . . .	161
4.3.1	Wichtig für SIMULINK: Struktur der Übertragungsfunktion . . . . .	163
4.3.2	Rückkopplungen und Übertragungsfunktion . . . . .	164
4.3.3	Eigenfunktionen . . . . .	166
4.3.4	Frequenzgang . . . . .	167
<b>5</b>	<b>Übertragungsfunktion vertieft: Filter</b>	<b>170</b>
5.1	Filter . . . . .	171
5.2	Analoge Filter im Zeitbereich . . . . .	173
5.2.1	Filter 1. Ordnung . . . . .	174
5.2.2	Filter 2. Ordnung . . . . .	176
5.2.3	Filter höherer Ordnung . . . . .	178
	Butterworth Filter . . . . .	178
	Tschebyscheff Filter . . . . .	180



5.2.4	Analoge Filter in SIMULINK . . . . .	181
5.3	Digitale Filter im Zeitbereich . . . . .	182
5.3.1	FIR-Filter . . . . .	182
5.3.2	IIR-Filter . . . . .	183
5.3.3	Gleitendes Mittel . . . . .	183
5.3.4	Gleitendes Mittel in SIMULINK und MATLAB . . . . .	186
5.3.5	Der Filter-Kern . . . . .	187
5.3.6	Die Impulsantwort . . . . .	188
5.3.7	Block Diagramm . . . . .	189
5.3.8	Digitale Filter in SIMULINK . . . . .	190
5.4	Filter im Spektralbereich . . . . .	191
5.4.1	$z$ -Transformation . . . . .	191
5.5	Filter im Ortsbereich . . . . .	191
 <b>III Dynamische Systeme</b>		<b>194</b>
<b>6</b>	<b>Etwas Chaos</b>	<b>195</b>
6.1	Fraktale: geometrische Annäherung an das Chaos . . . . .	196
6.1.1	Feigenbaum Attraktor und Apfelmännchen . . . . .	196
6.1.2	Selbstähnlichkeit und fraktale Dimension . . . . .	198
6.2	(Seltsame) Attraktoren und Abbildungen . . . . .	201
6.2.1	Attraktoren . . . . .	201
6.2.2	Logistische Gleichung . . . . .	202
6.2.3	Lorenz-Attraktor . . . . .	203
6.2.4	Attraktoren in SIMULINK . . . . .	204
	s-Function in SIMULINK . . . . .	206
6.3	Variationen über ein einzelnes Pendel . . . . .	206
6.3.1	Das Feder-Fadenpendel . . . . .	206
	Feder statt Faden . . . . .	207
6.3.2	Fadenpendel an Rolle . . . . .	210
6.3.3	Duffing-Oszillator . . . . .	211
6.4	Gekoppelte Federpendel . . . . .	213
6.4.1	Lineare Geometrie linearer Pendel . . . . .	214
	Alles identisch . . . . .	215
	Unterschiedliche lineare Federn . . . . .	218
6.4.2	Nicht-lineare Version: Toda's Soliton . . . . .	219
6.4.3	Fermi–Pasta–Ulam Problem: MANIAC's Test . . . . .	220
	Einschub: Hamilton-Systeme . . . . .	221
	FPU continued . . . . .	222
<b>7</b>	<b>Diffusions–Reaktions-Modelle</b>	<b>225</b>
7.1	Das Gleichungssystem . . . . .	225
7.2	Anwendungen . . . . .	226
7.2.1	Turing-Muster . . . . .	227
7.2.2	Kannibalen und Missionare oder Jäger–Beute Modelle . . . . .	228
7.2.3	Hirnareale . . . . .	229
7.2.4	Turing-Muster erweitert . . . . .	230
7.2.5	Selbstorganisation auf Oberflächen . . . . .	230
7.2.6	Hodgkins–Huxley-Modell eines Neurons . . . . .	230
7.3	Lösungsverfahren . . . . .	231
7.3.1	Vorübung: Central Pattern Generator CPG . . . . .	233
7.3.2	Hodgkins–Huxley . . . . .	234

<b>IV</b>	<b>Diskrete Systeme</b>	<b>236</b>
<b>8</b>	<b>Etwas Spieltheorie</b>	<b>237</b>
8.1	Gefangene, Kühe und Stichlinge . . . . .	238
8.1.1	Hofstaedter's Einmilliondollarspiel . . . . .	239
8.1.2	Nullsummenspiele . . . . .	240
8.1.3	Das Gefangenendilemma – Einzelspiel . . . . .	240
	Gefangenendilemma mit mehreren Spielern: die Gemeinewiese . . . . .	243
8.1.4	Iteriertes Gefangenendilemma . . . . .	244
8.2	Klimafolgenabminderung spieltheoretisch . . . . .	246
8.2.1	Das Klimasystem spieltheoretisch . . . . .	246
8.2.2	Globales Modell Umwelt und Gesellschaft – Einpersonenspiel . . . . .	247
	Green'sche Funktion des Klimamodells . . . . .	248
	Vorgegebene Szenarien – Response-Funktionen . . . . .	249
	Klimakostenmodell . . . . .	250
	Ergebnisse . . . . .	253
	Erweitertes Modell . . . . .	254
8.2.3	Globales Modell Umwelt und Gesellschaft – Mehrpersonenspiel . . . . .	254
	Gleichberechtigte nicht-kooperative Akteure . . . . .	255
	Ein einzelner Vermeider . . . . .	257
	Handelnde Akteure . . . . .	257
	Zusammenfassung Multi-Actor . . . . .	259
8.3	Strategien auswählen mit SIMULINK . . . . .	259
8.3.1	Strategien . . . . .	260
8.3.2	Beispiele für die SIMULINK-Lösung . . . . .	260
8.3.3	Und welche Strategie ist die Beste? . . . . .	261
<b>V</b>	<b>Anhang</b>	<b>263</b>
<b>9</b>	<b>Einige wichtige Begriffe und Konzepte</b>	<b>264</b>
9.1	Diffusions-Reaktions-Modell . . . . .	264
9.1.1	Diffusion . . . . .	264
9.1.2	Reaktion . . . . .	265
9.2	Emergenz . . . . .	265
9.3	Morphogenese, biologische . . . . .	267
9.4	Solitonen . . . . .	267
9.4.1	Tsunamis . . . . .	267
9.4.2	Parry und die voreilige Kanone . . . . .	268
9.4.3	Russel und Korteweg-de Vries . . . . .	268
9.4.4	Aufsteilen im Flachwasser . . . . .	270
9.4.5	Die mathematische Formulierung . . . . .	271
	Nicht-topologische Solitonen . . . . .	271
	Einhüllende Solitonen . . . . .	272
	Topologische Solitonen . . . . .	273
	Lattice Solitonen . . . . .	274
	Boomeronen . . . . .	274
9.5	Synchronisation . . . . .	274
9.5.1	Zum Einstieg: erzwungene Schwingung . . . . .	275
9.5.2	Zwei gekoppelte Oszillatoren . . . . .	277
9.5.3	Eine Population gekoppelter Oszillatoren . . . . .	278
9.6	Schmetterlingseffekt . . . . .	278

---

<b>10 Nützliche Ergänzungen</b>	<b>280</b>
10.1 Formelzeichen und Abkürzungen	280
10.1.1 Formelzeichen	280
10.1.2 Abkürzungen	281
10.2 Numerische Integration	282
10.2.1 Einfache Verfahren mit fester Schrittweite	282
Euler Verfahren	283
Crank–Nicolson Verfahren	283
Leap-Frog	283
Runge–Kutta	284
10.2.2 Stabilität und Fehler	284
Diskretisierung	284
Konsistenz und Genauigkeit	285
Stabilität	286
Anwendung dieser Begriffe auf die diskutierten Schemata	286
10.2.3 Verfahren mit variabler Schrittweite – Idee	286
10.2.4 Zentrale Differenz	287
Eindimensionaler Fall	287
Zweidimensionaler Fall	289
10.3 Code-Beispiele aus dem Text	290
10.3.1 Euler-Vorwärts für das Federpendel	290
10.4 Nullstellenbestimmung: Newton–Raphson	291
10.5 Etwas (historischer) Hintergrund	292
10.5.1 Block- und Flussdiagramme	292
10.5.2 Flussdiagramme in der Informatik	292
10.6 Lösungen ausgewählter Aufgaben	292

**Teil I**

**Der Anfang**

# Kapitel 1

## Einführung

Do cities learn? Not the individuals who populate cities, not the institutions they foster but the cities themselves. I think the answer is yes. And the silk weavers of Florence can help explain why.

S. Johnson [62], 2001

§ 7 Das Federpendel ist ein einfaches, durch eine DGL beschriebenes physikalisches System. An Hand dieses Beispiels werden die analytische Lösung, eine numerische Lösung sowie die Beschreibung des Systems in SIMULINK vorgestellt. Dabei erhalten wir einen ersten Einblick in letzteres Simulationsverfahren; außerdem können wir mit den anderen Lösungen vergleichen. Das Modell in SIMULINK erlaubt außerdem eine einfache Erweiterung auf den nichtlinearen Fall (z.B. Fadenpendel bei größeren Auslenkungen) und/oder gekoppelte Pendel. Ferner werden wir, zumindest für das Federpendel, alternative Lösungsverfahren kennen lernen: die Übertragungsfunktion und die Systemgleichungen. Ausgehend von diesem Beispiel werden die wesentlichen Grundbegriffe zur Beschreibung (dynamischer) Systeme eingeführt, ebenso wie Beispiele. Des weiteren wird die Struktur des Skripts beschrieben; außerdem finden Sie Hinweise zu Literatur.

**Lernziele** hat dieses Kapitel, da es im wesentlichen eine Wiederholung ist, nicht. Am Ende des Kapitels sollten Sie jedoch, entweder auf der Basis von Vorwissen oder auf Grundlage der folgenden Inhalte,

- gängige Lösungsverfahren für gewöhnliche Differentialgleichungen grob skizzieren können.
- Verfahren zur Lösung einfacher Differentialgleichungssysteme darstellen können.
- einfache Probleme mit Hilfe von Blockdiagrammen darstellen können.
- einfache Systeme mit Hilfe der gebräuchlichen systemtheoretischen Begriffe charakterisieren können.

### 1.1 Ein einfaches Beispiel: Federpendel

§ 8 Beginnen wir mit einem einfachen Beispiel für ein physikalisches System. In den Einführungsvorlesungen sowohl zur Physik als auch zur Mathematik haben wir das Federpendel ausführlich behandelt: ausgehend vom zweiten Newton'schen Axiom erhalten wir im eindimensionalen Fall für die Bewegungsgleichung

$$\begin{aligned} m\ddot{x} &= -kx && \text{rücktreibende Kraft} \\ & -\beta\dot{x} && \text{Reibung} \\ & +F_A && \text{antreibende Kraft} \end{aligned}$$

mit  $m$  als der Masse,  $x$  als dem Ort,  $k$  als der Federkonstante,  $\beta$  als dem Reibungskoeffizienten und  $F_A$  als antreibender Kraft. Vereinfacht lässt sich die Gleichung schreiben als

$$\ddot{x} + 2\gamma\dot{x} + \omega_0^2 x = f_A \quad (1.1)$$

mit  $\gamma = \beta/2m$ ,  $f_A = F_A/m$  sowie  $\omega_0^2 = k/m$  als der Frequenz der freien Schwingung.

§ 9 Gleichung (1.1) enthält als Spezialfall die freie Schwingung. In diesem Fall verschwinden die Dämpfung und die antreibende Kraft und (1.1) reduziert sich auf

$$\ddot{x} + \omega_0^2 x = 0. \quad (1.2)$$

§ 10 Dieses System besteht aus zwei Komponenten, der Masse  $m$  und der Feder mit der Federkonstanten  $k$ . Diese beiden Komponenten sind mit einander gekoppelt in dem Sinne, dass (a) die Feder auf Grund ihrer Auslenkung  $x$  eine rücktreibende Kraft  $F = -kx$  auf die Masse ausübt und diese damit beschleunigt, und (b) die Trägheit der bewegten Masse zu einer Kompression/Auslenkung der Feder führen kann. Das System ist statisch in dem Sinne, dass die Systemparameter  $k$  und  $m$  und die daraus resultierende Frequenz  $\omega_0 = \sqrt{k/m}$  Konstanten sind und sich nicht im Laufe der Bewegung verändern: das System entwickelt sich nicht weiter sondern verhält sich bei gleichen Anfangsbedingungen immer wieder auf die gleiche Weise. Dies gilt natürlich nicht nur für den Spezialfall der freien Schwingung sondern ebenso für eine gedämpfte oder angeregte.

### 1.1.1 Die analytische Lösung

§ 11 Eine analytische Lösung ist stets einer numerischen Lösung vorzuziehen. Zum einen ist die analytische Lösung exakt und leidet nicht unter den numerischen Verfahren inne wohnenden (und in Abschn. 10.2 beschriebenen) Diskretisierungs- und Abbruchfehlern. Zum anderen ist eine analytische Lösung allgemein während eine numerische Lösung stets nur für einen bestimmten Satz von Parametern sowie feste Anfangs- und/oder Randbedingungen bestimmt ist – außer man kann durch die Verwendung von dimensionslosen Variablen zumindest bestimmte Parameterkombinationen in einem Durchgang behandeln. Sonst muss bei jeder Variation eines Parameters die numerische Lösung neu bestimmt werden. Bei der analytischen Lösung dagegen muss nur der Parameter in die einmal bestimmte Lösungsfunktion eingesetzt werden. Daher sind die zum Verständnis eines Systems so hilfreichen Parameterstudien mit analytischen Lösungen wesentlich leichter durchzuführen als mit numerischen.

§ 12 Das Federpendel zeigt sich der analytischen Lösung leicht zugänglich. Auch das nächstkomplexe System, zwei gekoppelte Federpendel, lassen sich formal ähnlich behandeln. Mit komplexeren Systemen, wie z.B. dem Klimasystem, wird es jedoch schwierig: mathematisch wird dieses durch einen großen Satz gekoppelter partieller Differentialgleichungen beschrieben.<sup>1</sup> Hier sind analytische Lösungen nicht, oder bei einfacheren Systemen nur durch einschränkende Annahmen, möglich.

### Freie Schwingung

§ 13 Wenden wir uns jetzt den analytischen Lösungen des Federpendels zu. Betrachten wir als erstes die Lösung für den simplen Fall der freien Schwingung. Mit dem bei DGLs mit konstanten Koeffizienten stets anwendbaren Exponentialansatz  $x \sim e^{\lambda t}$  ergibt sich nach Einsetzen in die DGL (1.2) für die Eigenwerte  $\lambda_{1,2} = \pm i\omega_0$ . Damit ist die komplexe Lösung

$$z(t) = A e^{\omega_0 t} + B e^{-\omega_0 t}.$$

<sup>1</sup>Genauer handelt es sich um mehrere gekoppelte Sätze gekoppelter Systeme von DGLs, da das Klimasystem schon natürlich in mehrere Untersysteme (Ozean, Atmosphäre, Kryosphäre, Boden ...) mit eigenen Zeitkonstanten und jeweils unterschiedlichen räumlichen Skalen zerfällt. Zusätzlich werden für die Beschreibung einzelner Aspekte jedes Untersystems, wie z.B. Dynamik oder Chemie, jeweils unterschiedliche, nur schwach mit einander gekoppelte Systeme von DGLs benötigt.

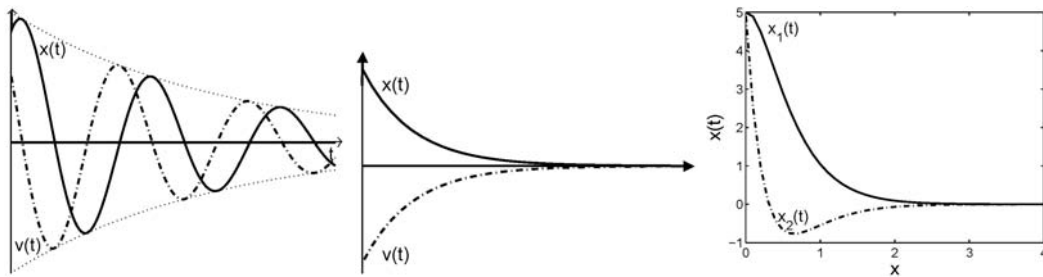


Abbildung 1.1: Lösungen für die DGL der gedämpften Schwingung,  $\ddot{x} + 2\gamma\dot{x} + \omega_0^2 = 0$ . *Links:* Schwingfall ( $\gamma^2 < \omega_0^2$ ), *Mitte:* Kriechfall ( $\gamma^2 > \omega_0^2$ ), *Rechts:* aperiodischer Grenzfall ( $\gamma^2 = \omega_0^2$ )

Darin sind  $A$  und  $B$  komplexwertigen Integrationskonstanten. Physikalisch sinnvoll ist nur der Realteil der Lösung:

$$x(t) = \Re(z(t)) = a \cos(\omega_0 t) + b \sin(\omega_0 t)$$

mit den reellen Integrationskonstanten  $a$  und  $b$  – zu bestimmen aus den Anfangsbedingungen.

### Erzwungene Schwingung

§ 14 Die analytische Lösung der inhomogenen DGL (1.1) ergibt sich als Superposition der allgemeinen Lösung der homogenen DGL

$$\ddot{x} + 2\gamma\dot{x} + \omega_0^2 x = 0 \quad (1.3)$$

sowie einer speziellen Lösung der inhomogenen DGL.

§ 15 Die allgemeine Lösung der homogenen DGL lässt sich wie bei der freien Schwingung mit einem Exponentialansatz  $x \sim e^{\lambda t}$  bestimmen. Einsetzen in die DGL liefert als Bestimmungsgleichung für die Eigenwerte  $\lambda$ :

$$\lambda_{1,2} = -\gamma \pm \sqrt{\gamma^2 - \omega_0^2}.$$

Formal wie physikalisch ergeben sich in Abhängigkeit vom Radikand  $\gamma^2 - \omega_0^2$  drei verschiedene Lösungen (siehe auch Abb. 1.1):

1. Schwingfall: hier ist  $\gamma^2 < \omega_0^2$ , so dass der Radikand negativ wird. Die Eigenwerte sind komplex

$$\lambda_{1,2} = -\gamma \pm i\omega \quad \text{mit} \quad \omega = \sqrt{\omega_0^2 - \gamma^2}$$

und die allgemeine Lösung der DGL ist:

$$z(t) = Ae^{(-\gamma+i\omega)t} + Be^{(-\gamma-i\omega)t} = e^{-\gamma t} [Ae^{i\omega t} + Be^{-i\omega t}].$$

Physikalisch sinnvoll ist nur der Realteil

$$x_h(t) = \Re(z(t)) = e^{-\gamma t} [a \cos(\omega t) + b \sin(\omega t)].$$

Die eckige Klammer entspricht der Lösung der freien Schwingung, der Vorfaktor beschreibt das Abklingen der Amplitude auf Grund der Dämpfung.

2. Kriechfall: hier ist  $\gamma^2 > \omega_0^2$ , so dass der Radikand positiv ist. Die Eigenwerte sind reell

$$\lambda_{1,2} = -\gamma + \tilde{\omega} \quad \text{mit} \quad \tilde{\omega} = \sqrt{\gamma^2 - \omega_0^2}$$

und die Lösung wird

$$x_h(t) = e^{-\gamma t} [a e^{\tilde{\omega} t} + b e^{-\tilde{\omega} t}].$$

Hier ist die Dämpfung so stark, dass sich der Körper asymptotisch der Ruhelage annähert ohne durch diese hindurch zu schwingen: die Lösung hat keine Nullstelle.

3. aperiodischer Grenzfall: hier ist  $\gamma^2 = \omega_0^2$ , so dass sich nur ein Eigenwert  $\lambda = -\gamma$  ergibt. Die zweite Lösung lässt sich mit Hilfe des Ansatz  $x(t) \sim t e^{\lambda t}$  finden. Als Gesamtlösung ergibt sich

$$x_h(t) = e^{-\gamma t} [a + bt] .$$

In diesem Fall nähert sich der Körper ebenfalls der Ruhelage an, kann jedoch noch einmal durch diese hindurchschwingen. Ein zweites Durchschwingen erfolgt nicht – würde dieses erfolgen, so ließe sich die Periodendauer bzw. Frequenz der Schwingung bestimmen und die Lösung würde durch den Schwingfall beschrieben.

Alle drei Varianten haben den Vorfaktor  $e^{-\gamma t}$  gemeinsam, d.h. für  $t \rightarrow \infty$  verschwindet die Lösung. Oder in anderen Worten: im stationären Fall ist die Lösung des homogenen Anteils der DGL (1.1) irrelevant, die Bewegung ist durch die spezielle Lösung der inhomogenen DGL bestimmt, siehe auch den linken Teil von Abb. 1.2.

§ 16 Die spezielle Lösung der inhomogenen DGL erhalten wir, in dem wir einen Ansatz machen, der der Inhomogenität  $f_A$  ähnlich ist. Nehmen wir z.B. als Inhomogenität eine periodische Funktion  $f_A = f_a e^{i\Omega t}$ . Der Ansatz für die spezielle Lösung der inhomogenen DGL wird dieser Inhomogenität ähnlich gewählt;  $z(t) = A^c e^{i\Omega t}$  mit  $A^c$  als gegebenenfalls komplexer Amplitude. Ableiten und einsetzen in die DGL liefert

$$(-\Omega^2 + 2i\gamma\Omega + \omega_0^2)A^c = f_a$$

und damit für den Zusammenhang zwischen der Amplitude  $f_a$  der antreibenden Kraftdichte und der Amplitude  $A$  der erzwungenen Schwingung

$$A = \sqrt{A^c A^{c*}} = \frac{f_a}{\sqrt{(\omega_0^2 - \Omega^2)^2 + (2\gamma\Omega)^2}} ,$$

wie im mittleren Teil von Abb. 1.2 dargestellt. Die Phase ergibt sich, wie im rechten Teil von Abb. 1.2 gezeigt, zu

$$\phi = -\arctan \frac{\Im(A^c)}{\Re(A^c)} = -\arctan \frac{2\gamma\Omega}{\omega_0^2 - \Omega^2} .$$

Als partikuläre Lösung der inhomogenen DGL ergibt sich damit

$$x_p = A \cos(\Omega t + \phi) .$$

Nach dem durch die Lösung der homogenen DGL beschriebenen Einschwingen, d.h. für  $t \rightarrow \infty$ , hat die erzwungene Schwingung die Frequenz der antreibenden Kraft, jedoch verschoben um den Phasenwinkel  $\phi$ . Dieser hängt ebenso wie die Amplitude  $A$  von der Dämpfung  $\gamma$  ab: für verschwindende Dämpfung ergibt sich bei  $\omega_0 = \Omega$  die maximale Phasenverschiebung, ebenso geht die Amplitude gegen  $\infty$  (Resonanzkatastrophe).

**Verständnisfrage 1** Die freie Schwingung des Federpendels beschreiben wir physikalisch als kontinuierliche Umwandlung von kinetischer in potentielle Energie und umgekehrt. Ist eine Resonanzkatastrophe energetisch eigentlich möglich? Oder gibt es in einem realen System selbst bei verschwindender Dämpfung eine Begrenzung der Amplitude?

## 1.1.2 Die numerische Lösung

§ 17 Gleichung (1.1) ist eine DGL zweiter Ordnung. Zu ihrer numerischen Lösung stehen uns zwei prinzipiell unterschiedliche Ansätze zur Verfügung:

- Zerlegung der DGL zweiter Ordnung in zwei DGLs erster Ordnung und alternierende Lösung dieser beiden Gleichungen, z.B. durch Euler, Leapfrog oder Runge–Kutta (vgl. Abschn. 10.2.1).
- geeignete Diskretisierung der zweiten Ableitung und Lösung der DGL in einem Schritt, in der Regel in einem impliziten Verfahren.



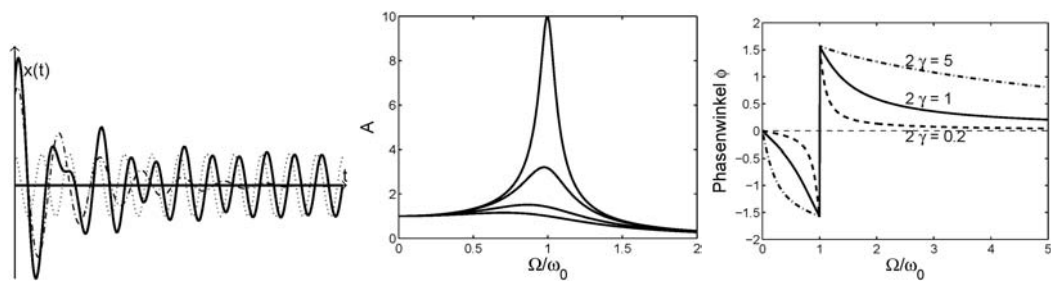


Abbildung 1.2: Erzwungene Schwingung *Links*: Einschwingverhalten, *Mitte*: Amplitude in Abhängigkeit vom Verhältnis  $\Omega/\omega_0$  der Frequenzen der antreibenden Kraft und der freien Schwingung, *Rechts*: Phasenverschiebung in Abhängigkeit von  $\Omega/\omega_0$

Letzteres Verfahren wird meist nur in speziellen Vorlesungen über numerische Verfahren oder Modellierung eingeführt (z.B. [65], siehe auch § 949) – wir werden es hier daher nicht weiter verfolgen; außerdem werden wir in SIMULINK den ersten Ansatz wieder antreffen, so dass sich seine Wiederholung hier lohnt.

### Freie Schwingung

§ 18 Die Zerlegung der DGL zweiter Ordnung in zwei DGLs erster Ordnung lässt sich durch Einführung einer Hilfsvariablen  $v = \dot{x}$  durchführen:

$$\dot{x} = v \quad \text{und} \quad \dot{v} = -\omega_0^2 x.$$

§ 19 Diese formale Zerlegung lässt sich anschaulich in ein Lösungsverfahren übersetzen (siehe auch [63, 66]). Diskretisierung der beiden DGLs liefert

$$\frac{\Delta x}{\Delta t} = v \quad \text{und} \tag{1.4}$$

$$\frac{\Delta v}{\Delta t} = -\omega_0^2 x. \tag{1.5}$$

Nach der Wahl einer geeigneten Schrittweite  $\Delta t$  ist das folgende Schema zu durchlaufen:

1. Bestimme gemäß (1.5) die Beschleunigung  $\dot{v}_i$  aus der Auslenkung  $x_i$ ; beim ersten Schritt wird der Anfangswert  $x_0$  verwendet. Bestimme daraus die Geschwindigkeitsänderung  $\Delta v$  während des Zeitintervalls  $\Delta t$  gemäß  $\Delta v = \dot{v}_i \Delta t = -\omega_0^2 x_i \Delta t$ . Bestimme daraus die neue Geschwindigkeit  $v_{i+1}$  am Ende des Zeitintervalls:

$$v_{i+1} = v_i + \Delta v = v_i + \omega_0^2 x_i \Delta t.$$

Damit ist die zweite der beiden DGLs für diesen Zeitschritt abgearbeitet.

2. Bestimme nun gemäß (1.4) die Änderung des Ortes während des Zeitintervalls:  $\Delta x = v_{i+1} \Delta t$ . Bestimme daraus den neuen Ort

$$x_{i+1} = x_i + \Delta x = x_i + v_{i+1} \Delta t.$$

Damit ist die erste der beiden DGLs für diesen Zeitschritt abgearbeitet.

3. die Zeit wird nun um  $\Delta t$  erhöht und die Schleife erneut durchlaufen – sofern nicht eine Abbruchbedingung erfüllt ist.

**Verständnisfrage 2** In dieser Darstellung wurde ein explizites Verfahren (Euler Vorwärts) verwendet. Ist die Zerlegung in zwei Gleichungen zwangsläufig an ein explizites Verfahren gebunden oder könnte diese auch in einem impliziten Verfahren durchgeführt werden?

**Wiederholungsfrage 1** Erläutern Sie den Unterschied zwischen impliziten und expliziten Verfahren.

**Wiederholungsfrage 2** Rufen Sie sich andere als das bereits bekannte Euler-Verfahren ins Gedächtnis zurück.

§ 20 Die Grundidee dieses Lösungsschemas lässt sich in einem Blockdiagramm anschaulich zusammen fassen. Ein derartiges Blockdiagramm entspricht dem vor der Codierung zu erstellenden Flussdiagramm. Zwar ist für letzteres die Verwendung der einzelnen Zeichen durch Normen geregelt (in diesem Fall DIN 66001),<sup>2</sup> wir verwenden nur eine einfache schematische Form; die Liste der wichtigsten Symbole finden Sie in Abschn. 10.5.1.

§ 21 Die im linken Teil von Abb. 1.3 dargestellte Variante entspricht der obigen verbalen Darstellung des Schemas. Nach dem Start (Oval) erfolgt die Eingabe (Parallelogramm) der Parameter. Anschließend wird die erste Operation (Rechteck) durchgeführt: die neue Geschwindigkeit wird gemäß (1.5) berechnet. In der folgenden Operation wird der neue Ort gemäß (1.4) berechnet. Anschließend erfolgt eine Verzweigung (Raute), die z.B. explizit über eine Abbruchbedingung gesteuert wird (z.B. vorgegebene Endzeit) oder implizit durch die bei der Berechnung zu durchlaufende Schleife. Ist die Abbruchbedingung nicht erreicht, wird auf die erste Operation zurück gesprungen, sonst erfolgt die Ausgabe (Parallelogramm) und das Programm wird beendet (Oval).

§ 22 Im rechten Teil der Abbildung ist der erste Block in der Schleife in zwei Teile aufgebrochen, d.h. die erste Operation wird in zwei Teile aufgespalten: es wird in einer ersten Operation die Beschleunigung bestimmt als  $a = -kx/m$ . Anschließend wird daraus die Geschwindigkeit bestimmt als  $v_{i+1} = v_i + a \Delta t$ , d.h. die Gleichung  $\dot{v} = a$  wird integriert.

§ 23 Die rechte Variante unterscheidet sich inhaltlich nicht von der kürzeren Variante im linken Teil von Abb. 1.3. Während letztere einen kompakteren Programmcode erzeugt, ist erstere konzeptuell befriedigender: in der ersten Operation erfolgt eine einfache Multiplikation während die beiden folgenden Operationen jeweils Integration einer homogenen DGL erster Ordnung mit konstantem Koeffizienten sind. Für die Lösung des Problems in SIMULINK wird sich die zweite Variante als vorteilhaft erweisen.

§ 24 Vor der Codierung in einer Programmiersprache müssen wir uns jetzt noch für ein numerisches Verfahren (vgl. Abb. 10.2) entscheiden – wir wählen das einfachste (und auch in der verbalen Beschreibung schon nahe gelegte) Verfahren, das Euler Vorwärts Verfahren. Eine mögliche Codierung in MATLAB ist in Abschn. 10.3.1 gegeben.

§ 25 Abbildung 1.4 zeigt die Lösungen für Ort  $x(t)$  (rot) und Geschwindigkeit  $v(t)$  (grün) für eine freie Schwingung mit verschiedenen Schrittzahlen  $n$  während einer Periode. Für  $n = 10$  ist das Ergebnis offensichtlicher Blödsinn;  $n = 100$  liefert zwar schon eine Schwingung, jedoch ist das Ergebnis auf Grund der anwachsenden Amplitude für die gegebene Bewegungsgleichung physikalisch nicht sinnvoll: da dem System weder Energie von außen zugeführt noch Energie durch Reibung in Wärme umgewandelt wird, sollte die Summe der mechanischen Energien  $\frac{1}{2}kx^2$  und  $\frac{1}{2}mv^2$  konstant sein. Selbst bei  $n = 1000$  (nicht gezeigt) ist noch ein leichtes Anwachsen der Amplitude erkennbar. Erst für  $n = 10000$  gibt sie numerische Lösung die analytische (und wichtiger noch den physikalischen Sachverhalt) korrekt wieder: die Zeitkonstante  $T$  wird bei diesem sehr einfachen numerischen Verfahren (Euler Vorwärts) also in ungefähr 4000 Schritte der Länge  $\Delta t$  unterteilt.

§ 26 Diese kurze Diskussion soll an einen wichtigen Sachverhalt erinnern: bei einer numerischen Lösung ist nicht nur die korrekte Codierung der Lösung wichtig sondern auch die korrekte Wahl der Schrittweite(n) des Lösungsschemas.

<sup>2</sup>Die DIN Norm (zumindest die Variante von 1996, heute ist es wahrscheinlich eine EU Norm) finden Sie z.B. unter <http://www.fh-jena.de/~kleine/history/software/DIN66001-1966.pdf>.

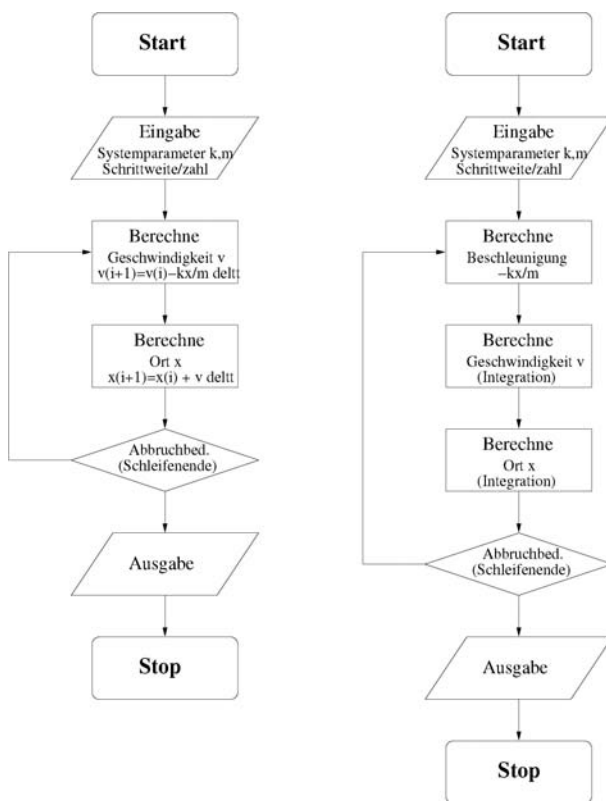


Abbildung 1.3: Flussdiagramm zur freien Schwingung

### Erzwungene Schwingung

§ 27 Zur Lösung der DGL (1.1) der erzwungenen Schwingung bedienen wir uns des gleichen Tricks wie bei der freien Schwingung: die DGL zweiter Ordnung wird auf zwei DGLs erster Ordnung zurück geführt. Wieder mit  $\dot{x} = v$  ergeben sich die beiden DGLs

$$\dot{v} = -2\gamma v - \omega_0^2 x + f_A \quad \text{und} \quad (1.6)$$

$$\dot{x} = v. \quad (1.7)$$

In der ersten Gleichung wird wieder die Beschleunigung aus der Summe der wirkenden Kräfte bestimmt und daraus durch Integration die Geschwindigkeit; in der zweiten Gleichung wird durch nochmalige Integration der Ort bestimmt.

§ 28 Die Struktur des Gleichungssystems entspricht der des Systems der freien Schwingung, d.h. wir können die numerische Lösung entsprechend dem Flussdiagramm im rechten Teil von

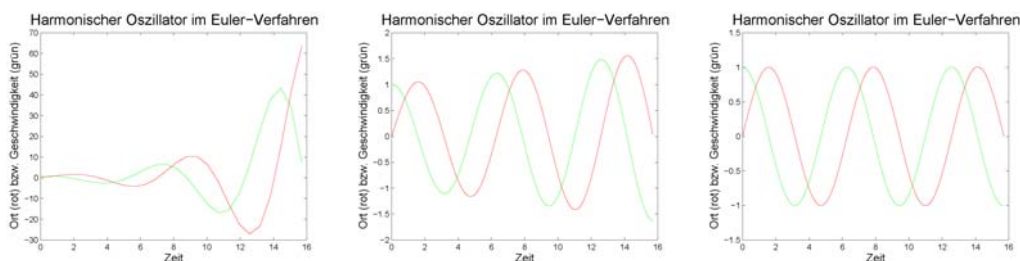


Abbildung 1.4: Numerische Lösungen zur freien Schwingung mit dem in § 974 gegebenen Code für Schrittzahlen  $n$  von 10 (*links*), 100 (*Mitte*) und 10 000 (*rechts*). Für die beiden kleineren Schrittzahlen liefert das numerische Verfahren unkorrekte Lösungen – in diesem Beispiel ist der Fehler glücklicherweise offensichtlich



Abbildung 1.5: Erzwungene Schwingung (numerische Lösungen): der Einschwingvorgang mit dem Übergang zur erzwungenen Schwingung (*links*); ohne externe Anregung ergibt sich eine gedämpfte Schwingung, entsprechend der Lösung der homogenen DGL (*Mitte*); mit verschwindender Reibung und Anregung, entsprechend der freien Schwingung (*rechts*)

Abb. 1.3 vornehmen. Die Struktur bleibt erhalten, lediglich die Operation im ersten Rechteck wird erweitert, da die Beschleunigung nicht mehr alleine durch  $-kx/m$  gegeben ist sondern durch  $-kx/m - \beta/m + f_A$ . Entsprechend können wir den MATLAB-Code aus § 974 nahezu unverändert übernehmen, siehe § 977.

§ 29 Abbildung 1.5 zeigt Lösungen dieses numerischen Schemas. Im linken Teil (mit dem Parametersatz  $k=1; m=1; \beta=0.5; f_{amp}=1; \Omega=2$ ) ergibt sich das Einschwingen mit dem Übergang zur erzwungenen Schwingung, d.h. die Lösung der inhomogenen DGL (1.1). Im mittleren Teil ist die Amplitude (pro Masse)  $f_{amp}$  der externen Kraft auf Null gesetzt, so dass sich (1.1) auf die homogene DGL (1.3) reduziert. Die hier gezeigte Lösung repräsentiert einen Schwingfall. Zur Überprüfung der Schrittweite des numerischen Schemas (zum Problem vgl. Abb. 1.1) ist im rechten Teilbild auch die Dämpfung  $\beta$  ausgeschaltet, so dass sich die Lösung der freien Schwingung ergibt – an dieser lässt sich die Schrittweite am einfachsten überprüfen, da die Amplituden konstant bleiben müssen.

**Verständnisfrage 3** Ist die Argumentation bezüglich der Schrittweite korrekt? Muss eine für die freie Schwingung numerisch sinnvolle Schrittweite in  $\Delta t$  zwingend auch für die Dämpfung sinnvoll sein (oder für die antreibende Kraft)?

### 1.1.3 Und die Lösung in SIMULINK?

§ 30 **Don't Panic!** Dieser Abschnitt wirft Sie einfach in SIMULINK hinein. Verheddern Sie sich nicht in Details, lassen Sie sich einfach auf die Idee ein! Die formalen Grundlagen lernen Sie in Kap. 3 kennen, das Handwerkszeug in Kap. 2.<sup>3</sup>

#### Die Idee

§ 31 SIMULINK ist ein dem modernen PC-Nutzer insofern entgegen kommendes Verfahren, als dass es auf dem Platzieren von Icons durch 'Drag and Drop' basiert. Diesen Icons müssen Aufgaben/Funktionen zugewiesen werden, so dass Funktionsblöcke entstehen. Diese Blöcke müssen Sie geeignet miteinander verbunden werden, um den Lauf eines Signals durch die Funktionsblöcke zu repräsentieren.<sup>4</sup> Optisch sieht die SIMULINK-Eingabe damit wie ein Block- oder Flussdiagramm aus. Für das bisher betrachtete Beispiel können wir daher auf das Flussdiagramm in Abb. 1.3 (oder genauer die rechte Version davon) zurück greifen.

<sup>3</sup>Die Inhalte von Kap. 2 werden wir im Rahmen der Vorlesung nicht systematisch durchgehen. Betrachten Sie das Kapitel stattdessen als ein einfaches Hilfsmittel zum Verständnis der Beispiele. Lediglich Abschn. 2.1 sollte gegebenenfalls schnell überflogen werden, um die im Text gezeigten Beispiele nachvollziehen zu können.

<sup>4</sup>In dieser Beschreibung erkennen Sie bereits eine weitere Möglichkeit der Anwendung von SIMULINK, die Signalanalyse bzw. Signalbearbeitung z.B. mit Hilfe von Filtern – wir werden kurz in Kap. 5 darauf eingehen.

§ 32 Während das Block- oder Flussdiagramm in der konventionellen Programmierung nur ein Hilfsmittel zur Strukturierung des Codes ist, ist das in SIMULINK erstellte Blockdiagramm ausführbar: die normalerweise der Erstellung des Flussdiagramms folgende Auswahl eines numerischen Verfahrens und dessen Codierung entfällt.<sup>5</sup> SIMULINK nimmt uns also nicht das Verständnis des Systems und eine Zerlegung eines komplexeren Systems in Teilsysteme (oder einer DGL zweiter Ordnung in zwei DGLs erster Ordnung) ab, es erspart uns aber die zeitaufwendige und fehleranfällige Codierung. Dadurch ist es für den Nutzer einfacher, sich auf die Struktur und das Verständnis seines Systems zu konzentrieren anstelle alle Energie auf die Vermeidung von Tippfehlern oder Syntaxprobleme aufzuwenden.

### Freie Schwingung

§ 33 Wenden wir diese etwas abstrakte Idee auf die freie Schwingung an. Dazu erstellen wir in SIMULINK ein entsprechendes Blockdiagramm, siehe auch Abb. 1.6. Das Flussdiagramm im rechten Teil von Abb. 1.3 besteht aus vier funktionalen Teilen, jeweils angedeutet durch die rechteckigen, eine Operation bezeichnenden Blöcke:

1. die Bestimmung der Beschleunigung mit Hilfe des Systemparameters  $\omega_0^2 = -k/m$ .
2. die Bestimmung der Geschwindigkeit durch Integration der Beschleunigung (unter Berücksichtigung des Anfangswertes bzw. des Wertes aus dem voran gegangenen Schritt).
3. die Bestimmung des Ortes durch Integration der Geschwindigkeit (unter Berücksichtigung des Anfangswertes bzw. des Wertes aus dem voran gegangenen Schritt).
4. die Darstellung der Ergebnisse.

§ 34 Diese Blöcke repräsentieren drei Funktionen: die Multiplikation mit einer Konstanten im ersten Block, die Integration in den beiden folgenden Blöcken und die Darstellung im letzten Block. Die entsprechenden Funktionen lassen sich in SimuLink realisieren:

1. die Multiplikation mit einer Konstanten wird durch ein als **Gain** (siehe § 170ff) bezeichnetes Dreieck symbolisiert.
2. die Integration wird durch ein Rechteck mit dem Eintrag  $\frac{1}{s}$ , den Integrator (siehe § 154ff), repräsentiert.
3. die Darstellung der Ergebnisse wird durch ein **Scope** (in Anlehnung an ein Oszilloskop, siehe § 181, alternativ können Sie auch einen xy-Schreiber zur Darstellung erhalten) symbolisiert.

Pfeile zwischen den Blöcken geben den Signalverlauf bzw. den Fluss der Variablen durch diese Blöcke. Den verschiedenen Blöcken können Parameter übergeben werden (siehe § 147): so muss dem Block **Gain** verständlicherweise der Faktor übergeben werden, mit dem zu multiplizieren ist während den Integrationsblöcken die jeweiligen Anfangsbedingungen zu übergeben sind.

§ 35 Die hier beschriebenen Blöcke sind in Abb. 1.6 zur Simulation der freien Schwingung zusammen gesetzt. Steigen wir unten beim Dreieck ein und folgen dem durch die Pfeile gegebenen Signalverlauf:

1. rechts des Dreiecks ist das Signal das Ausgangssignal, d.h. der Weg  $x$ . Dieses wird im Block **Gain** mit dem Systemparameter  $\omega_0^2 = -k/m$  multipliziert; wir erhalten also die Beschleunigung. In diesem Fall ist  $k = 1$  und  $m = 1$  und damit  $\omega_0^2 = -1$  (alles dimensionslos).
2. die sich so ergebende Beschleunigung wird integriert (erster Integrator-Block, links im Diagramm in Abb. 1.6), um die Geschwindigkeit zu erhalten.
3. diese wird nochmals integriert (zweiter Integrator-Block, in der Mitte im Diagramm), um den Ort zu erhalten.

<sup>5</sup>Die Auswahl des Verfahrens entfällt nur scheinbar. SIMULINK kennt ein Default-Verfahren – solange Sie nicht explizit ein anderes Lösungsverfahren auswählen, wird dieses verwendet. Eine Übersicht über die zur Verfügung stehenden Lösungsverfahren finden Sie in Abschn. 2.6.1.

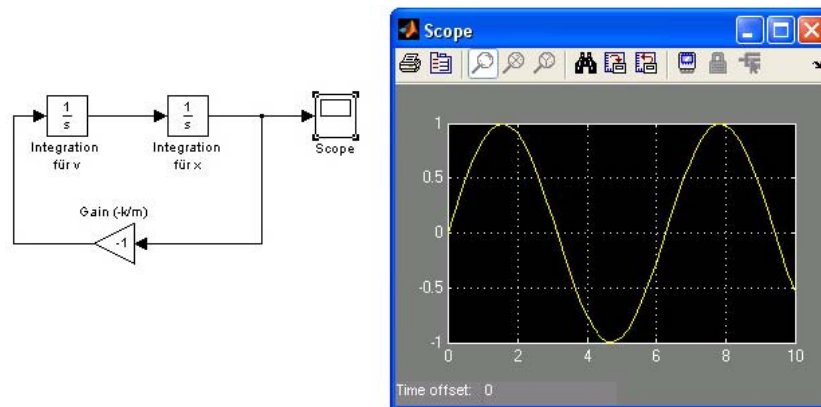


Abbildung 1.6: Freie Schwingung in SIMULINK: der dreieckige Block **Gain** bestimmt die Beschleunigung aus der Auslenkung  $x$ , die folgende Integration (rechteckiger Block) liefert die Geschwindigkeit, nochmalige Integration (zweiter rechteckiger Block) liefert den Ort und die Schleife wird erneut durchlaufen

4. dieser geht einerseits auf die Ausgabe (Scope), andererseits wird er zum nochmaligen Durchlauf der Schleife benötigt. Die einzelnen Integratorblöcke berücksichtigen den jeweiligen Zustand des Systems, d.h. in ihnen müssen auch die Anfangsbedingungen angegeben werden.<sup>6</sup>

Nach erfolgter Simulation lässt sich das Ergebnis durch einen Doppelklick auf das Symbol **Scope** graphisch darstellen, wie im rechten Teil von Abb. 1.6 angedeutet.

**Verständnisfrage 4** Und was passiert, wenn Sie die Angabe der Anfangsbedingungen vergessen?

§ 36 Im Gegensatz zu den vielen Zeichen in der numerischen Lösung wie in § 974 lässt sich in SIMULINK die Lösung durch die Positionierung von 4 Blöcken, deren Verbindung durch 4 Linien sowie die Angabe des Faktors für den Gain (und die Anfangsbedingungen) realisieren: weniger als 20 Mausklicks und Zeichen vs. einige hundert Zeichen in der numerischen Lösung. Daher die gute Eignung von SIMULINK oder entsprechenden symbolisch orientierten Simulationsverfahren für rapid prototyping bzw. genauer rapid application development.

§ 37 Während der Block **Gain** in Abb. 1.6 als Multiplikator den Faktor  $-k/m$  explizit enthält, ist die symbolische Darstellung der Integration durch einen Block mit dem Multiplikator  $\frac{1}{s}$  nicht ganz so offensichtlich. Diese Notation entstammt der Laplace Transformation: mit Hilfe dieser in Kapitel 3 genauer beschriebenen Integraltransformation kann eine Funktion  $x(t)$  aus dem Zeitbereich in eine Funktion  $X(s)$  in Abhängigkeit von einer (komplexen) Frequenz  $s$  transformiert werden. Der Charme dieser Transformation liegt darin, dass Differentiation und Integration durch die algebraischen Operationen Multiplikation und Division ersetzt werden können. Damit wird eine Differentialgleichung in eine algebraische Gleichung überführt. Während bei einer einfachen gewöhnlichen DGL der Aufwand mit der Laplace-Transformation recht hoch erscheint, wird ihr Vorteil bei einem System partieller DGLs offensichtlich: dieses wird einfach in ein System algebraischer Gleichungen umgewandelt; seine Lösung lässt sich damit auf die Lösung eines Gleichungssystems zurück führen.

§ 38 Ist unsere Interpretation der Blöcke  $\frac{1}{s}$  als Integration korrekt, so sollte das Signal zwischen den beiden Integrationsblöcken der Geschwindigkeit entsprechen. Da wir bei den analytischen und numerischen Lösungen stets Weg und Geschwindigkeit in Abhängigkeit von

<sup>6</sup>Die Anfangsbedingungen werden, wie in jedem konventionellen numerischen Verfahren, natürlich nur im ersten Schritt benötigt – in allen folgenden Zeitschritten wird als Anfangswert jeweils der Wert aus dem voran gegangenen Zeitschritt verwendet.

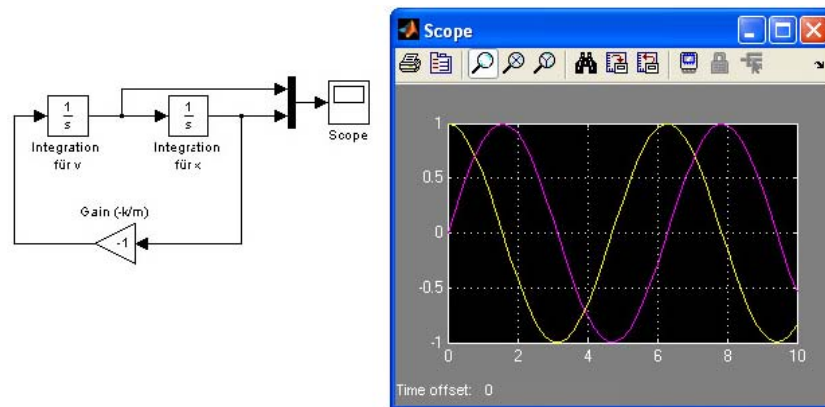


Abbildung 1.7: Freie Schwingung in SIMULINK: das Signal wird zusätzlich bereits nach der ersten Integration abgegriffen und gemeinsam mit dem Ausgangssignal über einen Multiplexer auf das Scope gegeben; Vorsicht Farbwechsel: Ort in Magenta, Geschwindigkeit in gelb

der Zeit betrachtet haben, sollten wir auch in SIMULINK das Signal zwischen den beiden Integratoren zusätzlich betrachten. Dazu können wir es entweder auf ein eigenes Scope geben (dann haben wir jeweils ein getrenntes Weg-Zeit- und ein Geschwindigkeits-Zeit-Diagramm) oder wir geben das Signal zusammen mit dem Ausgangssignal über einen Multiplexer Mux auf das Scope, um beide in einem Graphen darzustellen.

§ 39 Eine weitere Darstellungsmöglichkeit bietet die Ausgabe beider Größen auf den xy-Graph (siehe § 182). Dieser hat zwei Eingänge, d.h. der Multiplexer wird nicht benötigt. Die Darstellung erfolgt im Phasenraum – diese Variante ist für die freie Schwingung langweilig; wir werden bei der erzwungenen Schwingung in Abb. 1.9 darauf zurück kommen.

### Erzwungene Schwingung

§ 40 Die in Abb. 1.6 dargestellte Simulation einer freien Schwingung lässt sich leicht auf eine erzwungene Schwingung erweitern. Der wesentliche Unterschied liegt an der gleichen Stelle wie bei den numerischen Lösungen: vor der ersten Integration zu Bestimmung der neuen Geschwindigkeit müssen wir jeweils die sich aus allen wirkenden Kräften ergebenden Beschleunigungen aufsummieren. Zu den bereits bei der gedämpften Schwingung aus den Systemzuständen bestimmten Beschleunigungen müssen wir die durch die externe Kraft bewirkte Beschleunigung addieren. Dazu benötigen wir einen Summationsblock `sum` (siehe § 175ff), angezeigt durch den Kreis mit den Summenzeichen in Abb. 1.8.

§ 41 Die durch die rücktreibende Kraft bewirkte Beschleunigung kennen wir bereits aus der freien Schwingung: sie hängt von  $x$  ab, d.h. sie wird durch Multiplikation mit einem Gain aus dem Wert nach der zweiten Integration bestimmt (oberer Gain-Block in Abb. 1.8). Die Verzögerung durch Reibung hängt von der Geschwindigkeit ab, d.h. wir müssen das Ergebnis der ersten Integration multipliziert mit einem Faktor  $\beta/m = 2\gamma$  auf den Summationsblock führen. Im Blockdiagramm in Abb. 1.8 ist das durch den unteren Gain-Block angegeben. Als letzte Kraft benötigen wir die externe Kraft, in Abb. 1.8 durch die Sinus-Quelle realisiert. Ob wir zwischen Quelle und Summenblock einen Gain-Block einführen, der mit  $1/m$  multipliziert, oder diesen Faktor direkt in der Quelle berücksichtigen, ist Geschmackssache – entscheidend ist nur, dass im Summenblock nur Beschleunigungen addiert werden und nicht eine Mischung aus Beschleunigungen und Kräften.<sup>7</sup>

<sup>7</sup>Oder alternativ die Kräfte addieren und zwischen Summe und linkem Integrator mit einem Gain-Block mit  $1/m$  multiplizieren.

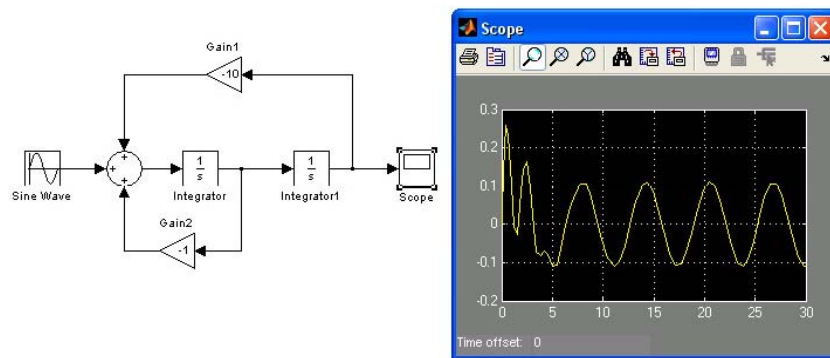


Abbildung 1.8: Erzwungene Schwingung in SIMULINK, zur Erläuterung siehe Text

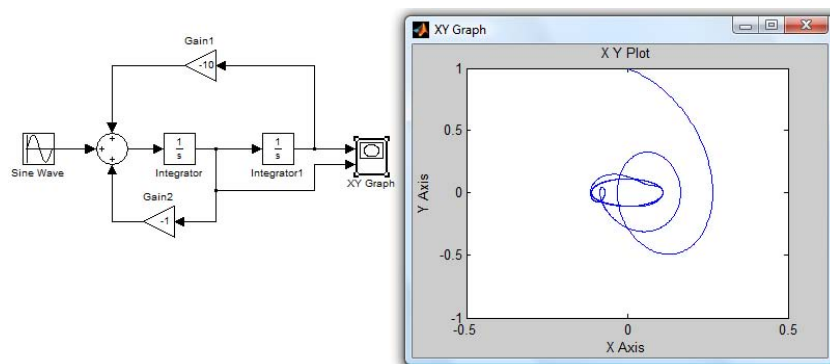


Abbildung 1.9: Erzwungene Schwingung in SIMULINK, Darstellung im Phasenraum

§ 42 Das Ausgangssignal des Summationsblocks ist die Beschleunigung. Diese geben wir wieder auf einen Integrationsblock (Anfangsbedingungen nicht vergessen!), um die Geschwindigkeit zu bestimmen. Nochmalige Integration liefert den Ort. Dieser wird in die Schleife zurück geführt und (gegebenenfalls gemeinsam mit der Geschwindigkeit) auf das Scope gegeben.

§ 43 Beim ersten Test des Modells lassen wir, entsprechend dem Vorgehen in Abb. 1.2, die externe Quelle weg (bzw. setzen deren Amplitude auf Null). Durch geeignete Wahl von  $\omega_0^2 = k/m$  und  $\gamma = \beta/2m$  sollte das Modell dann die drei Lösungen der homogenen DGL (1.3) reproduzieren. Anschließend können wir eine von Null verschiedene externe Kraftdichte angeben und erhalten eine erzwungene Schwingung wie in Abb. 1.8 gezeigt.

§ 44 Abbildung 1.9 zeigt die gleiche Simulation, jedoch werden die Ergebnisse nicht mit Hilfe des Scope als Orts-Zeit-Diagramm dargestellt sondern mit Hilfe des XY-Graph (siehe § 182) im Phasenraum gezeigt. Die Kurve beginnt entsprechend den Anfangsbedingungen in der Mitte oben; das Einschwingen erfolgt entlang einer irregulären Bahn bis diese im stationären Zustand in eine Ellipse um den Ursprung übergeht.

**Verständnisfrage 5** Die Darstellung im Phasenraum ist eine Möglichkeit, periodische Vorgänge zu erkennen: diese sind stets mit geschlossenen Bahnen im Phasenraum verbunden. Geben Sie andere formale Möglichkeiten zur Beschreibung einer periodischen Bewegung an. Wie stellt sich eine Bewegung konstanter Frequenz aber variabler Amplitude im Phasenraum dar? Kann man die konstante Frequenz in dieser Darstellung erkennen?



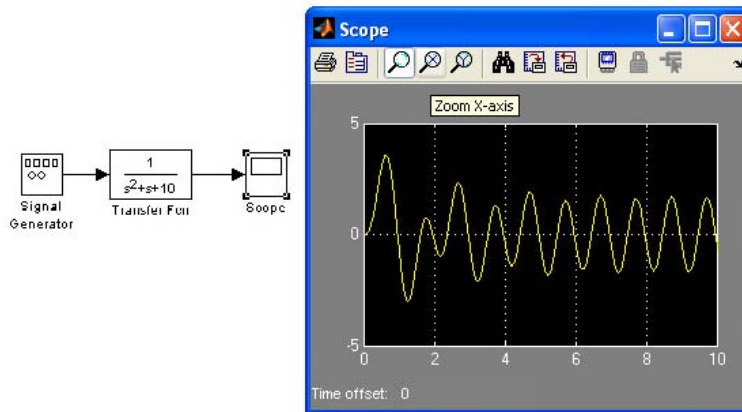


Abbildung 1.10: Erzwungene Schwingung in SIMULINK unter Verwendung des Blocks Transfer Function

### 1.1.4 Alternative Darstellungen: Übertragungsfunktion und State Space

§ 45 In SIMULINK lässt sich das System Federpendel auf unterschiedliche Weisen beschreiben. Die Lösung der Differentialgleichung haben wir bereits betrachtet, andere Varianten sind die Beschreibung des Systems durch seine Systemgleichungen (State Space) oder die abstraktere Darstellung mit Hilfe einer Übertragungsfunktion.

#### Übertragungsfunktion

§ 46 Die Übertragungsfunktion eines Systems beschreibt den Zusammenhang zwischen den Eingangs- und den Ausgangsgrößen des Systems (siehe Abschn. 4.3). Übertragungsfunktionen sind eine kompakte und elegante Beschreibung für lineare zeitinvariante Systeme.<sup>8</sup> Die Übertragungsfunktion wird in der Regel nicht im Zeitbereich bestimmt sondern mit Hilfe einer Laplace Transformation (siehe Abschn. 3.3). Diese wird in SIMULINK durch den Block Transfer Function (siehe § 161ff) abgebildet.

§ 47 Die Differentialgleichung

$$m\ddot{x} + \beta\dot{x} + kx = F_A$$

transformiert sich (unter Vernachlässigung der Anfangsbedingungen) zu

$$ms^2X(s) + \beta sX(s) + kX(s) = F(s) .$$

Die Übertragungsfunktion ist definiert als das Verhältnis von Ausgangs- zu Eingangssignal

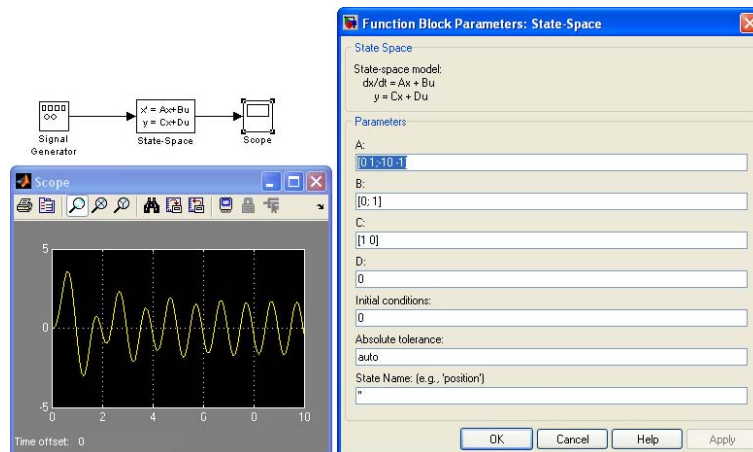
$$G(s) = \frac{X(s)}{F(s)} = \frac{1/m}{s^2 + \frac{\beta}{m}s + \frac{k}{m}} = \frac{1}{ms^2 + \beta s + k} .$$

Im Bildbereich gilt dieser Zusammenhang für beliebige Eingangssignale, d.h. wir sind nicht auf harmonische Kräfte beschränkt.

**Verständnisfrage 6** Darf man die Anfangsbedingungen einfach so vernachlässigen? Welche Konsequenzen hat das für die Lösung? Und als Nachfrage: Vernachlässigt man Anfangsbedingungen wirklich? Betrachten Sie die Lösung genau!

<sup>8</sup>Kapitel 4 wird sich ausführlich mit linearen zeitinvarianten oder LTI (linear time invariant) Systemen beschäftigen. Sie finden vielfältige Anwendung in der Physik, als Beispiele werden wir u.a. Filter betrachten. LTI Systeme sind für viele technische Anwendungen von Interesse – eben gerade, da sie den Zusammenhang zwischen Ausgangs- und Eingangsgrößen regeln und damit z.B. zur Filterung und Steuerung eingesetzt werden können. LTI Systeme sind als Systeme jedoch eher uninteressant, da sie sich nicht entwickeln und damit auch keine Emergenz zeigen können.

Abbildung 1.11: Erzwungene Schwingung unter Verwendung des Blocks State Space



§ 48 Abbildung 1.10 zeigt die Realisierung dieser Simulation in SIMULINK mit Hilfe des Blocks **Transfer Function**. Das ist die einfachste Form der Simulation dieses Problems, da wir nur drei Elemente benötigen: eine Quelle zur Darstellung des externen Antriebs, den Block für die Übertragungsfunktion und eine Senke zur Darstellung der Ergebnisse. Mit den bereits in Abb. 2.16 verwendeten Größen  $m = 1$ ,  $k/m = 10$  sowie  $\beta/m = 1$  ergibt sich für die Übertragungsfunktion  $G(s) = 1/(s^2 + 1s + 10)$  und damit die in Abb. 1.10 gezeigte Simulation.

**Zwischenrechnung 1** Die Simulation in Abb. 1.10 ist bereits mit einem allgemeinen Signalgenerator ausgestattet. Spielen Sie mit verschiedenen Eigenfrequenzen des freien Systems und verschiedenen Formen/Frequenzen der Anregungsfunktion. Können Sie Resonanz erzeugen? Was passiert bei Anregung mit einer Sprungfunktion (Block **Step**) oder einer Rampe (Block **Ramp**)? Wie lange müssen Sie bei verschiedenen Eigenfrequenzen/Dämpfungen simulieren, um aus dem Bereich des Einschwingens heraus zu kommen.

**Verständnisfrage 7** Steht diese Variante zur Behandlung des Systems Federpendel auch für die freie bzw. für die gedämpfte Schwingung zur Verfügung?

### Lösung der Systemgleichungen mit State Space

§ 49 Der Block **State Space** (siehe § 158) geht von einer Beschreibung des Systems durch Zustandsvariablen aus. Der Block ist insbesondere für die Beschreibung linearer Systeme geeignet; LTI Systeme gehen jedoch stets von einer Beschreibung des Systems durch eine DGL erster Ordnung aus während die erzwungene Schwingung durch eine DGL zweiter Ordnung beschrieben wird. Für dieses Modell betrachten wir ein Single-Input, Single-Output (SISO) System: wir haben eine antreibende Kraft als Input und den Ort der einen im System vorhandenen Masse als Output.

§ 50 Folgen wir also dem bereits in § 18 verwendeten Verfahren und ersetzen die DGLs zweiter Ordnung durch zwei DGLs erster Ordnung. Dann lässt sich unser System durch zwei Zustandsgrößen beschreiben:  $x_1 = x$  und  $x_2 = \dot{x}$ . Die zeitlichen Änderungen der beiden Zustandsvariablen sind wieder wie in (1.6) und (1.7)

$$\dot{x}_1 = x_2 \quad \text{und} \quad \dot{x}_2 = -\frac{k}{m}x_1 - \frac{\beta}{m}x_2 + \frac{1}{m}F_A.$$

In Matrixschreibweise ergibt sich

$$\dot{\vec{x}} = \mathbf{A}\vec{x} + \mathbf{B}\vec{u}$$

mit dem Zustandsvektor

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix},$$

der quadratischen Systemmatrix

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ -k/m & -\beta/m \end{pmatrix},$$

der Eingangsmatrix (bei der die Zeilenzahl der Zahl der Zustandsvariablen entspricht, die Spaltenzahl der Zahl der Eingangsgrößen)

$$\mathbf{B} = \begin{pmatrix} 0 \\ 1/m \end{pmatrix}$$

und dem (in diesem Fall recht kurzen) Vektor der Eingangsgröße

$$\vec{u} = F_A.$$

§ 51 Die Ausgangsgleichung des Systems ist

$$\vec{y} = \mathbf{C}\vec{x} + \mathbf{D}\vec{u}$$

mit der den Anfangszustand enthaltenden Ausgangsmatrix

$$\mathbf{C} = (1 \quad 0)$$

und  $\mathbf{D} = 0$ .

§ 52 Modell, Parametereinstellungen sowie Ergebnis sind in Abb. 1.11 gezeigt. Mathematisch unterscheidet sich die Lösung natürlich nicht von der in Abb. 1.8: auch hier werden zwei Integratoren benötigt, da beide Zustandsvariablen integriert werden müssen.

**Verständnisfrage 8** Gilt hier die gleiche Einschränkung wie bei der Übertragungsfunktion (äußerer Antrieb muss vorhanden sein) oder lässt sich diese Variante auch auf die freie bzw. gedämpfte Schwingung anwenden?

§ 53 Da wir die erzwungene Schwingung bereits mit Hilfe der Übertragungsfunktion betrachtet haben, können wir das System als eine Art Filter interpretieren. Die inhaltlich äquivalente Beschreibung mit Hilfe des Zustandsraumes **State Space** legt nahe, dass sich eine derartige Beschreibung auch auf andere Filter anwenden lässt. Wir werden in Abschn. 5.2.4 darauf zurück kommen.

### Zusammenfassung: Gewöhnliche lineare Differentialgleichung in SIMULINK

§ 54 Das Federpendel wird durch eine gewöhnliche inhomogene Differentialgleichung zweiter Ordnung beschrieben:

$$\ddot{x} + \alpha\dot{x} + \beta x = \gamma(t)$$

mit  $\alpha$  und  $\beta$  als konstanten Koeffizienten und  $\gamma$  als einer von der unabhängigen Variablen abhängigen Funktion.

§ 55 Zur Lösung der ODE stehen uns in SIMULINK die auch von analytischen Verfahren bekannten zwei Varianten zur Verfügung:

- die Lösung im Zeitbereich durch wiederholte Integration, wobei in SIMULINK im Gegensatz zur analytischen Lösung nicht zwischen der inhomogenen und der homogenen DGL unterschieden werden muss. Alternativ zur Integration kann die Lösung auch mit Hilfe des Blocks **State Space** erfolgen – mathematisch passiert das gleiche wie bei der doppelten Integration, da ja zwei Zustandsvariablen jeweils integriert werden müssen; lediglich die Darstellung ist kompakter.
- die Lösung im Bildbereich durch Laplace Transformation und Bestimmung der Übertragungsfunktion. Im Gegensatz zur analytischen Lösung entfällt hier die Rücktransformation, da SIMULINK uns mit dem Ergebnis im Zeitbereich versorgt.

§ 56 Im Zeitbereich benötigen wir für die Lösung einer ODE  $n^{\text{ter}}$  Ordnung  $n$  Integratorblöcke, für jeden Koeffizienten der ODE einen Gain-Block, einen Summationsblock sowie im Falle der inhomogenen DGL einen Input oder Funktionsgenerator, der die Inhomogenität beschreibt und ebenfalls auf den Summenblock geht. Außerdem natürlich den Output.

**Verständnisfrage 9** Wie wird ein verschwindender Koeffizient berücksichtigt, d.h. z.B. bei einer ODE  $3^{\text{ter}}$  Ordnung tritt die zweite Ableitung nicht auf?

§ 57 Das Herunterbrechen der Lösung einer ODE  $n^{\text{ter}}$  Ordnung auf die entsprechende Anzahl von Integrationen gibt einen Einblick in die von SIMULINK (und damit letztendlich auch von MATLAB) verwendete Lösungsstrategie für ODEs: die ODE wird auf ein System von  $n$  ODEs erster Ordnung zurück geführt, die dann gleichzeitig gelöst werden. Das Verfahren ist ökonomisch insofern, als dass nur Solver für ODEs erster Ordnung benötigt werden. Für diese Solver sind Stabilität und Genauigkeit zwar wohl definiert, allerdings summieren sich die Ungenauigkeiten der einzelnen Solver im Laufe des Verfahrens. Die Möglichkeiten einer echten handgestrickten Lösung, z.B. durch eine geschickte Diskretisierung auch höherer Ableitungen (für einige einfache erste Hinweise siehe z.B. [65] und Abschn. 10.2.1), werden auf diese Weise verschent. Dadurch ist SIMULINK für die Lösung komplexerer Probleme zumindest in der wissenschaftlichen Arbeit nur bedingt geeignet.<sup>9</sup>

§ 58 Im Bildbereich liefert die Laplace Transformation einer ODE  $n^{\text{ter}}$  Ordnung ein Polynom  $n^{\text{ter}}$  Ordnung in der komplexen Frequenz  $s$ :

$$\begin{aligned} x^{(n)'} + a_{n-1}x^{(n-1)'} + \dots + a_2\ddot{x} + a_1\dot{x} + a_0x &= b(t) \\ (s^n + a_{n-1}s^{n-1} + \dots + a_2s^2 + a_1s + a_0)X(s) &= \mathcal{L}(b). \end{aligned}$$

Dieses bestimmt die Übertragungsfunktion:

$$G(s) = \frac{X(s)}{\mathcal{L}(b)} = \frac{1}{s^n + a_{n-1}s^{n-1} + \dots + a_2s^2 + a_1s + a_0};$$

neben dem Output wird dann nur noch ein Signalgenerator oder anderer Input für die Inhomogenität benötigt.

§ 59 Unabhängig von der Lösungsstrategie hat SIMULINK gegenüber der analytischen Lösung den Vorteil, dass sich nicht nur die etwas schwerer formulierbaren Anregungen ( $\delta$ -Pulse, Rechteck) sondern auch stochastische Anregungen genauso einfach berücksichtigen lassen wie die harmonische Anregung. Gerade im Hinblick auf die Selbstorganisation ist der Zufall eine zu wichtige Größe als dass man ihn einfach vernachlässigen könnte.

### 1.1.5 Nichtlineare Erweiterung sorgt für Chaos: Fadenpendel

§ 60 Das Fadenpendel lernen wir meist in der linearisierten Variante, d.h. für kleine Auslenkungen, kennen:

$$\ddot{\varphi} + \frac{g}{l}\varphi = 0 \quad \Rightarrow \omega_0 = \sqrt{\frac{g}{l}}.$$

Für beliebige Auslenkungen ist die Bewegungsgleichung nicht-linear

$$\ddot{\varphi} + \frac{g}{l}\sin\varphi = 0.$$

<sup>9</sup>Das ist kein SIMULINK Problem sondern letztendlich nur ein MATLAB-Erbe, da SIMULINK auf die MATLAB-Funktionen zurück greift. Auch wenn komplexere Modelle (Atmosphäre, Ozean, Klima, interplanetarer Raum und darin befindliche Teilchen und Shocks) für die wissenschaftliche Arbeit immer noch FORTRAN oder verwandtes benötigen, schließt das nicht die Anwendung von SIMULINK auf diese Probleme in der Ausbildung aus: da SIMULINK sehr gut strukturiert ist, lassen sich die Prinzipien (und damit auch die Annahmen und Probleme der Modelle) darin viel besser erfassen als im realen 'wissenschaftlichen' Code, wo zuviel Aufmerksamkeit in das HickHack um eine effiziente/ökonomische/saubere Codierung geht. Und in Systemen, die auf viele ODEs niedriger Ordnung zurück greifen, ist die Reduktion auf ODEs erster Ordnung ohnehin keine Einschränkung – daher lassen sich viele technische Probleme gut in SIMULINK beschreiben. Entsprechend weit verbreitet ist der Einsatz in der Industrie.

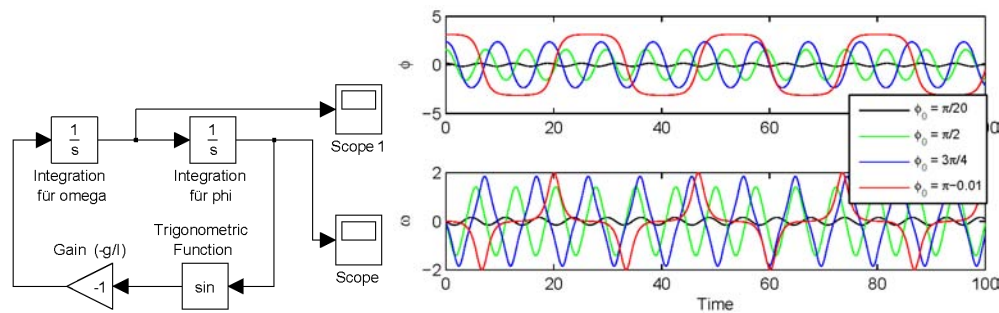


Abbildung 1.12: Fadenpendel nichtlinear (große Winkel). *Links*: Modell; *rechts*: Auslenkung und Winkelgeschwindigkeit für verschiedene Anfangsauslenkungen (jeweils mit  $\dot{\varphi}_0 = 0$ ). Die Frequenz des linearen Pendels ist  $\sqrt{g/l} = 2\pi$

Aus Gründen der Arbeitssicherheit ersetzen wir den Faden durch eine masselose Stange: dann gibt es keine Probleme, falls die Auslenkung mal größer als  $\pi/2$  wird.

**Verständnisfrage 10** Wie ließe sich die nichtlineare Differentialgleichung analytisch lösen?

§ 61 In SIMULINK lässt sich das Modell analog zum Federpendel konstruieren; der einzige Unterschied ist der zusätzliche trigonometrische Block um den Sinus von  $\varphi$  zu bestimmen, vgl. linkes Teilbild in Abb. 1.12. Im rechten Teilbild sind Auslenkung  $\varphi$  (oben) und Winkelgeschwindigkeit  $\omega$  für verschiedenen Anfangsauslenkungen  $\varphi_0$  bei verschwindender Anfangsgeschwindigkeit,  $\dot{\varphi}_0 = 0$ , gezeigt.

§ 62 Für kleine Auslenkungen (schwarze Kurve,  $\varphi_0 = \pi/20$ ) ergibt sich annähernd die harmonische Schwingung wie im linearen Fall. Mit zunehmender Auslenkung wird die Schwingungsperiode länger und die Amplitude weicht von einer harmonischen Funktion ab. Dies wird besonders deutlich in der roten Kurve, bei der das Pendel fast aus dem ‘Kopfstand’ gestartet wird. Alle Kurven sind jedoch periodisch, d.h. der Zustand des Pendels ist bei Kenntnis der Anfangsbedingung problemlos vorherzusagen.

**Verständnisfrage 11** Was ist eine kleine Auslenkung? Bis wann ist die lineare Näherung zulässig und wie quantifiziert man den durch die Näherung gemachten Fehler?

§ 63 Chaos (vgl. Abschn. ??) ist dadurch charakterisiert, dass eine kleine Abweichung in den Anfangsbedingungen zu großen Abweichungen in der Ausgangsgröße führen kann. Das ist bei der nichtlinearen Variante des Fadenpendels offensichtlich nicht der Fall – außer, wir starten das Fadenpendel wirklich aus der ‘Überkopf’-Position oder wählen eine andere Kombination von Anfangsbedingungen, die zu einer Annäherung an die Überkopf-Position führen. Diese entspricht einem instabilen Gleichgewicht, d.h. jede kleine Störung setzt das Pendel zur einen oder anderen Seite in Bewegung. In der Umgebung eines bestimmten Anfangswerts (bzw. einer bestimmten Kombination von Anfangswerten) wird das System chaotisch. Für andere Anfangswerte dagegen ist die Bewegung nicht chaotisch. Das gilt auch solche Anfangsgeschwindigkeiten, die eindeutig einen Überschlag und damit eine nicht-gleichförmige Kreisbewegung bewirken – letztere sind in den Ergebnissen daran zu erkennen, dass der Winkel  $\varphi \geq \pi$  wird.

§ 64 Um zu klassifizieren, für welchen Satz von Anfangsbedingungen ein chaotisches oder ein nicht-chaotisches Verhalten zu erwarten ist, bildet man die Trajektorie des Systems im Phasenraum. Für die in Abb. 1.12 verwendeten Anfangsbedingungen ergeben sich geschlossene Trajektorien (siehe linken Teil von Abb. 1.13), d.h. die Bahnen sind periodisch. Zwar ist nur die Trajektorie für  $\varphi_0 = \pi/20$  elliptisch (bzw. kreisförmig) und damit die Konsequenz einer harmonischen Bewegung, geschlossen sind die anderen Bahnen aber dennoch. Außerdem

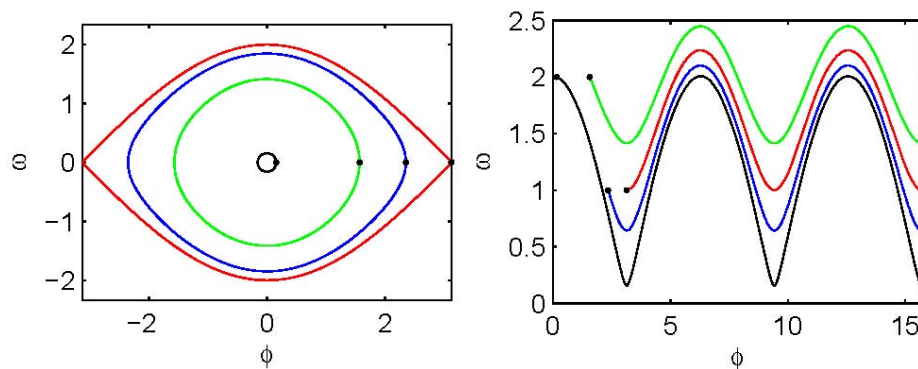


Abbildung 1.13: Fadenpendel im Phasenraum: *links* wurden die Ergebnisse aus Abb. 1.12 verwendet, *rechts* wurde mit den gleichen Anfangsauslenkungen und zusätzlich Anfangsgeschwindigkeiten  $\dot{\varphi} = 1$  für die beiden großen Auslenkungen und  $\dot{\varphi} = 2$  für die kleinen Auslenkungen verwendet (Anfangswerte jeweils als Punkte)

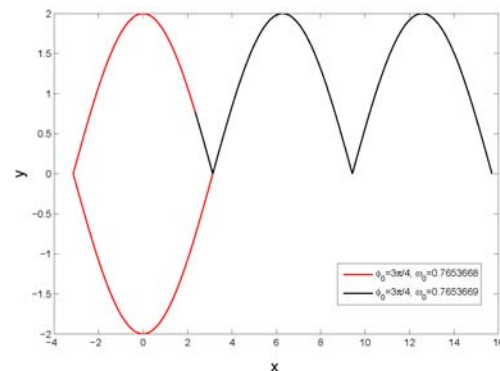


Abbildung 1.14: Eine kleine Abweichung in den Anfangsbedingungen ( $10^{-7}$  in  $\dot{\varphi}$ ) führen zu unterschiedlichen Bewegungen: Schwingung (rot) oder Rotation (schwarz)

zeigt sich eine systematische Verformung mit zunehmender Anfangsauslenkung: solange die Bahn periodisch ist, bleibt  $|\varphi|$  stets kleiner als  $\pi$ . Auch wenn die Werte für  $\varphi$  können nicht beliebig ansteigen: sonst geht die Schwingung in eine Kreisbewegung über. Die Anfangsbedingungen der einzelnen Bahnen sind jeweils durch den Schnitt mit der positiven  $\varphi$ -Achse gegeben, angedeutet durch die schwarzen Punkte.

§ 65 Geben wir dem System bei gleichen Anfangsauslenkungen jeweils ausreichend große Anfangsgeschwindigkeiten mit, so bewegt sich das Pendel auf einer Kreisbahn. Im Phasenraum ist diese Bahn nicht geschlossen, da der Winkel immer weiter zunimmt, vgl. rechtes Teilbild in Abb. 1.13. Für diese Anfangsbedingungen entweicht das Pendel aus dem Bereich des Phasenraums, der eine Schwingung beschreiben würde. Aber egal, welchen Anfangswert wir nehmen: die Trajektorien liegen stets relativ dicht beieinander und die Vorhersagbarkeit des Ausgangszustands bei kleinen Abweichungen von den Anfangsbedingungen ist gegeben.

§ 66 Die in Abb. 1.14 gezeigten Trajektorien im Phasenraum unterscheiden sich in der Anfangsgeschwindigkeit  $\dot{\varphi}$  lediglich um  $10^{-7}$ ; die sich ergebenden Bewegungen sind jedoch qualitativ völlig unterschiedlich: für den kleineren der beiden Werte ergibt sich eine Schwingung (und damit eine geschlossene Bahn im Phasenraum); für den größeren der Werte dagegen ergibt sich eine Rotation und damit eine Bahn im Phasenraum, die sich immer weiter vom Ursprung entfernt.

§ 67 Kombinieren wir die beiden Teilabbildungen aus 1.13, so ergibt sich ein dünner Ring zwischen den beiden Modi, in dem das Ergebnis kritisch von den Anfangsbedingungen abhängt: dort ist eine Vorhersagbarkeit realistischweise nicht mehr gegeben.

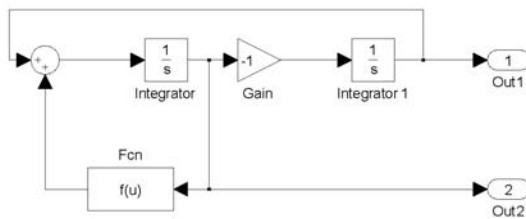


Abbildung 1.15: Modell `state_portrait.mdl` zur Lösung der nichtlinearen DGL zweiter Ordnung aus Abschn.1.1.6

§ 68 Kompliziertere Variationen des Fadenpendels werden wir im Zusammenhang mit einer detaillierteren Diskussion des Chaos in Abschn. 6.3.1 kennen lernen.

**Verständnisfrage 12** Reale Systeme sind nicht Reibungsfrei wie in diesem Modell sondern dissipativ. Wie ändert sich die Vorhersagbarkeit in diesem Fall? Ergeben sich wieder zwei Bereiche, in denen Vorhersagbarkeit gegeben ist oder verschwindet die Vorhersagbarkeit für einen oder beide der im reibungsfreien Fall beobachteten vorhersagbaren Bereiche?

### 1.1.6 Nichtlineares Beispiel in konventioneller Darstellung

§ 69 Steuern wir die Simulation aus dem normalen MATLAB-Workspace heraus, so können wir auf alle MATLAB-Möglichkeiten zurück greifen und eine Differentialgleichung auch wie aus der Mathematik gewohnt im Zustandsraum darstellen. Die Kommunikation mit dem Workspace wird im Detail in Abschn. 2.5 beschrieben.

§ 70 Betrachten wir als einfaches Beispiel eine gewöhnliche nicht-lineare Differentialgleichung zweiter Ordnung,

$$\ddot{x} = -x - \dot{x}^3 - 3\dot{x},$$

oder geschrieben als ein System von zwei gewöhnlichen DGLs erster Ordnung:

$$\begin{aligned} \dot{x}_1 &= -x_2 \\ \dot{x}_2 &= x_1 + x_2^3 - 3x_2. \end{aligned}$$

Zur Lösung lässt sich ein Modell ähnlich dem für das Fadenpendel in Abb. 1.12 konstruieren – der einzige inhaltliche Unterschied ist die zu verwendende mathematische Funktion. Diese wird mit Hilfe des Blocks `Fcn` aus der Blockgruppe `User-defined Functions` vereinbart als `pow(u[1],3)-3*u[1]`. Das in Abb. 1.15 gegebene Modell unterscheidet zusätzlich durch die beiden Output Ports `out`, die eine einfache Übergabe der entsprechenden Zustände an den Workspace zur weiteren Verarbeitung erlauben.

§ 71 Da die Ergebnisse der Simulation ohnehin im Workspace weiter verarbeitet werden sollen, können wir der Simulation auch die Parameter wie die Integrationskonstanten aus dem Workspace übergeben und sie sogar aus diesem starten. Dazu dient der folgenden m-File:

```
01 h=0.01; opts=simset('Solver','ode5','FixedStep',h);
02 x1=-2.5:0.25:2.5;x2=x1;
03 [nr,nc]=size(x1);x1m=zeros(nc,nc);x2m=x1m;
04 for nx1=1:nc
05     for nx2=1:nc
06         opts=simset(opts,'InitialState',[x1(nx1),x2(nx2)]);
07         [t,x,y]=sim('state_portrait',h,opts);
08         dx1=x(2,1)-x1(nx1);dx2=x(2,2)-x2(nx2);
09         l=sqrt(dx1^2+dx2^2)*7.5;
10         if l>1.e-10
11             x1m(nx2,nx1)=dx1/l;
12             x2m(nx2,nx1)=dx2/l;
```

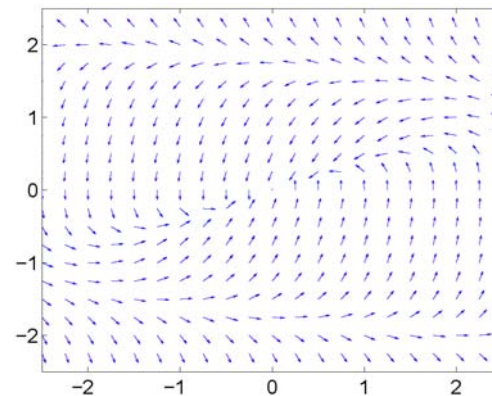


Abbildung 1.16: Quiver-Plot zum Modell `state_portrait.mdl` in Abb. 1.15

```

13     end
14     end
15     end
16     quiver(x1,x2,x1m,x2m,0);set(gca,'FontSize',14);
17     axis([-2.5,2.5,-2.5,2.5]);
18     set(gca,'Ytick',[-2,-1,0,1,2]);set(gca,'Xtick',[-2,-1,0,1,2]);
19     label('x1'),label('x2')

```

Der Befehl `simset` erlaubt die Übergabe von Parametern aus dem Workspace an die Simulation. In diesem Fall wird der Solver des numerischen Verfahrens bestimmt (in diesem Beispiel `ode5`, ein Dorman–Prince Verfahren mit fester Schrittweite  $h$ ) sowie über die Variablen dessen Schrittweite. Beides wird in die Variable `opts` geschrieben. In Zeile 6 werden diese Optionen noch einmal verfeinert, in dem die in Zeile 2 definierten Anfangswerte für die beiden Integratoren zusätzlich in die Variable `opts` geschrieben werden. Die Übergabe an die Simulation erfolgt in Zeile 7 beim Aufrufen der selben mit dem Befehl `sim`. Der dort übergebene Name `'state_portrait'` ist der Name des `.mdl`-Files, in dem das SIMULINK-Modell enthalten ist.

§ 72 Die sich ergebende Darstellung im Zustandsraum ist in Abb. 1.16 als Quiver-Plot gezeigt. Man erkennt den Ursprung als einen stabilen Punkt.

### 1.1.7 Geist ist geil: Im- und Exporte

§ 73 ‘Geiz ist geil’ hat der deutsche Einzelhandel zu vermitteln versucht, ‘Geist ist geil’ ist die Gegenkampagne eines Schweizer Buchhändlers zum Jahr der Geisteswissenschaften. Geist kann, insbesondere im Bereich der Modellierung und Simulation, ohne weiteres zu einem sehr ökonomischen Verhalten (viel Ergebnis bei wenig Aufwand) führen, Geiz dagegen kann das Arbeiten in diesem Bereich der Wissenschaft sehr anstrengend machen.

§ 74 SIMULINK hilft dabei in zweifacher Weise. Die ökonomische Codierung des Lösungsverfahrens haben wir bereits in § 36 angesprochen. Die von SIMULINK eingeforderte Strukturierung des zu untersuchenden Systems setzt eine abstrakte Betrachtung des Systems voraus: nicht die Details interessieren sondern Prinzipien. Abstraktion ist jedoch ein wesentliches Merkmal der Physik – die Simulation mit SIMULINK nähert sich damit stärker den Grundprinzipien der Physik als es eine analytische oder die konventionelle numerische Lösung tut. Auf Grund der Anfangsinvestition Abstraktion bietet SIMULINK auch den von der Physik bekannten Gewinn dieses Einsatzes, die Übertragungsfunktion.

§ 75 Betrachten wir unser Federpendel daher noch einmal etwas abstrakter: es handelt sich dabei um ein System, in dem drei Einflussgrößen wirken: eine ortsabhängige Kraft, eine geschwindigkeitsabhängige Kraft und eine äußere, nicht vom momentanen Zustand des Systems abhängige Kraft. Oder allgemeiner: eine (lineare) rücktreibende Kraft, eine dämpfende



Kraft und ein externer Antrieb. Alle Systeme, die von allen oder einigen dieser drei Einflüsse bestimmt sind, lassen sich damit im gleichen Modell beschreiben – lediglich die in den Gain-Blöcken einzutragenden Parameter haben unterschiedliche Bedeutung. Diese Übertragbarkeit gilt zwar auch für die DGL und ihre analytische und numerische Lösung, sie wird dort jedoch weniger deutlich, da wir in der Regel zu viele physikalische Größen explizit angeben und diese nicht auf wenige abstrakte, allerdings für das Systemverhalten relevante Parameter reduzieren: wir denken leichter in den mit physikalischen Objekten verbundenen Begriffen Masse  $m$  und Federkonstante  $k$  als in dem abstrakteren Systemparameter Schwingungsfrequenz  $\omega_0^2 = k/m$ .

§ 76 Den Vorteil in der Verwendung der abstrakteren Parameter können wir uns mit Hilfe eines anderen einfachen physikalischen Systems veranschaulichen: beim elektrischen Schwingkreis als Reihenschaltung aus einer Quelle  $f_A$ , einem Widerstand  $R$ , einem Kondensator mit Kapazität  $C$  und einer Spule mit Induktivität  $L$  betrachten wir ebenfalls eine angetriebene Schwingung, beschrieben durch die DGL für die Ladung  $q$  auf dem Kondensator

$$\ddot{q} + \frac{R}{L} \dot{q} + \frac{1}{LC} q = f_A .$$

Beschränkung auf den ersten und dritten Term auf der linken Seite liefert eine freie Schwingung mit dem Systemparameter  $\omega_0^2 = 1/\sqrt{LC}$ . Der mittlere Term auf der linken Seite gibt die Dämpfung durch den Ohm'schen Widerstand. Die Gleichung können wir analog zu (1.1) auch schreiben als

$$\ddot{q} + 2\gamma \dot{q} + \omega_0^2 q = f_A \quad \text{mit} \quad \gamma = \frac{R}{2L} \quad \text{und} \quad \omega_0^2 = \frac{1}{LC} .$$

Hier verwenden wir wieder als Systemparameter die Frequenz  $\omega_0$  und die Dämpfung  $\gamma$ . Damit können wir den gleichen Algorithmus wie zur Lösung von (1.1) verwenden, lediglich bei der Vereinbarung der Systemvariablen müssen die entsprechenden physikalischen Größen berücksichtigt werden.

§ 77 Die Anwendung dieses Blockdiagramms ist jedoch nicht nur auf die Beschreibung schwingfähiger Systeme beschränkt. Dazu interpretieren wir die Systemgleichung nicht über eine rücktreibende und eine dämpfende Kraft sondern über eine orts- und eine geschwindigkeitsabhängige Kraft. Der harmonische Oszillator wird durch eine Bewegungsgleichung der Form  $\ddot{x} + \omega_0^2 x = 0$  beschrieben. Das  $x$  ergibt sich in der Differentialgleichung aus der vom Weg abhängigen rücktreibenden Kraft  $-kx$ . Die beliebte Übungsaufgabe<sup>10</sup> mit einem reibungsfrei über eine Tischkante gleitenden Seil der Länge  $l$  führt ebenfalls auf eine vom Ort abhängige Kraft, eben die Länge  $x$  des überhängenden Seilstückchens. Dann ist die Kraft  $mgx/l$ . Da diese Kraft nicht rücktreibend ist, erhalten wir für die Bewegungsgleichung  $\ddot{x} - xg/l = 0$ . Ein Exponentialansatz liefert die (reellen) Eigenwerte  $\lambda_{1,2} = \pm\sqrt{g/l}$  und damit die allgemeine Lösung  $x(t) = c_1 e^{\lambda_1 t} + c_2 e^{\lambda_2 t}$  bzw. für die Anfangsbedingungen  $v(0) = 0$  und  $x(0) = x_0$ :

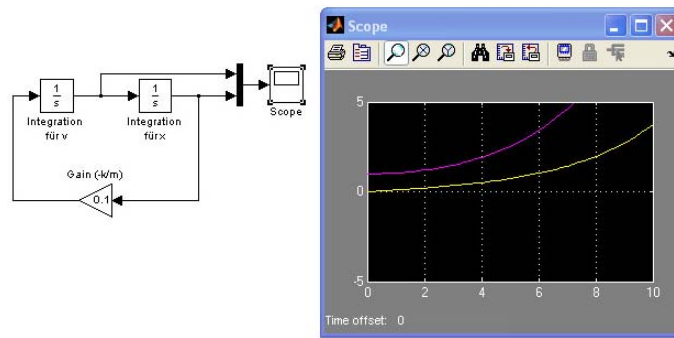
$$x(t) = \frac{x_0}{2} \left( \exp \left\{ \sqrt{\frac{g}{l}} t \right\} - \exp \left\{ -\sqrt{\frac{g}{l}} t \right\} \right) = x_0 \sinh \left( \sqrt{\frac{g}{l}} t \right) .$$

**Verständnisfrage 13** Müssen die Anfangsbedingungen wirklich so angepasst werden oder sind alle Kombinationen  $x_0$  und  $v_0$  in beiden physikalischen Problemen gleich sinnvoll (oder sinnlos)?

**Zwischenrechnung 2** Jetzt können Sie das Seil eigentlich auch unter Berücksichtigung der Reibung rechnen. Wie sieht die analytische Lösung dabei aus? Wie groß darf das überhängende Seilstück maximal sein, um ein Abrutschen zu verhindern? Oder lässt sich dieses in einem Modell analog zur gedämpften Schwingung doch nicht verhindern?

<sup>10</sup>Ein biegsames Seil der Länge  $l = 1$  m und der Masse  $m = 1$  kg gleitet reibungslos über die Kante eines Tisches; zur Zeit  $t = 0$  hängt ein Ende  $x_0$  über die Tischkante, das Seil ist aber noch in Ruhe. Gesucht ist der Ort und die Geschwindigkeit eines Punktes auf dem Seil als Funktion der Zeit.

Abbildung 1.17: Ein Seil gleitet reibungsfrei über eine Tischkante in SIMULINK, Ort in Magenta, Geschwindigkeit in gelb; zur Erläuterung siehe Text



§ 78 Für die Lösung in SIMULINK können wir das Blockdiagramm aus Abb. 1.6 wieder verwenden. Lediglich der Gain muss angepasst werden (im wesentlichen Vorzeichenwechsel); außerdem müssen gegebenenfalls die Anfangsbedingungen in den Integratorblöcken angepasst werden. Das sich ergebende Blockdiagramm ebenso wie das Ergebnis sind in Abb. 1.17 gezeigt.

**Verständnisfrage 14** Lässt sich die Simulation des Schwingkreises in § 76 in sinnvoller Form auf die elektromagnetische Welle in Materie ausdehnen? Dann hätten wir gleich einen Übergang zu einem vektoriiellen Problem und damit zu einem System von DGLs – auch wenn ein Separationsansatz ja eigentlich nicht auf gekoppelte DGLs führen sollte.

## 1.2 Eine erweiterte Form des einfachen Beispiels

§ 79 Ein einzelnes Federpendel ist ein sehr einfaches System: physikalisch, da es die minimal mögliche Zahl mit einander wechselwirkende Komponenten hat (zwei, die Feder und die Masse), und formal, da es durch eine einzelne gewöhnliche DGL beschrieben werden kann. Zwei Federpendel kombiniert versprechen dagegen eine interessantere Situation: die beiden Federpendel im linken Teil von Abb. 1.18 sind durch eine Feder gekoppelt, so dass sie mit einander wechselwirken.

§ 80 Der linke Teil von Abb. 1.18 kann mit den Begriffen der Systemtheorie auf verschiedene Weise beschrieben werden. Wir können ein einzelnes Federpendel als ein zwar minimalistisches aber vollständiges System beschreiben. Dann wäre das gekoppelte Pendel aus Abb. 1.18 eine Kombination zweier Systeme. Für unser prinzipielles Verständnis eines Systems hätte dieser Ansatz allerdings Konsequenzen. Für den reibungsfreien Fall lässt sich dies wie folgt illustrieren: das System Federpendel war u.a. durch die Erhaltung der (mechanischen) Energie definiert. Beim gekoppelten Pendel wird mechanische Energie von einem der Federpendel auf das andere transferiert. Außerdem steckt ein Teil der mechanischen Energie zumindest zeitweise in der koppelnden Feder. Gleichzeitig muss aber die (mechanische) Energie des Gesamtsystems aus den beiden Federpendeln plus der koppelnden Feder erhalten bleiben. Unter dem Gesichtspunkt, ein abgeschlossenes System zu schaffen, bei dem keine Flüsse von Energie oder Materie über die Umgrenzung berücksichtigt werden müssen, ist es sinnvoller, das gekoppelte Pendel komplett als ein System zu beschreiben.

**Verständnisfrage 15** Und wie sollte die Argumentation beim gedämpften Pendel erfolgen?

§ 81 Der Energieaustausch zwischen den beiden Federpendeln zeigt sich in den Systemgleichungen in Form eines Kopplungsterms. Formal lässt sich jedes Federpendel durch eine eigene Bewegungsgleichung beschreiben, die Kopplung mit dem anderen Federpendel erfolgt jeweils durch die durch die mittlere Feder vermittelte Kraft. Der entsprechende Kopplungsterm tritt in beiden DGLs in gleicher Form auf, allerdings mit entgegengesetztem Vorzeichen. Gehen wir davon aus, dass alle Federkonstanten identisch sind, lassen sich die Bewegungsgleichungen mit  $\omega_0^2 = k/m$  schreiben als

$$0 = \ddot{x}_1 + \omega_0^2 x_1 + \omega_0^2 (x_1 - x_2) = \ddot{x}_1 + \omega_0^2 (2x_1 - x_2)$$

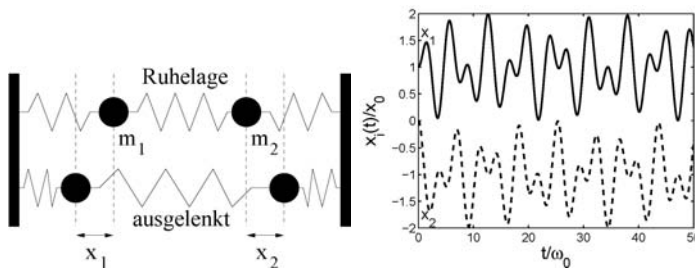


Abbildung 1.18: Gekoppelte Federpendel. *Links:* Definition der Geometrie; *Rechts:* Beispiel für die zeitliche Abhängigkeit der beiden Variablen  $x_1$  und  $x_2$

$$0 = \ddot{x}_2 + \omega_0^2 x_2 - \omega_0^2 (x_2 - x_1) = \ddot{x}_2 + \omega_0^2 (2x_2 - x_1) \quad (1.8)$$

oder in Matrixschreibweise

$$\ddot{\vec{x}} = \mathbf{A}\vec{x} = \begin{pmatrix} -2\omega_0^2 & \omega_0^2 \\ \omega_0^2 & -2\omega_0^2 \end{pmatrix} \vec{x}. \quad (1.9)$$

**Verständnisfrage 16** Wie ändern sich die Bewegungsgleichungen, wenn die Federn nicht alle identisch sind? Verändert sich dadurch die numerische Behandlung des Problems?

**Verständnisfrage 17** Stellen Sie die Bewegungsgleichung für gekoppelte Pendel mit Reibung auf.

### 1.2.1 Die analytische Lösung

§ 82 Das die gekoppelten Federpendel beschreibende System gekoppelter Differentialgleichungen hat konstante Koeffizienten und lässt sich daher ebenso wie die DGL für ein einfaches Federpendel mit einem Exponentialansatz  $\vec{x} = \vec{u}e^{\lambda t}$  lösen.

§ 83 Ableiten des Ansatzes und Einsetzen in die DGL (1.9) liefert

$$\lambda^2 \vec{u} = \mathbf{A}\vec{u}.$$

Diese Gleichung hat die Form einer Eigenwertgleichung für den Eigenwert  $\lambda^2$  und den Eigenvektor  $\vec{u}$ .

§ 84 Die charakteristische Gleichung für den Eigenwert ist

$$\begin{vmatrix} -2\omega_0^2 - \lambda^2 & \omega_0^2 \\ \omega_0^2 & -2\omega_0^2 - \lambda^2 \end{vmatrix} \stackrel{!}{=} 0 \quad \Rightarrow \quad (2\omega_0^2 - \lambda^2)(2\omega_0^2 - \lambda^2) - \omega_0^4 \stackrel{!}{=} 0.$$

Diese Gleichung ist erfüllt für  $\lambda^2 = -\omega_0^2$  und  $\lambda^2 = -3\omega_0^2$  und damit

$$\lambda_{1,2} = \pm i\omega_0 \quad \text{und} \quad \lambda_{3,4} = \pm i\sqrt{3}\omega_0.$$

Alle Eigenwerte sind komplex, wir erhalten also eine Schwingung.

§ 85 Mit diesen Eigenwerten ergibt sich die allgemeine komplexe Lösung zu

$$\vec{z}(t) = \vec{A}_1 e^{i\omega_0 t} + \vec{A}_2 e^{-i\omega_0 t} + \vec{A}_3 e^{i\sqrt{3}\omega_0 t} + \vec{A}_4 e^{-i\sqrt{3}\omega_0 t}$$

mit den aus den Anfangsbedingungen zu bestimmenden komplexen Integrationskonstanten  $\vec{A}_i$ .

Die physikalisch sinnvolle Lösung ist wieder durch den Realteil gegeben:

$$x(t) = \Re\{z(t)\} = a_1 \cos(\omega_0 t) + a_2 \sin(\omega_0 t) + a_3 \cos(\sqrt{3}\omega_0 t) + a_4 \sin(\sqrt{3}\omega_0 t).$$

§ 86 Die Lösung  $\lambda_{1,2}$  entspricht der des harmonischen Oszillators. In diesem Fall schwingen beide Massen in Phase, verhalten sich also wie eine einzige Masse: die die Massen koppelnde Feder wird weder gestreckt noch gestaucht, so dass der Kopplungsterm verschwindet. Damit könnten wir die koppelnde Feder auch weglassen.<sup>11</sup>

§ 87 Die Lösung  $\lambda_2$  entspricht dem Fall, dass beide Massen genau entgegen gesetzt schwingen, d.h. die Kopplung ist maximal. Diese Situation ist im linken Teil von Abb. 1.18 angedeutet. Für andere Anfangsbedingungen (z.B. eine Masse in der Ruhelage, die andere maximal ausgelenkt, beide ohne Anfangsgeschwindigkeit) ergeben sich Überlagerungen beider Frequenzen. Ein Beispiel ist im rechten Teil von Abb. 1.18 gegeben.

## 1.2.2 Die numerische Lösung

§ 88 Für die numerische Lösung versuchen wir, soweit möglich Anleihe bei der bisher betrachteten Lösung für die freie Schwingung zu nehmen. Betrachten wir zuerst nur die Bewegung einer Masse. Auf diese wirkt zum einen die rücktreibende Kraft, zum anderen die von der Verbindungsfeder zur anderen Masse ausgeübte Kraft – letztere hängt von der relativen Position der beiden Massen ab. Also müssen wir, wie bei der erzwungenen Schwingung, in Zeile 12a im Code in § 977 bei der Bestimmung der Beschleunigung diese zweite Kraft berücksichtigen – mit dem Unterschied, dass sich diese nicht aus dem System der ersten Masse alleine bestimmen lässt, sondern auch die Position der zweiten Masse benötigt. Aus diesem Grunde müssen wir die Bewegungsgleichungen beider Massen gleichzeitig lösen – es handelt sich eben um ein System gekoppelter Differentialgleichungen.

§ 89 Zur gleichzeitigen Behandlung beider Massen bedienen wir uns auch im numerischen Verfahren der vektoriellen Darstellung. Dazu müssen wir Ort und Geschwindigkeit jeweils als zweikomponentige Vektoren vereinbaren, ebenso die Anfangsbedingungen. Damit werden die Zeilen 8 und 9 des Codes in § 974 zu

```
08 x=zeros(2,length(t));v=x; %Festlegung Ausgabevektoren
09 x(1,1)=1.;x(2,1)=0.;v(1,1)=0.;v(2,1)=0; %Anfangsbedingungen
```

§ 90 Da die durch die koppelnde Feder ausgeübte Kraft für beide Massen unterschiedliches Vorzeichen hat, muss auch die Beschleunigung als Vektor definiert werden und wir können die eigentliche Rechenschleife codieren als

```
12a acc=[-omega2*(2*x(1,i-1)-x(2,i-1));-omega2*(2*x(2,i-1)-x(1,i-1))];
12b v(:,i)=v(:,i-1)+acc.*deltt;
13 x(:,i)=x(:,i-1)+deltt*v(:,i-1);
```

Außerdem muss die Darstellung angepasst werden, d.h. in den Plotbefehlen in Zeile 17 muss die entsprechende Komponente der Vektoren von Weg und Zeit für die darzustellende Masse ausgewählt werden. Abbildung 1.19 zeigt die Lösung des numerischen Verfahrens für die oben gegebenen Anfangsbedingungen.

§ 91 Nachdem wir den Code generiert haben aber bevor wir uns einer möglichen Realisierung in SIMULINK zuwenden, sollten wir uns überlegen, welche Änderungen sich im Block- bzw. Flussdiagramm aus dem rechten Teil von Abb. 1.3 ergeben. Außer der anderen Definition der Beschleunigung (wie wir sie ja ähnlich schon beim Übergang von der freien zur erzwungenen Schwingung vorgenommen haben) ergibt sich keine Änderung – unter der Voraussetzung, dass wir alle Größen, wie im Codebeispiel diskutiert, vektoriell betrachten.

## 1.2.3 Und die Lösung in SIMULINK?

§ 92 Da SIMULINK als Zustandsgrößen des Systems auch Vektoren und entsprechend zu

<sup>11</sup>Wir werden später den Begriff der Synchronisation verwenden. Die beiden Massen beim gekoppelten Federpendel schwingen für diesen Fall zwar synchron, sie synchronisieren sich jedoch nicht. Letzteres erkennt man daran, dass die koppelnde Feder irrelevant ist. Die Gleichläufigkeit der Bewegungen ergibt sich einzig aus den identischen Eigenfrequenzen und der Anfangsbedingung. Ob lineare Oszillatoren synchronisieren oder nicht, werden wir in Abschn. ?? genauer untersuchen.

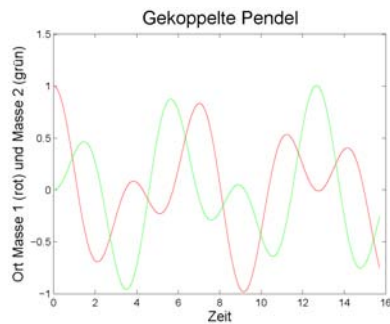


Abbildung 1.19: Numerische Lösung für die gekoppelten Pendel mit dem im Text gegebenen MATLAB-Code. Anfänglich sind beide Massen in Ruhe, Masse 1 (rot) ist ausgelenkt, Masse 2 (grün) befindet sich im Ursprung

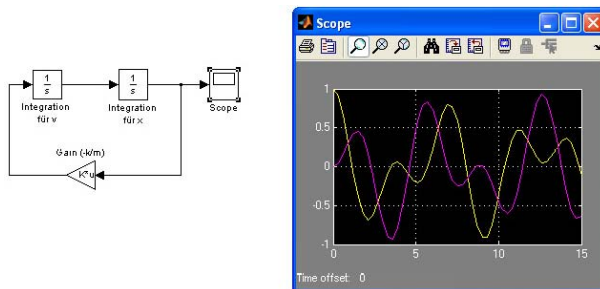


Abbildung 1.20: Gekoppelte Pendel in SIMULINK. Das Modell unterscheidet sich von dem der freien Schwingung in Abb. 1.6 durch die Verwendung von Vektoren als Signale und einer Matrix als Gain. Daher ist der Gain Faktor nicht mehr explizit im Block eingetragen und es gibt zwei Ausgangssignale (die jeweiligen Orte der beiden Massen)

deren Manipulation Matrizen akzeptiert, können wir das Modell der freien Schwingung direkt auf die beiden gekoppelten Pendel erweitern. Die Blöcke und ihre Verbindungen können unverändert übernommen werden, lediglich die von den einzelnen Blöcken verwendeten Parameter müssen angepasst werden. So muss der Integratorblock statt einer skalaren Größe eine vektorwertige Größe behandeln. Da dann auch der Anfangswert der jeweiligen Integration ein Vektor ist, können wir dem Integratorblock diese Information einfach durch die Übergabe eines Vektors statt eines Skalars als Anfangswert liefern.<sup>12</sup> Entsprechend muss der Gain als eine Matrix angegeben werden und nicht als ein Skalar.

§ 93 Das sich ergebende Modell ist in Abb. 1.20 gezeigt: die einzigen offensichtlichen Unterschiede sind die symbolische Angabe des Gains als Matrixmultiplikation und das Auftreten von zwei Ausgangssignalen, entsprechend den Orten der beiden Massen.<sup>13</sup> Da die Anfangsbedingungen denen der numerischen Lösung entsprechen, stimmen die Ergebnisse mit denen in Abb. 1.19 überein.

§ 94 Auch für die gekoppelten Pendel lässt sich die Darstellung im Phasenraum in SIMULINK direkt mit Hilfe des XY-Graph erzeugen, vgl. Abb. 1.21. Beide Massen bewegen sich im Phasenraum zwar nur in einem beschränkten Bereich, die Trajektorie ist jedoch (zumindest während des betrachteten Zeitintervalls) nicht geschlossen.

§ 95 Während der Übergang von skalaren zu vektoriellen Variablen in SIMULINK ebenso wie in MATLAB kein Problem ist, wird die Darstellung der Ergebnisse etwas komplizierter. Das Scope in Abb. 1.21 erhält eine vektorielle Eingangsgröße. Der Zeitverlauf der Komponenten dieses Vektors wird dargestellt; in diesem Beispiel also die jeweiligen Positionen der beiden Massen. Die Darstellung im Phasenraum dagegen ist auf diese Weise nicht möglich, da der

<sup>12</sup>Hier wird wieder eine MATLAB-Eigenschaft an SIMULINK vererbt: Variablen werden stets als Vektoren betrachtet, so dass eine Unterscheidung zwischen Skalar und Vektor nicht erforderlich ist. Jeder Algorithmus ist daher für Vektoren geschrieben – auch wenn wir häufig nur die erste Komponente des Vektors verwenden in der Meinung, das Problem nicht vektoriell sondern skalar zu behandeln.

<sup>13</sup>Erhält das Scope einen vektoriellen Input, so wird jede Komponente des Vektors als ein Track dargestellt. Wir haben diese Eigenschaft bereits ohne Kommentar in Abb. 1.7 gesehen: der Mux macht aus den Größen  $x$  und  $\dot{x}$  einen Vektor und das folgende Scope stellt wieder die einzelnen Komponenten des Vektors dar.

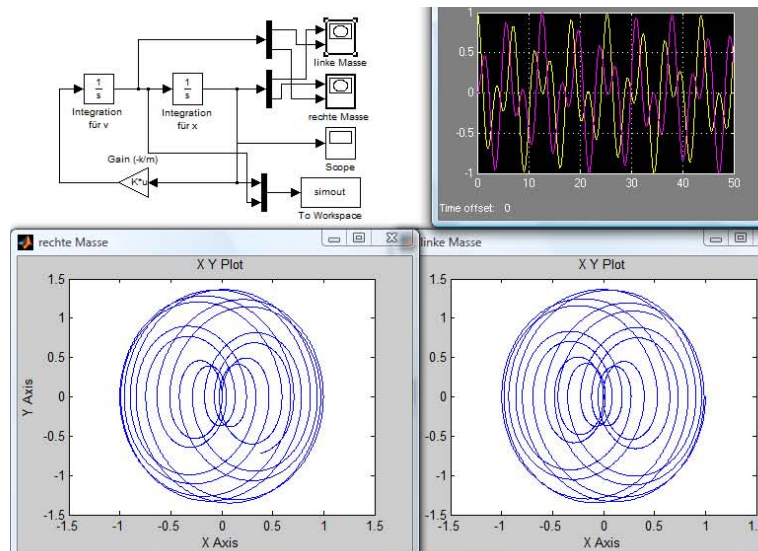


Abbildung 1.21: Gekoppelte Pendel, Darstellung im Phasenraum

Block XY-Graph zwar zwei Eingänge hat, an diesen aber nur skalare Eingangsgrößen akzeptiert. Daher müssen die vektoriellen Größen Ort und Geschwindigkeit jeweils mit Hilfe eines Demultiplexer-Blocks `Demux` in ihre Komponenten aufgespalten und auf getrennte XY-Graphen gegeben werden. Schon bei zweikomponentigen Variablen wird dies, wie in Abb. 1.21 erkennbar, leicht unübersichtlich – mit zunehmender Zahl von Komponenten wird dies nicht besser.

§ 96 Eine alternative Möglichkeit besteht darin, die Daten nicht direkt in SIMULINK darzustellen sondern wie an den MATLAB Workspace oder an eine Datei zu übergeben. In Abb. 1.21 wird das Ergebnis mit Hilfe des Blocks `To Workspace` in der rechten unteren Ecke des Modells als ein strukturiertes Element mit dem Namen `simout` an den Workspace übergeben. Zusätzlich wird ein Vektor `tout` übergeben, der die Zeiten enthält. Das strukturierte Element `simout` lässt sich durch Doppelklick genauer inspizieren; die gesuchten Werte stehen in einer Matrix `simout.signals.values`, in der die Spalten nacheinander die Komponenten des ersten Vektors (in diesem Fall der Ort) und die des zweiten (in diesem Fall die Geschwindigkeit) geben. Die Darstellung der Trajektorie des ersten Pendels im Phasenraum lässt sich erreichen durch

```
>> plot(simout.signals.values(:,1),simout.signals.values(:,3)) ←
```

§ 97 Die Ausgabe der Dateien an den Workspace (oder eine Datei) hat neben der größeren Übersichtlichkeit der Simulation den Vorteil, dass die Daten in beliebiger Form weiter verarbeitet werden können. Damit lassen sich nicht nur beliebige Darstellungen erzeugen; auch können die Simulationsergebnisse genau getestet werden. So ließe sich z.B. mit den Daten im Workspace die zeitliche Entwicklung der Gesamtenergie

$$E_{\text{ges}} = \frac{1}{2} (k_1 x_1^2 + k_3 x_2^2 + k_2 (x_2 - x_1)^2 + m_1 v_1^2 + m_2 v_2^2)$$

betrachten: ist bereits deren Konstanz nicht gewährleistet, so brauchen wir die weiteren Simulationsergebnisse nicht zu betrachten sondern sollten dringend die Simulation und das verwendete numerische Verfahren einer genaueren Betrachtung unterziehen.<sup>14</sup>

<sup>14</sup>Bei diesem konkreten Beispiel ist der einfachste Test immer noch der Vergleich mit der analytischen Lösung. Da diese als Referenz dienen kann, können wir durch Vergleich sogar eine Genauigkeit des numerischen Verfahrens bestimmen. Gibt es jedoch keine Referenz, so sollte eine Simulation stets überprüft werden auf Plausibilität und Erhaltungsgrößen (Masse, Energie, Impuls oder was sich gerade anbietet). Einige Testmöglichkeiten sind in [65] beschrieben.

**Zwischenrechnung 3** Bauen Sie die Simulation nach und testen Sie für verschiedenen Lösungsverfahren (ODE45, ODE23 ...), verschiedene Schrittweiten und verschiedene Genauigkeiten. Überlegen Sie sich Kriterien, an Hand derer Sie ein Ergebnis noch akzeptieren oder bereits verwerfen würden.

**Zwischenrechnung 4** Spielen Sie mit Variationen des Modells: nicht beide Enden fest sondern eins fest und eins lose oder den Extremfall, dass beide Enden lose sind. Letzteres entspricht zwei Massen, die durch eine Feder gekoppelt sind.

## 1.3 (Dynamische) Systeme

§ 98 Die Beispiele sollten verdeutlicht haben, dass sich SIMULINK zur Lösung von (Systemen von) Differentialgleichungen einsetzen lässt. Auch haben wir im letzten Paragraphen bereits einen Hinweis auf eine alternative Nutzungsform von SIMULINK mit stärkerer Orientierung an systemtheoretischen Begriffen erwähnt. In diesem Abschnitt wollen wir daher den Begriff des Systems etwas genauer untersuchen.

§ 99 Federpendel, Fadenpendel und selbst gekoppelte Federpendel sind sehr einfache physikalische Objekte für die SIMULINK eine schnelle, sogar mit einem elementaren physikalischen Verständnis verbundene Lösung erlaubt. Es sind jedoch nicht die Beispiele, für die sich der Aufwand der Einarbeitung in die formalen Grundlagen der Darstellung in SIMULINK lohnt – außer man geht von der anschaulichen Beschreibung zu den aus LTI-Systemen abgeleiteten Beschreibungen durch Übertragungsfunktion und State Space über. Seine Stärken dagegen zeigt SIMULINK in der Simulation von dynamischen Systemen.<sup>15</sup>

### 1.3.1 Was ist ein System?

§ 100 Den Begriff System verwenden wir im täglichen Sprachgebrauch in vielen Zusammenhängen: vom Ökosystem über das Wirtschaftssystem und das Sozialsystem [86] zum ewig kränkelnden Gesundheitssystem. Selbst Management wird gelegentlich mit systemtheoretischen Begriffen beschrieben [89, 90]. Die Existenz eines Studiengangs Umweltsystemwissenschaften legt die Vermutung nahe, dass sich auch die Umwelt als ein (oder vielleicht auch mehrere) System(e) beschreiben lässt. Wikipedia (<http://de.wikipedia.org/wiki/System>, 19.06.2007) hilft mit folgender Definition:

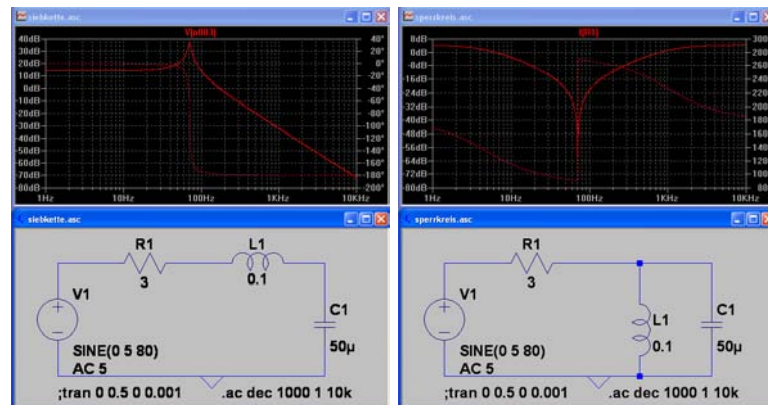
System (v. griech. *συστημα*, *systema*, ‘das Gebilde, Zusammengestellte, Verbundene’; Pl. Systeme) bezeichnet ein Gebilde, dessen wesentliche Elemente (Teile) so aufeinander bezogen sind und in einer Weise wechselwirken, dass sie (aus einer übergeordneten Sicht heraus) als aufgaben-, sinn- oder zweckgebundene Einheit (d.h. als Ganzes) angesehen werden (können) und sich in dieser Hinsicht gegenüber der sie umgebenden Umwelt auch abgrenzen.

Systeme organisieren und erhalten sich durch Strukturen. Struktur bezeichnet das Muster (Form) der Systemelemente und ihrer Beziehungsgeflechte, durch die ein System funktioniert (entsteht und sich erhält). Dahingegen wird eine strukturlose Zusammenstellung mehrerer Elemente auch als Aggregat bezeichnet.

§ 101 Das ist eine ziemlich gehaltvolle und sehr allgemeine Definition. Mit ihrer Hilfe können wir einige wesentliche Merkmale von Systemen, teilweise auch mit Hilfe von Beispielen, herausarbeiten. Außerdem können wir einige wichtige Schlagworte zur Charakterisierung von Systemen einführen – diese finden sich auch auf der oben gegebenen Wikipedia-Seite.

<sup>15</sup>Eine andere, im Rahmen des Skripts überhaupt nicht behandelte Stärke von SIMULINK ist die Verwendung in der Steuerungs- und Regelungstechnik.

Abbildung 1.22:  
Elektrische Schwingkreise: Serienschwingkreis oder Siebkette (*links*) und Parallelschwingkreis oder Sperrkreis (*rechts*)



### 1.3.2 Struktur oder Aggregat?

§ 102 Setzen wir die Stichworte des letzten Absatzes der Definition mit den bisher betrachteten Beispielen in Beziehung. Der Schwingkreis ist ein System aus den (System-)Elementen Kondensator mit Kapazität  $C$ , Spule mit Induktivität  $L$  und ohm'schem Widerstand  $R$  (sowie gegebenenfalls noch einer Spannungsquelle zur Anregung). Legen wir die Bauteile nebeneinander, so bilden sie bestenfalls ein Aggregat, sicherlich jedoch kein System.

§ 103 Ein System wird aus diesen Systemelementen erst durch die Erstellung einer Struktur, d.h. durch den Aufbau eines Beziehungsgeflechts. Hier sehen wir unmittelbar, dass sich aus den genannten Systemelementen verschiedene Systeme, vermutlich mit unterschiedlichen Funktionen, zusammensetzen lassen; als Beispiele sind in Abb. 1.22 der Serienschwingkreis (Siebkette) und der Parallelschwingkreis (Sperrkreis) gegeben. Für die Funktionen und damit die Bezeichnungen Serien und Parallel sind die Lage von Spule und Kondensator zueinander relevant, da diese beiden Bauelemente jeweils eine Phasenverschiebung in den Schaltkreis einführen.

§ 104 Ein funktionale Unterschied zwischen Serien- und Parallelschwingkreis ist naheliegend: im Parallelkreis liegen an Spule und Kondensator die gleiche Spannung an, es können aber unterschiedliche Ströme fließen; im Serienkreis dagegen ist der Strom durch beide Bauelemente der gleiche, es fallen aber unterschiedliche Spannungen über ihnen ab. Beide Schwingkreise haben eine Resonanz bei  $\omega_0 = 1/\sqrt{LC}$ : der Serienschwingkreis (Siebkette) lässt diese Frequenz passieren, der Parallelschwingkreis dagegen dämpft sie. Aus dem ursprünglich lose auf dem Schreibtisch liegenden Aggregat dieser elektronischen Bauelemente sind durch unterschiedliche Verknüpfungen zwei Systeme mit fast komplementären Eigenschaften entstanden.

§ 105 Auch wenn in diesem Beispiel durch unterschiedliche Beziehungsgeflechte zwischen den verschiedenen Systemelementen unterschiedliche Systeme erzeugt wurden, haben diese ein gemeinsames Merkmal: sie sind statisch. Das System ist in Systemelementen und Strukturen/Beziehungsgeflechten vorgegeben und verändert sich nicht. Zwar erzeugen unterschiedliche Eingangssignale auch unterschiedliche Ausgangssignale, legt man jedoch ein bestimmtes Eingangssignal zu verschiedenen Zeiten an das System an, so erhält man jeweils wieder das gleiche Ausgangssignal. Systemantworten sind reproduzierbar, da das Systemverhalten zeit-invariant ist.

§ 106 Statische Systeme haben sicherlich z.B. in der Filtertechnik ihre Berechtigung, es sind jedoch langweilige Systeme. Die Systeme, die uns wie Umweltsystem oder Gesundheitssystem, im alltäglichen Sprachgebrauch begegnen, sind dagegen dynamische Systeme, d.h. Systeme, die sich ihrer Umgebung anpassen können bzw. sich im Wechselspiel mit ihrer Umgebung formen.



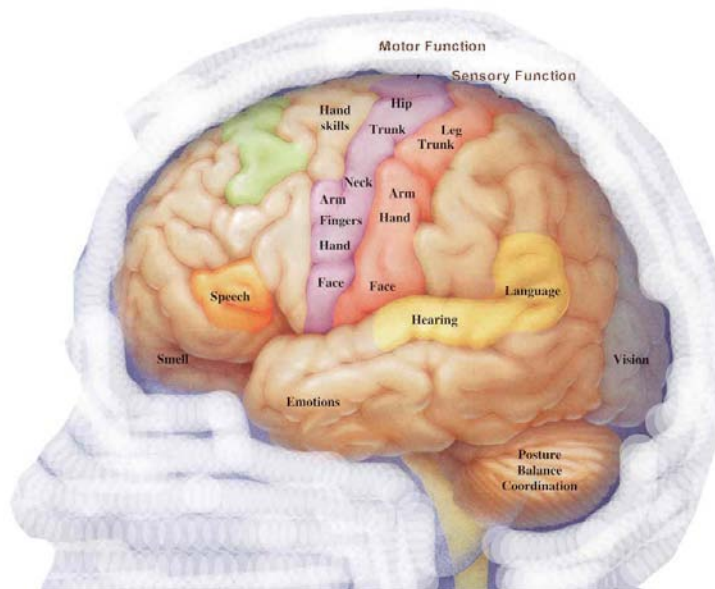


Abbildung 1.23: Spezialisierung verschiedener Regionen der Großhirnrinde (ok, inklusive Kleinhirn unten rechts, [http://www.epilepsy.uhhs.com/\\_borders/brain.1.jpg](http://www.epilepsy.uhhs.com/_borders/brain.1.jpg))

### 1.3.3 Selbstorganisation

§ 107 Eines der faszinierendsten dynamischen Systeme ist das zentrale Nervensystem. Aus der befruchteten Eizelle entwickelt sich durch fortgesetzte Zellteilung ein Häuflein identischer, allerdings noch omnipotenter Zellen. Ab einer bestimmten Größe des Zellhaufens fangen diese an, sich zu differenzieren - ein Prozess der durch An- bzw. Abschalten bestimmter Enzym-codierender Sequenzen der DNA erfolgt. Während dieser Differenzierung bilden sich Zellcluster aus (z.B. Leber-, Muskel- und Nierenzellen) die sich zu den entsprechenden Organen entwickeln. Entsprechend bilden sich auch Nervenzellen; die größte Ansammlung dieser Nervenzellen bildet das zentrale Nervensystem. Die Zellen dieses Systems differenzieren nur geringfügig; zwar gibt es verschiedene hochgradig spezialisierte Nervenzellen wie die Purkinje-Zellen<sup>16</sup> im Kleinhirn, deren Hauptaufgabe die Koordinaten verschiedenen Neuronen-Cluster ist; die Neuronen in der Großhirnrinde sind jedoch alle gleichen Typs obwohl sie räumlich ganz klar verschiedenen Aufgabenbereichen wie Sehrinde oder Sprachzentrum zugeordnet sind, vgl. Abb. 1.23. Diese Zuordnung ist durch die räumliche Anordnung sowie die Verknüpfung zwischen den Neuronen bestimmt. Wie schon bei den elektrischen Schwingkreisen haben wir hier also ein System, das aus wenigen verschiedenen einfachen Systemelementen besteht und bei dem die Funktionsweise durch das Beziehungsgeflecht zwischen diesen bestimmt ist.

§ 108 Bleibt nur das Problem: wie baut sich das Beziehungsgeflecht auf? Eine genetische Kodierung von Funktion, Lage und Verbindungen der Neuronen im Zentralnervensystem scheidet auf Grund der großen Zahl von sowohl Neuronen als auch Verbindungen aus: wären

<sup>16</sup>Den Purkinje-Zellen sieht man diese Aufgabe aus ihrer Struktur regelrecht an. Ein normales Neuron besteht aus einem Zellkörper sowie einem oder mehreren Dendriten (das ist die Eingabeseite) und einem Axon (Ausgabeseite). Signale an dem oder den Dendriten führen dazu, dass das Neuron feuert, d.h. ein elektrisches Signal entlang des Axons zu entweder den Dendriten weiterer Neuronen oder aber zu einer Muskel- oder sonstigen ausführenden Zelle führt. Ein bipolares Neuron hat nur einen Dendriten, d.h. es feuert oder feuert nicht, je nach Eingangssignal am Dendriten. Ein multipolares Neuron hat mehrere Dendriten, so dass die Frage 'feuern oder nicht feuern' durch die Kombination verschiedener Eingabesignale entschieden wird. Die Purkinje-Zellen sind eine Extremform des multipolaren Neurons - ihre Eingabeseite bildet einen regelrechten (nahezu 2D) Dendritenbaum mit Größenordnungsmäßig 150 000 Dendriten (siehe auch Abb. 1.24), d.h. das Ausgangssignal wird durch die Kombination sehr vieler Eingangssignale bestimmt. Daher eignen sich derartige Purkinje-Zellen sehr gut für die Koordinaten verschiedenen Neuronengruppen; sie sind maßgeblich an der motorischen Koordinaten beteiligt, ebenso aber auch an der Koordination der Verarbeitung verschiedener sensorischer Inputs.

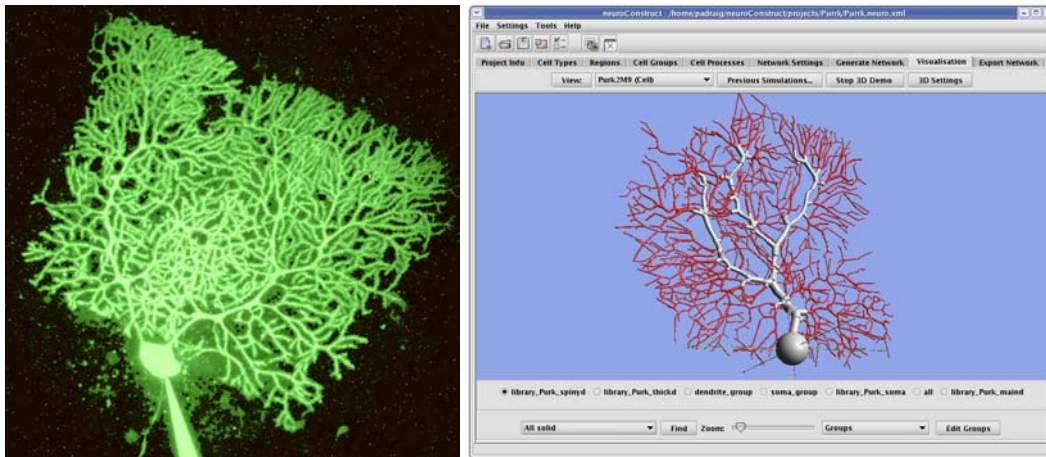


Abbildung 1.24: Purkinje-Zelle aus einer Fluoreszenzaufnahme (*links*; die feineren Verzweigungen verschwimmen in der Aufnahme auf Grund der begrenzten Auflösung; <http://synapse.princeton.edu/images/purkinje-green.jpg>) und schematisch in einem Modell zur Simulation neuronaler Netze auf biophysikalischer Basis (*rechts*; [http://www.physiol.ucl.ac.uk/research/silver\\_a/neuroConstruct/images/Purkinje.png](http://www.physiol.ucl.ac.uk/research/silver_a/neuroConstruct/images/Purkinje.png))

die  $10^{14}$  Verbindungen zwischen den  $10^{10}$  Neuronen der Großhirnrinde lediglich zufälligen Prozessen unterworfen, so würde man mindestens  $10^{15}$  Bit an Information zu ihrer Codierung benötigen – das menschliche Genom dagegen stellt lächerliche  $10^9$  Bit zur Verfügung. Wenn dem sich entwickelnden System die Information über seine Struktur nicht zur Verfügung gestellt werden kann, muss sich diese also aus dem System heraus entwickeln: das System ist selbst-organisierend.

§ 109 Selbstorganisation ist ein häufig verwendeter aber nicht sauber definierter Begriff (siehe z.B. den zugehörigen Wikipedia-Artikel <http://de.wikipedia.org/wiki/Selbstorganisation>). Als wichtigster Aspekt der Selbstorganisation gilt, dass die das System gestaltenden Einflüsse von den Elementen des sich organisierenden Systems selbst ausgehen. Die frühen Versuche der Chaos-Theorie und Erklärungen der Musterbildung, vgl. Abb. 1.25 und Abschn. 6.1, wie der Barnsley Farn [4] oder die Sierpinski Dreiecke [147, 148] ähnlichen Muster auf einigen Muschelschalen [98], nähern sich zwar der Beschreibung erstaunlich komplexer (und häufig natürlicher bzw. natur-ähnlicher) Gebilde an, beinhalten jedoch keine echte Selbstorganisation – sie sind algorithmisch bestimmt, nicht aber durch Reaktion auf ihre Umgebung. Letztendlich basieren alle diese Beschreibungen auf einem einfachen iterativen Algorithmus, sind damit also im wesentlichen Varianten der Koch'schen Schneeflocke [72] – auch wenn Mandelbrot's Apfelmännchen [91] oder der Feigenbaum Attraktor [26, 27] deutlich komplexer (und teilweise wesentlich ästhetischer) sind. Eine Zusammenfassung zu diesen verschiedenen Gebilden finden Sie unter dem Begriff Fraktal, gute Bücher sind [116, 117]. Fraktale eigen ist nur eine Selbstähnlichkeit, nicht jedoch eine Selbstorganisation; auch wenn umgekehrt einige selbstorganisierte Prozesse wie die Belousov-Zhabotinsky-Reaktion sich mit Hilfe von Selbstähnlichkeit beschreiben lassen.

§ 110 Die ersten, heute als Klassiker geltenden Ideen zur Selbstorganisation entwickelten sich im Grenzgebiet zwischen Chaos und Selbstorganisation in den 1950ern. Auf der experimentellen Seite ist die Belousov-Zhabotinsky-Reaktion BZR zu nennen. Irgendwann in den 1950ern stolperte Belousov über einen chemischen Oszillator: Er konnte bei der Oxidation von Zitronensäure mit schwefelsaurer Bromatlösung und Cer-Ionen als Katalysator einen periodisch auftretenden Wechsel der Farbe der Lösung zwischen gelb und farblos beobachten. Obwohl ähnliche chemische Oszillationen bereits seit Beginn des 19. Jahrhunderts gelegentlich beschrieben wurden, fand Belousov keine befriedigende Erklärung; im Gegenteil, er betrachtete

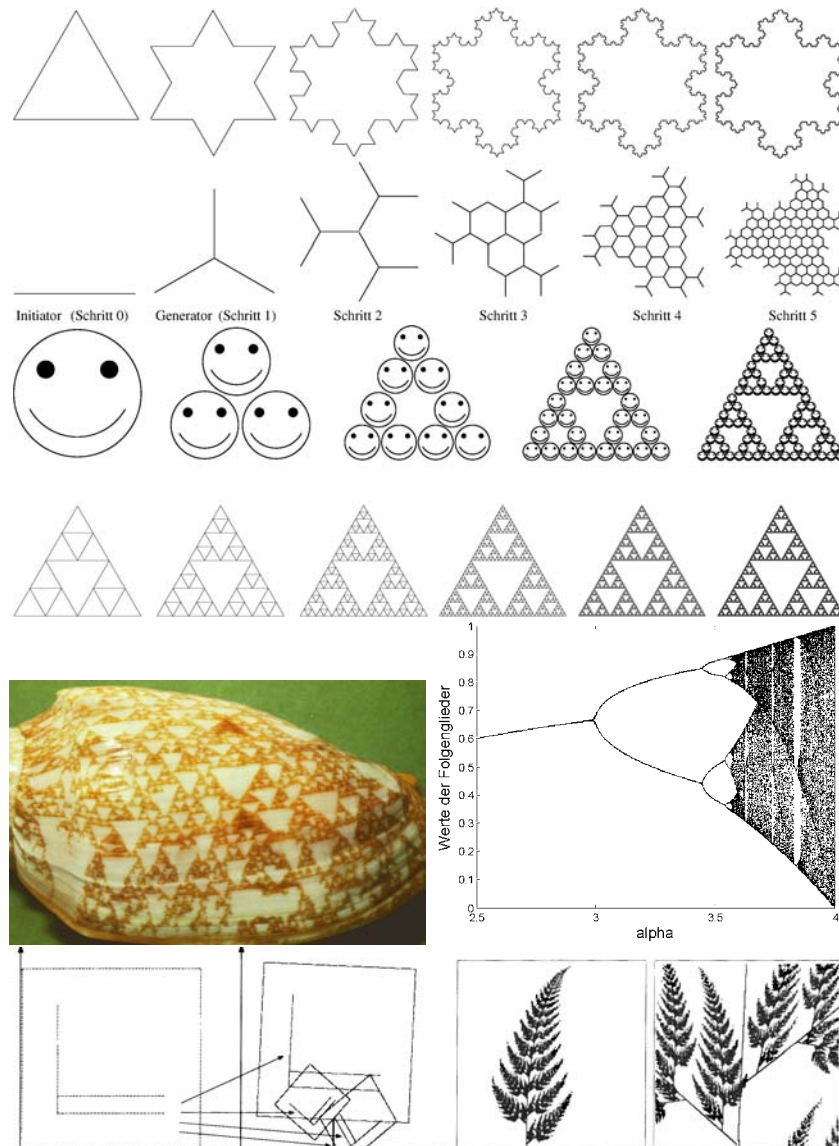
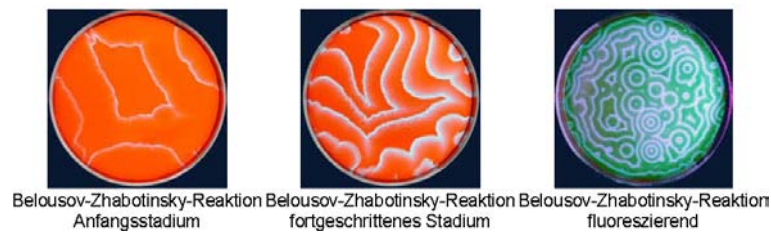


Abbildung 1.25: Verschiedene Fraktale. *Oben:* Koch'sche Schneeflocke, ursprünglich als Beispiel für eine stetige aber nicht differenzierbare Funktion eingeführt. Sie entsteht durch Dreiteilung jeder Seite eines gleichseitigen Dreiecks und Ersetzung des mittleren Drittels durch ein gleichseitiges Dreieck (abzüglich der Basisseite). *Zweite Zeile:* Koch verwendete statt des Dreiecks einen Dreispitz und konstruierte damit die Bienenwaabe als Urform des Fraktals. *Folgende Doppelzeile:* Das Sierpinski Dreieck entsteht durch Verkleinerung eines Elementardreiecks um den Faktor 2 und folgender Zusammensetzung von 3 Elementardreiecken zu einem neuen Dreieck. In der oberen Zeile ist das Elementardreieck aus Smileys zusammengesetzt - da diese zum Ende der Zeile ohnehin nicht mehr als solche zu erkennen sind, geht es in der folgenden Zeile nur noch mit Geraden weiter. Die letzten beiden Abbildungen der Zeile sind mit normaler Drucker- oder Bildschirmauflösung kaum noch zu unterscheiden, erste beim Reinzoomen erkennt man die Eigenschaft des Fraktals: egal, auf welcher Skala wir es betrachten, die Strukturen sind wieder die gleichen. *Vorletzte Zeile:* Sierpinski Dreiecke auf einer Muschelschale [98] und Feigenbaum Attraktor *Unterste Zeile:* Barnsley Farn [29]

Abbildung 1.26: Belousov-Zhabotinsky-Reaktion (<http://www.swisseduc.ch/chemie/orderchaos/reaktionen/musterbz.html>)



die chemische Oszillation als eher unwahrscheinlich. Erst 1959 veröffentlichte er einen kurzen Artikel darüber [5]; eine Erklärung für diese Reaktion lieferte Zhabotinsky 1964 [182]. Numerische Simulationen dieser Reaktionen basieren in der Regel auf zellulären Automaten, im Internet lassen sich weitere Beispiele finden, z.B. unter <http://jkrieger.de/bzr/inhalt.html> (ein Jugend-Forscht Beitrag mit vielen Literaturhinweise), <http://online.redwoods.cc.ca.us/instruct/darnold/deproj/Sp98/Gabe/> (gute Zusammenstellung der Differentialgleichungen), <http://www.cci.ethz.ch/mainmov.html?expnum=106&ismovie=0&picnum=-1&control=0&language=1&exchap=2+-1&exlist=-1> für ein Video, <http://www.hermetic.ch/pca/bz.htm> (mit Schwerpunkt auf dem zellulären Automaten); etwas anspruchsvoller auch Woltering [178] unter <http://eldorado.uni-dortmund.de:8080/FB2/ls7/forschung/2002/Woltering/maWolteringunt.pdf>.

§ 111 Die moderne Erklärung für die Belousov-Zhabotinsky-Reaktion holen wir uns von <http://www.swisseduc.ch/chemie/orderchaos/reaktionen/musterbz.html>:

Während normalerweise die Bromid-Ionen überall in der Lösung gleichmässig mit Bromat-Ionen und Malonsäure reagieren, kann ein Staubteilchen oder ein Kratzer an der Glaswand die Reaktion örtlich beschleunigen. Dadurch werden Bromid-Ionen an dieser Stelle verbraucht, während in der übrigen Lösung noch genügend davon vorhanden ist. Dies hat zweierlei Konsequenzen: Einmal schlägt die Farbe von Ferroin an der Stelle, welche kein Bromid enthält, von Rot nach Blau um. Zum anderen diffundieren Bromid-Ionen aus der Umgebung zu der Stelle an der das Bromid aufgebraucht ist. Dadurch verarmt die Umgebung an Bromid, so dass auch hier Ferroin durch Bromat oxidiert werden kann und sich blau färbt. Somit breitet sich eine blaue Welle in der Lösung aus. In ihrer Ausbreitungsrichtung ist sie schmal, da die blaue Farbe an ihrer Rückseite schnell verschwindet, wenn sich aus Brommalonsäure und blauem Ferroin wieder Bromid und rotes Ferroin zurückbilden.

Oder allgemeiner: eine kleine lokale Inhomogenität erzeugt einen Mangel einer der Substanzen. Dieser führt zu einem lokalen Gradienten, der seinerseits einen Diffusionsstrom treibt in Richtung dieser Fehlerstelle. Die damit verbundene Verringerung der Substanz in der Umgebung der Störstelle führt zu einer Verschiebung des Reaktionsgleichgewichts, angezeigt durch eine Umfärbung. Diese Störung breitet sich von der ursprünglichen Inhomogenität wellenförmig aus. Entscheidend ist das Auftreten der beiden Prozesse Diffusion (räumliche Umlagerung einer Substanz) und Reaktion (entsprechend der Umfärbung). Derartige Systeme werden als Reaktions-Diffusions-Systeme bezeichnet.

§ 112 Ebenfalls in den frühen 1950ern entwickelten sich die theoretischen Grundlagen zur Erklärung derartiger chemischer Oszillatoren sowie zu einem auf ähnlichen chemischen Reaktionen basierenden Konzept zur Beschreibung der Selbstorganisation. Es begann mit Turing [165], dem Vater des Code-Crackers für Enigma, und einem Tigerfell. Oder besser: es begann nicht mit dem Tigerfell sondern mit Turing's Interesse an Musterbildung in biologischen Strukturen wie Fibonacci Phyllotaxis, d.h. dem Auftreten von Fibonacci Zahlen in den Strukturen von Pflanzen, oder den Mustern von Tierhäuten. Turing's Ansatz basiert auf einem Diffusions-Reaktions-Modell mit einer aktivierenden und einer deaktivierenden Substanz. Daher stehen ein dämpfender und ein antreibende Prozess in Konkurrenz. Turing-Muster bilden nicht nur die formale Erklärung für die BZR und die Tierhäute;

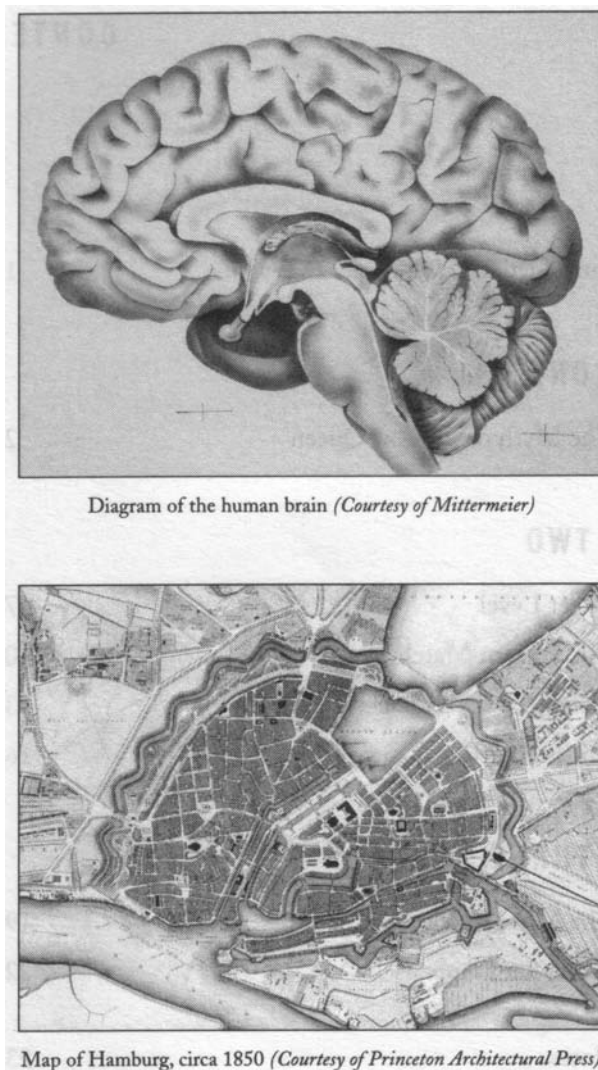


Abbildung 1.27: Selbstorganisierte Systeme: ein Gehirn und eine Stadt können sehr ähnliche Strukturen zeigen [62]

das System partieller Differentialgleichungen kann auch auf andere Probleme übertragen werden, z.B. Jäger-Beute-Modelle – eine bereits auf Turing selbst zurück gehende schöne anschauliche Version mit Missionaren und Kannibalen liefert Swinton (siehe z.B. <http://www.swintons.net/jonathan/Turing/talks/Manchester%202004.ppt>).

§ 113 Turing-Muster beschreiben die Selbstorganisation eines Systems mit dem Ergebnis, dass für eine Spezies (das entspräche einem Satz von Chemikalien mit zugehörigen Diffusionskonstanten und Reaktionsraten) stets ein prinzipiell gleiches Muster entsteht: ein Zebra hat Streifen, die Giraffe hat Flecken und selbst Spezies mit Fleckenmustern haben je nach Spezies zumindest für den Fachmann unterscheidbare Muster. Gleichzeitig sind die Muster innerhalb einer Spezies jedoch unterschiedlich genug, um an ihnen wie an einem Fingerabdruck einzelne Individuen zu unterscheiden. Diese Unterschiede sind durch die zufällige Anfangsverteilung der Reaktionspartner und kleine Inhomogenitäten bedingt.

§ 114 Auch für die Großhirnrinde benötigen wir eine solche Form der Organisation: entsprechend den Flecken im Leopardenfell müssen sich hier Cluster von Neuronen bilden, die z.B. den visuellen Cortex oder den auditorischen Cortex bilden. Die Organisation im Cortex geht jedoch einen Schritt über die Muster im Leopardenfell hinaus: es müssen nicht nur Cluster gebildet werden sondern diese müssen im sensomotorischen Cortex jeweils mit den

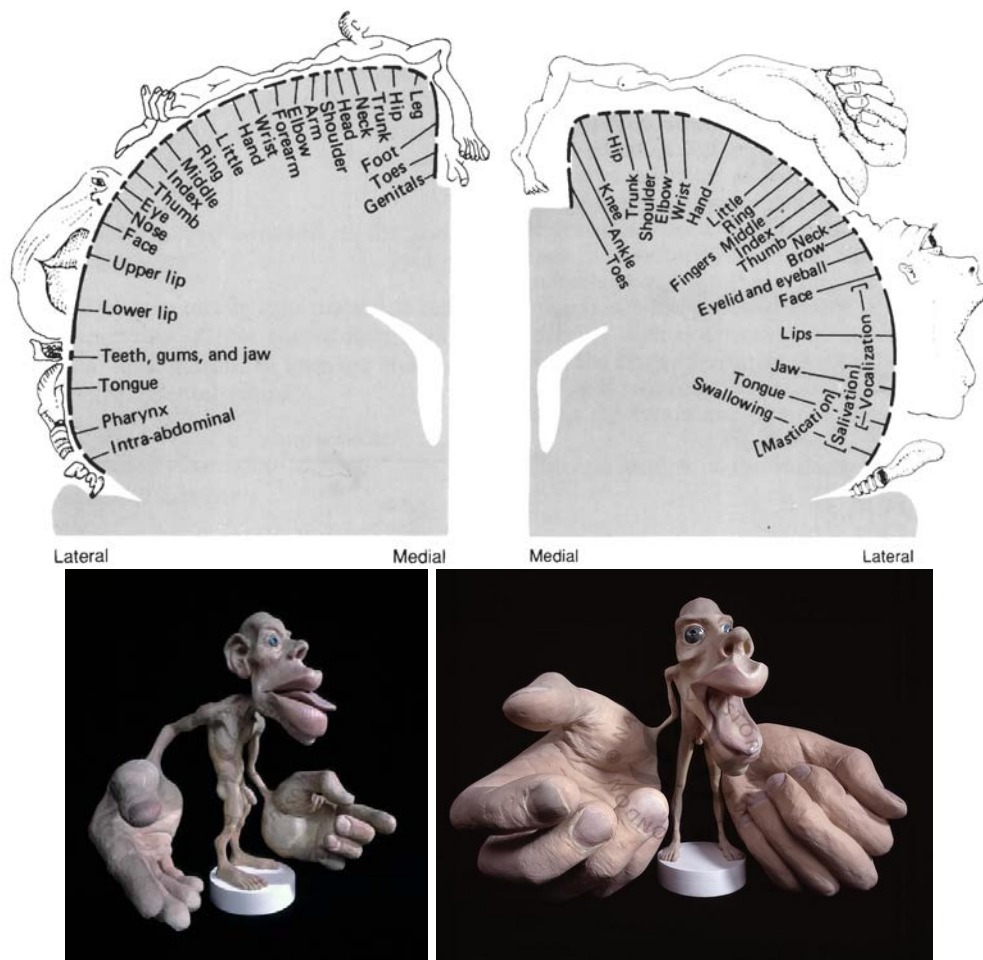


Abbildung 1.28: Humunculus: Abbildung der einzelnen Körperregionen auf den sensomotorischen Cortex (*oben*, [67]) sowie die anschauliche Darstellung der relativen Bedeutung der einzelnen Körperregionen entsprechend der zu ihrer Projektion im sensorischen (*links*) und motorischen (*rechts*) Teil des sensomotorischen Cortex benötigten Fläche (*unten*, ©Natural History Museum)

korrekten Sensoren bzw. Aktuatoren verbunden werden bzw. in anderen Bereichen müssen die Verbindungen zwischen den entsprechenden Teilen des Cortes erzeugt werden.

§ 115 Eine sehr anschauliche, allerdings mathematisch nicht unbedingt einfach zu fassende Beschreibung für die Ausbildung der Verbindungen ist das Wadi-Modell.<sup>17</sup> In einer Wüste gibt es keine Flüsse, allerdings kann es zeitweise sehr heftige Regenfälle geben. Da der Boden das Wasser nicht aufnehmen kann, muss es an der Oberfläche abfließen. Dabei folgt es bereits vorhandenen kleinen Rinnen und Rillen. Diese werden durch die Strömung tiefer ausgewaschen, so dass beim nächsten Regen dem Wasser schon deutliche Pfade vorgegeben sind. Mit jedem neuen Regen werden diese weiter vertieft, bis sie Wadis werden. Nimmt man Informationen anstelle von Wasser so lässt sich auf diese Weise die Verbindung zwischen Neuronen bzw. Gehirnarealen verstärken.<sup>18</sup> Einen kurzen, allerdings wesentlich fundierteren Überblick

<sup>17</sup>Das Modell ist an dieser Stelle nicht ganz korrekt – eigentlich beschreibt es eher Lernvorgänge, ist also ein Lern- und kein Entwicklungsmodell.

<sup>18</sup>Wie stark Nutzung und Neuronenbedarf mit einander verbunden sind, zeigt sich im sensomotorischen Cortex, d.h. jenem Streifen des Gehirns, in dem Informationen von den Sensoren ankommen und Informationen an den motorischen Apparat weiter gegeben werden, vgl. Abb. 1.23. Auch hier sind die Neuronen durch

über die Entwicklung des Gehirns gibt [173].

§ 116 Selbstorganisation bzw. Emergenz ist ein zur Zeit schnellwachsendes Forschungsfeld mit sehr unterschiedlichen Anwendungen. Populärwissenschaftliche aber dennoch fundierte Einführungen geben Johnson [62], aus dessen Buch auch das Eingangszitat und Abb. 1.27 stammen,<sup>19</sup> und Strogatz [156], der auf diesem Feld wissenschaftlich arbeitet. Eine mathematische (und auch theoretisch sehr gut fundierte) Einführung in Emergenz mit sehr vielen unterschiedlichen Beispielen gibt Scott [138].

## 1.4 Aufbau von Vorlesung und Skript

§ 117 Um einen inhaltlichen roten Faden zu ermöglichen, sind die für die Anwendung von SIMULINK erforderlichen Techniken in Kap. 2 zusammen gefasst. Wir werden aber im gesamten Skript, genauso wie im bisherigen Teil des ersten Kapitels, immer wieder im Rahmen von Beispielen neue Aspekte von SIMULINK einführen. Während diese in der Vorlesung je nach Bedarf mehr oder weniger genau diskutiert werden, sind sie im Skript in Kapitel 2 zusammen gefasst. Das Kapitel ist etwas sperrig und muss nicht unbedingt beim ersten Lesen vollständig durchgearbeitet werden sondern sollte jeweils bei Bedarf konsultiert werden. Ich habe dieser Variante des Aufbaus gegenüber einem stärkeren Einarbeiten in die einzelnen Kapitel bevorzugt, da es eine Trennung von Technischem und Inhaltlichem erlaubt und damit eine (nahezu) beliebige Auswahl der inhaltlichen Kapitel zulässt – ohne dass mühsam technische Grundlagen in anderen, nicht ausgewählten Kapitel nachgesucht werden müssen.

§ 118 In Kap. 2 werden wir nicht nur die grundlegende Elemente von SIMULINK kennen und anwenden lernen. Weitere Schwerpunkte des Kapitels sind die Definition der Eigenschaften der einzelnen Blöcke, die vor der Durchführung der Simulation zu bestimmenden Eigenschaften des Lösungsverfahrens sowie die Stabilität und Genauigkeit der Lösung. Als erstes Übungsbeispiel werden wir uns ausführlich mit (gekoppelten) Pendeln beschäftigen, am Rande werden aber auch Simulationen wie schwingende Kranlasten oder Zenerdioden auftreten.

---

ihre Verbindungen für bestimmte Aufgaben spezialisiert, d.h. es gibt Neuronen, die an die Fingerspitzen anknoppeln ebenso wie solche, die das Gesäß bedienen. Diese Verknüpfungen sind jedoch geometrisch nicht 1:1; das Verhältnis der Oberflächen von z.B. Lippen zu gesamtem Körper ist in der Repräsentation im sensomotorischen Cortex wesentlich größer als auf der realen Körperoberfläche. Das passt zu unserer Erfahrung: Lippen und Fingerspitzen sind wesentlich feinfühler als die vier Buchstaben, d.h. sie enthalten mehr Sensoren und liefern daher mehr Information. Um die Details der Information bei der Auswertung zu erhalten, muss diese mit gleichem Detailreichtum weiter gegeben werden, also wird eine entsprechend große Zahl von Neuronen im sensomotorischen Cortex benötigt. Anschaulich werden diese geometrischen Verzerrungen im Humunculus (vgl. Abb. 1.28) dargestellt: dieser bestimmt aus dem relativen Anteil der Fläche im sensomotorischen Cortex die Größe, die der entsprechende Körperteil real haben müsste.

<sup>19</sup>Zur Erläuterung: die Frage nach der Fähigkeit einer Stadt, zu Lernen, ist äquivalent zu der Frage, die den KI-Leuten und Roboterbauern soviel Probleme bereitet: das bewusste Lernen (Alphabet, kleines EinmalEins) lässt sich leicht vermitteln – dafür lassen sich Algorithmen konstruieren. Das unbewusste Lernen jedoch, also auch das, was der Körper macht, wenn er Bewegungsabläufe lernt, ist nicht in Algorithmen zu kleiden sondern erfolgt selbstorganisiert: durch Training lässt sich zwar organisieren, welche Bewegungsreize dem Körper angeboten werden, was er daraus macht, lässt sich aber kaum beeinflussen. Hier lernen Neuronen und Muskelzellen (bzw. jeweils kleine Gruppen von ihnen), ohne dass es von oben einen steuernden Masterplan gibt bzw. die beteiligten Zellen bewusst lernen – vgl. auch das bereits zitierte Wadi-Modell. Die Kommunikation ist jeweils einfach und auf wenige benachbarte Zellen beschränkt, ihre wesentlichen Kriterien sind: (1) more is different, (2) ignorance is useful, (3) encourage random encounters, (4) look for patterns in the signs, (5) pay attention to your neighbours. Auf ähnliche Weise können auch Städte aus einfachen lokalen Wechselwirkungen komplexe Wirtschaftssysteme ‘bottom up’ aufbauen: eine Zunft entwickelt sich einfach aus der lokalen begrenzten Interaktion der Beteiligten und kann u.U. eine Stadt dominieren ohne je darauf hingearbeitet zu haben. Pinker’s [122] Eingangszitat zu Kap. 2 weist darauf hin, dass selbst Sprache und der Spracherwerb selbstorganisiert sind – daher ist es auch schwer, einem Computer/Roboter freie Antworten beizubringen statt einer Auswahl aus vorgefertigten Sätzen/Satzfragmenten. Auch Universitäten zeigen im Zuge der Umstellungen der Studiengänge, der Exzellenzinitiative, der Evaluationen und der Leistungsbezogenen Mittelzuweisungen plötzlich emergentes (nicht einfach dynamisches) Verhalten – nur noch merkt es kaum jemand und hat auch noch keine Vorstellung von den Konsequenzen [106, 111].

§ **119** Die folgenden drei Kapitel werden sich mit linearen, zeit-invarianten Systemen (LTI-Systeme) beschäftigen – und damit die Grundbegriffe Übertragungsfunktion<sup>20</sup> und Systemvariable nutzen. Eine Übertragungsfunktion lässt sich als Output-Input-Relation auffassen: sie beschreibt z.B. das Verhältnis von Ausgangs- zu Eingangssignal eines Filters. Selbst bei einem einfachen Filter wie einem Hoch- oder Tiefpass sind der geschlossenen Darstellung der Übertragungsfunktion Grenzen gesetzt: in der Zeitdomäne lässt sich die Übertragungsfunktion nur für ein harmonisches Eingangssignal geschlossen darstellen, da sich das Ausgangssignal durch Multiplikation mit einem Dämpfungsfaktor und durch eine Phasenverschiebung aus dem Eingangssignal bestimmen lässt. Die Signalform selbst dagegen bleibt erhalten. Bei einem rechteckigen Eingangssignal dagegen verändert der Hoch- bzw. Tiefpass die Signalform, so dass eine Übertragungsfunktion im Zeitbereich nicht mehr (einfach?) zu bestimmen ist. Im Frequenzbereich ist die Bestimmung dieser Übertragungsfunktion jedoch weiterhin auf einfache Weise möglich. In Kap. 3 werden wir uns daher mit dem Übergang vom Zeit- in den Frequenzbereich befassen. Dazu können wir die Fourier oder die Laplace Transformation verwenden. Die Fourier Transformation ist Ihnen als Anwendung der Fourier Analyse zur Zerlegung zeitkontinuierlicher periodischer Signale in Grund- und Oberschwingungen bekannt. Die Laplace Transformation unterscheidet sich von der Fourier Transformation durch einen konvergenz-erzeugenden Faktor. Dadurch hat sie einen breiteren Anwendungsbereich, was insbesondere auch die Berücksichtigung der Sprungfunktion erlaubt und damit das Studium eines Systems nicht auf die Impulsantwort beschränkt sondern auch eine Untersuchung im Hinblick auf eine Sprungantwort zulässt.

§ **120** Zwar haben wir auf diese Weise die formalen Grundlagen zur Beschreibung einer Übertragungsfunktion gelegt – eine genaue Definition des Begriffes haben wir jedoch nicht vorgenommen. Diese erfolgt in Kap. 4 im Rahmen einer kurzen Einführung in die Systemtheorie. Dabei werden wir auch die theoretischen Grundbegriffe zum Verständnis des State space Blocks kennen lernen.

§ **121** In Kap. 5 werden wir die wesentlichen Aspekte der voran gegangenen Kapitel am Beispiel von Filtern wiederholen. Diese lassen sich durch einen analogen Aufbau einfach realisieren, so dass wir von physikalischer Realisierung über die Lösung der DGL mit Hilfe einer Laplace Transformation schnell zur Übertragungsfunktion gelangen. Mit Hilfe von Filtern zunehmender Ordnung lassen sich entsprechend Übertragungsfunktionen höherer Ordnung ‘schneiden’, so dass wir durch die Variation des entsprechenden Blocks in einer einfachen SIMULINK-Simulation schnell zu einem intuitiveren Verständnis dieser Übertragungsfunktionen kommen können. Wir werden ebenfalls kurz auf digitale Filter eingehen. Diese lassen sich von den Randbedingungen des analogen Aufbaus lösen und unterscheiden sich daher in einem Aspekt fundamental von diesen: analoge Filter erlauben keine Zeitumkehr, für digitale Filter ist dies kein Problem. Den einfachsten digitalen Filter, das gleitende Mittel, werden wir ausführlich diskutieren, ebenso die Unterscheidung in kausale und nicht-kausale Systeme. Im Hinblick auf SIMULINK werden wir den Umgang mit diskreten Systemen sowie die Darstellung der Filter nicht nur über die Übertragungsfunktion sondern auch mit Blockdiagrammen kennen lernen.

§ **122** Im dritten Teil werden wir uns mit dynamischen Systemen beschäftigen. Hier können die Konzepte linearer Systeme wie z.B. die Übertragungsfunktion nicht mehr verwendet werden. Als einfachen Einstieg in das Thema werden wir in Kapitel 6 mit typischen Beispielen für einfache chaotische Systemen beginnen. Dazu gehört ein kurzer Überblick über Fraktale und einige wesentliche Begriffe wie Feigenbaum-Diagramm, logistische Gleichung und Lorenz-Attraktor. Anschließend werden wir an Hand einfacher Variationen über ein einzelnes Pendel auf nichtlineare Probleme und die Darstellung von Chaos mit Hilfe von Poincare-Portraits, d.h. Schnitten im Phasenraum, eingehen. Die mechanischen Modelle lassen sich bis hin zum Duffing Oszillator erweitern. Mit Hilfe einer Kette von nichtlinearen Oszillatoren werden wir den Begriff des Solitons einführen und kurz auf das Fermi–Pasta–Ulam-Problem eingehen.

<sup>20</sup>Das ist der formal korrekte Ausdruck für jene Blöcke, die uns nicht eine einfache arithmetische Operation wie der Gain-Block erlauben sondern Operationen wie das Differenzieren und Integrieren ermöglichen.



§ 123 Das Soliton kann auch als ein erstes Beispiel für Emergenz oder Selbstorganisation verwendet werden: das Ganze ist mehr als die Summe seiner Teile und zeigt ein Verhalten, dass wir aus dem Verhalten der Bestandteile nicht erwartet hätten. In Kapitel 7 werden wir uns als Beispiel für typische nichtlineare, zur Selbstorganisation führende Modelle mit Diffusions-Reaktions-Modellen auseinander setzen. Diese finden Anwendung u.a. in Turing-Mustern, Jäger-Beute-Modellen oder dem Hodgkins-Huxly-Modell eines Neurons. Mit diesen Beispielen werden wir uns mehr oder weniger ausführlich auseinander setzen.

§ 124 Im vierten Teil werden, bis jetzt leider nur für ein Beispiel, diskrete Systeme betrachtet. In Kapitel 8 spielt der Zufall, auch wenn der Titel vielleicht eher anderes vermuten lässt, keine Rolle: wir werden uns nicht mit dem Glücksspiel beschäftigen sondern mit der Spieltheorie als einer Möglichkeit, Entscheidungsfindung zu beschreiben. Als klassische Beispiele werden das Gefangenendilemma und die Gemeindewiese dienen. Etwas formaler wird es in der Anwendung auf die Wirtschaftswissenschaften und letztlich im Versuch, die Spieltheorie zu verwenden, um zu einer globalen Reduktion der CO<sub>2</sub>-Emissionen zu kommen.

## Literaturhinweise

§ 125 Eine kompakte und gleichzeitig sehr weit gehende Einführung in SIMULINK ist *Mastering SimuLink* von Dabney und Harman [18]. Das Buch erklärt alle wichtigen Aspekte zu SIMULINK und hat viele schöne Beispiele – Grundbegriffe wie die Laplace Transformation oder was denn nun eine Systemvariable eigentlich sein soll werden jedoch nicht eingeführt sondern bestenfalls am Rande erwähnt. In diese Begriffe einführende Literatur wird in Kap. 3 und 4 gegeben.

§ 126 Einführungen in SIMULINK finden Sie auch in einigen MATLAB-Büchern wie z.B. [44] oder [53]. In letzterem ist die Einführung sehr kurz und ohne formalen Hintergrund, berücksichtigt aber die wesentlichen Aspekte. In [44] wird SIMULINK wesentlich ausführlicher behandelt – insbesondere unter technischen Aspekten. Der formale Anspruch ist eher gering, d.h. Sie lernen, mit SIMULINK zu arbeiten, haben aber keine Vorstellung, warum der Integrator-Block mit  $\frac{1}{s}$  bezeichnet ist. Auch [87] gibt eine kurze Einführung in SIMULINK mit Schwerpunkt auf der Regelungstechnik.

§ 127 Auch online gibt es verschiedene Hilfsmittel zu SIMULINK. Sehr Übersichtlich strukturiert sind die *Controls Tutorials and Modeling Tutorials for MatLab and SimuLink* der Carnegie Mellon University unter <http://www.me.cmu.edu/ctms/> – insbesondere, da diese auch den im Rahmen dieses Skriptes vernachlässigten Aspekt der Echtzeitsteuerung ausgiebig berücksichtigen. Ebenfalls sowohl MATLAB als auch SIMULINK werden im Tutorial von Tom Nguyen unter <http://edu.levitas.net/Tutorials/Matlab/index.html> berücksichtigt. Dieses Tutorial ist insbesondere für Leser, die keine oder nur eine geringe Vorerfahrung mit MATLAB haben von Interesse. Und ein etwas älteres (aber umfangreiches) Tutorial findet sich z.B. unter <http://www.physnet.uni-hamburg.de/physnet/matlab/help/toolbox/simulink/ug/ug.html>.

§ 128 Sollten Sie SCILAB anstelle von MATLAB verwenden, so finden Sie dort mit SCICOS ein Äquivalent zu SIMULINK. Alle systemtheoretischen Aspekte dieses Skriptes sind ohnehin unabhängig von der verwendeten Software, bei der Angabe von Beispielen werde ich mich auf SIMULINK beschränken, eine Einführung in SCICOS geben Campbell et al. [11]. Eine bessere MATLAB-Alternative als SCILAB ist OCTAVE – dort gibt es jedoch meines Wissens kein SIMULINK-Äquivalent.

§ 129 Egal, was Sie an Literatur wählen: das in diesem einführenden Kapitel hinreichend strapazierte Federpendel wird Ihnen mit sehr großer Wahrscheinlichkeit wieder über den Weg laufen.

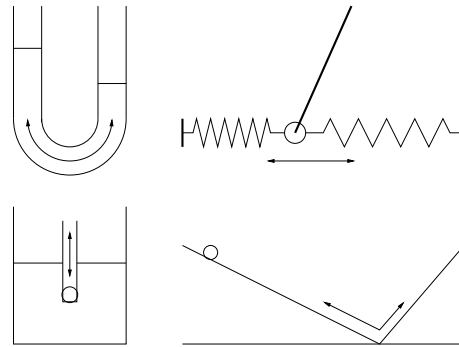


Abbildung 1.29: Zu Aufgabe 3: schwingende U-Röhre, Feder-Faden-Pendel, Schwingungstaucher, doppelt-schiefe Ebene

## Fragen

**Frage 1** Durch welche Eigenschaften ist ein System charakterisiert? Welche speziellen Eigenschaften hat ein dynamisches System?

**Frage 2** Diffusions–Konvektions-Modelle werden in vielen Bereichen verwendet. Beschreiben Sie die Grundlagen eines Diffusions–Konvektions-Modells und stellen Sie die Parameter dieses Modells zusammen.

## Aufgaben

**Aufgabe 1** Machen Sie sich (z.B. durch Recherche im Internet) mit den Begriffen Emergenz, Selbstorganisation und Synchronizität vertraut.

**Aufgabe 2** Falls sie Ihnen nicht gegenwärtig sind, setzen Sie sich noch einmal mit den Lösungsstrategien für die in diesem Kapitel betrachteten Differentialgleichungen auseinander.

**Aufgabe 3** Denken Sie etwas genauer über Schwingungen nach: Abbildung 1.29 zeigt verschiedene Pendel (schwingende U-Röhre, Feder-Faden-Pendel, Schwingungstaucher, doppelt-schiefe Ebene). Stellen Sie jeweils die Bewegungsgleichungen ohne Reibung auf. Sind alle Gleichungen linear; ergibt sich jeweils eine harmonische Schwingung? Wie ändern sich die Gleichungen bei Berücksichtigung von Reibung? Ändert sich der Schwingungstyp?

## SIMULINK-Spielereien

§ 130 Das Skript orientiert sich an komplexeren Anwendungen von SIMULINK. Daher ist im laufenden Text keine Einleitung in SIMULINK enthalten sondern stets nur Beispiele. Im Gegenzug kann in der Einführung in SIMULINK in Kapitel 2 eine vollständige Einführung gegeben werden, die auch dem Leser nützen kann, der sich nicht mit Systemen und Systemtheorie auseinander setzen möchte. Diese Aufgaben sollen auch den Anreiz bieten, sich parallel zum Text mit Kapitel 2 auseinander zu setzen.

**SIMULINK-Aufgabe 1** Reproduzieren Sie die verschiedenen Beispiele zu den Oszillatoren in diesem Kapitel.

**SIMULINK-Aufgabe 2** Wie würde sich das Modell des Federpendels ändern, wenn statt der linearen Rückstellkraft (Hooke'sches Gesetz) (a) eine quadratische Abhängigkeit und (b) eine Kombination aus quadratischer und linearer Abhängigkeit gegeben wäre? Spielen Sie jeweils mit den Modellen, z.B. im Rahmen einer Parameterstudie.

**SIMULINK-Aufgabe 3** Erweitern Sie das von der Tischplatte gleitende Seil um Reibung auf (a) der Tischplatte und (b) beim Fall durch die Luft. Gibt es, wie beim Fall mit Reibung, die Situation, dass das Seil sich mit konstanter Geschwindigkeit weiter bewegt? Welche Bedingungen müsste der Reibungsterm dafür erfüllen?

**SIMULINK-Aufgabe 4** Erstellen Sie das Modelle einer einfachen Raketengleichung (z.B. aus den Experimentalphysikübungen) in SIMULINK.

**SIMULINK-Aufgabe 5** Lösen Sie eine Zerfallsgleichung in SIMULINK.

**SIMULINK-Aufgabe 6** Erstellen Sie ein Modell für eine Zerfallskette mit einem instabilen Tochterisotop gefolgt von einem stabilen Tochterisotop.

# Kapitel 2

## SIMULINK – Die Grundlagen

Language is not a cultural artifact that we learn the way we learn to tell the time or how the federal government works. Instead, it is a distinct piece of the biological makeup of our brains. Language is a complex, specialized skill, which develops in the child spontaneously, without conscious effort or formal instruction, is deployed without awareness of its underlying logic, is qualitatively the same in every individual, and is distinct from more general abilities to process information or behave intelligently.

S. Pinker [122]

§ **131** In diesem Kapitel werden einige wichtige Grundbausteine einer Simulation mit SIMULINK vorgestellt – allerdings werden wir SIMULINK trotz seiner einfachen Struktur nicht wie eine natürliche Sprache unbewusst assimilieren können sondern wir müssen uns mit dem ganzen formalen Wust einer Sprache rumschlagen, d.h. dem Vokabular und der Grammatik. Dazu gehören neben der graphischen Benutzeroberfläche von SIMULINK Hinweise zu den wichtigsten Elementen und den daran vorzunehmenden Parametereinstellungen.<sup>1</sup> Dazu gehört insbesondere auch die Wahl eines für das Problem geeigneten Solvers. Auch die Steuerung der Simulation aus dem Workspace oder einem m-File sowie die Übergabe von Parametern und Vektoren zwischen diesen werden wir detailliert diskutieren. Letzteres ermöglicht insbesondere bei komplexeren Problemen (oder vielelementigen Zustandsvektoren) eine bessere Bearbeitung und Darstellung der Simulationsergebnisse. Beispiele für alle hier erwähnten Details haben wir bereits in der Einführung in Kap. 1 kennen gelernt.

§ **132** SIMULINK ist, wie durch die Verwendung von Blöcken bereits nahe gelegt, eine Art Lego® der Simulation; auf Grund der Definition eines Blockes Übertragungsfunktion ist es auch das Baukastensystem der numerischen Datenbearbeitung. Das klingt vielleicht vereinfachend und nicht sehr seriös – aber unterschätzen Sie Lego® bitte nicht (siehe Abb. 2.1). Für die Möglichkeiten zur Arbeit mit SIMULINK ist die Komplexität der einzelnen Bausteine sowie die Vielzahl der Kombinationsmöglichkeiten entscheidend. Und um uns über die Beschränkung auf die in SIMULINK vordefinierten Blöcke hinweg zu setzen, werden wir auch lernen, Blöcke selbst zu definieren. Auf diese Weise lassen sich u.a. Untersysteme eines größeren Systems kompakt darstellen.

§ **133** Gegenüber der Programmierung von Hand hat SIMULINK, wie bereits in der Einleitung angedeutet, einige Vor- und Nachteile:

<sup>1</sup>Alle Screenshots sind von einem PC-System mit MATLAB 2007B – auf Unix sieht es anders aus! Auch MATLAB 2008a sieht etwas anders aus – und verfügt bereits über einige zusätzliche Blöcke. Letzteres zeigt auch, dass SIMULINK ein dynamisches, sich weiter entwickelndes System ist.



Abbildung 2.1: Auch mit Lego® lassen sich komplexe Probleme wie Escher's unmögliche Bauten bearbeiten (<http://www.andrewlipson.com/escher/ascending.html>)

Vorteile	Nachteile
weniger anfällig für Codierungsfehler erzwingt Verwendung von Flussdiagramm komplexere Probleme bleiben übersichtlich viele vorprogrammierte Routinen Routinen aus MATLAB bekannt gute Tools für Daten	Test von Code nicht erzwungen weniger Beachtung von Stabilitäts- und Genauigkeitskriterien Solver teilweise sehr komplex

**Lernziele:** nach Durcharbeiten dieses Kapitels sollten Sie in der Lage sein

- die verschiedenen grundlegenden in SIMULINK zum Aufbau eines Modells zur Verfügung stehenden Blöcke in ihrer Funktion zu kennen und geeignet zu kombinieren.
- Simulationen für einfache physikalische Probleme zu konstruieren.
- Parameter, Randbedingungen und externe Treiber korrekt an die Simulation zu übergeben.
- die Simulation im Hinblick auf Genauigkeit und Stabilität kritisch zu hinterfragen.

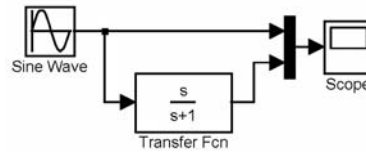
## 2.1 Erste Schritte: Start von SIMULINK

§ 134 In Abschn. 1.1.3 und 1.1.4 haben wir bereits verschiedene Varianten zur Simulation des Federpendels in SIMULINK kennen gelernt. Während in den Versionen in Abschn. 1.1.3 die Anlehnung an konventionelle numerische Verfahren am deutlichsten ist, ist die Simulation mit Hilfe der Übertragungsfunktion wie in Abb. 1.10 die einfachste Variante einer Simulation in SIMULINK. Und die Abbildung zeigt zugleich, wie sich z.B. Datenverarbeitung in SIMULINK darstellen lässt. Man benötigt lediglich drei Blöcke:

1. einen Eingabeblock (sei es wie in Abb. 1.10 eine Signalquelle oder der Zugriff auf Daten in einer Datei oder im Workspace),
2. die eigentliche Bearbeitung (beschrieben in diesem Fall durch die Übertragungssystem, sonst durch einen selbst erzeugten Block, in dem die zugehörigen Routinen zusammen gefasst sind) sowie
3. der Ausgabe (in diesem Fall einfach das Scope, sonst eine Datei oder der Workspace).


Zur Modellbildung stellt SIMULINK ein Graphical User Interface (GUI) zur Verfügung, in dem das Modell aus einfachen Blöcken mit bestimmten Eigenschaften erstellt werden kann. In

Abbildung 2.2: Tiefpass in SIMULINK



diesem Abschnitt werden wir in einem schnellen Überblick in die Verwendung von SIMULINK am Beispiel des Tiefpass einführen.<sup>2</sup>



§ 135 Zum Start von SIMULINK gibt es zwei Möglichkeiten: die Eingabe des Befehls `simulink` im Command Window von MATLAB oder die Verwendung des Buttons  aus der MATLAB-Werkzeugleiste. In beiden Fällen öffnet sich das Simulink Library Browser Fenster dessen Inhalt in Abschn. 2.2 genauer beschrieben wird.

§ 136 Ein Modell kann mit Hilfe des bereits erwähnten Graphical User Interfaces erzeugt werden. Dazu wird im Menü **File** der Menüpunkt **New** und dort das Untermenü **Model** ausgewählt: **File**  $\leftrightarrow$  **New**  $\leftrightarrow$  **Model**. Darauf öffnet sich ein Editor für das Modell. In diesem Fenster kann im Menü **View** über den Unterpunkt **Library Browser** der SIMULINK Library Browser ausgewählt werden. In diesem sind die für die Modellbildung zur Verfügung stehenden Blöcke in Untergruppen sortiert, z.B. Quellen und Senken. Durch Anklicken des betreffenden Icons öffnet sich das Untermenü. Im Fall von Quellen erhalten wir in diesem Untermenü u.a. einen Pulsgenerator, einen Signalgenerator oder eine einfache Sinuswelle. Letzteren Block klicken wir mit der linken Maustaste an und schieben ihn ins Simulationsfenster. Zurück im Hauptmenü des SIMULINK Library Browser wählen wir das Menü **Sinks** und darin den Unterpunkt **Scope**, der uns das Äquivalent eines Oszilloskops zur Verfügung stellt. Unter dem Hauptmenüpunkt **Continuous** finden wir einen allgemeinen Block **Transfer Function**, den wir als die Signal modifizierende Komponente ebenfalls im Modell-Editor ablegen. Die Eigenschaften dieses Blocks können wir später unseren Wünschen entsprechend modifizieren.

§ 137 Jetzt fehlen uns nur noch die Signalpfade. Wie viele Signale in einen Block hinein bzw. aus ihm heraus gehen können, wird durch die Anzahl der  $\rangle$  und  $\langle$  an diesem Block angezeigt: die Quelle erlaubt einen Ausgang, die Signal modifizierende Komponente einen Ein- und einen Ausgang, und das Scope hat nur einen Eingang. Um beide Signale (ursprüngliches Signal und Ausgang) auf den einen Eingang geben zu können, benötigen wir noch einen Multiplexer **Mux**, den wir ebenfalls unter dem Menüpunkt **Commonly Used Blocks** finden. Dieser hat zwei Eingänge und einen Ausgang, d.h. die beiden Signale werden als Vektor auf das Scope gegeben und dort komponentenweise dargestellt. Damit erhalten wir ein Modell entsprechend dem in Abb. 2.2.

§ 138 Die Verbindung der Blöcke erfolgt mit der linken Maustaste. Lediglich bei der Verzweigung hinter der Quelle muss man statt der linken die rechte Maustaste beim Verbinden verwenden.

§ 139 Nun ist das Modell bis auf die Spezifikation der Transfer-Funktion fertig. Hier erwartet SIMULINK eine Transferfunktion im Bildbereich, d.h. wir können Anleihe bei (4.14) nehmen. Die Übertragungsfunktion eines Tiefpass muss daher die Form

$$G(s) = \frac{1}{s + \frac{1}{RC}} = \frac{1}{s + \omega_0}$$

annehmen. Das entspricht der Struktur der von Simulink vorgegebenen Übertragungsfunktion: diese erwartet in Zähler und Nenner jeweils ein Polynom in  $s$ , wobei die Ordnung des Nenner-Polynoms größer gleich der des Zählerpolynoms sein muss. Die Koeffizienten werden für

<sup>2</sup>Das Beispiel **Building a Model :: Getting Started (Using Simulink)** aus der MATLAB-Hilfe entspricht im wesentlichen diesem Beispiel.

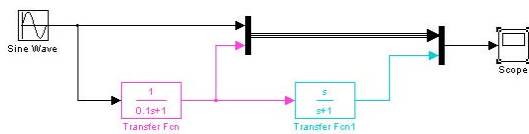


Abbildung 2.3: Der Bandpass lässt sich in SIMULINK durch hintereinander schalten der Übertragungsfunktion von Hoch- und Tiefpass erzeugen

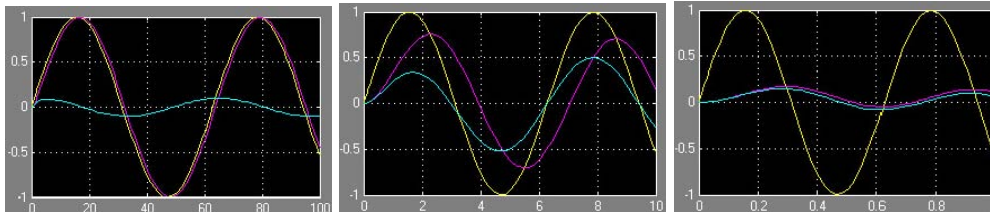


Abbildung 2.4: Signalverlauf für verschiedene Eingangsfrequenzen beim Bandpass aus Abb. 2.3; Eingang Gelb, Ausgang Tiefpass Magenta, Ausgang Bandpass Cyan.

Zähler und Nenner als Vektoren eingegeben, verschwindet einer der Vorfaktoren, so ist dieser als 0 explizit mit anzugeben. Simulink gibt als einfachste Übertragungsfunktion den Ausdruck

$$G(s) = \frac{1}{s + 1}$$

vor, das entspräche also einem Tiefpass mit der Grenzfrequenz  $f_g = 1$  rad/s. Die Koeffizienten in der Übertragungsfunktion lassen sich verändern, in dem man den Block mit der rechten Maustaste anklickt und aus dem Menü den Punkt **TransferFcn Parameter ...** anklickt – der Doppelklick mit der linken Maustaste auf das Icon ist die schnellere Variante. In dem sich dann öffnenden Fenster können die Koeffizientenvektoren für Zähler- und Nennerpolynom eingegeben werden – bitte die Nullen nicht vergessen; das letzte Element des Vektors ist jeweils die Konstante, d.h. der Faktor vor  $s^0$ .

§ 140 Die Eigenschaften der Signalquelle lassen sich ebenfalls durch Anklicken derselben mit der rechten Maustaste und Auswahl des Menüpunktes **Sin Parameters ...** ändern.

§ 141 Haben wir die Parameter der Signalquelle und der Transferfunktion eingestellt, so lässt sich die Simulation mit **Ctrl T** starten. Anklicken des Scope gibt uns dann eine graphische Ausgabe, entsprechend der auf einem Oszilloskop.<sup>3</sup> Details der Simulation, wie z.B. die Zeitauflösung, die Simulationsdauer und der Startpunkt der Datenausgabe (zum Abschneiden des Einschwingvorgangs) lassen sich auf der Steuerkarte für die Simulation eintragen – wir werden in Abschn. 2.2.5 darauf zurück kommen. Momentan müssen Sie gegebenenfalls mit etwas kantigen Kurven leben – und sollten diese auch als Hinweis auf die Probleme numerischer Verfahren im Gedächtnis behalten.

**Zwischenrechnung 5** Hier lässt sich mit verschiedenen Variationen der Grenzfrequenz und der Eingangsfrequenz spielen und so das Verhalten als Tiefpass überprüfen – tun Sie das.

**Zwischenrechnung 6** Andere einfache Übertragungsfunktionen sind der Integrator  $1/s$  oder  $s/(s + 1)$  für den Hochpass. Spielen Sie auch mit diesen.

§ 142 Interessant wird SIMULINK insbesondere bei Filtern höherer Ordnung. Diese basieren auf mehreren Bauelementen mit zeitlicher Verschiebung – so kann der Serienschwingkreis als Filter zweiter Ordnung verwendet werden. Da bei Filtern höherer Ordnung auch Resonanzen auftreten können, kann der Betrag der Übertragungsfunktion größer als Eins werden.

<sup>3</sup>Die Zuordnung der Farben zu den Kurven ist durch die Reihenfolge der Eingänge festgelegt. Erhält das Scope ein skalares Signal, so wird dieses in Gelb dargestellt. Bei vektorwertigen Signalen wird die erste Komponente ebenfalls in Gelb dargestellt; die zweite (im Beispiel entsprechend dem unteren Eingang des Multiplexers) wird in Magenta dargestellt, die weiteren werden in der Reihenfolge Cyan, Rot, Grün, Blau abgebildet. Sollen mehr Signale dargestellt werden, so werden diese Farben jeweils zyklisch wiederholt.

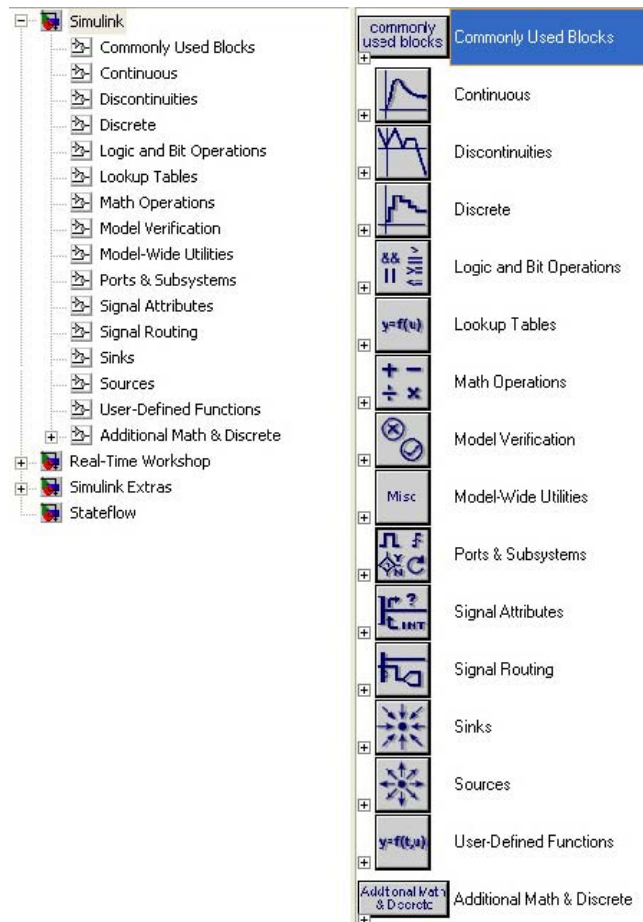


Abbildung 2.5: Übersicht über die verschiedenen Gruppen der in SIMULINK zur Simulation zur Verfügung stehenden Blöcke

Aber auch die Kombination von Filtern erster Ordnung ist möglich: so lässt sich z.B. ein Bandpass aus einem Hoch- und einem Tiefpass (jeweils mit geeigneten Grenzfrequenzen) konstruieren. Hier zeigt sich die eigentliche Stärke von SIMULINK (und der Rechnung mit Übertragungsfunktion): wir können die einzelnen Filter in SIMULINK einfach als Reihenschaltung wie in Abb. 4.4 hinter einander setzen, wie in Abb. 2.3 gezeigt. Signalverläufe für verschiedene Frequenzen (1/10 der Grenzfrequenz, Grenzfrequenz, 10fache Grenzfrequenz) sind in Abb. 2.4 gezeigt.

§ 143 Bevor wir uns in die eigentliche Simulation und ihre Steuerung vertiefen, wollen wir zuerst die einzelnen zur Verfügung stehenden Blöcke genauer betrachten. Dabei steht uns der SIMULINK Library Browser hilfreich zur Seite.

## 2.2 Der SIMULINK Library Browser

§ 144 Der SIMULINK Library Browser enthält alle Blöcke, die wir zur Simulation eines Systems in SIMULINK benötigen. Diese Blöcke sind in verschiedenen Untergruppen abgelegt. Beim Öffnen von SIMULINK erhalten wir eine Übersicht, ähnlich der in Abb. 2.5.<sup>4</sup>

§ 145 Da wir uns in unseren ersten Versuchen mit SIMULINK auf zeitkontinuierliche Systeme beschränken, sind die folgenden Untergruppen interessant:

<sup>4</sup>Die Ansicht ändert sich zwar von MATLAB-Version zu MATLAB-Version – die Grundstruktur, insbesondere die Sortierung der Blöcke in verschiedene Gruppen, ist im wesentlichen konstant.



- **Continuous:** enthält die Operationen wie Integration, Ableitung, Übertragungsfunktion oder verschiedene Delays, die sich auf zeitkontinuierliche Signale anwenden lassen.
- **Math Operations:** enthält im wesentlichen eine durch Symbole dargestellte Variante der meisten bereits aus MATLAB bekannten mathematischen Funktionen, inklusive der Manipulation von Vektoren und Matrizen.
- **Sinks:** enthält die Blöcke, die für die Ausgabe (graphisch, in eine Datei oder in den Workspace von MATLAB) benötigt werden.
- **Sources:** enthält verschiedene Signalformen, die als externer Antrieb verwendet werden können. Neben den konventionellen Signalgeneratoren (inklusive Rampe und Springfunktion) kann das externe Signal auch aus einer Datei oder dem Workspace bezogen werden. Letzteres ist insbesondere für die Verwendung von SIMULINK zur Signalbearbeitung von Bedeutung.

Die Untergruppe **Signal Routing** enthält unter anderem Multiplexer und Demultiplexer zum Zusammenführen und Aufspalten von Signalen – beide Blöcke sind auch in der Gruppe **Commonly Used Blocks** enthalten. Auf die Untergruppe **Ports & Subsystems** zum Bau von Untersystemen und zur Erstellung eigener Blöcke werden wir in Abschn. 2.4 eingehen.

§ 146 Die Gruppe **Commonly Used Blocks** enthält eine Zusammenstellung der (nach Meinung von MATHWORKS) am häufigsten benötigten Blöcke. Für alle in Kapitel 1 gezeigten Simulationen sind die Blöcke in dieser Gruppe ausreichend.

§ 147 Alle Blöcke sind in einer abstrakten Form gegeben. Anklicken eines Blocks mit der rechten Maustaste öffnet ein Menü, in dem unter anderem die Punkte 'Block Properties' und, je nach Block, z.B. 'Gain Parameters' oder 'Integration Parameters' eingestellt werden können. Anklicken der entsprechenden Punkte öffnet eine Karteikarte in der die Eigenschaften des Blockes (relevant für die technischen Aspekte der Simulation) und die Parameterfunktion des Blockes (z.B. die physikalischen Parameter, die Systemparameter oder die Anfangsbedingungen) eingetragen werden können.

§ 148 Um die Möglichkeiten von SIMULINK besser zu verstehen, ist ein genauerer Blick in die einzelnen Gruppen von Blöcken sinnvoll.

### 2.2.1 Blockgruppe Continuous

§ 149 Die zur Verarbeitung zeitkontinuierlicher Signale benötigten Blöcke sind in Abb. 2.6 zusammen gefasst. Die oberen vier Blöcke sind für uns am wichtigsten:

- **Derivative** bildet die Ableitung einer (vektorwertigen) Funktion.
- **Integrator** integriert eine (vektorwertige) Funktion, wobei die Anfangsbedingung(en) in der Karteikarte 'Integrator Properties' angegeben werden können.
- **State Space** erlaubt die Darstellung eines Systems nicht durch die Übertragungsfunktion sondern mit Hilfe der Systemgleichungen (vgl. (4.2)).
- **Transfer Fcn** erlaubt die Spezifikation einer Übertragungsfunktion in der aus § 528 bekannten Polynomdarstellung.

#### Derivative

§ 150 Der Block **Derivative** bildet die Ableitung des Signals. Ist das Signal vektorwertig, so wird ein Vektor der Ableitungen der jeweiligen Komponenten gebildet. Die Karteikarte mit den diesem Block zugeordneten Parametern ist zu minimalistisch, um ihr eine eigene Abbildung zu geben. Sie enthält ein Informationsfeld bezüglich der Eigenschaften des Blocks:

**Derivative:** Numerical derivative:  $du/dt$



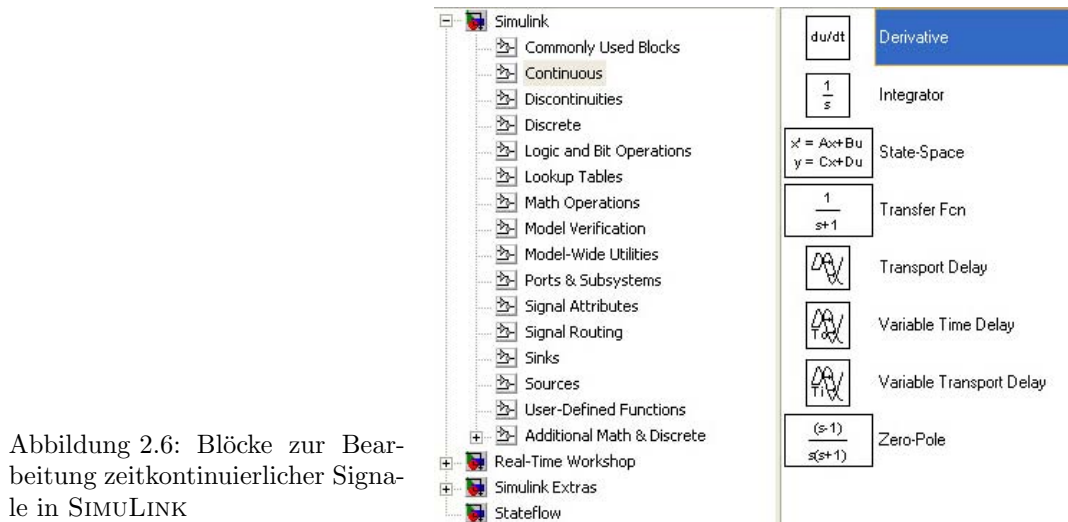


Abbildung 2.6: Blöcke zur Bearbeitung zeitkontinuierlicher Signale in SIMULINK

Der einzige einstellbare Parameter ist die Zeitkonstante für die Linearisierung, formal gegeben als  $s/(Ns + 1)$ .<sup>5</sup> Der Defaultwert ist `inf`. Diese Option wird jedoch erst dann interessant, wenn wir uns mit diskreten Signalen beschäftigen.

§ 151 Die Karteikarte enthält, wie natürlich auch die Karteikarten der anderen Blöcke, eine Schaltfläche `Help`, die ein Fenster mit der MATLAB-Hilfe zu diesem Block öffnet. Wie aus der MATLAB-Hilfe bereits bekannt, werden Funktion, Anforderungen an In- und Output, etwaige default-Einstellungen, unterstützte Datentypen usw. recht genau beschrieben.

§ 152 Inhaltlich greift SIMULINK im Block `derivative` auf die bereits aus MATLAB bekannte Funktion `diff` zurück. Zu `derivative`, ebenso wie zu den anderen Blöcken, finden Sie die notwendigen Informationen über die Hilfe zu den einzelnen Blöcken bzw. teilweise auch über die allgemeine MATLAB-Hilfe. In der Hilfe finden Sie in der Regel auch Informationen über die Kontrolle der Genauigkeit – sonst müssen Sie diese für die in der genaueren Spezifikation des Blocks eingestellten Solver unter dem entsprechenden MATLAB-Solver nachschlagen.

§ 153 Der Block `derivative` ist aus verschiedenen Gründen nicht ganz einfach zu verwenden: (a) wie von `diff` bekannt, ändert sich die Länge des Signals (Probleme beim Vergleich mit dem Ausgangssignal) und (b) die  $\delta$ -Pulse bei der Ableitung eines diskreten Signals sind für die Weiterverwendung ebenfalls suboptimal. In der MATLAB-Hilfe findet sich unter `Modeling Equations :: Modeling with Simulink` deshalb ein Abschnitt ‘Choosing Best-Form Mathematical Models’, in dem die Umformulierung eines Modells zur Vermeidung derartiger Probleme vorgeschlagen wird.

## Integrator

§ 154 Der Block `Integrator` integriert ein gegebenenfalls vektorwertiges Signal komponentenweise. Die Karteikarte für diesen Block erklärt als erstes wieder seine Funktion:

**Integrator:** Continuous-time integration of the input signal

Die Hilfe spezifiziert dieses Statement etwas genauer: ‘The following equation represents the output of the block  $y$  as a function of its input  $u$  and an initial condition  $y_0$ , where  $y$  and  $u$

<sup>5</sup>Formal greift SIMULINK beim diskreten Signal auf die  $z$ -Transformation (siehe auch Abschn. 5.4.1) zurück – das ist eine Art Laplace Transformation für diskrete Signale. Die Ableitung diskreter Signale unterscheidet sich von der kontinuierlicher Signale insofern, als dass sich bei Änderung des Funktionswertes ein  $\delta$ -Peak ergibt, sonst die Ableitung wie gewohnt Null ist. Es entsteht aber natürlich keine kontinuierliche Ableitung.

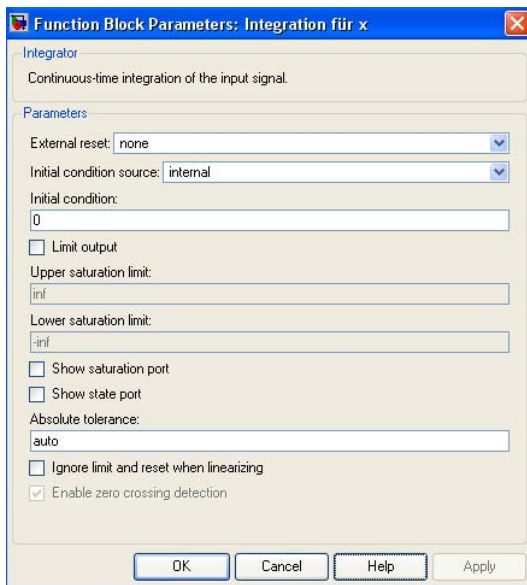


Abbildung 2.7: Karteikarte zur Eingabe der Parameter des Integrators. Für uns sind das zweite und dritte Feld wichtig, da sie die Eingabe der Anfangsbedingung erlauben

are vector functions of the current simulation time  $t$ :

$$y(t) = \int_{t_0}^t u(t) dt + y_0 .$$

Diese Darstellung ist natürlich nicht korrekt: SIMULINK ist ein numerisches Verfahren, d.h. die Integration wird numerisch nach einem auf Finiten Differenzen basierenden Schema ausgeführt. In der Hilfe wird dennoch die differentielle Form gegeben, da alle Darstellungen mit Hilfe finiter Differenzen sich bereits auf ein konkretes Verfahren beziehen würden. Letzteres wird jedoch nicht mit dem Integratorblock sondern mit der Simulationskontrolle bestimmt.

§ 155 Die Hilfe hilft uns auch beim Verständnis der anderen auf der in Abb. 2.7 gezeigten Karteikarte einzutragenden Parameter. Die ersten drei Eingaben betreffen inhaltliche Aspekte, die folgenden Zeilen beziehen sich eher auf die technischen Details der Simulation. Die erste Zeile erlaubt die Spezifikation von Bedingungen, unter denen der Integrator auf seinen Anfangszustand zurück gesetzt wird; ein Beispiel ist in § 318 gegeben.

§ 156 Die wichtigsten an den Integrator zu übergebenden Parameter betreffen die Anfangsbedingungen. Im zweiten Feld können wir wählen, ob die Anfangsbedingungen extern oder intern vorgegeben werden sollen. Entscheiden wir uns für den Fall intern, so werden die Anfangsbedingungen im dritten Feld eingegeben. Entscheiden wir uns für eine externe Eingabe der Anfangsbedingungen, so verschwindet die dritte Zeile aus der Karteikarte. Der Integratorblock erhält dann einen zusätzlichen Eingang, über den die Anfangsbedingung zugeführt werden kann – aus dem Workspace, aus einer Datei oder von einem anderen Punkt innerhalb der Simulation; der springende Ball in § 316 wird uns als Beispiel dienen.

§ 157 Mit Hilfe des Integrator-Blocks können wir eine einfache Simulation zusammen setzen, siehe Abb. 2.8. Außer dem Integratorblock benötigen wir eine Funktion zur Integration – in diesem Fall machen wir es uns einfach und wählen aus der Blockgruppe Sources die Funktion **Ramp** zur Darstellung einer linearen Funktion. Für die Ausgabe wählen wir ebenfalls die einfachste Möglichkeit, die graphische Darstellung mit Hilfe des **Scope** aus der Blockgruppe Sinks. Doppelklick mit links auf einen der drei Blöcke zeigt dessen Parameter. So können wir bei der Rampe einen Offset, die Steigung und die Startzeit vorgeben; die Eingabemöglichkeiten für den Integratorblock sind im Zusammenhang mit Abb. 2.7 bereits beschrieben.

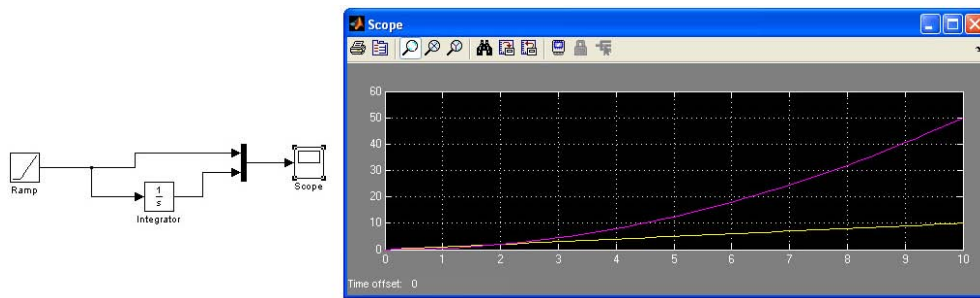


Abbildung 2.8: Integration einer linearen Funktion – die Quelle **Ramp** wird zur Erzeugung der Funktion benutzt

**Zwischenrechnung 7** Spielen Sie etwas mit der Rampe (Veränderung der Anfangsbedingungen, der Steigung der Rampe und eines eventuellen Offsets) und anderen zu integrierenden Signalen (andere Sources verwenden).

### State Space

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned}$$

§ 158 Der Block **State Space** erlaubt die Eingabe einer Systemmatrix mit Hilfe der zugehörigen Karteikarte. Diese erläutert als erstes die Funktion des Blocks:

**State Space:** State-space model  $dx/dt = Ax + Bu$ ,  $y = Cx + Du$

Die Gleichungen entsprechen den in (4.2) gegebenen, d.h. wir können an dieser Stelle ein System mit Hilfe seiner Systemmatrix spezifizieren ohne die Übertragungsfunktion angeben zu müssen.

§ 159 Auch ein Blick in die Hilfe erläutert diese kurze Beschreibung genauer. Der Block **State-Space** dient zur Beschreibung eines Systems, dessen Verhalten durch die Gleichungen

$$\dot{\vec{x}} = A\vec{x} + B\vec{u} \quad \text{und} \quad \vec{y} = C\vec{x} + D\vec{u}$$

beschrieben wird. Darin ist  $\vec{x}$  der Zustandsvektor,  $\vec{u}$  der Eingabevektor und  $\vec{y}$  der Ausgabevektor, entsprechend der Beschreibung eines Systems durch (4.2). Mit  $n$  als der Zahl der Zustände ist  $A$  eine  $n \times n$ -Matrix.  $B$  ist eine  $n \times m$ -Matrix, wobei  $m$  der Zahl der Eingaben und damit der Länge des Eingabevektors  $\vec{u}$  entspricht. Entsprechend ist mit  $r$  als der Zahl der Ausgaben (oder der Länge des Ausgabevektors)  $C$  eine  $r \times n$ -Matrix und  $D$  eine  $r \times m$ -Matrix. Das sind genau die Spezifikationen eines LTI-Systems wie in Abschn. 4.1.2 beschrieben; die erzwungene Schwingung als Beispiel haben wir bereits in Abschn. 1.1.4 diskutiert.

§ 160 Die Karteikarte ist selbsterklärend: es werden die vier Matrizen in den entsprechenden Größen benötigt, sowie ein Vektor mit den Anfangsbedingungen. Im letzten Feld kann eine Toleranz zur Bestimmung der Genauigkeit bei der Lösung dieses Blocks angegeben werden. Mit der Einstellung **auto** bestimmt SIMULINK diesen Wert selbst.

### Transfer Function

$$\frac{1}{s+1}$$

§ 161 Während der Block **State Space** eine explizite Beschreibung des Systems mit Hilfe der Systemmatrix enthält, verwendet der Block **Transfer Fcn** die direkte Darstellung der Input-Output-Relation als Übertragungsfunktion. Diese lässt sich wie in Abschn. 4.3 beschrieben als Quotient zweier Polynome der komplexen Frequenz  $s$  darstellen:

$$G(s) = \frac{y(s)}{u(s)} = \frac{a_1 s^{n-1} + a_2 s^{n-2} + \dots + a_n}{b_1 s^{m-1} + b_2 s^{m-2} + \dots + b_m} \quad \text{mit} \quad m > n.$$

§ 162 In der Definition des Blocks auf seiner Karteikarte sind die Details der Eingabe ebenfalls spezifiziert:

**Transfer Fcn:** The numerator coefficient can be a vector or matrix expression. The denominator coefficient must be a vector. The output width equals the number of rows in the numerator coefficient. You should specify the coefficients in descending order of powers of  $s$ . Und dabei bitte auch die Koeffizienten angeben, die Null werden – SIMULINK zählt einfach die Zahl der Koeffizienten, um die Ordnung des Polynoms zu bestimmen.

§ **163** Der Block **Transfer Fcn** ist ein Single-input multiple-output SIMO Block: er verarbeitet eine skalare Eingangsgröße und erzeugt eine vektorielle Ausgangsgröße. Bei letzterer entspricht jede Komponente einem mit Hilfe einer anderen Übertragungsfunktion erzeugten Ausgangssignal. Die Zahl der Komponenten des Ausgangssignals ist durch die Zeilen der Nennermatrix in der Transferfunktion bestimmt: jeder Zeilenvektor entspricht dem Nenner einer eigenen Übertragungsfunktion. Auf diese Weise lässt sich die Wirkung verschiedener Übertragungsfunktionen auf einfache Weise vergleichen.

§ **164** Hat man einmal die Übertragungsfunktion eines Systems bestimmt, so bietet SIMULINK mit Hilfe dieses Blocks eine einfache Möglichkeit, diese auf die verschiedensten Eingangssignale anzuwenden. Wir werden in Kap. 5 näher darauf eingehen.

**Verständnisfrage 18** Brauchen wir den Block **Transfer Function** überhaupt, wenn sich doch alles durch den Block **State Space** erschlagen ließe? Diskutieren Sie Vor- und Nachteile.

**Zwischenrechnung 8** Greifen Sie auf ein System mit bekannter Systemgleichung und bekannter Übertragungsfunktion zurück. Vergleichen Sie die Ergebnisse beider Verfahren.

## 2.2.2 Blockgruppe Math Operations

§ **165** Die verschiedenen mathematische Operationen enthaltenen Blöcke sind in Abb. 2.9 zusammen gefasst. Die für unsere Zwecke wichtigsten Blöcke sind:

- **Divide** multipliziert oder dividiert Systemgrößen. Wie von MATLAB gewohnt, können diese Operationen punktweise oder entsprechend der Regeln der Matrixmultiplikation durchgeführt werden.
- **Gain** multipliziert Systemgrößen mit Konstanten. Auch hier kann die Operation elementweise oder entsprechend den Regeln der Matrixmultiplikation durchgeführt werden.
- **Math Function** umfassen Logarithmen, die Exponentialfunktion, Potenzen und den Modulus. Die Funktion kann ein oder mehrere Argumente haben – in letzterem Fall hat der Block entsprechend viele Eingänge.
- **Sum** erlaubt die Erzeugung eines Summationspunktes, an dem mehrere Input-Signale addiert (oder auch subtrahiert) werden können.
- **Trigonometric Function** erlaubt die Anwendung trigonometrischer und hyperbolischer Funktionen auf ein Signal. Hat die Funktion mehr als ein Argument, so wird die Zahl der Eingänge in den Block entsprechend angepasst.

### Divide

§ **166** Im Gegensatz zu den bisher beschriebene Blöcken hat der Block **Divide** nicht nur einen sondern mindestens zwei Eingänge.<sup>6</sup> In seiner Karteikarte wird seine Funktion charakterisiert als



**Divide:** Multiply or divide inputs. Choose element-wise or matrix product

§ **167** Auf der zugehörigen Karteikarte ist die Zahl der miteinander zu verknüpfenden Signale und damit auch der Eingänge zu spezifizieren. Dies erfolgt über eine Liste aus den Symbolen ‘\*’ und ‘/’, jeweils für Multiplikation oder Division. Im nächsten Feld lässt sich auswählen, ob diese Operationen Punktweise oder als Matrixoperationen ausgeführt werden sollen.

<sup>6</sup>Mehr als zwei Eingänge klingt für die Division etwas befremdlich; da der Block aber auch für die Multiplikation verwendet werden kann, ist das ok.

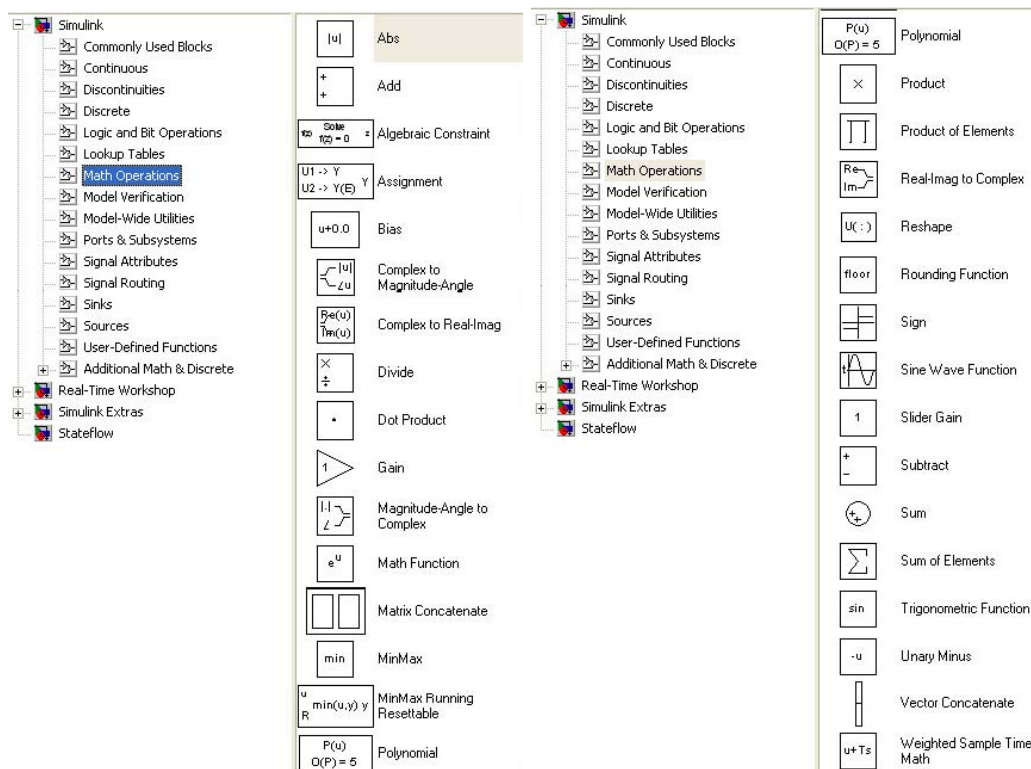


Abbildung 2.9: Blöcke für mathematische Operationen in SIMULINK. Die Blöcke geben schematische Darstellungen der auch aus MATLAB bekannten mathematischen Funktionen

§ 168 Im unteren Feld lässt sich in dieser Karteikarte, wie auch in den Karten der folgenden Blöcke, jeweils eine Sample Time einstellen. Durch diese kann der zeitliche Abstand zwischen zwei aufeinander folgenden Samples explizit angegeben werden. Falls es keine Gründe für die Wahl einer speziellen sample time gibt, sollte der Parameter auf -1 gesetzt bleiben: dann wird die sample time des Gesamtschemas verwendet.

§ 169 Verwandte Blöcke sind **Dot Product** zur Ausführung des Skalarprodukts (zwei vektorielle Signale werden punktweise multipliziert und die Produkte werden addiert), **Product** mit einer nahezu identischen Beschreibung und Syntax sowie **Product of Elements**.<sup>7</sup>

### Gain



§ 170 Der Block **Gain** beschreibt ebenfalls eine Multiplikation, jedoch nicht von Signalen untereinander sondern mit einem als Parameter spezifizierten konstanten Faktor  $K$ :

**Gain:** Element-wise gain ( $y=K.*u$ ) or matrix gain ( $y=K*u$  or  $y=u*K$ )

Während das zu bearbeitende Signal ein Skalar oder ein Vektor sein kann, kann der Gain-Faktor ein Skalar, ein Vektor oder eine Matrix sein.

§ 171 Die zu dem Block gehörige Karteikarte besteht aus drei Elementen. Im Main-Element werden der Gain-Faktor und die Art der Multiplikation spezifiziert. Die beiden anderen Elemente erlauben eine Spezifikation der Datentypen des Signals (Signal Data Types) und der Parameter (Parameter Data Types). Nähere Informationen zu diesen beiden Punkten finden Sie in der MATLAB-Hilfe.

<sup>7</sup>Aus der Auflistung erkennen wir eine Eigenschaft, die SIMULINK von MATLAB geerbt hat: viele Wege führen nach Rom bzw. eine Verknüpfung von Größen lässt sich oftmals durch verschiedenen MATLAB-Funktionen oder SIMULINK-Blöcke realisieren.

## Math Function

§ 172 Auf ein Signal kann mit Hilfe des Blocks **Math Function** eine mathematische Funktion angewandt werden. Auch hier liefert die Karteikarte die notwendigen Erklärungen:



**Math:** Mathematical functions including logarithmic, exponential, power, and modulus functions.

Die gewünschte Funktion kann über ein Pulldown Menü ausgewählt werden. Außerdem kann die Art des Output-Signals eingeschränkt werden: komplex, real oder auto.

§ 173 Für den letzten Eintrag gilt wieder das bereits in § 168 gesagte: die sample time sollte nur verändert werden, wenn es dafür gute Gründe gibt. Im zweiten Element der Karteikarte geht es um den Umgang mit dem Datentyp im Falle eines Overflows – wir werden dieses Problem vorerst vernachlässigen.

§ 174 Beim Durchscannen der im Pulldown-Menü gegebenen Funktionen fällt auf, dass die ebenfalls häufig benötigten Winkelfunktionen fehlen. Diese finden sich in einem separaten Block **Trigonometric Functions** (s.u.).

## Sum

§ 175 Die Kombination von Signalen erfolgt im Block **Sum**. Dieser kann (über den ersten Eintrag der Karteikarte) rund oder rechteckig dargestellt werden. Die runde Darstellung hat einen gewissen Charme, da sie an den Summationspunkt erinnert und damit diesen Block als ‘Signalkombinierer’ gegenüber den anderen rechteckigen Blöcken abhebt. Die Zahl der erforderlichen Eingänge wird im zweiten Feld der Karteikarte eingestellt, wobei jedes ‘+’ einem zu addierenden, jedes ‘-’ einem zu subtrahierenden Eintrag entspricht. Die Eingangsgrößen können Skalare, Vektoren oder Matrizen sein – aber für jeden einzelnen Summationspunkt natürlich einheitlich.



§ 176 Das zweite Element der Karteikarte, die ‘Signal Data Types’, sollte eines kurzen Blicks gewürdigt werden, da dort z.B. eingefordert werden kann, dass alle Eingangsgrößen vom gleichen Datentyp sein müssen. Ist das nicht der Fall, so hat SIMULINK feste Verknüpfungsregeln für die unterschiedlichen Datentypen (für Details siehe die MATLAB-Hilfe). Auch der Datentyp der Ausgabe lässt sich entweder auf der Basis der MATLAB-Regeln festlegen (Default-Einstellung) oder explizit vorgeben.

§ 177 Die Blöcke **Add**, **Subtract** und **Sum of Elements** sind andere Darstellungsformen der gleichen Operation – wählen Sie den Block, der Ihnen die Simulation am übersichtlichsten darstellt. Aufgrund dieser Gleichheit haben alle vier Blöcke nur eine Hilfe; die zugehörige Karteikarte liefert entsprechend als Beschreibung der Funktion

**Sum:** Add or subtract inputs

Die Hilfe spezifiziert etwas genauer: ‘The Sum block performs addition or subtraction on its inputs. This block can add or subtract scalar, vector, or matrix inputs. It can also collapse the elements of a single input vector.’

## Trigonometric Function

§ 178 Die bei den Math Functions bereits vermissten trigonometrischen und hyperbolischen Funktionen finden sich im Block **Trigonometric Function**. Dies besagt auch die Spezifikation auf der Karteikarte:



**Trigonometric Function:** Trigonometric and hyperbolic functions

Wie bei den mathematischen Funktionen wird die gewünschte Funktion über ein Pulldown-Menü ausgewählt. Als weiterer Parameter kann wieder der gewünschte Typ des Ausgangssignals angegeben werden.

§ 179 Verwandte Blöcke sind **Sine Wave Function** und **Math Function**; letztere ist nur formal verwandt, erstere auch inhaltlich.

Abbildung 2.10: In der Untergruppe Sinks sind die in SIMULINK zur Darstellung von Signalen zur Verfügung stehenden Blöcke zusammengefasst

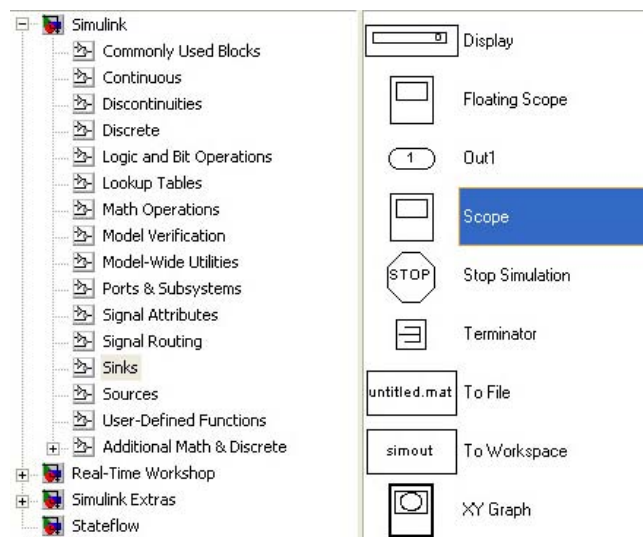
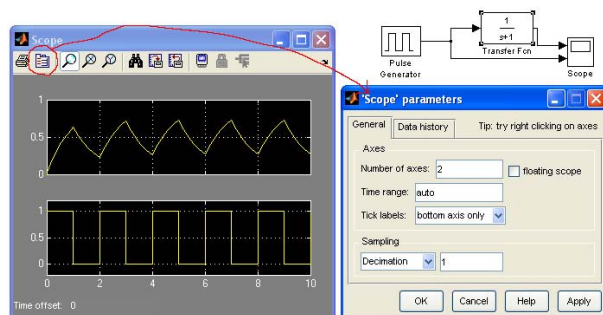


Abbildung 2.11: Scope mit mehr als einem Eingang, so dass sich z.B. gefiltertes und Eingangssignal in separaten Tracks darstellen lassen



### 2.2.3 Blockgruppe Sinks

§ 180 Die Blockgruppe Sinks enthält alle Senken für Daten, wobei mit Senken Speicher- und Darstellungsmöglichkeiten für Daten gemeint sind; eine Übersicht ist in Abb. 2.10 gegeben.



§ 181 Für die einfache und schnelle Darstellung von Daten werden wir uns in der Vorlesung auf das Scope beschränken – wie der Name schon andeutet eine Darstellung Signal vs. Zeit, wie wir sie vom Oszilloskop kennen. Mit dem Multiplexer Mux können Signale kombiniert und auf ein Scope gegeben werden (entspricht einem Mehrkanaloszilloskop), wie z.B. in den Beispielen in Kap. 1 oder Abschn. 2.1. Alternativ kann ein Oszilloskop mehrere Eingänge haben wie z.B. in Abb. 2.44.<sup>8</sup> Dann werden die verschiedenen Eingänge jeweils in eigenen Fenstern dargestellt.

§ 182 Der Block XY Graph bietet die Möglichkeit der Darstellung einer zweidimensionalen Bewegung oder der Darstellung einer eindimensionalen Bewegung im Phasenraum: hier werden nicht wie beim Scope  $x(t)$  und  $y(t)$  dargestellt sondern  $t$  wird als Parameter verwendet und  $y(x)$  dargestellt.

**Zwischenrechnung 9** Schiefer Wurf mit Reibung: erstellen Sie die Differentialgleichungen für dieses Problem. Erstellen Sie ein SIMULINK-Modell zur Lösung des Problems und stellen Sie in diesem die Wurfparabel graphisch dar.

<sup>8</sup>Letztere Variante wird nicht in allen SIMULINK-Versionen unterstützt – die Zahl der separaten Tracks und damit die Zahl der Eingänge lässt sich nicht über die Parameter des Blocks Scope einstellen sondern bei geöffnetem Fenster des Scopes über Anklicken der Scope Karte wie in Abb. 2.11 für ein auf einen Tiefpass gegebenes Rechtecksignal gezeigt.



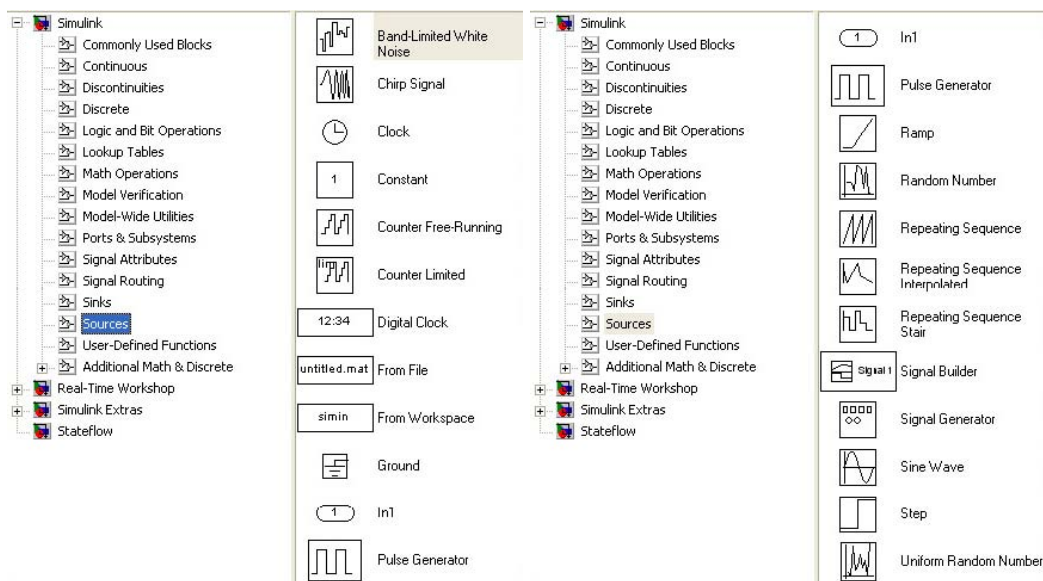


Abbildung 2.12: Blöcke zur Darstellung externer Antriebe in SIMULINK

§ 183 Für die Speicherung oder weitere Darstellung/Verarbeitung von Signalen sind die Blöcke **To File** und **To Workspace** hilfreich. Ersterer schreibt Simulationsergebnisse in eine Datei; der Name der Datei ebenso wie die in diese zu schreibenden Variablen ist in der zugehörigen Karteikarte zu spezifizieren. Letztere übergibt die auf der Karteikarte spezifizierten Variablen an den MATLAB Workspace – wo sie weiter bearbeitet (auch gespeichert) werden können. Wir werden in Abschn. 2.3.3 näher darauf eingehen; allerdings werden wir in Abschn. 2.5 auch die Variante der Simulationssteuerung direkt aus dem Workspace und damit der Rückgabe der Lösungen an selbigen betrachten.



§ 184 In der Rubrik **SimuLink Extras** gibt es eine Blockgruppe **Additional Sinks**. Diese enthält unter anderem einen Block zur Bestimmung des Leistungsdichtespektrums, siehe auch Abschn. 3.2.5 und insbesondere Abb. 3.20.

## 2.2.4 Blockgruppe Sources

§ 185 Der entgegengesetzte Vorgang, das Einfüttern von Signalen in die Simulation kann ebenfalls auf zwei Weisen erfolgen: zum einen können diese Signale mit Hilfe des Blocks **From File** aus einer in der zugehörigen Karteikarte zu spezifizierenden Datei oder mit Hilfe des Blocks **From Workspace** aus dem MATLAB Workspace eingelesen werden. Im Gegensatz zu MATLAB ist SIMULINK im Hinblick auf die Eingabeformate etwas eingeschränkt: eingelesen werden nur \*.mat-Dateien. Andere Datensätze, insbesondere Audio-, Bild- oder Video-Dateien müssen einmal separat in MATLAB eingelesen werden und dann entweder konvertiert in eine \*.mat-Datei geschrieben oder direkt aus dem Workspace eingelesen werden.



§ 186 Zum anderen stehen, in Analogie zum aus der Elektronik bekannten Oszilloskop als Senke, eine Vielzahl von Puls-, Signal- und Sinusgeneratoren als Quellen zur Verfügung. Teilweise haben diese Generatoren überlappende Funktionen (oder sind Spezialfälle eines allgemeineren Generators). Die Entscheidung für einen davon ist Geschmackssache – und sollte sich auch an der Übersichtlichkeit des zugehörigen Symbols in der Darstellung orientieren: wird ein Funktionsgenerator nur zur Erzeugung eines Sinus benötigt, so macht die Verwendung des Blocks Sinusgenerator das Diagramm einfacher lesbar.

§ 187 Für die Anwendung von Filtern und das Verständnis natürlicher Systeme sind insbesondere auch die Quellen von Bedeutung, mit denen sich Rauschen und/oder stochastische

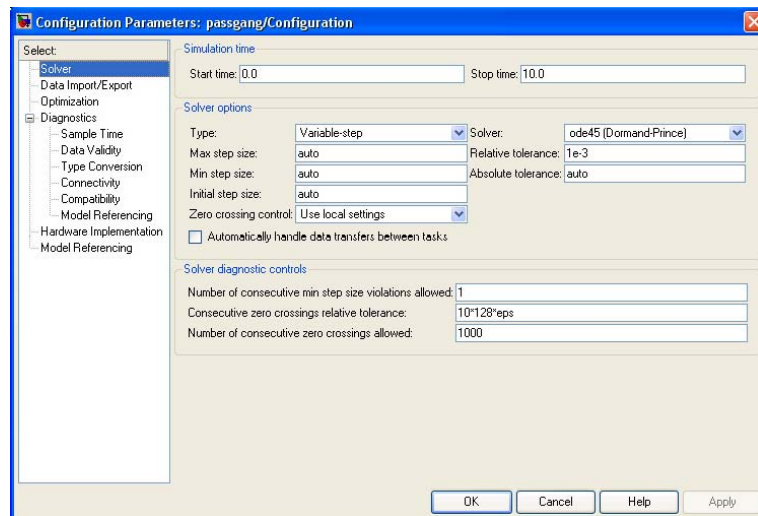





Abbildung 2.13: Simulationskontrolle in SIMULINK, die Karte zur Kontrolle des Solvers ist geöffnet

Signale erzeugen lassen. Auch dafür stehen verschiedene Generatoren zur Verfügung, siehe auch Abb. 2.12.

## 2.2.5 Ausführen der Simulation

§ 188 Der Start der Simulation kann über den Button  in der Werkzeugleiste von SIMULINK erfolgen, über die Tastenkombination CTRL + T oder über den Menüpunkt **Simulation**  **Start**.

§ 189 Vor dem Start der Simulation sollten jedoch noch die Parameter der Simulation eingestellt werden. Die Zeit, über die die Simulation laufen soll, lässt sich in einem Fenster in der Werkzeugleiste eintragen – als Default nimmt SIMULINK 10.0 Zeiteinheiten an. Die anderen Simulationsparameter lassen sich auf einer Karteikarte vereinbaren; diese kann über die Tastenkombination CTRL + E oder über den Menüpunkt **Simulation**  **Configuration Parameters** geöffnet werden. Die Karte für den Tiefpass aus Abb. 2.2 ist in Abb. 2.13 gezeigt.

§ 190 Die Simulationskontrolle bietet eine Vielzahl von Möglichkeiten; für den Anfang sollten wir uns auf die ersten beiden in der linken Übersicht genannten Punkte beschränken: Solver und Data Import/Export. Von diesen ist die Auseinandersetzung mit dem Solver die wichtigste, da diese die Genauigkeit (und Stabilität) der Simulation bestimmen. Eine detailliertere Diskussion der Simulationskontrolle folgt in Abschn. 2.6.

§ 191 Die ersten beiden Felder auf der Solver-Karte dienen der Definition des Simulationszeitraums. Der betrachtete Zeitraum sollte sinnvollerweise einige charakteristische Zeiten des Systems umfassen. Die Wahl der Startzeit hängt von der zu untersuchenden Frage ab: wird sie auf Null gesetzt, so enthält das Simulationsergebnis etwaige Einschwingvorgänge. Sind diese das Ziel der Untersuchung, so sollte der Zeitraum bei Null beginnen und bis zum Ende des Einschwingvorgangs reichen. Ist dagegen der anschließende stationäre Zustand das Ziel der Untersuchung, so braucht die Simulation auch erst ab diesem Zeitpunkt protokolliert zu werden.

§ 192 Die folgenden Einstellungen sind für die numerische Genauigkeit des Verfahrens von Bedeutung. Die von SIMULINK angebotene Default Version kann zumindest für einen ersten Überblick übernommen werden. In der Regel handelt es sich dabei um bereits aus MATLAB bekannte Solver für Differentialgleichungen mit variabler Schrittweite. Hat man es nicht zu eilig (und eine Vorstellung bezüglich einer sinnvollen Schrittweite), so kann man im linken Teil der Karte auch eine feste oder zumindest eine maximale Schrittweite einstellen. Ebenso kann es sinnvoll sein, im rechten Teil die relative Genauigkeit ein wenig zu tunen.

**Verständnisfrage 19** Sie begegnen hier dem Problem der numerischen Genauigkeit innerhalb eines Blocks. Ein System besteht aus vielen Blöcken – jeder mit seiner eigenen numerischen (Un-)Genauigkeit. Wie schätzt man die Genauigkeit der Gesamtsimulation ab? Müssen formal vergleichbare Blöcke alle jeweils mit dem gleichen Schema, Fehler und der gleichen Schrittweite laufen oder dürfen sich diese unterscheiden bzw. kann es sogar sinnvoll sein, dass sich diese unterscheiden? Sie können viele Analogien in den Verfahren zur numerischen Lösung umfangreicherer partieller Differentialgleichungen finden.

**Wiederholungsfrage 3** Die von SIMULINK verwendeten Solver gehen alle von einer variablen Schrittweite in der Zeit aus. Erläutern Sie die Grundidee derartiger Verfahren und diskutieren Sie Vor- und Nachteile gegenüber numerischen Verfahren mit fester Schrittweite.

**Wiederholungsfrage 4** Von einem numerischen Verfahren wird nicht nur Genauigkeit erwartet sondern auch Stabilität – die numerische Lösung für die freie Schwingung im linken und mittleren Teilbild in Abb. 1.4 verletzt ganz offenbar jegliche Stabilitätskriterien. Wie sind Stabilitätskriterien bei Verfahren mit fester Schrittweite definiert, wie bei Verfahren mit variabler Schrittweite?

## 2.3 Einfache Modelle: Federpendel und Verwandte(s)

§ 193 Als einfachstes Modell für ein vertieftes Spielen mit SIMULINK kann uns das bereits in der Einleitung etwas überstrapazierte Federpendel in allen seinen Variationen dienen. Am einfachen Federpendel können wir einmal einen Blick in die Spezifikationen der einzelnen Blöcke werfen; außerdem können wir die Grundlagen für eine animierte Darstellung kennen lernen. Am Beispiel gekoppelter Federpendel werden wir diese Grundlagen in Kap. 6 vertiefen und uns etwas Gedanken über Chaos und Emergenz machen. Die formalen Grundlagen für das Federpendel sind in Abschn. 1.1 ausführlich diskutiert; die SIMULINK-Simulationen wurden in Abb. 1.6–1.8 bereits gezeigt. Einige Abweichungen vom einfachen Federpendel werden wir ebenso kennen lernen wie einige Details zur Simulation(skontrolle).

### 2.3.1 Federpendel: Die Simulation

§ 194 Die Simulation der freien Schwingung in Abb. 1.6 ist ein minimalistisches Modell. Die interessierenden Größen ist der Ort  $x(t)$  (und die Geschwindigkeit  $v(t)$ ); die einzige wirkende Kraft ist die rücktreibende Kraft  $-kx$ . Die Simulation lässt sich dann mit Hilfe von drei Blöcken realisieren:

1. im Block **Gain** wird die Auslenkung mit  $\omega_0^2$  multipliziert. Das Ergebnis ist die Beschleunigung  $a = -\omega_0^2 x$ .
2. Diese wird im ersten Integratorblock integriert. Das Ergebnis ist die Geschwindigkeit:  $v = \int a dt$ .
3. Diese wird im zweiten Integratorblock integriert. Das Ergebnis ist der Ort:  $s = \int v dt$ . Dieser wird als Ausgabe auf das **Scope** gegeben und ist gleichzeitig Eingabe für den nächsten Durchlauf dieser Schleife.

§ 195 Betrachten wir jetzt die Blöcke genauer. Für die Simulation verwenden wir dimensionslose Größen und setzen Masse  $m$  und Federkonstante  $k$  jeweils gleich 1; dann ist auch die Kreisfrequenz  $\omega_0^2 = k/m$  gleich 1, der Gain Faktor also  $-1$ . Dieser ist für den Block **Gain** zu spezifizieren. Anklicken des Blocks mit der rechten Maustaste öffnet ein Menü, in dem wir die Karte **Gain Parameters** auswählen können – als Shortcut hilft für diese Karte auch ein Doppelklick mit Links auf das Symbol. Der Gain selbst wird auf der in Abb. 2.14 gezeigten Karte **Main** in die entsprechende Zeile eingetragen.<sup>9</sup> Da in diesem Beispiel die betrachtete Variable eine skalare Größe ist, ist der Gain ebenfalls ein Skalar. Daher ist in der zweiten Zeile

<sup>9</sup>Alternativ kann auch eine Variable eingetragen werden, die im Workspace oder in einer Datei definiert werden muss, vgl. Abschn. 2.4.2.

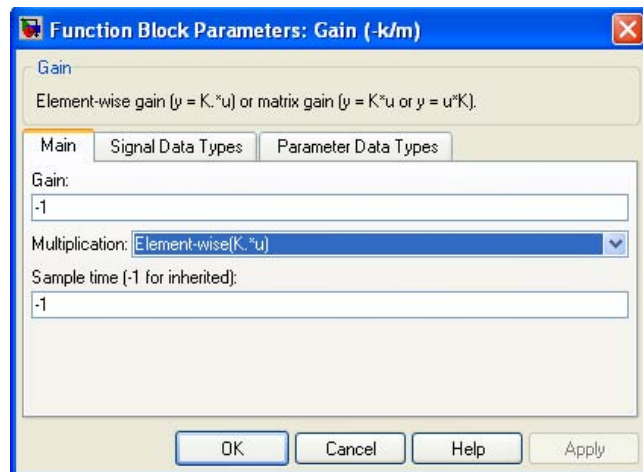


Abbildung 2.14: Einstellungen zum Funktionsblock `Gain`; hier für das Beispiel der freien Schwingung mit  $\omega_0^2 = -1$

die Spezifikation der Multiplikation als Elementweise oder den Regeln der Matrixmultiplikation gehorchend nicht erforderlich. Wenn es keine wirklich triftigen Gründe gibt, sollte in der letzten Zeile die Zeitauflösung nicht verändert werden sondern aus der Simulation bezogen werden: daher die  $-1$  stehen lassen.

§ 196 Lassen wir die Simulation so laufen, so passiert nichts: Ort und Geschwindigkeit sind identisch Null für alle Zeiten. Das ist nicht verwunderlich, da wir keine Anfangsbedingungen explizit spezifiziert haben. In diesem Fall startet der Integratorblock mit der Anfangsbedingung Null, d.h. wir betrachten die Schwingung eines Pendels, das sich im Anfangszustand in Ruhe in der Ruhelage befindet:  $v_0 = 0$  und  $x_0 = 0$ .

§ 197 Für die beiden Integratorblöcke müssen daher noch Einstellungen der Anfangsbedingungen vorgenommen werden. Abbildung 2.15 zeigt die zur Festlegung der `Integrator Parameters` benötigte Karte für den linken Integrationsblock, die die Geschwindigkeit liefert. Ebenso wie beim `Gain` werden wir in diesem Schritt alle rechentechnischen Optionen ignorieren und uns auf die inhaltlichen Angaben beschränken. Diese stecken in den ersten drei Zeilen. Davon ist die dritte die wichtigste: hier wird die Anfangsbedingung angegeben. Da wir ein numerisches Verfahren betrachten, kann nur ein bestimmtes Integral ausgeführt werden, d.h. die Aufgabe des Integrationsblocks lässt sich formal darstellen als

$$v_{i+1} = \int_{v_i}^{v_i + \Delta v} a \, dt = a \Delta t + v_i \quad \text{mit} \quad a = -\omega_0^2 x. \quad (2.1)$$

Die Anfangsbedingung ist also ein Simulationsparameter, nämlich die Geschwindigkeit aus dem voran gegangenen Zeitschritt. Daher ist in der zweiten Zeile die Initial Source Condition als internal zu vereinbaren. Lediglich im ersten Zeitschritt muss die Anfangsbedingung explizit vorgeben werden – oder, wie in der Fußnote zum `Gain`-Block diskutiert, über eine Variable eingeführt werden. In diesem Beispiel ist sie bei der Integration in  $v$  auf 1 gesetzt; bei der in  $s$  dagegen auf Null.<sup>10</sup> Das entspricht einer Masse, die sich zur Zeit  $t = 0$  im Ursprung befindet und sich mit der dimensionslosen Geschwindigkeit 1 in positiver  $x$ -Richtung bewegt – wie auch in Abb. 1.6 gezeigt.

§ 198 Jetzt fehlt noch die Darstellung des Ergebnisses. Soll der Ort die Zielgröße sein, so müssen wir das Signal hinter der zweiten Integration abgreifen (an die rechte Maustaste

<sup>10</sup>Da die Schwingungsgleichung eine DGL zweiter Ordnung ist, benötigen wir auch zwei Anfangsbedingungen. Als Default verwendet SIMULINK eine Null als Anfangsbedingung. Beim Federpendel sieht man daher sofort, ob die Angabe der Anfangsbedingungen vergessen wurde: war das der Fall, so ist  $x(t) = v(t) = 0$  und das Ergebnis trivial. Leider gibt nicht jede DGL diese Sicherheit.

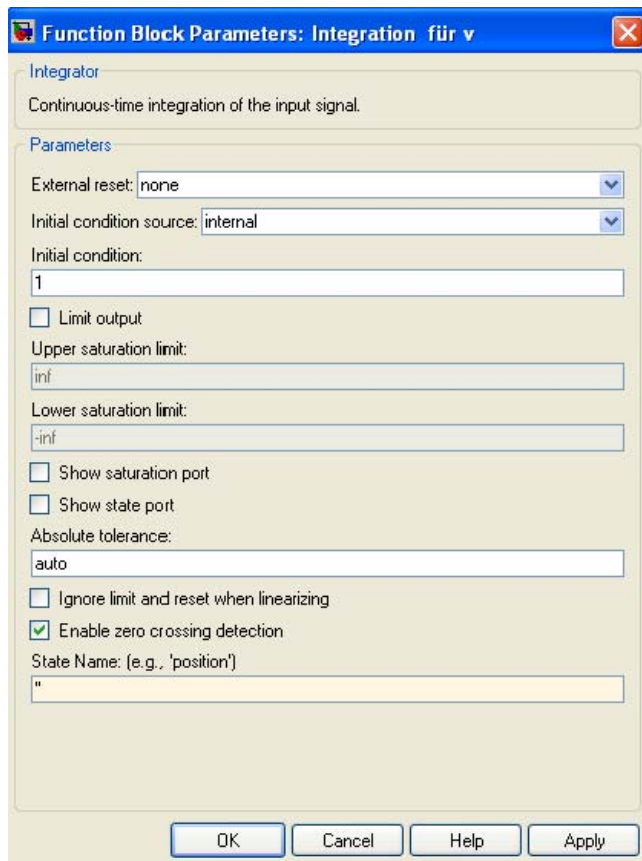


Abbildung 2.15: Einstellungen zum Funktionsblock *Integration*

denken, da Verzweigung!) und auf eine Senke legen. In Abb. 1.6 ist dies ein Scope, für das wir keine weiteren Einstellungen treffen müssen.

§ 199 Das Modell ist fertig und die Simulation kann im Prinzip gestartet werden. Vorher sollten wir allerdings entweder im entsprechenden Feld im SIMULINK-Fenster oder in der Karte zur Simulationskontrolle das Ende des Simulationsintervalls angeben (der Default Wert von 10 ist bei einem  $\omega_0$  von 1 ok). Die Simulation lässt sich mit `ctrl T` oder über den entsprechenden Unterpunkt des Menüs Simulation im SimuLink-Fenster starten. Die Ergebnisse werden durch Doppelklick mit Links auf das Scope in einem neuen Fenster angezeigt. Achsen und Skalierung dieser Darstellung lassen sich bearbeiten, auch lässt sich das Fenster verkleinern oder vergrößern.

§ 200 Mit der Default-Einstellung hat das Signal noch einiges an Ecken, d.h. eine höhere Zeitauflösung ist anzustreben. Diese lässt sich in der Simulationskontrolle einstellen, ein fester maximaler Zeitschritt von 0.05 liefert schon weniger kantige Ergebnisse. Alternativ kann man auch bei variabler Schrittweite eine größere Genauigkeit erzwingen. So lange wir kleine Modelle haben, werde ich mich auf die Variante einer festen maximalen ‘Angstschrittweite’ beschränken: das gibt eine gute Genauigkeit auf Kosten der Laufzeit. Die Steuerung von Genauigkeit und Zeitauflösung wird in Abschn. 2.6 näher eingehen.

§ 201 Um außer dem Ort auch die Geschwindigkeit darzustellen, können wir dieses Signal zwischen den beiden Integratoren abgreifen und entweder auf ein eigenes Scope geben oder, wie in Abb. 1.7 geschehen, über einen Multiplexer Mux (aus der Untergruppe **Signal Routing**) mit dem Ort zusammen auf das Scope führen. Der Multiplexer hat als Default zwei Eingänge, über die Karte **Mux Parameters** kann die Zahl der Eingänge verändert werden. Als Ausgang erhalten wir auf dem Scope eine Darstellung, die Ort und Geschwindigkeit in

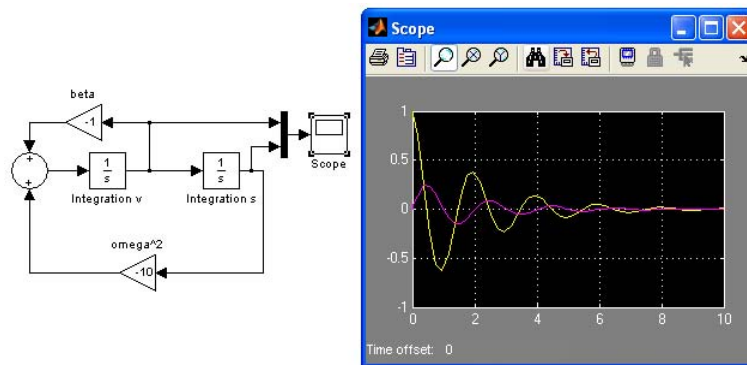


Abbildung 2.16: Ge-dämpfte Schwingung

Abhängigkeit von der Zeit enthält; dabei wird das erste Signal (oberer Eingang des Multiplexers) in gelb, das zweite in Magenta dargestellt.

§ 202 Die freie Schwingung lässt sich durch Hinzufügung einer von der Geschwindigkeit abhängigen dämpfenden Kraft  $-\beta v$  auf die gedämpfte Schwingung erweitern, vgl. Abb. 2.16. Dazu müssen wir nur das Signal Geschwindigkeit zwischen den beiden Integratoren abgreifen, mit Hilfe eines Blocks **Gain** mit  $-\beta/m$  multiplizieren und ebenso wie die aus der rücktreibenden Kraft resultierende Beschleunigung  $-\omega_0^2 x$  dem Integratorblock in  $v$  zuführen. Dazu benötigen wir einen Summationsblock, der die beiden Beschleunigungen vor ihrer Integration addiert. An der Integration ändert sich nichts, lediglich die Beschleunigung  $a$  in (2.1) ist jetzt  $a = -\omega_0^2 x - \beta v/m$ .

§ 203 Die Parameter des Summenblocks lassen sich über die Karte **Function Block Parameters: Sum** einstellen. Wichtig ist dabei nur die Zahl der Eingänge, in unserem Fall zwei, sowie deren Vorzeichen: bei  $+$  wird das Signal addiert, bei  $-$  subtrahiert. Das Hin- und Herschieben der einzelnen Eingänge müssen Sie durch Probieren hinkriegen.

§ 204 Für die erzwungene Schwingung muss anschließend, wie in Abb. 1.8 nur noch die externe antreibende Kraft zugefügt werden. Dieses erfolgt ebenfalls am Summenknoten. Dazu wird mit Hilfe der Karte **Function Block Parameters: Sum** ein weiterer Eingang erzeugt und auf diesen das Signal eines Funktionsgenerators, in diesem Fall **Sine Wave** für einen Sinus, gegeben. In der entsprechenden Karte **Source Block Parameters: Sin Wave** lassen sich Amplitude, Frequenz und Phase des Signals einstellen.

### 2.3.2 Test der Simulation

§ 205 Eine Simulation ist ein numerisches Verfahren. Wie in jedem derartigen Verfahren wird statt der kontinuierlichen Zeitachse eine Folge endlicher Zeitintervalle betrachtet: Differentiale und Differentialquotienten werden durch Differenzen und Differenzenquotienten ersetzt. Eine derartige Diskretisierung ist, wie bereits im Zusammenhang mit Abb. 1.4 angedeutet, stets mit einem Fehler behaftet. Dieser muss abgeschätzt und hinreichend klein gehalten werden.

§ 206 Bei einem manuell erstellten numerischen Verfahren mit konstanter Schrittweite lassen sich diese Diskretisierungsfehler abschätzen, siehe auch Abschn. 10.2.2. SIMULINK verwendet man (genauso wie MATLAB) unter anderem, weil man sich über diese Details keine Gedanken machen möchte und dem Solver vertraut. Da SIMULINK bei der Auswahl der Solver eine Angabe von Schrittweiten und Genauigkeiten erlaubt, lässt sich die voreingestellte Genauigkeit verändern – in diesem Fall ist eine Überprüfung der Lösung sicherlich sinnvoll. In unserem einfachen Beispiel kann diese Überprüfung durch Vergleich mit der analytischen Lösung erfolgen. Normalerweise ist das nicht möglich, da außer zu didaktischen Zwecken im Falle einer existierenden analytischen Lösung diese statt einer numerischen Lösung verwendet wird.

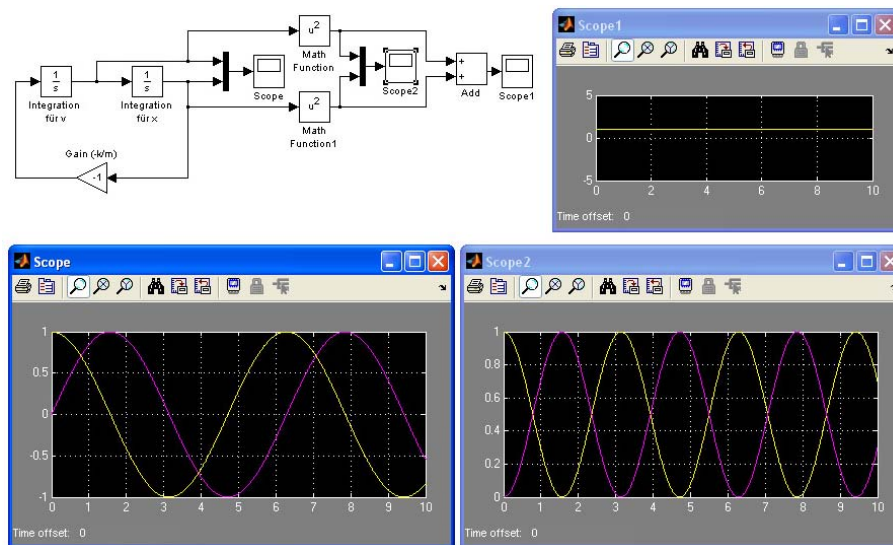


Abbildung 2.17: Freie Schwingung in SIMULINK mit zusätzlichen Darstellungen zur Überprüfung der Lösung: Scope 2 gibt jeweils die kinetische und die potentielle Energie, Scope 1 die Gesamtenergie

§ 207 Wie allgemein in der Modellierung bieten sich Plausibilitätsbetrachtungen, Abschätzungen für Extremfälle und die Erhaltungsgrößen für einen Test an, also z.B. Energie, Masse oder Impuls. Bei der freien Schwingung aus Abb. 1.7 ist die mechanische Energie die Erhaltungsgröße. Diese lässt sich in der Simulation als  $E_{\text{mech}}(t) = \frac{1}{2}mv^2 + \frac{1}{2}kx^2$  aus der Auslenkung und der Geschwindigkeit bestimmen. Dazu greifen wir beide Größen ab, vgl. Abb. 2.17. Um die kinetische und die potentielle Energie zu bestimmen, müssen diese Größen quadriert sowie mit dem Faktor  $\frac{1}{2}$  und der Masse  $m$  bzw. Federkonstante  $k$  multipliziert werden. Da  $m$  und  $k$  jeweils Eins sind, können wir uns die Multiplikation sparen; den Faktor  $\frac{1}{2}$  sparen wir uns ebenfalls: er ist konstant und wir wollen nur wissen, ob die Energie ebenfalls konstant ist. Das quadrieren erfolgt mit dem Block **Math Function** aus der Untergruppe **Math Operations**, in dem wir verschiedene mathematische Funktionen auswählen können. Alternativ lässt sich auch der Block **Fcn** aus der Untergruppe **User-Defined Functions** verwenden. In ihm lassen sich beliebige Funktionen entsprechend den Regeln von MATLAB definieren.

§ 208 Beide Signale werden über einen Multiplexer auf Scope 2 gegeben, das uns damit den zeitlichen Verlauf von kinetischer und potentieller Energie gibt. Außerdem werden beide Signale auf einen Additionsblock gegeben. Dessen Ausgang, also die Gesamtenergie, wird auf Scope 1 gegeben: die Gesamtenergie bleibt konstant – auch wenn das hier betrachtete, nicht einmal zwei Schwingungsperioden umfassende Zeitfenster vielleicht etwas kurz ist.

§ 209 Alternativ hätten wir Ort und Geschwindigkeit auch über einen Multiplexer in eine vektorielle Größe zusammen fassen können. Dann hätten wir nur einen Block **Math Function** benötigt, um den Vektor komponentenweise zu quadrieren. Das Ergebnis hätten wir auf einem Scope darstellen können; außerdem hätten wir in einem weiteren Block die Elemente des Vektors addieren und ebenfalls darstellen können.

§ 210 Und die kürzeste Variante würde hinter dem Multiplexer das Skalarprodukt des Vektors mit sich selbst bilden und das Ergebnis dann als Gesamtenergie darstellen.

### 2.3.3 Der Export der Daten

§ 211 Der Einbau des Tests in die eigentliche Simulation ist eher störend, da der Test nicht Bestandteil der späteren lauffähigen Simulation sein sollte sondern nur in der Designphase

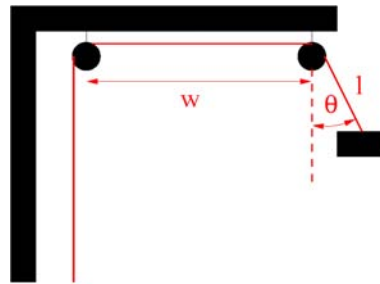


Abbildung 2.18: Der Kran als Fadenpendel

benötigt wird. Daher ist ein separater Test erstrebenswert. Dazu können wir die für den Test benötigten Daten in den Workspace oder in eine Datei exportieren und dort mit den üblichen MATLAB-Tools bearbeiten.

§ 212 Beide Varianten benötigen eine Datensenke, d.h. die entsprechenden SIMULINK-Blöcke werden in der Untergruppe `Sinks` mit den in Abb. 2.10 gezeigten Elemente zu finden sein. Diese Blöcke werden wie das `Scope` verwendet; sie können auch parallel zum `Scope` verwendet werden.

§ 213 Mit Hilfe des Blocks `ToWorkspace` wird ein einzelnes strukturiertes Element mit dem im Block spezifizierten Namen (Default: `simout`) erzeugt. Dieser kann mit Hilfe der Karte `ToWorkspace Parameters` neu gesetzt werden. Auch können die Zahl der Nachkommastellen, die Schrittweite der Ausgabe sowie die Art der Ausgabe (Array statt strukturiertes Element bzw. strukturiertes Element plus Zeit) angegeben werden. Bei der Ausgabe als Array können die Daten direkt verwendet werden, was insbesondere für einen schnellen Test hilfreich ist. Bei der Ausgabe als strukturiertes Element muss man dieses zur Weiterverarbeitung zwar erst mit dem Array-Editor öffnen.<sup>11</sup> Dafür enthält das Element jedoch zusätzliche Informationen, was für eine auf längere Sicht angelegte Speicherung von Ergebnissen von Vorteil ist.

§ 214 Die Ausgabe in eine Datei mit Hilfe des Blocks `ToFile` erzeugt stets einen `*.mat`-File mit einem Array, wobei die erste Zeile die Zeit enthält, die folgenden Zeilen jeweils einen der Simulationsparameter. Die sonstige Ansteuerung (inklusive der Karten) entspricht der im Block `ToWorkspace`.

§ 215 Selbstverständlich sind beide Ausgabeoptionen nicht nur für Tests hilfreich sondern geben die Simulationsergebnisse allgemein zur weiteren Verarbeitung heraus; wir werden in Abschn. 2.5 näher darauf eingehen und bei allen komplexeren Modellen in den folgenden Kapiteln davon Gebrauch machen. Außerdem ist der externe Test zumindest für den Umgang mit MATLAB gewohnte User deutlich einfacher, da auf den vollständigen Funktionsumfang von MATLAB zurück gegriffen werden kann. So ließe sich z.B. beim Federpendel neben der Energieerhaltung auch auf eine Drift in der Schwingungsfrequenz prüfen.

### 2.3.4 Technische Anwendung: schwingender Container am Kran

§ 216 Die DGL für das Fadenpendel unterscheidet sich für kleine Auslenkungen  $\vartheta$  nicht von der des Feder-Masse Systems: die rücktreibende Kraft ist wieder linear von der Auslenkung abhängig, da für kleine Winkel der Sinus des Winkels gleich dem Winkel ist.

§ 217 Hunt et al. [53] geben ein auch physikalisch anspruchsvolles Beispiel einer technischen Anwendung für ein Fadenpendel veränderlicher Länge. Ein Kran zum Verladen von Containern kommt als technisches Beispiel diesem physikalischen Problem recht nahe. Betrachten wir dazu den Kran wie in Abb. 2.18 dargestellt. Die beiden Umlenkrollen am Kransausleger

<sup>11</sup>Das ist nur eingeschränkt korrekt. Die Bezeichnung der Teile des strukturierten Elements folgt natürlich einem bestimmten Schema, so dass wir das Array der Variablen in `simout.signals.values` (bzw. bei anderem Namen als `simout` unter `NAME.signals.value`) finden und direkt aus dem Workspace ansprechen können.



sowie der Container seien infinitesimal klein. Die linke Rolle ist fest und bildet den Ursprung des Koordinatensystems; die rechte Rolle kann durch den Kranführer bewegt werden, so dass sich der Abstand  $w$  zwischen den beiden Umlenkrollen verändern lässt. Ebenso lässt sich die Länge  $l$  des Seils zwischen rechter Rolle und Container verändern. Außerdem kann der Container um den Winkel  $\vartheta$  gegenüber der Vertikalen ausgelenkt werden. Der momentane Aufenthaltsort des Containers relativ zur festen linken Rolle ist also beschrieben durch die drei Parameter  $w$ ,  $l$  und  $\vartheta$  mit

$$\vec{r} = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} w + l \sin \vartheta \\ -l \cos \vartheta \end{pmatrix}.$$

§ 218 Die zu untersuchende Bewegung ist das Anheben des Containers am Anfangsort  $\vec{r}_0 = (w_0, -l_0)$  und sein Absetzen an einem anderen Ort  $\vec{r}_1 = (w_1, -l_0)$ . Unser Interesse gilt dabei den vom Container ausgeführten Schwingungen – der Container soll transportiert werden, nicht aber eine Zweitkarriere als Abrissbirne beginnen. Der Luftwiderstand sei vernachlässigbar, die einzigen wirkenden Kräfte sind die Gravitation und die Spannung im Seil.

**Verständnisfrage 20** Sie haben als Vertreter der Berufsgenossenschaft der Hafentarbeiter diese Simulation in Auftrag gegeben und erhalten jetzt das unter Vernachlässigung der Reibung gewonnene Ergebnis. Ist das für Sie eine problematische Einschränkung? Würde der Hafentreiber bezüglich dieser Einschränkung zu dem gleichen Schluss kommen wie Sie?

§ 219 Da es sich bei dem Kran um ein älteres Modell handelt, müssen die verschiedenen Bewegungen nacheinander ausgeführt werden: (1) der Kranführer kann den Container durch Einholen des Seils senkrecht anheben ( $l$  wird verändert,  $w$  ist konstant), (2) die horizontale Bewegung der rechten Rolle verändert  $w$  unter Beibehaltung der Länge  $l$ , und (3) der Container wird bei festem  $w$  vertikal abgesetzt ( $l$  wird verändert).

§ 220 Solange keine äußeren Kräfte (z.B. böiger Wind) wirken, ist Schritt (1) recht langweilig: die Bewegung des Containers erfolgt ausschließlich in der Vertikalen; da keine Auslenkung vorhanden ist, tritt auch keine Schwingung auf. Auch Schritt (3) ist nicht besonders interessant: zwar kann der Container am Anfang von Schritt (3) noch schwingen, erfolgt die Veränderung der Seillänge hinreichend langsam, so lässt sich das Problem durch eine adiabatische Invariante beschreiben. Wirklich interessant ist nur der zweite Schritt: der horizontale Transport eines Fadenpendels.<sup>12</sup>

**Wiederholungsfrage 5** Beschreiben Sie kurz, was unter einer adiabatischen Invarianten einer Bewegung zu verstehen ist. Was ist bei einer adiabatischen Invarianten eigentlich invariant und unter welchen Bedingungen? Nennen Sie weitere Beispiele für adiabatische Invarianten.

§ 221 Zur Beschreibung der Bewegung greifen wir auf die Lagrange'sche Mechanik zurück. Die kinetische und potentielle Energie des Containers der Masse  $m$  sind

$$\begin{aligned} K &= \frac{m}{2} \dot{\vec{r}}^2 = \frac{m}{2} (\dot{x}^2 + \dot{y}^2) = \frac{m}{2} \left( (\dot{w} + l \dot{\vartheta} \cos \vartheta)^2 + (l \dot{\vartheta} \sin \vartheta)^2 \right) \\ &= \frac{m}{2} (\dot{w}^2 + l^2 \dot{\vartheta}^2 + 2l \dot{w} \dot{\vartheta} \cos \vartheta) \\ V &= mgy = -mgl \cos \vartheta. \end{aligned}$$

Aus der Lagrange Funktion  $L = K - V$  ergibt sich die Bewegungsgleichung gemäß

$$\frac{\partial L}{\partial \vartheta} = \frac{d}{dt} \frac{\partial L}{\partial \dot{\vartheta}}$$

nach einigem Kürzen und zusammen fassen zu

$$\ddot{\vartheta} + \frac{\cos \vartheta}{l} \ddot{w} + \frac{g}{l} \sin \vartheta = 0 \quad \text{oder} \quad \ddot{\vartheta} = -\frac{\cos \vartheta}{l} \ddot{w} - \frac{g}{l} \sin \vartheta.$$

Im Gegensatz zur gewöhnlichen Schwingungsgleichung ergibt sich hier ein Zusatzterm, der die horizontale Beschleunigung  $\ddot{w}$  des Krans enthält.

<sup>12</sup>Ein verwandtes, um  $\pi$  gedrehtes Problem ist das invertierte Pendel auf einem Wagen; dieses ist in der SIMULINK-Hilfe vorgestellt.

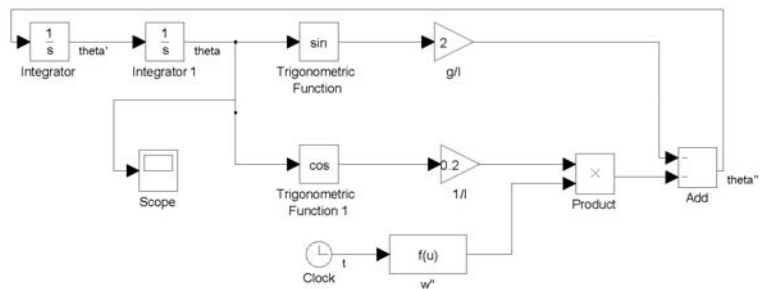


Abbildung 2.19:  
SIMULINK Modell  
zum Kran aus Abb. 2.18

**Wiederholungsfrage 6** Falls es Ihnen nicht mehr ganz geläufig ist, wäre dies eine gute Stelle, noch einmal schnell über die Grundideen der Lagrange'schen Mechanik nachzudenken.

§ 222 Für die Realisierung des Problems in SIMULINK ist nur dieser Zusatzterm mit der horizontalen Beschleunigung neu, den Rest können wir vom Federpendel ausleihen. Wie bei einer externen Kraft sollte dieser Beschleunigungsterm variabel gestaltet sein, so dass sich Lösungen  $\vartheta(t)$  für verschiedene Operationsweisen des Krans bestimmen lassen.

§ 223 Eine mögliche Realisierung ist in Abb. 2.19 gegeben: die beiden Integratoren benötigen wir, da wir eine DGL 2<sup>ter</sup> Ordnung betrachten. Das Signal  $\dot{\vartheta}$  zwischen den Integratoren wird nicht abgegriffen, da die DGL keinen Term proportional  $\dot{\vartheta}$ , wie z.B. Reibung, enthält. Der obere Gain Block ist der Faktor vor der rüctreibenden Kraft: da wir die Einschränkung auf kleine Winkel fallen lassen, muss erst der Gain auf den Sinus und nicht auf  $\vartheta$  direkt angewendet werden, wie bereits beim Fadenpendel in Abschn. 1.1.5 verwendet. Das Ergebnis ist die Beschleunigung auf Grund der rüctreibenden Kraft und wird für den nächsten Zeitschritt auf den Integrator gegeben – nicht ohne vorher die zweite Kraft zu addieren. Letztere ist das Produkt aus der externen Beschleunigung, beschrieben durch eine externe Funktion  $f(u)$ , und dem Kosinus des Winkels  $\vartheta$  (multipliziert mit einem Gain). Die Clock erzeugt den Zeittakt zur Bestimmung der externen Funktion  $f(u)$ . Ihr Ergebnis entspricht dem z.B. mit `linspace` erzeugten Abszissen-Vektor in MATLAB mit dem Unterschied, dass das Spacing nicht linear sein muss sondern durch die momentane Schrittweite des Solvers jeweils wieder neu vorgegeben wird.

§ 224 Nach dem Prinzip kommen die Zahlen:  $g = 10 \text{ m/s}^2$ ,  $\Delta w = w_1 - w_0 = 10 \text{ m}$ ,  $\Delta l = 5 \text{ m}$ . Und da der Kranführer eine hinreichende Stückzahl bewältigen muss, billigen wir ihm eine Zeit von  $t_{\text{end}} = 200 \text{ s}$  pro Container zu. Damit erhalten wir die Gain Faktoren wie in Abb. 2.19. Die einzige Herausforderung ist die Eingabe von  $w(t)$ : dies sollte eine stetige Funktion sein mit  $w(0) = 0$  und  $w(t_{\text{end}}) = w_1$ . Aus physikalischen Gründen wäre eine Funktion, die Punkt-symmetrisch um  $(t_{\text{end}}/2, w_1/2)$  keine schlechte Idee. Außerdem sollte die erste Ableitung zu Beginn und Ende der Bewegung verschwinden:  $\dot{w}(0) = \dot{w}(t_{\text{end}}) = 0$ . Diese Einschränkungen ließen sich mit einer geeignet verschobenen Winkelfunktion erfüllen. Fordern wir zusätzlich noch ein Verschwinden der Beschleunigung am Anfang und Ende der Bewegung,  $\ddot{w}(0) = \ddot{w}(t_{\text{end}}) = 0$ , so kann man ein geeignet verschobenes Polynom verwenden. Hunt et al. [53] geben als Beispiel

$$w(t) = -\frac{35}{8} + \frac{8}{32}t + \frac{(t-100)^3}{160000} + \frac{(t-100)^5}{16000000000}$$

und damit

$$\ddot{w}(t) = -\frac{3t}{80000} + \frac{3}{800} + \frac{3(t-100)^3}{800000000}.$$

§ 225 Die aus dieser Bewegung resultierende Auslenkung des Containers ist im linken Teil von Abb. 2.20 gegeben: eine maximale Auslenkung von  $2 \cdot 10^{-4}$  ergibt selbst bei einer Seillänge von 5 m nur eine minimale Versetzung des Containers; die Sicherheit für Mensch und Materie in der Umgebung ist gewährleistet.

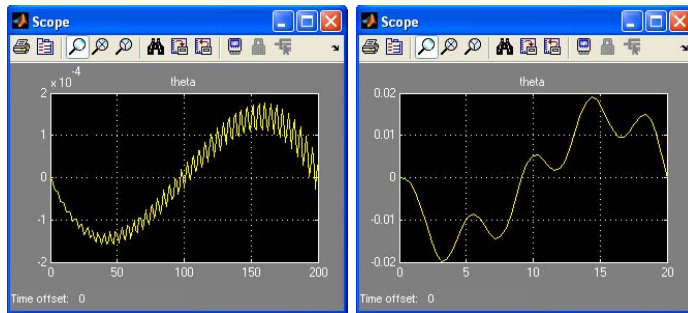


Abbildung 2.20: Auslenkung des Containers aus der vertikalen für eine langsame (*links*) und eine schnelle (*rechts*) Bewegung

§ **226** In einem modernen Hafenbetrieb sind 200 s Vershubzeit für einen Container aber viel zu lang – Reduktion um einen Faktor 10 heißt zehnmal so viele Container (und damit mehr Geld). Zum Vergleich behalten wir den Verlauf von  $w$ , allerdings muss die Zeitachse entsprechend skaliert werden. Wir übernehmen erstmal wieder von [53]:

$$\ddot{w} = \frac{3t}{40} + \frac{3t^3}{8000} - \frac{9t^2}{800} .$$

Mit diesen Parametern (und einer auf 20 s reduzierten Simulationszeit) ergibt sich das rechte Teilbild in Abb. 2.20: die Auslenkung liegt im Bereich von 0.02, was bei 5 m Seillänge immerhin schon 10 cm entspricht: nicht genug für eine Abrissbirne aber für einen unschönen blauen Fleck oder eine Delle reicht das.

**Zwischenrechnung 10** Spielen Sie mit der Funktion  $w(t)$ , insbesondere auch unter physikalisch sinnvollen Randbedingungen – bitte daran denken, dass im Funktionsblock nicht die Funktion sondern die zweite Ableitung eingesetzt wird!

**Zwischenrechnung 11** Und jetzt packen Sie dem armen Kranführer noch einen böigen Wind als Störung hinzu: vernachlässigen Sie den mittleren Wind und drücken Sie die die fluktuierende Komponente als eine stochastische Beschleunigung (in was?) aus.

### 2.3.5 Etwas komplizierter: zwei gekoppelte Federpendel

§ **227** Dieser Abschnitt bildet die Ergänzung zum bereits in Abschn. 1.2 und insbesondere in Abb. 1.20 dargestellte Modell zur Beschreibung des Doppelpendels aus zwei Massen und drei Federn. Mathematisch betrachten wir ein (zugegebenermaßen einfaches) System gekoppelter gewöhnlicher Differentialgleichungen.

#### Die Simulation

§ **228** Wie der Vergleich von Abb. 1.20 und Abb. 1.6 zeigt, unterscheiden sich die SIMULINK Modelle für die freie Schwingung einer Masse und das Doppelpendel in ihrer Struktur überhaupt nicht. Der einzige Unterschied wird in der Ausgabe auf dem Scope sichtbar: statt einer Kurve finden wir zwei Kurven im Weg-Zeit-Diagramm – für jede Masse eine. Da wir nur ein Signal haben, behandelt SIMULINK das Problem offenbar genauso, wie wir es auch in der analytischen Lösung getan haben: die Orte der Massen (und ebenso die Geschwindigkeiten) werden jeweils in einem Vektor zusammen gefasst. Konsequenterweise ist der Gain dann kein Skalar mehr sondern die bereits bei der analytischen Lösung verwendete Matrix.

§ **229** Der Übergang zu Vektoren bzw. Matrizen erfolgt auf den zu den jeweiligen Blöcken gehörenden Karten: in der Karte für den Gain, vgl. Abb. 2.14, wird in der ersten Zeile statt des Skalars 1 die Matrix in der MATLAB-typischen Form  $[-2, 1; 1, -2]$  eingetragen. In der zweiten Zeile muss jetzt darauf geachtet werden, dass Multiplikation entsprechend den Regeln der Matrizenrechnung verwendet wird.

§ **230** Wichtiger noch ist der korrekte Eintrag auf den Integratorkarten, vgl. Abb. 2.15. Hier erkennt SIMULINK aus der Tatsache, dass in der dritten Zeile als Anfangswert statt eines Skalars ein Vektor eingetragen ist, dass es sich bei den Signalen nicht um skalare sondern eben vektorielle Größen handelt mit der Länge des Vektors als Dimension des Problems.

§ **231** Am Scope ist ein Übergang von skalaren zu vektoriellen Größen nicht erforderlich: ist das Eingangssignal am Scope vektoriell, so werden automatisch die einzelnen Komponenten dargestellt.<sup>13</sup>

### Gibt es Alternativen?

§ **232** Für das einzelne Pendel haben wir bereits festgestellt, dass die Simulation mit Hilfe einer Übertragungsfunktion nur dann Sinn macht, wenn es eine antreibende Kraft gibt; die Bewegungsgleichungen 1.9 sind also um diese zu ergänzen:

$$\ddot{\vec{x}} = \begin{pmatrix} -2\omega_0^2 & \omega_0^2 \\ \omega_0^2 & -2\omega_0^2 \end{pmatrix} \vec{x} + \vec{f}_{\text{ext}} . \quad (2.2)$$

Laplace Transformation liefert

$$s^2 \vec{X} = \begin{pmatrix} -2\omega_0^2 & \omega_0^2 \\ \omega_0^2 & -2\omega_0^2 \end{pmatrix} \vec{X} + \mathcal{L}(\vec{f}_{\text{ext}})$$

oder umgeformt

$$\left( s^2 \mathbf{E} - \begin{pmatrix} -2\omega_0^2 & \omega_0^2 \\ \omega_0^2 & -2\omega_0^2 \end{pmatrix} \right) \vec{X} = \mathcal{L}(\vec{f}_{\text{ext}}) . \quad (2.3)$$

Der Faktor vor dem  $\vec{X}$  erinnert stark an (die Inverse) der Resolvente (4.11); allerdings tritt statt des  $s\mathbf{E}$  in (4.11) ein  $s^2\mathbf{E}$  auf, da wir von einer DGL zweiter Ordnung ausgehen.

**Zwischenrechnung 12** Überzeugen Sie sich, dass wir diese Gleichung auch erhalten hätten, wenn wir das Gleichungssystem transformiert und dann in Matrix-Schreibweise zusammengefasst hätten.

§ **233** Wir können (2.3) zwar nach  $\vec{X}$  auflösen – dann erhalten wir auch die Resolvente. Wir können aber keine Übertragungsfunktion bestimmen, da letztere nur skalar definiert ist. SIMULINK betrachtet die Übertragungsfunktion zwar als Single-Input Multiple-Output Funktion: das bedeutet jedoch nur, dass der Ausgang vektoriell sein kann; der Eingang ist immer nur eine skalare Größe.

§ **234** Bei der erzwungenen Schwingung eines einzelnen Pendels haben wir in Abschn. 1.1.4 mit dem Zustandsvektor gearbeitet. Da der Block `State Space` nicht auf Single-Input, Single-Output Systeme beschränkt ist sondern auch Multiple-Input, Multiple-Output Systeme beschreiben kann, können wir die gekoppelten Pendel auf diese Weise modellieren. Die Bewegungsgleichungen (2.2) sind zwei Gleichungen für zwei Inputs (die jeweils direkt auf eine Masse wirkende Kraft) sowie zwei Outputs (die Orte der Massen). Für jede Masse haben wir zwei Zustandsvariablen, Ort und Geschwindigkeit, so dass das System insgesamt durch vier Zustandsvariablen beschrieben ist:

$$z_1 = x_1 \quad z_2 = \dot{x}_1 \quad z_3 = x_2 \quad \text{und} \quad z_4 = \dot{x}_2 .$$

Darin sind  $z_1$  und  $z_3$  auch gleichzeitig die Ausgangsgrößen.

**Zwischenrechnung 13** Führen Sie das Modell zu Ende aus. Verändern Sie das Modell ferner so, dass (a) die Massen und Federkonstanten nicht identisch sein müssen und (b) zusätzlich Reibung auftreten kann.

<sup>13</sup>Auch der Multiplexer `Mux` erzeugt aus seinen zwei (oder mehr) skalaren Eingangsgrößen letztendlich nur einen Vektor. Und diesen haben wir in verschiedenen Simulationen bereits auf das Scope gegeben.

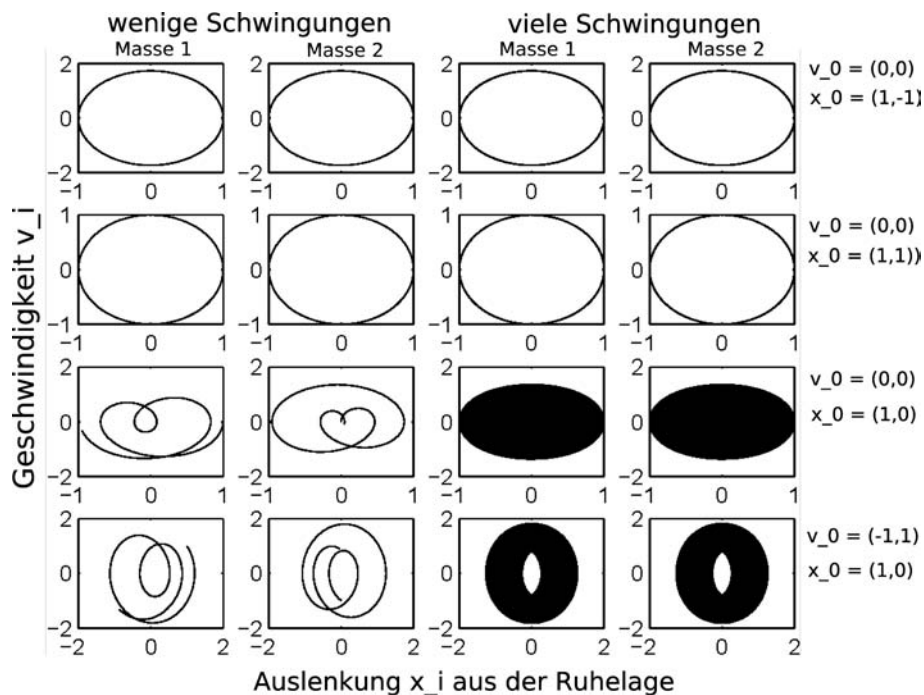


Abbildung 2.21: Darstellung der Bewegung des Doppelpendels im Phasenraum für die beiden Massen für verschiedene Anfangsbedingungen (rechte Zeile); *links* jeweils ein Ausschnitt aus dem Simulationszeitraum, *rechts* der vollständige Zeitraum

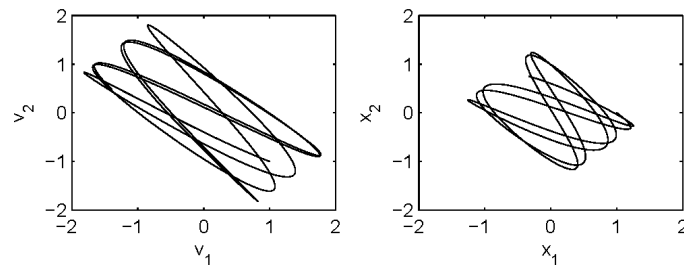
### Einschub: Ist die Bewegung periodisch oder nicht?

§ 235 Nach MATLAB noch eine inhaltliche Anmerkung: weder die Animation noch die verschiedenen Weg- oder Geschwindigkeits-Zeit-Diagramme verhelfen uns auf einfache Weise zu einem tieferen Verständnis der Bewegung: ist dieses einfache Problem noch periodisch oder bereits chaotisch? Zur genaueren Untersuchung lassen wir die Simulation über einen längeren Zeitraum (einige hundert Schwingungsperioden) laufen und exportieren die Daten in ein mat-File oder in den Workspace. Die einfachste Form Untersuchung ist die Darstellung der Bewegung im Phasenraum. Dazu tragen wir, jeweils für die Massen getrennt, die Geschwindigkeit  $v$  gegen den Ort  $x$  auf. In Abb. 2.21 ist dies für nur wenige Schwingungen (links) bzw. eine größere Zahl von Schwingungen (rechts) und unterschiedliche Anfangsbedingungen gezeigt.

§ 236 Die ersten beiden Zeilen in Abb. 2.21 zeigen symmetrische Anfangsbedingungen. Die Anfangsgeschwindigkeit  $\vec{v}_0$  ist in beiden Fällen Null, beide Pendel sind ausgelenkt. In der oberen Zeile ist das linke Pendel nach rechts, das rechte nach links ausgelenkt, d.h. die beiden Massen schwingen gegeneinander. In der zweiten Zeile sind beide nach rechts ausgelenkt, sie schwingen im Gleichtakt. Die Bahnen sind im Phasenraum geschlossen, d.h. die Bewegung ist periodisch; aus der analytischen Lösung bzw. bei geeigneter Darstellung im Phasenraum ergeben sich als Frequenzen  $\sqrt{3}\omega_0$  bzw.  $\omega_0$ . Die Bahnen dieser beiden Eigenmoden unterscheiden sich in der Skalierung: bei der gegensinnigen Schwingung ist die Geschwindigkeit größer, da die anfängliche rücktreibende Kraft größer ist. Oder formuliert über die Energie: die Kompression bzw. Dehnung der koppelnden Feder entspricht einer zusätzlichen potentiellen Energie.

§ 237 Bei den in den beiden unteren Zeilen gezeigten asymmetrischen Anfangsbedingungen dagegen ist eine Periodizität nicht offensichtlich. In der oberen Zeile ist die linke Masse nach rechts ausgelenkt, die rechte befindet sich in der Ruhelage und die Anfangsgeschwindigkeit

Abbildung 2.22: Geschwindigkeiten (links) bzw. Orte (rechts) der beiden Pendelmassen gegeneinander aufgetragen ergeben im allgemeinen Fall nicht geschlossene Lissajous-Figuren



verschwindet. In der unteren Zeile wurde die gleiche Auslenkung verwendet, allerdings bewegt sich die linke Masse zusätzlich nach links, die rechte nach rechts. In beiden Fällen bilden sich kompliziertere Bahnen im Phasenraum, deren Trajektorien nach einiger Zeit einen gewissen Bereich des Phasenraums ausfüllen: im Rahmen dieser Darstellung können wir keine Hinweise auf Periodizität erkennen.

**Verständnisfrage 21** Woher kommt in der unteren Zeile von Abb. 2.21 das ‘Auge’ in der Mitte? Schließt es sich bei längeren Simulationen?

§ 238 Im Prinzip könnte die mangelnde Periodizität durch numerische Ungenauigkeiten bestimmt sein – auch wenn die rechten beiden Teilbilder der oberen Zeilen diesen Schluss nicht nahe legen. Betrachten wir nochmals die analytische Lösung

$$\vec{z}(t) = \vec{A}_1 e^{i\omega_0 t} + \vec{A}_2 e^{-i\omega_0 t} + \vec{A}_3 e^{i\sqrt{3}\omega_0 t} + \vec{A}_4 e^{-i\sqrt{3}\omega_0 t}$$

bzw. ihren Realteil

$$\vec{x}(t) = \vec{a}_1 \cos(\omega_0 t) + \vec{a}_2 \sin(\omega_0 t) + \vec{a}_3 \cos(\sqrt{3}\omega_0 t) + \vec{a}_4 \sin(\sqrt{3}\omega_0 t).$$

Daraus ergibt sich für die Geschwindigkeit

$$\vec{v}(t) = \omega_0 \left( \vec{a}_2 \cos(\omega_0 t) - \vec{a}_1 \sin(\omega_0 t) + \sqrt{3} \left( \vec{a}_4 \cos(\sqrt{3}\omega_0 t) - \vec{a}_3 \sin(\sqrt{3}\omega_0 t) \right) \right).$$

Eine geschlossene Darstellung der Bahn im Phasenraum ist nicht möglich, da wir keine der beiden Gleichungen nach dem Parameter  $t$  auflösen und in die andere Gleichung einsetzen können.

§ 239 Für allgemeine Randbedingungen besteht jede einzelne Lösung aus einer Superposition zweier harmonischer Funktionen mit einem nicht-rationalen Frequenzverhältnis. Damit ist die Bahn zwar nicht chaotisch, aber auch nicht wiederkehrend.

**Wiederholungsfrage 7** Rekapitulieren Sie aus der Theoretischen Mechanik die Möglichkeiten, chaotische Bahnen zu beschreiben bzw. zu charakterisieren. Können wir mit solchen Verfahren noch etwas über unsere gekoppelten Pendel lernen?

§ 240 Die fehlende Geschlossenheit der Kurven im Phasenraum findet sich auch an anderer Stelle wieder. Trägt man die Orte (oder die Geschwindigkeiten) der beiden Massen gegeneinander auf, so ergeben sich nicht geschlossene Lissajous-Figuren wie in Abb. 2.22 angedeutet. Letztere sind allgemein definiert als die Überlagerung zweier harmonischer Schwingungen wobei die Zeit als Parameter einer Parameterdarstellung verwendet wird. Lissajous-Figuren sind periodisch (und damit geschlossen), wenn das Verhältnis der Perioden der beiden Schwingungen rational ist – was in unserem Fall durch das Auftreten der  $\sqrt{3}$  außer bei den Eigenmoden nicht der Fall sein wird. Da die beiden Massen die gleiche relative Lage nicht wieder einnehmen, wird auch in dieser Darstellung die mangelnde Periodizität der Bewegung deutlich.

**Verständnisfrage 22** Sie sollen Lissajous-Figuren vorführen. Überlegen Sie sich einfach zu realisierende Beispiele und/oder rufen Sie sich ins Gedächtnis zurück, wo Ihnen Lissajous-Figuren bereits begegnet sind.

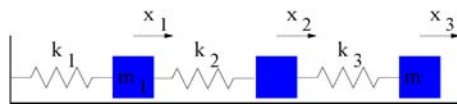


Abbildung 2.23: Geometrie für gekoppelte Federpendel zur Darstellung mit Hilfe von Untersystemen

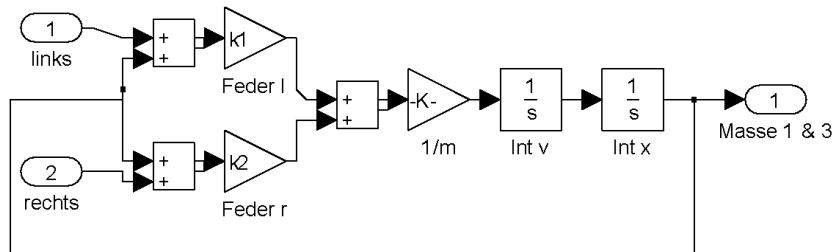


Abbildung 2.24: Subsystem für eine einzelne Masse der gekoppelten Pendel aus Abb. 2.23; die Ovale bilden die Übergabepunkte zu den anderen Blöcken der Simulation

## 2.4 (Maskierte) Untersysteme

§ 241 Ein einzelnes Federpendel bildet ein System und lässt sich als solches mit Hilfe einzelner Blöcke oder einer Übertragungsfunktion simulieren. Für gekoppelte Federpendel kann man daher abstrakt die in Abschn. 2.3.5 gezeigte vektorielle Darstellung verwenden oder alternativ jedes Pendel als ein Untersystem betrachten. Diese müssen dann geeignet verknüpft werden, um die Kopplung der Pendel korrekt zu beschreiben. Für gekoppelte Pendel ist natürlich die vektorielle Darstellung viel einfacher als die über Untersysteme – wobei letztere auch bei den gekoppelten Pendeln einfach ist, da man nur ein Untersystem wirklich erzeugen muss, die anderen können einfach kopiert werden.

§ 242 Um eine hinreichend komplexe aber immer noch überschaubare Geometrie zu erhalten, betrachten wir als Beispiel für Untersysteme drei gekoppelte Massen, von denen die linke über eine Feder mit der linken Wand verbunden ist, die rechte aber nur mit der mittleren Masse über eine Feder verbunden ist und nach rechts frei schwingen kann (vgl. Abb. 2.23). Unser ‘Federseil’ hat damit ein festes und ein loses Ende.

### 2.4.1 Das Untersystem

§ 243 Da jede Masse als ein eigenes System betrachtet wird, benötigen wir für jede Masse die zugehörige Bewegungsgleichung. Allgemein ist die Bewegungsgleichung einer einzelnen Masse  $m_n$  gegeben als

$$\ddot{x}_n = \frac{1}{m_n} [k_n(x_{n-1} - x_n) - k_{n+1}(x_n - x_{n+1})]$$

mit  $x_n$  als der Position dieser Masse,  $x_{n-1}$  und  $x_{n+1}$  als den Positionen der Massen links und rechts davon sowie  $k_n$  und  $k_{n+1}$  als den Federkonstanten der Federn links bzw. rechts der Masse.

§ 244 Aus dieser Bewegungsgleichung lässt sich das Subsystem wie in Abb. 2.24 gezeigt konstruieren. Dazu öffnen wir konventionell ein neues Modell in SIMULINK und ziehen einen Subsystems-Block auf die Arbeitsfläche. Dieser hat, wie die meisten SIMULINK-Blöcke als Default einen Ein- und Ausgang. Über diese Eingänge kann die Verknüpfung mit anderen Untersystemen oder mit einem konventionellen SIMULINK-Block erfolgen. Linker Doppelklick auf dieses Modell öffnet in einem neuen Fenster eine Arbeitsfläche, auf der sich das Untersystem so aufbauen lässt, wie wir es von einem normalen Modell kennen.<sup>14</sup> Zusätzliche Input-

<sup>14</sup>Um nicht in Platz-Probleme zu kommen, sollten Sie das Modell eher von links nach rechts aufbauen als umgekehrt: der linke Rand lässt sich nicht sehr weit zurück schieben.

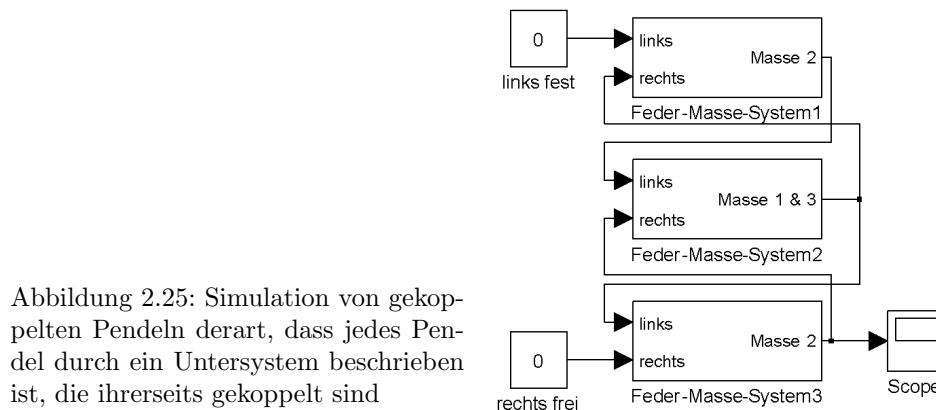
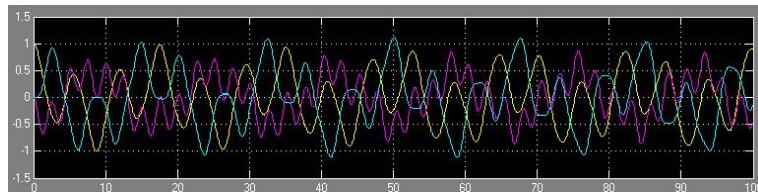


Abbildung 2.25: Simulation von gekoppelten Pendeln derart, dass jedes Pendel durch ein Untersystem beschrieben ist, die ihrerseits gekoppelt sind

Abbildung 2.26: Simulationsergebnis zum System aus drei Massen und drei Federn aus Abb. 2.23



oder Output-Ports lassen sich durch kopieren im Subsystem selbst erzeugen. In diesem Beispiel sind die Parameter der einzelnen Gain Blöcke nicht explizit angegeben sondern durch Variablen (die Federkonstanten der mit der betrachteten Masse verbundenen Federn sowie die Masse).

§ 245 Der Aufbau der Simulation beinhaltet die beiden bereits hinlänglich bekannten Integratoren. Der Output des zweiten Integrators, d.h. die Größe  $x_k$  wird zum einen auf den Output des Untersystems gegeben, um den anderen Untersystemen zur Verfügung zu stehen. Zum anderen wird dieser Output wieder zurückgeführt: seine Differenz zu den Outputs der beiden anderen Untersysteme gibt den Abstand zwischen den Massen und damit, multipliziert mit der entsprechenden Federkonstanten, die von den Federn ausgeübten Kräfte. Diese werden summiert und mit der Masse skaliert, um die Beschleunigung zu erhalten. Im Prinzip können wir uns diesen Gain Block sparen und stattdessen in den beiden anderen Blöcken jeweils statt der Federkonstanten die Federkonstanten/Masse angeben. Das würde bei der konventionellen Parameterdefinition durch Anklicken der Symbole etwas Klicken sparen – diese Form des Untersystems hat den Vorteil, dass die Parameter der Systembestandteile wie Massen und Federkonstanten in reiner Form vorliegen und nicht bereits in einem abstrakteren Systemparameter wie der Schwingungsfrequenz zusammengefasst sind.

§ 246 Das Gesamtmodell ergibt sich durch Kopieren des Blocks für das Subsystem und korrekte Verbindung der einzelnen Subsysteme wie in Abb. 2.25 gezeigt. In den einzelnen Subsystemen müssen jetzt noch die Variablennamen für die verschiedenen Parameter (und falls gewünscht die Bezeichnungen/Erklärungen der einzelnen Blöcke) angepasst werden.

## 2.4.2 Übergabe der Modellparameter

§ 247 Im Gegensatz zu früheren Modellen werden in diesem Modell die Parameter nicht mehr direkt in die Blöcke eingetragen – schließlich müssen wir sonst jeden Block in jedem Untersystem einmal anfassen. Stattdessen definieren wir diese in einem MATLAB-Skript. Dieses wird vor dem Start der Simulation unter dem MATLAB prompt aufgerufen. Daher muss es einen anderen Namen haben als das Modell – sonst versucht MATLAB den mdl-File statt des m-Files auszuführen.



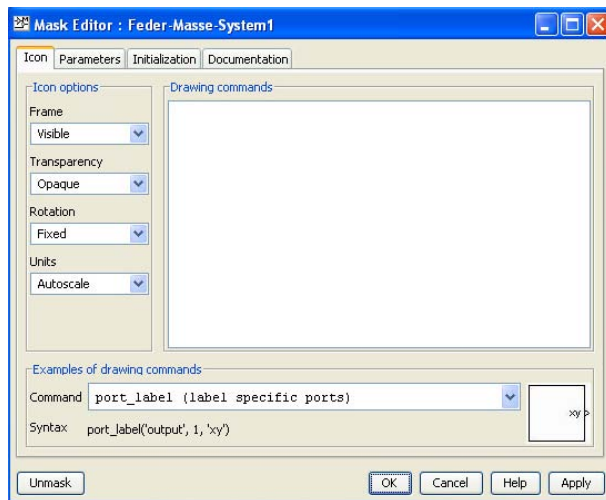


Abbildung 2.27: Editor für ein maskiertes Subsystem, Eröffnungsseite

§ 248 In den einzelnen Untersystemen tragen wir dazu in den ‘Gain Parameters’ keinen Zahlenwert für den Gain ein sondern einen Variablennamen. In den Untersystemen sind die Federkonstanten jeweils als  $k_i$ , die Massen als  $m_i$  bezeichnet. Die zugehörige Datei mit den Parametern ordnet einfach diesen Variablennamen Werte zu:

Parameter für Feder-Masse System mit Untersystemen

```
k1=1;
k2=2;
k3=3;
m1=3;
m2=1;
m3=2;
```

Damit haben wir uns das Klicken für die Parameter erspart – allerdings ist die Simulation recht unbefriedigend solange wir nicht an irgendeinem Integrator eine Anfangsbedingung ungleich Null spezifizieren. Für diesen Run ist im ersten Subsystem die Masse ausgelenkt, d.h. der Integrator für den Ort ist auf die Anfangsbedingung 1 gesetzt.<sup>15</sup> Die Ergebnisse sind in Abb. 2.26 gezeigt.

### 2.4.3 Maskierte Untersysteme

§ 249 SIMULINK bietet eine weitere einfache Möglichkeit im Umgang mit Untersystemen: das Untersystem wird ‘maskiert’ und als ein neuer SIMULINK-Block definiert. Jedes Untersystem hat dann eine einzige Karte ‘Block Parameters’, in der alle Parameter eingetragen werden können.

§ 250 Als Beispiel nehmen wir wieder das Modell der drei Federn und Massen aus Abb. 2.23. Die dort verwendeten Untersysteme wandeln wir in Blöcke um, in dem wir im Hauptsystem den Block des entsprechenden Untersystem markieren und über den Menüpunkt **Edit** ↔ **Mask Subsystem** den Masken-Editor aufrufen. Dieser erscheint wie in Abb. 2.27; er besteht aus vier Karteikarten.

#### Parameter

§ 251 Die Karteikarte Parameter, vgl. Abb. 2.28, erlaubt die Definition von Parametern für die im Untersystem enthaltenen Blöcke. Im oberen Teil der Seite lassen sich die Dialog-

<sup>15</sup>Alternativ können wir diesem Feld natürlich auch einen Variablennamen zuweisen und dann die Parameter mit der Datei (oder von Hand aus dem Workspace heraus) übergeben.

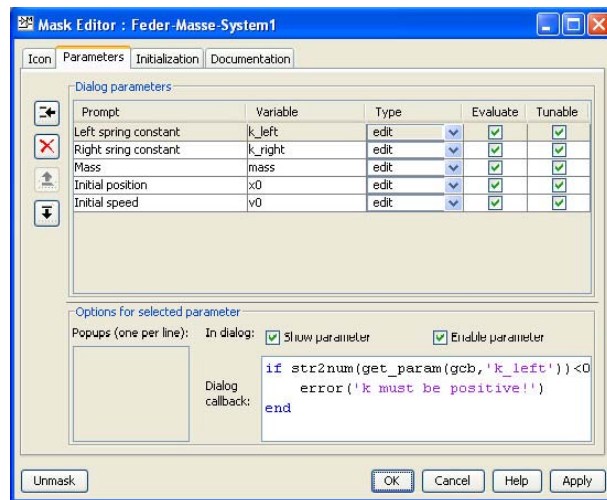


Abbildung 2.28: Editor für ein maskiertes Subsystem, Parameter

felder einstellen und mit einer MATLAB-Variablen verknüpfen; im unteren Teil lassen sich zusätzliche Optionen für diese Parameter einstellen.

§ 252 Die Schaltflächen an der Seite des oberen Feldes erlauben das Hinzufügen, Löschen und Verschieben von Dialogfeldern zur Eingabe von Parametern. Der oberste Button ‘add’ fügt ein Dialogfeld hinzu: in dieses lässt sich die Abfrage auf der Karteikarte (Prompt) angeben sowie der Name der Variablen, der dieser Wert zugewiesen werden soll. Weiteres Hinzufügen von Dialogfeldern erlaubt die Eingabe von weiteren Parametern – in der Version in Abb. 2.28 haben wir auch die Anfangswerte der Integratoren mit übergeben.

§ 253 Das dritte Feld in der Liste bezeichnet den Typ der Eingabe. Der default ist edit, d.h. die Werte werden eingegeben. Als Alternativen stehen **Checkbox** und **Popup** zur Verfügung: in ersterem Fall wird ein Feld erzeugt, in dem zwei Möglichkeiten zur Auswahl stehen, je nachdem, ob die Checkbox markiert ist oder nicht. Ein derartiges Feld kann z.B. verwendet werden, um bei Winkelgrößen zwischen Grad- und Bogenmass zu unterscheiden, so dass die Eingabe nicht von vornherein auf die eine Möglichkeit eingeschränkt ist. In der Variante **Popup** kann in einem Menü zwischen verschiedenen Möglichkeiten ausgewählt werden.

§ 254 Die anderen beiden Spalten schränken die Behandlung der Variablen ein. Falls in der Simulation einer Variablen bereits ein Wert zugewiesen wurde, so erscheint dieser bei angekreuztem Feld **Evaluate** im Dialogfenster. Ist der Variablen kein Wert zugeordnet oder ist **Evaluate** ausgeschaltet, so wird nur der Variablenname als Zeichenkette angezeigt. **Tunable** erlaubt eine Veränderung der Variablen während des Programmablaufs. So lassen sich einerseits Parameter gegen versehentliches Überschreiben schützen – bei einer Anfangsbedingung wäre dieses Verbot des Überschreibens allerdings fatal. Daher entweder immer **tunable** angekreuzt lassen oder vor Veränderung der Markierung gründlich nachdenken.

§ 255 Das **Callback**-Feld im unteren Teil der Karte erlaubt es, mit dem ausgewählten Parameter ein Stück MATLAB-Code auszuführen, z.B. um den wahrscheinlichen Vorzeichenfehler bei der Eingabe zu vermeiden: der Benutzer weis möglicherweise nicht genau, ob wirklich die Federkonstante als positive Größe oder der Eintrag in den Gain-Block (also das negative der Federkonstante) gewünscht ist.

## Initialisierung

§ 256 Die Karte für die Initialisierung, wie in Abb. 2.29 gezeigt, liefert eine Liste der im Block behandelten Variablen. Sie kann beliebige MATLAB-Statements enthalten und auf diese Weise MATLAB-Variablen Werte zu weisen, die in der Konfiguration der Blöcke im

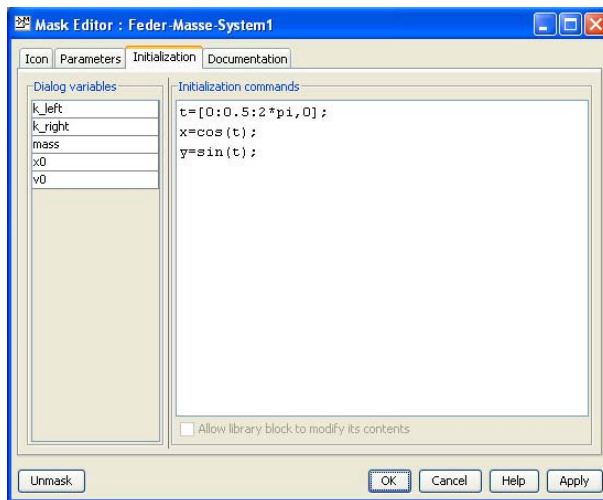


Abbildung 2.29: Editor für ein maskiertes Subsystem, Initialisierung

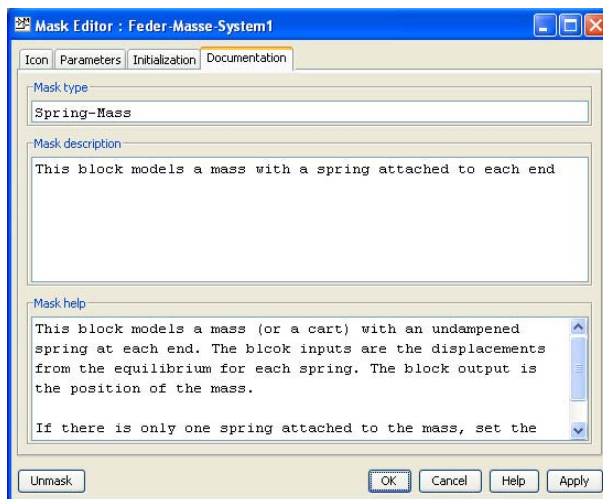


Abbildung 2.30: Editor für ein maskiertes Subsystem, Dokumentation und Hilfe

maskierten Untersystem benötigt werden können. So können wir z.B. unserem maskierten Block ein Icon geben, in dem die Masse wirklich als Wägelchen und nicht einfach als Rechteck dargestellt ist. Dazu benötigen wir z.B. Räder, die wir mit Hilfe der in Abb. 2.29 erzeugten Vektoren darstellen können.

§ 257 Die hier erzeugten Variablen sind lokale Variablen. Sie gehen weder in den MATLAB-Workspace ein noch können von dort Variablen importiert werden. Das gilt übrigens für alle Variablen innerhalb eines maskierten Untersystems. Damit unterscheiden sie sich von einem gewöhnlichen Untersystem: dem konnten wir in § 248 einfach Variablen aus dem Workspace zuweisen. Das maskiert Untersystem dagegen ist gegenüber dem Workspace blind und umgekehrt – daher auch der Name. Die einzigen ausgetauschten Größen sind die, die im Dialogfeld des maskierten Untersystems auftreten.

§ 258 Kommandos im Initialisierungsfenster sollten jeweils mit einem Semikolon beendet werden (Echo off); es können alle MATLAB-Operationen und -Funktionen ebenso wie selbst geschriebenen Funktionen verwendet werden. Auch die üblichen Steuerstrukturen wie `if`, `while` und `end` können benutzt werden.

### Dokumentation

§ 259 Die letzte Karteikarte, siehe Abb. 2.30, dient der Dokumentation und ist optional. Trotzdem sollte man sie nicht vernachlässigen: die Einträge erscheinen auf der Block Pa-

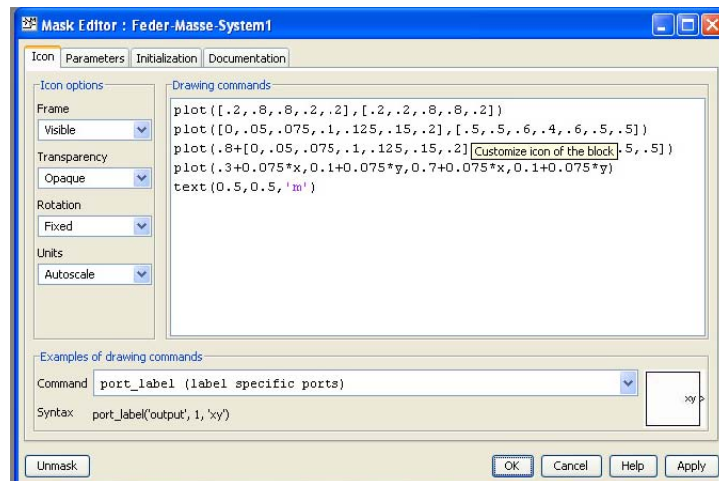


Abbildung 2.31: Editor für ein maskiertes Subsystem, Icon

Parameters Karte und liefern nicht nur eine Kurzbeschreibung (der Text aus dem Feld Mask description) sondern auch die richtige Hilfe.

### Icon

§ 260 Die erste Karte haben wir uns bis zuletzt aufbewahrt. Für das Icon können wir entweder mit `imread` ein `.tif`-Bild einlesen oder wir müssen etwas basteln. Die Karte erlaubt eine Vielzahl von Einstellungsmöglichkeiten (Icon mit oder ohne Rahmen, transparent oder nicht ...), das Icon selbst müssen wir aber im Fenster drawing commands zusammen basteln. Dabei können wir auf die üblichen MATLAB-Befehle zurück greifen.

§ 261 Schließen wir jetzt den Editor, so sollte sich das Subsystem in dieses Icon umgewandelt haben. Anklicken des Icons öffnet das Dialog-Feld zum Eintrag der Parameter. Diese können als numerische Werte eingetragen oder als Variablen definiert werden – letztere können dann wie bei den gewöhnlichen Blöcken über den Workspace oder ein m-File eingegeben werden.

### Das zusammengebaute System

§ 262 Das komplette System erhalten wir, in dem wir auch die anderen beiden Untersysteme durch das maskierte Untersystem ersetzen, vgl. Abb. 2.32. Da die maskierten Blöcke die gleichen Ein- und Ausgabeports haben wie die unmaskierten Untersysteme, ändert sich an den Verbindungen nichts – und am Ergebnis zum Glück auch nichts.

### Und wohin mit den maskierten Blöcken?

§ 263 Maskierte Blöcke haben in SIMULINK die gleiche Funktion wie Unterprogramme in der konventionellen Programmierung oder Funktionen in MATLAB: sie halten nicht nur ein spezielles Programm übersichtlich sondern stehen auch zur weiteren Verwendung in anderen Programmen, bzw. im Falle der Blöcke in anderen Modellen, zur Verfügung. Entsprechend erlaubt SIMULINK die Erstellung einer Block Library. Kopieren wir einen Block aus dieser Bibliothek in ein Modell, so bleibt die Verbindung vom Modell zum Block erhalten: Änderungen des Blocks in der Library werden auch im Modell wirksam.

§ 264 Eine neue Bibliothek lässt sich in SIMULINK über `File` ↔ `New` ↔ `Library` erzeugen. Es öffnet sich ein Fenster `Library:untitled`, in das wir den maskierten Block (und bei Bedarf weitere Blöcke, die wir in dieser Bibliothek ablegen wollen) hinein kopieren können. Speichern erzeugt den Bibliotheksblock, alle seine Kopien sind Referenzblöcke. Zu Änderungen und dem Vererben von Eigenschaften gelten die üblichen Regeln, weitere Informationen in der MATLAB-Hilfe.

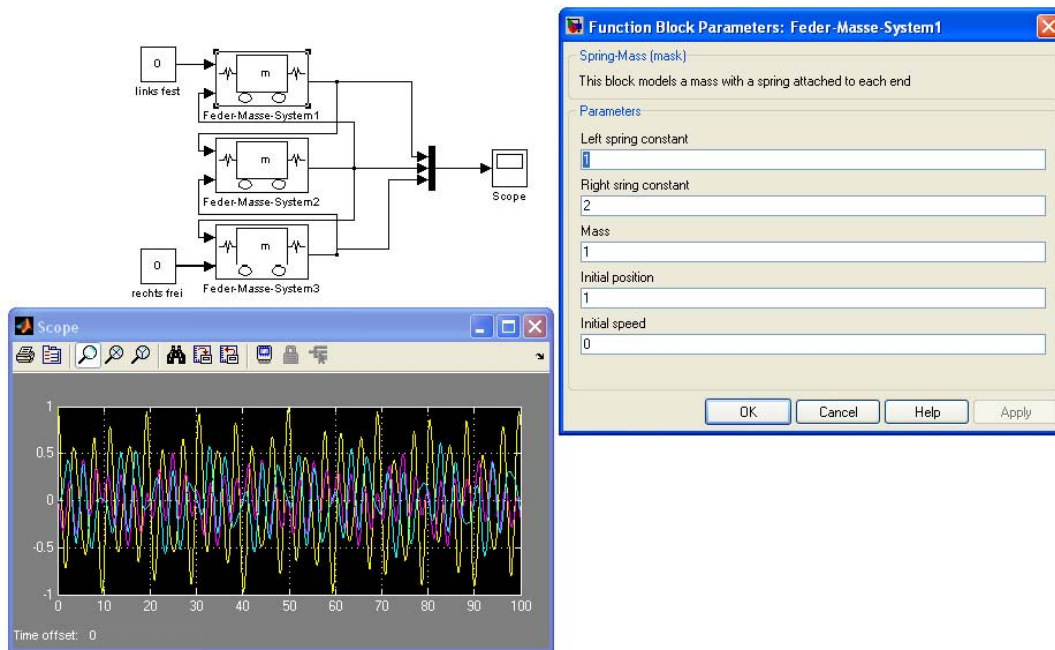


Abbildung 2.32: Das Feder-Masse System aus Abb. 2.23 mit maskierten Untersystemen. Im rechten Teil ist die Dialog-Box für das obere Untersystem geöffnet

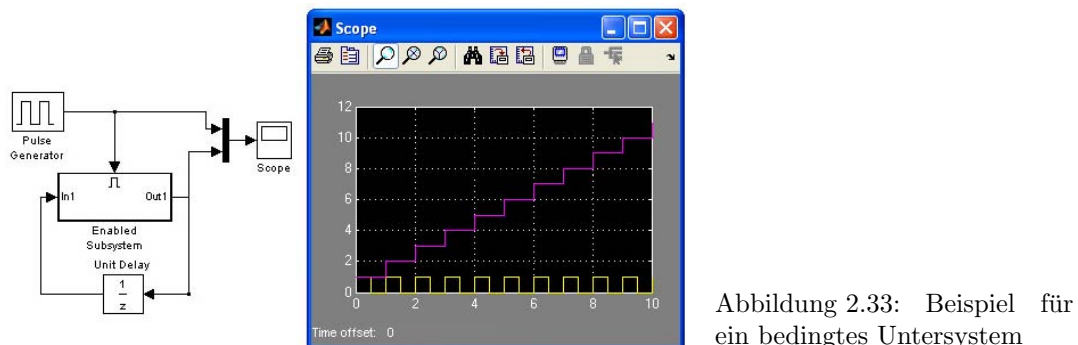


Abbildung 2.33: Beispiel für ein bedingtes Untersystem

## 2.4.4 Bedingte und getriggerte Untersysteme

§ 265 Neben dem bisher betrachteten normalen Untersystem gibt es in der Port & Subsystems Library zwei Blöcke für nur unter bestimmten Bedingungen auszuführende Untersysteme. Das **Enabled Subsystem** wird immer dann ausgeführt, wenn ein Kontrolleingang einen positiven Wert annimmt. Das **Triggered Subsystem** dagegen wird einmal ausgeführt, wenn ein Trigger-Signal eintrifft. Der Block **Enabled** and **Triggered Subsystem** hat zwei steuernde Eingänge, den Trigger und den Kontrolleingang.

§ 266 Das bedingte Untersystem ist insbesondere bei der Modellierung eines Systems von Interesse, das in verschiedenen Phasen oder Moden unterschiedlich funktioniert. Die Aerodynamik eines Flugzeugs in der Startphase ist sehr unterschiedlich von der in der Flugphase. Beide Betriebsmodi werden daher durch unterschiedliche Algorithmen beschrieben – was zur Reduktion des Rechenaufwandes die Verwendung verschiedener Untersysteme nahe legt.

§ 267 Abbildung 2.33 zeigt ein Beispiel für ein einfaches bedingtes Untersystem. Im Untersystem selbst wird nur jeweils eine 1 addiert. Da die Addition nur dann erfolgt, wenn der Kontrolleingang ein positives Signal hat, eignet sich dieses Verfahren zum Zählen von Pul-

Abbildung 2.34: Wie Abb. 2.33 aber mit anderem Eingangssignal und erzwingener Zeitaufösung

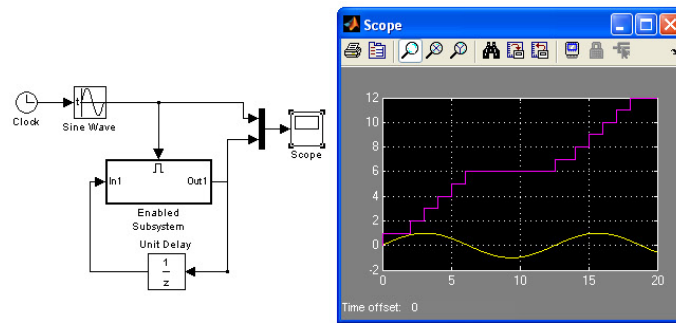
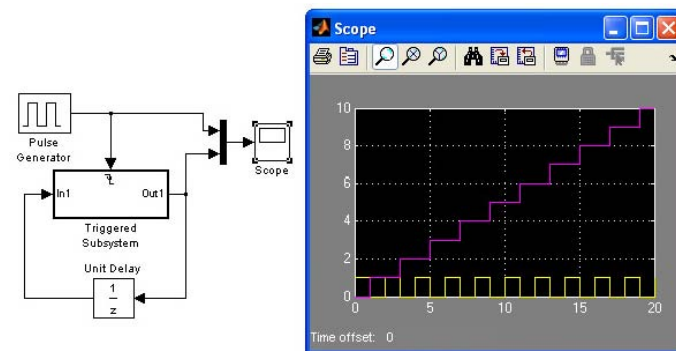


Abbildung 2.35: Getriggertes Untersystem



sen. Der Block Unit Delay ist hier irrelevant; er dient nur dem Aufbrechen der sonst durch eine direkte Rückführung des Ausgangs auf den Eingang entstehenden algebraischen Schleife (siehe auch Abschn. 2.6.6).

**Verständnisfrage 23** Eignet es sich wirklich zum Zählen von Pulsen oder ist das nur Zufall?

§ 268 Zum Zählen von Pulsen ist der Enabled Subsystem Block natürlich nicht geeignet: er zählt immer, wenn das Kontrollsignal positiv ist. Das scheinbare Zählen der Pulse in Abb. 2.33 war Zufall, da die Zeitaufösung der Simulation gerade der Pulsdauer entsprach. Ist dies nicht der Fall, so kann man bestenfalls die Zahl der Zeitintervalle zählen, in denen das Signal positiv ist, wie in Abb. 2.34 für ein Sinusförmiges Eingangssignal dargestellt.

§ 269 Das Zählen der Pulse lässt unabhängig von Problemen der Zeitaufösung mit dem Triggered Subsystem erreichen, siehe auch Abb. 2.35. Hier lässt sich Einstellen, ob die steigenden, fallenden oder beide Flanken zum Triggern verwendet werden sollen (in der Abbildung wurde offensichtlich auf die fallende Flanke getriggert), oder ob gemäß einer selbst definierten Funktion getriggert werden soll.

§ 270 Der Triggered and Enabled Subsystem Block hat sowohl einen Trigger- als auch einen Kontrolleingang. Bei Auftreten des Trigger-Signals wird zuerst der Kontrolleingang überprüft und dann bei Anliegen des positiven Eingangssignals das Untersystem ausgeführt. Trigger und Bedingung sind daher über ein logisches Und verknüpft.

### 2.4.5 Startende Rakete: einfaches zeitvariantes System mit Untersystem

§ 271 Eine startende Rakete ist ein einfaches mechanisches Beispiel für ein sich mit der Zeit entwickelndes System. Die zeitliche Abhängigkeit von Eingangs-, Ausgangs- und Zustandsgrößen ist auch in zeitinvarianten Systemen wie dem Hoch- und Tiefpass gegeben; die Systemgleichungen dagegen hängen in einem zeitinvarianten System nicht von der Zeit ab.

§ 272 In einem dynamischen oder zeitvarianten System dagegen hängen die Systemgleichungen von der Zeit ab. Bei der startenden Rakete ist dies offensichtlich: ihre Bewegungsgleichung enthält die Masse und diese ist eine Funktion der Zeit:  $m = m(t)$ . Während die Rakete ein dynamisch recht langweiliges System ist, werden wir an Hand des Feder-Fadenpendels und des Pendels an einer Rolle sehr einfache zeitvariante Systeme kennen lernen, die den Übergang zum Chaos erlauben.

§ 273 Als Beispiel folgen wir Lutz und Wendt [87] und betrachten eine Stufe einer Saturn-V-Rakete. Um das System etwas realistischer zu gestalten (und die Verwendung eines Untersystems zu motivieren) berücksichtigen wir eine von der Höhe abhängige Erdbeschleunigung sowie Reibung.

### Die Bewegungsgleichung

§ 274 Da sich die Masse der Rakete während der Beschleunigung ändert, ist die Bewegungsgleichung gegeben zu

$$\dot{m}\ddot{x} = \sum F_i \quad \Rightarrow \quad \ddot{x} = \frac{1}{\dot{m}} \sum F_i .$$

Als Kräfte sind auf der rechten Seite die Schubkraft, die Reibung und die Gravitation einzusetzen.

§ 275 Die von der Höhe (und der zeitlich veränderlichen Masse) abhängige Gravitationskraft ist

$$F_g = -m(t)g(x(t)) = -m(t)g_0 \frac{R}{R+x}$$

mit  $g_0$  als der Gravitationsbeschleunigung am Erdboden,  $R$  als dem Erdradius und  $x$  als der Höhe (und damit gleichzeitig als der Ortskoordinate der Rakete).

**Verständnisfrage 24** Erläutern/Begründen Sie die gegebene Abhängigkeit von  $g(x)$ .

§ 276 Die Schubkraft hängt von der Ausströmgeschwindigkeit  $v_{\text{rel}}$  der Verbrennungsgase und dem Massenstrom  $\dot{m}$  ab:

$$F_{\text{Schub}}(t) = -\dot{m}(t) v_{\text{rel}}(t) = -\frac{m_{\text{leer}} - m_0}{t_B} v_{\text{rel}} .$$

Dabei wird die Ausströmgeschwindigkeit  $v_{\text{rel}}$  als konstant angenommen. Der Massenstrom ergibt sich bei konstanter Verbrennungsrate zu

$$\dot{m} = \frac{m_{\text{leer}} - m_0}{t_B}$$

mit  $m_0$  als der Startmasse der vollbetankten Rakete,  $m_{\text{leer}}$  als der Masse der leeren Rakete bei Brennschluss und  $t_B$  als der Brenndauer. Aus dem Massenstrom lässt sich auch direkt die Abhängigkeit der Raketenmasse von der Zeit bestimmen als

$$m(t) = m_0 - \frac{m_0 - m_{\text{leer}}}{t_B} t .$$

§ 277 Die Reibungskraft ist bestimmt durch den Widerstandsbeiwert  $c_w$ , die angeströmte Fläche  $A$ , die von der Höhe abhängige Dichte  $\varrho(x)$  der Luft sowie der Geschwindigkeit  $v$ :

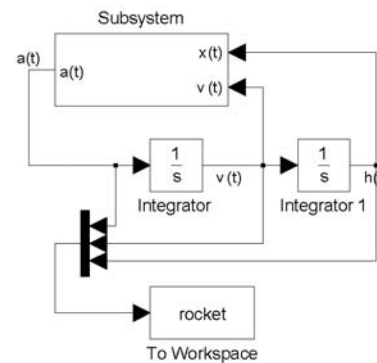
$$F_{\text{Reib}} = -\frac{1}{2} c_w A \varrho(x) \dot{x}^2 .$$

Die Höhenabhängigkeit der Dichte lässt sich abschätzen zu

$$\varrho(x) = \varrho_0 \left( 1 - 22.6 \cdot 10^{-6} \frac{x}{1 \text{ m}} \right)^{4.255}$$

mit  $\varrho_0$  als der Dichte am Erdboden.

Abbildung 2.36: Raketenstart  
 Hauptsystem: da das Modell auf der Bewegungsgleichung basiert, muss letztendlich nur die Beschleunigung zwei mal integriert werden



**Verständnisfrage 25** Erläutern Sie die einzelnen Abhängigkeiten in der Reibungskraft anschaulich.

§ 278 Kombination dieser Kräfte liefert für die Beschleunigung in Abhängigkeit von Zeit  $t$ , Ort  $x$  und Geschwindigkeit  $\dot{x}$

$$\ddot{x}(t, x, \dot{x}) = \frac{1}{m_0 - \frac{m_0 - m_{\text{leer}}}{t_B} t} \left( \frac{m_0 - m_{\text{leer}}}{t_B} v_{\text{rel}} - \frac{1}{2} c_w A \rho_0 \left( 1 - 22.6 \cdot 10^{-6} \frac{x}{1 \text{ m}} \right)^{4.255} \dot{x}^2 \right) - g_0 \frac{R}{R + x}$$

### Das Modell

§ 279 Das Modell zerfällt in zwei wesentliche Bestandteile: die Berechnung der zeitabhängigen Beschleunigung aus der ortsabhängigen Gravitationskraft, der zeitabhängigen Masse und der geschwindigkeitsabhängigen Reibungskraft sowie die zweimalige Integration dieser Beschleunigung, da wir wieder von der Bewegungsgleichung und damit einer DGL zweiter Ordnung ausgehen. Den letzten Teil erledigt das eigentliche System wie in Abb. 2.36 gezeigt. Da die drei interessierenden Größen  $x(t)$ ,  $\dot{x}(t)$  und  $\ddot{x}(t)$  einen sehr unterschiedlichen Dynamikbereich haben, macht es keinen Sinn, sie auf ein Oszilloskop zu geben. Da drei Oszilloskope nicht gerade übersichtlich sind, erfolgt die Ausgabe stattdessen als strukturiertes Element mit dem Namen `rocket` in den Workspace.

§ 280 Bei der erzwungenen Schwingung haben wir die Beschleunigung jeweils mit Hilfe von Gain und Summationsblöcken im Hauptmodell bestimmt. Hier ist die Beschleunigung so komplex, dass das Modell dadurch unübersichtlich würde. Daher lagern wir sie in ein Untersystem aus wie in Abb. 2.37 gegeben. Dieses hat insgesamt drei Eingänge, entsprechend der Zahl der Variablen, von denen die Beschleunigung abhängt. Dabei werden  $x$  und  $\dot{x}$  aus dem Hauptmodell bezogen (entsprechend den sichtbaren Eingängen des Untersystems), die Zeit dagegen wird von der Uhr `Clock` bezogen.

§ 281 Das Untersystem besteht im wesentlichen aus Blöcken für mathematische Funktionen (haben links einen Eingang) und Blöcken für Konstanten (haben keinen Eingang). Da die Funktionen nicht immer so einfach wie ‘quadrieren’ sind, wird nicht der der Block `Math Functions` aus der Blockgruppe Math Operations verwendet sondern der Block `Fcn` aus der Gruppe User-Defined Functions. In allen Blöcken sind die Parameter über Variablenamen gegeben, die aus dem Workspace oder über ein m-File übergeben werden.

§ 282 Die obere Funktion berechnet die zeitabhängige Masse  $m(t)$  bzw. genauer deren Kehrwert. Der zweite Block bestimmt die Schubbeschleunigung. Da diese konstant ist, wird der Block als Konstante und nicht als Funktion vereinbart. Die folgenden drei Blöcke sind die verschiedenen Faktoren der Reibungsbeschleunigung: der obere Block enthält die konstanten Faktoren wie Fläche und Widerstandsbeiwert, der mittlere Block ist eine Funktion, die das



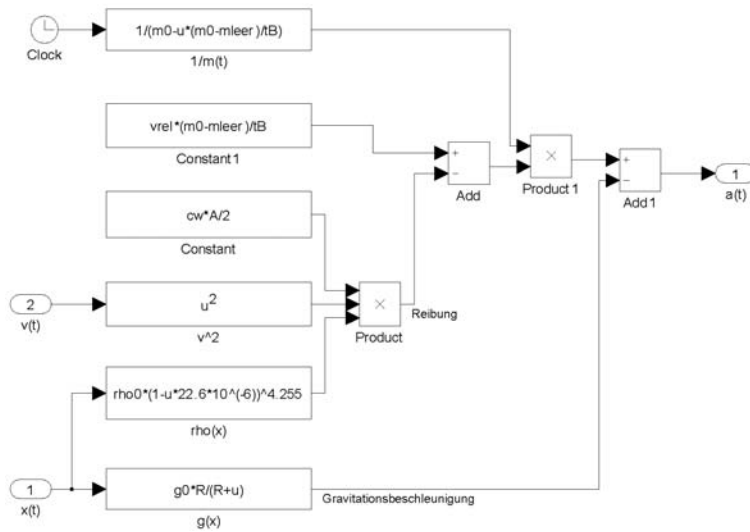


Abbildung 2.37: Raketenstart Subsystem: die zeitabhängige Beschleunigung wird bestimmt aus der ortsabhängigen Gravitationskraft, der zeitabhängigen Masse und der geschwindigkeitsabhängigen Reibungskraft

Quadrat der Geschwindigkeit bildet; der untere Block berechnet die Dichte in Abhängigkeit vom Ort, ist also ebenfalls eine Funktion. Die unterste Funktion bestimmt die Gravitationsbeschleunigung in Abhängigkeit von der Höhe.

§ 283 Die Parameter werden angepasst an eine Saturn-V gewählt und in einem m-File definiert. Da auch die Ausgabe in den Workspace erfolgt, verwenden wir den m-File gleich zur Steuerung von Simulation und Ausgabe: mit dem `sim` Befehl wird die Simulation gestartet. Die folgenden Zeilen lesen aus dem an den Workspace zurück gegebenen strukturierten Element die einzelnen Variablen aus und stellen sie mit den üblichen MATLAB-Befehlen graphisch dar.

```
%Parameter für Modell zum Raketenstart
m0=2.95*10^6; %Masse Anfangswert [kg]
mleer=10^6; %Masse nach Brennschluss (Leermasse) [kg]
tB=130; %Brennschluss [s]
vrel=2220; %Ausströmgeschwindigkeit [m/s]
cw=0.34; %Luftwiderstandsbeiwert
rho0=1.2255; %Luftdichte am Erdboden [kg/m^3]
A=132; %angeströmte Fläche [m^2]
R=6378163; %Erdradius [m]
g0=9.81; %Gravitationsbeschleunigung am Erdboden [m/s^2]
sim('raketenstart',[0,tB])
subplot(3,1,1);plot(tout,rocket.signals.values(:,1),'k')
ylabel('a(t) [m/s^2]');axis([0 tB 0 25])
subplot(3,1,2);plot(tout,rocket.signals.values(:,2)'/3.6,'k')
ylabel('v(t) [km/h]');axis([0 tB 0 350])
subplot(3,1,3);plot(tout,rocket.signals.values(:,3)'/1000,'k')
xlabel('Time [s]');ylabel('h(t) [km]');axis([0 tB 0 50])
```

§ 284 Die Option, statt der Ausgabe in das strukturierte Element `rocket` beim Aufruf der Simulation die Variablen in den Workspace zu holen als `[t,x]=sim('raketenstart',[0,tB])`, hätte nur zu einer Übergabe der Größen jeweils am Ausgang der Integratoren, d.h. von Ort und Zeit geführt. Die Beschleunigung hätten wir auf diese Weise nicht in den Workspace einlesen können. Um auch diese zu berücksichtigen (insbesondere, da sie als Testgröße geeignet ist), wurde der Weg über das explizite rausschreiben gewählt.

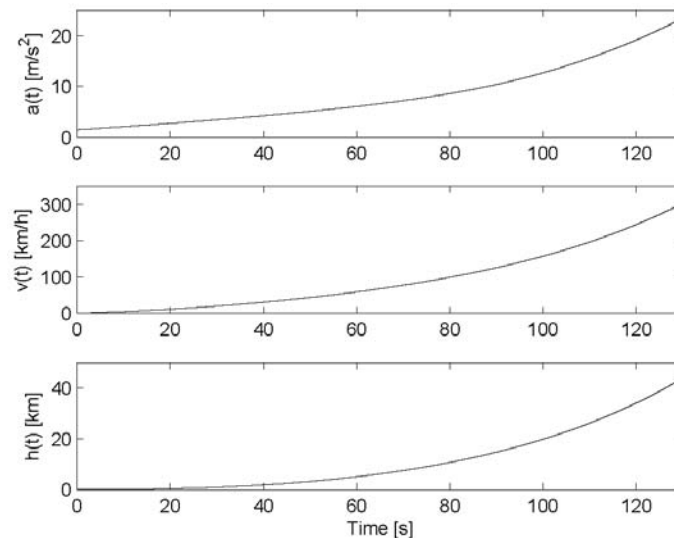


Abbildung 2.38: Raketenstart  
Ergebnisse: Beschleunigung,  
Geschwindigkeit und Flughöhe  
in Abhängigkeit von der Zeit

§ 285 Abbildung 2.38 zeigt die Zeitverläufe von Beschleunigung (oben), Geschwindigkeit (Mitte) und Flughöhe (unten) in Abhängigkeit von der Zeit bis zum Brennschluss der Raketenstufe. Wie zu erwarten steigen alle drei Größen zuerst langsam, mit zunehmender Zeit immer schneller an: da die Schubkraft konstant ist, wird die Beschleunigung mit abnehmender Masse und damit mit zunehmender Zeit größer. Zusätzlich wächst die Beschleunigung auch mit zunehmender Flughöhe, da die Gravitationskonstante abnimmt – wobei die mit quadratisch mit der Geschwindigkeit zunehmende Reibungskraft das Wachstum der Beschleunigung begrenzt.

**Zwischenrechnung 14** Strukturieren Sie die Simulation so um, dass auch nach Brennschluss noch korrekt weiter gerechnet wird. Wo muss der Switch gesetzt werden?

§ 286 Auf der Basis der Bewegungsgleichung lassen sich etliche ähnliche zeitvariante Systeme konstruieren: (a) ein Güterwagen wird von oben befüllt während er unter einem Silo hindurch rollt (wieder veränderliche Masse); (b) der Streuwagen vom Winterdienst ist die Umkehrung des Güterwagens und der Rakete recht ähnlich; (c) ein Fadenpendel mit veränderlicher Fadenlänge; (d) ein Fadenpendel, bei dem der Faden durch eine Feder ersetzt ist (Absch. 6.3.1); (e) ein Fadenpendel an einem rotierenden Galgen; (f) ein Federpendel, das horizontal um die Aufhängung rotiert; .... lassen Sie sich weitere, vielleicht nicht nur auf die Mechanik beschränkte Beispiele einfallen.

**Verständnisfrage 26** Ist das Fadenpendel mit veränderlicher Masse (die Pendelmasse ist eine Eiskugel, die langsam sublimiert) ein vergleichbares Problem? Wie sieht es beim Federpendel mit veränderlicher Masse aus?

## 2.5 Kommunikation mit dem Workspace

§ 287 Da wir in Abschn. 2.4.2 die Parameter für die Gain-Blöcke einfach mit Hilfe eines m-Files in den Workspace schreiben konnten und sie dort von SIMULINK abgeholt wurden, findet ganz offenbar eine Kommunikation zwischen Simulation und Workspace statt. In diesem Abschnitt wollen wir diese etwas genauer betrachten und uns zu Nutze machen.

### 2.5.1 GUI oder nicht-GUI?

§ 288 SIMULINK ist uns bisher hauptsächlich in Form des GUIs begegnet: Modelle haben wir durch verbinden geeigneter Blöcke ähnlich einem Flussdiagramm erzeugt. Die dabei erzeugte mdl-Datei ist eine Textdatei, die sich mit einem normalen Editor öffnen lässt. In ihr finden sich

alle Einstellungen zur Simulation im allgemeinen (entsprechend der Simulationssteuerung), die Informationen über die verwendeten Blöcke und die darin enthaltenen Parameter, sowie die zur graphischen Darstellung notwendigen Positionsinformationen. Im Prinzip können wir das Modell nicht nur im graphischen Modus modifizieren sondern auch in dieser Datei manuell Änderungen vornehmen.

§ 289 Aus dem Workspace können wir (ohne Verwendung des GUI) auf ein Modell und auch auf diese Informationen zurück greifen. In § 247 haben wir bereits darauf hingewiesen, dass das m-File zur Übergabe von Parametern nicht den gleichen Namen haben darf, wie das mdl-File des Modells, da MATLAB sonst das mdl-File statt des m-Files ausführt. Ohne SIMULINK explizit zu öffnen können wir nämlich mit

```
>> uiopen('gedaempfte_schwingung.mdl',1)
```

das Modell `gedaempfte_schwingung.mdl` (vgl. Abb. 2.16) aus dem MATLAB Kommando-Fenster heraus öffnen; die einfache Eingabe des Namens des mdl-Files reicht hierzu auch schon aus. Mit dem Befehl `gcs` (get current system) lässt sich aus dem Kommandofenster heraus das gerade in SIMULINK geöffnete Modell identifizieren:

```
>> gcs
ans =
gedaempfte_schwingung
```

§ 290 Informationen über das so identifizierte System können wir mit `find_system` erhalten:

```
>> find_system(gcs, 'Type', 'block')
ans =
'gedaempfte_schwingung/Integration s'
'gedaempfte_schwingung/Integration v'
'gedaempfte_schwingung/Mux'
'gedaempfte_schwingung/Scope'
'gedaempfte_schwingung/Sum'
'gedaempfte_schwingung/beta'
'gedaempfte_schwingung/omega^2'
```

Mit diesem Befehl lassen sich, je nach übergebenen Parametern, auch nach Ports, Verbindungen und Anmerkungen suchen. Auch ein Blick hinter die Masken von maskierten Blöcken ist, bei entsprechender Einstellung, möglich; für weitere Informationen die MATLAB-Hilfe konsultieren.

§ 291 Mit Hilfe von `get_param` können wir den Parameter Denominator des Blocks Transfer Function abfragen:

```
>> get_param('gedaempfte_schwingung/beta', 'Gain')
ans =
-1
```

Verändern können wir diesen (oder jeden anderen) Parameter mit `set_param`:

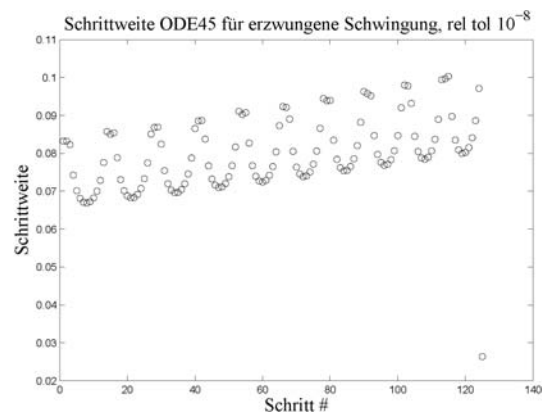
```
set_param('gedaempfte_schwingung/beta', 'Gain', '-0.2')
```

§ 292 Auch der Start der Simulation kann aus dem Kommando-Fenster erfolgen:

```
>> [t,results] = sim('gedaempfte_schwingung', [0,40]);
```

Der Start der Simulation aus dem Workspace hat den Vorteil, dass die Ausgabe ebenfalls direkt in den Workspace in die beim Aufruf der Simulation festgelegten Variablen erfolgt: in diesem Fall Geschwindigkeit und Position als erste und zweite Spalte in der Matrix `Results`. Damit sind wir bei der Darstellung der Ergebnisse nicht mehr auf das Scope angewiesen sondern können aus dem Workspace direkt mit

Abbildung 2.39: Schrittweite des variablen Solvers ODE45 bei Lösung der erzwungenen Schwingung aus Abb. 2.16 für  $\omega_0^2 = 10$  und  $\beta = 0.2$ , Schrittweiten 'auto', relative Toleranz auf  $10^{-8}$  erzwungen



```
>> plot(t,results)
```

eine konventionelle MATLAB-Abbildung erzeugen, die wir mit den üblichen Werkzeugen weiter bearbeiten können. Um die Abbildung im Scope-Fenster zu reproduzieren, können wir es alternativ auch mit `simplot` versuchen

```
>> simplot(t,results)
```

Da die Abbildung mit `simplot` an das Oszilloskop erinnern soll, verwendet MATLAB für die Kurven helle Farben auf dunklem Grund. Für den Druck ist das, wie Sie in den Beispielen im Skript schon hinreichend festgestellt haben, nicht praktisch. Um dunkle Linien vor hellem Hintergrund zu erreichen, kann man vor dem Druck die für das Oszilloskop erforderliche Farbinversion mit der Sequenz `set(gcf, 'InvertHardcopy', 'off')` rückgängig machen.

### Starten einer Simulation mit `sim`

§ 293 Die vollständige Syntax des `sim` Befehls ist

```
>> [t,x,y] = sim(model,TimeSpan,Options,ut)
```

Die Liste der zurück zu gebenden Variablen ist optional; sie kann aus einer, zwei oder allen drei Variablen bestehen. Die erste Variable `t` ist der Wert der simulierten Zeit an jedem Ausgabepunkt. Diese Variable ist nicht nur im Zusammenhang mit den anderen Simulationsergebnissen interessant sondern auch für eine genauere Untersuchung der Arbeitsweise des Solvers: der Solver erzeugt am Ende jedes Integrationsintervalls eine neue Zeitmarke, d.h. die Änderungen in den Abständen der Zeitmarken zeigen die Anpassung des Solvers an die Eigenschaften der zu integrierenden Funktion. Abbildung 2.39 zeigt als Beispiel die Änderung der Schrittweite des Solvers ODE45 für die erzwungene Schwingung; Parameter siehe Bildbeschriftung. Die Ausgabezeitpunkte können jedoch auch mit Hilfe der Simulationskontrolle verändert werden, vgl. Abschn. 2.6.3.

§ 294 Die zweite Rückgabeveriable `x` enthält die Trajektorie der Zustandsvariablen: jede Spalte der Matrix entspricht einer Zustandsvariablen des Systems; jede Zeile entspricht einer festen Zeit, spezifiziert durch das entsprechende Element in `t`. Die letzte Rückgabeveriable `y` enthält die Trajektorie der Ausgangsvariablen. Enthält das Modell keinen Output Block obwohl eine Output Variable erzeugt wurde, so enthält diese Rückgabeveriable einen leeren Vektor `[]`.

§ 295 Die Variablen `model` als String mit dem Namen des Modells und `TimeSpan` als Vektor mit Start- und Endzeit sind bereits aus dem Beispiel bekannt. In der einfachsten Form werden Anfangs- und Endzeit übergeben: enthält der Vektor mehr als zwei Komponenten, so werden die Simulationsergebnisse zu diesen Zeiten ausgegeben. Auch eine Angabe in der Form `[T_Start:TimeSpacing:T_Final]` ist möglich.

§ **296** Während die erste Variable übergeben werden muss, sind alle folgenden optional und erlauben es, verschiedene Parameter in der Konfiguration des Modells zu verändern. Die Variable `Options` erlauben die Veränderung einer großen Zahl von Simulationsparametern; ihre Struktur kann mit `simset` dargestellt und verändert werden. Der Parameter `ut` erlaubt eine Veränderung der vom Modell aus dem Workspace einzulesenden Größen wie in den Simulationsparametern festgelegt. `ut` kann eine Tabelle sein in der Form `[t, u1, u2, ...]` oder als String der Name einer MATLAB Funktion. Eingabegrößen müssen mit einem entsprechendem Import Block im Modell verknüpft sein.

§ **297** Die mit `sim` übergebenen Parameter beeinflussen nur diesen einen Run des Modells; die in der Simulationskontrolle eingetragenen Parameter werden nicht verändert, d.h. sie sind vor dem Ausführen von `sim` die gleichen wie danach.

### Veränderung der Simulationsparameter mit `simset`

§ **298** Die Struktur der mit `sim` zu übergebenden Optionen kann, ebenso wie einzelnen Simulationsparameter, mit `simset` modifiziert werden. `simset` tritt daher in drei Formen auf. Die einfachste Form erlaubt die Spezifikation eines Solvers und seiner Eigenschaften, z.B.

```
options = simset('RelTol',1.0E-4,'Solver','ode4')
```

Mit Hilfe der zweiten Form lässt sich eine bestehende Optionenstruktur verändern:

```
options = simset(olddopts,'name_1',value_1, ...)
```

Mit der dritten Form können die beiden Sätze von Optionen, `olddopts` und `newopts` zu einem neuen Satz verschmolzen werden, wobei `newopts` Priorität hat:

```
options = simset(olddopts,newopts)
```

§ **299** Mit `simset` können die folgenden Eigenschaften gesetzt werden:

`Solver` Auswahl des Solvers; neben den in § 302 beschriebenen Solvern für zeitkontinuierliche Systeme kann auch `'FixedStepDiscrete'` oder `'VariableStepDiscrete'` spezifiziert werden.

`RelTol` relative Toleranz des Verfahrens, siehe § 308.

`AbsTol` absolute Toleranz des Verfahrens, siehe § 308.

`Refine` Verfeinerung Output, siehe auch Abschn. 2.6.3; wird ignoriert, wenn die Zeitpunkte der Ausgabe explizit angegeben werden.

`MaxStep` maximale Schrittweite, siehe § 306.

`InitialStep` Schrittweite im ersten Schritt, siehe § 307.

`MaxOrder` falls ODE15s verwendet, wird die maximale Ordnung des Solvers spezifiziert; sonst wird der Wert ignoriert.

`FixedStep` Schrittweite für einen Solver mit fester Schrittweite.

`OutputPoints` entweder `'specified'` oder `'all'`. In letzterem Fall werden alle Zeitschritte ausgegeben, in ersterem nur an den in `TimeSpan` definierten. Äquivalent zu `specified output only` oder `Produce additional output` in § 2.6.3.

`Outputvariables` überschreibt die drei typischerweise ausgegebenen und in § 293 beschriebenen Ausgabevariablen; die Variable wird durch einen String aus bis zu drei Zeichen definiert, der Kombinationen von `t`, `x` und `y` enthält, z.B. `tx`.

`MaxRows` limitiert die maximale Zeilenzahl der Ausgabe, entspricht `Limit to Last` in Abschn. 2.6.3.

`Decimation` positiver Integer; bestimmt, ob jeder (1) oder jeder `nte` (`n`) Zeitpunkt ausgegeben wird, vgl. Abschn. 2.6.3.

`InitialState` Vektor, der die Anfangswerte der Zustandsvariablen übergibt, vgl. Abschn. 2.6.3.

`FinalStateName` Name der Variablen im Workspace, in die die letzten Werte der Zustandsvariablen geschrieben werden können. Mit Hilfe dieses Vektors kann eine Simulation an dieser Stelle fortgesetzt werden.

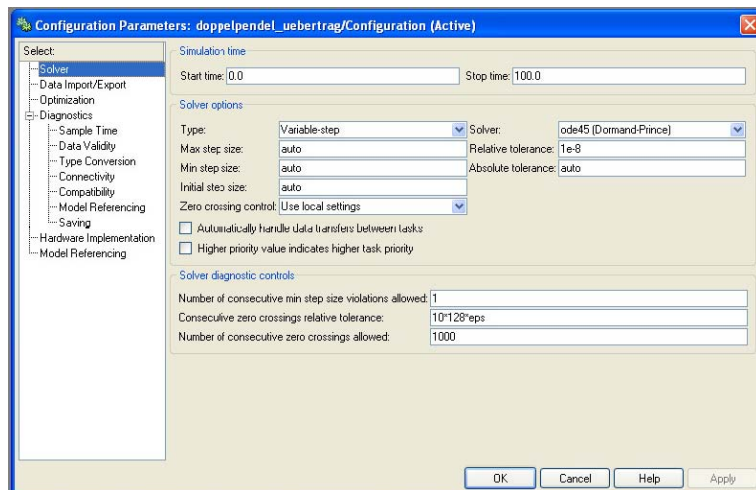


Abbildung 2.40:  
Simulationskontrolle

`Trace` erzeugt Systeminfos wie `Compilation Listing`; siehe `SIMULINK Hilfe`.

`SrcWorkspace` wählt den `MATLAB Workspace` aus, in dem die im Modell verwendeten `MATLAB`-Ausdrücke auszuwerten sind. Die Option ist besonders wichtig, wenn ein Modell aus einem `m-file`, d.h. einer `MATLAB`-Funktion, heraus läuft. `'base'` bezeichnet, dass alle Variablen der Simulation aus dem Basis-Workspace von `MATLAB` bezogen werden. `'current'` holt die Variablen aus dem persönlichen Workspace, aus dem heraus das Modell aufgerufen wurde; `'parent'` dagegen nimmt die Variablen aus dem Workspace, aus dem die momentan verwendete Funktion (also die, die das Modell aufgerufen hat) aufgerufen wurde. Auf den Simulationsparameter-Karten gibt es keine entsprechende Einstellmöglichkeit, da diese Frage ohnehin nur bei Steuerung aus dem Workspace heraus auftreten kann.

`DstWorkspace` ähnlich `SrcWorkspace` jedoch nicht bezogen auf den In- sondern den Output. `ZeroCross` verändert die `Disable zero crossing detection` Box.

§ 300 Den Wert einer bestimmten Variablen oder die Struktur der Optionen kann man `simget` erhalten:

```
opts = simget(model)
```

Der Wert einer bestimmten Eigenschaft lässt sich gezielt abfragen mit

```
value=simget(model,property_name)
```

mit den Eigenschaften wie in § 299.

## 2.6 Simulationskontrolle

§ 301 Bisher haben wir die Frage verdrängt (bzw. eine Angstschrittweite erzwungen): unter welchen Bedingungen ist das von `SIMULINK` verwendete Verfahren stabil und wie können wir die Genauigkeit beeinflussen? Werfen wir einen genaueren Blick auf die Karte `Simulation` ↔ `Configuration Parameters`; ein Beispiel ist in Abb. 2.40 gezeigt. Hier lassen sich verschiedene Bereiche vom Solver über Datenim- und -export, Optimierung und Diagnose bis hin zur Hardware Implementierung auswählen.

### 2.6.1 Solver

§ 302 Der größte Handlungsbedarf besteht bei der Wahl des Solvers. Alle Solver sind für `DGLs` erster Ordnung. Die meisten von ihnen dürften aus `MATLAB` bekannt sein:

- ODE45** allgemeines einstufiges Lösungsverfahren, basiert auf dem auf ein Runge–Kutta Verfahren vierter/fünfter Ordnung zurück greifendes Dorman–Prince Verfahren: die Funktion wird in jedem Integrationsintervall durch Polynome (Taylor-Entwicklung) entsprechender Ordnung angenähert. Die Kombination von zwei Ordnungen erlaubt die Überprüfung von Genauigkeit und Stabilität durch Vergleich. Ist der Default Solver; wenn keinen Gegenargumente bekannt sind, lohnt sich der Solver auf jeden Fall für den ersten Versuch.
- ODE23** Es ist schneller aber auch weniger genau als **ODE45**, d.h. seine Genauigkeit ist von nur niedriger Ordnung. Dennoch kann es für schnelle Abschätzungen sinnvoll sein. Bei einigen Differentialgleichungen kann **ODE23** Genauigkeiten liefern, die mit denen von **ode45** vergleichbar sind.
- ODE113** (Adams–Bashforth–Moulton Verfahren) ist im Gegensatz zu den beiden vorherigen Methoden kein einstufiges Verfahren sondern ein mehrstufiges: es basiert, ähnlich dem Euler Rückwärts Verfahren [63, 66], auf einem Prädiktor–Korrektor Schema. Außerdem ist es kein Verfahren fester Ordnung: die Ordnung des Polynoms wird unter Berücksichtigung von Genauigkeit und Rechenzeit automatisch angepasst. Die sich daraus ergebende höhere Genauigkeit und Stabilität erkaufen wir mit einem mehrstufigen Verfahren und entsprechend größerer Rechenzeit. Daher ist **ode113** nicht unbedingt das erste Verfahren mit dem man einer gewöhnlichen Differentialgleichung auf den Leib rücken sollte. Es ist aber ein Verfahren, das eventuell auch für DGLs eine Lösung liefert, am denen **ode45** scheitert.
- ODE15S** ist ebenfalls ein mehrstufiges Verfahren variabler Ordnung. Es ist, ebenso wie alle anderen auf S endenden Solver, speziell für steife Differentialgleichungen entwickelt. Die Bezeichnung stammt (sorry, es ist ein sehr strapaziertes Beispiel) vom (gedämpften) Federpendel mit einer sehr steifen Feder. In diesem Fall ergibt sich die Überlagerung einer sehr schnellen Dynamik (die Schwingung) mit einer sehr langsamen Dynamik (dem Abklingen der Amplitude auf Grund der Dämpfung). Die Kombination der unterschiedlichen Zeitkonstanten ist mit einem hohen Rechenaufwand verbunden, so dass Solver für steife DGLs in der Regel für nicht-steife Probleme ineffizient sind.
- ODE23S** basiert auf dem Rosenbrock Algorithmus; nur für steife DGLs.
- ODE23T** Trapez-Integration einer steifen DGL.
- ODE23TB** wie **ODE23T** aber mit anderem Verfahren zur Bestimmung des Rundungsfehlers. Etwas langsamer aber dafür etwas stabiler als **ODE23T**.
- Discrete** spezielles Lösungsverfahren für Systeme ohne kontinuierliche Zustände; zwar sind alle anderen Solver ebenfalls auf derartige Systeme anwendbar, dieser ist aber der schnellste.
- ODE5** Dormand–Prince Verfahren wie in **ODE45**, jedoch mit fester Schrittweite
- ODE4** ganz klassisches Runge–Kutta Verfahren vierter Ordnung (siehe z.B. [66, 63], Kurzfassung in Abschn. 10.2.1), per definitionem mit fester Schrittweite
- ODE3** Bogacki–Shampine Verfahren wie in **ODE23** aber mit fester Schrittweite
- ODE2** ebenfalls ein Runge–Kutta Verfahren mit fester Schrittweite aber nur von zweiter Ordnung; auch als Heun’s Verfahren bezeichnet.
- ODE1** Euler Verfahren – da weiß man zumindest, was man macht. Feste Schrittweite sowieso.
- ODE14x** impliziter Solver mit fester Schrittweite und schlechter Dokumentation; kann in verschiedenen Ordnungen spezifiziert werden. Alle anderen Solver mit fester Schrittweite sind explizite Verfahren.

Die Auswahl ist auf der Karte eingeschränkt: ist im linken Bereich eine variable Schrittweite eingestellt, so bietet **SIMULINK** auch nur Solver mit variabler Schrittweite an (**ODE45** bis **Discrete**). Ist dagegen eine feste Schrittweite gewählt, so werden auch nur die entsprechenden Solver (**Discrete** bis **ODE14x**) zur Auswahl gestellt. Die Details der einzelnen Solver sind in [51] und [141], teilweise auch in [142], gegeben.

**Zwischenrechnung 15** Betrachten Sie ein Federpendel (gedämpft, kein externer Antrieb) beschrieben durch

$$\ddot{x} + 100\dot{x} + 0.9999x = 0$$

mit den Anfangsbedingungen  $x(0) = 0$  und  $\dot{x}(0) = 0$ . Laplace Transformation ergibt

$$s^2 X(s) - s x(0) - \dot{x}(0) + 100(sX(s) - x(0)) + 0.9999X(s) = 0.$$

Einsetzen der Anfangsbedingungen und Auflösen nach  $X(s)$  liefert

$$X(s) = \frac{s + 100}{s^2 + 100s + 0.9999}.$$

Rücktransformation liefert

$$x(t) = -0.0001 e^{-99.99t} + 1.0001 e^{-0.01t}.$$

Der erste Term beginnt bei kleiner Amplitude und zerfällt sehr schnell, der zweite Term beginnt bei einer 4 Größenordnungen höheren Amplitude und zerfällt langsam. Damit dominiert er das Verhalten. Untersuchen Sie das Verhalten der Lösung für verschiedene Solver, insbesondere einem Solver für steife DGLs, z.B. ODE15s und dem Standardsolver ODE45. Achten Sie dabei auf Laufzeit, Größe des Outputs (Länge des Zeitvektors) und eventuelle Schwingungen (Ausschnittvergrößerungen!).

§ **303** Grundsätzlich gibt es keinen besten Solver – lediglich für eine bestimmte Aufgabe kann der eine oder andere Solver besser geeignet sein. Um einen Solver vernünftig auszuwählen, müssen wir also auch die Dynamik des Systems verstehen.

### Feste oder variable Schrittweite?

§ **304** Feste Schrittweite ist in jedem Fall eine langsame (und bei ungeschickter Wahl der Schrittweite auch Fehler anfällige) Variante, vgl. Abb. 1.4. Die Verwendung eines Lösungsverfahrens fester Schrittweite ist gerechtfertigt, wenn man die Vorzüge der genauen formalen Beschreibung von Stabilitätskriterien und Abbruchfehlern ausnutzen möchte (siehe auch Abschn. 10.2.2). Kann man sich mit derartigen Fragen beschäftigen, so ist in der Regel die korrekte Wahl des Zeitschritts kein Problem mehr. Außerdem ist die feste Schrittweite bei allen Echtzeit-Applikationen sinnvoll, da sich sonst kein gemeinsamer Takt mit der Außenwelt finden lässt. Das ist der wesentliche Grund dafür, dass SIMULINK im Gegensatz zu MATLAB die verschiedenen Solver mit fester Schrittweite sehr offensiv zur Verfügung stellt.

§ **305** Wählen wir einen Solver variabler Schrittweite, so liefert die Karte vier Felder zur Kontrolle der Schrittweite, vgl. Abb. 2.40. Die beiden Felder **Max Step Size** und **Initial step Size** erlauben es, das Risiko zu reduzieren, dass der Solver wichtige Aspekte des Systemverhaltens übersieht.

§ **306** Die maximale Schrittweite ist als Default so definiert, dass das zu simulierende Zeitintervall in 50 Schritte unterteilt wird:

$$h_{\max} = \frac{t_{\text{stop}} - t_{\text{start}}}{50}.$$

Ein Federpendel mit einer Frequenz von 1 Hz lässt sich damit über eine Simulationsdauer von 10 s noch recht vernünftig (wenn auch etwas zackig) betrachten – über eine Simulationsdauer von 200 s ist die maximale Schrittweite aber so groß, dass der Solver u.U. einen wichtigen Aspekt des Systems, nämlich die Schwingung, übersieht. Für eine periodische Bewegung schlägt MATHWORKS vor, die maximale Schrittweite auf 1/4 der Frequenz zu setzen.

§ **307** Ein Solver variabler Schrittweite sucht sich seine Schrittweite aus der (bekannten) Ableitung. Für den ersten Schritt gibt es keine ‘Vergangenheit’, d.h. SIMULINK kann nur auf der Basis der Anfangswerte der Zustandsgrößen und der Ableitungen in diesen Punkten arbeiten – ohne die für die Anpassung der Schrittweite wichtige Änderung der Ableitung bewerten zu können. Daher kann insbesondere im ersten Schritt wichtiges Systemverhalten übersehen werden. Gibt es einen Grund zu der Annahme, dass sich das System kurz nach Start der Simulation zur Zeit  $t_{\text{Start}}$  deutlich verändert, so sollte der Parameter **Initial step Size** auf einen entsprechend kleinen Wert gesetzt werden. Besteht dieser Verdacht nicht, so



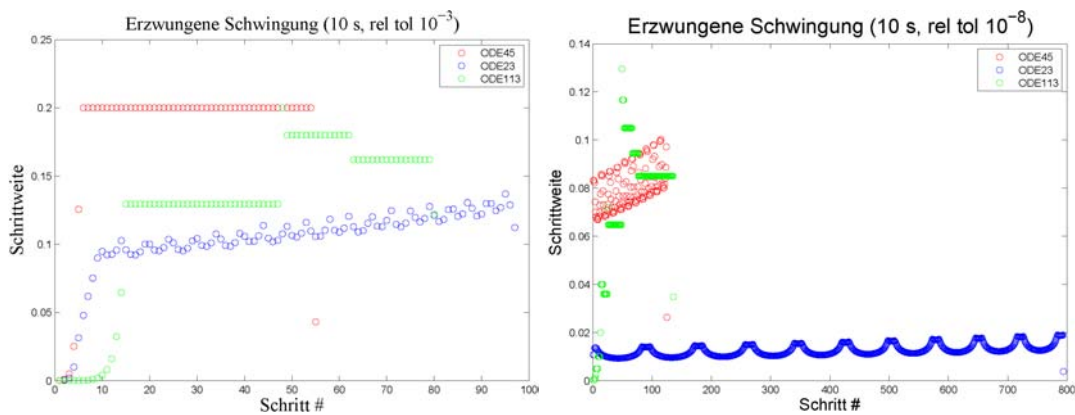


Abbildung 2.41: *Links*: Wie Abb. 2.39 für drei verschiedene Solver und eine relative Toleranz von  $10^{-3}$ ; *Rechts*: wie links aber für eine relative Toleranz von  $10^{-8}$ ; für ODE45 vgl. auch Abb. 2.39

kann man sich auf den Solver verlassen und diesen Parameter auf auto stellen (Abb. 2.41 deutet an, dass SIMULINK ohne weitere Anweisung mit einer relativ kleinen Schrittweite startet).

§ 308 Die beiden anderen Parameter, **Relative Tolerance** und **Absolute Tolerance**, erlauben es, für jeden Zustand  $x_i$  den zulässigen Wert  $e_i$  des Integrationsfehlers zu bestimmen gemäß

$$e_i \leq \max(t_{\text{rel}}|x_i|, t_{\text{abs}}) .$$

Ist die Zustandsgröße  $x_i$  groß, so bestimmt die relative Toleranz den Integrationsfehler; für kleine Werte der Zustandsgrößen ist die absolute Toleranz maßgeblich.

**Verständnisfrage 27** Begründen Sie anschaulich, warum es sinnvoll ist, bei kleinen Werten eine absolute statt der relativen Toleranz zu verwenden und umgekehrt.

§ 309 Das, was auf dieser Karte an Parametern verändert werden kann, wird bei Verwendung des entsprechenden Solvers aus MATLAB heraus mit der Funktion `odeset` verändert.

### Beispiele variable Schrittweite

§ 310 In Abb. 2.41 sind im linken Teil die von den Solvern ODE45, ODE23 und ODE113 verwendeten Schrittweiten für die Simulation der gedämpften Schwingung mit den Parametern wie in Abb. 2.39 bei einer relativen Toleranz von  $10^{-3}$  (Default) gegeben. ODE45 (rot) kommt mit der geringsten Zahl an Schritten aus und verwendet fast durchgängig die maximal mögliche Schrittweite – der Graph ist auf Grund der großen Schrittweite im Vergleich zur Schwingungsperiode entsprechend zackig.

§ 311 ODE23 (blaue Symbole) arbeitet mit einer kleineren Schrittweite und benötigt entsprechend eine größere Zahl von Schritten. Hier erkennt man, ähnlich dem Ergebnis in Abb. 2.39, die häufige Anpassung der Schrittweite: der Solver beginnt mit einer kleinen Schrittweite, da noch keine Informationen über die Änderung der Steigung vorliegen.<sup>16</sup> Diese wird vergrößert bis sie in Abhängigkeit von der Phase um einen langsam ansteigenden Wert zu schwingen beginnt. Der langsame mittlere Anstieg ist mit der Dämpfung verbunden: da die Amplitude der Schwingung abnimmt, reduziert sich auch die Steigung und damit die Änderung der Steigung; daher kann die gleiche Genauigkeit mit einer etwas größeren Schrittweite erreicht werden. Die maximale Schrittweite wird jedoch innerhalb des Simulationsintervalls nicht erreicht. In der

<sup>16</sup>Der Anfangswert der Schrittweite ist übrigens in allen drei Solvern in Abb. 2.41 gleich; die weitere Entwicklung der Schrittweite und damit die Geschwindigkeit der Anpassung des Solvers ist jedoch unterschiedlich.

graphischen Darstellung ist die Lösung auf Grund der größeren Zahl von Schritten weniger zackig.

§ 312 Das implizite Verfahren mit Prädikatorschritt in ODE113 hat über längere Zeiten konstante Schrittweiten, erreicht aber die maximale Schrittweite nicht. Auf Grund der relativ großen Schrittzahl (im Vergleich zu ODE45) und der langen Rechenzeit pro Schritt (im Vergleich zu ODE45 und noch mehr im Vergleich zu ODE23) ist das Verfahren langsam – und der optische Eindruck ist immer noch recht zackig.<sup>17</sup>

§ 313 Weniger zackige Kurven lassen sich durch Erhöhung der relativen Toleranz oder durch Reduktion der maximalen Schrittweite erzeugen. Im rechten Teil von Abb. 2.41 ist eine relative Toleranz von  $10^{-8}$  gewählt. Die Schrittzahlen in ODE45 und ODE113 sind vergleichbar, ODE23 braucht eine mehr als fünfmal so große Schrittzahl. In ODE23 (und ebenso in ODE45 in der Darstellung wie in Abb. 2.39) wird die schnelle Variation der Schrittweite mit der Schwingung deutlich. ODE113 zeigt auch hier über längere Zeiten konstante Schrittweiten – diese werden offenbar an der Steigung der Hüllkurve und weniger an der Frequenz ausgerichtet.

§ 314 Ähnliche Verhältnisse zwischen den einzelnen Solvern im rechten Teil von Abb. 2.41 ergeben sich, wenn man statt der relativen Toleranz von  $10^{-8}$  den Default-Wert von  $10^{-3}$  verwendet und die maximale Schrittweite mit 0.1 begrenzt.

## 2.6.2 Nullstellendetektion

§ 315 Auf der Karte zur Kontrolle der Solver in Abb. 2.40 können außerdem Einstellungen zur Behandlung von Nullstellen vorgenommen werden. Diese können z.B. zur Stroboskop-Darstellung von Synchronisation genutzt werden. Nullstellendetektion ist jedoch auch bei einem einfachen mechanischen Problem wie z.B. bei einem springenden Ball interessant: dieser soll beim Auftreffen auf den Boden detektiert werden, d.h. die Höhe  $x = 0$  muss erkannt werden. Egal ob Solver mit konstanter oder variabler Schrittweite verwendet werden: es ist nicht zu erwarten, dass der Solver genau beim Auftreffen des Balls auf den Boden eine Stützstelle in der Zeit hat. In der Regel wird der Solver das Auftreffen des Balls erst entdecken, wenn dieser in der Lösung bereits ein Stückchen in den Boden eingedrungen ist. Die Detektion von Nullstellen erlaubt es dem Solver, sich entsprechend anzupassen und so den Ball vor seinem Eindringen in den Boden korrekt zu reflektieren.

### Springender Ball

§ 316 Ein Ball wird aus der Höhe  $y(0) = h$  fallen gelassen. Beim Aufprall auf den Boden wird er elastisch reflektiert, die Abstoßgeschwindigkeit ist proportional der Aufprallgeschwindigkeit mit einem Proportionalitätsfaktor -0.8. Die Differentialgleichung der Bewegung ist

$$\ddot{y} = -g \quad \text{mit} \quad y(0) = h \quad \text{und} \quad \dot{y}(0) = 0.$$

Diese Kombination aus Bewegungsgleichung und Anfangsbedingungen gilt bis der Ball zur Zeit  $t_R$  auf den Boden trifft; dort ist  $y = 0$ . Dann gilt zwar die gleiche Bewegungsgleichung, jedoch mit den neuen Anfangsbedingungen  $y(t_R) = 0$  und  $\dot{y}(t_R) = -0.8v_{\text{Aufprall}}$ .

§ 317 Zur Steuerung der in Abb. 2.42 gezeigten Simulation ist auf der Karte Simulationskontrolle im Feld ‘zero crossings control’ die Einstellung ‘use local settings’ aktiviert. Integrator 2 liefert als Ergebnis die Geschwindigkeit. Bei der Einstellung der Parameter des Blocks wird das Feld Limit Output aktiviert: dann lassen sich ein oberer und ein unterer Sättigungspunkt

<sup>17</sup>Das Argument der Zackigkeit des Graphen als numerisches Artefakt ist zwar anschaulich aber eigentlich bedeutungslos. Zum einen sagt eine mehr oder weniger deutliche Zackigkeit nichts über die Genauigkeit des Solvers aus. Zum anderen ist SIMULINK in der Lage, einerseits mit dem Solver variabler Schrittweite zu arbeiten, andererseits aber auch eine Ausgabe zu vordefinierten Zeiten und insbesondere auf einem äquidistanten Zeitvektor vorzunehmen, vgl. § 295. Daher ist auch der Wunsch nach einer äquidistanten Zeitachse mit vorgegebener Teilung kein Grund für die Verwendung eines Solvers mit fester Schrittweite.

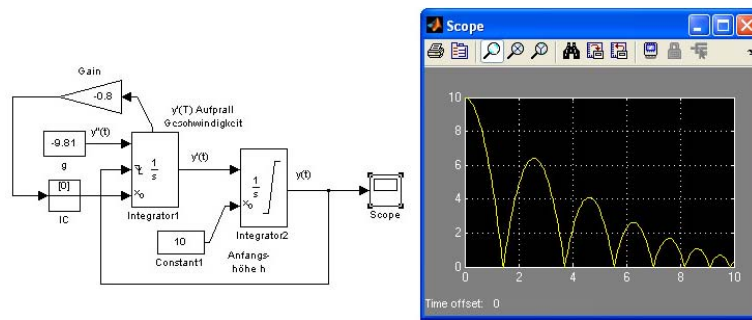


Abbildung 2.42: Springender Ball in SIMULINK, basierend auf [44]

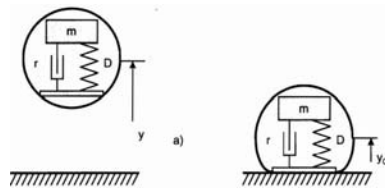


Abbildung 2.43: Modell eines elastischen Balls [44]

eingeben. In diesem Fall ist der untere Sättigungspunkt Null relevant, der obere kann einen beliebigen Wert (auch  $\infty$  für unendlich) annehmen. Außerdem ist auf dieser Karte die Quelle für die Anfangsbedingung als extern eingestellt. Diese wird durch einen konstanten Block von außen vorgegeben.

§ 318 Für den ersten Integrator benötigen wir keine Einstellung der Sättigungspunkte. Dieser Integrator hat auf der linken Seite des Blocks drei Eingänge: der obere Eingang für die Eingangsgröße, in unserem Beispiel die konstante Gravitationsbeschleunigung. Der zweite Eingang dient der Rücksetzung. Dies erfolgt genau dann, wenn der erste Integrator eine Null sendet, d.h. wenn er seinen unteren Sättigungspunkt erreicht hat. Der Integrator wird dann von der Anfangsbedingung, die an seinem unteren Eingang ebenfalls von außen anliegt, wieder neu gestartet. Dieser Eingang wird aus dem mit der Option 'show state port' aktivierten Ausgang über einen Gain Block mit dem Faktor -0.8 und einem IC oder `initial condition` Block auf den Eingang für die Anfangsbedingung geführt. Der IC Block ist anfangs auf Null gesetzt (verschwindende Anfangsgeschwindigkeit bei Beginn der Bewegung). Die Verwendung des State Port anstelle des normalen Ausgangs ist mathematisch irrelevant, da beide Größen identisch sind. Allerdings wird durch den State Port die sonst auftretende algebraische Schleife unterbrochen.

### Elastischer Ball

§ 319 Wir können das Beispiel noch etwas komplexer gestalten und dabei den Block `Switch` einsetzen, wie in Hoffmann und Brunner [44] dargestellt. Ein elastischer Ball lässt sich als ein Feder-Masse System darstellen, vgl. Abb. 2.43.

§ 320 In unserem Modell soll die Bewegung in der Luft reibungsfrei erfolgen. Die Bewegungsgleichung ist

$$m\ddot{y} = -mg \quad \text{mit} \quad y(0) = h \quad \text{und} \quad \dot{y}(0) = 0.$$

Besteht dagegen Kontakt mit dem Boden ( $y \leq 0$ ), so ist die Bewegung beschrieben durch die Differentialgleichung

$$m\ddot{y} + \beta\dot{y} + ky = 0.$$

Die Anfangswerte sind durch die Endwerte des freien Falls gegeben.

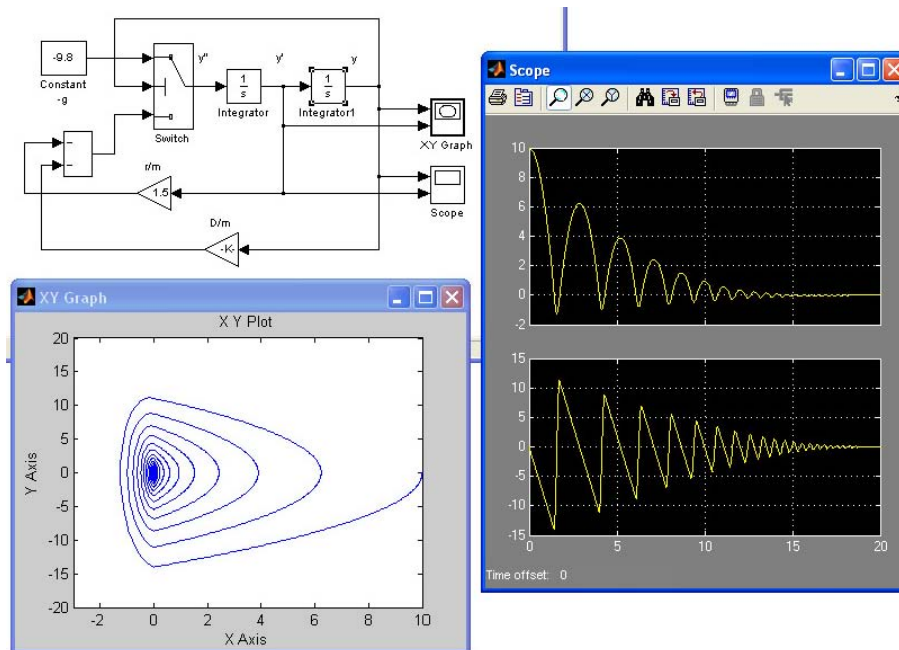


Abbildung 2.44: Elastischer Ball als Feder–Masse System in SIMULINK, basierend auf dem Modell aus [44]

§ 321 Das zugehörige Modell ist in Abb. 2.44 gegeben. Der Switch-Block schaltet zwischen dem freien Fall und dem Bodenkontakt. Da in beiden Bewegungsabschnitten die gleichen Integratoren verwendet werden, sind die Anfangsbedingungen automatisch korrekt und müssen nicht gesondert übergeben werden. Für  $y > 0$  steht der Switch auf dem oberen Eingang; das Modell beschreibt den freien Fall des Balls. Wird  $y \leq 0$ , so steht der Switch auf dem unteren Eingang und das Modell beschreibt die Differentialgleichung des elastischen Balls – wobei die Integratoren beim ersten Schritt für den elastischen Ball noch genau die Endwerte des freien Falls als Anfangsbedingungen haben.

**Zwischenrechnung 16** Verändern Sie das Modell so, dass zusätzlich die Reibung in der Luft berücksichtigt wird.

**Zwischenrechnung 17** Da der Ball sich in zwei unterschiedlichen Modi bewegt, wäre eine Simulation mit Hilfe eines bedingten Untersystems möglicherweise sinnvoll. Konstruieren Sie ein entsprechendes Modell und untersuchen Sie Vor- und Nachteile gegenüber der Realisierung in Abb. 2.44.

### Diskontinuierliches System: Idealisierte (Zener-)Diode

§ 322 Am Beispiel des elastischen Balls haben wir SIMULINK in einem System eingesetzt, dass zwischen zwei Zuständen, dem freien Fall und der Kompression des Balls, wechselt. Dazu haben wir den Schalter verwendet. Für derartige nicht kontinuierliche Systeme stehen in der **Discontinuities Block Library** weitere Hilfsmittel zur Verfügung.

§ 323 Der Block **Dead Zone** z.B. erlaubt die Simulation einer idealisierten Zener-Diode. Diese sperrt im Durchlassbereich unterhalb der Schwellspannung von 0.7 V, für höhere Spannungen fließt in unserem idealisierten Modell ein der anliegenden Spannung minus der Schwellspannung proportionaler Strom. Diesen betrachten wir als Ausgangsgröße.<sup>18</sup> Im Sperrbereich sperrt die Zener-Diode bis zum Erreichen der Zenerspannung – dann wird sie auch

<sup>18</sup>Statt des Stromes können wir auch den Spannungsabfall über dem in Serie geschalteten Widerstand verwenden.

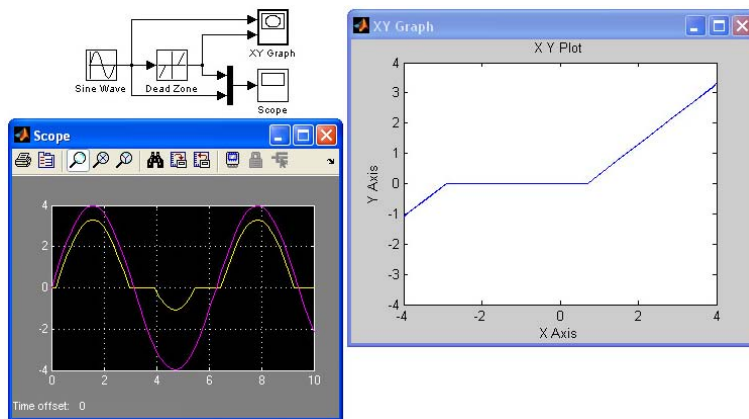


Abbildung 2.45: Idealisierte Zener-Diode simuliert mit dem Dead Zone Block

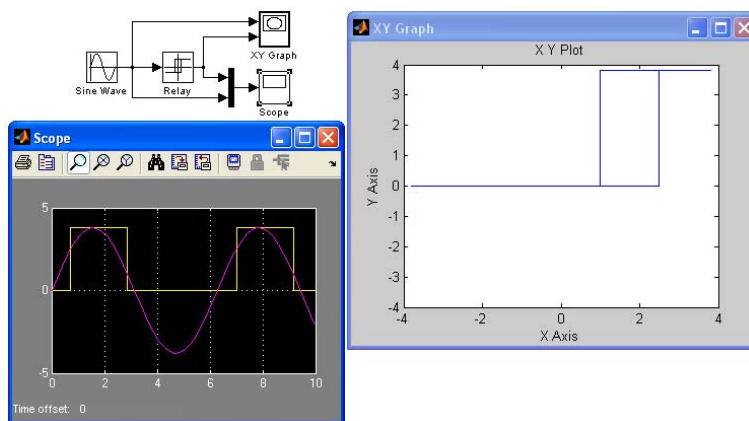


Abbildung 2.46: Schmitt Trigger simuliert mit dem Relay Block

dort leitfähig mit dem Strom proportional der Differenz zwischen anliegender Spannung und Zenerspannung. Im Block `Dead Zone` müssen wir nur diese beiden Spannungen definieren: zwischen den beiden Spannungen wird der anliegende Funktionswert auf Null gesetzt, im Bereich außerhalb werden die Signale entsprechend der Werte an den Kanten der `Dead Zone` versetzt.

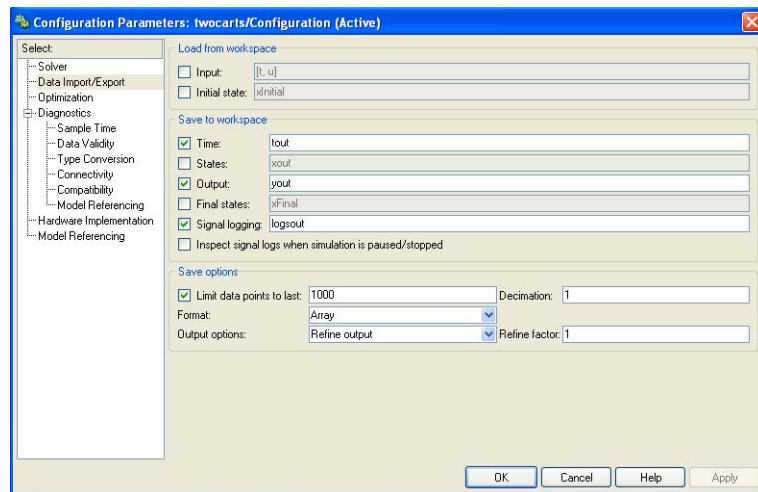
§ 324 Abbildung 2.45 gibt das zugehörige Modell in `SIMULINK` für eine harmonische Eingangsgröße, einen  $xy$ -Plot von Ausgangs- gegen Eingangsspannung sowie den zeitlichen Verlauf von Ein- (magenta) und Ausgangssignal (gelb). Der  $xy$ -Plot veranschaulicht noch einmal die Funktion und die Grenzen des Blocks `Dead Zone`, die Spannungsverläufe auf dem Oszilloskop entsprechen den Erwartungen.

**Verständnisfrage 28** Müsste der  $xy$ -Plot nicht eigentlich die Kennlinie der Diode wiedergeben? Ist die hier gegebene Darstellung sinnvoll oder lassen sich noch Verbesserungen vornehmen, um zu einem besseren Diodenmodell zu kommen?

### Diskontinuierliches System: Schmitt-Trigger

§ 325 Den berühmten berüchtigten Schmitt-Trigger können wir mit dem Block `Relay` aus gleicher Bibliothek ebenfalls simulieren. Der Block ist für alle Elemente mit einer Hysterese gedacht. Der Schmitt-Trigger ist eines der einfachsten Beispiele dafür: zwei verschiedene Schwellen für die Eingangssignale werden definiert. Oberhalb einer Schwelle wird angeschaltet, unterhalb der anderen Schwelle ausgeschaltet. Eingangsgrößen zwischen den beiden Schwellen führen nicht zum Schalten. Ein typisches Anwendungsbeispiel ist ein Dämmerungsschalter. Schaltet dieser nur auf der Basis des direkt anliegenden Signals, so führen Fluktuationen im Signal insbesondere während der Dämmerung zu einem steten An- und Ausschalten. Um dieses abzufangen, wird eine Schwelle zum An- und eine zweite zum Ausschalten definiert.

Abbildung 2.47: Simulationskontrolle: auf der Karte DataIm- und Export lassen sich die Details des Austausches mit dem Workspace angeben



§ 326 Abbildung 2.46 zeigt das entsprechende SIMULINK Modell zusammen mit der Hysteresekurve im  $xy$ -Diagramm und dem zeitlichen Verlauf eines harmonischen Eingangssignals (magenta) sowie des Ausgangssignals (gelb). Bei Überschreiten einer Eingangsspannung von 2.5 V schaltet der Schmitt-Trigger in den Arbeitszustand, bei Unterschreiten von 1 V schaltet er zurück.

### 2.6.3 Datenim- und -export

§ 327 Zur Steuerung des Datenaustauschs mit dem Workspace bietet in der Simulationskontrolle die Karte `DataImport/Export`, siehe auch Abb. 2.47, die Möglichkeit. Diese Karte beinhaltet auch die bereits bei der Steuerung aus dem Workspace heraus in § 293 angesprochenen Variablen sowie das Übergeben der Variablen an den Workspace.

§ 328 Neu ist der letzte Block von Eingabegrößen, die `Save Options`. Diese sind insbesondere bei längeren Simulationen wichtig, um die Datenmengen des Outputs überhaupt noch behandeln zu können. Die erste Option, die Begrenzung auf die letzten  $n$  Datenpunkte ist Ihnen wahrscheinlich schon beim Spielen mit der Zeitauflösung in einem ganz einfachen Modell aufgefallen: verfeinert man z.B. beim harmonischen Oszillator in Abb. 1.6 die Zeitauflösung, so wird ab einem bestimmten Punkt im Oszilloskop nicht mehr das gesamte Zeitintervall dargestellt sondern nur noch der letzte Teil. Dessen Länge wird bei der Übergabe an den Workspace mit dem  $n$  spezifiziert; bei der Ausgabe im Scope wird dieser Wert für das Scope auf der entsprechenden Parameterkarte nochmals separat angegeben. Dazu wird die entsprechende Länge auf der Data History Karte (siehe auch Abb. 2.11) eingetragen.

§ 329 Das nächste Feld, `Decimation`, erfordert die Eingabe einer natürlichen Zahl  $m$ . Es definiert, dass nur jeder  $m$ te-Datenpunkte an die Ausgabe übergeben werden soll; alle Zeitpunkte werden bei der Einstellung  $m = 1$  zurück gegeben. Im Feld `Format` lässt sich, wie in § 293 angesprochen zwischen `Array` und strukturiertem Element (gegebenenfalls mit Zeit) auswählen.

§ 330 Die beiden unteren Felder erlauben die Erzeugung von Daten auch zwischen den Zeitschritten der Simulation oder an spezifizierten Zeitpunkten. Details finden Sie in der SIMULINK-Hilfe. Die Einstellung `Refine Output` basiert zumindest bei der Verwendung von `ode45` als Solver nicht auf einer simplen Interpolation<sup>19</sup> sondern wirkt direkt auf den Solver zurück. Das gleiche Ergebnis kann man im Prinzip auch durch die Definition geeigneter Schrittweiten und Genauigkeiten in der Simulationskontrolle erreichen – letztere Variante ist

<sup>19</sup>Eine Interpolation erzeugt keine zusätzliche Information sondern fügt nur Werte ein, um eine glattere Darstellung zu ermöglichen.

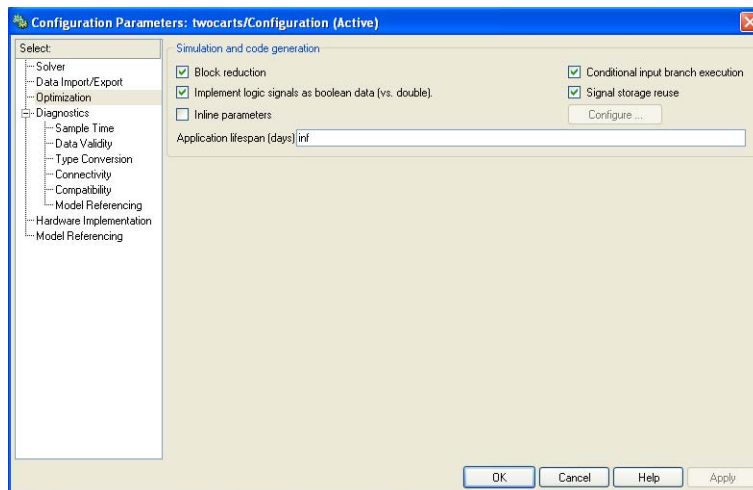


Abbildung 2.48: Simulationskontrolle: Optimierung erfolgt in SIMULINK Defaultmäßig

kontrollierter, erfordert aber mehr Auseinandersetzung mit dem Solver. Die Angabe spezifizierter Zeitschritte hilft z.B. beim Auffinden von Nullstellen. Die Angabe der spezifizierten Zeitschritte wirkt ebenfalls auf den Solver zurück: die Zeitauflösung bzw. Schrittweite wird so angepasst, dass die spezifizierten Punkte getroffen werden. Eine Interpolation erfolgt nicht.

### 2.6.4 Optimierung

§ **331** Zur Optimierung der Simulation können mit Hilfe der in Abb. 2.48 gezeigten Karte Einstellungen vorgenommen werden. Die Optimierung erfolgt unter zwei Aspekten: zum einen wird die Simulation selbst optimiert, zum anderen wird der zur Ausführung der Simulation erstellte Code optimiert.

§ **332** Zur Optimierung der Simulation gehören unter anderem

- **Block Reduction.** SIMULINK entfernt als erstes den sogenannten toten Code (Dead Code), d.h. alle Blöcke und Signale in nicht genutzten Signalpfaden.<sup>20</sup> Außerdem werden Redundanzen in der Konversion von Datentypen entfernt sowie bestimmte Gruppen von Blöcken zu einem größeren Block zusammen gefasst. Die Erscheinung des Modells wird von SIMULINK nicht verändert!
- **Conditional Input Branch Execution** verbessert die Ausführung in Modellen, die einen Switch oder einen Multiport Switch enthalten.
- **Inline Parameters** überführt veränderbare Parameter in Konstanten. Dadurch verbessert sich die Nutzung des RAM.
- **Implement Logic Signals as Boolean Data (vs. Double)** hilft bei der Fehlerdiagnose von gemischten Datentypen aus den Bereichen Boolean and Double.
- **Signal Storage Reuse** verbessert die Ausnutzung des Speichers.

### 2.6.5 Diagnostik

§ **333** SIMULINK verfügt über eine Vielzahl von Hilfsmitteln zur genaueren Diagnose eines Modells. Diese sind auf verschiedenen Karten zusammengefasst:

- Diagnostics (Solver),
- Sample Time,
- Data Validity,
- Type Conversion,
- Connectivity,
- Compatibility,

<sup>20</sup>Das ist eine recht offensichtliche Reduktion, die eigentlich in einem guten Modell nicht nötig sein sollte.

Abbildung 2.49: Simulationskontrolle: Diagnostik erlaubt die Einstellung etlicher Warnungen und Fehlermeldungen für Solver, Datentypen, Schrittweiten und vieles mehr

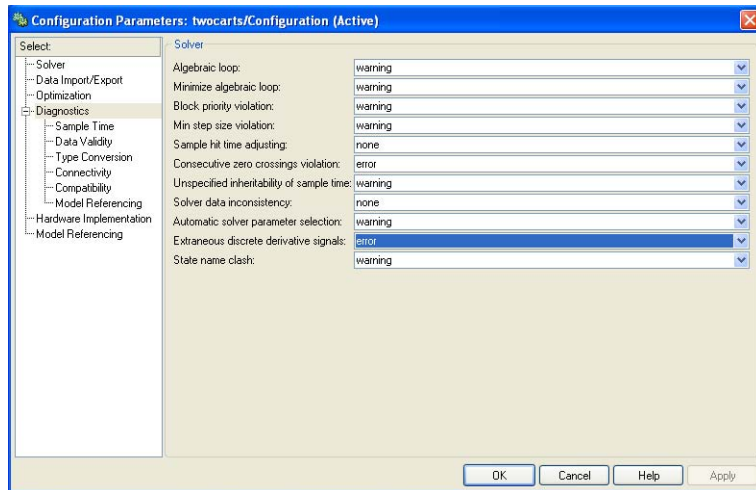
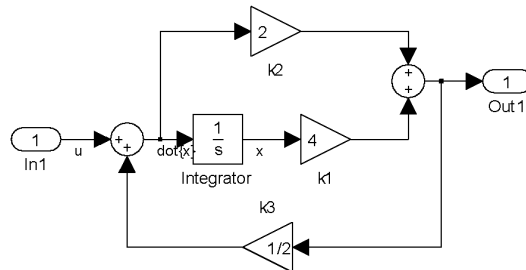


Abbildung 2.50: Modell zur Lösung DGL erster Ordnung mit algebraischer Schleife:  $\dot{x} = u - k_2 k_3 \dot{x} - k_1 k_3 x$



- Model Referencing,
- Saving.

Nahezu alle Eingaben bestehen aus einem Pulldown-Menü mit zwei oder drei der Optionen none, warning und error. Mit den ersten beiden Optionen wird die Simulation durchgeführt (bei none ohne eine Überprüfung!), bei der letzteren kommt es gegebenenfalls zum Abbruch. Bei Abbruch gibt SIMULINK eine entsprechende Fehlermeldung und, wenn möglich, im Modell durch Hervorhebung der betroffenen Blöcke, einen Hinweis auf den Fehler verursachenden Block.

§ 334 Die mit den Solvern befasste Eingangskarte zur Diagnostik ist die wichtigste. Die ersten beiden Einträge beziehen sich auch algebraische Schleifen (vgl. Abschn. 2.6.6) – einem häufig auftretenden, recht viel Rechenzeit verschwendendem Problem. Die Warnung über die Unterschreitung der minimalen Schrittweite sollte man ernst nehmen: eine spezifizierte minimale Schrittweite wird nur dann unterschritten, wenn die geforderte Genauigkeit eine geringere Schrittweite erfordert. Ist dies der Fall, so sollte man die Einstellungen des Solvers überprüfen. Für die weiteren Einträge auf dieser Karte und die Einträge auf den anderen Karten bitte die SIMULINK-Hilfe verwenden.

## 2.6.6 Direkte Rückkopplung: algebraische Schleifen

§ 335 Jeder Block, dessen Ausgang vollständig oder teilweise direkt auf seinen Eingang zurück gekoppelt wird, bildet eine algebraische Schleife.<sup>21</sup> Die Behandlung dieser Schleifen in SIMULINK erfordert spezielle Aufmerksamkeit.

<sup>21</sup>In SIMULINK beachten wir die algebraischen Schleifen, weil sie ein Störfaktor sind – umgekehrt sind algebraische Schleifen die mathematische Grundlage für das Chaos und bescheren uns so schöne Bildchen wie Mandelbrot's Apfelmännchen, vgl. Abschn. 6.1.



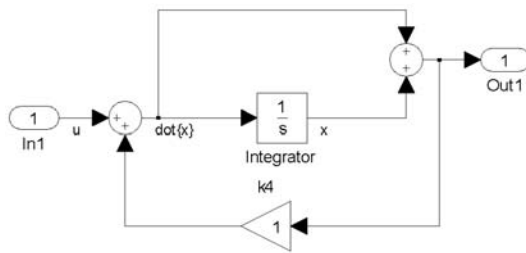


Abbildung 2.51: Das System erster Ordnung aus Abb. 2.50 umformuliert, so dass es keine algebraische Schleife mehr enthält

§ **336** SIMULINK ist ein numerisches System. Daher kann es einen Ausdruck der Form

$$\dot{x} = u - k_2 k_3 \dot{x} - k_1 k_3 x \quad (2.4)$$

zwar wie in Abb. 2.50 als Modell behandeln, nicht jedoch algebraisch auflösen, um die Rückkopplung des Ausgangs auf den Eingang zu vermeiden.

§ **337** Mathematisch handelt es sich bei (2.4) um eine Gleichung, in der  $\dot{x}$  eine Funktion von  $u$ ,  $x$  und  $\dot{x}$  ist:  $\dot{x} = \dot{x}(x, u, \dot{x})$ . Da SIMULINK keine algebraischen sondern nur numerische Fertigkeiten besitzt, kann diese Gleichung nur iterativ gelöst werden. Das hat zwei Nachteile: (a) iterative Lösungen benötigen Zeit und (b) iterative Verfahren finden nicht zwingend die (oder alle) Lösungen.

§ **338** Im Beispiel in Abb. 2.50 wird die Schleife aus den Gain Blöcken **k2** und **k3** sowie den beiden Summenblöcken gebildet. Algebraische Schleifen sind jedoch nicht auf diese beiden Typen von Blöcken beschränkt: jeder Block, bei dem der momentane Output direkt oder teilweise auf den momentanen Input zurück wirkt, kann Teil einer algebraischen Schleife sein. Das gilt für alle Blöcke aus der **Math Operations** Block Library. Das gilt auch für **Transfer Function**-Blöcke, in denen der Grad von Nenner- und Zählerpolynom identisch ist sowie für **State Space**-Blöcke mit nicht verschwindender Transmissionsmatrix  $D$ .

**Verständnisfrage 29** Treten in den bisher behandelten Modellen algebraische Schleifen auf? Und wenn ja, in welchen?

§ **339** SIMULINK behandelt die algebraische Schleife mit Hilfe einer robusten Version der Newton–Raphson Methode (siehe auch Abschn. 10.4). Für ein skalares Problem kann die algebraische Schleife dargestellt werden als  $\dot{x} = f(x, \dot{x}, u, t)$ , d.h. die Newton–Raphson Methode wird eingesetzt, um die Nullstelle der Funktion  $\Phi = \dot{x} - f(x, \dot{x}, u, t)$  zu suchen. Ist  $\partial\Phi/\partial\dot{x}$  eine Konstante, so ist  $f(x, \dot{x}, u, t)$  linear im Bezug auf  $\dot{x}$  und das Verfahren konvergiert nach einem Schritt. Ist  $\partial\Phi/\partial\dot{x}$  nicht konstant, so benötigt das Verfahren u.U. viele Iterationen oder konvergiert überhaupt nicht. In letzterem Fall ist die Lösung instabil – die Lösung, nicht zwingend das System. In ersterem Fall findet SIMULINK eine Lösung, allerdings geht u.U. viel Zeit verloren.

§ **340** Wenn möglich sollte man daher algebraische Schleifen in einem SIMULINK Modell vermeiden. Mit Hilfe der **Diagnostics**-Karte der Simulationskontrolle lässt sich eine automatische Warnung oder Fehlermeldung für eine algebraische Schleife einschalten. Ist diese einmal entdeckt, so bestehen zwei Möglichkeiten: die algebraische Schleife bestehen lassen oder sie entfernen. Ist die Geschwindigkeit der Simulation akzeptabel und besteht keine Veranlassung zu der Annahme, dass die algebraische Schleife eine Instabilität bewirkt, so kann man sie beibehalten. Ist die Lösung sehr langsam, so bietet es sich an, die Schleife zu entfernen.

§ **341** Zur Entfernung der algebraischen Schleife genügt manchmal eine einfache Umformulierung des Modells. Abbildung 2.51 zeigt ein System mit dem gleichen Input–Output Verhalten wie das in Abb. 2.50 jedoch ohne algebraische Schleife. Entsprechend ist eine Simulation mit diesem Modell deutlich schneller.

**Zwischenrechnung 18** Experimentieren Sie mit dem angegebenen Beispiel und anderen Modellen mit algebraischen Schleifen.

Abbildung 2.52: Das System erster Ordnung aus Abb. 2.50 mit Memory Block zum Aufbrechen der algebraischen Schleife

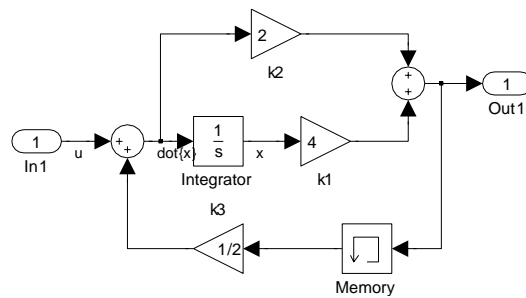
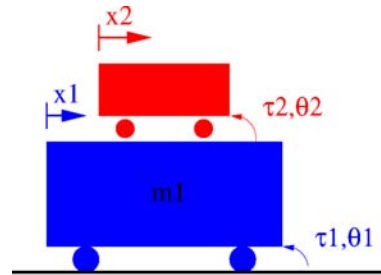


Abbildung 2.53: Bewegung zweier Wagen Huckepack



§ 342 Manchmal ist es nicht einfach möglich, eine algebraische Schleife durch Umformulierung zu entfernen, insbesondere, wenn sich diese z.B. direkt aus der Bewegungsgleichung eines physikalischen Systems ergibt. In diesem Fall kann die algebraische Schleife mit Hilfe eines Memory Block (zu finden in der Blockgruppe `Discrete`) aufgebrochen werden, vgl. Abb. 2.52. Die Hilfe erklärt die Aufgabe dieses Blocks: ‘The Memory block outputs its input from the previous time step, applying a one integration step sample-and-hold to its input signal.’ Der Memory Block kann daher stets verwendet werden, um einen algebraische Schleife zu unterbrechen, allerdings kann die mit dem Einfügen des Memory Blocks verbundene Verzögerung zu einer Reduktion der Genauigkeit führen.<sup>22</sup>

§ 343 Sind beide Verfahren, umformulieren des Modells und Einfügen des Memory Blocks nicht sinnvoll, so lässt sich auch eine `MatLab Function Block` einfügen, in dem das algebraische Problem direkt gelöst wird. Das setzt das Erstellen des entsprechenden `MATLAB Codes` voraus.

### Ein Beispiel mit Memory Block: zwei Wagen Huckepack

§ 344 Dabney und Harman [18] illustrieren das mit dem Memory Block verbundene Problem der Verzögerung am Beispiel zweier Wägelchen. Abbildung 2.53 zeigt die Situation: Wagen 1 bewegt sich auf dem Boden, Wagen 2 bewegt sich relativ zu Wagen 1 (und natürlich auch relativ zum Boden). Die Variablen  $x_1$  und  $x_2$  sind die Positionen der beiden Wagen im Inertialsystem; die Winkelverschiebungen der Wagen sind  $\theta_1$  und  $\theta_2$ , die wirkenden Drehmomente  $\tau_1$  und  $\tau_2$ . Die Drehung eines Rades ist mit der Position des Wagens im Inertialsystem verbunden über

$$x_1 = -r_1\theta_1 \quad \text{und} \quad x_2 = x_1 - r_2\theta_2 .$$

§ 345 Die kinetische Energie des Systems lässt sich mit Hilfe der Winkelgeschwindigkeiten der Räder ausdrücken als

$$T = \frac{1}{2}m_1(-r_1\dot{\theta}_1)^2 + \frac{1}{2}m_2(-r_1\dot{\theta}_1 - r_2\dot{\theta}_2)^2 .$$

Mit einer Lagrange Formulierung erhalten wir für die auf die Räder wirkenden Drehmomente

$$\tau_1 = m_1r_1^2\ddot{\theta}_1 + m_2r_1^2\ddot{\theta}_1 + m_2r_1r_2\ddot{\theta}_2$$

<sup>22</sup>An dieser Stelle möchte ich die Warnung aus der `MATLAB`-Hilfe weiter geben: ‘Avoid using the Memory block when integrating with `ode15s` or `ode113`, unless the input to the block does not change.’

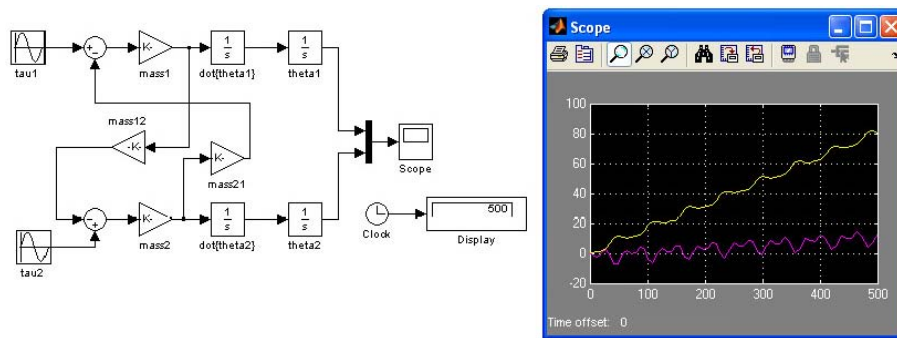


Abbildung 2.54: Algebraische Schleife im Modell zweier gekoppelter Wagen aus Abb. 2.53; die Uhr erlaubt auf langsamen Rechnern ein Abschätzen der Simulationsgeschwindigkeit

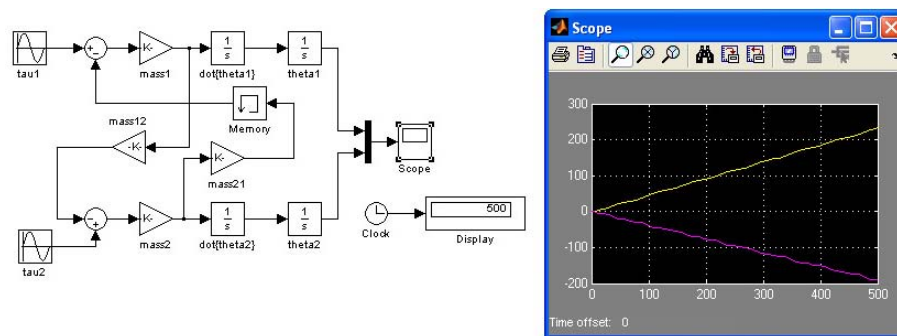


Abbildung 2.55: Wie Abb. 2.54 aber mit Memory Block zum Aufbrechen der algebraischen Schleife

$$\tau_2 = m_2 r_1^2 \ddot{\theta}_1 + m_2 r_2^2 \ddot{\theta}_2 .$$

Auflösen nach den Winkelbeschleunigungen liefert ein System gekoppelter Differentialgleichungen

$$\begin{aligned} \ddot{\theta}_1 &= K_{11}(\tau_1 - K_{21}\ddot{\theta}_2) \quad \text{und} \\ \ddot{\theta}_2 &= K_{22}(\tau_2 - K_{12}\ddot{\theta}_1) \end{aligned}$$

mit

$$k_{11} = \frac{1}{(m_1 + m_2)r_1^2}, \quad K_{12} = m_2 r_1 r_2 = K_{21} \quad \text{und} \quad K_{22} = \frac{1}{m_2 r_2^2}. \quad (2.5)$$

§ 346 Abbildung 2.54 zeigt im linken Teil ein Modell zur Beschreibung dieses Systems. Für die im rechten Teil des Bildes gezeigt Lösung (Dauer der Simulation 500 Zeiteinheiten) wurden die Parameter mittels eines m-Files gesetzt als

$$m_1 = 10, \quad m_2 = 5, \quad r_1 = 2 \quad \text{und} \quad r_2 = 1.5 .$$

Außerdem wurden die Parameter  $K_{ij}$  gemäß (2.5) vereinbart. Die Drehmomente als Eingangsgrößen wurden vereinbart als

$$\tau_1 = \sin(0.1t) \quad \text{und} \quad \tau_2 = \sin(0.2t - 1) .$$

§ 347 Die Ausführung der Simulation liefert eine Warnung vor einer algebraischen Schleife im MATLAB Workspace. Das Simulationsergebnis ist korrekt, auch wenn die Simulation auf Grund der algebraischen Schleife etwas langsam läuft.

§ 348 Abbildung 2.55 zeigt das gleiche Modell, jedoch mit einem **Memory Block** zum Aufbrechen der algebraischen Schleife. Die Ausführung des Modells ist wesentlich schneller, allerdings entsprechen die Ergebnisse im rechten Teil von Abb. 2.55 nicht denen in Abb. 2.54: das Einfügen des **Memory Block** verfälscht die Simulation so weit, dass qualitativ andere Ergebnisse entstehen. Das bedeutet, dass die durch den **Memory Block** eingeführte Verzögerung zu lang ist im Vergleich zu den Zeitskalen des Systems. Der Takt des **Memory Block** wird durch die Zeitauflösung der Simulation gegeben: eine Reduktion der maximalen Schrittweite der Simulation auf 0.1 führt wieder zu korrekten Resultaten. Allerdings geht damit der Vorteil des Aufbrechens der algebraischen Schleife verloren: zwar spart die fehlende algebraische Schleife Zeit, allerdings benötigt die deutlich reduzierte zeitliche Schrittweite ebenfalls Zeit.

§ 349 Das Aufbrechen der algebraischen Schleife in diesem Modell ist daher mit einem gewissen Risiko verbunden: wird nur die Schleife erkannt, nicht aber der mit dem Aufbrechen der Schleife eingeführte Fehler, so liefert die Simulation zwar Resultate, aber leider völligen Blödsinn.

## 2.6.7 Callbacks

§ 350 Einem Callback sind wir bereits in § 255 begegnet. Ein Callback ist eine **MATLAB**-Befehl, der im Falle eines bestimmten Ereignisses ausgeführt wird, z.B. wenn ein Modell geöffnet oder ein Block angeklickt wird. Mit Hilfe von Callbacks lassen sich GUIs für **SIMULINK** Modelle erzeugen.

§ 351 Callback Parameter in Zusammenhang mit einem Modell sind u.a.

- **PreLoadFcn** (Model pre-load function); wird ausgeführt, bevor das Modell geladen wird. Kann z.B. zum Laden von Variablen verwendet werden, die das Modell benötigt.
- **PostLoadFcn** (Model post-load function); wird nach dem Laden des Modells ausgeführt. Kann z.B. zur Erzeugung eines (graphischen) Interface verwendet werden.
- **InitFcn** (Model initialization function); wird vor der Ausführung des Modells aufgerufen. Kann z.B. zur Initialisierung von Variablen oder zur Konfiguration verwendet werden.
- **StartFcn** (Model start function); wird vor dem Start der Simulation ausgeführt nachdem die Variablen aus dem **MATLAB** Workspace gelesen wurden.
- **StopFcn** (Model stop function); wird nach der Beendigung der Simulation ausgeführt und nachdem der Output in den Workspace geschrieben wurde. Mit diesem Callback werden z.B. auch die automatischen Plots beim Scope erzeugt.
- **PreSaveFcn** (Model pre-save function); bevor das Modell gespeichert wird.
- **PostSaveFcn** (Model post-save function); wird ausgeführt, nachdem das Modell gespeichert wurde.
- **CloseFcn** (Model close function); wird vor dem Schließen eines Modells ausgeführt. Räumt entsprechend der in den Blöcken spezifizierten Befehle zum Löschen oder Schließen hinter dem Modell wieder auf und gibt so Speicher frei.

Callback Parameter für ein Modell können über **File:Model Properties** verändert werden. Die Parameter können jeweils über ein Menü ausgewählt werden und müssen nicht von Hand eingegeben werden.

§ 352 Callback Parameter für einzelne Blöcke ähneln denen für ein Modell; es gibt jedoch eine größere Zahl von Callbacks, da z.B. auch das Ausschneiden, Kopieren oder Verschieben von Blöcken berücksichtigt werden muss.

## Literaturhinweise

§ 353 Wie bereits in der Einleitung benannt, ist [18] die für dieses Kapitel am besten geeignete Literaturempfehlung. Weitergehende Details finden sich in der **MATLAB**-Hilfe. Für

ein schnelles Einlesen zum Überblick ist auch der Abschnitt in [53] geeignet. Eine sehr umfangreiche Quelle ist das Handbuch von Mathworks [93]. Einen schnellen Überblick über die Möglichkeiten von SIMULINK gibt auch die Präsentation von Robert White (<http://www.biomed.cas.cz/physiolres/pdf/51/51.65.pdf>).

**SIMULINK-Aufgabe 7** Betrachten Sie statt des Federpendels ein Fadenpendel der Masse  $m$  und der Länge  $l$  – und zwar nicht in der Näherung der Auslenkung für kleine Winkel. Dann ist die Bewegungsgleichung  $ml\ddot{\varphi} + cl\dot{\varphi} + mg \sin \varphi = 0$ . Lösen Sie das Problem (wenn möglich) in alle Varianten: explizites Modell, Übertragungsfunktion und State Space. Besteht mathematisch ein Unterschied zum Federpendel?

**SIMULINK-Aufgabe 8** Betrachten Sie statt der gekoppelten Federpendel in Abschn. 2.3.5 zwei Fadenpendel, die durch eine Feder gekoppelt sind (klassischer Versuchsaufbau aus dem Anfängerpraktikum). Stellen Sie die Bewegungsgleichung auf, bestimmen Sie eine analytische Lösung und modifizieren Sie die Simulation in SIMULINK für dieses Problem. Unterscheiden sich die Lösungen von gekoppelten Faden- und gekoppelten Federpendeln? Begründen Sie! Diskutieren Sie sowohl kleine als auch große Auslenkungen des Pendels; lassen Sie auch Reibung zu.

**SIMULINK-Aufgabe 9** Modifizieren Sie das Szenario der gekoppelten Federpendel aus Abschn. 2.3.5 derart, dass Sie eine der Randfedern weglassen. Stellen Sie die Bewegungsgleichungen auf und lösen Sie das Problem analytisch und in SIMULINK. Diskutieren Sie die Lösung physikalisch.

**SIMULINK-Aufgabe 10** Zwei Massen sind durch eine Feder gekoppelt (reibungsfreie Bewegung auf einer Luftkissenbahn ist möglich). Welche Bewegung ergibt sich bei Auslenkung einer der Massen. Stellen Sie wieder die Bewegungsgleichung auf und lösen Sie diese analytisch und in SIMULINK. Das Beispiel ist mit Animation in MATLAB als `Double Spring Mass System` enthalten. Betrachten Sie das ganze noch einmal mit Reibung.

**SIMULINK-Aufgabe 11** Verwenden Sie das Fadenpendel aus Aufg. 7, erlauben Sie jedoch eine Veränderung der Länge des Fadens, z.B. als  $l(t) = l_0 + vt$  mit  $v = \text{const}$  oder  $l(t) = l_0(1 + 0.5 \sin t)$ .

**SIMULINK-Aufgabe 12** Untersuchen Sie, z.B. für SIMULINK-Aufgabe 11 systematisch das Verhalten der verschiedenen Solver (jeweils unter Verwendung der Default-Einstellungen).

**SIMULINK-Aufgabe 13** Untersuchen Sie für ein einfaches System, dessen analytische Lösung Sie bestimmen können, die Abweichung der SIMULINK-Lösung verschiedener Solver von der analytischen Lösung für Default-Einstellungen. Verwenden Sie lange Zeiträume (z.B. 5000 s bei einem Federpendel mit  $\omega_0 = 1/\text{s}$ ).

**SIMULINK-Aufgabe 14** Weiten Sie das Modell der Rakete in Abschn. 2.4.5 auf eine Mehrstufenrakete aus. Realistische Zahlenwerte finden Sie z.B. unter ...

## Kurzprojekte in SIMULINK

### Agent Blue in Mäusen

§ 354 Dimethylarsinsäure wird als Unkrautvernichtungsmittel auf Brachland verwendet, wurde allerdings während des Vietnamkrieges auch als Agent Blue zur Zerstörung von Reisfeldern eingesetzt. Wie alle chemischen Substanzen sucht sich Dimethylarsinsäure sein Target nicht gezielt sondern wird von allem, was krecht und fleucht (oder vielleicht als Pflanze eher ortsfest ist) aufgenommen – und kann den neuen Wirt schädigen. Ein interessantes und gut dokumentiertes Modell für die Aufnahme von Dimethylarsinsäure durch Mäuse ist, zusammen mit den entsprechenden experimentellen Befunden, in Evans et al. [25] gegeben. Da

das Modell bereits gut in Compartments zerlegt ist, lässt es sich, wie auch von den Autoren vorgenommen, einfach in SIMULINK implementieren. Tun Sie dies. Das Paper und der zugehörige Anhang sollten die notwendigen Informationen liefern.

## Teil II

# Lineare zeitinvariante Systeme

# Laplace Transformation

We may regard the present state as the effect of its past and the cause of its future. An intellect which at a certain moment would know all forces that set nature in motion, and all positions of all items of which nature is composed, if its intellect were also vast enough to submit these data to analysis, it would embrace a single formula for the movements of the greatest bodies of the universe and those of the tiniest atom; for such an intellect nothing would be uncertain and the future just like the past would be present before its eyes.

P. Laplace

§ **355** Laplace hat der Wissenschaft nicht nur den oben in seinen Worten beschriebenen Laplace'schen Dämon gegeben sondern – neben weiteren Dingen – auch die nach ihm benannte Integraltransformation. Eine Laplace Transformation ist eine Integraltransformation, die eine von der Zeit abhängige Funktion  $x(t)$  in eine von einer komplexen Frequenz  $s$  abhängige Funktion  $X(s)$  überführt. Dabei wird die Integration bzw. Differentiation in eine Division durch bzw. Multiplikation mit dieser neuen Variablen  $s$  überführt – daher ist der Integratorblock in SIMULINK nicht mit einem  $\int$  versehen sondern mit  $\frac{1}{s}$ . Letzendlich reduziert diese Transformation eine Differentialgleichung auf eine algebraische Gleichung.

§ **356** Auf Grund dieser Eigenschaft sind Laplace Transformationen Standard-Verfahren zur analytischen Lösung von Differentialgleichungen bzw. Differentialgleichungssystemen. Diese Lösung erfolgt in einem dreistufigen Verfahren:

1. Transformation der Differentialgleichung aus dem Zeitbereich in eine algebraische Gleichung im Frequenzbereich,
2. Lösung der algebraischen Gleichung,
3. Rücktransformation der Lösung aus dem Frequenz- in den Zeitbereich.

Der Vorteil der Umwandlung in algebraische Gleichungen wird bei der Lösung einer gewöhnlichen Differentialgleichung nicht sofort offensichtlich – bei der Lösung eines Systems partieller Differentialgleichungen dagegen bringt die Reduktion auf ein lineares Gleichungssystem einen offensichtlichen Vorteil. Einen Haken hat das Verfahren allerdings: die Rücktransformation ist nicht unbedingt einfach. Verwenden wir die Laplace Transformation jedoch als Verfahren zur Bestimmung einer Übertragungsfunktion, so können wir auf die Rücktransformation verzichten (bzw. diese an SIMULINK delegieren).

§ **357** Die Verwendbarkeit zur Lösung einer DGL hat die Laplace Transformation mit der Fourier Transformation gemeinsam. Da letztere die bekanntere der Integraltransformationen ist und in der Signalanalyse weite Anwendung findet, wollen wir das Kapitel mit ihr beginnen. Die dabei einzuführenden Grundbegriffe sind für beide Transformationen gleichermaßen



gültig; dies gilt insbesondere für den Faltungssatz, der eine Faltung zweier Funktionen im Zeitbereich in ein Produkt dieser Funktionen im Frequenzbereich überführt, sowie für die Überführung der Ableitung in ein Produkt mit der neuen Variablen.

**Lernziele:** nach Durcharbeiten dieses Kapitels sollen Sie in der Lage sein,

- die grundlegenden Regeln und Ideen einer Laplace Transformation darzustellen und im Rahmen einer Simulation anzuwenden.
- einfache Differentialgleichungen bzw. Differentialgleichungssysteme unter Verwendung der Tabellen für Laplace und Fourier Transformation mit Hilfe derartiger Transformationen zu lösen.
- die Laplace Transformation vom Zeit- in den komplexen Frequenzbereich durchzuführen, um die Übertragungsfunktion eines Systems aus der Differentialgleichung zu bestimmen.
- eine Fast Fourier Transformation mit Hilfe eines Programmpaketes wie z.B. MATLAB oder SCILAB durchzuführen und die Ergebnisse zu interpretieren.

Die manuelle Durchführung von Transformationen sollten Sie an den Beispielen nachvollzogen haben; ein Drill insbesondere der Rücktransformation ist nicht Ziel dieses Kapitels – schließlich ist SIMULINK ein numerisches und kein analytisches Verfahren.

## 3.1 Fourier Reihe

§ **358** Ein Kapitel über Laplace Transformationen mit einem längeren Abschnitt über Fourier Transformationen zu beginnen, wirkt ein bisschen wie ‘Thema verfehlt’. Da sich die Laplace Transformation nur um einen integrierenden Faktor von der (kontinuierlichen) Fourier Transformation unterscheidet, lassen sich die wesentlichen Aspekte der Laplace Transformation an Hand der anschaulicheren und auf Grund ihrer vielfältigen Anwendungen in der Signalanalyse bekannteren Fourier Transformation aufzeigen.

§ **359** Die erste Begegnung mit Fourier erfolgt meist nicht mit der Integraltransformation sondern im Zusammenhang mit der anschaulicheren Fourier Reihe. So wird häufig bereits im Schulunterricht gezeigt, wie sich eine Grundschwingung und verschiedenen Oberschwingungen zu unterschiedlichen periodischen Signalen zusammensetzen lassen. Fourier selbst hat die Fourier Reihe allerdings nicht in diesem anschaulichen Zusammenhang eingeführt sondern im Zusammenhang mit der Lösung einer partiellen Differentialgleichung – und zwar nicht mit der Schwingungsgleichung, bei der die Überlagerung von Grundschwingung und höheren Harmonischen offensichtlich ist, sondern mit der Wärmeleitungsgleichung. Wir werden ebenfalls von einer PDGL ausgehen, allerdings von der anschaulicheren Wellengleichung. Fourier Reihen treten z.B. bei der Lösung des räumlichen Teils dieser DGL auf. Durch Überlagerung von Grund- und verschiedenen Oberschwingungen in veränderlichen Gewichtsanteilen lassen sich mit Hilfe der Fourier Reihe beliebige periodische Funktionen erzeugen. Das klassische Anwendungsbeispiel für die 1D Wellengleichung ist die schwingende Saite – und von ihr ist die Zerlegung in Grund- und Oberschwingung sowie die Bedeutung der Anteile der einzelnen Schwingungsmoden für die Klangfarbe des Instruments bekannt.

§ **360** Die wesentlich gebräuchlichere Anwendung ist die Umkehrung dieser Superposition: eine (nicht einmal zwingend periodische) Funktion wird in Grund- und Oberschwingungen zerlegt – es wird gleichsam der Fingerabdruck des Schallsignals erzeugt, mit dessen Hilfe sich die Schallquelle identifizieren lässt. Das Frequenzspektrum wird allerdings nicht diskret sondern kontinuierlich gewählt: bei dieser kontinuierlichen Fourier Analyse wird ein Zeitsignal in ein Frequenzspektrum umgewandelt. Der Vorgang wird daher als kontinuierliche Fourier Transformation bezeichnet.

§ **361** Die kontinuierliche Fourier Transformation ist ein mathematische Konstrukt – und hat formal ihre Berechtigung insbesondere als Hilfsmittel bei analytischen Betrachtungen. Zwar ist die Fourier Transformation in der Signalanalyse von großer Bedeutung, jedoch liegt ein

Signal in der Regel nicht als eine zeitkontinuierliche Funktion sondern bedingt durch endliche Sampling-Intervalle als eine Folge diskreter Messwerte vor. Bei der Untersuchung derartiger Zeitserien wird die diskrete Fourier Transformation verwendet.

**Verständnisfrage 30** Die Fourier Analyse wird auf zeitliche wie auf räumliche Signale angewandt – im Gegensatz zu einer in  $\mathbb{R}$ , d.h. in einem unendlichen Bereich definierten kontinuierlichen Funktion haben diese Signale eine endliche Länge/Dauer; auch ist der Definitionsbereich nicht kontinuierlich sondern diskret – Signale sind mathematisch also eher eine endliche Folge. Macht die Abbildung auf einen kontinuierlichen, unendlichen Frequenzbereich überhaupt Sinn oder lassen sich aus den Eigenschaften des Datensatzes Einschränkungen für den Frequenzbereich definieren, auf den das Zeitsignal abgebildet werden kann?

**Verständnisfrage 31** Da wir ohnehin schon die schwingende Saite erwähnt haben noch eine Frage zu Audio-Signalen: sind diese kontinuierlich oder diskret oder beides oder immer mal was anderes? An welcher Stelle könnte welche Form von Fourier Transformation sinnvoll sein?

### 3.1.1 Fourier Reihen anschaulich eingeführt

§ 362 An Hand der schwingenden Saite lässt sich die Fourier Reihe anschaulich einführen. Die partielle Differentialgleichung für die Auslenkung (Amplitude)  $A(x, t)$  lautet

$$\frac{\partial^2 A}{\partial x^2} = \frac{1}{c^2} \frac{\partial A^2}{\partial t^2}$$

mit den Randbedingungen  $x(0, t) = 0$  und  $x(L, t) = 0$  (Saite der Länge  $L$ , an beiden Enden fest eingespannt). Separation der PDGL liefert mit der Separationskonstante  $-\beta^2$  je eine gewöhnliche DGL für den zeitlichen und den räumlichen Anteil von  $A(x, t) = X(x)T(t)$ :

$$X'' + \beta^2 X = 0 \quad \text{und} \quad T'' + \beta^2 c^2 T = 0 .$$

Beide Gleichungen beschreiben eine freie Schwingung (harmonischer Oszillator) mit den Lösungen

$$\begin{aligned} X(x) &= \gamma_1 \cos(\beta x) + \gamma_2 \sin(\beta x) & \text{und} \\ T(t) &= \gamma_3 \cos(\omega t) + \gamma_4 \sin(\omega t) & \text{mit} \quad \omega = \beta c . \end{aligned} \quad (3.1)$$

Darin sind die  $\gamma_i$  die aus den Anfangsbedingungen zu bestimmenden Integrationskonstanten.

§ 363 Die Separationskonstante  $\beta$  im räumlichen Teil der Lösung (3.1) ist nicht beliebig sondern kann nur diskrete Werte annehmen: da die Saite an beiden Enden eingespannt ist, muss die Auslenkung dort verschwinden, d.h. die Lösung muss sich auf die Sinus-Terme beschränken derart, dass stets eine ganze Zahl von Wellenbergen zwischen den Endpunkten auftritt. Damit ist die Saitenlänge stets ein ganzzahliges Vielfaches der halben Wellenlänge:  $L = n\lambda_n/2$ . Oder formuliert über die Separationskonstante:  $L\beta = n\pi$  mit  $n = 1, 2, 3, \dots$ . Die räumliche Lösung besteht aus den Eigenmoden

$$X_n = \gamma_{2n} \sin\left(\frac{n\pi x}{l}\right) .$$

Die Lösung der PDGL für eine der Eigenmoden ergibt sich durch Multiplikation der zeitlichen und räumlichen Anteile der Lösung zu

$$A_n = (\gamma_{3,n} \cos \omega_n t + \gamma_{4,n} \sin(\omega_n t)) \gamma_{2,n} \sin\left(\frac{n\pi x}{l}\right) = a_n \cos(\omega_n t + \varphi_n) \sin \frac{n\pi x}{l} .$$

Die vollständige Lösung der PDGL ergibt sich durch Summation über die Eigenmoden:

$$\begin{aligned} A(x, t) &= \sum_{n=1}^{\infty} (\gamma_{3,n} \cos \omega_n t + \gamma_{4,n} \sin(\omega_n t)) \gamma_{2,n} \sin\left(\frac{n\pi x}{l}\right) \\ &= \sum_{n=1}^{\infty} a_n \cos(\omega_n t + \varphi_n) \sin \frac{n\pi x}{l} . \end{aligned} \quad (3.2)$$

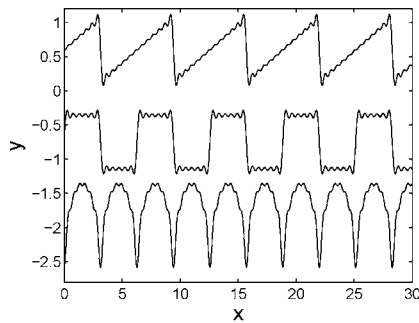


Abbildung 3.1: Fourier Reihe: Sägezahn, Rechteck und 'pulsierende Gleichspannung', Kurven durch additive Konstante getrennt; jeweils nur die ersten fünf Terme der Reihe wurden verwendet; die höchste verwendete Frequenz ist jeweils als Welligkeit der Funktionsgraphen zu erahnen

§ **364** Eine unendliche Summe über (die Summe von) Sinus- und/oder Kosinus-Termen wie in (3.2) wird als Fourier Reihe bezeichnet, allgemein definiert als

$$f(t) = \sum_{n=0}^{\infty} (A_n \cos(\omega_n t) + B_n \sin(\omega_n t)) \quad \text{mit} \quad \omega_n = \frac{2\pi n}{T} \quad \text{und} \quad B_0 = 0. \quad (3.3)$$

Darin ist  $T$  die Periode der Funktion  $f(t)$ . Beschränkung auf die Kosinus-Terme liefert eine ungerade Funktion,  $f(t) = -f(-t)$ , Beschränkung auf die Sinus-Terme dagegen eine gerade,  $f(-t) = f(t)$ .

**Verständnisfrage 32** In der Fourier Reihe (3.3) ist der Koeffizient  $B_0$  gleich Null gesetzt während für  $A_0$  keine Einschränkungen gemacht werden. Warum werden Sinus- und Kosinus-Terme so unterschiedlich behandelt? Oder ist die Einschränkung  $B_0 = 0$  nicht zwingend erforderlich?

**Verständnisfrage 33** Können Sie aus der mit Verständnisfrage 32 verbundenen Diskussion eine anschauliche Bedeutung von  $A_0$  finden?

§ **365** Fourier Reihen können auch als Annäherungen an analytisch nicht (bzw. nur für eine Periode) darstellbare Funktionen verwendet werden. So kann z.B. eine Sägezahnfunktion (oberes Teilbild in Abb. 3.1) analytisch dargestellt werden als  $f(x) = x$  für  $-\pi < x < \pi$ . Für größere Intervalle dagegen ist

$$f(x) = 2 \left( \frac{\sin x}{1} + \frac{\sin(2x)}{2} + \frac{\sin(3x)}{3} + \dots \right) = 2 \sum_{n=1}^{\infty} \frac{\sin(nx)}{n}.$$

Die Rechteckfunktion  $f(x) = a$  für  $0 < x < \pi$  und  $y = -a$  für  $\pi < x < 2\pi$  (vgl. mittleres Teilbild in Abb. 3.1) ist darstellbar als

$$f(x) = \frac{4a}{\pi} \left( \sin x + \frac{\sin(3x)}{3} + \frac{\sin(5x)}{5} + \dots \right) = \frac{4a}{\pi} \sum_{n=1}^{\infty} \frac{\sin((2n-1)x)}{2n-1},$$

der pulsierende Sinus  $f(x) = \sin x$  für  $0 \leq x \leq \pi$  als

$$f(x) = \frac{2}{\pi} - \frac{4}{\pi} \left( \frac{\cos(2x)}{1 \cdot 3} + \frac{\cos(4x)}{3 \cdot 5} + \frac{\cos(6x)}{5 \cdot 7} + \dots \right) = \frac{2}{\pi} - \frac{4}{\pi} \sum_{n=1}^{\infty} \frac{\cos(2nx)}{(2n-1)(2n+1)}.$$

**Zwischenrechnung 19** Lässt sich diese Superposition auf einfache Weise in SIMULINK darstellen? Oder ist das in MATLAB mit deutlich weniger Aufwand möglich?

### Bestimmung der Fourier Koeffizienten

§ **366** Die Superposition der verschiedenen Schwingungsmoden lässt sich einfach bewerkstelligen – aber wie bestimmt man die Koeffizienten für eine gegebene periodische Funktion

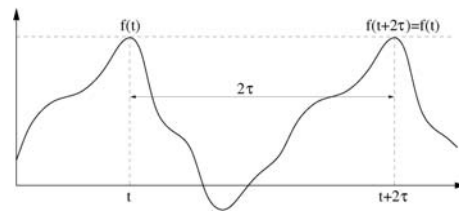
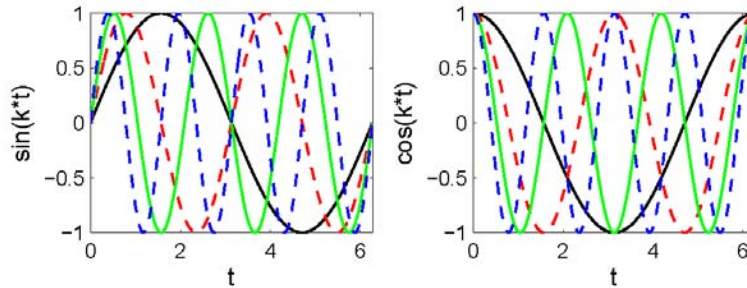


Abbildung 3.2: Periodische Funktion der Periodenlänge  $2\tau$

Abbildung 3.3: Sinus (links) und Kosinus (rechts) als Basisfunktionen der Fourier Transformation für  $k = 1$  (schwarz),  $k = 2$  (rot),  $k = 3$  (grün) und  $k = 4$  (blau)



$f(t)$ ? Betrachten wir eine periodische Funktion  $f(t)$ , d.h. eine Funktion, die sich stückweise zyklisch wiederholt, vgl. Abb. 3.2. Die Länge dieses Intervalls sei  $2\tau$ , so dass gilt

$$f(t + 2\tau) = f(t) = f(t - 2\tau) = f(t \pm 2n\tau) \quad \text{mit} \quad n = 0, 1, 2, \dots$$

§ 367 Gemäß (3.3) lässt sich eine derartige Funktion als eine Überlagerung von Sinus- und Kosinuswellen schreiben

$$f(t) = A_0 + \sum_{n=1}^{\infty} [A_n \cos(n\omega_0 t) + B_n \sin(n\omega_0 t)] \quad (3.4)$$

mit  $\omega_0 = \pi/\tau$  als der Grundfrequenz. In dieser Darstellung wird die Bedeutung des  $A_0$  in (3.3) deutlich (vgl. Verständnisfrage 33): dieser Term gibt den Gleichstromanteil. In dieser Bezeichnung spiegelt sich der Ursprung der Fourier Analyse in der Verarbeitung elektrischer Signale wieder: ein allgemeines elektrisches Signal lässt sich als eine Summe aus einem konstanten Signal (Gleichstrom oder -spannung) und dem überlagerten Wechselsignal darstellen. Bei der Mittelung über eine Periode oder Vielfache der Periode fallen die Wechselanteile weg und der Gleichanteil überlebt.  $A_0$  gibt daher gleichzeitig den Mittelwert des Signals.

§ 368 Formal begegnen wir hier einem nicht-geometrischen Vektorraum: der Raum der periodischen Funktionen wird aufgespannt von den Basisfunktionen Sinus und Kosinus, vgl. Abb. 3.3. Die Koeffizienten  $A_i$  und  $B_i$  der Fourier Reihe (3.4) entsprechen den Komponenten der Funktionen  $f(t)$  entlang dieser Basen.

§ 369 Mit Hilfe des Additionstheorems  $\cos(\alpha \pm \beta) = \cos \alpha \cos \beta \mp \sin \alpha \sin \beta$  können wir ein einfaches Beispiel konstruieren. Mit  $\alpha = \beta = \omega t$  ist

$$f(t) = \cos^2(\omega t) = \frac{1}{2} + \frac{1}{2} \cos(2\omega t)$$

unmittelbar die Darstellung der periodischen Funktion  $f(t) = \cos^2(\omega t)$ . Diese Darstellung beschränkt sich auf die beiden Koeffizienten  $A_0 = 1/2$  und  $A_2 = 1/2$ , alle anderen Koeffizienten verschwinden, vgl. Abb. 3.4. In diesem Fall bildet die Zerlegung in eine endliche Fourier Reihe nicht nur eine Annäherung an die Funktion sondern liefert die Funktion exakt.

**Zwischenrechnung 20** Sie haben gerade die Additionstheorem nicht im Kopf und keine geeignete Formelsammlung zur Hand. Verifizieren Sie die Fourier Reihe für  $f(t) = \cos^2(t)$  durch Darstellung mit Hilfe einer Potenzreihe oder durch Rückgriff auf die Euler'sche Formel.

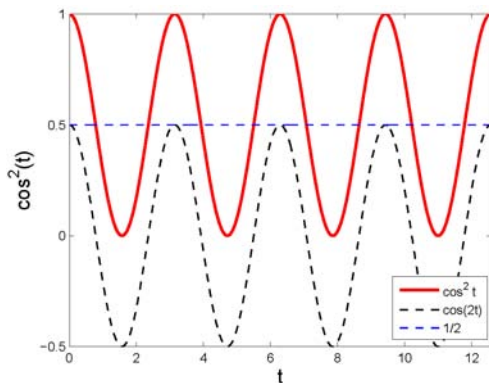


Abbildung 3.4: Die Funktion  $f(t) = \cos^2(t)$  (rot) lässt sich mit Hilfe einfacher trigonometrischer Umformungen in eine aus einer Konstanten  $a_0 = 1/2$  (blau) und dem Term  $\cos(2t)/2$  bestehende Fourier Reihe zerlegen

§ **370** Für technische Zwecke wie digitalisierte Signalübertragung ist es natürlich viel ökonomischer, diese Funktion  $f(t) = \sin^2(t)$  nicht als eine Zeitreihe entsprechend einer Wertetabelle zu übertragen sondern nur die Fourier Koeffizienten anzugeben. Dieses  $f(t)$  ist so konstruiert, dass die Zahl der zu seiner Darstellung erforderlichen Fourier Koeffizienten sehr gering ist, die Funktion ist also bandbreitenlimitiert.

**Faustregel 1** *Glatte Funktionen, d.h. stetige und differenzierbare Funktionen, sind bandbreitenlimitiert.*

Oder anschaulich: um bandbreitenlimitiert zu sein, darf eine Funktion weder Stufen (Unstetigkeiten) noch Knicke (und damit Unstetigkeiten in der ersten Ableitung) aufweisen. Dieser Begriff wird im Zusammenhang mit Abb. 3.6 und 3.7 noch deutlicher werden.

§ **371** Die meisten periodischen Funktionen lassen sich nicht wie in § 369 analytisch in eine passende Fourier Reihe entwickeln. Die Fourier Koeffizienten in (3.4) sind

$$A_n = \frac{1}{\tau} \int_{-\tau}^{\tau} f(t) \cos(n\omega_0 t) dt \quad \text{und} \quad B_n = \frac{1}{\tau} \int_{-\tau}^{\tau} f(t) \sin(n\omega_0 t) dt \quad n \in \mathbb{N} \quad (3.5)$$

bei Verwendung der Notation aus Abb. 3.2 bzw.

$$A_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \cos(n\omega_0 t) dt \quad \text{und} \quad B_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \sin(n\omega_0 t) dt \quad n \in \mathbb{N}$$

bei Verwendung von  $T$  als der Periode. Beide Ausdrücke gelten für  $n \geq 1$ , für  $n = 0$  ergibt sich zusätzlich der Mittelwert (Gleichstromanteil)

$$A_0 = \frac{1}{T} \int_{-T/2}^{T/2} f(t) dt .$$

§ **372** Alternativ lässt sich die Fourier Reihe auch in komplexer Schreibweise darstellen als

$$f(t) = \sum_{n=-\infty}^{\infty} C_n e^{in\omega_0 t} \quad \text{mit} \quad \omega_0 = \frac{2\pi}{T} \quad (3.6)$$

mit den Fourier Koeffizienten

$$C_n = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-in\omega_0 t} dt . \quad (3.7)$$

Die Koeffizienten  $C_{-n}$  und  $C_n$  sind konjugiert komplex zueinander.

**Verständnisfrage 34** Warum läuft bei der komplexen Darstellung die Summation von  $n = -\infty$  statt von  $n = 0$ ?

**Verständnisfrage 35** Wie geht man damit um, dass die physikalischen Größen reell sind, die gemäß (3.7) definierten Fourier Koeffizienten dagegen komplex? Das Problem und seine Lösung sollten Ihnen bekannt sein!

§ 373 Die Koeffizienten  $A_n$  und  $B_n$  stellen im Frequenzbereich ein Linienspektrum dar: durch die Wichtung der Winkelfunktionen mit  $f(t)$  pickt man sich bei der Integration die spektralen Komponenten aus  $f(t)$  heraus, die den geraden bzw. ungeraden Anteilen der Frequenz  $\omega_n = n\omega_0$  entsprechen. Für die Anwendungen interessiert das Amplitudenspektrum  $a_n$  der einzelnen Frequenzen

$$a_n = \sqrt{A_n^2 + B_n^2} \quad \text{wegen} \quad a_n \sin \varphi_n = A_n \quad \text{und} \quad a_n \cos \varphi_n = B_n ,$$

bzw. das dazu gehörige Phasenspektrum

$$\varphi_n = \arctan \frac{A_n}{B_n} .$$

Die Darstellung von Amplitude und Phase der einzelnen Komponenten ist in der komplexen Darstellung (3.7) natürlich offensichtlicher.

### Winkelfunktionen als Basen eines nicht-geometrischen Vektorraums

§ 374 Zwischen der Fourier Reihe und den Fourier Koeffizienten einerseits und der Projektion eines Vektors auf eine bestimmte Zahl von Basisvektoren besteht, wie bereits in § 368 angedeutet, ein enger Zusammenhang. Betrachten wir dazu eine gerade Funktion, d.h. die  $a_n$  verschwinden und die Fourier Reihe wird

$$f(t) = \frac{A_0}{2} + \sum_{n=1}^{\infty} B_n \sin(n\omega_0 t) . \quad (3.8)$$

In Anlehnung an die Basisvektoren können wir Basisfunktionen definieren als

$$u_n(t) = \frac{1}{\sqrt{T}} \sin(n\omega_0 t) .$$

§ 375 Wie für die Basisvektoren lässt sich für Basisfunktionen ein inneres Produkt oder Skalarprodukt zweier Funktionen  $f$  und  $g$  definieren als

$$f \cdot g = \int_{-L}^L f(t)g(t) dx .$$

Entsprechend den Regeln für Basisvektoren gilt für die Basisfunktionen

$$(u_n \cdot u_m) = \delta_{nm} = \begin{cases} 0 & \text{für } n \neq m \\ 1 & \text{für } n = m \end{cases} ,$$

d.h. die Basisfunktionen sind normiert und wechselseitig orthogonal. Sie können daher als die Verallgemeinerungen von orthogonalen Einheitsvektoren in einen Funktionenraum betrachtet werden. Damit lässt sich unsere ungerade Funktion  $f(t)$  als eine Summe von Basisfunktionen  $u_n(t)$  schreiben:

$$f(t) = \sum_{n=1}^{\infty} c_n u_n(t) . \quad (3.9)$$

Der Koeffizient  $c_n$  entspricht der Projektion von  $f$  auf die Basisfunktion:  $c_n = (u_n \cdot f)$ , so dass (3.9) auch geschrieben werden kann als

$$f(t) = \sum_{n=1}^{\infty} (u_n \cdot f) u_n(t) .$$

**Zwischenrechnung 21** Zeigen Sie, dass (3.5) orthogonale Basen sind:

$$\int_0^{2\pi} \sin(nx) \sin(mx) \, dx = \int_0^{2\pi} \cos(nx) \cos(mx) \, dx = \int_0^{2\pi} \sin(nx) \cos(mx) \, dx = 0.$$

§ **376** Die Rekonstruktion der Funktion  $f(t)$  erfordert, dass man über ihre Projektionen  $(u_n \cdot f)u_n(t)$  auf alle Basisfunktionen  $u_n(t)$  summiert. Ebenso wie bei den Vektoren eine Projektion in Unterräume vorgenommen werden kann, kann man die Fourier Serie ebenfalls über einen beschränkten Satz von Basisfunktionen summieren

$$f(t)_{\text{gefiltert}} = \sum_{n=1}^{n_2} (u_n \cdot f)u_n(t).$$

Diese Art von Projektion wird als gefiltert bezeichnet, da es sich dabei um eine Filterung handelt. Diese Eigenschaft macht die Fourier Transformationen so nützlich, so sind sie die Grundlage für viele digitale Filtertechniken; auch das, allerdings nicht verlustfreie, Bildkompressionsformat JPEG basiert auf einer Fourier Transformation (oder genauer nur auf deren Kosinus-Termen, d.h. auf einer Kosinus-Transformation).

### Fourier Reihe und nichtperiodische Funktion

§ **377** Auch eine nicht periodische Funktion kann, zumindest in einem endlichen Intervall, durch eine Fourier Reihe angenähert werden. Betrachten wir als Beispiel  $f(x) = x^2$ . Da es sich hierbei um eine gerade Funktion handelt,  $f(x) = f(-x)$ , verschwinden die Sinus-Terme und die Funktion lässt sich als Fourier Reihe in der Form

$$x^2 = \sum_{n=0}^{\infty} A_n \cos(nx)$$

schreiben. Zur Bestimmung der Fourier Koeffizienten müssen die Integrale (3.5) ausgeführt werden. Beschränken wir uns auf den Bereich  $-\pi < x \leq \pi$ , so ergibt sich unter Verwendung der Produktintegration

$$A_n = \frac{1}{\pi} \int_{-\pi}^{\pi} x^2 \cos(nx) \, dx = \frac{4}{n^2} (-1)^n \quad n > 0 \quad (3.10)$$

sowie

$$A_0 = \frac{1}{\pi} \int_{-\pi}^{\pi} x^2 \, dx = \frac{2}{3} \pi^2.$$

§ **378** Zusammen gefasst ergibt sich für  $-\pi < x \leq \pi$  als Annäherung an die Funktion  $x^2$ :

$$x^2 = \frac{\pi^2}{3} + 4 \sum_{n=1}^{\infty} \frac{(-1)^n \cos(nx)}{n^2}.$$

Der Aufbau der Funktion ist in Abb. 3.5 für die ersten fünf Glieder der Reihe (rechts) gezeigt, die Beiträge der einzelnen Summanden (entsprechend den Gliedern der zu Grunde liegenden Folge) sind im linken Teil gezeigt. Für  $x = \pi$  ergibt sich

$$\pi^2 = \frac{\pi^2}{3} + 4 \sum_{n=1}^{\infty} \frac{1}{n^2} \quad \rightarrow \quad \pi^2 = 6 \sum_{n=1}^{\infty} \frac{1}{n^2}.$$

Diese Verknüpfung einer einfachen Reihe natürlicher Zahlen mit der transzendenten Zahl  $\pi$  ist ein überraschender Nebeneffekt dieser Analyse.

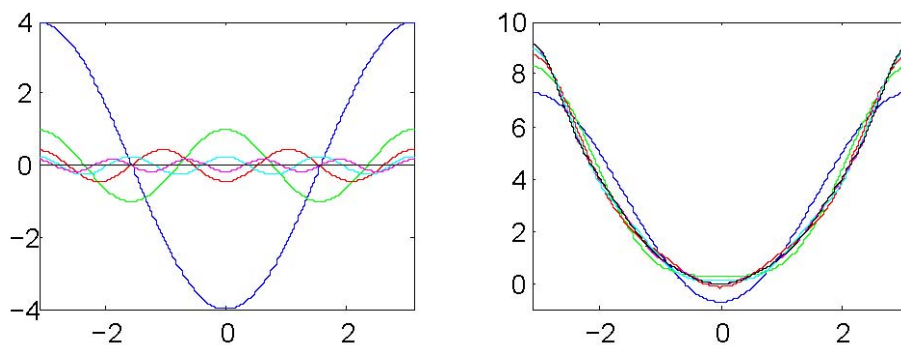


Abbildung 3.5: Fourier Reihe zur Darstellung der Funktion  $f(x) = x^2$  im Intervall  $-\pi < x \leq \pi$ ; links die Beiträge der einzelnen Terme, d.h. die Glieder der Reihe zu Grunde liegenden Folge, rechts die ersten Glieder der Folge

**Zwischenrechnung 22** Wie verändert sich das Ergebnis bei Änderung des betrachteten Bereichs, z.B. bei  $-10\pi < x \leq 10\pi$ ? Was passiert, wenn der betrachtete Bereich asymmetrisch ist, z.B.  $-\pi < x \leq 2\pi$  oder im Extremfall  $0 < x \leq 2\pi$ ?

§ 379 Formal lässt sich diese Annäherung auch nicht-periodischer Signale informationstheoretisch begründen. Entscheidend ist dabei, dass sich der Informationsgehalt des Signals nicht verändert. Da Information<sup>1</sup> über die Neuigkeit definiert ist, können wir jedes nichtperiodische aber zeitbegrenzte Signal in ein unendliches periodisches Signal überführen, in dem wir es einfach periodisch fortsetzen. Für das Beispiel aus Abb. 3.5 würden wir dann eine unendliche Kette von Parabeln erhalten, die aber abgesehen von der Verschiebung um  $nT$  als Vielfaches

<sup>1</sup>Information kann mit einer der drei folgenden Formulierungen definiert werden:

(1) Information ist das, was wir nicht wissen.

(2) Information ist die Eigenschaft einer Nachricht, beim Empfänger Kenntniszuwachs zu erzeugen.

(3) Information ist Wissenszuwachs und damit Abbau von Unsicherheiten über einen Sachverhalt.

Information hat also etwas mit Unbekanntem oder Unerwartetem zu tun. Alles, was wir bereits wissen, enthält keine Information. Nur der Aspekt des Unbekannten ist wichtig, nicht die Inhaltsschwere oder Bedeutsamkeit der Nachricht: die Nachricht ‚Die Kuh XYZ auf der Alm QRS hat heute gekalbt‘ hat den gleichen Informationsgehalt wie ‚Sie haben den Jackpott mit 21 Mio. Euro geknackt‘. Die Nachricht ‚Irgendwo auf der Welt hat heute irgendeine beliebige Kuh gekalbt‘ hat dagegen einen Informationsgehalt von Null: zwar kann die Meldung irritieren, da man nicht weiß, was man mit ihr anfangen soll – die Information in dieser Nachricht ist jedoch trivial.

Um Information wahrscheinlichkeitstheoretisch zu beschreiben betrachten wir ein Urnen-Experiment: In einer Urne befinde sich eine große Menge Kugeln, davon seien 70% weiß, 20% schwarz und 10% rot. X zieht jeweils eine Kugel und ruft Y das Ergebnis zu. Dann ist X zusammen mit der Urne eine Nachrichtenquelle, die die Nachrichten ‚weiß‘, ‚rot‘ und ‚schwarz‘ erzeugt. Y ist ein Nachrichtenempfänger, der sich genau für diese Informationen interessiert. Welche der drei Nachrichten enthält nun für Y die größte Information? Nach der Ziehung und Übermittlung von vielen Kugeln kennt der Empfänger Y die Wahrscheinlichkeit  $p_x$  für das Auftreten der einzelnen Farben:  $p_w=0.7$ ,  $p_s=0.2$  und  $p_r=0.1$ . Damit kennt er auch den Informationsgehalt dieser Nachrichten: die Nachricht ‚weiß‘ enthält die geringste Information (sie ist aufgrund ihrer großen Wahrscheinlichkeit das Standardsignal, interessant sind erst Abweichungen von ‚weiß‘); die Nachricht ‚rot‘ enthält am meisten Information. Das ist anschaulich, wenn man sich vorstellt, dass die Urne nur Kugeln einer Farbe enthält, z.B. weiß. Dann ist  $p_w=1$  und der Empfänger Y weiß bereits vorher, welche Nachricht ihm X übermitteln wird. Da es sich dann um eine sichere Nachricht handelt, ist ihr Informationsgehalt gleich Null. Aufgrund dieser statistischen Beschreibungsweise wird auch der Begriff Versuchsausgang anstelle von Signal, Zeichen oder Information verwendet.

Aus diesem Experiment können wir zwei Kriterien zur Definition des Informationsgehaltes einer Nachricht ableiten:

(K1) Der Informationsgehalt  $I_x$  einer Nachricht ist umso größer, je kleiner die Wahrscheinlichkeit  $p_x$  ihres Auftretens ist, d.h. je größer ihr ‚Überraschungswert‘ ist. Damit wird Information als der Neuigkeitsgehalt aber nicht die Inhaltsschwere einer Nachricht definiert.

(K2) Eine Nachricht mit der Wahrscheinlichkeit  $p_x=1$ , d.h. das sichere Ereignis, hat den Informationsgehalt  $I_x=0$ .

Ein drittes Kriterium zur Definition des Informationsgehaltes erhält man bei der Verwendung mehrere Nachrichten:

(K3) Der Informationsgehalt verschiedener voneinander unabhängiger Nachrichten soll sich addieren.



der Signaldauer  $T$  alle identisch sind: ist eine beliebige der Parabeln ‘gelesen’, so ist der Informationsgehalt des unendlich großen Signals bekannt. Damit ist das endliche Signal auf ein periodisches Signal zurück geführt und kann wie ein solches einer Fourier Transformation unterzogen werden. Das größtmögliche Abtastintervall  $T_A = 1/f_A < (2f_{\max})^{-1}$  für ein Signal mit der maximal auftretenden Frequenz  $f_{\max}$  ist definiert über

$$f(t) = \sum_{n=-\infty}^{\infty} f(nT_A) \operatorname{si} \left( \frac{\pi}{T_A} (t - nT_A) \right).$$

Darin ist  $\operatorname{si}$  eine Abkürzung für  $\operatorname{sinc}$  oder  $\operatorname{sine cardinal}$ , vgl. § 388 und insbesondere die Fußnote dazu.

## 3.2 Fourier Transformation

§ 380 Die Fourier Transformation ist die Abbildung einer beliebigen Funktion  $f(t)$  auf eine Fouriertransformierte Funktion  $F(\omega)$ . Die Einschränkung auf periodische Funktionen  $f(t)$  wie bei der Fourier Reihe ist nicht erforderlich. Damit gibt es allerdings auch nicht wie bei der periodischen Funktion ein natürliches endliches Integrationsintervall mit der Länge  $T$  der Periode. Stattdessen wird von  $-\infty$  bis  $+\infty$  integriert.

**Verständnisfrage 36** Ergibt sich damit ein Widerspruch zum endlichen Integrationsintervall in (3.10)?

§ 381 Diese Abbildung existiert in zwei Varianten, der kontinuierlichen Fourier Transformation und der diskreten Fourier Transformation.

### 3.2.1 Kontinuierliche Fourier Transformation

§ 382 Bei der Entwicklung einer Funktion  $f(t)$  in eine Fourier Reihe verwenden wir diskrete Basen  $\omega_n = n\omega_0$  mit  $\omega_0 = 2\pi/T$ . Da wir bei der Fourier Transformation nicht nur periodische Funktionen betrachten, entfällt die Beschränkung auf ein endliches Zeitintervall, so dass  $T$  gegen unendlich geht:  $T \rightarrow \infty$ . Damit gibt es keine diskreten Werte  $\omega_n$  mehr sondern wir erhalten ein kontinuierliches Frequenzspektrum  $\omega$ .

§ 383 Für die Fourier Koeffizienten gilt in der komplexen Darstellung (3.6)

$$C_n = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-in\omega_0 t} dt \quad \Rightarrow \quad \lim_{T \rightarrow \infty} (T C_n) = \int_{-\infty}^{\infty} f(t) e^{i\omega_n t} dt.$$

§ 384 Darauf aufbauend ergibt sich für die Transformationsgleichungen der kontinuierlichen Fourier Transformation  $f(t) \rightarrow F(\omega)$

$$\begin{aligned} F(\omega) &= \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt && \text{Hintransformation} \\ f(t) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega t} d\omega && \text{Rücktransformation.} \end{aligned} \quad (3.11)$$

Die Transformationen sind nicht symmetrisch, da bei  $F(\omega) \rightarrow f(t)$  noch ein Vorfaktor  $2\pi$  auftritt. Dieser ist zur Normierung erforderlich, damit  $f(0) = \int_{-\infty}^{\infty} f(t) dt$  den Mittelwert gibt. Die auf der Frequenz  $\nu$  anstelle der Kreisfrequenz  $\omega$  basierende Notation von Weaver [171] ist symmetrisch, handelt sich dafür aber den Faktor  $2\pi$  im Exponenten ein:

$$F(\nu) = \int_{-\infty}^{\infty} f(t) e^{-2\pi i \nu t} dt \quad \text{und} \quad f(t) = \int_{-\infty}^{\infty} F(\nu) e^{2\pi i \nu t} dt.$$

Abbildung 3.6: Im Zeitbereich (*links*) langsam (rot) und schnell (blau) veränderliche Funktion und ihr Fourier Spektrum (*rechts*)

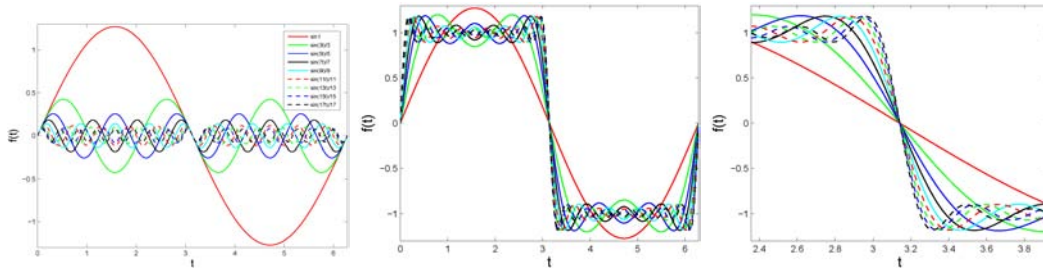
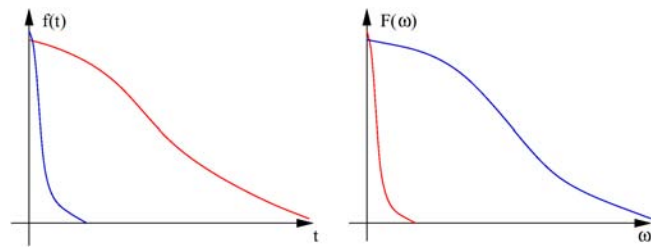


Abbildung 3.7: Fourier Reihe eines Rechtecks: die Amplitude nimmt mit zunehmender Frequenz ab, allerdings sind die hochfrequenten Beiträge unerlässlich, um den Sprung bei  $t = n\pi$  zu erzeugen – was insbesondere in der Ausschnittvergrößerung rechts deutlich wird

### Fourier Transformation Beispiele

§ 385 Die Fourier Transformation bildet die Funktion  $f(t)$  aus dem Zeitbereich auf ein (kontinuierliches) Frequenzspektrum  $F(\omega)$  ab. Aus unserer Erfahrung mit der Fourier Reihe können wir eine Faustregel formulieren:

**Faustregel 2** *Eine langsam veränderliche Funktion  $f(t)$  hat im wesentlichen niederfrequente spektrale Komponenten, eine schnell veränderliche Funktion dagegen hat spektrale Komponenten über einen weiten Frequenzbereich.*

Die Regel wird vielleicht am einfachsten bei der Annäherung einer Rechteckfunktion (oder der Sprungfunktion) deutlich: die Fourier Reihe ist [9, 154]

$$f(t) = \frac{4}{\pi} \sum_{n=1,3,5,\dots}^{\infty} \frac{1}{n} \sin(n\omega_0 t) \quad \text{mit} \quad \omega_0 = \frac{2\pi}{T}.$$

§ 386 Im linken Teil von Abb. 3.7 sind die einzelnen Terme mit ihren Vorfaktoren gezeigt: die Amplitude der einzelnen Summanden nimmt mit zunehmendem  $n$  und damit mit zunehmender Frequenz der Schwingung ab. Die Grundschwingung dominiert eindeutig das Amplitudenspektrum, ihr Makel wird in der Nähe der Nulldurchgänge am deutlichsten: hier hat die Rechteckfunktion ihre Sprungstellen und gleichzeitig die größte Abweichung von der Grundschwingung. Um diesen Makel zu beseitigen, sind nur die Oberschwingungen interessant, die beim Nulldurchgang den Vorzeichenwechsel in gleicher Richtung vornehmen wie die Grundschwingung. Diese können die Lücke zwischen Grundschwingung und Rechteck auffüllen, vgl. mittleren Teil von Abb. 3.7. Daher wird die Rechteckfunktion nur aus den ungeraden Termen der Reihe zusammen gesetzt. Um eine Annäherung an die Sprungstelle zu erreichen, müssen auch die Oberschwingungen berücksichtigt werden, die beim Nulldurchgang recht steil sind, d.h. die hochfrequenten Anteile sind um so wichtiger, je steiler die anzunähernde Funktion ist. Dies wird in der Ausschnittvergrößerung um den Nulldurchgang nochmals deutlich.

§ 387 Allerdings schießt die Berücksichtigung aller Oberwellen im wahrsten Sinne des Wortes über das Ziel hinaus: an der Sprungstelle entsteht jeweils ein Überschwinger, vgl. Abb. 3.8. Dieser wird bereits bei recht niedrigen Ordnungen deutlich, mit zunehmender Ordnung wird

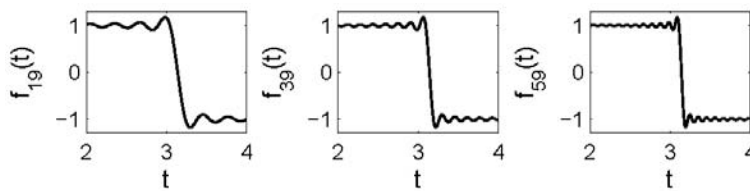


Abbildung 3.8: Gibbs'sches Phänomen: an der Sprungstelle entsteht ein Überschwinger, Index gibt die berücksichtigten Glieder an

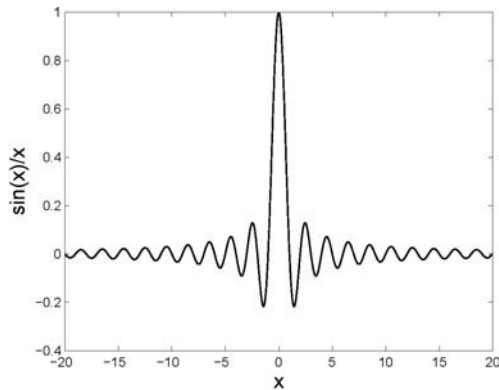


Abbildung 3.9: Die Fourier Transformierte einer Rechteckfunktion hat die Form der Spaltfunktion  $\sin(x)/x$

er zwar immer schmaler, verschwindet aber nicht. Dieser Überschwinger wird als Gibbs'sches Phänomen bezeichnet, eine genauere Diskussion findet sich in [10].

**Querverbindung 1** Das Problem, ein Rechteck aus Winkelfunktionen zu synthetisieren, zeigt sich auch in JPEG. Strichzeichnungen lassen sich in JPEG nicht vernünftig darstellen: entweder gibt es Artefakte (ausgewaschene Kanten, Überschwingen) oder die Kompression ist vernachlässigbar, da zu hohe Koeffizienten der Kosinus-Reihe berücksichtigt werden. Bei langsamen Veränderungen wie im größten Teil normaler Photos dagegen ist JPEG effizient, da nur die untersten Koeffizienten der Kosinus-Transformation benötigt werden.

§ 388 Betrachten wir statt der Rechteckfunktion ein einzelnes Rechteck, d.h. eine nicht-periodische Funktion der Form

$$f(t) = \begin{cases} 1 & \text{für } -T/2 \leq t \leq T/2 \\ 0 & \text{sonst} \end{cases} .$$

Da diese Funktion nicht-periodisch ist, lässt sie sich nicht durch eine Fourier Reihe beschreiben. Da es sich um eine gerade Funktion handelt, wird sie nur Kosinus-Terme und keine Sinus-Terme enthalten, d.h. der Imaginärteil in der Fourier Transformation verschwindet und wir erhalten

$$F(\omega) = 2 \int_0^{T/2} \cos(\omega t) dt = T \frac{\sin(\omega T/2)}{\omega T/2} .$$

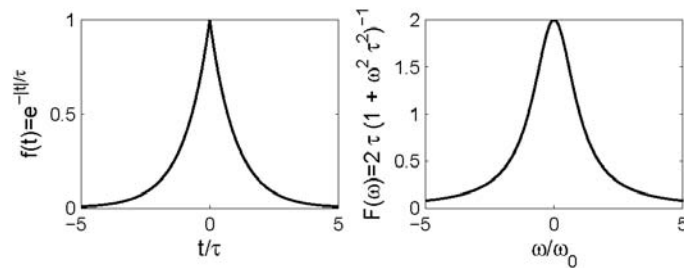
Die Fourier Transformierte eines Rechtecks ist also eine Funktion der Form  $(\sin x)/x$  wie in Abb. 3.9 gezeigt.<sup>2</sup>

**Querverbindung 2** Die Funktion  $f(x) = \sin(x)/x$  hat für  $x = 0$  eine Definitionslücke. Ist das ein ernstes Problem (die Definitionslücke von  $f(x) = x^{-1}$  ist immerhin eine Polstelle!) oder kann man über die Lücke einfach hinweg gehen. Wenn ja, mit welcher Begründung?

<sup>2</sup>Die Funktion  $\sin(x)/x$  wird als Spaltfunktion bezeichnet, da sich das Beugungsmuster hinter einem Einfachspalt mit ihr beschreiben lässt. Sie spielt eine so wichtige Rolle im Zusammenhang mit dem Abtasttheorem, dass sie teilweise mit einem eigenen Namen versehen wird: sinc für sine cardinal, abgekürzt als si. Zwei Definitionen sind gebräuchlich:

$$\text{si}(t) = \begin{cases} 1 & \text{für } t = 0 \\ \frac{\sin t}{t} & \text{sonst} \end{cases} \quad \text{mit} \quad \int_{-\infty}^{\infty} \text{si}(t) dt = \pi$$

Abbildung 3.10: Die Fourier Transformierte der beidseitigen Exponentialfunktion (*links*) ist die Lorentz Funktion (*rechts*)



§ 389 Lassen wir  $T$  gegen Null gehen bei gleichzeitiger Erhaltung der Fläche unter dem Rechteck, so erhalten wir statt der Rechteckfunktion die  $\delta$ -Funktion. Über diese Kastenfunktion für Kastenbreite gegen Null wird die  $\delta$ -Funktion häufig eingeführt. Die Fourier Transformierte erklärt damit auch eine der häufig als Annäherung an die  $\delta$ -Funktion gegebenen Varianten:

$$\delta(x) = \lim_{n \rightarrow \infty} g_n(x) = \lim_{n \rightarrow \infty} \frac{1}{\pi} \frac{\sin nx}{x} = \lim_{n \rightarrow \infty} \frac{1}{2\pi} \int_{-n}^{+n} e^{ikx} dk = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{ikx} dk .$$

§ 390 Die normierte Gauß Funktion

$$f(t) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{t^2}{2\sigma^2}}$$

ist ebenfalls eine gerade Funktion, d.h. auch hier ist die Beschränkung auf die Kosinus Terme ausreichend. Ihre Fourier Transformierte ist daher

$$F(\omega) = \frac{1}{\sigma \sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-\frac{t^2}{2\sigma^2}} e^{i\omega t} dt = \frac{2}{\sigma \sqrt{2\pi}} \int_0^{\infty} e^{-\frac{t^2}{2\sigma^2}} \cos(\omega t) dt = e^{-\frac{\sigma^2 \omega^2}{2}}$$

und damit ebenfalls eine normierte Gauß Funktion.

**Verständnisfrage 37** Sagt die Tatsache, dass die Fourier Transformierte einer normierten Gauß-Funktion wieder eine normierte Gauß-Funktion ist, auch etwas über die Fourier Transformierte einer Exponentialfunktion aus? Oder genauer: bleibt die Fourier Transformierte einer Exponentialfunktion eine Exponentialfunktion? Und weiter: bleibt die Fourier Transformierte einer nicht normierten Gauß Funktion eine Gauß Funktion? Versuchen Sie die Frage aus der Definition der Transformation (3.11) zu beantworten, sonst hilft Tabelle 3.1 weiter.

§ 391 Mit Hilfe der normierten Gauß Funktion können wir auch ein formales Beispiel für Faustregel 2 geben: bei  $f(t)$  erscheint die Varianz  $\sigma$  im Nenner des des Exponenten, bei  $F(\omega)$  dagegen in dessen Zähler. Je schmaler also die Verteilung  $f(t)$ , um so breiter ihre Fourier Transformierte  $F(\omega)$  und umgekehrt.

und

$$\text{si}_\pi = \begin{cases} 1 & \text{für } t = 0 \\ \frac{\sin(\pi t)}{\pi t} & \text{sonst} \end{cases} \quad \text{mit} \quad \int_{-\infty}^{\infty} \text{si}_\pi(t) dt = 1 .$$

Eine frühe Beschreibung geht auf Franciscus Vieta zu, der 1539 die Spaltfunktion als Produkt von Kosinusfunktionen definiert hat:

$$\text{si}(t) = \prod_{n=1}^{\infty} \cos\left(\frac{t}{2^n}\right) .$$

Frage: si kann, wie aus der Definition ersichtlich und auch aus Abb. 3.9 zu erkennen, negative Werte annehmen. Wie kann si dann ein Beugungsmuster beschreiben?

$F(\omega)$	$f(t)$
$2\pi \delta(\omega)$ (Deltafkt.)	1
$\pi \delta(\omega) + (i\omega)^{-1}$	$H(t)$ (Heavyside, Sprungfkt.)
1	$\delta(t)$ (Deltafkt.)
$i\pi \delta(\omega + \omega_0) - i\pi \delta(\omega - \omega_0)$	$\sin(\omega_0 t)$
$\pi \delta(\omega + \omega_0) + \pi \delta(\omega - \omega_0)$	$\cos(\omega_0 t)$
$F_s(\omega)$	$f(t)$
$\pi/2$	$t^{-1}$
$\pi/2 e^{-b\omega}$	$t/(t^2 + b^2)$
$\omega/(\omega^2 + b^2)$	$e^{-bt}$
$F_c(\omega)$	$f(t)$
$b/(\omega^2 + b^2)$	$e^{-bt}$
$\pi/(2b) e^{-b\omega}$	$1/(t^2 + b^2)$
$\sqrt{\pi/(2\omega)}$	$t^{-1/2}$

Tabelle 3.1: Fourier Transformierte wichtiger Funktionen,  $F_s$  bzw.  $F_c$  bezeichnen Fourier Sinus bzw. Fourier Kosinus Transformierte

§ **392** Für die Spektroskopen liefert die Fourier Transformation ebenfalls etwas. Der angeregte Zustand eines Kerns entvölkert sich mit der Lebensdauer  $\tau$  gemäß  $e^{-t/\tau}$ , die zugehörige Emmisionslinie wird durch eine Lorentz Funktion beschrieben. Die Funktion im Zeitbereich  $f(t)$  ist

$$f(t) = e^{-|t|/\tau},$$

d.h. eine gerade Funktion. Die Fourier Transformierte dieser beidseitigen Exponentialfunktion ist mit

$$F(\omega) = \int_{-\infty}^{\infty} e^{-|t|/\tau} e^{-i\omega t} dt = \int_{-\infty}^{\infty} e^{-|t|/\tau} \cos(\omega t) dt = \frac{2\tau}{1 + \omega^2\tau^2}$$

genau diese Lorentz Funktion im Frequenzbereich. Beide Funktionen sind in Abb. 3.10 gezeigt.

**Querverbindung 3** Warum sind Linienbreite und Lebensdauer gekoppelt? Gibt es weitere Größen mit ähnlicher Kopplung?

§ **393** Die Fourier Transformierten für viele gebräuchliche Funktionen finden sich in den gängigen Tabellenwerken wie Bronstein [9] oder Stöcker [154]; einige Beispiele sind auch in Tabelle 3.1 gegeben.

§ **394** Eine Transformation können wir uns in Anlehnung an die Fourier Reihe ohne Rückgriff auf die Transformationsgleichungen klar machen: die Winkelfunktionen Sinus und Kosinus bilden die Basisfunktionen. Eine Funktion  $\sin(\omega_0 t)$  hat im Frequenzspektrum einen einzigen Peak, eben bei  $\omega_0$ , d.h. die Transformierte einer Winkelfunktion ist eine  $\delta$ -Funktion  $\delta(\omega \pm \omega_0)$ . Die Argumentation gilt allerdings auch für  $\cos(\omega t)$ . Auch hier ergibt sich im Frequenzspektrum ein einziger Peak bei  $\omega$ . Da die Fourier Transformation eindeutig ist, muss die Phaseninformation zur Unterscheidung von Sinus und Kosinus mit geliefert werden: dazu kriegt der Sinus einen Faktor  $i$  vor die  $\delta$ -Funktion, vgl. Tabelle 3.1.

### Eigenschaften der kontinuierlichen Fourier Transformation

§ **395** Um außer der Signalanalyse noch etwas mit der Fourier Transformation – und später mit der Laplace Transformation – anfangen zu können, müssen wir uns über ihre Eigenschaften im Klaren sein:

1. die Fourier Transformation ist eine lineare Transformation:

$$\mathcal{F}\{\alpha f(t) + \beta g(t)\} = \alpha \mathcal{F}\{f(t)\} + \beta \mathcal{F}\{g(t)\} = \alpha F(\omega) + \beta G(\omega).$$

Skalierungen und additive Überlagerungen von Funktionen können also vor oder nach der Transformation ausgeführt werden, ohne dass sich das Ergebnis der Transformation ändert.

Die Linearität der Fourier Transformation lässt sich unmittelbar aus der Definition der Transformation in (3.11) begründen:

$$\begin{aligned} \mathcal{F}\{\alpha f(t) + \beta g(t)\} &= \int_{-\infty}^{\infty} \{\alpha f(t) + \beta g(t)\} e^{-i\omega t} dt \\ &= \int_{-\infty}^{\infty} \alpha f(t) e^{-i\omega t} dt + \int_{-\infty}^{\infty} \beta g(t) e^{-i\omega t} dt \\ &= \alpha \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt + \beta \int_{-\infty}^{\infty} g(t) e^{-i\omega t} dt \\ &= \alpha \mathcal{F}\{f(t)\} + \beta \mathcal{F}\{g(t)\} = \alpha F(\omega) + \beta G(\omega). \end{aligned}$$

Die Linearität der Fourier Transformation ist eine direkte Konsequenz der Linearität der Integration.

2. eine Verschiebung im Zeitbereich entspricht einer Modulation im Frequenzbereich (erster Verschiebungssatz)

$$\mathcal{F}\{f(t - \tau)\} = F(\omega) e^{-i\omega\tau}.$$

Wenn wir an ein kontinuierliches periodisches Signal denken, ist diese Aussage vielleicht auf den ersten Blick verwirrend: sollte ein verschobener Sinus sich nicht wieder auf die gleiche Frequenz abbilden? Das ist korrekt insofern, als dass sich der Betrag der Fourier-transformierten, das Frequenzspektrum, nicht ändert. Allerdings ändern sich die Real- und die Imaginärteile: gehen wir von  $f(t) = \sin(\omega_0 t)$ , so hat die Fourier-Transformierte gemäß Tabelle 3.1 nur zwei imaginäre Komponenten  $F(\omega) = \pm i\pi \delta(\omega \pm \omega_0)$ ; die um  $-\pi/2$  verschobene Funktion  $f(t) = \cos(\omega_0 t)$  dagegen hat die Fourier Transformierten  $F(\omega) = \pi \delta(\omega \pm \omega_0)$ . Im einen Fall erhalten wir nur reelle, im anderen nur imaginäre Anteile – bei einer Verschiebung um eine beliebige Phase dagegen hat die Fourier Transformierte sowohl einen reellen als auch einen imaginären Anteil, wobei das Verhältnis aus Imaginär- und Realteil wieder dem Tangens der Phase entspricht.

3. eine Modulation im Zeitbereich entspricht einer Verschiebung im Frequenzbereich (zweiter Verschiebungssatz)

$$\mathcal{F}\{f(t) e^{-i\omega_0 t}\} = F(\omega - \omega_0).$$

4. eine Streckung (Stauchung) im Zeitbereich entspricht einer Stauchung (Streckung) im Frequenzbereich (Skalierungssatz)

$$\mathcal{F}\{f(at)\} = \frac{1}{|a|} F\left(\frac{\omega}{a}\right).$$

Diese Eigenschaft ist praktisch, da sie es uns ermöglicht, mit dimensionslosen Größen zu arbeiten (d.h. insbesondere die Frequenz in Einheiten von  $\omega_0$  anzugeben bzw. die Zeitachse in Einheiten der Schwingungsdauer).

5. aus einem Faltungsintegral<sup>3</sup> wird durch Fourier Transformation das Produkt der Fourier

<sup>3</sup>Zur Erinnerung: eine Faltung ist ein mathematischer Operator, der für zwei Funktionen  $f$  und  $g$  eine dritte Funktion liefert, die die Überlappung zwischen  $f$  und einer verschobenen Version von  $g$  ist und damit einem gleitenden Durchschnitt ähnelt:

$$f(t) \otimes g(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau) d\tau$$

Eine anschauliche Deutung der Faltung ist die Gewichtung einer Funktion mit einer anderen – wir werden im Zusammenhang mit der Lösung der Systemgleichungen und der Übertragungsfunktion genauer auf diese anschauliche Interpretation eingehen. Der Funktionswert der Gewichtsfunktion  $f$  an einer Stelle  $\tau$  gibt dann

Transformierten (Faltungsregel)

$$\mathcal{F}(f(t) \otimes g(t)) = \mathcal{F} \left\{ \int_0^t f(t-\tau) h(\tau) d\tau \right\} = F(\omega) H(\omega) .$$

Das Faltungsintegral ist vielleicht die Eigenschaft der Fourier Transformation, die insbesondere in der Systemtheorie die größte rechen-technische Erleichterung bringt, da eine häufig auftretende Integration durch ein einfaches Produkt ersetzt wird.

Der Zusammenhang lässt sich unmittelbar durch Anwendung der Transformationsregeln einsehen:

$$\begin{aligned} \mathcal{F}(f(t) \star g(t)) &= \mathcal{F} \left( \int_{-\infty}^{\infty} f(\tau) g(t+\tau) d\tau \right) \\ &= \int_{-\infty}^{\infty} dt e^{i\omega t} \left( \int_{-\infty}^{\infty} f(\tau) g(t+\tau) d\tau \right) \\ &= \int_{-\infty}^{\infty} d\tau f(\tau) e^{i\omega\tau} \int_{-\infty}^{\infty} g(t-\tau) e^{-i\omega(t-\tau)} \\ &= \int_{-\infty}^{\infty} d\tau f(\tau) e^{i\omega\tau} \int_{-\infty}^{\infty} dt' g(t') e^{-i\omega t'} \quad \text{mit} \quad t' = t - \tau \\ &= F(\omega) G(\omega) . \end{aligned}$$

Die Faltungsregel liefert die Begründung dafür, dass sich die Übertragungsfunktion im Bildbereich einfach als Quotient aus Aus- und Eingangssignal bestimmen lässt: die linke Seite ist die Laplace Transformierte des Ausgangs, die rechte Seite ist ein Produkt aus der Laplace Transformierten des Faltungskerns, also der Übertragungsfunktion, mit der Laplace Transformierten des Eingangs.

Da das Faltungsintegral in allen Filtertechniken (bei Bildern sogar im zwei-dimensionalen) auftritt und eine Faltung durch die Integration (bzw. im diskreten Bereich die Summation) rechenaufwendig ist, führt z.B. MATLAB die Faltung über die Funktionen `conv` und `conv2` mit Hilfe einer Multiplikation in der Frequenzdomäne durch: das Signal wird Fourier transformiert, mit dem transformierten Kernel multipliziert und das Ergebnis wird in die Zeitdomäne zurück transformiert.

In engem Zusammenhang mit dem Faltungsintegral steht Parseval's Theorem

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} |F(\omega)|^2 d\omega , \quad (3.12)$$

das eine Beziehung zwischen der Energie des Signals im Zeitbereich und seiner Energie im Spektralbereich herstellt.

#### 6. Die Kreuzkorrelation<sup>4</sup> zwischen zwei Funktionen $f(t)$ und $g(t)$ lässt sich ebenfalls von

an, wie stark der um  $\tau$  zurückliegende Wert der gewichteten Funktion, also  $g(t-\tau)$ , in den Wert der Ergebnisfunktion an der Stelle  $t$  eingeht.

Die Faltung ist eine lineare Operation, d.h. es gilt ein Distributivgesetz

$$f(t) \otimes (g(t) + h(t)) = f(t) \otimes g(t) + f(t) \otimes h(t) .$$

Die Faltung ist ferner kommutativ,  $f(t) \otimes g(t) = g(t) \otimes f(t)$ , (Beweis durch Substitution der Integrationsvariablen) und assoziativ,  $f(t) \otimes [g(t) \otimes h(t)] = [f(t) \otimes g(t)] \otimes h(t)$  (Beweis: Doppelintegral mit Vertauschung der Integrationsreihenfolge).

<sup>4</sup>Die Kreuzkorrelation

$$f(t) \star g(t) = \int_{-\infty}^{\infty} f(\tau) g^*(t+\tau) d\tau$$

wird verwendet, um zu untersuchen, ob eine (gemessene) Funktion  $f(t)$  irgendetwas gemeinsam hat mit einer anderen gemessenen Funktion  $g(t)$ . Der  $*$  bedeutet hier das konjugiert komplexe. Das Integral ähnelt dem in der Faltung, bis auf das Vorzeichen der Verschiebung (+ statt -) und eben der Verwendung der konjugiert komplexen Funktion. Die Kreuzkorrelation ist distributiv und assoziativ, jedoch nicht kommutativ.

einem Integral in ein Produkt transformieren:

$$\mathcal{F}(f(t) \star g(t)) = \mathcal{F}\left(\int_{-\infty}^{\infty} f(\tau)g^*(t + \tau) d\tau\right) = F(\omega) G^*(\omega).$$

7. Die Autokorrelation<sup>5</sup>, d.h. die Kreuzkorrelation einer Funktion mit sich selbst, wird entsprechend gebildet und entspricht dem Quadrat der Fourier-Transformierten:  $f(t) \star f(t) = F(\omega) F^*(\omega) = |F(\omega)|^2$ .
8. die zeitliche Ableitung entspricht einer Multiplikation der Fourier Transformierten mit  $i\omega$ :

$$\mathcal{F}\left\{\frac{df(t)}{dt}\right\} = i\omega F(\omega).$$

Der Beweis erfolgt durch partielle Integration:

$$\mathcal{F}\left(\frac{df}{dt}\right) = \int_{-\infty}^{\infty} f'(t) e^{i\omega t} dt = [f(t) e^{i\omega t}]_{-\infty}^{\infty} - (-i\omega) \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt = i\omega F(\omega).$$

9. entsprechend lässt sich bei einer Funktion  $f(x, t)$  die möglicherweise auftretende räumliche Ableitung durch Multiplikation der Fourier Transformierten mit  $ik$

$$\mathcal{F}\left\{\frac{df(x, t)}{dx}\right\} = ik F(\omega) \tag{3.13}$$

bestimmen. Betrachten wir nur die räumliche Ableitung. Diese können wir entsprechend der zeitlichen Ableitung behandeln – allerdings müssen wir auf die geeigneten Variablen im Ausgangs- und im Bildbereich achten. Zur Zeit  $t$  gehört im Bildbereich die Frequenz  $\omega$ , zur Lineardimension Strecke  $x$  die Wellenzahl  $k$ . Die Fourier Transformation lässt sich also auf verschiedenen Variablenpaare anwenden, diese müssen jedoch jeweils zueinander konjugiert sein.<sup>6</sup>

10. die Anwendung des Nabla Operators  $\nabla$  auf eine von  $\vec{r}$  abhängige skalare bzw. vektorwertige Funktion entspricht daher der Multiplikation der Fourier Transformierten mit  $i\vec{k}$ :

$$\mathcal{F}\{\nabla f\} = i\vec{k} F, \quad \mathcal{F}\{\nabla \cdot \vec{f}\} = i\vec{k} \cdot \vec{F} \quad \text{und} \quad \mathcal{F}\{\nabla \times \vec{f}\} = i\vec{k} \times \vec{F},$$

### Fourier Transformation und harmonischer Oszillator

§ 396 Betrachten wir unter diesem Gesichtspunkt nochmals den harmonischen Oszillator. Die Bewegungsgleichung ist  $\ddot{x} + \omega_0^2 x = 0$ . Die Differentialgleichung ist eine Bestimmungsgleichung für die Funktion  $x(t)$  im Zeitbereich. Transformieren wir die DGL schrittweise in den Frequenzbereich:

$$\mathcal{F}\{\ddot{x} + \omega_0^2 x\} = 0.$$

Da die Fourier Transformation linear ist, können die Terme auf der linken Seite einzeln transformiert werden:

$$\mathcal{F}\{\ddot{x}\} + \mathcal{F}\{\omega_0^2 x\} = 0.$$

---

<sup>5</sup>Die Autokorrelation

$$f(t) \star f(t) = \int_{-\infty}^{\infty} f(\tau) f^*(t + \tau) d\tau$$

ist formal die Kreuzkorrelation einer Funktion mit sich selbst. Optisch lässt sich damit das Signal-zu-Rausch-Verhältnis im Frequenzspektrum aufpäppeln, allerdings um den Verlust der Linearität. Bei stochastischen Prozessen dagegen hat die Autokorrelation ihre Bedeutung – anschaulich wird sie z.B. in der Berechnung des mittleren Abstandsquadrats im Random Walk. Als simple Merkregel: die Autokorrelation ist die Korrelation einer Folge von gleichartigen Zufallsvariablen; die Kreuzkorrelation lässt sich entsprechend ebenfalls für Zufallsvariablen bestimmen, gibt dann aber die Korrelation zwischen verschiedenen Merkmalen.

<sup>6</sup>Falls Sie immer noch über Querverweis 3 grübeln, sollte dies eigentlich der Hinweis gewesen sein.



Die zeitliche Ableitung bringt jeweils einen Faktor  $i\omega$  vor die Fourier Transformierte, bei der zweiten zeitlichen Ableitung ist der Vorfaktor daher  $(i\omega)^2 = -\omega^2$ :

$$-\omega^2 X(\omega) + \omega_0^2 X(\omega) = 0 \quad \Rightarrow \quad \omega^2 = \omega_0^2 \quad \Rightarrow \quad \omega = \pm\omega_0 .$$

Die DGL wird im Frequenzbereich nur für zwei Frequenzen  $\omega_0$  und  $-\omega_0$  gelöst. Das entspricht den bereits bekannten Eigenfrequenzen des harmonischen Oszillators. Das Frequenzspektrum  $X(s)$  lässt sich daher als eine Kombination von zwei  $\delta$ -Funktionen darstellen,  $\delta(\omega - \omega_0)$  und  $\delta(\omega + \omega_0)$ . Diese transformieren gemäß Tabelle 3.1 auf die Winkelfunktionen zurück, so dass wir auch auf diese Weise das bereits bekannte Ergebnis erhalten.

**Verständnisfrage 38** Ist die Darstellung korrekt? Was erlaubt die Einführung der imaginären Anteile, um neben dem Kosinus auch den Sinus als Lösung zu erhalten?

### 3.2.2 Diskrete Fourier Transformation

§ 397 Die diskrete Fourier Transformation bildet eine periodische Zahlenfolge  $\{f_k\}$  auf eine Fourier Transformierte Zahlenfolge  $\{F_l\}$  ab. Die zeitkontinuierliche Fourier Transformation ist daher für analytische Beschreibungen hilfreich sowie in der analogen Signalverarbeitung. Digitale Signale dagegen liegen nicht als zeitkontinuierliche sondern auf Grund der endlichen Dauer der Sampling Intervalle immer als diskrete Signale vor.

§ 398 Ohne die Phantasie zu sehr überstrapazieren zu müssen, können wir fast vermuten, dass bei der diskreten Fourier Transformation die Integrale in den Transformationsgleichungen (3.6) durch Summen ersetzt werden. Das hat die Nebenwirkung, dass diese Transformationen numerisch bestimmt werden können – für die digitale Signalverarbeitung ein großer Vorteil.<sup>7</sup> Ein typischerweise verwendeter Algorithmus zur diskreten Fourier Transformation ist die Fast Fourier Transform FFT.

§ 399 Die diskrete Fourier Transformation können wir analog zur kontinuierlichen Fourier Transformation einführen. Die Fourier Reihe war für eine Funktion  $f(t)$  definiert, die periodisch mit  $T$  in  $t$  ist. In der komplexen Darstellung werden ihre Koeffizienten in die der diskreten Fourier Reihe überführt durch

$$c_n = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-2\pi i n t / T} dt \quad \rightarrow \quad c_n = \frac{1}{N} \sum_{k=0}^{N-1} f_k e^{-2\pi i n k / N} .$$

Für die diskrete Fourier Reihe ist nicht die Periode  $T$  relevant sondern die Zahl  $N$  der Sampling Intervalle der Länge  $\Delta t$  innerhalb einer Periode, d.h.  $T = N \Delta t$ . Statt der Integration von  $-T/2$  bis  $+T/2$  ist hier das Integrationsintervall verschoben, so dass die Summation bei Null startet. Das ist irrelevant, wichtig ist nur, dass über eine vollständige Periode integriert wird. Dazu muss die Integration nur bis  $N - 1$  laufen, da hier das letzte Sampling Intervall beginnt und damit an dessen Ende die Periode abgeschlossen ist.

§ 400 Mit diesen Vorüberlegungen können wir die diskrete Fourier Transformation direkt angeben:

$$F_n = \frac{1}{N} \sum_{k=0}^{N-1} f_k e^{-i k w_0 n} \quad \text{mit} \quad w_0 = \frac{2\pi}{N} .$$

Für die inverse Fourier Transformation gilt entsprechend:

$$f_n = \sum_{k=0}^{N-1} F_k e^{i k w_0 n} .$$

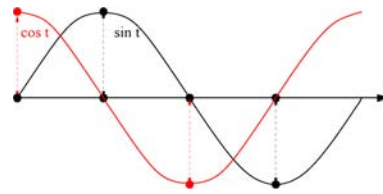


Abbildung 3.11: Diskrete Fourier Transformation Winkelfunktionen

### Beispiele

§ 401 Betrachten wir als Beispiele die Winkelfunktionen mit einer etwas reduzierten Abtastrate von 4 Punkten pro Periode, vgl. Abb. 3.11. Für den Kosinus (rote Kurve) erhalten wir  $N = 4$  Stützstellen mit den Funktionswerten

$$f_0 = 1, \quad f_1 = 0, \quad f_2 = -1 \quad \text{und} \quad f_3 = 0.$$

Eine weitere Stützstelle benötigen wir nicht, da  $f_4 = f_0$  bzw. allgemein  $f_n = f_{n+4}$ . Für die Fourier Koeffizienten ergibt sich  $F_0 = 0$ , da der Mittelwert Null ist (Gleichstromanteil). Die anderen Koeffizienten sind

$$\begin{aligned} F_1 &= \frac{1}{4} (1 + (-1)(-1)) = \frac{1}{2} \\ F_2 &= \frac{1}{4} (1 + (-1)1) = 0 \\ F_3 &= \frac{1}{4} (1 + (-1)(-1)) = \frac{1}{2}. \end{aligned}$$

Hier sind zwei Koeffizienten,  $F_1$  und  $F_3$  von Null verschieden – wir erhalten zwei Peaks im Frequenzspektrum, obwohl die Eingangsfunktion nur durch eine Grundfrequenz  $\omega_1 = 2\pi \frac{1}{4} \Delta t$  bestimmt ist. Die zweite spektrale Komponente entspricht  $-\omega_1$ , sie wird bei der Entwicklung ‘herum gewrapped’: statt das Frequenzspektrum symmetrisch um Null im Intervall von  $F_{-2}$  bis  $F_2$  zu bestimmen, erhalten wir ein Frequenzspektrum im Intervall von  $F_0$  bis  $F_4$  derart, dass die negativen Frequenzen vom rechten Intervallrand nach links absteigend bis zur Intervallmitte lokalisiert sind, siehe auch obere Zeile in Abb. 3.12. Bei reellem Input gilt allgemein

$$F_{N-j} = F_j.$$

Bei geradem Input steht daher in der rechten Intervallhälfte das selbe wie in der linken; bei ungeradem Input dagegen dessen konjugiert komplexes (bzw. bei reellen Größen das Negative). Addition von  $F_1$  und  $F_3 = F_1$  ergibt dann, wie von der Normierung gefordert, 1:

$$f_k = 1 \cos\left(\frac{2\pi}{4}k\right).$$

§ 402 Überprüfen wir das am Sinus (schwarze Kurve in Abb. 3.11). Die Funktionswerte an den Stützstellen sind

$$f_0 = 0, \quad f_1 = 1, \quad f_2 = 0 \quad \text{und} \quad f_3 = -1.$$

Daraus ergeben sich die folgenden Fourier-Koeffizienten

$$F_0 = 0, \quad F_1 = -\frac{i}{2}, \quad F_2 = 0 \quad \text{und} \quad F_3 = \frac{i}{2}.$$

Auch hier verschwindet der Gleichstromanteil (Mittelwert)  $F_0$ . Das unterschiedliche Vorzeichen berücksichtigen wir dadurch, dass wir die negative Frequenz mit  $-1$  multiplizieren. Die Summe wird daher  $F_1 + (-1)F_3 = -i$ . Da die Rücktransformation einen weiteren Faktor  $i$  liefert, ergibt sich die Intensität der Sinuswelle zu

$$f_k = i(F_1 + (-1)F_3) = \sin\left(\frac{2\pi}{4}k\right).$$

<sup>7</sup>Oder formulieren Sie es anders: den Übergang zur Summe müssten wir in einem numerischen Verfahren immer machen – das ist ja gerade die Bedeutung der numerischen Integration.

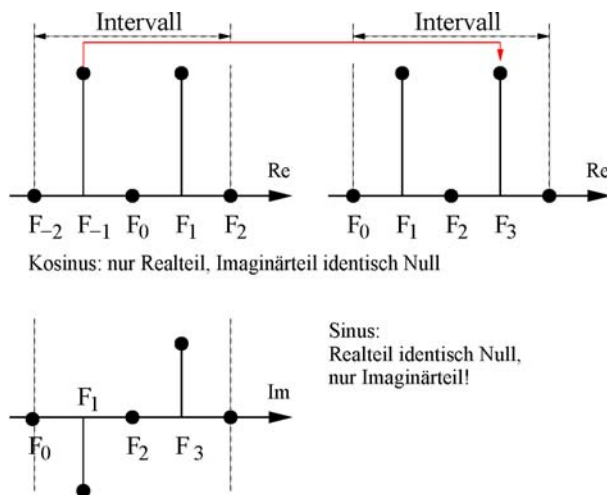


Abbildung 3.12: Die Fourier-Koeffizienten für den Kosinus (*oben*) sind reell, die für den Sinus (*unten*) imaginär

**Zwischenrechnung 23** Auch wenn es etwas mühsam ist, führen Sie das ganze Verfahren noch einmal mit der doppelten Zahl von Stützstellen durch.

**Zwischenrechnung 24** Wenden Sie das Verfahren ferner (wieder mit einer geringen Zahl von Stützstellen) auf die bereits aus § 369 bekannte Funktion  $f(t) = \cos^2(\omega t)$  an.

### Eigenschaften der diskreten Fourier Transformation

§ 403 Die Eigenschaften der diskreten Fourier Transformation entsprechen im wesentlichen denen der kontinuierlichen Fourier Transformation – lediglich in der formalen Darstellung werden die Integrale durch Summen ersetzt. Außerdem sind die Ableitungen nicht mehr sinnvoll, da diese für den Grenzwert unendlich kleiner Differenzen definiert sind während die diskrete Fourier Transformation ja gerade auf endlichen Differenzen basiert.

§ 404 Der in der kontinuierlichen Fouriertransformation auftretende Faktor  $e^{i\omega t}$  lässt sich für diskrete Zeiten  $t_k = k \Delta t$  mit  $k = 0, 1, \dots, N-1$  und  $T = N \Delta t$  schreiben als

$$e^{i\omega t} = e^{i \frac{2\pi t}{T}} = e^{\frac{2\pi i k}{N}} = W_N^k.$$

Mit dem so definierten Kern(el)

$$W_N = e^{\frac{2\pi i}{N}}$$

lassen sich die Eigenschaften der Fourier Transformation einfach darstellen:

1. die diskrete Fourier Transformation ist eine lineare Operation:

$$\mathcal{F}\{\alpha \{f_n\} + \beta \{g_n\}\} = \alpha \{F_n\} + \beta \{G_n\}.$$

2. eine Verschiebung im Zeitbereich entspricht einer Multiplikation mit dem Phasenfaktor im Frequenzbereich (Verschiebungssatz im Zeitbereich):

$$\mathcal{F}\{f_{k-j}\} = \mathcal{F}\{f_k W_N^{-kn}\} = F_k W_N^{kn} \quad n \in \mathbb{Z}.$$

3. eine Modulation in der Zeitdomäne mit dem Phasenfaktor entspricht einer Verschiebung im Frequenzbereich (Verschiebungssatz im Frequenzbereich):

$$\mathcal{F}\{f_k W_N^{-kn}\} = F_{k-n} \quad n \in \mathbb{Z}.$$

4. Durch das Wrappen der Fourierkoeffizienten entspricht der mittlere Koeffizient der höchsten Frequenz  $\omega_{\max}$  bzw.  $-\omega_{\max}$ . Diese Nyquist Frequenz [109] oder Abschneidefrequenz

$$\omega_N = \frac{\pi}{\Delta t} = 2\omega_{\max}$$

stellt eine Beziehung zwischen der Sampling-Rate  $\Delta t$  und der höchsten, bei dieser Samplingrate noch zu rekonstruierenden Frequenz dar. Das Shannon'sche Abtasttheorem [145] verwendet diesen Begriff:

$$f_{\text{abtast}} < 2f_{\text{max}} \quad \text{bzw.} \quad f_{\text{abtast}} > 2(f_{\text{max}} - f_{\text{min}}) .$$

Die erste Gleichung bezieht sich auf den Fall einer Grundfrequenz gleich Null, die zweite auf den Fall einer höheren Grundfrequenz  $f_{\text{min}}$ .<sup>8</sup> Der durch die Nyquist-Frequenz beschriebene Zusammenhang wird auch als Skalierungssatz bezeichnet, da sich durch die Wahl von  $\Delta t$  bei fester Zahl  $N$  von Samples die Zeitachse und damit die Frequenzen skalieren lassen.

5. Faltung, Auto- und Kreuzkorrelation werden ebenfalls analog zur kontinuierlichen FT definiert, allerdings muss die Zahl der Samples in beiden Funktionen gleich sein – gegebenenfalls durch Auffüllen mit Nullen. Dieses Zero Padding ist ein interessantes Problem für sich, soll uns aber hier nicht weiter beschäftigen.

### 3.2.3 Fast Fourier Transformation FFT

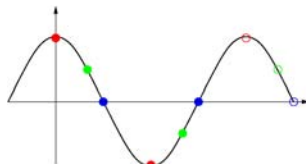
§ 405 In der Signalanalyse liegt die Funktion  $f(t)$  nicht in analytischer Form sondern als Wertetabelle vor, d.h. hier kann nur die diskrete Fourier Transformation verwendet werden. Die schnelle Fourier Transformation (Fast Fourier Transform FFT) ist keine neue Form der Fourier Transformation sondern nur Algorithmus zur schnellen Berechnung einer diskreten Fourier Transformation. Ein Teil des Algorithmus geht bereits auf Gauß zurück, als entscheidendes Paper zur Einführung des Algorithmus gilt Cooley und Tukey [15].

§ 406 Eine Zahlenfolge mit nur einer reellen Zahl ( $N = 1$ )  $f_0$  hat eine ganz simple Fourier Transformierte:  $F_0 = \frac{1}{1} f_0 W_1^{-0} = f_0$ . Oder, wie von Butz [10] als Antwort eines Philosophen formuliert:

Wir wissen, dass die Fouriertransformierte einer  $\delta$ -Funktion eine Konstante ergibt und umgekehrt. Wie stellt man eine Konstante in der Welt der eingliedigen Zahlenfolgen dar? Durch die Zahl  $f_0$ . Wie stellt man in dieser Welt eine  $\delta$ -Funktion dar? Durch diese Zahl  $f_0$ . Es verschwindet also in dieser Welt der Unterschied zwischen einer Konstanten und einer  $\delta$ -Funktion. Fazit:  $f_0$  ist seine eigene Fouriertransformierte.

§ 407 Die FFT ist ein Teile-und-Herrsche-Verfahren, bei dem man durch geschickte Halbierung des Inputs zu eine Folge der Länge  $N = 1$  gelangen will, da man sich dann die

<sup>8</sup>Das Abtasttheorem besagt letztendlich nur, dass ein periodisches Signal mit mehr als der doppelten der maximal auftretenden Frequenzen abgetastet werden muss. Das Abtasttheorem wird manchmal fälschlicherweise mit einem  $\leq$  statt eines  $<$  angegeben, d.h. es wird angenommen, dass man mit zwei Samples pro Periode auskommt. Das ist nicht korrekt, da das Ergebnis der Abtastung dann von der Phase des Signals abhängt:



Führt die Phasenlage auf die roten Punkte als Abtaststellen, so lässt sich mit den zwei Samples pro Periode scheinbar (s.u.) ein Kosinus rekonstruieren. Bei einer Phasenverschiebung um  $\pi/2$  (blaue Punkte, das entspricht der Abtastung eines Sinus oder genauer eines invertierten Sinus), dagegen verschwindet die Amplitude an allen Abtaststellen; entsprechend würde das Signal als Null interpretiert. Bei einer Phasenlage dazwischen (grüne Punkte) wird die Aussage 'scheinbar' beim Kosinus deutlich: hier verschwindet die Amplitude nicht, allerdings lässt sich ohne die zusätzliche Angabe einer Phase das Signal nicht eindeutig rekonstruieren: man könnte schließlich auch eine Phasenverschiebung annehmen und die Werte an den Abtaststellen als die maximale Amplitude interpretieren – so, wie wir umgekehrt bei den roten Punkten ohne Kenntnis der Phase nicht wissen, ob wirklich im Maximum abgetastet wurde.

In der Praxis verwendet man eine modifizierte Formel  $f_{\text{abtast}} \approx 2.2 f_{\text{max}}$  – daher wird eine Audio-CD, mit der Frequenzen bis 20 kHz übertragen werden sollen, auch mit 44.1 kHz abgetastet; ebenso sind die Samplingstufen bei der MP3 Codierung an Vielfache dieser 44.1 kHz angelehnt.

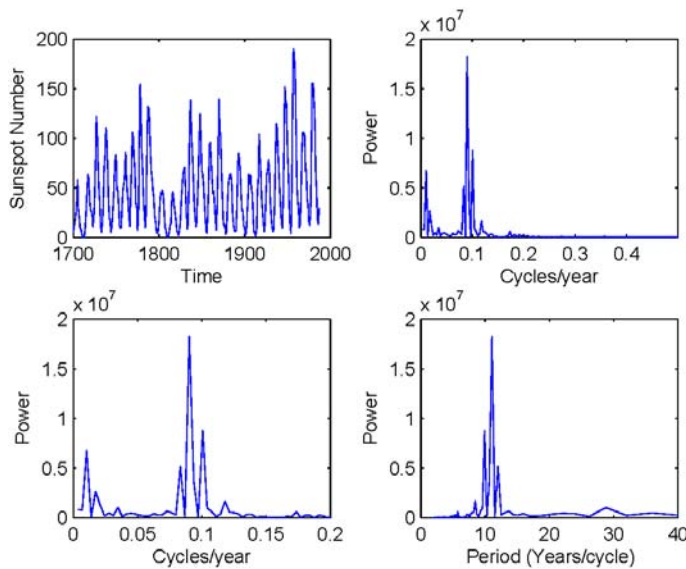


Abbildung 3.13: Zeitserie der Sonnenfleckenzahl (*oben links*), Periodogramm (Frequenzspektrum) (*oben rechts* und *unten links*) in verschiedenen Bereichen, sowie Leistungsdichtespektrum aufgetragen über Der Periode statt der Frequenz (*rechts unten*)

Bestimmung der Fourier Transformaten ersparen kann. Daher muss die Zahl der Abtastpunkte eine Zweierpotenz sein. Diese Folge von  $N = 2^p$  Messwerten zerlegen wie in zwei Folgen mit je  $2^{p-1}$  Messwerten, von denen die eine Folge nur die geraden, die andere nur die ungeraden Indizes enthält:

$$\begin{aligned} \{f_{1,k}\} &= \{f_{2k}\} \\ \{f_{2,k}\} &= \{f_{2k+1}\} \end{aligned} \quad k = 0, 1, 2, \dots, M-1 \quad \text{und} \quad M = N/2$$

mit den Fourier Transformaten

$$F_{1,j} = \frac{1}{M} \sum_{k=1}^{M-1} f_{1,k} W_M^{-kj} \quad \text{und} \quad F_{2,j} = \frac{1}{M} \sum_{k=1}^{M-1} f_{2,k} W_M^{-kj} \quad \text{mit} \quad j = 0, 1, \dots, M-1.$$

§ 408 Die Fourier Transformierte der Ausgangsfolge dagegen ist

$$F_j = \frac{1}{N} \sum_{k=0}^{N-1} f_k W_N^{-kj} = \frac{1}{N} \sum_{k=0}^{M-1} f_{2k} W_N^{-2kj} + \frac{1}{N} \sum_{k=0}^{M-1} f_{2k+1} W_N^{-(2k+1)j}.$$

Manipulation des Kerns  $W_M$  erlaubt seine Darstellung mit Hilfe von  $W_N$ , so dass sich der Ausdruck auch schreiben lässt als die (gewichtete) Summe der Fourier Transformaten der beiden Teilreihen.

$$F_j = \frac{1}{N} \sum_{k=0}^{M-1} f_{1,k} W_M^{-kj} + \frac{W_N^j}{N} \sum_{k=0}^{M-1} f_{2,k} W_M^{-kj} = \frac{1}{2} F_{1,j} + \frac{W_N^{-j}}{2} F_{2,j}.$$

Dieses Verfahren wird rekursiv bis  $N = 1$  weiter geführt. Während die normale Fourier Transformation  $N^2$  Rechenoperationen benötigt, reduziert sich deren Zahl bei der FFT auf  $pN = N \log_2(N)$  Operationen.

### 3.2.4 FFT in MATLAB

§ 409 MATLAB kennt für die FFT eine Funktion `FFT`. Diese kann auf einen Vektor oder auf eine Matrix angewendet werden – in letztem Fall werden die Fourier Koeffizienten für die einzelnen Spalten der Matrix bestimmt. Einen eigenen Block für FFT in SIMULINK gibt es nicht; auch im einzigen SIMULINK bezogenen Beispiel für die FFT erfolgt letztere wieder in MATLAB.<sup>9</sup> SIMULINK verfügt jedoch für den schnellen Zugriff (ohne die Möglichkeit, die

<sup>9</sup>Die Beschränkung der FFT auf MATLAB stellt keine wirkliche Einschränkung für SIMULINK dar, da komplexere Probleme ohnehin aus dem Workspace bzw. einem m-File gesteuert werden – und dort FFT verfügbar ist.

Ergebnisse in den Workspace oder eine Datei zu exportieren) über den Block `Power Spectral Density`, siehe auch Abschn. 3.2.5.

§ 410 Das einfachste Anwendungsbeispiel können wir aus der MATLAB Hilfe importieren: wir bestimmen die Periode der Sonnenflecken, den Solarzyklus. Die notwendige Datei liegt in MATLAB vor, das Codefragment

```
01 load sunspot.dat
02 year = sunspot(:,1); wolfer = sunspot(:,2);
03 subplot(2,2,1),plot(year,wolfer); xlabel('Time'); ylabel('Sunspot Number')
```

liefert den linken Teil von Abb. 3.13. Die Fourier Transformation erfolgt mit

```
04 Y = fft(wolfer);
```

Das Ergebnis ist ein komplexer Vektor, sein Quadrat gibt das Leistungsdichtespektrum in Abhängigkeit von der Frequenz. Da bei den Eingangsdaten die Zeitachse vor der Fourier Transformation entfernt wurde, ist die Frequenz in diesem Periodogramm ebenfalls nicht enthalten sondern gemäß des Skalierungssatz als normierte Größe durch den Index des entsprechenden Vektorelements gegeben. Vor der Darstellung muss daher ein passender Vektor für die  $x$ -Achse erzeugt werden. Außerdem lassen wir die erste Komponente bei der Darstellung weg: sie enthält einen großen Wert (Summe der Daten), der die eigentlich interessanten Fourier Koeffizienten in der graphischen Darstellung nur unterdrücken würde. Mit diesen Einschränkungen lässt sich die graphische Darstellung des Periodogramms (rechts oben in Abb. 3.13) mit folgenden MATLAB-Fragment erzeugen:

```
05 N = length(Y); Y(1) = []; nyquist = 1/2; freq = (1:N/2)/(N/2)*nyquist;
06 power = abs(Y(1:N/2)).^2;
07 subplot(2,2,2),plot(freq,power);xlabel('Cycles/year');ylabel('Power')
```

§ 411 Im Periodogramm im rechten oberen Teilbild von Abb. 3.13 ragt der ungefähr 11-jährige Solarzyklus als dominanter Peak heraus, die Struktur an seiner Basis zeigt aber auch, dass der Solarzyklus nicht wie Uhrwerk abläuft sondern seine Dauer Schwankungen unterworfen ist.<sup>10</sup> Ein zweiter Peak mit ca. 1/3 der maximalen Power findet sich bei ca. 0.01 Zyklen pro Jahr (vgl. auch die Vergrößerung im linken unteren Teil von Abb. 3.13): mit diesem Gleissberg-Zyklus ist die solare Aktivität moduliert.

§ 412 Ein genauer Blick zeigt, dass der Ausgangsdatensatz 288 Samples umfasst – was keine Zweierpotenz ist. Der Cooley–Tukey-Algorithmus ist für Zweierpotenzen formuliert, Datensätze mit anderer Länge können manchmal auf einen Cooley–Tukey-Algorithmus reduziert werden. Ist das nicht möglich, so stehen verschiedene andere Algorithmen zur Verfügung. Einige Hinweise dazu finden Sie z.B. in der MATLAB-Hilfe, bei Wikipedia unter [http://de.wikipedia.org/wiki/Schnelle\\_Fourier-Transformation](http://de.wikipedia.org/wiki/Schnelle_Fourier-Transformation) bzw. [http://en.wikipedia.org/wiki/Fast\\_Fourier\\_transform](http://en.wikipedia.org/wiki/Fast_Fourier_transform) und natürlich in den Numerical Recipes [125].

### Überprüfung mit selbsterzeugten Datensätzen

§ 413 Erzeugen wir mit Hilfe einer Fourier Reihe ein Rechtecksignal wie in Abb. 3.7, jedoch über einen längeren Zeitraum (linkes Bild in Abb. 3.14). Auf dieses Signal wenden wir die FFT aus MATLAB an und erhalten das im rechten Teil von Abb. 3.14 gezeigte Frequenzspektrum. Die wichtigsten Aspekte des Eingangsspektrums werden rekonstruiert: (a) die Power der Grundfrequenz ist am höchsten; (b) mit zunehmender Frequenz nimmt die Power ab; (c) die Abnahme der Power lässt sich grob qualitativ mit  $1/f$  annähern, d.h. in Einheiten der Grundfrequenz  $f_0$  ergibt sich  $1/n$ ; (d) es treten nur die ungeraden Vielfachen der Grundfrequenz auf; (e) die im Periodogramm auftretenden Frequenzen erstrecken sich bis zum 17-fachen der

<sup>10</sup>Die Verbreiterung eines Peaks ist allerdings nicht nur durch die Streuung der Periode um einen Wert bedingt sondern auch durch die endliche Auflösung in einem diskreten Datensatz: Sampling Rate und Länge des Datensatzes bestimmen die auftretenden diskreten Frequenzen – die 'reale' Frequenz muss aber nicht auf einer dieser Stützstellen liegen, vgl. auch Abb. 3.14f.

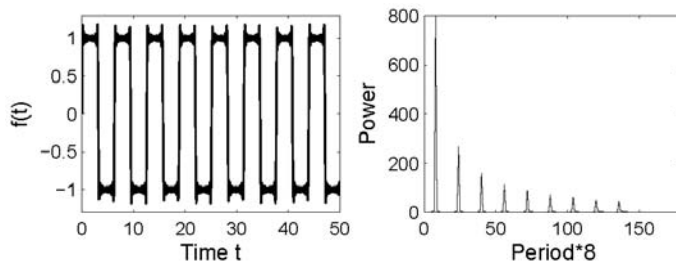


Abbildung 3.14: Rechteck als Fourier Reihe (*links*) und FFT dieses Signals (*rechts*)

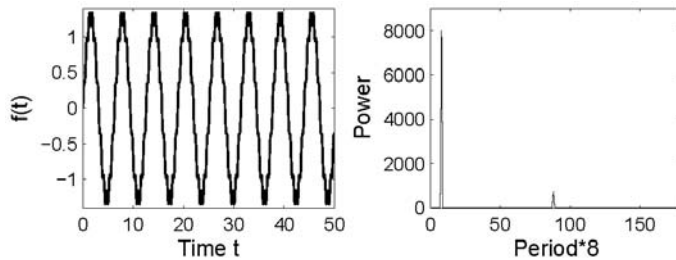


Abbildung 3.15: Reduziertes Rechteck und FFT; wie Abb. 3.14, jedoch wurde bei der Konstruktion der Funktion nur die Grundfrequenz und das 11-fache der Grundfrequenz verwendet

Grundfrequenz – mehr Frequenzen wurden auch nicht zur Konstruktion des Rechtecks verwendet. Die Peaks im Frequenzspektrum sind nicht, wie theoretisch erwartet, scharf sondern verschmiert: die Stützstellen in der Periode sind durch die Zeitaufösung und die Länge des Datensatzes bestimmt, nicht aber durch die Periode des Eingangssignals. Letztere liegen zwischen den Stützstellen, so dass das numerische Verfahren den  $\delta$ -Peak über die benachbarten Stützstellen verschmiert.

§ 414 Abbildung 3.15 entspricht Abb. 3.14, jedoch wurden bei der Konstruktion des Eingangssignals nur die Grundfrequenz und ihr 11-faches verwendet. Entsprechend ist das Periodogramm auf diese beiden Frequenzen beschränkt.

§ 415 Und wie sieht es mit einem realen Rechteck aus? Dieses können wir aus Sprüngen zusammensetzen und erhalten, wie in Abb. 3.16 gegeben, ein Frequenzspektrum, dessen oberste Frequenz nur noch durch die Zeitaufösung (Sampling Rate) der Eingangsfunktion begrenzt ist.

**Zwischenrechnung 25** Die Idee sollte klar geworden sein – Spielen Sie zur Übung, zum Vertrauensgewinn und zum Training einer saubereren Darstellung als in Abb. 3.14–3.16 mit weiteren Funktionen. Spielen Sie auch mit anderen Funktionen wie der in § 377f diskutierten nicht periodischen Funktion.

### Fensterln mit Fourier

§ 416 Während die Zeitaufösung des Datensatzes die höchste zu detektierende Frequenz bestimmt, bestimmt die Länge des Datensatzes die niedrigste Frequenz. Zusätzlich bestimmt die Länge des Datensatzes aber auch, wie genau diese Frequenz detektiert werden kann; sie kann

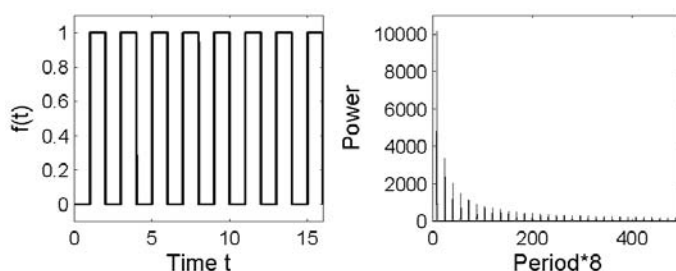


Abbildung 3.16: ‘Echtes’ Rechteck und FFT; wie Abb. 3.14, jedoch wurde das Rechteck nicht über die Fourier-Reihe konstruiert sondern aus Sprungfunktionen zusammengesetzt

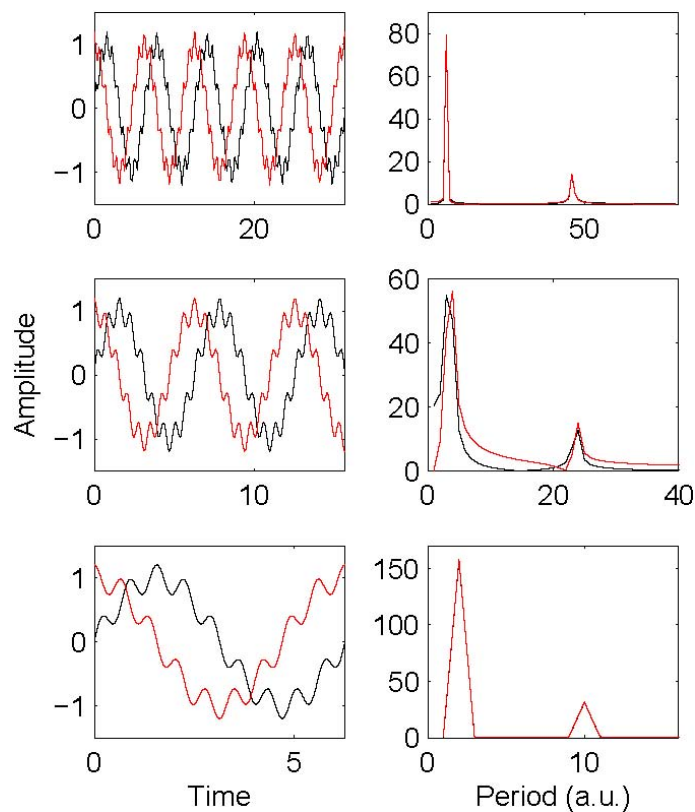


Abbildung 3.17: Periodogramm (*rechts*) der Funktion  $f(t) = \sin(t) + 0.2 \sin(9t)$  (schwarz) bzw.  $f(t) = \cos(t) + 0.2 \cos(9t)$  (rot) für verschiedene Längen des Datensatzes (*links*)

auch numerische Artefakte bis hin zu Peaks bei nicht existierenden Frequenzen einführen. Dazu zeigt Abb. 3.17 für verschieden lange Abschnitte der Funktionen  $f(t) = \sin(t) + 0.2 \sin(9t)$  (schwarz) bzw.  $f(t) = \cos(t) + 0.2 \cos(9t)$  (rot; jeweils linke Teilbilder) die zugehörigen Periodogramme. Ist die Fensterlänge ein Vielfaches der Periode (oberes und unteres Bild), so sind die Peaks im Periodogramm scharf und für die Sinus und Kosinus Variante identisch. Im mittleren Teilbild dagegen ist ein Fenster mit der 2.5fachen Periodenlänge gewählt. Die Peaks liegen an den richtigen Stellen, aber (a) sie sind breiter als durch die Auflösung des numerischen Verfahrens erfordert, (b) die Periodogramme für Sinus und Kosinus unterscheiden sich und (c) die unterste Frequenz (Null) beim Sinus verschwindet nicht.

§ 417 Der letzte Unterschied ist am einfachsten zu verstehen: der erste Wert im Periodogramm gibt den Gleichstromanteil und damit den Mittelwert des betrachteten Datensatzes. Dieser verschwindet beim Kosinus (rote Kurve) auch für ganzzahlige Vielfache der Hälfte der Periodenlänge, beim Sinus jedoch nur für ganzzahlige Vielfache der Periodenlänge. Dieser nicht verschwindende Gleichstromanteil enthält einen Teil der Leistung des gesamten Signals, so dass sich das Periodogramm auch in den anderen Teilen von dem des Kosinus unterscheiden muss.

§ 418 Insbesondere fallen beim Kosinus im Leistungsdichtespektrum die hohen Frequenzen auf: das Spektrum fällt zu diesen nur recht langsam ab während es beim Sinus schnell abfällt. Der langsame Abfall zu hohen Frequenzen wäre verständlich, wenn wir statt der Winkelfunktion eine Funktion mit einem Sprung oder zumindest einem sehr steilen Anstieg betrachten würden, wie z.B. das Rechteck in Abb. 3.16. Dieser Sprung ist in der zu analysierenden Funktion  $f(t) = \cos(t) + 0.2 \cos(9t)$  mit  $t \in [0, 5\pi]$  aus unserer Sicht nicht vorhanden – aus der Sicht eines Algorithmus zur Fourier Transformation dagegen schon. In § 379 haben wir gesehen, dass sich der Informationsgehalt eines endlichen Signals nicht verändert, wenn es beliebig oft wiederholt wird. Bei der Fourier Transformation wird aus dem nicht zwingend-periodischen Signal eine periodische Funktion erzeugt, in dem dieses Signal wiederholt wird.



§ 419 Das wiederholte Aneinanderhängen ist während der FFT bei allen Signalen in Abb. 3.17 erfolgt – die hohen Frequenzen treten nur in einem der dort verwendeten sechs Signale auf. Warum? In den oberen und unteren Datensätzen ist der betrachtete Zeitraum jeweils ein ganzzahliges Vielfaches der Periodendauer der Grundschwingung – damit schließt diese aber beim aneinander hängen stetig und sogar differenzierbar an. Nur im mittleren Teilbild werden die hohen Frequenzen benötigt. Besonders stark von der Kosinus-Variante, da diese beim aneinander hängen einen Sprung bewirkt – entsprechend der Rechteckfunktion werden dann die hochfrequenten Anteile zur Rekonstruktion des Signals benötigt. In der Sinus-Variante werden wesentlich weniger hochfrequente Anteile im Leistungsdichtespektrum benötigt, da die Funktion beim aneinander hängen keinen Sprung aufweist. Einige hochfrequente Anteile werden aber dennoch benötigt, da der Übergang zwar stetig aber nicht differenzierbar ist. Bei beiden Funktionen fallen die hochfrequenten Anteile vollkommen weg, wenn nur Vielfache der Periodenlänge betrachtet werden – und dabei ist es egal, ob eine (unten) oder mehrere (oben) Perioden im Signal enthalten sind. Dieses diffundieren von Leistung von den eigentlichen Frequenzen wird als leakage bezeichnet, das Gegenmittel als windowing.

**Zwischenrechnung 26** Spielen Sie noch mit anderen Datensätzen und unterschiedlichen Fenstern um ein Gefühl für diese leakage zu kriegen.

§ 420 Im Gegensatz zu den willkürlich erzeugten Datensätzen in diesem Abschnitt wissen wir in einem gemessenen Datensatz nicht, welche Frequenzen auftreten. Daher können wir Fenster nicht als Vielfache der größten Periode wählen und gehen das Risiko von numerischen Artefakten ein. Um Sprünge beim Aneinanderhängen zu vermeiden, wird ein als Windowing bezeichnetes Verfahren verwendet: der zu analysierende Datensatz wird mit einem Filterkernel multipliziert, der bewirkt, dass die Amplitude an den Rändern des betrachteten Datenabschnitts gegen Null geht, so dass beim aneinander hängen keine Sprünge auftreten können. Press et al. [125] helfen beim Fensterln mit Erklärungen und Algorithmen für die gebräuchlichsten: Welch window, Hann window und Bartlett window.

§ 421 Und Press et al. [125] helfen auch beim Verständnis des Zero Paddings: neben der Spektralanalyse ist die Faltung zweier Funktionen eine wichtige Domäne der Fourier Transformation. Numerisch macht diese Faltung jedoch nur dann Sinn, wenn die Datensätze beider Funktionen die gleiche Basis haben und gleich lang sind. Ungleiche Länge umschiffet man dadurch, dass der kürzere Datensatz mit Nullen aufgefüllt wird, um ihn an den längeren anzugleichen.

**Verständnisfrage 39** Die meisten von Ihnen werden die Fourier Transformation zur Bestimmung des Leistungsdichtespektrums schon verwendet haben, z.B. im Rahmen des Elektronikpraktikums. Dort haben Sie es in der Regel mit langen Datensätzen zu tun gehabt und sich nicht um windowing gekümmert. Haben Sie damals die leakage nur übersehen oder war die unbedeutend?

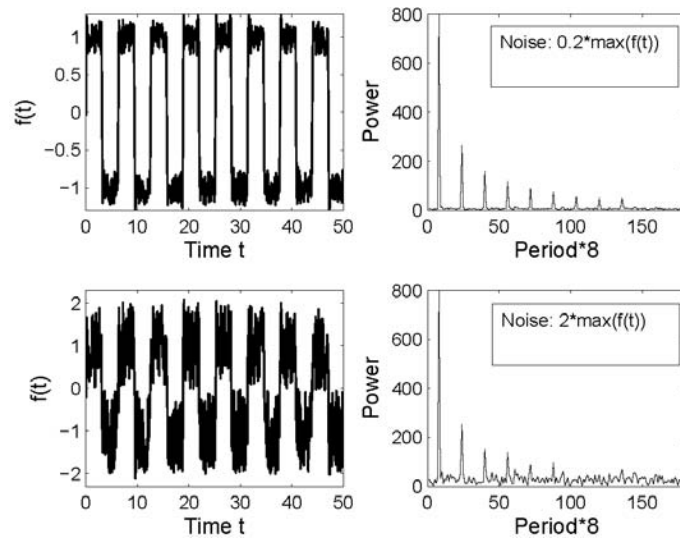
### Verrauschter Fourier

§ 422 In einem realen Datensatz ist dem Nutzsignal in der Regel ein Rauschen überlagert. Die Auswirkungen dieses Rauschens auf die FFT hängen vom Signal-zu-Rausch-Verhältnis ab. Eine entsprechende Situation ist durch Überlagerung von Zufallswerten über das Nutzsignal in Abb. 3.18 gezeigt. In den Beispielen wird deutlich, dass das Rauschen die leakage verstärkt: Power diffundiert von den im Nutzsignal vorhandenen Frequenzen in benachbarte Frequenzen.

**Verständnisfrage 40** Welches Fourier Spektrum hat Rauschen bzw. eine Reihe von Zufallszahlen? Begründen Sie!

**Zwischenrechnung 27** Spielen Sie mit einigen weiteren verrauschten Signalen und drehen Sie das Rauschen kräftig auf.

Abbildung 3.18: Verrauschtes Rechteck und FFT; wie Abb. 3.14, jedoch wurde dem über die Fourier-Reihe konstruierten Rechteck mit Hilfe des Zufallsgenerators Rauschen überlagert; *Obere Reihe*: die Amplitude des Rauschens erreicht maximal 20% der Amplitude des Nutzsignals; *Untere Reihe*: die Amplitude des Rauschens entspricht der des Nutzsignals



### Fourier und Faltung

§ 423 Die Faltungsregel können wir nun auch praktisch ausprobieren. Dazu erzeugen wir ein periodisches Signal (z.B. durch Überlagerung verschiedener Sinus- und Kosinusfunktionen), transformieren dieses mit Hilfe der FFT in den Frequenzbereich, multiplizieren die transformierte Funktion einmal mit der Übertragungsfunktion eines Tiefpass und einmal mit der eines Hochpass. Diese Multiplikationen liefern die gefilterte Funktion im Frequenzbereich. Rücktransformation mit IFFT liefert die gefilterter Funktion im Zeitbereich.

§ 424 Abbildung 3.19 zeigt das Ergebnis einer derartigen Prozedur für das aus Abb. 3.16 recycelte Rechteck für einen Tiefpass (oben) sowie einen Hochpass (unten) mit verschiedenen Grenzfrequenzen. Im linken Teil ist die schwarze Kurve jeweils das Ausgangssignal, das schwarze Leistungsdichtespektrum im rechten Teil gehört dazu. Die mit den jeweiligen Übertragungsfunktionen multiplizierten Frequenzspektren sind jeweils farbig in den rechten Teilbildern dargestellt; die daraus rekonstruierten Zeitverläufe sind in gleicher Farbe in den linken Teilbildern dargestellt.

§ 425 Beim Tiefpass verhindert das Abschneiden der hohen Frequenzen die Reproduktion der steilen Flanke des Rechtecks: bei der roten Kurve (Eingangsfrequenz kleiner als Grenzfrequenz) führt dies zu einer leichten Verzerrung, ist die Grenzfrequenz in der Größe der Eingangsfrequenz, so beobachtet man den Kondensator beim Auf- und Entladen (grüne Kurve). Für Frequenzen sehr viel größer als die Grenzfrequenz (rote Kurve) wirkt der Tiefpass als Integrator.

§ 426 Beim Hochpass wird das Leistungsdichtespektrum für hohe Frequenzen nicht modifiziert, stattdessen werden die tiefen Frequenzen unterdrückt. Für Frequenzen deutlich unter der Grenzfrequenz wirkt der Hochpass als Differentiator.

**Zwischenrechnung 28** Erzeugen Sie verschiedenen Signale (z.B. Sägezahn, Rechteck, pulsierende Gleichspannung ...), bilden deren Fourier Transformation und manipulieren dann das Spektrum. Untersuchen Sie den Einfluss dieser Manipulation auf das rekonstruierte Signal.

§ 427 MATLAB führt bei einer Faltung von zwei Funktionen diese ebenfalls im Frequenzbereich durch. Die Funktionen `conv` bzw. `conv2` für zweidimensionale Daten wie Bilder erzeugen erst die Fourier Transformierten beider Funktionen, multiplizieren diese und transformieren anschließend in den Zeitbereich zurück.

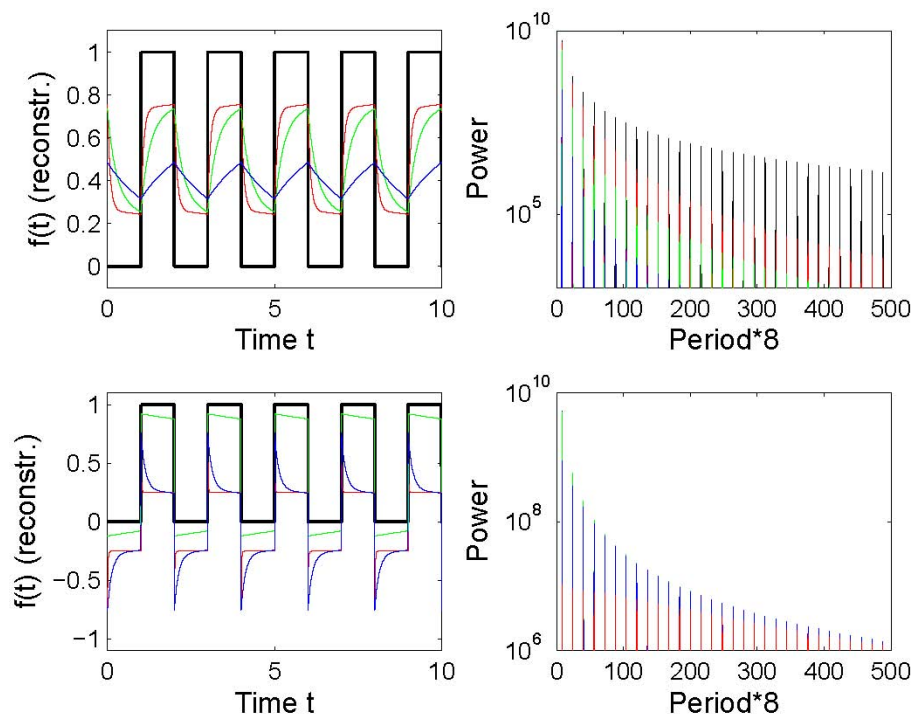


Abbildung 3.19: Multiplikation im Frequenzbereich statt Faltungsintegral im Zeitbereich, zur Erläuterung siehe Text

§ 428 Wenn Sie sich den Algorithmus in der MATLAB-Hilfe ansehen, so fällt auf, dass die Befehlssequenz `conv(x,y) = ifft(X.*Y)` nur dann korrekte Resultate liefert, wenn die Fourier Transformaten  $X$  und  $Y$  erzeugt werden als

```
X = fft([x zeros(1,length(y)-1)])
Y = fft([y zeros(1,length(x)-1)])
```

Beide Eingavektoren werden also mit Nullen aufgefüllt. Dieses Zero Padding ist erforderlich, da die Fourier Transformation mittel `fft` das wrap-around der Fourier Koeffizienten bewirkt. Außerdem müssen die Längen der Signale angeglichen werden.

### 3.2.5 Fourier in SIMULINK

§ 429 SIMULINK bietet zwar keinen Block für die FFT, dafür aber relativ einfache Möglichkeiten der Bestimmung eines Leistungsdichtespektrums oder der Spektralzerlegung. Diese Blöcke haben keinen Ausgang, d.h. die Ergebnisse können nicht in den Workspace ausgelesen werden. Entsprechend sind diese Blöcke auch als Senken einsortiert im SIMULINK Library Browser unter `Simulink Extras.Additional Sinks`.

§ 430 Abbildung 3.20 gibt ein einfaches Beispiel. Im linken Teil wird das Signal aus der Überlagerung zweier harmonischer Schwingungen und einem weißen Rauschen erzeugt. Im rechten Teil kann das sich ergebende Signal einmal als Zeitverlauf auf dem Oszilloskop dargestellt werden, zum anderen mit Hilfe des Blocks `Power Spectral Density` analysiert werden. In letzterem Block lassen sich als Parameter einstellen: die Länge des Buffers, die Zahl der Datenpunkte für die FFT, die Zahl der Punkte zwischen zwei konsekutiven Darstellungen sowie die Sampling Time. Ein mit dem Modell in Abb. 3.20 erzeugtes Leistungsdichtespektrum ist in Abb. 3.21 gezeigt; die dem Block zu übergebenden Parameter entsprechen denen bei der FFT..

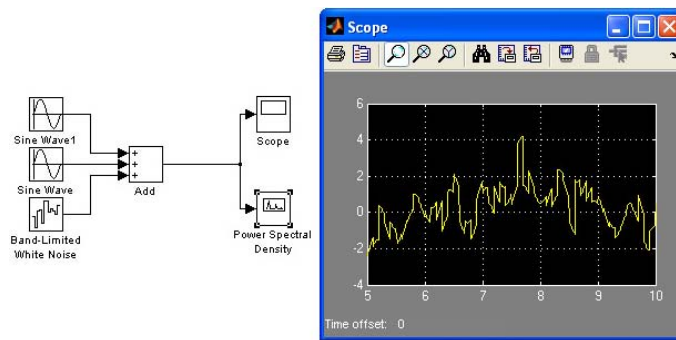


Abbildung 3.20: Leistungsdichtespektrum

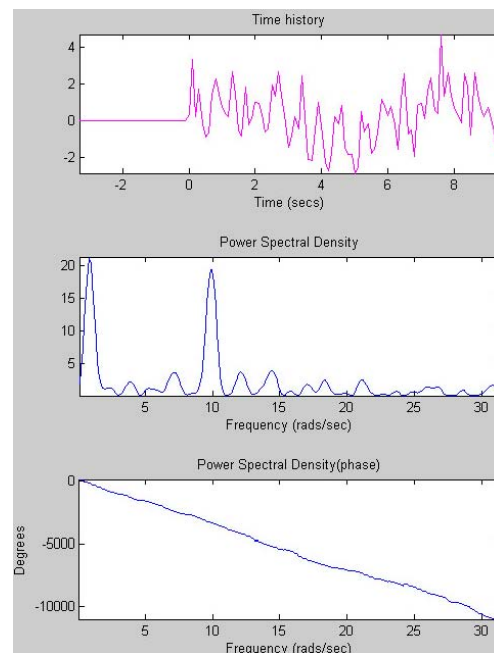


Abbildung 3.21: Beispiel für die Ausgabe eines Leistungsdichtespektrums

§ 431 Im gleichen Ordner des Library Browsers befinden sich auch Blöcke für die Auto- und Kreuzkorrelation – beide sind jedoch nur im Zusammenhang mit der Signal Processing Toolbox zu verwenden.

**Zwischenrechnung 29** Untersuchen Sie das Frequenzspektrum der beiden gekoppelten Pendel aus Abschn. 2.3.5.

**Zwischenrechnung 30** Reproduzieren Sie die Beispiele aus Abschn. 3.2.4 unter Verwendung des Blocks Leistungsdichtespektrum in SIMULINK.

### 3.3 Laplace Transformation

§ 432 Während die Fourier Transformation mathematisch für die Behandlung einer Impulsfunktion als Testfunktion im Zeitbereich geeignet ist, entsteht bei Verwendung einer Sprungfunktion ein Problem: der Ausdruck ist nicht integrierbar, d.h. wir erhalten keine Sprungantwort. Um diese Integrierbarkeit zu erzeugen, wird die Laplace Transformation verwendet, die zusätzlich einen Konvergenz erzeugenden Faktor enthält.

**Verständnisfrage 41** Warum ist der Ausdruck nicht integrierbar? Und warum löst ein Konvergenz erzeugender Faktor das Problem?

§ **433** Statt der Transformationsgleichungen der Fourier Transformation

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega t} d\omega \quad \text{bzw.} \quad F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$$

wird mit dem zusätzlichen Faktor  $e^{-\sigma t}$  aus  $F(\omega)$  ein  $F(\omega, \sigma)$  mit

$$F(\omega, \sigma) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} e^{-\sigma t} dt .$$

Der Ausdruck ist absolut integrierbar, so lange  $f(t)$  nicht stärker ansteigt als die Exponentialfunktion mit dem Exponenten  $\sigma$ . Der niedrigste Wert von  $\sigma$ , für den diese Bedingung erfüllt ist, wird als Konvergenzabszisse bezeichnet. Die beiden Exponentialfunktionen werden zusammen gefasst, so dass mit einer komplexen Frequenz  $s = \sigma + i\omega$  gearbeitet werden kann:

$$F(s) = \int_0^{\infty} f(t) e^{-st} dt .$$

Die Integration beginnt bei  $t = 0$  als dem Zeitpunkt, zu dem das Signal beginnt, d.h. es ist  $f(t) = 0$  für  $t < 0$ .

§ **434** Die Laplace Transformierte  $F(s)$  einer Funktion  $f(t)$  ist also definiert als

$$F(s) = \mathcal{L}\{f(t)\} = \int_0^{\infty} f(t) e^{-st} dt , \quad (3.14)$$

wobei  $s$  eine beliebige, auch komplexe Variable sein kann, z.B.  $i\omega$ . Einzige Einschränkung ist, dass das Integral in (3.14) konvergieren muss.

§ **435** Die Rücktransformation wird entsprechend geschrieben als  $\mathcal{L}^{-1}\{F(s)\} = f(t)$ :

$$f(t) = \mathcal{L}^{-1}\{F(s)\} = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(s) e^{st} ds .$$

Die Funktion im  $t$ -Bereich wird, wie bereits von der Fourier Transformation bekannt, als Originalfunktion bezeichnet, die Funktion im  $s$ -Bereich als Bildfunktion.

§ **436** Die Laplace Transformation erlaubt, genauso wie die Fourier-Transformation, eine formal einfache Lösung linearer Differentialgleichungen mit konstanten Koeffizienten: statt der direkten Lösung der Differentialgleichung im Zeitbereich  $t$  wird eine algebraische Gleichung im Bildraum  $s$  gelöst. Das Lösungsverfahren besteht aus drei Schritten:

1. die Differentialgleichung wird mit Hilfe einer Laplace Transformation in eine algebraische Gleichung im Bildraum überführt;
2. die algebraische Gleichung wird im Bildraum gelöst, es ergibt sich die Bildfunktion der gesuchten Lösung;
3. die inverse Laplace Transformation der Bildfunktion ergibt die Lösung im Zeitbereich.

§ **437** Die Laplace Transformation hat praktisch alle Eigenschaften der Fourier Transformation geerbt. Insbesondere ist sie eine lineare Operation:

$$\mathcal{L}\{\alpha f(t) + \beta g(t)\} = \alpha \mathcal{L}\{f(t)\} + \beta \mathcal{L}\{g(t)\} . \quad (3.15)$$

Auch die Faltungsregel bleibt bestehen, d.h. die Faltung zweier Funktionen kann überführt werden in das Produkt ihrer Transformaten:

$$\mathcal{L}\left\{ \int_0^t f(t-\tau)h(\tau)d\tau \right\} = f(s)h(s) . \quad (3.16)$$

Tabelle 3.2: Laplace Transformierte wichtiger Funktionen mit  $C = 0.577216$  als der Euler'schen Konstante

$F(s)$	$f(t)$
0	0
1	$\delta$ -Funktion
$1/s$	1 (Sprungfunktion)
$1/s^n$	$t^{n-1}/(n-1)!$
$1/(s-\alpha)^n$	$e^{\alpha t} t^{n-1}/(n-1)!$
$1/[(s-\alpha)(s-\beta)]$	$[e^{\beta t} - e^{\alpha t}]/[\beta - \alpha]$
$s/[(s-\alpha)(s-\beta)]$	$[\beta e^{\beta t} - \alpha e^{-\alpha t}]/[\beta - \alpha]$
$\alpha/(s^2 + \alpha^2)$	$\sin(\alpha t)$
$[\alpha \cos \beta + s \sin \beta]/[s^2 + \alpha^2]$	$\sin(\alpha t + \beta)$
$s/(s^2 + \alpha^2)$	$\cos(\alpha t)$
$[s \cos \beta - \alpha \sin \beta]/[s^2 + \alpha^2]$	$\cos(\alpha t + \beta)$
$\alpha/(s^2 - \alpha^2)$	$\sinh(\alpha t)$
$s/(s^2 - \alpha^2)$	$\cosh(\alpha t)$
$[s^2 + 2\alpha^2]/[s(s^2 + 4\alpha^2)]$	$\cos^2(\alpha t)$
$2\alpha^2/[s(s^2 + 4\alpha^2)]$	$\sin^2(\alpha t)$
$[s^2 - 2\alpha^2]/[s(s^2 - 4\alpha^2)]$	$\cosh^2(\alpha t)$
$2\alpha^2/[s(s^2 - 4\alpha^2)]$	$\sinh^2(\alpha t)$
$\alpha s/(s^2 + \alpha^2)^2$	$\frac{1}{2}t \sin(\alpha t)$
$\alpha s/(s^2 - \alpha^2)^2$	$\frac{1}{2}t \sinh(\alpha t)$
$1/\sqrt{s}$	$1/\sqrt{\pi t}$
$1/(s\sqrt{s})$	$2\sqrt{t}/\pi$
$1/\sqrt{s + \alpha}$	$e^{-\alpha t}/\sqrt{\pi t}$
$\sqrt{\sqrt{s^2 + \alpha^2} - s}$	$\sin(\alpha t)/[t\sqrt{2\pi t}]$
$\arctan(\alpha/s)$	$\sin(\alpha t)/t$
$\ln s/s$	$-C - \ln t$

Mit  $f(t=0) = f_0$  gilt die Differentiationsregel

$$\mathcal{L}\left\{\frac{df(t)}{dt}\right\} = sf(s) - f_0, \quad (3.17)$$

auch diese Regel ist bereits von der Fourier Transformation bekannt – lediglich die imaginäre Einheit als Vorfaktor verschwindet, da  $s$  komplex ist.<sup>11</sup>

§ 438 Bei einer vektorwertigen Funktion wird die Laplace Transformation komponentenweise durchgeführt – das gilt auch für die Fourier Transformation, wie eigentlich bereits aus der spaltenweisen Anwendung von FFT auf Matrizen in MATLAB zu erahnen war.

§ 439 Laplace Transformierte für einige wichtige Funktionen sind in Tab. 3.2 gegeben, umfangreichere Tabellen finden Sie z.B. in [9] oder [154].

§ 440 Die Laplace-Transformation ist, bis auf den Konvergenz erzeugenden Faktor, das Äquivalent zur kontinuierlichen Fourier Transformation. Eine diskrete Laplace Transformation als Äquivalent zur DFT gibt es nicht – entsprechend gibt es auch in MATLAB keine Funktionen, die irgendetwas mit Laplace Transformationen zu tun haben. In SIMULINK dagegen tritt zumindest die aus der Laplace Transformation bekannte Notation für die Integration und die Übertragungsfunktion auf; allerdings sollte man sich davon nicht aufs Glatteis führen lassen: SIMULINK verwendet zwar die Notation, rechnet als numerisches System aber natürlich nicht mit den Laplace-Transformierten sondern verwendet konventionelle Algorithmen zur Integration.

<sup>11</sup>Und die Bezeichnung  $\frac{1}{s}$  auf dem Integratorblock wird unmittelbar einsichtig: wenn die Differentiation einer Multiplikation mit der komplexen Frequenz  $s$  im Bildraum entspricht, sollte die Integration als ihre Umkehrung der Division durch die komplexe Frequenz entsprechen.

### 3.3.1 Laplace Transformation und Lösung der DGLs

§ 441 Auf Grund dieser Einschränkungen ist die Anwendung der Laplace Transformation auf analytische Verfahren beschränkt; insbesondere in der Systemtheorie kommt man an der Laplace Transformation kaum vorbei – nur auf diese Weise lässt sich eine allgemeine Übertragungsfunktion angeben.

§ 442 Ansonsten lässt sich die Laplace Transformation ebenso wie die kontinuierliche Fourier Transformation bei der Lösung von Differentialgleichung(ssystem)en einsetzen. Wie bei der kontinuierlichen Fourier Transformation wird die DGL in eine algebraische Gleichung überführt, die Lösung eines Differentialgleichungssystems wird damit auf die Lösung eines Systems algebraischer Gleichungen reduziert. Sind die DGLs erster Ordnung, so ist das algebraische Gleichungssystem linear; handelt es sich um DGLs höherer Ordnung, ist das System nicht mehr linear. Gegenüber der Fourier Transformation hat die Laplace Transformation bei der Lösung von Differentialgleichungen den Vorteil, dass die  $\delta$ -Funktion nicht so häufig auftritt – die Funktionen im Ausgangs- und im Bildbereich sind in der Regel ‘well-behaved’.

**Zwischenrechnung 31** Wenn die Laplace Transformation abgesehen von der anderen Transformationstabelle völlig äquivalent zur kontinuierlichen Fourier Transformation verwendet werden kann, so muss auch das Beispiel aus § 396 (freie Schwingung) mit Hilfe einer Laplace Transformation zu lösen sein. Überprüfen Sie dies.

#### Gewöhnliche DGL 1. Ordnung: Auf- und Entladen eines Kondensators

§ 443 Betrachten wir als einfaches Beispiel eine gewöhnliche Differentialgleichung erster Ordnung. Das Entladen eines Kondensators (oder alternativ der radioaktive Zerfall) wird beschrieben durch eine homogene DGL vom Zerfallstyp:

$$\dot{q} = -\frac{1}{RC}q$$

mit  $q(t)$  als der auf dem Kondensator befindlichen Ladung,  $C$  der Kapazität und  $R$  dem Widerstand. Laplace Transformation der Gleichung liefert eine lineare algebraische Gleichung für  $Q(s)$ , d.h. die Funktion im Bildraum:

$$sQ(s) - Q_0 = -\frac{1}{RC}Q(s).$$

Die Anfangsbedingung ist durch  $Q_0(s)$  berücksichtigt. Die Lösung der Bildgleichung ist

$$Q(s) = \frac{1}{s + \frac{1}{RC}}Q_0.$$

Rücktransformation in den Zeitbereich liefert gemäß Tabelle 3.2

$$q(t) = q_0 e^{-t/(RC)} = q_0 e^{-t/\tau} \quad \text{mit} \quad \tau = RC.$$

§ 444 Und die Umkehrung? Laden wir den Kondensator auf. Dazu liegt eine (konstante) äußere Spannung  $U_0$  an, die homogene DGL wird zu einer inhomogenen:

$$\dot{q} = -\frac{q}{RC} + \frac{U_0}{R}.$$

Bei der Laplace Transformation tritt entsprechend ein zusätzlicher Term auf, nämlich der, der die Transformation der Konstanten  $U_0/R$  beschreibt:

$$sQ(s) - Q_0 = -\frac{1}{RC}Q(s) + \frac{1}{s} \frac{U_0}{R}.$$

Die Lösung der Bildgleichung ist

$$Q(s) = \frac{1}{s + \frac{1}{RC}}Q_0 + \frac{1}{s(s + \frac{1}{RC})} \frac{U_0}{R}.$$

Rücktransformation liefert die Lösung im Zeitbereich:

$$q(t) = Q_{\max}(1 - e^{c-t/\tau})$$

mit  $CU_0 = Q_{\max}$ ,  $\tau = RC$  und  $c =$ . Verschwindet die Ladung zur Zeit  $t = 0$ , d.h.  $q_0 = 0$ , so reduziert sich die Lösung auf

$$q(t) = Q_{\max}(1 - e^{-t/\tau})$$

**Zwischenrechnung 32** Falls Ihnen der Kondensator nicht bereits zum Hals raushängt – legen Sie statt der konstanten Spannung doch einfach mal eine Wechselspannung an und betrachten, was dann passiert (Hinweis: wenn Sie das tun, haben Sie den Tiefpass gerechnet).

§ 445 Gewöhnliche Differentialgleichungen höherer Ordnung können entsprechend behandelt werden – liefern dann allerdings algebraische Gleichungen höherer Ordnung. Das wurde bereits bei der Lösung der Schwingungsgleichung mit Hilfe der Fourier Transformation deutlich. Ein Blick auf Tabellen zur Laplace Transformation, wie z.B. Tab. 3.2, legt nahe, dass das bei DGLs  $n$ ter Ordnung entstehende Polynom  $n$ ter Ordnung in  $s$  vor der Rücktransformation faktorisiert werden muss.

### Partielle Differentialgleichungen

§ 446 Analytische Lösungen für partielle Differentialgleichungen basieren, wie bei der Wellengleichung in Abschn. 3.1.1 beschrieben, auf einem Separationsansatz: die partielle Differentialgleichung wird durch Separation der räumlichen und zeitlichen Abhängigkeit in mehrere gewöhnliche Differentialgleichungen überführt. Das Verfahren zur Lösung einer partiellen Differentialgleichung mit Hilfe einer Laplace Transformation ist ähnlich:<sup>12</sup>

1. die partielle Differentialgleichung mit ihren Anfangs- und Randbedingungen wird mit Hilfe der Laplace Transformation in eine gewöhnliche Differentialgleichung im Bildbereich überführt. In dieser gewöhnlichen DGL sind die Anfangsbedingungen bereits automatisch berücksichtigt, die Randbedingungen gehen bei der Transformation in Randbedingungen über.
2. die gewöhnliche Differentialgleichung im Bildbereich wird gelöst.
3. die Lösung wird aus dem Bildbereich in den Originalraum zurück transformiert – dieser Schritt ist häufig der Schwierigste.

Und wenn wir gemein sind, lösen wir die gewöhnliche DGL durch – klar, eine Laplace Transformation.

§ 447 Als Beispiel für eine partielle DGL soll die eindimensionale Diffusions- bzw. Wärmeleitungsgleichung dienen – wir werden ihr in Abschn. 7.1 ohnehin noch häufiger begegnen. Setzen wir die Diffusionskonstante gleich 1, so wird die Struktur der Diffusionsgleichung deutlich:

$$\frac{\partial^2 u}{\partial x^2} = \frac{\partial u}{\partial t}.$$

Diese Diffusionsgleichung ist eine Bestimmungsgleichung für die Funktion  $u(x, t)$ .

§ 448 Als Anfangsbedingung nehmen wir auf einem linearen Leiter ein (Temperatur)Profil  $u(x, 0) = U_0(x)$  an. Für die Randbedingungen nehmen wir an, dass die Enden  $x = 0$  und  $x = l$  des Leiters auf einer vorgegebenen (aber gegebenenfalls von der Zeit abhängigen) Temperatur/Dichte gehalten werden können:  $u(0, t) = a_0(t)$  und  $u(l, t) = a_1(t)$ .

<sup>12</sup>Dieses Herunterbrechen der PDGL in gewöhnliche DGLs erfolgt letztendlich auch bei numerischen Lösungen: dort wird bei der Diffusionsgleichung eine große Zeitschleife angelegt und in jedem Zeitschritt wird der räumliche Transport gelöst. Also frei nach einem bekannten Sohn der Stadt Osnabrück: 'Im Bildraum nichts Neues'.



§ 449 Als erster Schritt ist die Laplace Transformation durchzuführen. Für die Gleichung im Bildbereich erhalten wir

$$sU(x, s) - U_0(x) = \frac{\partial^2 U(x, s)}{\partial x^2}.$$

Dabei haben wir auf der rechten Seite vorausgesetzt, dass die Ableitung nach  $x$  mit der Laplace Transformation vertauschbar ist. Da nur noch nach einer Variablen  $x$  abgeleitet wird, können wir die partielle Ableitung durch eine gewöhnliche Ableitung ersetzen und  $s$  als einen Parameter betrachten. Die transformierten Randbedingungen schreiben wir als  $U(0, s) = A_0(s)$  und  $u(l, s) = A_1(s)$ .

**Verständnisfrage 42** Zeigen Sie mit Hilfe der Transformationsregeln, dass die Laplace Transformation und die Ableitung nach einer anderen Variablen vertauschbar sind.

§ 450 Betrachten wir den einfachsten Fall: die Anfangstemperatur verschwindet. Dann haben wir im Bildbereich eine gewöhnliche homogene DGL 2. Ordnung:

$$\frac{d^2 U}{dx^2} = sU.$$

Die Gleichung kennen wir; als Lösungsverfahren können wir den Exponentialansatz  $U = e^{\lambda x}$  verwenden und erhalten die beiden Eigenwerte  $\lambda_{1,2} = \sqrt{s}$  und damit die allgemeine Lösung  $c_1 e^{x\sqrt{s}} + c_2 e^{-x\sqrt{s}}$ . Die Integrationskonstanten  $c_1$  und  $c_2$  sind aus den Randbedingungen zu bestimmen. Von diesem Punkt an ist eine allgemeine Lösung nicht mehr sinnvoll, der weitere Verlauf der Lösung hängt stark von den Randbedingungen ab – und damit kann auch die Rücktransformation beliebig unhandlich werden.

§ 451 Lassen wir die Anfangstemperatur nicht verschwinden, so muss im Bildbereich eine inhomogene gewöhnliche DGL 2. Ordnung gelöst werden. Beide Varianten finden Sie ausführlicher diskutiert in [22].

## Literaturhinweise

§ 452 Die wichtigsten Aspekte zur Laplace Transformation inklusive der wichtigsten Transformationen finden Sie in den üblichen Handbüchern wie dem Bronstein [9] oder dem Stöcker [154]. Die praktischen Aspekte der Laplace Transformation werden in recht anschaulicher Weise im Doetsch [22] erklärt; etwas tabellarischer, dafür aber mit vielen (physikalischen) Beispielen, ist die Darstellung in Spiegel [151]. Die Fourier Transformation in ihren verschiedenen Geschmacksrichtungen und insbesondere ihre Anwendung in der Signalanalyse inklusive Windowing und Zero Padding ist im entsprechenden Kapitel in Meffert und Hochmuth [95] beschrieben, ebenso in McClellan et al. [94].

## Fragen

**Frage 3** Erläutern Sie Unterschiede und Zusammenhänge zwischen Fourier Reihe und Fourier Transformation.

**Frage 4** Erläutern Sie Unterschiede und Zusammenhänge zwischen kontinuierlicher Fourier und Laplace Transformation.

**Frage 5** Erläutern Sie den Begriff und die Bedeutung der Nyquist Frequenz.

**Frage 6** Wodurch werden die minimale und die maximale Frequenz bei der FFT bestimmt?

**Frage 7** Erläutern Sie die Bedeutung von windowing und zero padding in der FFT.

**Frage 8** Erläutern Sie die wesentlichen Eigenschaften der kontinuierlichen Fourier Transformation.

**Frage 9** Erläutern Sie die Verwendung der Laplace Transformation bei der Lösung von Differentialgleichungen.

## Aufgaben

**Aufgabe 4** Entwickeln Sie die Funktion

$$F(x) = \begin{cases} 2 - x & 0 < x < 4 \\ x - 6 & 4 < x < 8 \end{cases}$$

in eine Fourier Reihe mit Periode 8.

**Aufgabe 5** Lösen Sie die Differentialgleichungen aus Kapitel 1 (harmonischer Oszillator, gekoppelte Oszillatoren, vom Tisch gleitendes Seil) mit Hilfe einer Laplace Transformation. Für die Rücktransformation können Sie sich von den Ihnen bekannten Lösungen inspirieren lassen.

**Aufgabe 6** Lösen Sie die Differentialgleichung  $\ddot{x} - 3\dot{x} + 2x = 4e^{2t}$  mit den Anfangsbedingungen  $x(0) = -3$  und  $\dot{x}(0) = 5$  mit Hilfe einer Laplace Transformation.

**Aufgabe 7** Lösen Sie das Differentialgleichungssystem

$$\begin{aligned} \dot{x} &= 2x - 3y & \text{mit} & & x(0) &= 0 \\ \dot{y} &= y - 2x & & & y(0) &= 3 \end{aligned}$$

mit Hilfe einer Laplace Transformation.

**Aufgabe 8** Ein Balken liegt an den Punkten  $x = 0$  und  $x = l$  auf, er trägt eine gleichmäßige Last von  $W_0$  pro Längeneinheit. Die DGL für die Durchbiegung des Balkens ist

$$\frac{d^4 y}{dx^4} = \frac{X_0}{E} \quad 0 < x < l \quad y(0) = y''(0) = y(l) = y''(l) = 0 .$$

Bestimmen Sie die Durchbiegung  $y(x)$  des Balkens mit Hilfe einer Laplace Transformation.

**Aufgabe 9** Versuchen Sie sich an der Lösung der Wellengleichung mit Hilfe einer Laplace Transformation.

**Aufgabe 10** Chladni als 2D-Variante des Balkens?

# Kapitel 4

## Lineare zeitinvariante Systeme: Systemtheorie im Vorbeigehen

Every occurrence in nature is preceded by other occurrences which are its cause, and succeeded by others which are its effects. The human mind is not satisfied with observing and studying any natural occurrence alone, but takes pleasure in connecting every natural fact which has gone before it, and with what is to come after it.

J. Tyndall [166]

§ 453 Systemtheorie und die Modellierung dynamischer Systeme ist insbesondere auf Grund der vielfältigen Anwendungen ein für ein einfaches Skript über simple Simulation viel zu umfangreiches Feld. Dieses Kapitel beschränkt sich auf eine etwas formale, an die in der Schaltungs- und Regelungstechnik gebräuchliche Betrachtung von linearen zeitinvarianten Systemen um die grundlegenden Begriffe einzuführen und damit auch die Benennung und Funktion einiger der SIMULINK-Blöcke (insbesondere des State Space Blocks, siehe auch § 158) besser zu verstehen.

§ 454 Das Kapitel ist nur etwas mehr als Liste von Definitionen. Wir werden uns an Hand einfacher Beispiele und auch im Vorgriff auf Kapitel 5 auf den Hoch- und Tiefpass sowie Serienschwindkreis beschränken.

§ 455 Auch wenn man im Vergleich zu den in Abschn. 1.3 gegebenen Beispielen eine elektronische Schaltung nicht unbedingt als System betrachten würde, ist sie ein gutes Beispiel für Systeme und Untersysteme. In einem HiFi-Verstärker sind jede Menge an Filtern zur Vermeidung von Signalverzerrungen und zur Optimierung des Outputs vorhanden. Diese sind jeweils Kombinationen aus Widerständen, Spule und Kondensatoren. In einer EDA (Electronic Design Automation) werden die Bauteile einzeln betrachtet, in einem eher systemorientierten Ansatz dagegen können wir die entsprechenden Bauteile jeweils zur funktionalen Gruppe des Filters zusammen fassen und nicht mehr durch die Parameter der Bauteile sondern die Eigenschaften des Filters, d.h. seine Übertragungsfunktion, beschreiben. Diese ist nicht durch die Bauteilparameter sondern die charakteristische Größe des Systems bestimmt: so ist die Grenzfrequenz eines Tief- oder Hochpass durch die Zeitkonstante  $\tau$  bestimmt. Zwar geht in diese das Produkt aus Widerstand und Kapazität ein, allerdings lässt sich ein festes  $\tau$  (zumindest im Prinzip) mit unendlich vielen Kombinationen von  $R$  und  $C$  realisieren. Welche davon gewählt wurde, ist für den systemtheoretischen Ansatz irrelevant.

§ 456 Das Kapitel ist beschränkt auf lineare und zeitinvariante Systeme, häufig als LTI-Systeme (für Linear Time Invariant) bezeichnet. Ziel ist die Beschreibung der Reaktion eines

Systems auf eine definierte Testfunktion. Der dazu erforderliche Formalismus kann im Zeitbereich oder im Frequenzbereich betrachtet werden; beide Varianten können durch Fourier oder Laplace Transformationen verknüpft werden. Als Testfunktionen werden im Zeitbereich die  $\delta$ -Funktion, im Frequenzbereich periodische Funktionen verwendet. Während die  $\delta$ -Funktion eine Impulsantwort liefert, mit deren Hilfe die Antwort des Systems auf eine andere Testfunktionen durch Faltung bestimmt werden kann, liefert die periodische Funktion die Übertragungsfunktion des Systems.

**Lernziele:** nach Durcharbeiten dieses Kapitels sollten Sie in der Lage sein

- die Grundbegriffe der Systemtheorie, insbesondere Übertragungsfunktion und Rückkopplungen, zu kennen und korrekt anzuwenden.
- einfache Systeme unter systemtheoretischen Aspekten zu beschreiben.
- die formale Beschreibung eines Systems durch die Systemgleichungen zu kennen – SIMULINK hat einen genau auf diese Beschreibung zurück greifenden Block.

## 4.1 Grundbegriffe: Was ist ein System?

§ 457 In diesem Abschnitt über LTI-Systeme werden wir mit einfachen Beispielen aus der Elektronik arbeiten, d.h. im wesentlichen mit dem bereits in § 423ff verwendeten Hoch- und Tiefpass oder dem Bandpass. Eine elektronische Schaltung werden wir dabei als ein System betrachten, die Ströme oder Spannungen in diesem Netzwerk als Signale, auch bezeichnet als Systemgröße oder Systemvariable. Damit können wir die langatmige Definition eines Systems aus § 100 reduzieren auf die folgende Gebrauchsdefinition:

**Definition 1** *Ein System ist eine Zusammenfügung verschiedener, miteinander wechselwirkender Komponenten (z.B. elektronische Bauteile). Innerhalb des Systems sind Rückkopplungen möglich. Ein System kann in Untersysteme zerlegt werden.*

Entsprechend können wir die dieses System erst lebendig machenden Signale definieren als

**Definition 2** *Ein Signal ist eine messbare Größe innerhalb dieses Systems. Insbesondere sind auch die Ein- und Ausgangsgröße des Systems Signale.*

§ 458 Typische Signale in der Elektronik (aber auch in der Reizleitung im Nervensystem) sind Ströme und Spannungen. Dabei treten spezielle Zeitverläufe immer wieder auf:

- konstante Spannung  $u = U_0 = \text{const}$ ,
- Sprungfunktion  $u = H(t_0) U_0$  (in ihrer Bedeutung hinreichend bekannt von der Sprungantwort),
- abfallende Exponentialfunktion  $u = U_0 e^{-\alpha t}$  (alles im Zerfall begriffene, daher auch bei Verzögerungen/leckenden Speichern beliebt),
- Rampe  $u = kt$  (und sei es nur bei der Aufnahme einer Kennlinie),
- Rechteckimpuls der Dauer  $T$  als  $u = U_0 H(t) H(T-t)$ , d.h. die Kombination zweier Sprungfunktionen, oder
- harmonische bzw. sinusoidale Spannung  $u = U_0 \sin(\omega t + \varphi)$ .

Die  $\delta$ -Funktion mit ihrer Bedeutung für die Impulsantwort tritt in einem realen System nicht auf, da sie eine rein mathematische Funktion ist. Real lässt sich die Sprungantwort eines Systems nicht bestimmen; der Rechteckimpuls mit kürzest möglicher Dauer  $T$  ist die Annäherung an die  $\delta$ -Funktion.<sup>1</sup>

<sup>1</sup>Aus dieser Vorbemerkung können wir eine allgemeine Schlussfolgerung bezüglich der formalen Behandlung realer Systeme ziehen: die Fourier Transformation ist in einem realen System nur bedingt brauchbar, da sich der Impuls als Input eigentlich nicht realisieren lässt: selbst beim Abschlag eines Golfballs ist die Kraft keine  $\delta$ -Funktion sondern kann bestenfalls durch eine Dreieck- oder Trapezfunktion angenähert werden. Auf Grund dieser endlichen Signaldauern ist es daher sinnvoller, in Sprungfunktionen bzw. deren Kombination zu denken. Daher ist in der formalen Beschreibung die Laplace Transformation auf Grund des Konvergenz erzeugenden Faktors besser geeignet. Konzeptionell ist die Impulsantwort als Green'sche Funktion aber für

§ 459 Die Methodik der Systemtheorie besteht in der Abstraktion vom realen physikalischen System mit allen seinen Details auf ein allgemeineres mathematisches Modell. In der Elektronik bedeutet dies z.B. den Übergang von den konkreten Eigenschaften der Bauelemente wie Widerständen oder Kapazitäten auf sich aus daraus ergebende funktionale Eigenschaften der Schaltung wie z.B. Zeitkonstanten. *Das mathematische Modell ist daher, da es sich von den Details gelöst hat, auf unterschiedliche physikalische Realisierungen eines Systems anwendbar.* Daher lässt ein Neuron sich ja auch praktisch in diese Struktur pressen, obwohl seine physikalische Realisierung sich sicherlich deutlich von der mit konventionellen elektronischen Bauteilen unterscheidet.

§ 460 Die Grundaufgabe in der Systemtheorie besteht also darin (a) die wesentlichen Prozesse eines Systems zu identifizieren und (b) sie adäquat formal zu beschreiben.

### 4.1.1 Ein- und Ausgangsgrößen, Zustandsvariable

§ 461 Größere Systeme werden häufig in Teilsysteme zerlegt: zum einen sind die Teilsysteme überschaubarer und damit einfacher formal zu erfassen. Zum anderen entsteht auf diese Weise ein Baukasten von Systemmodulen, von denen einzelne auch als Teilsysteme in anderen Systemen verwendet werden können. Dadurch wird es möglich, z.B. in SIMULINK nahezu beliebige Systeme mit einem kleinen Satz an Systemmodulen zu realisieren.

§ 462 In diesen (Teil-)Systemen werden die Systemgrößen erfasst und beschrieben. Die Systemgrößen können in der Regel in zwei Kategorien unterteilt werden: Größen innerhalb des Systems (z.B. Strom durch den Widerstand in einem Tiefpass) und die Ein- und Ausgangsgrößen, die die Verknüpfung des Systems mit der Außenwelt bilden – wobei letztere Systemgrößen sein können (der Spannungsabfall über dem Kondensator beim Tiefpass) aber nicht sein müssen.

**Definition 3** Eine Systemgröße wird als Zustandsvariable oder kurz Zustand bezeichnet, wenn sie durch die Angabe eines Anfangswerts  $x_0 = x(t_0)$  und die Angabe einer Eingangsgröße  $u(t)$  das Verhalten des Systems für alle  $t > t_0$  festlegt.

**Verständnisfrage 43** Für ein LTI-System ist das eine einfach einzusehende Bedingung – hier schlägt der Laplace'sche Dämon mit aller Macht zu. Wie sieht es in einem nicht-linearen System aus? Lässt sich die Zustandsvariable dort auf die gleiche Weise definieren? Hinweis: Sie können das Fadenpendel und das Klima- und Wirtschaftssystem als Extremfälle für ein einfaches und ein komplexes System verwenden.

**Verständnisfrage 44** Zustandsgrößen bzw. -variablen sollten Ihnen aus der Thermodynamik bekannt sein. Ist deren Definition mit der obigen Definition für ein LTI-System konsistent oder nicht? Oder anders gefragt: ist ein ideales Gas ein LTI System? Und wie sieht es mit einem realen Gas aus?

§ 463 Wird ein System durch  $n$  Zustandsvariable  $x_i$  beschrieben, so lassen sich diese in einem  $n$ -dimensionalen Zustandsvektor zusammen fassen:<sup>2</sup>

$$\vec{x}(t) = (x_1(t), x_2(t), \dots, x_n(t)) .$$

§ 464 Ein Beispiel für ein einfaches System ist ein Tiefpass, vgl. linken Teil in Abb. 4.1. Die von der Spannungsquelle gelieferte Spannung  $u(t)$  ist die Eingangsgröße, die über dem Kondensator abfallende Spannung  $u_C(t)$  die Ausgangsgröße. Für die Entwicklung eines mathematischen Modells dieses Netzes gehen wir von idealen Bauteilen aus, bei denen die Beziehungen zwischen dem Strom  $i$  und der Spannung  $u$  jeweils durch die Bauteilparameter

---

unsere Vorstellung von Punktladung und Punktmasse wichtig; das Poisson Integral ist die Faltung der Impulsantwort (elektrisches Feld einer Punktladung) mit der Ladungsverteilung. Und selbst für soziale Systeme ist der unit act (die kleinste Handlungseinheit, der Handlungsimpuls) von Parsons [115] eingeführt worden.

<sup>2</sup>Da wir an keiner Stelle ein dyadisches Produkt zweier Vektoren benötigen, werde ich mit der Darstellung der Vektoren etwas nachlässig sein und nicht zwischen Zeilen- und Spaltenvektoren unterscheiden: drehen Sie sich den Vektor bei Bedarf einfach in die von den Regeln der Vektormultiplikation geforderte Lage.

Kapazität  $C$  bzw. Widerstand  $R$  gegeben sind:  $i(t) = C \dot{u}_C(t)$  und  $u_R(t) = i(t)R$ . Aus der Maschenregel erhalten wir als die das System beschreibende Differentialgleichung

$$u(t) = u_R(t) + u_C(t) \quad \Rightarrow \quad RC \dot{u}_C(t) = u(t) - u_C(t). \quad (4.1)$$

Für bekanntes  $u(t)$  und vorgegebenen Anfangswert  $u_C(t_0)$  ist die Lösung  $u_C(t)$  der DGL eindeutig festgelegt, d.h. die Spannung  $u_C(t)$  ist eine Zustandsvariable des Systems.

### 4.1.2 LTI-System

§ 465 Für die formale Betrachtung wollen wir uns auf Systeme beschränken, die linear sind und die sich nur langsam verändern im Vergleich zu den betrachteten zeitlichen Änderungen der Systemgrößen (linear time invariant LTI). Daher werden die Modelle als zeitinvariant bezeichnet.

**Definition 4** Linearität eines Systems bedeutet, (a) die Gültigkeit des Superpositionsprinzips und (b) eine Kausalität in dem Sinne, dass zwischen Ursache und Wirkung eine Proportionalität besteht.

**Querverbindung 4** Langsame Änderungen werden ebenfalls noch als zeitinvariant betrachtet. Was bedeutet langsam in diesem Zusammenhang? Nennen Sie Beispiele!

§ 466 Betrachten wir dazu eine lineare und konstante Zuordnungsvorschrift zwischen der Ausgangsgröße  $y(t)$ , den Zustandsvariablen  $\vec{x}$  und der Eingangsgröße  $u(t)$ :

$$y(t) = \vec{c} \cdot \vec{x}(t) + du(t) = \sum_{i=1}^n c_i x_i(t) + du(t)$$

mit  $\vec{c} = \text{const}$  und  $d = \text{const}$ . Bei einem derartigen Bildungsgesetz hängen die Werte des Zustandsvektors  $\vec{x}$  zur Zeit  $t$  nur von seinem Anfangswert  $\vec{x}_0$  und dem zeitlichen Verlauf der Eingangsgröße  $u(t)$  bis zum Zeitpunkt  $t$  ab. Daher ist das System ein kausales System, symbolisch geschrieben als

$$\vec{x}(t) = \Gamma \begin{pmatrix} \vec{x}_0 \\ u(\tau) \end{pmatrix} \quad \text{mit} \quad t_0 \leq \tau \leq t$$

bzw. in verkürzter Schreibweise

$$\vec{x} = \Gamma \begin{pmatrix} \vec{x}_0 \\ u \end{pmatrix}.$$

Der obere Teil des Vektors enthält die Abhängigkeit von der Eingangsgröße und damit den homogenen Teil der Systemgleichungen, die untere Komponente gibt die Eingangsgröße und damit die Inhomogenität.

§ 467 In einem linearen System gilt das Superpositionsprinzip. Es erlaubt die Zerlegung des Lösungsverfahrens in zwei Teile: die Überlagerung des Beitrags auf Grund des Anfangswerts  $\vec{x}_0$  bei verschwindendem  $u$  (homogener Anteil) und des Beitrags der Eingangsgröße  $u$  bei verschwindendem  $\vec{x}_0$  (inhomogener Anteil bei trivialem Anfangszustand). Dieser Aspekt des Superpositionsprinzips ist für die Lösung des das System beschreibenden DGL-Systems von Bedeutung – wir haben es in einem 1D System bereits bei der erzwungenen Schwingung in der Einleitung verwendet.

§ 468 Das Superpositionsprinzip erlaubt aber auch die Überlagerung der Reaktionen auf unterschiedliche Eingangssignale: liefern die Eingangsgrößen  $u_1$  und  $u_2$  jeweils die Zustände  $\vec{x}_1$  und  $\vec{x}_2$ , oder symbolisch  $u_1 \rightarrow \vec{x}_1$  und  $u_2 \rightarrow \vec{x}_2$ , so liefert die kombinierte Eingangsgröße  $u = u_1 + u_2$  den Zustand  $\vec{x} = \vec{x}_1 + \vec{x}_2$ , oder symbolisch:  $(u_1 + u_2) \rightarrow \vec{x}_1 + \vec{x}_2$ .

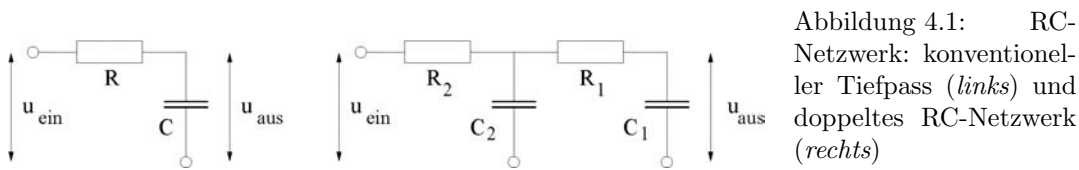


Abbildung 4.1: RC-Netzwerk: konventioneller Tiefpass (*links*) und doppeltes RC-Netzwerk (*rechts*)

§ 469 Mit der verkürzten Schreibweise erhalten wir als Formulierung für die Linearität des Systems bezüglich der Eingangsgrößen

$$\vec{x}_1 = \Gamma \begin{pmatrix} 0 \\ u_1 \end{pmatrix} \wedge \vec{x}_2 = \Gamma \begin{pmatrix} 0 \\ u_2 \end{pmatrix} \Rightarrow \Gamma \begin{pmatrix} 0 \\ u_1 + u_2 \end{pmatrix} = \Gamma \begin{pmatrix} 0 \\ u_1 \end{pmatrix} + \Gamma \begin{pmatrix} 0 \\ u_2 \end{pmatrix}$$

bzw. bei verschwindender Eingangsgröße für die Linearität des Systems bezüglich der Anfangszustände

$$\vec{x}_1 = \Gamma \begin{pmatrix} \vec{x}_{01} \\ 0 \end{pmatrix} \wedge \vec{x}_2 = \Gamma \begin{pmatrix} \vec{x}_{02} \\ 0 \end{pmatrix} \Rightarrow \Gamma \begin{pmatrix} \vec{x}_{01} + \vec{x}_{02} \\ 0 \end{pmatrix} = \Gamma \begin{pmatrix} \vec{x}_{01} \\ 0 \end{pmatrix} + \Gamma \begin{pmatrix} \vec{x}_{02} \\ 0 \end{pmatrix}.$$

Entsprechend lässt sich die Proportionalität zwischen Ursache und Wirkung formulieren als

$$\vec{x} = \Gamma \begin{pmatrix} 0 \\ u \end{pmatrix} \Rightarrow \alpha \vec{x} = \Gamma \begin{pmatrix} 0 \\ \alpha u \end{pmatrix} \quad \text{und} \quad \vec{x} = \Gamma \begin{pmatrix} \vec{x}_0 \\ 0 \end{pmatrix} \Rightarrow \alpha \vec{x} = \Gamma \begin{pmatrix} \alpha \vec{x}_0 \\ 0 \end{pmatrix}.$$

Insgesamt lässt sich die Linearität in dieser verkürzten Schreibweise formulieren als

**Definition 5** *Das System*

$$\vec{x} = \Gamma \begin{pmatrix} \vec{x}_0 \\ 0 \end{pmatrix}$$

heißt linear, wenn für beliebige Konstanten  $\alpha$ ,  $\beta$ ,  $\vec{x}_{01}$  und  $\vec{x}_{02}$  sowie beliebige Funktionen  $u_1$  und  $u_2$  das Superpositionsprinzip gilt

$$\Gamma \left[ \alpha \begin{pmatrix} \vec{x}_{01} \\ u_1 \end{pmatrix} + \beta \begin{pmatrix} \vec{x}_{02} \\ u_2 \end{pmatrix} \right] = \alpha \Gamma \begin{pmatrix} \vec{x}_{01} \\ u_1 \end{pmatrix} + \beta \Gamma \begin{pmatrix} \vec{x}_{02} \\ u_2 \end{pmatrix}.$$

§ 470 Die zweite Einschränkung an das System, die Zeitinvarianz, bedeutet anschaulich, dass eine Verschiebung von Anfangs- und Eingangsgrößen um ein Zeitintervall  $T$  an den Zustandsgrößen keine weitere Änderung als eine Verschiebung um eben dieses Zeitintervall bewirkt:

$$\vec{x}(t - T) = \Gamma \begin{pmatrix} \vec{x}_0 \\ u(t - T) \end{pmatrix}.$$

§ 471 Für ein lineares System gilt die folgende Definition:

**Definition 6** Lineare Systeme lassen sich charakterisieren durch die Gleichungen

$$\dot{\vec{x}}(t) = \mathbf{A}\vec{x}(t) + \vec{b}u(t) \quad \text{und} \quad y(t) = \vec{c} \cdot \vec{x}(t) + du(t). \quad (4.2)$$

Die erste Gleichung ist eine Differentialgleichung, die die Entwicklung der Zustandsgrößen  $\vec{x}$  beschreibt. Die Systemmatrix  $\mathbf{A}$  und der Vektor  $\vec{b}$  legen die Systemeigenschaften fest. Die zweite Gleichung gibt den Zusammenhang zwischen Eingangsgröße  $u(t)$  und Ausgangsgröße  $y(t)$  in Abhängigkeit vom Zustand  $\vec{x}$  des Systems. Damit wird über den Zustandsvektor  $\vec{x}$  auch der Anfangszustand  $\vec{x}_0$  des Systems berücksichtigt (z.B. ob der Kondensator des Tiefpass anfangs geladen war oder nicht – oder ob ein Neuron gerade gefeuert hat oder nicht).<sup>3</sup>

§ 472 Betrachten wir als System ein RC-Netzwerk aus zwei Widerständen  $R_1$  und  $R_2$ , zwei Kondensatoren  $C_1$  und  $C_2$  sowie einer Spannungsquelle  $u$  wie im rechten Teil von Abb. 4.1 dargestellt. Die Spannung  $u(t)$  ist die Eingangsgröße des Systems, den Strom  $i_R$  durch den Widerstand  $R_1$  wählen wir als Ausgangsgröße.

<sup>3</sup>Die in der Definition 6 gegebenen Systemgleichungen werden auch im Block State Space benötigt, siehe § 158.

§ 473 Wir haben in § 464 bereits festgestellt, dass der Spannungsabfall über einem Kondensator eine mögliche Zustandsvariable ist, d.h. wir können einen Zustandsvektor  $\vec{x} = (u_{C1}, u_{C2})$  verwenden und erhalten für die Kondensatorströme  $i_{C1} = i_{R1} = C_1 \dot{x}_1$  und  $i_{C2} = C_2 \dot{x}_2$ . Die Kirchhoff'schen Gesetze liefern ferner für die rechte Masche  $x_2 = R_1 i_{C1} + x_1$  und für die linke Masche  $u = R_2(i_{C2} + i_{C1}) + x_2$ . Zusammenfassen liefert die beiden Differentialgleichungen

$$\dot{x}_1 = -\frac{1}{\tau_1} x_1 + \frac{1}{\tau_1} x_2 \quad \text{und} \quad \dot{x}_2 = \frac{\varrho}{\tau_2} x_1 - \frac{1+\varrho}{\tau_2} x_2 + \frac{1}{\tau_2} u$$

mit den Abkürzungen  $\tau_1 = R_1 C_1$ ,  $\tau_2 = R_2 C_2$  und  $\varrho = R_2/R_1$ . Für die Ausgangsgröße  $y = i_{C1}$  gilt ferner  $y = (x_2 - x_1)/R_1$ .

§ 474 Das komplette mathematische Modell können wir in Matrixschreibweise darstellen:

$$\dot{\vec{x}} = \begin{pmatrix} -1/\tau_1 & 1/\tau_1 \\ \varrho/\tau_2 & -(1+\varrho)/\tau_2 \end{pmatrix} \vec{x} + \begin{pmatrix} 0 \\ 1/\tau_2 \end{pmatrix} u \quad \text{und} \quad y = \begin{pmatrix} -1/R_1 \\ 1/R_1 \end{pmatrix} \vec{x}. \quad (4.3)$$

In dieser Formulierung enthält die Systemmatrix nicht mehr explizit die Parameter einzelner Bauteile wie Widerstände oder Kapazitäten sondern Zeitkonstanten und Widerstandsverhältnisse, die die Eigenschaften einer Schaltung unabhängig von den konkret verwendeten Bauteilen beschreiben. Formal entspricht (4.3) der linken Gleichung in (4.2); diese inhomogene DGL beschreibt die Entwicklung der Systemgrößen  $\vec{x}$  für einen Eingang  $\vec{u}$ .

**Querverbindung 5** Der Übergang zu den Systemgrößen  $\tau$  und  $\varrho$  ist analog zur Einführung der Frequenz  $\omega_o$  in der Schwingungsgleichung – man will von der physikalischen Realisierung unabhängig werden.

### 4.1.3 Nicht-Eindeutigkeit der Zustandsvariablen

§ 475 In den bisherigen Beispielen wurde die Wahl der Zustandsvariablen nicht begründet. In § 472ff z.B. haben wir die Spannungsabfälle über den Kondensatoren als Zustandsvariablen gewählt. Nach § 464 ist dieser Ansatz gerechtfertigt. Die Wahl der Zustandsvariablen ist jedoch nicht eindeutig: für ein gegebenes System gibt es verschiedene Zustandsbeschreibungen. Deren Zahl wächst mit der Zahl der zur Verfügung stehenden Bauteile und damit mit der Zahl der möglichen Zustände.

§ 476 Stellen wir dazu den Zustandsvektor  $\vec{x}$  im Zustandsraum mit Einheitsvektoren dar:

$$\vec{x} = x_1 \vec{e}_1 + x_2 \vec{e}_2 + \dots + x_n \vec{e}_n = \sum_{i=1}^n x_i \vec{e}_i.$$

Die  $n$  Einheitsvektoren sind linear unabhängig; sie bilden die Basis des Zustandsraums.

§ 477 Der gleiche Zustandsraum lässt sich auch mit beliebigen anderen Vektoren aufspannen – sofern sie linear unabhängig sind. In einem durch die Einheitsvektoren  $\vec{t}_i$  aufgespannten Raum lässt sich der Zustandsvektor  $\vec{x}$  schreiben als

$$\vec{x} = z_1 \vec{t}_1 + z_2 \vec{t}_2 + \dots + z_n \vec{t}_n = \mathbf{T} \vec{z}$$

mit

$$\mathbf{T} = (\vec{t}_1, \vec{t}_2, \dots, \vec{t}_n) \quad \text{und} \quad \vec{z} = (z_1, z_2, \dots, z_n).$$

§ 478 Mit Hilfe der aus den Basisvektoren gebildeten Transformationsmatrix  $\mathbf{T}$  lässt sich das mathematische Modell

$$\dot{\vec{x}} = \mathbf{A} \vec{x} + \vec{b} u \quad \text{mit} \quad y = \vec{c} \cdot \vec{x} + du$$

überführen in ein neues Modell

$$\dot{\vec{z}} = (\mathbf{T}^{-1} \mathbf{A} \mathbf{T}) \vec{z} + (\mathbf{T}^{-1} \vec{b}) u \quad \text{mit} \quad y = (\vec{c} \mathbf{T}) \vec{z} + du$$

in dem statt der Zustandsvariablen  $\vec{x}$  die neue Zustandsvariable  $\vec{z}$  verwendet wird.



§ 479 Für diese Transformation gilt:

- die wichtigen Eigenschaften und Merkmale des System hängen nicht von der Wahl der Zustandsvariablen ab, bleiben also bei der Transformation unverändert.
- die Wahl einer geeigneten Transformation kann die rechentechnische Behandlung des Problems vereinfachen.

§ 480 Zur Illustration des letzten Punkts betrachten wir nochmals das Beispiel in § 472ff. Als Werte für die Systemparameter wählen wir  $\tau_1 = 1/3$ ,  $\tau_2 = 3/4$  und  $\rho = 1/2$ . Die Zustandsgleichung wird

$$\dot{\vec{x}} = \begin{pmatrix} -3 & 3 \\ 2/3 & -2 \end{pmatrix} \vec{x} + \begin{pmatrix} 0 \\ 4/3 \end{pmatrix} u.$$

Mit der Transformationsmatrix

$$\mathbf{T} = \begin{pmatrix} 1 & 1 \\ 2/3 & -1/3 \end{pmatrix}$$

ergibt sich die neue Differentialgleichung zu

$$\dot{\vec{z}} = \begin{pmatrix} -1 & 0 \\ 0 & -4 \end{pmatrix} \vec{z} + \begin{pmatrix} 4/3 \\ -4/3 \end{pmatrix} u,$$

das System hat sich also auf zwei entkoppelte Differentialgleichungen bezüglich der neuen Zustandsvariablen  $z_1$  und  $z_2$  reduziert und ist damit einfach zu lösen.

**Zwischenrechnung 33** Erstellen Sie das entsprechende Modell in SIMULINK. Bestimmen Sie Lösungen für verschiedene Eingänge (Frequenz(gemisch), Signalform). Diskutieren Sie, ob die jeweiligen Lösungen plausibel sind.

**Wiederholungsfrage 8** Und wie findet man eine geeignete Transformationsmatrix? Aus mathematischer Sicht sollte Ihnen das Problem bereits aus der Anfangsvorlesung 'Lineare Algebra' bekannt sein.

§ 481 Diese Entkopplung der Differentialgleichungen reflektiert die Tatsache, dass das Gesamtsystem in zwei Teilsysteme zerlegt werden kann, für die die Übertragungsfunktionen getrennt bestimmt und anschließend geeignet kombiniert werden können.

## 4.2 Lösung der Systemgleichungen

§ 482 Zur Lösung der Systemgleichungen gibt es neben der oben indirekt erwähnten Diagonalisierung der Systemmatrix zwei Standard-Verfahren: die Lösung des Systems gewöhnlicher inhomogener Differentialgleichung erster Ordnung oder die Laplace Transformation. Letztere hat den Vorteil, dass nur eine algebraische Gleichung zu lösen ist – um den Preis, eine Transformation und deren Rücktransformation durchzuführen. Da die Transformationen tabelliert sind, lässt sich die algebraische Lösung leicht in einen Algorithmus einbauen.

### 4.2.1 Wiederholung: Lösung im Zeitbereich

§ 483 Die Lösung im Zeitbereich ist mathematisch trivial, die nötigen Handwerkszeuge haben Sie bereits in früheren Veranstaltungen kennen gelernt. Sie wird hier nur kurz wiederholt, um an das Auftreten des Faltungsintegrals zu erinnern und damit die Laplace Transformation interessant zu machen.

§ 484 Die Differentialgleichung  $\dot{\vec{x}} = \mathbf{A}\vec{x} + \vec{b}u$  ist eine inhomogene DGL erster Ordnung. Ihre Lösung ergibt sich durch die Superposition der allgemeinen Lösung der homogene DGL  $\dot{\vec{x}} = \mathbf{A}\vec{x}$  (freies System, verschwindende Eingangsgröße  $u$ ) und einer partikulären Lösung der inhomogenen DGL (angeregtes System).

### Homogener Anteil

§ 485 Da wir konstante Koeffizienten voraussetzen können, bietet sich zur Lösung des homogenen Differentialgleichungssystems  $\vec{x} = \mathbf{A}\vec{x}$  der Exponentialansatz an:

$$\vec{x}_h(t) = \vec{x}_0 e^{\mathbf{A}t} = \sum_{k=0}^{\infty} \frac{(\mathbf{A}t)^k}{k!} \vec{x}_0 = \Phi(t) \vec{x}_0. \quad (4.4)$$

Der einzige Unterschied zur Standardanwendung des Exponentialansatz auf eine einzelne DGL ist die Matrix im Exponenten. Die Erweiterung der Exponentialfunktion auf eine konstante Matrix  $\mathbf{A}$  im Exponenten lässt sich durch Reihenentwicklung der Exponentialfunktion in die rechts stehende Reihe erreichen. Die Eigenschaften der Exponentialfunktion mit einer Matrix als Argument entsprechen denen einer Exponentialfunktion mit reellem oder komplexen Argument:

$$\begin{aligned} e^{\mathbf{A}t_1} e^{\mathbf{A}t_2} &= e^{\mathbf{A}(t_1+t_2)}, \\ (e^{\mathbf{A}t})^{-1} &= e^{-\mathbf{A}t}, \\ \frac{d}{dt}(e^{\mathbf{A}t}) &= \mathbf{A} e^{\mathbf{A}t}, \\ \int_0^t e^{\mathbf{A}t} dt &= \mathbf{A}^{-1}(e^{\mathbf{A}t} - \mathbf{E}) \quad (\mathbf{A} \text{ regulär vorausgesetzt}), \\ e^{\mathbf{A}t} e^{\mathbf{B}t} &= e^{(\mathbf{A}+\mathbf{B})t} \quad \text{nur für } \mathbf{AB} = \mathbf{BA}. \end{aligned}$$

**Zwischenrechnung 34** Machen Sie sich die einzelnen Regeln durch Anwendung der Reihenentwicklung aus (4.4) klar.

**Verständnisfrage 45** Bei der Integration wird eine reguläre Matrix vorausgesetzt. Was bedeutet das? Begründen Sie diese Einschränkung. Warum gilt in der letzten Zeile die Addition nur für den Fall, dass die Matrizen bei Multiplikation ausnahmsweise kommutativ sind? Was bedeutet das anschaulich?

**Wiederholungsfrage 9** Normalerweise lösen wir ein System gekoppelter Differentialgleichungen mit Hilfe von Eigenwerten und Eigenvektoren, wie in Abschn. 1.2.1. Ist das ein ganz anderes Verfahren oder ist das eine andere Formulierung des hier verwendeten Exponentialansatzes?

§ 486 Mit dem Exponentialansatz erhalten wir als Lösung der homogenen Gleichung

$$\vec{x}_h(t) = e^{\mathbf{A}t} \vec{\chi} = \sum_{k=0}^{\infty} \frac{(\mathbf{A}t)^k}{k!} \vec{\chi} = \Phi(t) \vec{\chi}$$

und mit der Anfangsbedingung  $\vec{\chi} = \vec{x}_0$

$$\vec{x}_{\text{hom}} = \Phi(t) \vec{x}_0.$$

§ 487 Die Transitionsmatrix  $\Phi(\mathfrak{t})$  erfüllt die Differentialgleichung des freien Systems, da gilt

$$\dot{\Phi} = \mathbf{A}\Phi. \quad (4.5)$$

Sie beschreibt die Transition des freien Systems von einem Zustand zur Zeit  $t_0$  in den Zustand zur Zeit  $t$ .

### Inhomogener Anteil

§ 488 Zusätzlich zu der allgemeinen Lösung der homogenen DGL benötigen wir eine partikuläre Lösung des inhomogenen Differentialgleichungssystems 1. Ordnung. Diese lässt sich wie bei der gewöhnlichen DGL durch Variation der Konstanten finden: statt die Integrationskonstante  $\chi$  mit Hilfe des Anfangswerts zu bestimmen, machen wir einen Ansatz, der eine zeitliche Abhängigkeit von  $\chi$  zu lässt.<sup>4</sup>

<sup>4</sup>Ein alternatives Verfahren zur Lösung des inhomogenen Anteils ist das Aufsuchen einer partikulären Lösung durch einen Ansatz ähnlich dem der Inhomogenität. Dieses Verfahren ist aber per definitionem von

§ 489 Auf unser System übertragen bedeutet dies:

$$\vec{x}_s = \Phi(t) \vec{z}(t)$$

mit  $\vec{z}(t)$  als einer noch nicht bekannten vektoriellen Funktion. Dieser Ansatz wird in die inhomogene DGL eingesetzt und liefert unter Berücksichtigung von (4.5)

$$\Phi \dot{\vec{z}} = \vec{b}u.$$

Durch Multiplikation mit dem Inversen der Transitionsmatrix ergibt sich die DGL für  $\vec{z}(t)$

$$\dot{\vec{z}} = \Phi^{-1} \vec{b}u$$

mit der Lösung

$$\vec{z}(t) = \int_0^t \Phi^{-1}(\tau) \vec{b}u(\tau) d\tau = \int_0^t \Phi(-\tau) \vec{b}u(\tau) d\tau.$$

**Verständnisfrage 46** Begründen Sie  $\Phi(-\tau) = \Phi^{-1}(\tau)$  anschaulich und formal. Welche mathematische und welche physikalischen Implikationen hat dieser Zusammenhang?

§ 490 Die Lösung der inhomogenen Gleichung ist damit

$$\vec{x}_s(t) = \Phi(t) \int_0^t \Phi(-\tau) \vec{b}u(\tau) d\tau = \int_0^t \Phi(t) \Phi(-\tau) \vec{b}u(\tau) d\tau = \int_0^t \Phi(t-\tau) \vec{b}u(\tau) d\tau \quad (4.6)$$

Sie entsteht durch Faltung der Funktionen  $\Phi(t)$  und  $\vec{b}u(t)$ . Auf Grund dieser Faltung ist die Lösung zum Zeitpunkt  $t$  nicht alleine von dem momentanen Wert der Eingangsgröße abhängig sondern vom ihrem gesamten Verlauf im Intervall  $[0,t]$ ; das System hat eine Erinnerung an seine früheren Zustände. Physikalisch ist dieses Erinnerungsvermögen bei integrierenden Systemkomponenten wie Kondensatoren offensichtlich.

§ 491 Rechentechisch ist es die Faltung, die die Lösung im Zeitbereich aufwendig macht. Spätestens an dieser Stelle könnte man auf den Gedanken kommen, alles weitere im Frequenzbereich zu rechnen um die Faltung durch eine Multiplikation zu ersetzen. Vorsicht: die Laplace Transformation scharrt schon mit den Hufen und harrt ihres Einsatzes.

### Gesamtlösung

§ 492 Die Gesamtlösung ergibt sich durch Superposition der allgemeinen Lösung der homogenen und der partikulären Lösung der inhomogenen DGL zu:

$$\vec{x} = \vec{x}_h + \vec{x}_s = \Phi(t) \vec{x}_0 + \int_0^t \Phi(t-\tau) \vec{b}u(\tau) d\tau. \quad (4.7)$$

Dabei stecken die Einflüsse des Anfangszustands  $\vec{x}_0$  nur im homogenen, die der Eingangsgröße  $u(t)$  nur im inhomogenen Teil der Lösung.<sup>5</sup> Die Ausgangsgröße erhalten wir daraus zu

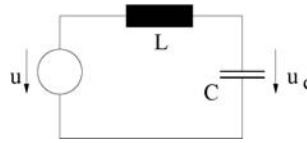
$$y(t) = \vec{c} \Phi(t) \vec{x}_0 + \vec{c} \int_0^t \Phi(t-\tau) \vec{b}u(\tau) d\tau + du(t)$$

---

der Inhomogenität abhängig und lässt sich daher nicht auf beliebige Eingangsgrößen/Inhomogenitäten verallgemeinern. Dies wird beim Tiefpass besonders anschaulich: im Zeitbereich finden wir leicht die Lösung für eine harmonische Inhomogenität – bei einem Rechtecksignal als Inhomogenität dagegen fehlt uns schon die Idee für einen vernünftigen Ansatz.

<sup>5</sup>Das kennen Sie anschaulich bereits von der erzwungenen Schwingung: der Anfangszustand geht nur in den Einschwingvorgang (homogener Anteil) ein und wird mit zunehmender Zeit heraus gedämpft, so dass nur die der Anregung folgende erzwungene Schwingung (inhomogener Anteil) verbleibt.

Abbildung 4.2:  
Harmonischer Oszillator



oder im Fall des trivialen Anfangszustands  $\vec{x}_0 = 0$

$$y(t) = \vec{c} \int_0^t \Phi(t-\tau) \vec{b} u(\tau) d\tau + du(t).$$

§ 493 Mit Hilfe einer Gewichtsfunktion

$$g(t) = \vec{c} \Phi(t) \vec{b} \quad (4.8)$$

können wir dies zu einer einfachen Eingangs–Ausgangs–Relation umschreiben:

$$y(t) = \int_0^t g(t-\tau) u(\tau) d\tau + du(t). \quad (4.9)$$

Der Wert der Ausgangsgröße  $y$  ergibt sich daher im wesentlichen durch Faltung der Gewichtsfunktion  $g$  mit der Eingangsgröße  $u$ .

**Verständnisfrage 47** Was soll die Einschränkung ‘im wesentlichen’ in dieser Aussage?

§ 494 Die Gewichtsfunktion ist für ein System charakteristisch, sie bildet gleichsam seinen Fingerabdruck: die Gewichtsfunktion ist invariant gegenüber einer Zustandstransformation.<sup>6</sup>

### Beispiel: Harmonischer Oszillator

§ 495 Betrachten wir, da wir uns ja bereits in der Einleitung intensiv mit seiner mechanischen Realisierung beschäftigt haben, als Beispiel einen linearen Oszillator als Serienschaltung aus einer Induktivität  $L$ , einem Kondensator  $C$  und einer Spannungsquelle, d.h. einen Serienschwingkreis ohne Dämpfung durch einen Widerstand  $R$  – das entspricht der angeregten Schwingung bei verschwindender Dämpfung. Als Zustandsgrößen wählen wir den Strom  $i_L$  durch die Induktivität und die Spannung  $u_C$  an der Kapazität. Die Eingangsgröße ist die Spannung  $u(t)$ . Das Systemverhalten wird beschrieben durch  $i_L = C \dot{u}_C$  und  $u = L \dot{i}_L + u_C$ .

§ 496 Mit den Zustandsvariablen  $x_1 = u_C \sqrt{C}$  und  $x_2 = i_L \sqrt{L}$  sowie der skalierten Zeit  $\tau = t/\sqrt{LC} = t/\omega_0$  erhalten wir die Differentialgleichung

$$\dot{x} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ \sqrt{C} \end{pmatrix} u.$$

§ 497 Die Transitionsmatrix bestimmen wir durch Auswertung der unendlichen Reihe

$$\begin{aligned} e^{A\tau} &= \sum_{k=0}^{\infty} \frac{(A\tau)^k}{k!} = \mathbf{E} \left( 1 - \frac{\tau^2}{2!} + \frac{\tau^4}{4!} - \frac{\tau^6}{6!} \dots \right) + \mathbf{A} \left( \tau - \frac{\tau^3}{3!} + \frac{\tau^5}{5!} - \frac{\tau^7}{7!} \dots \right) \\ &= \mathbf{E} \cos \tau + \mathbf{A} \sin \tau = \begin{pmatrix} \cos \tau & \sin \tau \\ -\sin \tau & \cos \tau \end{pmatrix} = \begin{pmatrix} \cos \frac{t}{\sqrt{LC}} & \sin \frac{t}{\sqrt{LC}} \\ -\sin \frac{t}{\sqrt{LC}} & \cos \frac{t}{\sqrt{LC}} \end{pmatrix}. \end{aligned}$$

Das Ergebnis hat die Form einer Drehmatrix mit einem linear mit der Zeit wachsenden Drehwinkel  $t/\sqrt{LC}$ : der Zustandsvektor rotiert mit konstanter Winkelgeschwindigkeit im Zustandsraum, die Trajektorien des Systems sind also Kreise. Dieser lineare Oszillator hat, wie erwartet, die Kreisfrequenz

$$\omega_0 = \frac{1}{\sqrt{LC}}.$$

<sup>6</sup>Das sollte sie auch sein: der Output eines Systems auf einen gegebenen Input sollte nicht davon abhängen, welches System linear unabhängiger Zustandsvariablen ich zur Beschreibung des Systems verwendet habe.

**Verständnisfrage 48** Wo kommt die Einheitsmatrix in der Reihenentwicklung her?

**Verständnisfrage 49** Was rotiert denn da? Machen Sie sich klar, welche Größen in der Matrix beschrieben werden!

**Verständnisfrage 50** Sind die Trajektorien wirklich Kreise oder ist der Kreis nur eine mögliche Trajektorie im Phasenraum?

§ 498 Für die Ausgangsgröße, die Spannung am Kondensator, ergibt sich

$$y = \begin{pmatrix} \frac{1}{\sqrt{C}} & 0 \end{pmatrix} \vec{x};$$

die Gewichtsfunktion ergibt sich gemäß (4.8) zu

$$g(t) = \begin{pmatrix} \frac{1}{\sqrt{C}} & 0 \end{pmatrix} \begin{pmatrix} \cos(\omega_0 t) & \sin(\omega_0 t) \\ -\sin(\omega_0 t) & \cos(\omega_0 t) \end{pmatrix} \begin{pmatrix} 0 \\ \sqrt{C} \end{pmatrix} = \sin(\omega_0 t).$$

**Zwischenrechnung 35** Harmonische Oszillatoren haben wir in Kapitel 1 bereits hinreichend strapaziert. Implementieren Sie trotzdem auch diesen nochmals in SIMULINK – diesmal unter Verwendung des Blocks State Space.

### Beispiel: Tiefpass

§ 499 Kehren wir nochmals zum Tiefpass aus § 464 zurück. Die allgemeine Lösung im Zeitbereich (4.9) reduziert sich auf den eindimensionalen Fall:

$$y(t) = \int_0^t g(t-\tau)u(\tau) d\tau + du(t).$$

§ 500 Zur Bestimmung der Gewichtsfunktion (4.8)  $g(t) = \vec{c}\Phi(t)\vec{b}$  benötigen wir die Transitionsmatrix, die sich in diesem Fall auf eine  $1 \times 1$ -Matrix reduziert. Mit den Größen wie im linken Teil von Abb. 4.1 und (4.1) erhalten wir als Systemgleichung für  $u_C$

$$\dot{u}_C = \frac{u}{RC} - \frac{u_C}{RC}, \quad (4.10)$$

d.h. in der Systemgleichung  $\dot{\vec{x}} = \mathbf{A}\vec{x} + \vec{b}u$  erhalten wir für die Systemmatrix  $\mathbf{A} = -((RC)^{-1})$  und den Vektor  $\vec{b} = (RC)^{-1}$ . Da die Systemgröße  $u_C$  gleichzeitig die Ausgangsgröße ist, erhalten wir aus der Gleichung  $y = \vec{c}\vec{x} + du$  für  $\vec{c} = (1)$  und  $d = 0$ . Zur Bestimmung der Gewichtsfunktion benötigen wir noch die Transitionsmatrix  $\Phi$ , die, da sie die DGL des freien Systems erfüllt,  $\dot{\Phi} = \mathbf{A}\Phi$ , sich in unserem Falle reduziert auf  $\Phi = e^{-t/RC}$ . Die Gewichtsfunktion (4.8) wird damit

$$g = \vec{c}\Phi\vec{b} = \frac{e^{-t/RC}}{RC}.$$

§ 501 Für eine harmonische Eingangsfunktion  $u(t) = Ue^{i\omega t}$  erhalten wir gemäß (4.9) als Ausgangsgröße<sup>7</sup>

$$y = \int_0^t \frac{1}{RC} e^{-(t-\tau)/RC} U e^{i\omega\tau} d\tau = \frac{U}{RC} \int_0^t e^{\frac{-t+\tau+i\omega RC\tau}{RC}} d\tau = \frac{u}{1+i\omega RC}$$

<sup>7</sup>Die Lösung für eine harmonische Eingangsgröße kann man leicht aus (4.10) bestimmen. Die angelegte Spannung  $u$  muss gleich der Summe aus dem Spannungsabfall  $u_R$  über dem Widerstand und dem Spannungsabfall  $u_C$  über dem Kondensator sein:  $u = u_R + u_C$ . Der Spannungsabfall über dem Widerstand ist durch den Strom  $i$  und den Widerstand  $R$  bestimmt:  $u_R = iR$ . Der Strom wiederum ergibt sich aus der Änderung der Ladung auf dem Kondensator zu  $i = C\dot{u}_C$  mit  $C$  als der Kapazität. Damit erhalten wir als DGL

$$u = u_R + u_C = iR + u_C = CR\dot{u}_C + u_C.$$

Für ein harmonisches Signal  $u \sim e^{i\omega t}$  ist  $\dot{u} = i\omega u$ , also auch  $\dot{u}_C = i\omega u_C$  und damit

$$u = (i\omega RC + 1)u_C.$$

Da gleichzeitig  $u$  das Eingangssignal ist,  $u = u_{\text{ein}}$  und  $u_C$  das Ausgangssignal,  $u_C = u_{\text{aus}}$ , ergibt sich für die

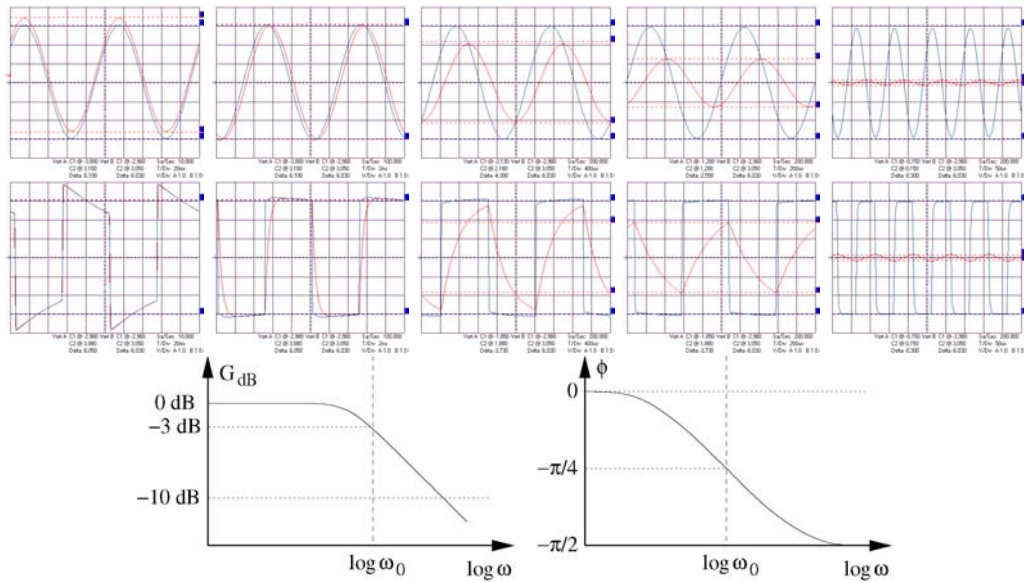


Abbildung 4.3: Tiefpass: Ein- und Ausgangssignal für unterschiedliche Frequenzen und Signalformen; *Oben*: harmonisches Eingangssignal, *Mitte*: Rechteckfunktion als Eingang. Eingangsfrequenzen von links nach rechts: 10 Hz, 100 Hz, 500 Hz, 1 kHz und 10 kHz; die Grenzfrequenz beträgt 525 Hz; *unten*: Betrag und Phase der Übertragungsfunktion für ein harmonisches Eingangssignal dargestellt im Bode-Diagramm

Physikalisch sinnvoll ist nur der Realteil:

$$u_{\text{aus}} = \Re(y) = \frac{U}{\sqrt{1 + \omega^2 R^2 C^2}} \sin(\omega t + \varphi).$$

§ 502 Abbildung 4.3 zeigt die Ein- und Ausgangssignale für verschiedene Frequenzen und Formen des Eingangspulse. Die Größen sind experimentell bestimmt mit Hilfe eines Tiefpasses mit  $R = 1 \text{ k}\Omega$  und  $C = 0.33 \text{ }\mu\text{F}$ , d.h. einer Grenzfrequenz  $f_g = (2\pi RC)^{-1}$  von 525 Hz. Die obere Reihe zeigt Eingang (blau) und Ausgang (rot) für ein harmonisches Signal für Frequenzen von 10 Hz, 100 Hz, 500 Hz, 1 kHz und 10 kHz. Liegt die Eingangsfrequenz deutlich unter der Grenzfrequenz, so sind Eingangs- (blau) und Ausgangssignal (rot) nahezu identisch. Mit zunehmender Eingangsfrequenz nimmt die Amplitude des Ausgangssignals ab, zusätzlich erfolgt eine mit der Eingangsfrequenz zunehmende Phasenverschiebung. Bei der

Übertragungsfunktion im Zeitbereich

$$G(t) = \frac{u_{\text{aus}}}{u_{\text{ein}}} = \frac{1}{1 + i\omega RC}.$$

Für den Betrag der Übertragungsfunktion ergibt sich

$$|G(t)| = \frac{1}{\sqrt{1 + \omega^2 R^2 C^2}} = \frac{1}{\sqrt{1 + (\omega/\omega_0)^2}} \quad \text{mit} \quad \omega_0 = \frac{1}{RC},$$

die Phase der Ausgangsspannung ist

$$\tan \varphi = \frac{\Im\{G\}}{\Re\{G\}} = -\omega RC = -\frac{\omega}{\omega_0} \quad \text{oder} \quad \varphi = \arctan(\omega RC).$$

In komplexer Form lässt sich das Ausgangssignal (natürlich wieder nur für den harmonischen Input) schreiben als

$$u_C = \Re\{Gu\} = \Re\{|G| U e^{i(\omega t + \varphi)}\} \quad \text{mit} \quad G = \frac{1}{1 + i\omega RC}.$$

Betrag und Phase der Übertragungsfunktion stellt man im Bode-Diagramm dar.

Grenzfrequenz von 525 Hz (Kurve für 500 Hz gezeigt), ist die Amplitude des Ausgangssignal auf  $1/\sqrt{2}$  abgesunken und um  $\pi/4$  verschoben.<sup>8</sup>

§ 503 Dieses Absinken der Amplitude mit zunehmender Frequenz wird auch in der unteren Zeile für das Rechtecksignal deutlich.<sup>9</sup> Während das harmonische Signal unter Wahrung der Form – wenn auch mit einer Phasenverschiebung – übertragen wird, findet beim Rechteck eine Verformung statt: bei Frequenzen deutlich kleiner der Grenzfrequenz bleibt die Form gewahrt, mit zunehmender Frequenz werden die Flanken weniger steil. Physikalisch lässt sich dies durch die endliche Zeit zum Aufladen des Kondensators in der ansteigenden Flanke erklären bzw. der zum Entladen in der abfallenden Flanke benötigten Zeit.

§ 504 Mathematisch lässt sich diese Verformung ebenfalls begründen. Das Rechtecksignal können wir als eine Überlagerung unendlich vieler harmonischer Signale interpretieren (vgl. Abb. 3.7). Eine Fourier Analyse gibt daher ein sich zu unendlich hohen Frequenzen erstreckendes Spektrum. Von diesem können die niederfrequenten Anteile den Tiefpass nahezu ungedämpft passieren während die höheren Frequenzen mit zunehmender Frequenz immer stärker gedämpft werden. Diese sind aber für die steile Flanke erforderlich, so dass die Signalform bei der Übertragung nicht erhalten werden kann. In Abb. 3.19 haben wir dieses Verfahren bereits verwendet, die Teilbilder in der oberen Reihe dort entsprechen sinngemäß der unteren Reihe in Abb. 4.3.

§ 505 Ist die Eingangsgröße ein  $\delta$ -Impuls, so ergibt sich als Systemantwort  $u(t) = U_0 e^{-t/RC}$  – das ist auch anschaulich, da sich in diesem Fall der durch den  $\delta$ -Puls aufgeladene Kondensator über den Widerstand entlädt.

§ 506 Ist die Eingangsgröße eine Sprungfunktion, so verschiebt sich die untere Integrationsgrenze auf den Einsatzpunkt  $t_S$  der Sprungfunktion:

$$y = \int_{t_S}^t \frac{U}{RC} e^{-(t-\tau)/RC} d\tau = U(1 - e^{-(t-t_S)/\tau}).$$

Anschaulich sehen wir dabei nur dem Kondensator beim Aufladen zu. Für  $t \gg t_S$  verschwindet der zweite Term in der Klammer. Auf diese Weise können wir uns auch anschaulich die Verformung des Rechteckimpulses herleiten.

**Zwischenrechnung 36** Überprüfen Sie Abb. 4.3 in SIMULINK unter (a) Verwendung der Übertragungsfunktion und (b) Verwendung des Blocks State Space.

**Zwischenrechnung 37** Untersuchen Sie das Verhalten beider Modelle bei nicht harmonischen Eingangssignalen (z.B. Rechteck).

## 4.2.2 Lösung mit Hilfe einer Laplace Transformation

§ 507 Für die Laplace Transformierte des Systems  $\dot{\vec{x}}(t) = \mathbf{A}\vec{x}(t) + \vec{b}u(t)$  erhalten wir unter Berücksichtigung von (3.15) und (3.17)

$$s\vec{x}(s) - \vec{x}_0 = \mathbf{A}\vec{x}(s) + \vec{b}u(s).$$

§ 508 Im Bildbereich ergibt sich damit eine einfache algebraische Beziehung für den Zustandsvektor  $\vec{x}(s)$ :

$$\vec{x}(s) = (s\mathbf{E} - \mathbf{A})^{-1}\vec{x}_0 + (s\mathbf{E} - \mathbf{A})^{-1}\vec{b}u.$$

<sup>8</sup>Das ist die Definition der Grenzfrequenz: die Amplitude des Ausgangssignals beträgt an der Grenzfrequenz  $1/\sqrt{2}$  oder 70.7% der des Eingangssignals.

<sup>9</sup>Linkes Teilbild bitte entschuldigen, hier war der Ausgang des Signalgenerators zu stark belastet, um eine saubere Rechteckform zu erzeugen. Was mal wieder zeigt, dass es doch wesentlich einfacher ist, alles zu simulieren als sich mit echten Experimenten abzugeben.

Der erste Term hängt nur vom Anfangszustand  $\vec{x}_0$  ab, entspricht also der homogenen Lösung im Zeitbereich; der zweite Term hängt nur von der Eingangsgröße  $u$  ab, entspricht also der inhomogenen Lösung im Zeitbereich:

$$\vec{x}_h = (s\mathbf{E} - \mathbf{A})^{-1}\vec{x}_0 \quad \text{und} \quad \vec{x}_s = (s\mathbf{E} - \mathbf{A})^{-1}\vec{b}u.$$

In beiden Termen tritt die Resolvente

$$\mathcal{A} = (s\mathbf{E} - \mathbf{A})^{-1} \tag{4.11}$$

der Matrix  $\mathbf{A}$  auf.

§ 509 Ein Vergleich mit der Lösung (4.7) im Zeitbereich zeigt den Zusammenhang zwischen der transformierten und der nicht-transformierten Funktion:

$$\mathcal{L}\{\Phi(t)\vec{x}_0\} = \mathcal{L}\{\vec{x}_h(t)\} = \vec{x}_h(s)$$

und

$$\mathcal{L}\left\{\int_0^t \Phi(t-\tau)\vec{b}u(\tau) d\tau\right\} = \mathcal{L}\{\vec{x}_s(t)\} = \vec{x}_s(s).$$

Die letzte Gleichung verdeutlicht den Vorteil der Laplace Transformation: die Faltung im Zeitbereich wird in eine Multiplikation im Bildbereich transformiert.

§ 510 Außerdem erhalten wir einen Zusammenhang zwischen der Resolvente und der Transitionsmatrix: die Resolvente von  $\mathbf{A}$  ist die Laplace Transformierte der Transitionsmatrix  $\mathcal{L}\{\Phi(t)\} = \Phi(s) = (s\mathbf{E} - \mathbf{A})^{-1}$  oder formuliert mit Hilfe der Matrix-Exponentialfunktion

$$\mathcal{L}\{e^{At}\} = (s\mathbf{E} - \mathbf{A})^{-1}. \tag{4.12}$$

Im skalaren Fall gilt analog

$$\mathcal{L}\{e^{at}\} = \int_0^\infty e^{at} e^{-st} dt = \int_0^\infty e^{(a-s)t} dt = \left[ \frac{1}{a-s} e^{(a-s)t} \right]_0^\infty = \frac{1}{s-a} = (s1 - a)^{-1}.$$

§ 511 Das Kochrezept zur Lösung der DGL besteht also darin, die Transitionsmatrix auf folgendem Wege mit Hilfe einer Laplace Transformation zu ermitteln: aus der bekannten Systemmatrix  $\mathbf{A}$  wird deren Resolvente  $\mathcal{A} = (s\mathbf{E} - \mathbf{A})^{-1}$  gebildet. Ihre Elementen sind gebrochen rationale Funktionen und können jeweils in eine Summe von Partialbrüchen zerlegt werden. Anschließend werden die einzelnen Summanden in den Zeitbereich zurück transformiert und dort addiert.

§ 512 Greifen wir zur Erläuterung nochmals auf das RC-Netzwerk in § 480ff zurück. Für die Systemmatrix

$$\mathbf{A} = \begin{pmatrix} -3 & 3 \\ 2/3 & -2 \end{pmatrix}$$

erhalten wir mit (4.11) als Laplace Transformierte

$$(s\mathbf{E} - \mathbf{A})^{-1} = \begin{pmatrix} s+3 & 1-3 \\ -2/3 & s+2 \end{pmatrix}^{-1} = \frac{1}{(s+1)(s+4)} \begin{pmatrix} s+2 & 3 \\ 2/3 & s+3 \end{pmatrix}.$$

§ 513 Für die Rücktransformation müssen wir nur den Vorfaktor betrachten. Partialbruchzerlegung liefert

$$\frac{1}{(s+1)(s+4)} = \frac{1}{3} \left( \frac{1}{s+1} - \frac{1}{s+4} \right) = \phi_1(s).$$

Unter Berücksichtigung der Linearität (3.15) erhalten wir für die Rücktransformierten

$$\mathcal{L}^{-1}\left\{\frac{1}{(s+1)(s+4)}\right\} = \frac{1}{3} \left( \mathcal{L}^{-1}\left\{\frac{1}{s+1}\right\} - \mathcal{L}^{-1}\left\{\frac{1}{s+4}\right\} \right) = \frac{1}{3} (e^{-t} - e^{-4t})$$



und

$$\mathcal{L}^{-1} \left\{ \frac{s}{(s+1)(s+4)} \right\} = \mathcal{L}^{-1} \{s \phi_1(s)\} = \frac{1}{3} \frac{d}{dt} (e^{-t} - e^{-4t}) = \frac{1}{3} (-e^{-t} + 4e^{-4t}) .$$

Die Transitionsmatrix wird damit zu

$$\Phi(\mathbf{t}) = \begin{pmatrix} \frac{1}{3}(e^{-t} + 2e^{-4t}) & e^{-t} - e^{-4t} \\ \frac{2}{3}(e^{-t} - e^{-4t}) & \frac{1}{3}(2e^{-t} + e^{-4t}) \end{pmatrix} .$$

§ 514 Betrachten wir nochmals den linearen Oszillator aus § 495 (vereinfachte Schreibweise mit  $\omega_0 = 1$ ). Die DGL ist dann

$$\dot{\vec{x}} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \vec{x} .$$

Für die Laplace Transformierte der Transitionsmatrix ergibt sich

$$(s\mathbf{E} - \mathbf{A})^{-1} = \begin{pmatrix} s & -1 \\ 1 & s \end{pmatrix}^{-1} = \frac{1}{s^2 + 1} \begin{pmatrix} s & 1 \\ -1 & s \end{pmatrix} .$$

Die Rücktransformation des Vorfaktors liefert  $\sin t$ , d.h. wir erhalten für die Transitionsmatrix

$$\Phi(\mathbf{t}) = \begin{pmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{pmatrix} .$$

### 4.3 Übertragungsfunktion

§ 515 Bisher haben wir die Transitionsmatrix und die Gewichtsfunktion eines Systems betrachtet: mit Hilfe der Transitionsmatrix wird die Entwicklung der Systemzustände beschrieben; daraus wird die Gewichtsfunktion bestimmt, die in die Bestimmung der Ausgangsgröße eingeht.

§ 516 Für die praktische Arbeit interessanter ist jedoch die Übertragungsfunktion  $G$ , die einen direkten Zusammenhang zwischen den Ein- und Ausgangsgrößen liefert. Im Zeitbereich lässt sich die Übertragungsfunktion als Quotient  $G_t = y(t)/u(t)$  nur dann bilden, wenn die Eingangsgröße eine Eigenfunktion ist – im Falle von Tief- und Hochpass in der Form  $Ue^{\xi t}$ .

**Wiederholungsfrage 10** Was ist eine Eigenfunktion und wie bestimmen sie sich? Falls Sie bisher nicht über den Begriff gestolpert sind, nehmen Sie Anleihe in der Linearen Algebra! Dort sollten Sie auch etwas finden, an das die Resolvente erinnert.

§ 517 Im Bildbereich dagegen ist die Übertragungsfunktion

$$G_s = \frac{y(s)}{u(s)} \Big|_{\vec{x}_0=0} = \vec{c}(s\mathbf{E} - \mathbf{A})^{-1} \vec{b} + d . \quad (4.13)$$

eine universelle Größe, unabhängig von der Eingangsfunktion  $u(s)$ . Hier ist die Übertragungsfunktion die Laplace Transformierte der Gewichtsfunktion  $g(t) = \vec{c}\Phi(t)\vec{b}$ :

$$G(s) = \mathcal{L}\{g(t)\} + d .$$

§ 518 Angewandt auf ein elektr(on)isches Netzwerk wie z.B. den Tiefpass bedeutet dies, dass wir im Zeitbereich zwar eine Übertragungsfunktion  $G_t$  für harmonische Eingangsgrößen bestimmen können, nicht jedoch eine Übertragungsfunktion für allgemeine Größen. Die Übertragungsfunktion für allgemeine Größen lässt sich nur im Bildbereich bestimmen – das sollte bereits bei der anschaulichen Darstellung des Tiefpass in § 499ff deutlich geworden sein.

§ 519 Zur Bestimmung der Systemantwort können wir bei bekannter Übertragungsfunktion im Bildbereich nach folgendem Kochrezept vorgehen:

1. die Übertragungsfunktion  $G(s)$  wurde bestimmt.
2. die Eingangsfunktion  $u(t)$  wird Laplace transformiert:  $u(s)$ .

3. das Produkt  $y(s) = G(s)u(s)$  wird bestimmt.
4. das Produkt wird in elementare Beiträge zerlegt (meist Partialbruchzerlegung) und rücktransformiert.

In einer vereinfachten Version haben wir das im Zusammenhang mit Abb. 3.19 bereits durchgeführt.

**Verständnisfrage 51** In Abb. 3.19 wurde die FFT statt einer Laplace-Transformation verwendet. Begründen Sie, warum das Konzept trotzdem funktioniert.

§ 520 Gleichung (4.13) ist für den trivialen Anfangszustand  $\vec{x} = 0$  formuliert. Für einen nicht-trivialen Anfangszustand  $\vec{x}_0 \neq 0$  würden wir unter Berücksichtigung der Tatsache, dass die Transformierte des Differentialquotienten  $d\vec{x}/dt$  durch  $s\vec{x}(s) - \vec{x}_0$  gegeben ist, eine allgemeine Eingangs-Ausgangs-Relation

$$y(s) = \vec{c}(s\mathbf{E} - \mathbf{A})^{-1}\vec{x}_0 + G(s)u(s)$$

erhalten. Zum Verständnis der wesentlichen Struktur der Übertragungsfunktion ist es ausreichend, den trivialen Anfangszustand zu betrachten, da der nicht-triviale Anfangszustand nur zu einem additiven Term führt.<sup>10</sup>

§ 521 Betrachten wir als 1D-Beispiel nochmals den RC-Tiefpass aus § 499. Die Systemmatrix war  $\mathbf{A} = -((RC)^{-1})$ . Die Übertragungsfunktion im Bildbereich ergibt sich aus der Resolvente (4.11) und (4.13):

$$G(s) = \vec{c}(s\mathbf{E} - \mathbf{A})^{-1}\vec{b} + d = \left(s + \frac{1}{RC}\right)^{-1} RC. \quad (4.14)$$

Für ein harmonisches Eingangssignal  $u(t) = Ue^{i\omega t}$  wird die Laplace Transformierte

$$u(s) = \frac{U}{s - i\omega}.$$

Das Ausgangssignal im Bildbereich wird damit zu

$$y(s) = \frac{U}{RC\left(s + \frac{1}{RC}\right)(s - i\omega)}.$$

Mit Hilfe von Tabelle 3.2 erhalten wir als Rücktransformation in den Zeitbereich

$$y(t) = \frac{U}{RC\left(\frac{1}{RC} + i\omega\right)} \left(e^{-t/(RC)} - e^{i\omega t}\right) = \frac{U}{1 + i\omega RC} \left(e^{-t/(RC)} - e^{i\omega t}\right) =$$

§ 522 Setzen wir das Beispiel aus § 512 fort. Mit der der gegebenen Systemmatrix erhalten wir als Systemgleichung

$$\dot{\vec{x}} = \begin{pmatrix} -3 & 3 \\ 2/3 & -2 \end{pmatrix} \vec{x} + \begin{pmatrix} 0 \\ 4/3 \end{pmatrix} u.$$

Die Ausgangsgröße (Spannung an C1) ist in Abhängigkeit von der Zustandsvariablen  $x_1$ :

$$y = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \vec{x}.$$

Die Übertragungsfunktion im Bildbereich wird

$$G(s) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} s + 3 & -3 \\ -2/3 & s + 2 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 4/3 \end{pmatrix} = \frac{4}{(s + 1)(s + 4)}.$$

Die Anwendung dieser Übertragungsfunktion auf eine harmonische Eingangsgröße werden wir in § 549 betrachten.

<sup>10</sup>Auch physikalisch ist das sinnvoll: der Anfangszustand tritt nur im homogenen Teil der Lösung auf und wird im Laufe der Zeit weg gedämpft. Wenn wir nicht am Einschwingvorgang interessiert sind, ist seine Betrachtung nicht erforderlich – und beim Equalizer einer HiFi-Anlage sollte es egal sein, welche Ladung zufällig zum Einstellungszeitpunkt auf einem Kondensator war.

§ 523 Die Gewichtsfunktion des linearen Oszillator aus § 514 war  $g(t) = \sin t$ . Wegen  $d = 0$  ist die Übertragungsfunktion

$$G(s) = \mathcal{L}\{\sin t\} = (s^2 + 1)^{-1}.$$

Betrachten wir nun die Systemantwort auf eine Sprungfunktion (Sprungantwort),  $u(t) = 1$  für  $t \geq 0$ . Im Bildbereich ergibt sich die Antwort durch Multiplikation mit der Eingangsgröße zu

$$y(s) = G(s) \cdot u(s) = \frac{1}{s^2 + 1} \mathcal{L}\{u(t)\} = \frac{1}{(s^2 + 1)s}.$$

Partialbruchzerlegung liefert

$$y(s) = \frac{1}{s^2 + 1} \frac{1}{s} = \frac{-0.5}{s + i} + \frac{-0.5}{s - i} + \frac{1}{s}$$

und damit nach Rücktransformation als komplexe Ausgangsgröße

$$y(t) = -0.5e^{-it} - 0.5e^{it} + 1$$

und bei Reduktion auf den physikalisch sinnvollen Realteil

$$y(t) = -\cos t + 1.$$

### 4.3.1 Wichtig für SIMULINK: Struktur der Übertragungsfunktion

§ 524 Die Struktur der Übertragungsfunktion ergibt sich aus dem Bildungsgesetz der Resolventen  $\mathcal{A} = (s\mathbf{E} - \mathbf{A})^{-1}$  zu

$$G(s) = \vec{c}(s\mathbf{E} - \mathbf{A})^{-1}\vec{b} + d = \frac{\mu(s)}{\nu(s)}.$$

Dabei hat das Zählerpolynom die gleiche oder eine geringere Ordnung  $n$  als das System, das Nennerpolynom dagegen ist von  $n$ -ter Ordnung. Der kleinste gemeinsame Nenner aller Nennerpolynome ist das charakteristische Polynom  $\det(s\mathbf{E} - \mathbf{A})$ .

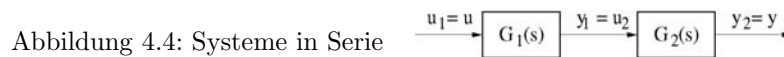
§ 525 Genau diese Struktur aus Zähler- und Nennerpolynom ist es, die auch in SIMULINK direkt übergeben wird; eine genauere Erläuterung des Blocks **Transfer Function** ist in § 161 zu finden, ein Beispiel haben wir bereits in § 46 kennen gelernt. Auch in Abschn. 2.1 wird bei der Einführung in SIMULINK auf den Block **Transfer Function** und damit die Übertragungsfunktion zurück gegriffen.

§ 526 Für das Verhalten des Systems sind die Pole und Nullstellen der Übertragungsfunktion entscheidend. Deren Definition ist eine direkte Erweiterung der aus dem Reellen bekannten Definitionen: Eine (komplexe) Zahl  $\alpha$  heißt Pol von  $G(s)$ , wenn  $\nu(\alpha) = 0$ . Und entsprechend: Eine (komplexe) Zahl  $\beta$  heißt Nullstelle von  $G(s)$ , wenn  $\mu(\beta) = 0$ .

§ 527 Ist die Eingangsgröße eine Eigenfunktion des Systems, so ist an den Nullstellen des Systemverhalten nur durch die Transitionsmatrix bestimmt und unabhängig von der Eingangsgröße. An einem Pol dagegen wird bei verschwindender Eingangsgröße die Ausgangsgröße durch einen exponentiellen Abfall beschrieben, dessen Zeitkonstante dem Pol der Übertragungsfunktion entspricht.

§ 528 Mit Hilfe der Pol- und Nullstellen lässt sich die Übertragungsfunktion in faktorisierte Form darstellen als

$$G(s) = K \frac{\prod_{i=1}^m (s - \beta_i)}{\prod_{i=1}^l (s - \alpha_i)} \quad \text{mit} \quad m \leq l \leq n.$$



### 4.3.2 Rückkopplungen und Übertragungsfunktion

§ 529 Eine Rückkopplung liegt dann vor, wenn das Ausgangssignal bzw. ein Teil desselben, auf den Eingang zurück gegeben wird. Ein einfaches Beispiel ist der Pfeifton, der entsteht, wenn ein Mikrofon sich zu nahe am Lautsprecher befindet und dadurch das Ausgangssignal (Lautsprecher) nochmals auf den Eingang (Mikrofon) gegeben und verstärkt wird. Das ist ein Beispiel für eine positive Rückkopplung. In vielen Systemen werden negative Rückkopplungen von Schaltungen eingesetzt bzw. benötigt: Aus- und Eingangssignal sind um  $\pi$  Phasen verschoben, so dass die Rückkopplung eines Teils des Ausgangssignals auf den Eingang das Eingangssignal reduziert und so ein zu großes Anwachsen der Ausgangsamplitude verhindert.

§ 530 Betrachten wir diese beiden Arten von Rückkopplungen genauer:

- bei der positiven Rückkopplung wird ein Teil des Ausgangssignals gleichphasig auf das Eingangssignal addiert, verstärkt dieses also. Der Pfeifton beim System Lautsprecher/Mikrofon ist ein Beispiel. Ein System mit positiver Rückkopplung wird daher leicht instabil und läuft gegen einen Grenzzustand; positive Rückkopplung wird auch als Mitkopplung bezeichnet. Ein einfaches Beispiel für eine positive Rückkopplung im System Erde–Atmosphäre ist das Eis–Albedo–Feedback: Eis hat eine hohe Albedo, d.h. es reflektiert einen großen Teil der einfallenden solaren Strahlung – ganz im Gegensatz zu Boden, Fels oder Meerwasser. Verringert sich aus irgendeinem Grunde (z.B. Klimaerwärmung) die Eisbedeckung, so wird weniger Sonnenstrahlung reflektiert und stattdessen absorbiert. Dies führt zu einer Erwärmung und damit zu einem Abschmelzen des noch verbliebenen Eis. Damit wird die Albedo auf einer noch größeren Fläche verringert, noch mehr einfallende Sonnenstrahlung wird absorbiert, es wird wärmer und das Eis schmilzt weiter. Durch diese positive Rückkopplung kann sich eine anfänglich kleine Erwärmung immer weiter aufschaukeln, so dass der Grenzzustand ‘eisfrei’ relativ schnell erreicht werden kann.
- bei der negativen Rückkopplung wird ein Teil des Ausgangssignals gegenphasig auf das Eingangssignal addiert. Damit wird das Eingangssignal abgeschwächt und das System stabilisiert. Negative Rückkopplung wird als Gegenkopplung bezeichnet. Das System Erde–Atmosphäre liefert auch für diese negative Rückkopplung ein Beispiel, die Druck–Temperatur–Rückkopplung: eine erhöhte Luftbewegung führt zu einer Verstärkung der Meeresströmungen. Dadurch werden Temperaturunterschiede im Ozean besser ausgeglichen, so dass es an der Meeresoberfläche keine thermischen Druckunterschiede mehr gibt und damit auch keine antreibenden Kräfte für die Luftbewegungen.

§ 531 Mit Hilfe der Übertragungsfunktion im Bildbereich ist die Beschreibung gekoppelter Systemen (bzw. umgekehrt die Zerlegung eines Systems in Teilsysteme) besonders einfach. Die beiden Grundformen gekoppelter Systeme sind ihre Serien- bzw. Parallelschaltung.

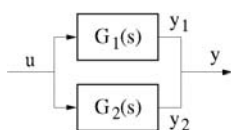
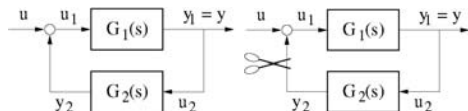
§ 532 Bei Systemen in Serie, vgl. Abb. 4.4, ist die Eingangsgröße des zweiten Systems die Ausgangsgröße des ersten. Damit ergibt sich die Übertragungsfunktion dieser Reihenschaltung durch Multiplikation der Übertragungsfunktionen der einzelnen Systeme: aus

$$y = u_2 G_2 = y_1 G_2 = u_1 G_1 G_2 = u G_1 G_2 = uG$$

ergibt sich unmittelbar

$$G(s) = G_1(s) G_2(s) . \quad (4.15)$$

In diesem Fall sind die Nullstellen und Pole von  $G$  auch Nullstellen bzw. Pole von  $G_1$  oder  $G_2$ , d.h. die ausgezeichneten Punkte jedes Einzelsystems setzen sich auch in der Übertragungsfunktion des Gesamtsystems durch.

Abbildung 4.5:  
Parallele SystemeAbbildung 4.6: Rückkopplung (links)  
und offener Regelkreis (rechts)

§ 533 Betrachtet man dagegen zwei parallel geschaltete Systeme, vgl. Abb. 4.5, so ist die Eingangsgröße in beiden Systemen gleich. Jedes Teilsystem erzeugt die Ausgangsgröße mit seiner jeweiligen Übertragungsfunktion und beide Ausgangsgrößen werden addiert. Daher ergibt sich die Gesamtübertragungsfunktion als Summe der Übertragungsfunktionen der Einzelsysteme: aus

$$y = y_1 + y_2 = G_1 u_1 + G_2 u_2 \stackrel{u_1 = u_2 = u}{=} (G_1 + G_2) u$$

folgt unmittelbar

$$G(s) = G_1(s) + G_2(s). \quad (4.16)$$

In diesem Fall sind die Pole in  $G$  auch Pole in  $G_1$  oder  $G_2$ , die Nullstellen bleiben in der Regel jedoch nicht erhalten.

§ 534 Die Serien- und Parallelschaltung von Systemen ist eine Hilfe zur Kombination von Systemen oder zur Zerlegung von Systemen in Untersysteme. Dabei werden die Systeme als unabhängig von einander arbeitend betrachtet. Interessanter ist eine Kombinationen von Systemen, die eine Rückkopplung erlaubt, vgl. linken Teil von Abb. 4.6. In diesem Fall wird ein Teil des Ausgangssignals des ersten Systems (Übertragungsfunktion  $G_1$ ) im zweiten System (Übertragungsfunktion  $G_2$ ) verarbeitet und erneut auf den Eingang des ersten Systems gegeben. Das gesamte Eingangssignal ist proportional dem Eingangssignal plus dem Ausgang des zweiten Systems, also  $\sim 1 + G_1(s)G_2(s)$ . Dieses Signal wird vom ersten System verarbeitet, so dass die Übertragungsfunktion insgesamt gegeben ist als

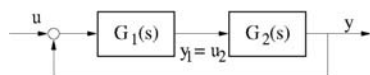
$$G(s) = \frac{G_1(s)}{1 + G_1(s)G_2(s)} = \frac{\mu_1 \nu_2}{\mu_1 \mu_2 + \nu_1 \nu_2}.$$

§ 535 An dieser Gleichung ist die zweite Darstellungsform mit Hilfe der Polynome die interessante: durch geschickte Wahl von  $G_2$  können die Nullstellen im Nennerpolynom und damit die Pole der Übertragungsfunktion gezielt beeinflusst werden. Dabei betrachten wir das zweite System als eine Art freien Parameter, der nur mit dem Ziel eingeführt wird, die Übertragungsfunktion  $G_1$  zu verändern.

§ 536 Als Grundlage für die Beschreibung derartiger Regelkreise führt man das Konzept des offenen Regelkreises ein, vgl. rechten Teil von Abb. 4.6. Der offene Regelkreis entsteht, in dem man die Rückkopplung auftrennt. Die Übertragungsfunktion des offenen Regelkreises entspricht dann der der beiden hinter einander geschalteten Systeme

$$L(s) = G_1(s)G_2(s).$$

Obwohl der offene Regelkreis etwas künstlich erscheint, kann mit Hilfe von  $L(s)$  das Verhalten eines Gesamtsystems häufig sehr gut beschrieben werden.

Abbildung 4.7: Standardregel-  
kreis

§ 537 Dies ist insbesondere beim Standardregelkreis, vgl. Abb. 4.7, der Fall. Hierbei wird die Rückkopplung einer Serienschaltung zweier Systeme mit den Übertragungsfunktionen  $G_1(s)$  und  $G_2(s)$  betrachtet. Die Gesamtübertragungsfunktion ist dann

$$G(s) = \frac{G_1(s)G_2(s)}{1 + G_1(s)G_2(s)} = \frac{L(s)}{1 + L(s)},$$

wobei im letzten Ausdruck die Übertragungsfunktion  $L(s)$  des offenen Regelkreises verwendet wurde.

### 4.3.3 Eigenfunktionen

§ 538 Eigenfunktionen sind spezielle (komplexe) Eingangsfunktionen eines LTI Systems wie durch (4.2) beschrieben. Die Eigenfunktionen bilden die Basen eines nicht-geometrischen Vektorraums, in dem sich beliebige Eingangsfunktionen darstellen lassen – die Fourier Reihe verwendet Sinus und Kosinus als Basen zur Darstellung beliebiger periodischer Funktionen. Eigenfunktionen eines Systems sind entsprechend den Eigenvektoren einer Matrix die Funktionen, die auf sich selbst abgebildet werden, gegebenenfalls skaliert mit einem Faktor, dem Eigenwert.

§ 539 Aus systemtheoretischen Gesichtspunkten ist die folgende Eigenschaft wichtiger: die Eigenfunktionen ermöglichen eine anschauliche Interpretation der Pole und Nullstellen der Übertragungsfunktion. Sie erlauben auch die anschauliche Deutung des Begriffes Frequenzgang.

§ 540 Die Gleichung der Resolvente erinnert in ihrer Form bereits an eine Eigenwertgleichung. Für die hier betrachteten Beispiele Tiefpass, erweitertes RC-Netzwerk und linearer Oszillator sind die Eigenfunktionen der Form

$$u(t) = Ue^{\xi t}.$$

Die skalaren Konstanten  $U$  und  $\xi$  können komplex sein. Damit erlaubt diese Gleichung auch die Beschreibung der für viele Anwendungen wichtigen harmonischen Funktionen (und natürlich der Exponentialfunktion bzw. der Konstanten).

§ 541 Beschränkt sich die Betrachtung eines Systems auf Eingangsgrößen, die Eigenfunktionen sind, so ist Übertragungsfunktion im Zeitbereich allgemein gültig, unabhängig von der speziellen Eingangsfunktion – wir genießen also die Vorteile der Darstellung von Übertragungsfunktionen im Bildbereich auch im Zeitbereich.

§ 542 Die allgemeine Lösung des Systems setzt sich wieder aus der Lösung der homogenen DGL und einer speziellen Lösung der inhomogenen DGL zusammen, wobei wir für letztere einen Ansatz wählen, der der Eingangsfunktion möglichst ähnlich ist. Als allgemeine Lösung des inhomogenen Systems im Zeitbereich erhalten wir damit

$$y(t) = \vec{c}e^{At}\eta + \left[ \vec{c}(\xi\mathbf{E} - \mathbf{A})^{-1}\vec{b} + d \right] Ue^{\xi t}.$$

Die eckige Klammer gibt den Wert der Übertragungsfunktion an Stelle  $\xi$  – dies ist die Eigenschaft, die es erlaubt, für Eigenfunktionen eine allgemeine Übertragungsfunktion  $G(s) = \vec{c}(s\mathbf{E} - \mathbf{A})^{-1}\vec{b} + d$  auch im Zeit- und nicht nur im Bildbereich zu erhalten.

§ 543 Berücksichtigen wir den Anfangswert, so erhalten wir für die Lösung der Systemgleichung

$$\vec{x} = e^{A(t-t_0)} \left[ \vec{x}_0 - (\xi\mathbf{E} - \mathbf{A})^{-1}\vec{b}Ue^{\xi t_0} \right] + (\xi\mathbf{E} - \mathbf{A})^{-1}\vec{b}Ue^{\xi t}$$

und damit für die Ausgangsgröße

$$y = \vec{c}e^{A(t-t_0)} \left[ \vec{x}_0 - (\xi\mathbf{E} - \mathbf{A})^{-1}\vec{b}Ue^{\xi t_0} \right] + \left[ \vec{c}(\xi\mathbf{E} - \mathbf{A})^{-1}\vec{b} + d \right] Ue^{\xi t}.$$

§ 544 Da die allgemeine Übertragungsfunktion  $G(s) = \vec{c}(s\mathbf{E} - \mathbf{A})^{-1}\vec{b} + d$  an der Stelle  $\xi$  die Ausgangsgröße

$$y(t) = \vec{c}e^{At}\eta + G(\xi)Ue^{\xi t}$$

erzeugt, kann man für eine geeignete Wahl des Anfangszustand  $\vec{x}_0$  in der Form

$$\vec{x}_0 = (\xi\mathbf{E} - \mathbf{A})^{-1}\vec{b}Ue^{\xi t_0}$$

den Zusammenhang zwischen Aus- und Eingangsgröße schreiben als

$$y(t) = G(\xi)Ue^{\xi t},$$

d.h. für die Eigenfunktion wird die Übertragungsfunktion auch im Zeitbereich der Quotient aus Ausgangs- und Eingangsgröße.

§ 545 Betrachten wir nun Nullstellen der Übertragungsfunktion, d.h. die Stellen, an denen das Zählerpolynom der Übertragungsfunktion verschwindet. An einer Nullstelle  $\beta$  ergibt sich für die Ausgangsgröße

$$y(t) = \vec{c}e^{At}\eta + G(\beta)Ue^{\beta t} = \vec{c}e^{At}\eta + G(\beta).$$

In dieser Beziehung tritt die Eingangsgröße  $u(t)$  nicht mehr auf, d.h. an den Nullstellen der Übertragungsfunktion ist das Zeitverhalten nur durch die Transitionsmatrix bestimmt und unabhängig von der Eingangsgröße. Wählen wir als Anfangszustand

$$\vec{x}_0 = (\beta\mathbf{E} - \mathbf{A})^{-1}\vec{b}Ue^{\beta t_0},$$

so ist die Ausgangsgröße unabhängig von der Eingangsgröße stets Null!

§ 546 Zur Betrachtung des Verhaltens an den Polstellen wählen wir die Eingangsgröße  $u(t) = 0$ . Die Ausgangsgröße reduziert sich dann auf

$$y(t) = \vec{c}e^{A(t-t_0)}\vec{x}_0.$$

Wählen wir jetzt eine Anfangszustand  $\vec{x}_0$  auf einer Eigenrichtung des Systems. Dieser Anfangszustand entspricht einem Eigenvektor der Systemmatrix  $\mathbf{A}$  zum Eigenwert  $\lambda$ . In diesem Fall gilt  $\vec{x} = \vec{x}_0 e^{\lambda t}$  und die Ausgangsgröße wird

$$y(t) = \vec{c}\vec{x}_0 e^{\lambda(t-t_0)} = Y e^{\lambda t} :$$

bei verschwindender Eingangsgröße wird die Ausgangsgröße durch einen exponentiellen Verlauf beschrieben, die Zeitkonstante entspricht dem Pol der Übertragungsfunktion.

#### 4.3.4 Frequenzgang

§ 547 Der Frequenzgang ist die Übertragungsfunktion  $G(s)$  auf der imaginären Achse,  $G(s) = i\omega$ . Betrachten wir eine Systemmatrix, deren Eigenwerte alle einen negativen Realteil besitzen und beschränken uns auf Eingangsgrößen der Form  $u(t) = Ue^{i\omega t}$  mit  $U$  und  $\omega$  reell. Außerdem wollen wir das System im eingeschwungenen Zustand betrachten. Das stationäre Systemverhalten ist beschrieben durch die Gleichungen

$$\vec{x}(t) = (i\omega\mathbf{E} - \mathbf{A})^{-1}\vec{b}Ue^{i\omega t} \quad \text{und} \quad y(t) = G(i\omega)Ue^{i\omega t}.$$

§ 548 Verwenden wir als Eingangsgröße die harmonische Schwingung, so erhalten wir als Ausgangsgröße eine harmonische Schwingung mit einer Phasenverschiebung  $\varphi$ :

$$y(t) = |G(i\omega)|U \cos(\omega t + \varphi)$$

mit  $\varphi$  als Phase der komplexen Zahl. Die Ausgangsschwingung hat also die gleiche Frequenz, ist um  $\varphi$  phasenverschoben und hat die Amplitude  $|G(i\omega)|U$ . Diese Aussage ist einer verallgemeinerte Form dessen, was wir bereits als Übertragungsfunktion für den Tiefpass kennen gelernt haben.

§ 549 Für das RC-Netz hatten wir § 522 bereits die Übertragungsfunktion bestimmt zu

$$G(s) = \frac{4}{(s+1)(s+4)}.$$

Verwenden wir jetzt als Eingangsgröße  $u(t) = \sin \omega_0 t$ , so erhalten wir für die Ausgangsgröße im stationären Zustand

$$y(t) = |G(i\omega_0)| \sin(\omega_0 t + \arg\{|G(i\omega_0)|\}),$$

wie bereits in Abb. 4.3 und der zugehörigen Fußnote gegeben. Das Auftragen von Phase und Betrag von  $G(i\omega)$  liefert den Frequenzgang (Bode-Diagramm).

## Literaturhinweise

§ 550 Als einfache Einführungen in diesen Abschnitt eignen sich Bücher zur Schaltungs- und Regelungstechnik; ein Beispiel ist Horn und Dourdoumas [50], an dem ich mich in diesem Kapitel deutlich orientiert habe. Weitere Hinweise finden sich auch in Lutz und Wendt [87] und, allerdings mit starkem Bezug auf die Elektronik, Wupper [179].

## Fragen

**Frage 10** Erläutern Sie die Grundbegriffe zur Beschreibung eines Systems.

**Frage 11** Welche Rolle spielt die Laplace Transformation in der Systemtheorie? Erläutern Sie die Vorteile gegenüber anderen Lösungsverfahren.

**Frage 12** Erläutern Sie die Bedeutung der Transitionsmatrix.

**Frage 13** Was versteht man unter einer Übertragungsfunktion? Ist die Übertragungsfunktion im Zeitbereich eine universelle Konstante des Systems?

**Frage 14** Erläutern Sie die Grundideen der Rückkopplung. Diskutieren Sie die Stabilität.

**Frage 15** Wie verhalten sich die Übertragungsfunktionen zweier Systeme bei Serien- und Parallelschaltung? Begründen Sie (Skizze).

## Aufgaben

**Aufgabe 11** Stellen Sie die Systemgleichung für einen RC-Hochpass auf und lösen Sie sie (a) im Zeitbereich und (b) mit Hilfe einer Laplace Transformation für eine harmonische Eingangsgröße und für eine Sprungfunktion.

**Aufgabe 12** Ein Tiefpass kann auch durch Reihenschaltung einer Spule und eines Widerstands realisiert werden, wobei die Spannung über dem Widerstand abgegriffen wird. Stellen Sie die Systemgleichungen auf und lösen Sie sie im Zeitbereich.

**Aufgabe 13** Ein Tiefpass zweiter Ordnung wird aus einer Serienschaltung aus Spule, Widerstand und Kondensator realisiert, die Ausgangsspannung wird über dem Kondensator abgegriffen (Serienschwingkreis). Stellen Sie die Systemgleichungen auf. Bestimmen Sie die Übertragungsfunktion. Diskutieren Sie auch deren Pol(e). Zeichnen Sie das zugehörige Bode-Diagramm.

**Aufgabe 14** Bestimmen Sie die Antwort des Tiefpass aus Aufg. 13 auf eine Sprungfunktion und auf einen  $\delta$ -Puls.



## SIMULINK-Spielereien

**SIMULINK-Aufgabe 15** Arbeiten Sie Hoch- und Tiefpass in SIMULINK ab – Abschn. 2.1 kann als Hilfe dienen.

**SIMULINK-Aufgabe 16** Konstruieren Sie Bandpass und Bandsperre in SIMULINK.

**SIMULINK-Aufgabe 17** Überlegen Sie, ob und wenn ja wie, Sie ein Bode-Diagramm mit Hilfe von SIMULINK erzeugen können.

**SIMULINK-Aufgabe 18** Nehmen Sie nochmals den Hoch- und/oder Tiefpass und spielen Sie mit den Einstellungen zur Simulationskontrolle (siehe Abschn. 2.2.5).

**SIMULINK-Aufgabe 19** Weitere Beispiele für Übertragungsfunktionen bestimmen und einbauen lassen (z.B. über die Tischkante hüpfendes Seil als zusätzliche antreibende Kraft bei Fedef)

# Kapitel 5

## Übertragungsfunktion vertieft: Filter

Where is the wisdom we have lost in knowledge? Where is the knowledge we have lost in information?

T.S. Eliot, *The rock*, 1934

§ 551 Filter sind einfache Beispiele für LTI-Systeme und damit zu einer genaueren Betrachtung der Übertragungsfunktion geeignet. Vom Gesichtspunkt eines Systems her sind Filter als statische Systeme jedoch langweilig – sie erfüllen zwar eine Aufgabe, entwickeln sich jedoch nicht weiter.

§ 552 Andererseits sind Filter (aufgebaut in analoger Form oder eingesetzt als digitale Filter) die wichtigsten Werkzeuge der Signalanalyse – und werden damit häufig auch in der Analyse des Verhaltens dynamischer Systeme eingesetzt. Analoge Filter enthalten in der Regel verzögernde Elemente, sie haben eine Art Gedächtnis: ein analoger Filter ist nicht symmetrisch in dem Sinne, dass eine Zeitumkehr möglich ist. Oder einfacher: ein analoger Filter ist ein kausaler Filter: er kennt nur die Vergangenheit. Digitale Filter dagegen sind als mathematische Konstrukte von dieser Einschränkung befreit; sie können, wie ein zentrales gleitendes Mittel, auch ‘zukünftige’ Zustände des Systems berücksichtigen.

§ 553 Analoge und digitale Filter unterscheiden sich in einem weiteren, für SIMULINK sogar noch wichtigeren Aspekt: analoge Filter sind zeit-kontinuierliche Systeme, digitale Filter dagegen diskrete Systeme. Beide können durch eine Übertragungsfunktion und unter Rückgriff auf die entsprechenden Transformationen beschrieben werden. Dafür benötigt man jedoch beim analogen Filter die Laplace Transformation und die zeit-kontinuierliche Übertragungsfunktion. Beim digitalen Filter werden diese ersetzt durch die  $z$ -Transformation und entsprechend die diskrete Übertragungsfunktion.

§ 554 In diesem Kapitel betrachten wir im wesentlichen FIR (finite impulse response) Filter: in diesen Systemen ist jedes Element des Ausgangssignal die Summe einer endlichen Zahl gewichteter Elemente des Eingangssignals. Das einfachste Beispiel ist das gleitende Mittel. FIR Filter können auf zeit-kontinuierliche ebenso wie auf diskrete Signale angewendet werden: während die letzteren formal durch eine Summation beschrieben werden, müssen erstere mit Hilfe von Integralen dargestellt werden.

§ 555 Filter in SIMULINK sind gleich bedeutend mit einer intensiven Nutzung der Übertragungsfunktion. Die Koeffizienten von Zähler- und Nennerpolynom können dabei extern bestimmt und eingegeben werden. Alternativ können wir für viele gängige Filter auch auf die MATLAB-Funktion `filter` für eindimensionale bzw. `filter2` für zweidimensionale Datensätze. Auch SIMULINK hält in der Blockgruppe `Discrete` verschiedene Formen von Filtern vor und insbesondere eine Übertragungsfunktion für diskrete Signale.

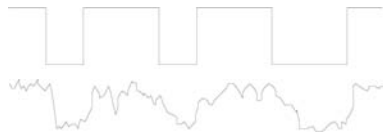


Abbildung 5.1: Analoge Repräsentation (*unten*) eines digitalen Signals (*oben*)

**Lernziele:** nach Durcharbeiten dieses Kapitels sollten Sie in der Lage sein,

- die grundlegenden Unterschiede zwischen analogen und digitalen Filtern darzustellen,
- den Begriff des FIR-Filters zu erläutern,
- einige grundlegende Filter in ihrer Funktion (und einer möglichen physikalischen Realisierung) zu beschreiben.
- die Übertragungsfunktion für einfache Fälle aus der Differentialgleichung herzuleiten,
- diskrete Systeme in SIMULINK zu simulieren,
- Datensätze (Zeitreihen ebenso wie Bild- oder Audiodateien in gängigen Formaten) in MATLAB bzw. SIMULINK einzulesen und weiter zu bearbeiten.

## 5.1 Filter

**Definition 7** Ein Filter ist ein speziell angepasstes System, dass aus einem Signal eine bestimmte Komponente entfernen soll oder die Charakteristik eines Signals modifizieren soll.

§ 556 Informationstheoretisch ist ein Signal die physikalische Repräsentation einer Nachricht (z.B. [40, 42, 76, 143, 144]); das kann z.B. die Farbe einer Verkehrsampel sein, die Intensität oder Frequenz einer Licht- oder Schallquelle oder der zeitliche Verlauf einer Spannung. Diese Kopplung zwischen Information und Signal als physikalischer Repräsentation der Information hat eine für einen Physiker auf den ersten Blick befremdliche, dem KI-ler oder Roboterbauer dagegen wohl vertraute Konsequenz: kleine Variationen eines Signals ändern nicht notwendigerweise auch dessen Bedeutung. Eine rote Ampel werden wir auch bei von der DIN abweichendem RGB-Wert als solche erkennen; die akustischen Signale der Sprache werden gleichermaßen verstanden, wenn ein Kind in heller Stimmlage oder ein Erwachsener in einer ganz anderen Stimmlage spricht. Auch Handschriften ergeben völlig unterschiedliche Signale, selbst wenn sie denselben Text darstellen.

**Verständnisfrage 52** Ist diese Unterscheidung zwischen Information und Signal gerechtfertigt? Welche Interpretation des Begriffes Information steckt dahinter?

§ 557 Bei genauerem Hinsehen ist uns dieses Übersehen von Variationen aber auch in elektronischen Systemen (und insbesondere in unserem Rechner) geläufig: ein digitales Signal wird durch zwei Spannungspegel, High oder Low, gebildet. Die realen Spannungen, z.B. das bei der NASA auf gefangene Signal einer Raumsonde, dagegen bilden diese beiden Pegel nur recht ungenau ab (vgl. Abb. 5.1). Aber selbst ohne einen langen und gestörten Übertragungsweg werden die Pegel durch Verluste in den einzelnen Gattern nicht punktgenau weiter gegeben. Daher wird z.B. in der TTL-Logik jedes Signal unterhalb 0.8 V in eine Signalklasse 0 oder Low eingeordnet; jedes Signal oberhalb 2 V dagegen wird in die Signalklasse 1 oder High eingeordnet.<sup>1</sup>

§ 558 Signalverarbeitung bedeutet letztendlich, Information aus einer Zeichenkette zu extrahieren. Dabei kann die Zielsetzung von ‘extrahiere alle im Signal enthaltene Information’ bis hin zu ‘Überprüfe, ob und in welchen Anteilen, im Signal eine bestimmte Frequenz enthalten ist’. Die erste Frage würden wir eher einem unbekanntem System stellen, um möglichst viel über dieses System zu erfahren. In diesem Fall sollte die Datenaufbereitung ohne Filterung

<sup>1</sup>Eine derartige Signalklasse bezeichnet man auch als Zeichen. Weitere Beispiele für Zeichen sind die Buchstaben des Alphabets, die Ziffern und die Satzzeichen. Diese Zeichen lassen sich ihrerseits wieder zu Zeichenketten (Hyperzeichen) kombinieren, so dass eine Zeichenhierarchie entsteht; in einem Text wäre diese Hierarchie gegeben durch Buchstaben – Wörter – Sätze – Paragraphen usw.

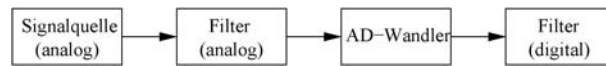
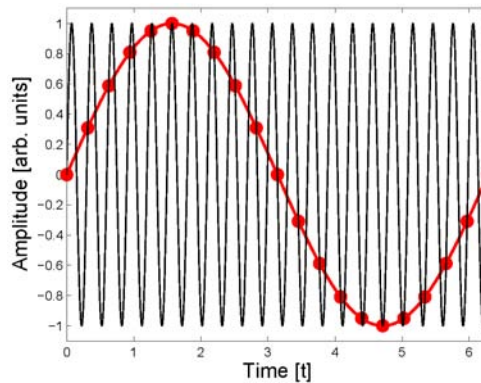


Abbildung 5.2: Signalkette: das analoge Signal wird mit einem analogen Filter für die Digitalisierung im AD-Wandler aufbereitet und anschließend mit digitalen Filtern weiter aufbereitet oder analysiert

Abbildung 5.3: Unterabtastung: ist die Abtastfrequenz (rote Punkte) deutlich kleiner als die Frequenz des Signals (schwarze Kurve), so kann fälschlicherweise eine viel zu geringe Frequenz rekonstruiert werden



erfolgen; bei der Analyse können selbstverständlich Filter eingesetzt werden. Die zweite Frage dagegen stellen wir eher an ein System, dessen allgemeines Verhalten wir gut verstanden haben, und von dem wir ein spezielles Detail näher untersuchen wollen. Hier können bereits früh Filter verwendet werden, um die nicht interessierende Information (z.B. bestimmte Frequenzbereiche) zu entfernen. Damit sinkt der Informationsgehalt im Signal, so dass es (a) leichter zu bearbeiten und (b) knapper zu codieren ist – und damit auch schneller zu übertragen und kompakter zu speichern.

§ 559 Da die eigentliche Informationsextraktion auch aus einem analogen Signal in der Regel digital erfolgt, werden in einer Signalverarbeitungskette Filter an zwei Stellen angewendet, vgl. Abb. 5.2. Das analoge Signal wird vor dem AD-Wandler gefiltert, um insbesondere die hochfrequenten, auf Grund der endlichen Sampling Rate des AD-Wandlers ohnehin nicht mehr erfassbaren Anteile zu entfernen. Hierfür verwendet man analoge Filter (Abschn. 5.2) zur Bandbegrenzung.

§ 560 Die nicht-Verwertbarkeit der hochfrequenten Anteile durch den AD-Wandler ist eigentlich nicht das korrekte Argument für die Verwendung des Tiefpass. In Abschn. 3.2.4 haben wir die Verbreiterung der Frequenzpeaks und das Auftreten nicht vorhandener hochfrequenten Anteile durch das Vervielfachen des Signals erwähnt. Ein Fehlen des Tiefpass würde zusätzlich niederfrequente Anteile in das Frequenzspektrum einführen – nicht bedingt durch die Eigenschaften der FFT sondern bedingt durch die endliche Abtastrate des AD-Wandlers. Ist die Frequenz des Signals viel größer als die Abtastfrequenz, so dass maximal einmal während einer Periode des Signals abgetastet wird (siehe auch Abb. 5.3), so kann auf Grund der Unterabtastung gegebenenfalls ein nicht vorhandenes niederfrequentes Signal erzeugt werden. Um derartige Störeinflüsse abzuschneiden ohne gleichzeitig bereits die noch vom AD-Wandler korrekt erfassbaren Frequenzen zu dämpfen, wird ein Tiefpass mit steiler Flanke benötigt, z.B. der Butterworth- oder Tschebyscheff-Filter.

§ 561 Hinter dem AD-Wandler liegt das Signal in digitaler Form vor, d.h. hier werden digitale Filter verwendet. Diese können ebenso wie analoge Filter im Zeitbereich eingesetzt werden, ebenso jedoch auch im Orts- oder Frequenzbereich.

**Verständnisfrage 53** Und analoge Filter? Gibt es dort im Ortsbereich Möglichkeiten? Und im Frequenzbereich?

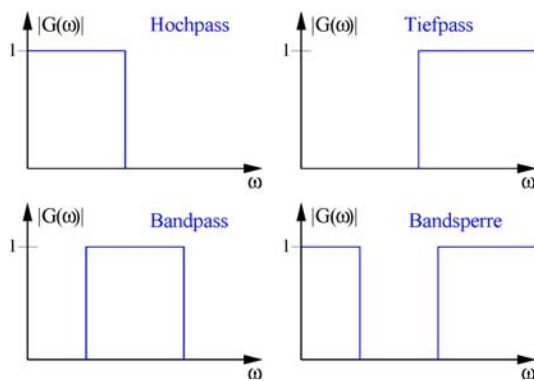


Abbildung 5.4: Beträge idealer Übertragungsfunktionen von vier typischen Filtern

§ 562 Wir werden in diesem Kapitel einige häufig verwendete Filter wie Tief-, Band- und Hochpässe verschiedener Ordnung kennen lernen. Insbesondere bei den Pässen lässt sich die Funktion ebenso wie die Bestimmung der Übertragungsfunktion auf einfache Weise aus den analogen Aufbauten herleiten. Darauf aufbauend lässt sich die allgemeine Struktur einer Übertragungsfunktion auch für komplexere Filter herleiten.

§ 563 Die Idee dazu haben wir im Vorbeigehen schon in § 46 kennen gelernt: haben wir eine den Filter beschreibende DGL so ist eine Laplace Transformation ausreichend, um seine Übertragungsfunktion zu bestimmen. Damit erklärt sich auch die Ordnung eines Filters: diese ist bestimmt durch die Ordnung der ihn beschreibenden DGL und gibt damit auch die höchste Potenz mit der die komplexe Frequenz  $s$  in der Übertragungsfunktion auftritt. LTI Systeme können mit linearen DGLs mit konstanten Koeffizienten  $A$  und  $B$  beschrieben werden:

$$\left[ A_n \frac{d^n}{dt^n} + \dots + A_1 \frac{d}{dt} + A_0 \right] u_{\text{out}} = \left[ B_n \frac{d^n}{dt^n} + \dots + B_1 \frac{d}{dt} + B_0 \right] u_{\text{in}} .$$

Da im Frequenzbereich jede Differentiation ein  $i\omega$  (Fourier) bzw. ein  $s$  (Laplace) als Faktor trägt, ergibt sich eine Übertragungsfunktion

$$G(s) = \frac{U_{\text{out}}(s)}{U_{\text{in}}(s)} = \frac{\sum_{k=0}^m B_k s^k}{\sum_{k=0}^n A_k s^k}$$

wie bereits in § 528 in anderem Zusammenhang gegeben.

## 5.2 Analoge Filter im Zeitbereich

§ 564 Signalfilterung in der Analogtechnik hat vielfältige Anwendungen, wie z.B. Rauschunterdrückung, Bandbreitenbegrenzung, Eliminierung störender Frequenzen, Glättung von Messwerten oder das Hervorheben von Trends im Signal.<sup>2</sup> Auch wenn die Anwendung vielfältig sind, sind die Verfahren für gängige Filter einfach. Letztendlich ist die Analogfilterung eine frequenzabhängige Bewertung von Amplitude und Phase des Signals. Diese wird durch die Übertragungsfunktion dargestellt, vgl. unteres Teilbild in Abb. 4.3. Die Wirkung eines analogen Filters lässt sich daher in der spektralen Darstellung des zu filternden Signals am einfachsten beschreiben.

§ 565 Die Ordnung der Übertragungsfunktion und damit die des Filters hängt von der Ordnung der den Filter beschreibenden Differentialgleichung ab. Mit zunehmender Ordnung lassen sich immer komplexere Übertragungsfunktionen und damit auch immer komplexere Manipulation des Eingangssignals realisieren.

<sup>2</sup>Diese Anwendungen sind natürlich nicht auf analoge Filter beschränkt. Auch digitale Filter können diese Aufgaben erfüllen.

§ 566 So werden der einfache Hoch- und Tiefpass jeweils durch DGLs erster Ordnung beschrieben. Damit ist ihr Funktionsumfang eingeschränkt: in einem gewissen Frequenzbereich ist ihre Verstärkung idealerweise 1, oberhalb beim Tiefpass bzw. unterhalb beim Hochpass einer Grenzfrequenz sollte die Verstärkung möglichst schnell gegen Null gehen. Das Bode-Diagramm in Abb. 4.3 zeigt, dass der Frequenzgang eines als Filter 1. Ordnung realisierten Tief- bzw. Hochpass weit vom Idealbild entfernt ist.

§ 567 Geschickte Kombination von Hoch- und Tief liefert einen Bandpass. Realisiert man diesen nicht mit zwei Filtern erster Ordnung sondern einem Filter zweiter Ordnung (Schwingkreis), so kann auf Grund der auftretenden Resonanz im Durchlassbereich eine Verstärkung größer 1 auftreten – die Filterfunktion ist nicht auf die Unterdrückung einzelner Frequenzen beschränkt sondern es können Frequenzen selektiv verstärkt werden.

**Verständnisfrage 54** Wie sieht es dabei mit der Energiebilanz aus? Wird die Energie innerhalb des Signals von einem Frequenzbereich in einen anderen verschoben oder von außen zugeführt? Oder anders formuliert: in der Resonanz kann der Strom in einem Bauteil des Schwingkreises größer werden als der von der Quelle aufgenommene Strom (eben dann, wenn die Verstärkung größer eins wird). Ist das physikalisch sinnvoll?

§ 568 Aber selbst der konventionelle Hoch- oder Tiefpass kann durch ein Filter höherer Ordnung näher an den Verlauf eines idealen Tiefpass gebracht werden – so wird beim Butterworth Filter der Übergang zwischen Transmission und Sperrbereich steiler, so dass dieser z.B. vor dem AD-Wandler wesentlich effizienter die höherfrequenten Anteile entfernt als es ein konventioneller Tiefpass erster Ordnung machen würde.

§ 569 Die einen Filter für den praktischen Einsatz charakterisierenden Größen sind die Grenzfrequenz  $\omega_g$ , die Sperrfrequenz  $\omega_s$  sowie die Parameter  $\varepsilon$  und  $\lambda$ , die bestimmen, wie stark die Amplitude bei diesen Frequenzen reduziert ist. Beim Tief bzw. Hochpass 1. Ordnung ist z.B. die Amplitude des Eingangssignals bei der Grenzfrequenz auf  $1/\sqrt{2}$  abgesunken, d.h. der Betrag der Übertragungsfunktion ist  $|G(\omega)| = 1/\sqrt{2}$ . Allgemeiner definiert man die Änderung der Amplitude bei der Grenzfrequenz mit Hilfe des Betragsquadrats der Übertragungsfunktion und eines Parameters  $\varepsilon$  als

$$|G(\omega_g)|^2 = \frac{1}{1 + \varepsilon^2}.$$

Beim Tiefpass wird der Durchlassbereich definiert für  $\omega < \omega_g$  und damit  $|G(\omega_g)|^2 = 1/(1+\varepsilon^2)$ .

§ 570 Entsprechend wird ein Sperrbereich definiert über eine Sperrfrequenz  $\omega_s$ . Im Sperrbereich ist das Betragsquadrat der Amplitudenfunktion unter einen bestimmten Wert gesunken, definiert mit Hilfe eines Parameters  $\lambda$ :

$$|G(\omega_s)|^2 = \frac{1}{1 + \lambda^2}.$$

§ 571 Ein idealer Tiefpass wie in Abb. 5.4 ergibt sich für  $\varepsilon = 0$  und  $\lambda \rightarrow \infty$ .

**Verständnisfrage 55** Und der ideale Hochpass?

### 5.2.1 Filter 1. Ordnung

§ 572 Hoch- und Tiefpass erster Ordnung leiten sich von der gleichen elementaren Serienschaltung aus Widerstand und Kondensator ab; lediglich wird beim Tiefpass der Spannungsabfall über dem Kondensator, beim Hochpass dagegen der über dem Widerstand betrachtet. Die Summe aus beiden Spannungsabfällen entspricht der Eingangsspannung. Damit ist die Summe aus dem Hoch- und dem Tiefpass gefilterten Signal gleich dem Eingangssignal: Hoch- und Tiefpass sind komplementär zu einander.

§ 573 Die DGL für den Tiefpass wie im linken Teil von Abb. 4.1 lässt sich auf der Basis der Maschenregel schreiben als

$$u = u_R + u_C = iR + u_C = CR\dot{u}_C + u_C .$$

Laplace Transformation liefert (wieder unter Vernachlässigung der Anfangswerte)

$$U(s) = (sCR + 1)U_C(s) \quad \Rightarrow \quad U_C(s) = \frac{1}{RCs + 1} U(s)$$

und damit die für SIMULINK brauchbaren Übertragungsfunktionen

$$G_{\text{TP}}(s) = \frac{1}{RCs + 1} .$$

Alternativ liefert die Fourier Transformation  $U(\omega) = (i\omega CR + 1)U_C(\omega)$  und damit die vielleicht bekannteren Übertragungsfunktionen

$$G(\omega) = \frac{1}{1 + i\omega CR} = \frac{1}{1 + i\omega/\omega_g}$$

mit der Grenzfrequenz  $\omega_g = (RC)^{-1}$  und dem Bode Diagramm wie im unteren Teil von Abb. 4.3.

**Verständnisfrage 56** Die aus der Fourier Transformation bestimmte Übertragungsfunktion ist identisch mit der in § 499ff für ein harmonisches Eingangssignal gefundenen. Gilt die aus der Fourier Transformation bestimmte Übertragungsfunktion nun für allgemeine Eingangsgrößen oder doch wieder nur für harmonische?

**Verständnisfrage 57** Bei der Laplace Transformation zur Herleitung der Übertragungsfunktion haben wir den Anfangswert vernachlässigt – machen wir damit eine unzulässige Vereinfachung oder nicht?

**Zwischenrechnung 38** Wenn Sie mit Verständnisfrage 57 Schwierigkeiten haben oder Ihre Antwort mit Hilfe einer Simulation überprüfen wollen: im Library Browser finden sich unter Simulink Extras: Additional Linear auch Blöcke für Übertragungsfunktionen und den State Space, die eine Angabe von Anfangswerten erlauben.

§ 574 Da beide komplementär sind, ergibt sich die Übertragungsfunktion des Hochpass direkt aus der des Tiefpass. Wegen  $u = u_R + u_C$  ist  $u_r = u - u_C$  und damit nach Laplace Transformation

$$U_R(s) = U(s) - U_C(s) = U(s) - \frac{1}{RCs + 1}U(s) = \frac{RCs}{RCs + 1}U(s) .$$

Daraus ergibt sich unmittelbar die Übertragungsfunktion

$$G_{\text{HP}}(s) = \frac{RCs}{RCs + 1} .$$

Entsprechend lässt sich die Übertragungsfunktion in der imaginären Frequenz wieder durch Fourier Transformation oder durch Differenzbildung mit der des Tiefpass bestimmen:

$$G_{\text{HP}}(\omega) = \frac{1}{1 + \frac{1}{i\omega RC}} .$$

§ 575 Hoch- und Tiefpass werden beide durch Differentialgleichungen erster Ordnung beschrieben, ihre Übertragungsfunktionen sind daher Brüche aus Polynomen maximal erster Ordnung.

§ 576 Aus der Abb. 5.5 lässt sich die Auswirkung der Filter auf ein Signal beschreiben. Das Eingangssignal (cyan) besteht aus einer schnellen Fluktuation, die sich auf einen langsamen Trend aufsetzt:  $u(t) = a_1 \sin(\omega_1 t) + a_2 \sin(\omega_2 t)$  mit  $a_1 = 1$ ,  $a_2 = 0.2$ ,  $\omega_1 = 1$  und  $\omega_2 = 20$ . Die Grenzfrequenzen der beiden Pässe sind  $\omega_{g,\text{TP}} = 2$  und  $\omega_{g,\text{HP}} = 10$ . Der Tiefpass filtert die schnellen Fluktuationen und lässt den Trend (gelb) übrig, der Hochpass dagegen entfernt den Trend und lässt die schnellen Fluktuationen übrig (magenta).

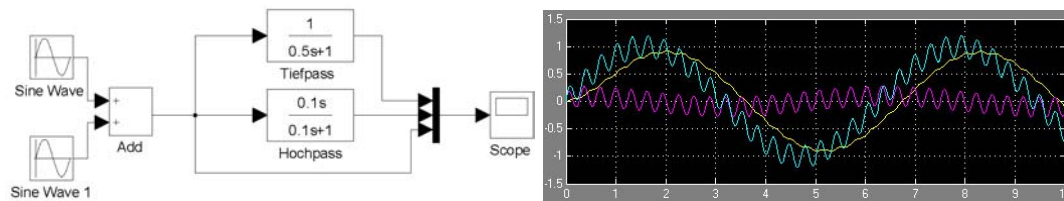


Abbildung 5.5: Hoch- und Tiefpass: aus dem Eingangssignal (cyan) filtert der Tiefpass die schnellen Fluktuationen heraus (gelb), der Hochpass dagegen lässt die Fluktuationen passieren und filtert den langsamen Trend heraus

§ 577 Damit lässt sich die Anwendbarkeit von Hoch- und Tiefpass auch von den klassischen Anwendungen auf elektrische Signale abstrahieren: ein Tiefpass glättet während ein Hochpass einen Trend entfernt. Eine typische nicht elektrische Anwendung wäre ein Datensatz mit den Tagestemperaturen (Mittelwert, Minimum, oder Maximum) über einen Zeitraum von etlichen Jahrzehnten. Hier sind kurzzeitige Fluktuationen (Witterung) einem mittelfristigen Zyklus (Jahresgang) überlagert, der sich wiederum auf einen noch längerfristigen Trend (Klimaveränderung) aufsetzt. Ein Tiefpass geeigneter Frequenz entfernt den Jahresgang, es verbleibt der Trend. Aus diesem lässt sich die Klimaänderung abschätzen. Ein Hochpass geeigneter Frequenz entfernt diesen Trend und es verbleiben die Jahresgänge, aus denen sich dann einfacher abschätzen lässt, ob sich die Amplitude im Jahresgang ändert, d.h. ob die Jahreszeiten ausgeprägter oder weniger ausgeprägt werden. Und Trend plus Fluktuationen ergibt das Gesamtsignal, wie auch von Hoch- und Tiefpass als komplementären Filtern erwartet.

§ 578 Auch wenn eine derartige Anwendung natürlich nicht auf die technische Realisierung aus Widerstand und Kondensator zurück greift, werden die Bezeichnungen Tief- und Hochpass für diese Form der Glättung (Tiefpass) bzw. der Entfernung eines Trends (Hochpass) auch hier verwendet. Auch wenn die Begriffe verwendet werden, so besteht doch ein wesentlicher Unterschied: bei der Glättung bzw. bei der Entfernung eines Trends ist die mit dem konventionellen Hoch- oder Tiefpass verbundene Phasenverschiebung ein unerwünschter Effekt.

§ 579 Der Bandpass als Serienschaltung aus einem Tiefpass mit  $R_T$  und  $C_T$  und einem Hochpass mit  $R_H$  und  $C_H$  wird wie in Abb. 4.4 angedeutet durch eine Übertragungsfunktion beschrieben, die sich als Produkt der Übertragungsfunktion von Hoch- und Tiefpass darstellen lässt:

$$G_{BP}(s) = G_{HP}(s) G_{TP}(s) = \frac{1}{R_T C_T s + 1} \frac{R_H C_H s}{R_H C_H s + 1} .$$

**Verständnisfrage 58** Ist der Bandpass nun ein Filter erster oder zweiter Ordnung? Argumentieren Sie physikalisch und formal.

## 5.2.2 Filter 2. Ordnung

§ 580 Als einfachsten Filter zweiter Ordnung können wir den Serienschwingkreis aus Spule, Widerstand und Kondensator verwenden, vgl. Abb. 4.2. Aus der Maschenregel ergibt sich

$$u_{in} = u_L + u_C + u_R \quad \text{und} \quad u_{in} = LC\ddot{u}_C + RC\dot{u}_C + u_C ,$$

Hierbei ist als Ausgangssignal der Spannungsabfall über dem Kondensator gewählt.

**Verständnisfrage 59** Was würde sich ergeben, wenn statt des Spannungsabfalls über dem Kondensator der über dem Widerstand oder über der Spule als Ausgangsgröße verwendet wird? Begründen Sie physikalisch und formal.



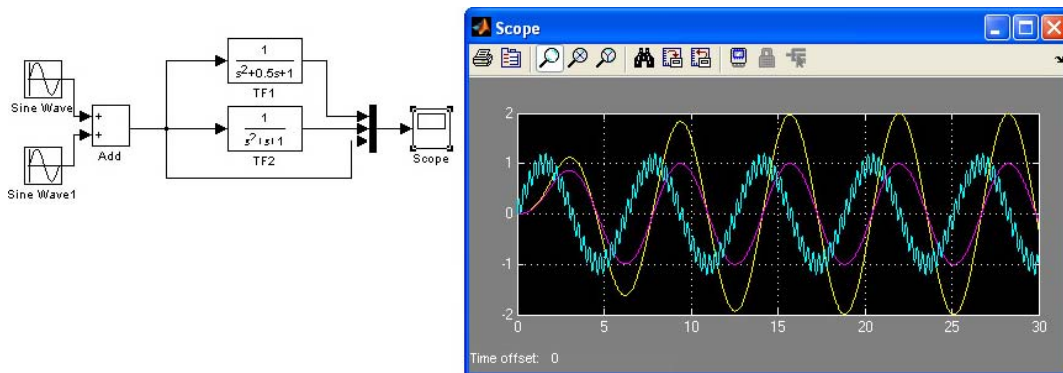


Abbildung 5.6: Serienschwingkreis als Bandpass: die tiefere Frequenz kann passieren, so dass die Kurve (cyan, wie in Abb. 5.5 erzeugt) geglättet wird. Der Betrag der Übertragungsfunktion kann größer 1 werden (gelb, Resonanz), muss aber nicht (magenta)

§ 581 Laplace Transformation der Differentialgleichung liefert

$$U_{\text{in}}(s) = s^2 L C U_C(s) + s R C U_C(s) + U_C(s) = (L C s^2 + R C s + 1) U_C(s) .$$

Damit ergibt sich die Übertragungsfunktion in der komplexen Frequenz  $s$  (und damit in der SIMULINK kompatiblen Form) zu

$$G(s) = \frac{U_C(s)}{U_{\text{in}}(s)} = \frac{1}{L C s^2 + R C s + 1} .$$

§ 582 Aus der Übertragungsfunktion in der komplexen Frequenz  $s$  lässt sich die Form der allgemeinen Übertragungsfunktion eines Tiefpass zweiter Ordnung ablesen als

$$G(s)_{\text{TP2}} = \frac{1}{a(s/\omega_g)^2 + bs/\omega_g} + 1 .$$

§ 583 Für die gebräuchlichere Darstellung der Übertragungsfunktion im Frequenzbereich liefert die Fourier Transformation

$$U_{\text{in}}(\omega) = (i\omega)^2 L C U_C(\omega) + i\omega R C U_C(\omega) + U_C(\omega) = (-\omega^2 L C + R C i\omega + 1) U_C(\omega) .$$

ebenfalls eine Übertragungsfunktion zweiter Ordnung

$$G(\omega) = \frac{U_C(\omega)}{U_{\text{in}}(\omega)} = \frac{1}{-\omega^2 L C + R C i\omega + 1} .$$

§ 584 Der Serienschwingkreis wirkt als Bandpass, d.h. es lässt sich ein Frequenzbereich auswählen. Wie bei Hoch- und Tiefpass erfolgt eine Phasenverschiebung des Signals. Da der Schwingkreis bei entsprechender Kombination von Eingangs- und Eigenfrequenz in Resonanz gerät, kann der Betrag der Übertragungsfunktion größer 1 werden. Die gelbe Kurve in Abb. 5.6 illustriert dies. Am Anfang ist das Einschwingen zu erkennen, nach zwei Schwingungen hat das geglättete Ausgangssignal konstante Amplitude.

**Verständnisfrage 60** Der Serienschwingkreis ist hier als Bandpass bezeichnet. In einigen anderen Darstellungen wird er in gleicher Schaltungsvariante und wieder mit dem Spannungsabfall über dem Kondensator als Ausgangssignal als Tiefpass bezeichnet. Was ist korrekt? Oder gibt es Bedingungen, in denen die eine Bezeichnung passender ist als die andere?

§ 585 Und wie sieht es in einem realen Serienschwingkreis aus? Kann der Spannungsabfall über einem Bauteil, in diesem Fall dem Kondensator, überhaupt größer werden als die anliegende Spannung? Das SIMULINK Ergebnis ist zumindest in Übereinstimmung mit der in Abb. 5.7 gegebenen Simulation in LTSpice (letztere für etwas andere Parameter). Und

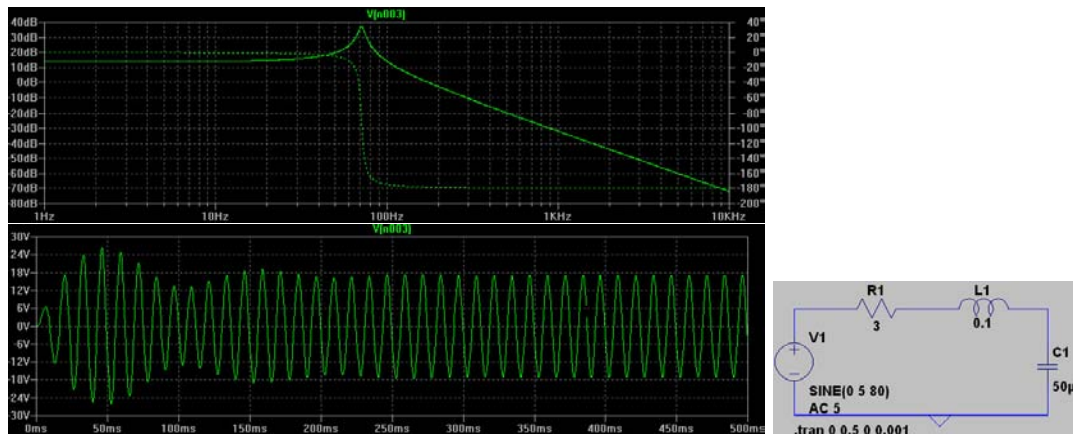


Abbildung 5.7: Serienschwingkreis in LTSpice: Übertragungsfunktion (*oben*), Spannungsabfall über dem Kondensator als Ausgangssignal (*unten links*) und Schaltbild mit Bauteilparametern (*unten rechts*)

größer als das Eingangssignal werden kann der Spannungsabfall über einem Bauteil ohne mit der Maschenregel (und damit der Energieerhaltung) in Konflikt zu geraten: die Summe der Spannungsabfälle über den einzelnen Bauteilen muss gleich der Eingangsspannung sein. Kondensator und Spule sind bei verschwindender Dämpfung jeweils mit einer Phasenverschiebung um  $\pi/2$  bzw.  $-\pi/2$  gegenüber der anliegenden Spannung verbunden. Damit ist aber der Spannungsabfall am Kondensator dem der Spule gegenüber um  $\pi$  verschoben: beide können bei verschwindender Dämpfung beliebig groß werden und sich trotzdem aufheben. Daher wird der Betrag der Übertragungsfunktion auch nicht durch irgendwelche anliegenden Spannungen begrenzt sondern lediglich durch die Dämpfung, in diesem Fall also den Widerstand.

§ 586 Ein Tiefpass zweiter Ordnung lässt sich auch durch eine Kettenschaltung zweier Tiefpässe wie im rechten Teil von Abb. 4.1 angedeutet erzeugen. In diesem Fall ergibt sich im Gegensatz zum Serienschwingkreis keine Resonanz sondern eine im Durchlassbereich flache Übertragungsfunktion, die jedoch steiler abknickt als beim Tiefpass erster Ordnung.

**Zwischenrechnung 39** Simulieren Sie das doppelte RC-Glied und vergleichen Sie die Ergebnisse mit denen eines geeigneten Serienschwingkreises.

### 5.2.3 Filter höherer Ordnung

§ 587 Häufige verwendete Filter höherer Ordnung sind der Butterworth und der Tschebyscheff Filter. Beide sorgen je nach Ordnung für mehr oder weniger steilere Flanken der Übertragungsfunktion und damit eine bessere Annäherung an die ideale Übertragungsfunktion aus Abb. 5.4 – der Tschebyscheff Filter ist die einfachere Realisierung, allerdings um den Preis einer geringen Welligkeit im Sperr- oder Durchlassbereich.

#### Butterworth Filter

§ 588 Der Butterworth Filter hat je nach Ordnung eine mehr oder weniger steile Flanke der Übertragungsfunktion. Dadurch ergibt sich eine scharfe Begrenzung des Sperr- und Durchlassbereichs sowie ein flacher Verlauf der Übertragungsfunktion in diesen Bereichen.

§ 589 Technisch lässt sich ein Butterworth Filter durch eine Kettenschaltung von Spulen und Kondensatoren (Cauer Topologie) realisieren, vgl. Abb. 5.8. Für diese Topologie sind

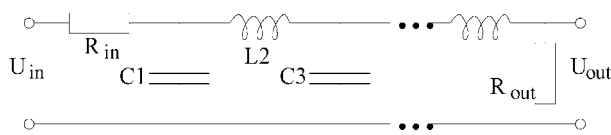


Abbildung 5.8: Ein Butterworth Filter  $n$ ter Ordnung lässt sich durch eine Kettenschaltung von  $n$  Tiefpässen erster Ordnung realisieren

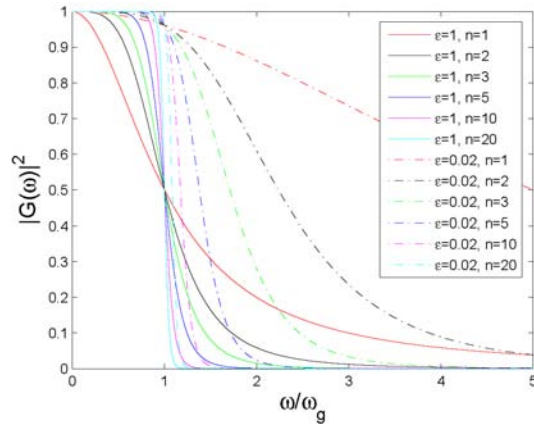


Abbildung 5.9: Betragssquadrat der Übertragungsfunktion von Butterworthfiltern verschiedener Ordnung  $n$

die Algorithmen zur Berechnung der Kapazitäten und Induktivitäten dokumentiert; für die Nummerierung wie in Abb. 5.8 ergibt sich

$$C_k = 2 \sin\left(\frac{2k-1}{2n}\pi\right) \quad \text{und} \quad L_k = 2 \sin\left(\frac{2k-1}{2n}\pi\right),$$

wobei die Indizes für die Kondensatoren ungeradzahlig sind, die für die Spulen dagegen geradzahlig.

§ 590 Für das Betragssquadrat der Übertragungsfunktion eines Butterworth Filters im Frequenzbereich gilt

$$|G(\omega)|^2 = \frac{1}{1 + \varepsilon^2 \left(\frac{\omega}{\omega_g}\right)^{2n}} \quad n \in \mathbb{N}.$$

Darin gibt  $n$  die Ordnung des Filters. In dieser Darstellung ist die Übertragungsfunktion alleine durch die Wahl von  $\varepsilon$  und der Grenzfrequenz festgelegt.

§ 591 Bezieht man auch die Sperrfrequenz explizit mit ein, so lässt sich die für diese Frequenzkombination erforderliche Ordnung des Butterworth Filters aus den Definitionen von Sperr- und Grenzfrequenz mit Hilfe der Parameter  $\varepsilon$  und  $\lambda$  bestimmen zu

$$n \geq \frac{\log\left(\frac{\lambda}{\varepsilon}\right)}{\log\left(\frac{\omega_s}{\omega_g}\right)}.$$

Bei der Auswahl der Ordnung des Filters muss ein Kompromiss zwischen Steilheit ( $\omega_g \rightarrow \omega_s$ ) und Aufwand gefunden werden. Beispiele für die Übertragungsfunktion für  $\varepsilon = 1$  und  $\varepsilon = 0.2$  sowie für verschiedene Ordnungen des Filters sind in Abb. 5.9 gegeben. Je höher die Ordnung ist, um so steiler wird die Flanke des Filters (und um so weniger wichtig ist die Wahl von  $\varepsilon$ ).

**Verständnisfrage 61** Ist der normale Tiefpass im Butterworth Filter enthalten und wenn ja, wo?

**Zwischenrechnung 40** Untersuchen Sie mit Hilfe von SIMULINK die Auswirkung verschiedener Butterworthfilter auf Überlagerungen harmonischer Eingangssignale und auf verrauschte Signale.

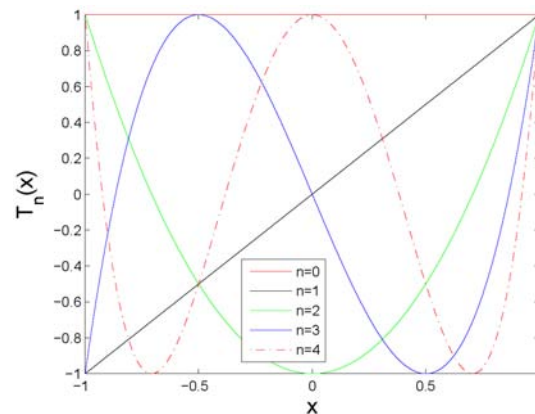


Abbildung 5.10: Tschebyscheff Polynome der Ordnungen 0 bis 4

**Zwischenrechnung 41** Untersuchen Sie mit Hilfe von SIMULINK die Wirkung verschiedener Butterworthfilter auf rechteckförmige Eingangssignale. Wie unterscheidet sich das Signal hinter dem Butterworthfilter von dem hinter einem Tiefpass gleicher Grenzfrequenz?

### Tschebyscheff Filter

§ 592 Der Tschebyscheff Filter erlaubt einen Kompromiss zwischen einer scharfen Trennung von Durchlass- und Sperrbereich und der Ordnung des Filters. Dafür muss jedoch eine geringe Welligkeit der Übertragungsfunktion im Durchlass- oder Sperrbereich zugelassen werden.

§ 593 Die Realisierung erfolgt wieder mit Hilfe einer Cauer Topologie wie in Abb. 5.8, jedoch mit einer anderen Dimensionierung der Bauteile.

§ 594 Die Übertragungsfunktion eines Tschebyscheff Filter wird durch die gleichnamigen Polynome angenähert. Diese zeichnen sich durch folgende Eigenschaft aus:

**Definition 8** Von allen Polynomen des Grades  $n$ , deren 0. Koeffizient stets den Wert 1 hat, besitzen die normierten Tschebyscheff Polynome  $T_x(n)/(2^{n-1})$  im Intervall  $[-1; 1]$  die kleinste maximale Amplitude.

Für alle  $n \geq 0$  sind diese gegeben als

$$T_n(x) = \begin{cases} \cos(n \arccos(x)) & \text{für } |x| \leq 1 \\ \cosh(n \operatorname{acrcosh}(x)) & \text{für } |x| > 1 \end{cases}$$

oder explizit<sup>3</sup> für  $|x| \leq 1$

$$\begin{aligned} T_0(x) &= \cos(0) = 1, \\ T_1(x) &= \cos(\arccos(x)) = x, \\ T_2(x) &= 2x^2 - 1, \\ T_3(x) &= 4x^3 - 3x, \\ T_4(x) &= 8x^4 - 8x^2 + 1, \\ &\vdots = \dots, \\ T_n(x) &= 2xT_{n-1}(x) - T_{n-2}(x). \end{aligned}$$

Der Verlauf dieser Polynome im Intervall  $[-1, 1]$  ist für die ersten fünf Ordnungen in Abb. 5.10 gegeben.

<sup>3</sup>Die explizite Darstellung lässt sich leicht bestimmen mit der Substitution  $x = \cos \alpha$  und anschließender Verwendung der Additionstheoreme des Kosinus für Vielfache des Winkels, z.B.  $\cos(2\alpha) = 2 \cos^2 \alpha - 1$  usw.

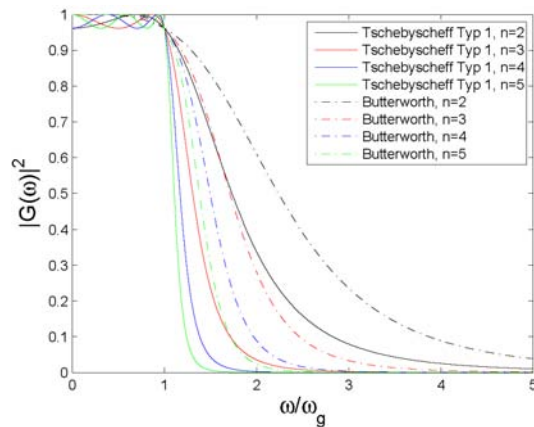


Abbildung 5.11: Betragsquadrate der Übertragungsfunktion für Tschebyscheff Filter vom Typ 1 (durchgezogene Linien) und Butterworth Filter (gestrichelte Linien) gleicher Ordnung für  $\varepsilon = 0.2$ .

§ 595 Den Tschebyscheff Filter gibt es in zwei Varianten: Typ 1 hat eine Welligkeit im Durchlassbereich, Typ 2 eine im Sperrbereich. Das Betragsquadrat der Übertragungsfunktion für den Tschebyscheff Filter vom Typ 1 ist

$$|G_{T1}(\omega)|^2 = \frac{1}{1 + \varepsilon^2 T_n^2\left(\frac{\omega}{\omega_g}\right)},$$

das Betragsquadrat der Übertragungsfunktion für den Tschebyscheff Filter vom Typ 2 ist

$$|G_{T2}(\omega)|^2 = \frac{1}{1 + \varepsilon^2 \frac{T_n^2\left(\frac{\omega_s}{\omega_g}\right)}{T_n^2\left(\frac{\omega_s}{\omega}\right)}},$$

§ 596 Die Beträge für einen Tschebyscheff Filter vom Typ 1 für verschiedenen Ordnungen sind im Vergleich zu Butterworthfiltern gleicher Ordnung in Abb. 5.11 dargestellt; für beide wurde  $\varepsilon = 0.2$  angenommen. Der Tschebyscheff Filter zeigt eine steilere Flanke, aber eben um den Preis der Welligkeit im Durchlassbereich. Die Amplitude der Welligkeit ist durch  $\varepsilon$  bestimmt – der Betrag der Übertragungsfunktion im Durchlassbereich schwankt zwischen 1 und ihrem Betrag bei der Grenzfrequenz.

**Zwischenrechnung 42** Wiederholen Sie Zwischenrechnungen 40 und 41 für einen Tschebyscheff Filter.

### 5.2.4 Analoge Filter in SIMULINK

§ 597 Analoge Filter können in SIMULINK am einfachsten mit Hilfe der kontinuierlichen Übertragungsfunktion realisiert werden. Diesen Ansatz haben wir in den bisher betrachteten Beispielen und Zwischenrechnungen verfolgt. Er ist immer dann sinnvoll, wenn wir die Übertragungsfunktion kennen (Tabellen) oder die Differentialgleichung kennen bzw. leicht aufstellen können.

§ 598 Ist die Differentialgleichung bzw. das Differentialgleichungssystem des Systems bekannt, so können wir auch die Beschreibung über den Zustandsraum **State Space** wählen. In Abb. 1.11 haben wir dies am Beispiel der erzwungenen Schwingung betrachtet – die DGL entspricht der des Serienschwingkreis, d.h. für letzteren haben wir die Formulierung mit Hilfe des Zustandsraums bereits durchgeführt.

**Zwischenrechnung 43** Damit sich die Filter erster Ordnung nicht benachteiligt fühlen, sollten Sie die Beschreibung mit Hilfe des **State Space** auch für einen Tief- oder Hochpass durchführen.

§ 599 Und wenn wir den Serienschwingkreis als Filter zweiter Ordnung ohnehin schon auf das Federpendel reduziert haben, können wir bei dem auch die dritte Möglichkeit der Simulation in SIMULINK ausleihen: die anschauliche Abbildung des Physik mit den entsprechenden Blöcken.

**Zwischenrechnung 44** Gönnen Sie dem Tief- oder Hochpass auch diese Realisierungsmöglichkeit. Wie gehen Sie in diesem Modell mit der Phasenverschiebung um?

## 5.3 Digitale Filter im Zeitbereich

§ 600 Analoge Filter arbeiten das Zeitsignal sequentiell ab: die Geschichte des analogen Signals ist z.B. in der Ladung auf einem Kondensator enthalten. Aber der Blick geht nur in die Vergangenheit, nicht in die Zukunft. Daher sind analoge Filter stets asymmetrisch und nicht umkehrbar. Ein digitaler Filter wird dagegen auf den bereits digitalisierten, vollständig vorliegenden Datensatz angewendet. Daher lassen sich auch symmetrische digitale Filter realisieren. Außerdem ist ein digitaler Filter ein diskreter Filter, da die digitalisierten Daten ja nur an bestimmten Stützstellen vorliegen.

§ 601 Ein einfaches Beispiel für die digitale Filterung ist die Glättung von Daten. Im analogen Bereich würde man einen Tiefpass verwenden, um dem Signal überlagerte hochfrequente Störungen heraus zu filtern – allerdings um den Preis einer Phasenverschiebung. Selbstverständlich lässt sich dessen Arbeitsweise auch als digitales Filter emulieren. Allerdings würde man zur Glättung digitaler Daten eher auf einen wesentlich einfacheren Filter zurück greifen: das gleitende Mittel. Wählt man dieses als zentrales gleitendes Mittel, so tritt die vom analogen Filter bekannte Phasenverschiebung nicht auf.

### 5.3.1 FIR-Filter

§ 602 Im Rahmen des Skriptes und der Vorlesung werden wir uns auf FIR (Finite Impulse Response) Filter beschränken: jeder Ausgangswert ist gegeben als die Summe einer endlichen Zahl gewichteter Eingangsgrößen.

§ 603 Dieser Ansatz ist ähnlich dem bei analogen Filtern verwendeten: beim Tiefpass wirkt der Kondensator als Verzögerungsglied, da sein Auf- und Entladen Zeit benötigt. Daher ist die momentan auf ihm befindliche Ladung, und damit auch der Spannungsabfall über ihm, nicht nur vom momentan anliegenden Eingangssignal sondern auch von früheren Eingangssignalen abhängig – wobei mit zunehmendem zeitlichen Abstand das Gewicht des Eingangssignals ebenfalls abnimmt. Formal führt dies auf ein Faltungsintegral: die über dem Kondensator abfallenden Spannung zu jedem Zeitpunkt ergibt sich aus dem Momentanwert der anliegenden Spannung und den geeignet gewichteten früheren Werten der Eingangsgröße.

**Verständnisfrage 62** Wie lange reicht das Gedächtnis eines Tiefpass wirklich zurück? Bis zum Einschalten? Oder wird der Tiefpass irgendwann vergesslich?

§ 604 Beim digitalen Filter arbeiten wir mit diskreten Signalen. Daher wird das Integral ersetzt durch eine gewichtete Summe über eine endliche Zahl von Eingangswerten, z.B.

$$\begin{aligned} u_{\text{out}}(?) &= a_4 u(n-4) + a_3 u(n-3) + a_2 u(n-2) + a_1 u(n-1) + a_0 u(n) \\ &= \sum_k a_k u(n-k). \end{aligned}$$

Außer in der Verwendung einer Summe statt eines Integrals unterscheidet sich der digitale Filter in einem weiteren Punkt wesentlich vom analogen Filter: bei einem analogen Filter müsste diese Ausgangsgröße  $u_{\text{out}}$  an der Stelle  $n$ , also am Ende des bei der Integration relevanten Zeitintervalls gesetzt werden:  $u_{\text{out}}(n)$ . Bei digitalen Filter dagegen kann dieser Ausgabewert z.B. auch in die Mitte des Summationsintervalls gesetzt werden: digitale Filter können symmetrisch sein während analoge Filter stets asymmetrisch sind. Oder anders

formuliert: analoge Filter können nur die Vergangenheit bis einschließlich des gegenwärtigen Zeitpunktes berücksichtigen (das steckt ja genau im Faltungsintegral). Digitale Filter dagegen greifen auch oder sogar ausschließlich (wenn wir den Ausgangswert an den Anfang des Summationsintervalls legen) auf die Zukunft zu.

§ 605 Greift ein Filter ausschließlich auf den aktuellen Wert und in der Vergangenheit liegende Werte zurück, so wird er als *kausal* bezeichnet: sein Output ist die Folge der früheren Inputs. Analoge Filter und damit alle Filter in Echtzeit-Anwendungen sind stets *kausale* Filter (außer der Tiefpass hat eine erfolgreiche parapsychologische Ausbildung absolviert und ist von Herrn von Däniken geschult worden). Filter, die wie ein zentrales gleitendes Mittel auf zukünftige Werte zugreifen, werden entsprechend als *nicht-kausale* Filter bezeichnet.

§ 606 Im Bezug auf das mathematische Handwerkszeug haben wir es jetzt nicht mehr mit Funktionen sondern Folgen als Signal zu tun. Entsprechend werden die Eigenschaften des Filters nicht mehr über ein Faltungsintegral sondern über eine Differenzgleichung beschrieben. Der Übergang zwischen der Folge  $y[n]$  der Ausgangsgrößen und der der Eingangsgrößen  $x[n]$  lässt sich aber weiterhin mit Hilfe eines Operators  $\mathcal{T}$  beschreiben:

$$y[n] = \mathcal{T}(x[n]) .$$

Der Operator  $\mathcal{T}$  kann sehr unterschiedliche Formen annehmen: von elementweisen Operationen über beliebige mathematische Verknüpfungen mehrerer Element bis hin zu einfachen vergleichen wie z.B.  $y_n = \max\{x_n, x_{n-1}, x_{n-2}\}$ .

### 5.3.2 IIR-Filter

§ 607 Durch Rückkopplung des Filterausgangs  $y[k]$  entstehen rekursive Filter. Diese können eine unendliche Impulsantwort haben und werden daher als IIR Filter (Infinite Impulse Response) bezeichnet:

$$y[n] = \sum_{k=0}^m b_k x[n-k] + \sum_{j=1}^l a_j y[n-j] .$$

Darin sind die  $b_k$  die bereits bekannten nicht-rekursiven (transversalen) und  $a_j$  die rekursiven Koeffizienten. Der rekursive Zweig des Filters hat die Ordnung  $m-1$ , der rekursive Zweig die Ordnung  $l-1$ . Auch rekursive Filter können eine endliche Impulsantwort haben!

### 5.3.3 Gleitendes Mittel

§ 608 Der einfachste digitale Filter ist das gleitende Mittel: in einem Datensatz wird jeweils über eine Zahl von  $m$  aufeinander folgenden Punkten gemittelt. Der Mittelwert wird in der Regel in die Mitte des Intervalls gesetzt, d.h.  $m$  sollte ungerade sein:  $m = 2k + 1$ . Dann ergibt sich für die Ausgangsgröße  $y(n)$  in Abhängigkeit von den Eingangsgrößen  $x_{n-k}$  bis  $x_{n+k}$ :

$$y(n) = \frac{1}{(2k+1)} \sum_{j=-k}^{j=+k} x(n+j) .$$

Diese Differenzgleichung definiert die Eigenschaften des digitalen Filters vollständig. Die Ordnung des digitalen Filters ist in diesem Fall  $m-1 = 2k$ : sie entspricht der Zahl der zusätzlich zum aktuellen Eingangswert einbezogenen Datenpunkte.

§ 609 Das gleitende Mittel wirkt wie ein Tiefpass: es glättet die schnellen Fluktuationen und macht einen möglichen Trend in den Daten sichtbar. Abbildung 5.12 zeigt im linken oberen Teil ein diskretes Zeitsignal, das idealisiert aus der Funktion im unteren Teilbild gewonnen wurde. Aus dem diskreten Datensatz lässt sich weder ein Trend noch eine Periodizität erkennen – aber auch nicht ausschließen. Ein gleitendes Mittel über drei Punkte (rote Kurve) im rechten oberen Teilbild lässt die längere Periode  $\omega = 1$  bereits deutlich hervortreten.

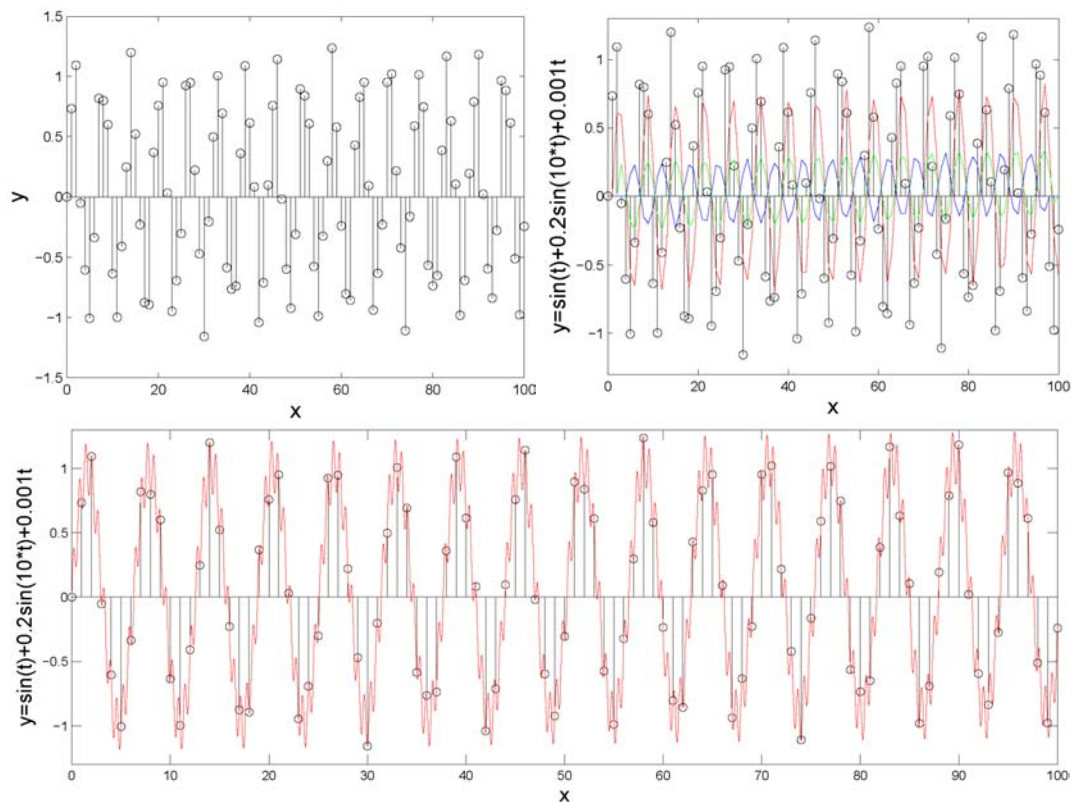


Abbildung 5.12: Auf den linken Satz diskreter Werte wurde ein gleitendes Mittel über drei (rot), fünf (grün) und neun (blau) Datenpunkte angewandt. Das gleitende Mittel wirkt wie ein Tiefpass: zumindest bei den längeren Mittelungsperioden wird der unterliegende Trend sichtbar

Gleitende Mittel über fünf (grün) bzw. neun (blau) Punkte dämpfen diese Periode bereits, so dass der unter liegende Trend deutlich wird.

**Verständnisfrage 63** Lässt sich das im unteren Teil von Abb. 5.12 gegebene diskrete Signal (schwarz) eigentlich wirklich auf analoge Weise aus dem zeitkontinuierlichen Signal (rot) erzeugen?

§ 610 Filter werden auf einen endlichen Datensatz angewendet. Die Zahl der zum Datensatz gehörigen Punkte wird als Basis oder Support bezeichnet. Für den Datensatz linken oberen Teilbild von Abb. 5.13 erstreckt sich der Datensatz über den Bereich  $x \in [5, 12]$ , die Ränder sind mit Nullen aufgefüllt. Das gleitende Mittel, wie auch fast alle anderen Filter, ist ein Verfahren, das zusätzlich zu dem Punkt an der Zielstützstelle auch benachbarte Punkte einbezieht. Für die Punkte an den Rändern des Datensatzes sind aber nicht immer alle im Algorithmus geforderten Nachbarn auch im Datensatz vorhanden. Verwendet man die benachbarten Nullen mit, so führt dies zu einer Aufweitung des Datensatzes: bei einem zentrierten gleitenden Mittel über drei Punkte (rot im rechten oberen Teilbild in Abb. 5.13) wird auf jeder Seite ein benachbarter Punkt einbezogen, beim zentrierten gleitenden Mittel über fünf Punkte (grün) wird der Datensatz auf jeweils zwei benachbarte Punkte ausgedehnt. Bei der kausalen Variante des gleitenden Mittels (rechts unten in Abb. 5.13) werden die Datenpunkte aus der Vergangenheit bezogen – entsprechend beginnt der gefilterte Datensatz gleichzeitig mit dem echten Datensatz, endet aber erst später. Beim nicht-kausalen Filter (links unten) endet der gefilterte Datensatz gleichzeitig mit dem Originaldatensatz, beginnt aber entsprechend früher.



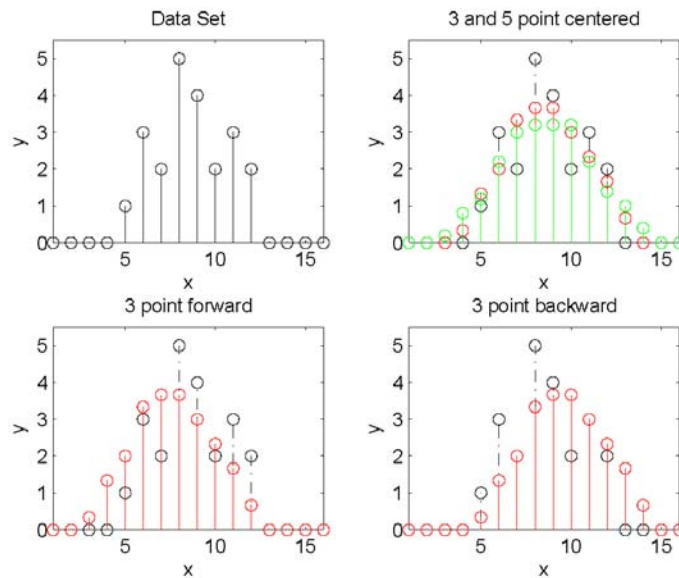


Abbildung 5.13: Gleitendes Mittel: der Datensatz ist von Null verschieden im Bereich  $x \in [5, 12]$  (*oben links*). Gleitende Mittel (zentriert) über 3 (rot) bzw. 5 (grün) Punkte erzeugen auch in dem Bereich Werte, in dem der Originaldatensatz Null ist (*oben rechts*). Vorwärts (*unten rechts*) oder rückwärts (*unten links*) gerichtete Mittel statt des zentrierten Mittels verschieben dieses nur

§ 611 Der Umgang mit diesen Datenpunkten ist eine Frage der eigenen Auffassung von Information. Das Ergänzen der Nullen um den Datensatz ist uns als Zero Padding im Zusammenhang mit der DFT z.B. in § 404 begegnet.<sup>4</sup> Unsere Entscheidung ist die Bewertung der Randpunkte: bei einem Filter  $m$ -ter Ordnung gibt es entsprechend  $m$  Punkte, die in irgendeiner Form von den ergänzten Nullen bestimmt sind. Bei einem kausalen Filter liegen alle diese Punkte am Anfang des Originaldatensatzes. Erst beim  $m + 1$ ten Punkt wird bei der Mittelbildung nicht auf eine ergänzte Null zurück gegriffen. Beim nur Zukunftsorientierten Filter sind die gefilterten Daten bereits ab dem ersten Datenpunkt des Originaldatensatzes korrekt, allerdings werden die letzten  $m$  Punkte durch das Einbeziehen der Nullen hinter dem Signal verfälscht. Beim zentrierten gleitenden Mittel sind die ersten und letzten  $m/2$  Punkte verfälscht. Alle zusätzlichen, außerhalb des Originaldatensatzes erzeugten, von Null verschiedenen Punkte sind ohnehin verfälscht, da ja dort bereits der aktuelle Wert eine ergänzte Null ist.

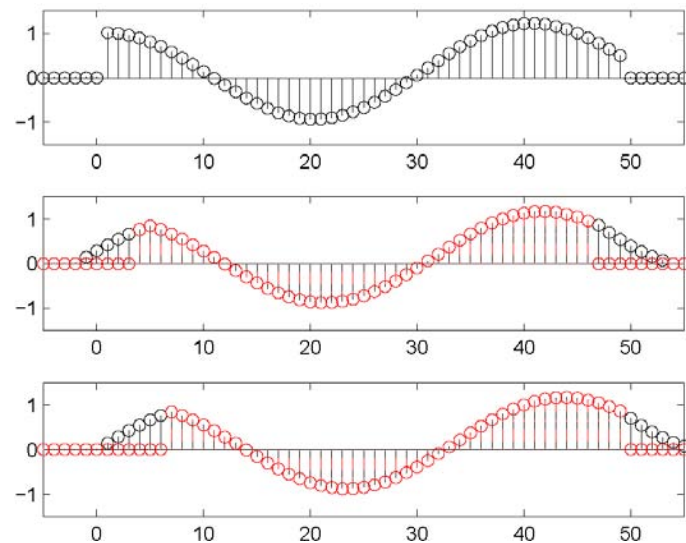
§ 612 Bei einem kausalen Filter würde man den am Anfang des Datensatzes entstehenden Fehler auch bei einem analogen Filter finden: die Lösung der Systemgleichungen enthält stets auch die Lösung der homogenen Gleichung und damit die Anfangsbedingungen. Diese werden gedämpft und sind daher nach einiger Zeit für das Systemverhalten nicht mehr von Bedeutung. Die ergänzten Nullen an den Kanten sind eine andere Variante von Anfangsbedingungen – im Prinzip hätten wir ja auch andere Zahlen willkürlich ergänzen können.

§ 613 Auch wenn in einigen Algorithmen die am Rand überlaufenden Werte und die fehlerhaften Werte mit genommen werden, bevorzuge ich die saubere Variante: nur die gefilterten Werte, die ausschließlich auf Werten aus dem ungefilterten Datensatz basieren, werden weiter berücksichtigt. Alle anderen werden verworfen. Auf diese Weise wird keine Pseudo-Information erzeugt und alle Datenpunkte im gefilterten Datensatz sind gleich behandelt. Der Nachteil besteht darin, dass sich der Datensatz verkürzt, siehe Abb. 5.14. Bei mehrfacherer Filterung eines Datensatzes kann dieser dann recht klein werden.

**Verständnisfrage 64** Wäre es denn eine bessere Alternative, erst mehrfach zu filtern und dann ganz am Ende die fehlerhaften ‘Randdaten’ zu verwerfen?

<sup>4</sup>Da uns das Zero Padding dort im Zusammenhang mit der Faltung begegnet ist und ein (analoger) Filter durch den Faltungskern im Faltungsintegral klassifiziert wird, machen wir an dieser Stelle durch das Ergänzen der Nullen auch nichts anderes als ein Zero Padding.

Abbildung 5.14: Gleitendes Mittel: der Datensatz ist für  $x \in [1, 49]$  gegeben, die Nullen an den Rändern sind ergänzt (*oben*). Ein zentriertes gleitendes Mittel 6. Ordnung (*Mitte*) über alle Punkte (schwarz) enthält aber nur im Bereich  $[4, 46]$  korrekte Werte; ein kausales gleitendes Mittel 6. Ordnung (*unten*) vergrößert den zuverlässigen Bereich nicht sondern schiebt ihn nach hinten



§ 614 Um ein zu starkes Schrumpfen des Datensatzes zu vermeiden, kann man statt der die ergänzten Nullen enthaltenden Datenpunkte innerhalb des Datensatzes mit einem entsprechenden reduzierten Filter Kernel arbeiten. Dieser berücksichtigt nur die wirklich vorhandenen Eingangsgrößen und passt die Gewichtungsfaktoren entsprechend an. Bei einem zentrierten gleitenden Mittel über sieben Punkte würde dann der erste Punkt als Mittel der ersten vier Datenpunkte gebildet (die drei Fehlenden werden gar nicht berücksichtigt), der zweite als Mittel über die ersten fünf Datenpunkte, der dritte entsprechend über sechs und ab dem vierten stehen alle benötigten Eingangsgrößen zur Verfügung. Entsprechend wird am Ende verfahren. Der viertletzte Punkt ist noch mit dem vollständigen Kernel bestimmt, beim drittletzten wird nur über sechs, beim vorletzten über fünf und beim letzten über vier Eingangsgrößen gemittelt.

§ 615 Gegenüber der Berücksichtigung der ergänzten Nullen hat das Verfahren bei allen Datensätzen, die nur Größen eines Vorzeichens enthalten, einen Vorteil. Betrachten wir z.B. die Helligkeitswerte eines Bildes. Diese sind stets positiv. Die Berücksichtigung der Nullen am Rand verringert diese Werte systematisch, wie auch in den Daten in Abb. 5.14 zu erkennen. Bei einem Bild kommt es damit zu einer Art Randverdunklung. Reduziert man dagegen den Kernel, so werden zwar weniger Punkte in die Mittelung einbezogen und das Mittel damit ungenauer; sein Wert liegt aber in einem angemesseneren Bereich und führt nicht zu einer systematischen Randverdunklung.

### 5.3.4 Gleitendes Mittel in SIMULINK und MATLAB

§ 616 MATLAB erzeugt ein gleitendes Mittel mit Hilfe der Funktion `filter`. Der gefilterte Datensatz hat die gleiche Länge wie der Eingangsdatensatz, die Randpunkte werden nach dem allgemeinen Algorithmus durch Ergänzen der Nullen erzeugt.

§ 617 Betrachten wir als Beispiel ein gleitendes Mittel ohne relative Gewichte, d.h. ein Filter der Länge  $l$  mit  $l$  identischen Koeffizienten  $a_i = 1/l$ . Den Filter Kernel können wir in MATLAB erzeugen (hier für  $l = 9$ ) als

```
bb = ones(9,1)/11
```

Dieser Kern ist mit den Daten `xx` zu falten:

```
yy=conv(bb,xx)
```

Die MATLAB-Funktion `filter` kennt eine abkürzende Schreibweise dafür:

```
filter(ones(1,9)/9,1,xx)
```

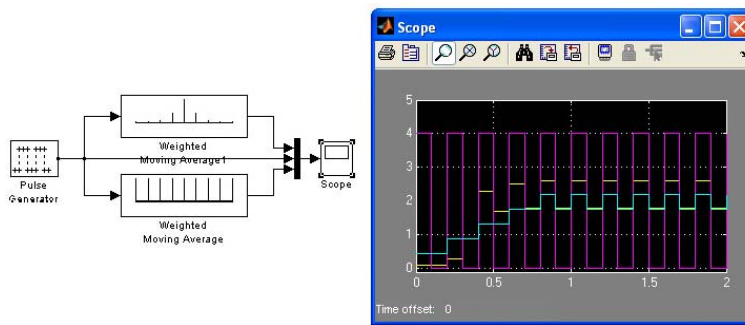


Abbildung 5.15: In SIMULINK lässt sich ein gleitendes Mittel mit Hilfe des Blocks **Weighting Moving Average** erzeugen

Und ein Blick in die MATLAB Hilfe zeigt uns, wie MATLAB dieses Problem löst: MATLAB hält sich an das Blockdiagramm, das wir bereits in Abb. 5.16 und 5.17 kennen gelernt haben.

§ 618 In SIMULINK steht im Library Browser im Ordner **Discrete** der Block **Weighting Moving Average** zur Bildung des gewichteten gleitenden Mittels zur Verfügung. In diesem Block muss lediglich der Filter-Kern spezifiziert werden; Abb. 5.15 zeigt ein Beispiel für die Verwendung des Blocks mit zwei unterschiedlichen Filter-Kernen. Im oberen Block wird mit maximalem Gewicht auf dem zentralen Datenpunkt gewichtet mit einem Kern  $[0.025, 0.025, 0.05, 0.2, 0.5, 0.2, 0.05, 0.025, 0.025]$ ; im unteren Block wird über neun Datenpunkte gleichförmig gewichtet.

§ 619 Dieses Modell hätte man einfacher aufbauen können. Der Block **Weighting Moving Average** kann im SISO (single input – single output) oder SIMO (single input – multiple output) Modus betrieben werden. In ersterem Fall wird der Filter-Kern als Vektor übergeben und erzeugt aus einem skalaren Input-Signal einen skalaren Output. Im zweiten Fall wird der Filter-Kern als Matrix übergeben, wobei jede Zeile der Matrix einen eigenen Filter beschreibt. Entsprechend wird aus dem skalaren Input ein Vektor, dessen Länge der Zahl der Zeilen der Matrix entspricht. Während sich beim SISO Modus die Gewichte in der Darstellung des Blocks abschätzen lassen, ist das im SIMO Modus natürlich nicht gegeben.

### 5.3.5 Der Filter-Kern

§ 620 Die gewichtete Summe definiert die Eigenschaften des Filters vollständig. Die Differenzgleichung lässt sich allgemein schreiben als

$$y[n-l] = \sum_{k=0}^m b_k x[n-k] = b[k] * x[k], \quad (5.1)$$

der letzte Term beschreibt die Filterung als Faltung der Eingangsgröße mit dem Filterkern. Sie bildet den Filter-Kern oder Kernel. Die Struktur bestimmt die Länge  $m+1$  des Filters (und damit seine Ordnung  $m$ ). Die Lage des Ausgabepunktes bestimmt, ob es sich um einen kausalen ( $l=0$ ), zentrierten ( $l=m/2$ ) oder nicht-kausalen ( $l < m$ ) Filter handelt, die Gewichtungsfaktoren  $b_k$  definieren die Form des Filters. In vielen Darstellungen digitaler Filter wird nur die Variante für den kausalen Filter verwendet, d.h. der Kernel wird angegeben als

$$y[n] = \sum_{k=0}^m b_k x[n-k] = b[k] * x[k], \quad (5.2)$$

§ 621 Ein zentriertes gleitendes Mittel über neun Datenpunkte mit gleicher Gewichtung ergibt sich damit für  $m=8$  (Ordnung des Filters als Länge  $L=9$  des Kernels minus 1),  $l=4$  und  $b_k=1/9$ .

§ 622 Verschiedene Gewichtungen der einzelnen Datenpunkte drücken sich in unterschiedlichen  $b_k$  aus. Sind alle  $b_k$  identisch, so werden die Datenpunkte gleichberechtigt behandelt

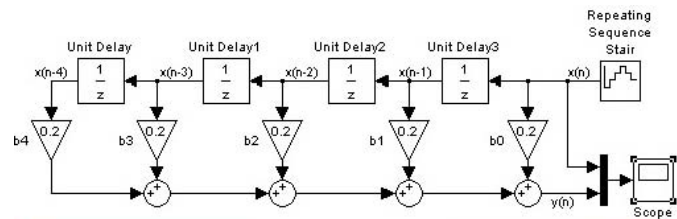
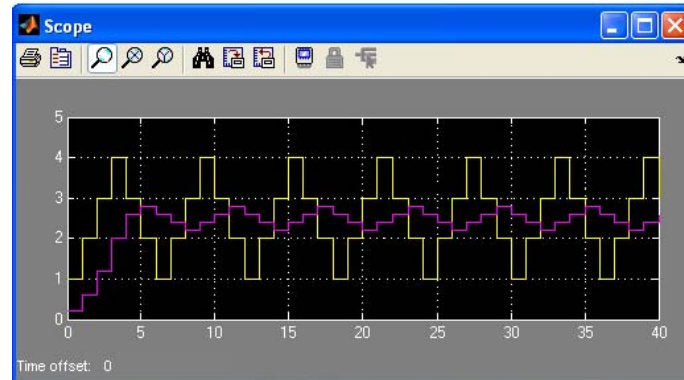


Abbildung 5.16: Darstellung eines digitalen Filters mit Hilfe eines Blockdiagramms, in diesem Fall kausales gleitendes Mittel 4. Ordnung. Die Darstellung mit Hilfe eines Blockdiagramms gilt allgemein auch wenn es hier aus dem SIMULINK Modell genommen wurde. Beim allgemeinen Blockdiagramm fallen Quelle, Multiplexer und Senke weg



(lineare Filterung). Vielleicht gibt es Gründe, dem zentralen Datenpunkt ein größeres Gewicht zuzuschreiben. Dann werden die benachbarten Punkte als kleine Korrekturen aufgefasst und erhalten einen entsprechenden kleineren Gewichtungsfaktor. Der Gauß'sche Filter als ein Beispiel für eine derartige Gewichtung wird häufig zur Rauschunterdrückung verwendet.

### 5.3.6 Die Impulsantwort

§ 623 Der Filter-Kernel ist eine Möglichkeit, die Eigenschaften eines Filters zu beschreiben; die Impulsantwort ist eine andere. Letztere ist die Antwort des Filters auf eine  $\delta$ -Eingangsgröße mit

$$\delta[n] = \begin{cases} 1 & \text{für } n = 0 \\ 0 & \text{sonst} \end{cases} .$$

Der Filter-Kernel (5.1) lässt sich damit umschreiben als

$$y[n+l] = \sum_{k=0}^m b_k x[k] \delta[n-k] .$$

Oder verbal: das Ausgangssignal wird dadurch erzeugt, dass man skalierte verschobene Impulse aufsummiert.

§ 624 Beide Darstellungen sind äquivalent. Die Darstellung mit Hilfe der Impulsantwort hat den Vorteil, dass sich eine Verschiebung oder Verzögerung um  $n_0$  Samples einbauen lässt:  $y[n] = x[n-n_0]$ . Mit  $n_0 = 1$  ergibt sich die Einheitsverzögerung; den als **Unit Delay** bezeichneten Block findet man in SIMULINK in der Bibliothek **Discrete**. Dieses Verzögerungssystem lässt sich, hier für das Beispiel  $n_0 = 2$  auch mit Hilfe eines normalen Kernels für einen kausalen Filter (5.2) durch die Angabe der Filterkoeffizienten  $\{b_k\} = \{0, 0, 1\}$  erreichen:

$$y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] = x[n-2] .$$

In kurzer Form lässt sich dieses Filter durch  $h[n] = \delta[n-2]$  beschrieben mit  $h[n]$  als Impulsantwort.

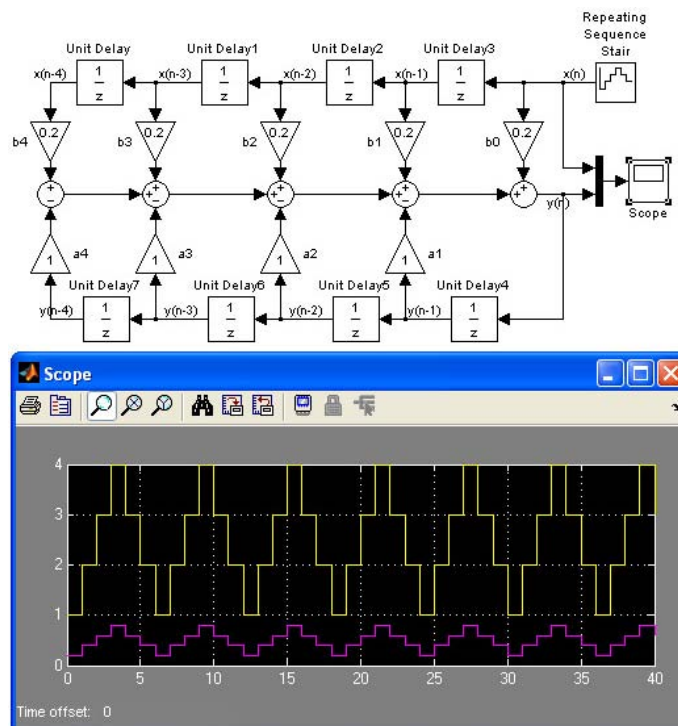


Abbildung 5.17: Blockdiagramm für ein rekursives Filter, aufbauend auf dem FIR Filter gleitenden Mittel in Abb. 5.16

### 5.3.7 Block Diagramm

§ 625 Ein digitaler Filter lässt sich auch mit Hilfe eines Blockdiagramms darstellen. Dazu werden die folgenden Blöcke benötigt: der Multiplizierer, der Addierer und die Einheitsverzögerung. Oder in den Begriffen von SIMULINK: der Gain, der Sum-Block und der Unit Delay.

§ 626 Betrachten wir dazu ein kausales Filter 4. Ordnung. Allgemein gilt dann

$$y[n] = b_0x[n] + b_1x[n-1] + b_2x[n-2] + b_3x[n-3] + b_4x[n-4].$$

Für eine ungewichtete Mittelung ist  $b_i = 0.2$  für alle  $i$ . Die zugehörige Darstellung in einem Blockdiagramm ist, gleich als SIMULINK Modell, im oberen Teil von Abb. 5.16 gegeben. Beim eigentlichen Blockdiagramm gehören Quelle, Multiplexer und Scope nicht dazu. Das Eingangssignal wird in diesem Fall als sich wiederholende Stufenfunktion realisiert (ein diskreter Sinus würde es auch tun). Das Signal wird von oben rechts elementweise gelesen. Das erste Element des Signals ist  $x[1]$ , die anderen Felder rechts davon enthalten jeweils den Anfangswert, in der Regel Null. Mit jedem neuen aus dem Eingangssignal gelesenen Element werden die bereits vorhandenen Elemente um einen Block nach links verschoben. Diese werden mit den Gewichtungsfaktoren multipliziert und anschließend addiert.

§ 627 Das untere Teilbild in Abb. 5.16 zeigt das Eingangssignal (gelb) sowie das gefilterte Signal (magenta). Der Filter ist kausal, d.h. die fehlerhaften Werte (durch Ergänzung der Null) treten nur am Anfang auf – und sind dort mit etwas Kopfrechnen auch zu identifizieren. Da wir ein periodisches Eingangssignal gewählt haben, ist die durch das gleitende Mittel bewirkte Glättung gut zu erkennen.

§ 628 Auch rekursive Filter (IIR Filter) lassen sich im Blockdiagramm darstellen. Dazu muss die Ausgangsgröße nochmals auf den Filter zurück gegeben werden. Abbildung 5.17 gibt ein auf dem gleitenden Mittel aus Abb. 5.16 aufbauendes rekursives Filter. Mit  $a_i = 1$  ergibt sich ein stabiles Ausgangssignal, das den Eingang effizient dämpft. Mit  $a_i = 2$  dagegen wird der Filter instabil – der Filter hätte keine endliche Impulsantwort.

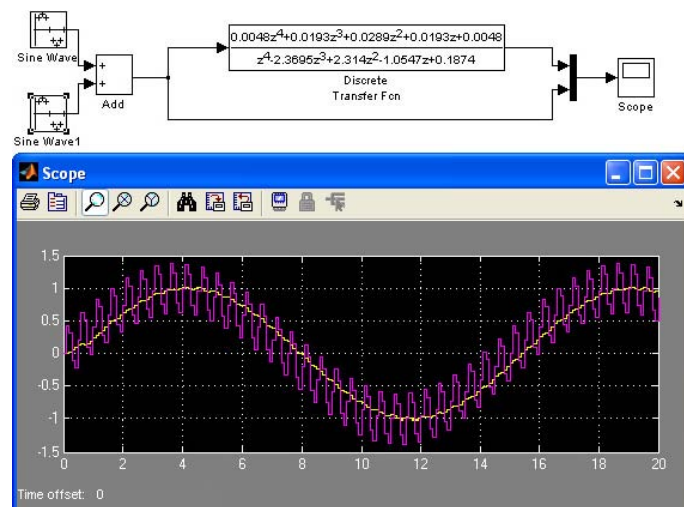


Abbildung 5.18: Butterworth Filter vierter Ordnung als Beispiel für die Verwendung einer diskreten Übertragungsfunktion

### 5.3.8 Digitale Filter in SIMULINK

§ 629 Zur Simulation digitaler Filter stehen in SIMULINK die gleichen Verfahren zu Verfügung wie in Abschn. 5.2.4 im Zusammenhang mit analogen Filtern genannt: Übertragungsfunktion, Zustandsraum oder Blockdiagramm sowie das bereits in Abschn. 5.3.4 beschriebene abgekürzte Verfahren für ein gewichtetes gleitendes Mittel. Der einzige Unterschied besteht in der Verwendung von Blöcken für diskrete anstelle kontinuierlicher Variablen; es muss im Library Browser aus dem Ordner `Discrete` ausgewählt werden. Die Variante mit dem Blockdiagramm haben wir in Abschn. 5.3.7 und insbesondere Abb. 5.16 und 5.17 bereits kennengelernt. Die Verwendung der diskreten Übertragungsfunktion betrachten wir am Beispiel eines Butterworth Filters.

§ 630 Abbildung 5.18 zeigt die Realisierung eines digitalen Butterworth Filters vierter Ordnung in SIMULINK. Der Filter entfernt das überlagerte hochfrequente Signal effizient und bewirkt, im Gegensatz zum Tiefpass in Abb. 5.5 keine Phasenverschiebung.

§ 631 Da wir ein digitales Filter betrachten, haben wir es nicht mit zeitkontinuierlichen sondern diskreten Signalen zu tun. Der Aufbau der Simulation entspricht dem auch in Abb. 5.5 für eine zeitkontinuierliche Simulation verwendeten; der einzige Unterschied ist die Verwendung des Blocks `Discrete Transfer Function` anstelle des Blockes `Transfer Function` für zeitkontinuierliche Signale. Die beiden Signalquellen links sind die normalen Sinus-Blöcke; jedoch ist auf der Parameterkarte des Sinus der Sine Type nicht als Time Based angegeben sondern als Sample Based. Dann müssen wir auch eine Sample Time (in diesem Beispiel 0.1) angeben sowie die Zahl der Samples pro Period (hier 150 bei einer Amplitude von 1 im oberen Sinus Generator und 5 bei einer Amplitude von 0.4 im unteren). Alle anderen Blöcke beziehen ihre Sample Time von diesen Signalgebern, d.h. dort muss die Sample Time nicht eingestellt werden sondern kann auf -1 stehen bleiben. In der Simulationskontrolle können wir uns für einen normalen Solver entscheiden oder für einen diskreten: ein diskreter Solver kann nicht auf ein System angewendet werden, in dem es zeitkontinuierliche Zustände gibt. Ein zeitkontinuierlicher Solver dagegen kann auch auf ein diskretes System angewendet werden.

§ 632 Und wo kriegt man die Koeffizienten für die Übertragungsfunktion her? In Abb. 5.18 ist die Übertragungsfunktion so gewählt, dass die Ordnung des Filters 4 beträgt und die Sperrfrequenz  $\omega_s = 0.2$ . Ist man glücklicher Besitzer der MATLAB Signal Processing Toolbox, so lassen sich die Koeffizienten des Zähler- und Nennerpolynoms der Übertragungsfunktion mit dem Befehl `butter` bestimmen. Daher ist es sinnvoll, diese Simulation aus einem m-File aufzurufen, in dem man vorher die entsprechenden Parameter bestimmen lässt.

## 5.4 Filter im Spektralbereich

§ **633** Filter im Spektralbereich basieren auf einer Verwendung des Faltungsprodukts anstelle des Faltungsintegrals: Funktion und Filter Kernel werden bei einem kontinuierlichen Signal Fourier transformiert und multipliziert; das sich ergebende Produkt wird zurück transformiert. Dieses Verfahren liegt der Angabe der Übertragungsfunktion in  $s$  zu Grunde – wir haben es also bereits die ganze Zeit mehr oder weniger bewusst verwendet.

### 5.4.1 $z$ -Transformation

§ **634** Bei einem digitalen Filter gibt es ein entsprechendes Verfahren. Die  $z$ -Transformation ist eine allgemeine Form der diskreten Fourier Transformation DFT. Für unsere Zwecke können wir die  $z$ -Transformation als eine diskrete Laplace Transformation auffassen.

§ **635** Betrachten wir dazu ein Signal  $x[n]$  endlicher Länge:

$$x[n] = \sum_{k=0}^N x[k] \delta[n - k] .$$

Dieses wird aus der  $n$ -Domäne, die ja letztendlich nach Multiplikation mit der Sampling Zeit auch nur eine, wenn auch diskrete Zeitdomäne ist, in die  $z$ -Domäne. Dabei ist  $z$  eine komplexe Größe, manchmal auch als Frequenz bezeichnet. Das transformierte Signal ist

$$X(z) = \sum_{k=0}^N x[k] z^{-k} = \sum_{k=0}^N x[k] (z^{-1})^k .$$

Die letzte Umformung weist nur noch einmal darauf hin, dass  $X(z)$  letztendlich ein Polynom  $N$ ter Ordnung in  $z^{-1}$  ist.

§ **636** Die  $z$ -Transformation hat als wichtigste Eigenschaften mit der Fourier und der Laplace Transformation die Faltung gemeinsam – also jenes mathematische Konstrukt, das uns den Umgang mit der Übertragungsfunktion so einfach macht. Die Verwendung der  $z$ -Transformation bei diskreten Signalen wird auch in den SIMULINK Blöcken deutlich: die Übertragungsfunktion für zeitkontinuierliche Signale ist in  $s$  formuliert, die für diskrete Signale in  $z$ .

§ **637** Für die Eigenschaften des Filters sind, wie bereits in Abschn. 4.3.1 angesprochen, die Pole und die Nullstellen. Deren Angabe ist zur Charakterisierung der Filtereigenschaften völlig ausreichend.

## 5.5 Filter im Ortsbereich

§ **638** Im Ortsbereich können die gleichen Filter verwendet werden wie im Zeitbereich. Auch die Fourier-Transformation und das damit verbundene Faltungsprodukt können direkt übertragen werden. Während bei der Filterung im Zeitbereich der Übergang von der Zeit auf die Frequenz erfolgt, wird im Ortsbereich der Übergang von der Raumkoordinate zur Wellenzahl gemacht.

§ **639** In zwei Punkten unterscheidet sich die Filterung im Ortsbereich von der im Zeitbereich:

1. die Idee des kausalen Filters ist nicht mehr notwendig: alle Ortsdaten liegen gleichzeitig vor. Daher können symmetrische Filter verwendet werden, so dass es nicht zu einer Translation der gefilterten Punkte (entsprechend der Phasenverschiebung in der zeitlichen Domäne kommt).

2. entsprechend der Zahl der Raumdimensionen kann die Ordnung der Daten im Ortsbereich eine höhere Dimension haben als die im Zeitbereich: (fast) jeder Pixel eines Bildes hat neben dem linken und rechten Nachbarn (entsprechend dem Vorgänger und Nachfolger im Zeitbereich) noch weitere Nachbarn oben, unten und entlang der Diagonalen. Da der Filter-Kernel auch den Zusammenhang mit diesen Pixeln berücksichtigt, wird er nicht wie bei einem 1D-Datensatz durch einen Vektor beschrieben sondern durch eine Matrix.

## Literaturhinweise

§ 640 Filterung von Signalen, egal ob mit SIMULINK oder einem anderen Tool, ist Thema der Signalanalyse und -verarbeitung. Da die Implementation einer Übertragungsfunktion in SIMULINK nach den Übungen aus den voran gegangenen Kapiteln keine Probleme aufwerfen sollte, beschränken sich die Literaturangaben auf die Inhalte. Einfach zu lesen, aber dennoch recht weit gehend und mit vielen Beispielen versehen, ist ‘Signal Processing First’ von McClellan, Schafer und Yoder [94]. Eine deutschsprachige Alternative ist ‘Werkzeuge der Signalverarbeitung’ von Meffert und Hochmuth [95]. Adaptive Filterverfahren unter Verwendung von MATLAB werden in Poularkis und Ramadan [124] beschrieben. Die Anwendung vieler dieser Filter auf zweidimensionale Datensätze ist nichts anderes als Bildverarbeitung. Eine gute Einführung in die digitale Bildverarbeitung gibt Jähne [60], für die Anwendungen in der Erdfernerkundung siehe auch Sabins [133], für medizinische Anwendungen [34] und mit sehr viel mathematischem Background [23].

## Fragen

**Frage 16** Erläutern Sie die wesentlichen Unterschiede zwischen digitalen und analogen Filtern.

**Frage 17** Was ist ein kausaler Filter? Muss ein Filter kausal sein?

**Frage 18** Welche Möglichkeiten der formalen Darstellung haben Sie für Filter?

**Frage 19** Warum lassen die die Konzepte der Filterung problemlos aus dem Zeit- in den Raumbereich übertragen – insbesondere, wenn man auf das Faltungsintegral zurück greift?

**Frage 20** Was ist ein FIR, was ein IIR-Filter? Wie unterscheiden sich die beiden konzeptionell?

**Frage 21** Was versteht man unter dem Filter-Kern(el)?

**Frage 22** Was ist die Basis bzw. der Support eines Filters?

**Frage 23** Erläutern Sie den Zusammenhang zwischen Filter-Kern und Übertragungsfunktion.

## Kurzprojekte

### Filterung in Bildern

§ 641 Bildverarbeitungsprogramme von GIMP bis Photoshop kennen verschiedenen Formen von Filtern. Dazu gehören z.B. die Glättung (Unterdrückung des statistischen Rauschens) mit Hilfe eines gleitenden Mittelwertes (lineare oder Gauß’sche Gewichtung, unterschiedlich große Basen). Glättung ist im zweidimensionalen Bereich genauso wie im Zeitbereich letztendlich nur die Anwendung eines Tiefpass. Auch das Äquivalent zum Hochpass erlauben derartige Programme: die Kantenanreicherung erlaubt eine bessere Identifikation von Übergängen, z.B. Grenzen zwischen Objekten. Unter anderem können sie mit der Kantenanreicherung die Umrisse eines Objekts erzeugen: Subtraktion des Originalbildes von dem Hochpass-gefilterten



Bild liefert nur an den Übergängen ein Signal. Suchen Sie sich Beispielbilder, konstruieren Sie verschiedene Filter und spielen Sie mit deren Anwendung – oder kurz: bauen Sie sich einen Mini-Photoshop in SIMULINK oder MATLAB. Hinweis: zum Einlesen der Bilder benötigen Sie in MATLAB die Funktionen `imread` bzw. `imwrite`. Hilfe liefern z.B. [16, 34, 133].

### Nonlinear Diffusion Toolbox

§ 642 Ein eher auf MATLAB als auf SIMULINK bezogenes Filterproblem insbesondere bei Bildern und mehrdimensionalen Datensätzen ist die diffusive Filterung. Die Nonlinear Diffusion Toolbox bzw. das Anisotropic Diffusion Model (<http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=14995&objectType=file>) stellt einige der Filter unter MATLAB zur Verfügung. Arbeiten Sie sich in die Theorie und die Möglichkeiten dieses Verfahrens ein. Eine Implementation der Algorithmen in SIMULINK ist nicht unbedingt sinnvoll.

**Teil III**

**Dynamische Systeme**

## Etwas Chaos

This is only true when small variations in the initial circumstances produce only small variations in the final state of the system. In a great many physical phenomena this condition is satisfied; but there are other cases in which small initial variations may produce a very great change in the final state of the system, as when the displacement of the ‘points’ causes a railway train to run into another instead of keeping its proper course.

J.C. Maxwell, 1874

Embedded in the mud, glistening green and gold and black was a butterfly, very beautiful and very dead. It fell to the floor, an exquisite thing, a small thing that could upset balances and knock down a line of small dominoes and then big dominoes and then gigantic dominoes, all down the years across time.

R. Bradbury, 1952

§ 643 Klassischerweise wird in der Physik nach einfachen Beschreibungsmöglichkeiten natürlicher Vorgänge gesucht; diese Konzepte und Modelle sollen vielfältig anwendbar sein bei gleichzeitiger einfacher mathematischer Handhabbarkeit. Die Beschäftigung mit dem (deterministischen) Chaos liefert Erkenntnisse, die im Gegensatz zu dieser traditionellen Herangehensweise stehen. Dazu gehören:

- Eine längerfristige Vorhersage über das Verhalten mancher Systeme ist prinzipiell nicht möglich – ganz im Gegensatz zu dem am Anfang von 3 bemühten Laplace’schen Dämon. Diese Unmöglichkeit der langfristigen Vorhersage wird schon bei einfachen Problemen wie der Planetenbewegung offenbar. Andere, experimentell einfach zu realisierende Beispiele sind das Pendulumonium (Pendelmann) oder ein Magnetpendel (Fadenpendel mit Eisenkugel, die über zwei oder mehr Magneten schwingt). Das prominenteste Beispiel ist der Schmetterlingseffekt [85] – Bradbury hat über ein Jahrzehnt vor Lorenz diese Idee bereits verwendet.
- Sehr kleine Änderungen bei bestimmten Systembedingungen können bedeutsame große Wirkungen hervorbringen: das System reagiert sehr empfindlich auf die Anfangsbedingungen. Daher können kleine Ursachen große Wirkungen haben – die Linearität, die wir in vielen Kausalbeziehungen implizit annehmen, ist in chaotischen Systemen nicht gegeben. Der Übergang zwischen Schwingung und Kreisbewegung beim Fadenpendel ist uns als einfaches Beispiel bereits begegnet.
- Die linearen Näherungen der klassischen Physik bilden nur einen begrenzten Teil der Welt ab und nichtlineares Verhalten ist eher der Regelfall. Einerseits können sich komplexe Systeme bei bestimmten Systembedingungen ganz einfach verhalten (Ordnung aus Chaos) und andererseits können bereits einfache Systeme chaotisches Verhalten zeigen.

- Chaostheorie liefert, wieder im Sinne der konventionellen Vorstellung von Physik, ein Werkzeug zur Beschreibung von Vorgängen, das in vielen Wissenschaftsbereichen angewendet werden kann.

§ 644 In diesem Kapitel werden wir kurz die Grundbegriffe der Chaostheorie an einigen Beispielen einführen bzw. wiederholen. Der Anfang ist sehr elementar: im Rückgriff auf die algebraische Schleife werden wir einen kurzen Blick auf Attraktoren und Fraktale werfen. Der für das Verständnis dynamischer Systeme wichtige Begriff der Bifurkation wird dabei ebenfalls eingeführt. Einfache Wachstumsmodelle, insbesondere die logistische Gleichung, helfen zur Illustration.

§ 645 Chaos haben wir bereits in Abschn. 1.1.5 am Beispiel des Fadenpendels eingeführt. Hier beginnen wir mit einfachen Variationen über ein einzelnes Pendel mit dessen Hilfe wir u.a. das Poincare-Portrait als Beschreibungsmöglichkeit in einem chaotischen System einführen werden.

§ 646 Im folgenden Schritt werden wir den Übergang zu einer größeren Zahl gekoppelter Pendel – auch hier haben wir in Abschn. 1.2 bereits einen ersten Eindruck gewonnen. Die gekoppelten Pendel können bei nicht-linearer Kopplung den Übergang zur Emergenz bilden: neue Strukturen oder Eigenschaften, in diesem Fall Solitonen, entstehen, die mehr sind als die Summe der Eigenschaften der einzelnen Oszillatoren: das Ganze ist mehr als die Summe seiner Teile.

§ 647 Alle in diesem Kapitel betrachteten Beispiele für Chaos sind deterministisch: die Regeln sind eindeutig und die Ergebnisse sind reproduzierbar – das chaotische Verhalten entsteht lediglich durch die Sensitivität der Systeme gegenüber den Anfangsbedingungen. Deterministisches Chaos ist in Übereinstimmung mit dem Laplace'schen Dämon: dieser muss ja neben den physikalischen Gesetzen auch eine Momentaufnahme aller Anfangsbedingungen haben – und die Sensitivität auf letztere ist es, die ins Chaos führt und im realen Systeme die Vorhersagbarkeit erschweren kann. Chaotisches Verhalten auf Grund stochastischer Einflüsse auf das System wird hier nicht betrachtet,

**Lernziele:** nach Durcharbeiten dieses Kapitels sollen Sie in der Lage sein,

- die Frage, was deterministisches Chaos ist, zu erläutern.
- Grundbegriffe wie Attraktor, Bifurkation und Selbstähnlichkeit zu erläutern und anzuwenden.
- das Verhalten chaotischer Systeme zu beschreiben (z.B. mit Hilfe von Poincare-Portraits),
- das Verhalten einfacher chaotischer Systeme in MATLAB oder SIMULINK zu simulieren.

## 6.1 Fraktale: geometrische Annäherung an das Chaos

§ 648 In Abschn. 2.6.6 sind wir der algebraischen Schleife als einer Quelle möglicher Unsicherheit in einer Simulation begegnet. Dieses Problem ist nicht auf SIMULINK beschränkt: algebraische Schleifen können auch in allen anderen Hochsprachen zu einem unerwünschten Systemverhalten führen. Eine algebraische Schleife ist also kein Problem einer speziellen Realisierung eines Algorithmus sondern bereits das Problem des Algorithmus.

### 6.1.1 Feigenbaum Attraktor und Apfelmännchen

§ 649 Dieses Systemverhalten ist nicht nur unerwünscht – es ist ein typisches Beispiel für Chaos. Mathematisch bedeutet eine algebraische Rückkopplung im einfachsten Fall nur die Definition einer Folge, z.B.  $a_{n+1} = ba_n + c$ . Mitchell Feigenbaum [26, 27] hat sich intensiv mit diesem Problem befasst. Bei der Untersuchung der Folge

$$x_{k+1} = \alpha x_k (1 - x_k) \quad \text{mit} \quad x_0 = 0.5$$

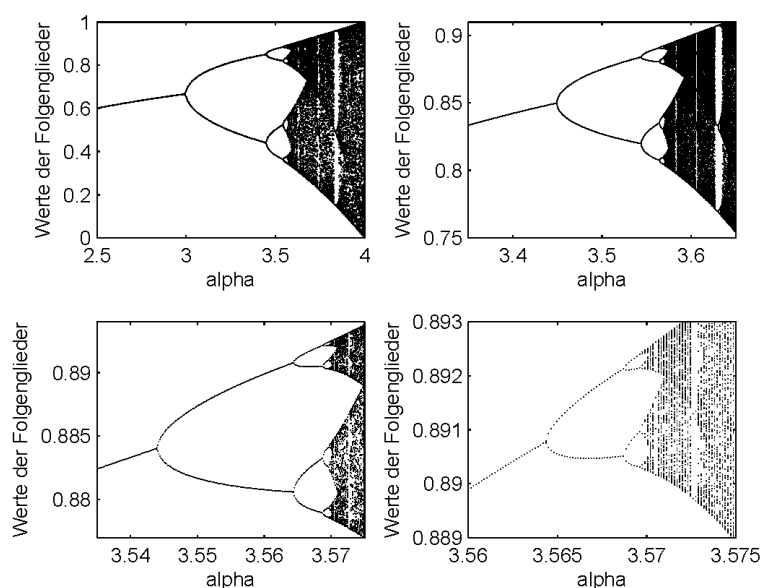


Abbildung 6.1: Der Feigenbaum Attraktor ist selbstähnlich

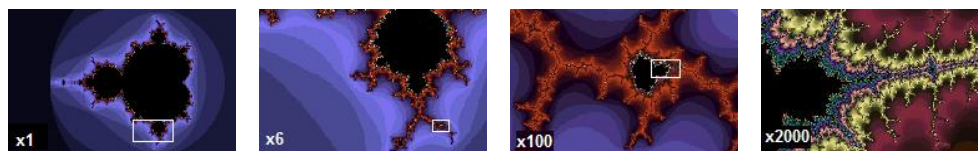


Abbildung 6.2: Apfelmännchen und Selbstähnlichkeit <http://en.wikipedia.org/wiki/Fractal>

fand er z.B. eine starke Abhängigkeit der Konvergenzeigenschaften vom Parameter  $\alpha$ . Abbildung 6.1 ist eine Reproduktion des nach ihm benannten Feigenbaum Attraktors.<sup>1</sup> Betrachten wir nur das linke obere Teilbild: auf der Abszisse ist der Parameter  $\alpha$  aufgetragen. Dieser wurde in kleinen Schritten variiert. Für jeden der Werte von  $\alpha$  sind auf der Ordinate die Werte der Folgenglieder  $x_{201}$  bis  $x_{300}$  aufgetragen. Für  $\alpha < 3$  scheint die Folge zu konvergieren: die Werte von  $x_{201}$  bis  $x_{300}$  fallen alle in die enge Umgebung eines Punktes. Dieser wird als Attraktor bezeichnet. Bei einem  $\alpha$  in der Nähe von 3 verzweigt sich die Lösung: die Lösung wird an dieser Bifurkation plötzlich sensitiv gegen kleine Variationen im Anfangswert.

§ 650 Hinter dieser Bifurkation gibt es für jedes  $\alpha$  nicht nur einen Attraktor sondern zwei Punkte, in deren Nähe sich die Glieder der Folge sammeln. Mit zunehmendem  $\alpha$  kommt es auf beiden Zweigen zu einer erneuten Bifurkation bis die Lösung schließlich chaotisch wird.

§ 651 Die Umgebung der oberen zweiten Bifurkation ist im rechten oberen Teilbild noch einmal genauer betrachtet (allerdings für die ersten 3000 Folgeglieder, von denen jeweils  $x_{2001}$  bis  $x_{3000}$  aufgetragen sind): qualitativ unterscheidet sie sich nicht von dem ursprünglichen Bild. Wir können auch hier wieder in die Umgebung der zweiten oberen Bifurkation hinein zoomen und erhalten nochmals ein ähnliches Bild. Der Feigenbaum Attraktor hilft nicht nur, den Begriff des Attraktors einzuführen sondern auch den der Selbstähnlichkeit.<sup>2</sup>

§ 652 Ein weiteres und wahrscheinlich bekannteres Beispiel für Selbstähnlichkeit ist das Ap-

<sup>1</sup>Ein physikalisches Beispiel, in dem die graphische Lösung dem Feigenbaum Attraktor ähnelt, ist der Duffing-Oszillator, siehe auch Abschn. 6.3.3.

<sup>2</sup>Die Verzerrungen zwischen den einzelnen Teilbildern sind durch eine suboptimale Auswahl der Grenzen des Teilbildes bedingt; die geringere Punktedichte im rechten unteren Teilbild ergibt sich durch das Hineinzoomen in das Bild: zwar wurde der Ausschnitt verkleinert, die Rechartiefe jedoch nicht erhöht.



Abbildung 6.3: Rügen's Küstenlänge hängt vom verwendeten Maßstab ab; zwischen den Abbildungen wird der Maßstab jeweils um einen Faktor 4 verringert. Abbildungen aus einem älteren DLR Archiv (KOSMOS Satellit), für neuere Bilder und eigene Spielereien hilft Google-Earth (<http://earth.google.com/>)

felmännchen, auch bezeichnet als Mandelbrot Menge.<sup>3</sup> Es ist eine andere Form der Darstellung von Konvergenz. Hier wird die Folge

$$z_{k+1} = z_k^2 + c \quad \text{mit} \quad z_0 = 0 \quad \text{und} \quad z, c \in \mathbb{C}$$

betrachtet. In Abhängigkeit vom Wert des komplexen Kontrollparameters  $c$  liegt der Häufungspunkt im Endlichen (dann gehört  $c$  zur Mandelbrot Menge) oder nicht (dann gehört  $c$  nicht zur Mandelbrot Menge). Das bedeutet, dass für jeden Wert von  $c$  die Reihe gebildet und auf Konvergenz untersucht werden muss. Dem Rechner ist dies, zumindest in einem numerischen Programm, nicht möglich. Daher beschränkt man die Bildung der Reihe auf eine bestimmte Zahl  $N$  von Schritten, z.B. 1000. Zwar kann man Konvergenz streng genommen nicht zeigen, allerdings lässt sich mathematisch zeigen, dass die Folge auf jeden Fall divergiert, wenn für ein Glied der Folge  $|z_k| > 2$  gilt. Ist dies nicht der Fall bevor  $z_N$  erreicht wurde, so geht man davon aus, dass die Folge für dieses  $c$  konvergiert. In der komplexen Ebene lassen sich so alle Punkte markieren, die zur Mandelbrot Menge gehören. Für das klassische Apfelmännchen wird dabei der Ausschnitt  $\Re(c) \in [-2, 0.5]$  und  $\Im(c) \in [-1.25, 1.25]$  betrachtet – außerhalb dieses Bereiches divergiert die Folge sehr schnell. Unterscheidet man nur nach Divergenz und Konvergenz, so bietet sich eine Schwarz-Weiß-Darstellung des Apfelmännchens an. Bei Farbdarstellungen wie in Abb. 6.2 wird die Farbkodierung abhängig davon gewählt, bei welchem Index  $k$  für das Folgeglied  $x_k$  Divergenz sicher festgestellt werden konnte. Mit zunehmendem  $N$  werden die Abbildungen immer Detail reicher.

§ 653 Apfelmännchen sind, wie für Fraktale gefordert, selbstähnlich. Dies zeigt sich am deutlichsten, wenn man immer weiter in ein Apfelmännchen hinein zoomt: überall finden sich neue Apfelmännchen, siehe Abb. 6.2.

§ 654 Für das Spielen mit Fraktalen können Sie die Testversion von Ultra Fractal (<http://www.ultrafractal.com/>) oder die leider nicht mehr gewartete Freeware FractInt (<http://spanky.triumf.ca/www/fractint/fractint.html>) verwenden. Auf der Seite finden Sie auch weitere Links. Ein recht umfangreiches MATLAB-Programm zur Untersuchung von Fraktalen ist der `fractal_explorer` von L. Cavin: für Informationen siehe <http://ltdmail.ethz.ch/cavin/fractals.html#SOFT>, für ds Programm auch <http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=5573&objectType=FILE>.

### 6.1.2 Selbstähnlichkeit und fraktale Dimension

§ 655 Einen Aspekt von Fraktalen haben wir sowohl im Zusammenhang mit dem Feigenbaum-Diagramm in Abb. 6.1 als auch Zusammenhang mit Abb. 1.25 angesprochen: die Selbstähnlichkeit.

<sup>3</sup>Falls Sie Literaturrecherche betreiben wollen: die Mandelbrot Menge ist ein Spezialfall einer Julia Menge, d.h. Sie sollten unter letzterem Stichwort suchen.



Abbildung 6.4: Schneeflocken [81]; viele weitere schöne Bilder auch unter <http://www.snowcrystals.com/>

Angefangen hat alles, zumindest der Legende nach, mit einem einfachen Messproblem: bestimme die Küstenlänge Großbritanniens. Das Ergebnis hängt von der Methode ab: messen in einem Weltatlas liefert auf Grund des großen Maßstabs von ca. 1:4 Mio eine recht kurze Küstenlänge. Auf einer Autokarte mit Maßstab 1:250 000 sind bereits mehr Details zu erkennen, entsprechend variiert ist die Küsten – und damit länger. Und auf einer topographischen Grundkarte im Maßstab 1:10 000 ist die Küste noch länger. Auch in der moderneren Variante, dem Satellitenbild, bleibt dieses Problem bestehen, vgl. Abb. 6.3 – mit einem Satelliten mit gutem Bodenaufklärungsvermögen ergibt sich ein größerer Wert für die Küstenlänge als mit einem alten Satelliten mit schlechtem Auflösungsvermögen.

§ 656 Also ohne Karten mit Landvermessers. Normalerweise peilen Landvermesser. Damit nähern sie die Küstenlinie stückweise linear an – die sich ergebende Gesamtlänge nimmt mit abnehmender Länge der Peilstrecken zu. Immer noch zu ungenau, daher wird Messung mit konventionellen Methoden, also einem Maßstab, angeordnet. Das Problem der Peilstrecke bleibt erhalten: auch dieses Verfahren liefert stückweise Diskretisierung. Allerdings werden jetzt nicht mehr, wie beim Peilverfahren, kleinere Buchten übersehen. Stattdessen beginnen die Landvermesser, jede Ausbeulung durch einen halb im Wasser liegenden Felsen mit zu vermessen.

§ 657 Eine Eigenschaften der Küstenlinie führt zu diesem Problem: sie ist selbstähnlich. Selbstähnlichkeit bedeutet, dass sich Strukturen auf verschiedenen Größenskalen immer wieder holen: Buchten und Landvorspünge sehen wir auf den unterschiedlichsten räumlichen Skalen. Da sind die Buchten der großen Flüsse und die Halbinseln bei Betrachtung der ganzen Insel; da sind die Einschnitte der Bäche und die kleinen Landzungen bei Betrachtung einer Bucht oder Halbinsel; da sind die Vorwölbungen von Felsen ins Wasser und die kleinen Einschnitte zwischen ihnen. Wir können dies bis auf die Ebene des Sandkorns fortsetzen.

§ 658 Selbstähnlichkeit ist ein verbreitetes Phänomen in der Natur. Ähnlich offensichtlich, jedoch auf deutlich kleinerem Maßstab, tritt sie bei Schneeflocken auf, siehe Abb. 6.4. Insbesondere dendritische Flocken zeigen eine hohe fraktale Dimension, bei eher hexagonalen Flocken (wie im Beispiel rechts unten) ist diese eher gering.

§ 659 Die Koch'sche Schneeflocke, 1904 von H. von Koch [72] eingeführt, kann als erster Einstieg in die mathematische Beschreibung von Fraktalen verwendet werden.<sup>4</sup> Geometrisch wird sie aus einem gleichseitigen Dreieck mit Seitenlänge  $a$  konstruiert, siehe obere Zeile in Abb. 1.25. Selbstähnlichkeit ist das Leitmotiv, d.h. auf der jeweils kleineren Skala soll wieder ein gleichseitiges Dreieck auftreten. Also Dritteln wir jede Seite und ersetzen das mittlere Drittel durch ein gleichseitiges Dreieck dessen Spitze nach außen weist. Dieser Vorgang wird immer wieder wiederholt. Wie groß ist der Umfang der Schneeflocke?

§ 660 Da alle drei Seiten der Koch'schen Schneeflocke abgesehen von ihrer Orientierung identisch sind, reicht es, eine Seite zu betrachten. Dieser Initiator wird mit einem Kochrezept, dem Generator, verändert. Aus der einen Strecke sind jetzt 4 Strecken geworden, allerdings mit einer um den Faktor  $1/3$  reduzierten Länge. Die Länge der Ausgangsstrecke ist um den Faktor  $4/3$  vergrößert:  $L_1 = \frac{4}{3}a$ . Jede der neuen Teilstrecken kann als Initiator des nächsten Schritts verwendet werden. Nach Schritt 2 ist die Länge  $L_2 = a(4/3)^2$ . Allgemein erhalten wir damit für den Umfang  $U_k$  nach dem  $k^{\text{ten}}$  Schritt

$$\frac{1}{3}U_k = L_k = \frac{4}{3}L_{k-1} = \left(\frac{4}{3}\right)^k a. \quad (6.1)$$

Da in dieser Reihe jeder Wert um einen festen Faktor größer ist als der voran gegangene, konvergiert diese Reihe nicht.

§ 661 Setzen wir die Konstruktion weiter fort, so sind die Kurven ab einem bestimmten  $k$  kaum noch zu unterscheiden, da die mit jedem Schritt entstehenden Änderungen unter die Sichtbarkeitsgrenze fallen – bei einem schlechten Drucker erscheint das vielleicht als etwas ausgefranste und verbreiterte Linie.

§ 662 Zur Charakterisierung eines Fraktals wird die Hausdorff Dimension oder fraktale Dimension verwendet. Diese ist bestimmt durch die Zahl  $N$  der identischen Objekte, die mit jedem Schritt entsteht, und durch den Skalierungsfaktor  $s$ , um den das Objekt verkleinert wird. Für die Koch'sche Schneeflocke ist  $N = 4$  und  $s = 3$ : aus einer Geraden werden  $N = 4$  Geraden, allerdings beträgt die Länge dieser Geradenstücke nur jeweils  $\frac{1}{s} = \frac{1}{3}$ . In einer selbstähnlichen Struktur ist der Zusammenhang zwischen den beiden Parametern durch ein Potenzgesetz

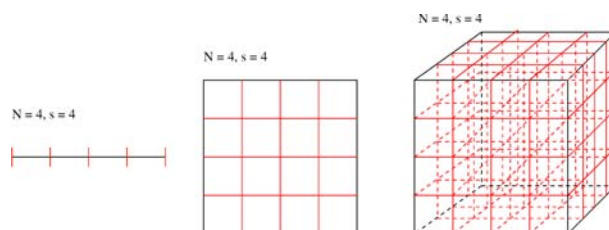
$$N = s^D$$

gegeben mit  $D$  als der Hausdorff Dimension. Für die Koch'sche Schneeflocke ergibt sich aus diesem Skalierungsgesetz eine fraktale Dimension  $D = 1.262$ .<sup>5</sup>

§ 663 Ein anderes bekanntes Fraktal ist das Sierpinski Dreieck, siehe dritte Zeile in Abb. 1.25. Hier werden in jedem Schritt drei identische Objekte erzeugt, d.h.  $N = 3$ . Diese sind mit dem Faktor  $s = 2$  skaliert, so dass sich eine Hausdorff Dimension  $D = 1.585$  ergibt.

<sup>4</sup>Koch hat bei seiner Arbeit weniger an Fraktale gedacht als vielmehr an die Konstruktion einer stetigen Kurve, die an keiner Stelle differenzierbar ist: führt man die Iteration unendlich oft aus, so ist jeder Punkt ein Eckpunkt, d.h. die Kurve ist an der Stelle nicht differenzierbar, da es keine eindeutige Tangente gibt. Trotzdem ist sie stetig, da sie in einem Zug durchgezeichnet werden kann. Eingeführt (oder zumindest verbreitet) wurde der Begriff Fraktal erst durch die Arbeiten Mandelbrots.

<sup>5</sup>Bei den meisten Objekten der klassischen Geometrie stimmen euklidische und Hausdorff Dimension überein. So gilt für eine Strecke bei  $N = 4$  auch  $s = 4$  und damit  $N = s^D = s^1$ , also  $D = 1$ . Bei einem Quadrat ergibt sich bei  $N = 16$  ein Skalierungsfaktor von  $s = 4$  und damit eine Hausdorff Dimension  $D = 2$ . Beim Würfel ergibt sich bei  $N = 64$  ein Skalierungsfaktor  $s = 4$  und damit  $D = 3$ :





§ 664 Wie die Schneeflocke bereits nahe legt, bieten Fraktale einfache Verfahren zur Erzeugung selbstähnlicher Strukturen wie Schneeflocken, Farne, Flussdeltas, Küstenlinien, Gebirgszüge usw.<sup>6</sup> Benoit Mandelbrot hat sich in seinem Buch *The fractal geometry of nature* [91] ausführlich mit diesem Thema beschäftigt. Schöne Beispiele finden Sie u.a. in Peitgen und Richter [116]; eine ausführliche Darstellung über Fraktale und ihren Bezug zum Chaos geben die beiden Bände von Peitgen et al. [118, 119]. Mit der Verwendung der Selbstähnlichkeit zur Kompression der in einem Bild enthaltenen Information wird ebenfalls experimentiert, siehe z.B. [29]. Eine sich von den algebraischen Ansätzen entfernende, auf Beschreibungen durch Differentialgleichungen beruhende Darstellung von Fraktalen gibt Strichartz [155].

## 6.2 (Seltsame) Attraktoren und Abbildungen

§ 665 Im Zusammenhang mit dem Feigenbaum-Diagramm in Abb. 6.1 haben wir den Begriff des Attraktors verwendet. In diesem Abschnitt wollen wir den Begriff des Attraktors etwas genauer betrachten und die typischen Abbildungen einführen.

### 6.2.1 Attraktoren

**Definition 9** *Ein Attraktor ist ein geometrisches Objekt im Phasenraum, in dem die Trajektorien von verschiedenen Anfangspunkten konvergieren.*

§ 666 Die obige Definition des Attraktors betrachten wir für das einfachste geometrische Objekt, den Punkt. Die Definition des Punkt-Attraktors lautet entsprechend:

**Definition 10** *Ein Punkt-Attraktor ist ein Punkt im Phasenraum, in dem die Trajektorien von verschiedenen Anfangspunkten konvergieren.*

Diese Definition können wir uns (mal wieder) mit Hilfe des einfachen harmonischen Oszillators veranschaulichen. Betrachten wir den gedämpften Oszillator, so konvergieren alle Trajektorien im Phasenraum auf den Ursprung. Dieser ist ein Punkt-Attraktor oder Fixpunkt. Er wird teilweise auch als Gleichgewichtspunkt bezeichnet: das Teilchen bewegt sich in einem parabolischen Potential. Dissipation bewirkt, dass es früher oder später zur Ruhe kommt.

§ 667 Das Potential können wir auch verwenden, um uns eine Möglichkeit für das Entstehen einer Bifurkation zu veranschaulichen: bewegt sich das Teilchen in einem Potential mit zwei Minima (im einfachsten Fall ein  $\omega$ -ähnliches Potential), so gibt es beim dissipativen System zwei Attraktoren bzw. Fixpunkte: in welchem der beiden Minima das Teilchen landet, kann durch kleine Abweichungen in den Anfangsbedingungen entschieden werden.

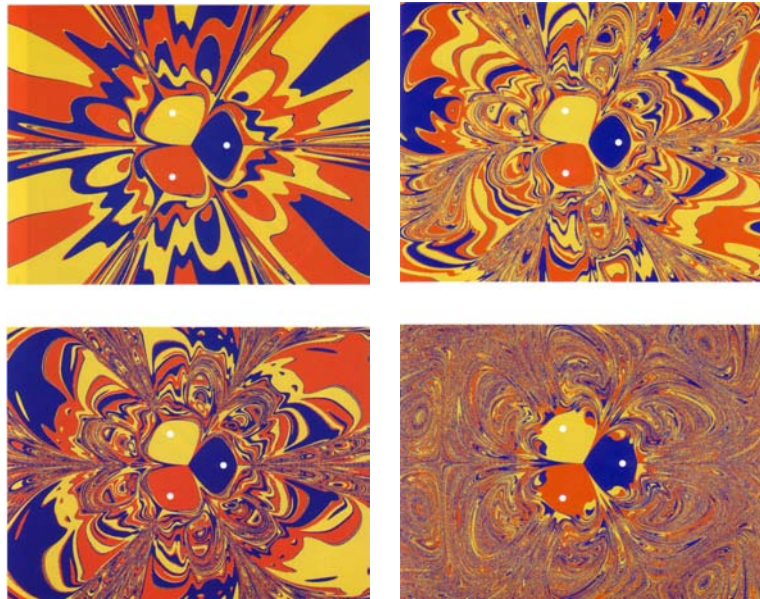
**Definition 11** *Das Attraktor-Basin (basin of attraction) ist die Menge aller Anfangsbedingungen im Phasenraum, für die die Trajektorie des Systems sich einem bestimmten Attraktor annähert. Die Grenzen zwischen den zu verschiedenen Attraktoren führenden Bereichen können fraktale Dimension haben.*

§ 668 Für ein einfaches gedämpftes Pendel ist der Bereich, aus dem die Trajektorien sich dem Gleichgewichtspunkt annähern, der gesamte Phasenraum.

§ 669 Betrachten wir dagegen ein angeregtes Pendel. Dessen Bahn im Phasenraum wird nach dem Einschwingen eine Ellipse. Auch hier gilt: unabhängig von den Anfangsbedingungen, wird stets die gleiche Ellipse als Trajektorie eingenommen, d.h. die Trajektorien aus dem gesamten Phasenraum konvergieren auf diese Ellipse. Die Ellipse ist ein Attraktor der Dimension Eins, da sie durch eine eindimensionale Kurve gebildet wird. Da die beschriebene Bewegung periodisch ist, wird der Attraktor auch als limit cycle bezeichnet:

<sup>6</sup>Den Initiator der Schneeflocke hat bereits Koch selbst mit einem etwas anderen Generator behandelt, um daraus einen Bienenkorb zu erzeugen, vgl. zweite Zeile in Abb. 1.25.

Abbildung 6.5: Magnetpendel: die drei Gleichgewichtspunkte sind jeweils als weiße Punkte markiert; die Farben markieren, in welchem Gleichgewichtspunkt ein Pendel landet, wenn es mit verschwindender Anfangsgeschwindigkeit von diesem Punkt gestartet wird [159]; die Teilabbildungen unterscheiden sich im Reibungskoeffizienten, der Frequenz der freien Schwingung und dem Abstand des Pendels über der Ebene



**Definition 12** Ein limit cycle ist ein Attraktor der Dimension eins; er beschreibt eine periodische Bewegung.

**Verständnisfrage 65** Und wie sieht es mit der freien Schwingung aus? Gibt es einen Attraktor oder nicht? Wie groß oder wodurch ist der Bereich definiert, aus dem sich die Trajektorien dem Attraktor annähern?

**Definition 13** Ein seltsamer Attraktor (*strange attractor*) tritt in chaotischen Systemen auf und hat eine fraktale Dimension.

§ 670 Ein einfaches chaotisches System ist ein Magnetpendel. Dieses besteht aus einem Magneten an einem Fadenpendel; auf der Grundfläche unter dem Pendel befinden sich mehrere Magnete. Die Polaritäten sind so gewählt, dass diese Magnete den Pendelkörper jeweils anziehen. Die Bewegung des Fadenpendels ist damit nicht wie beim konventionellen Aufbau auf eine Ebene beschränkt sondern erfolgt im 3D. Im Gegensatz zum Fadenpendel hat das Magnetpendel nicht nur einen Gleichgewichtspunkt; stattdessen ist die Zahl der Gleichgewichtspunkte gleich der Zahl der Magneten. Abbildung 6.5 illustriert die Gleichgewichtspunkte und die Attraktorbereiche für ein durch drei Magnete (jeweils die weißen Punkte als Markierung der zugehörigen Gleichgewichtszustände). In SIMULINK-Spielerei 20 können Sie mit diesem Problem weiter spielen.

## 6.2.2 Logistische Gleichung

§ 671 Die logistische Gleichung ist eines der einfachsten Modelle für eine in das Chaos führende Folge:

$$x_{n+1} = \alpha(x_n - x_n^2).$$

. In Abb. 6.1 haben wir die Entwicklung dieser Folge bereits für den Anfangswert in  $x_0 = 0.5$  in Abhängigkeit vom Parameter  $\alpha$  betrachtet. In Abb. 6.6 sind zur Veranschaulichung die ersten 100 Glieder der Folge aufgetragen für einen Anfangswert  $x_0 = 0.5$  und verschiedene Werte von  $\alpha$ . Für  $\alpha = 2$  (oberes Teilbild) nimmt die Folge sofort den konstanten Wert  $x_\infty = 0.5$  an. Ein Schnitt bei  $\alpha = 2$  in Abb. 6.1 liefert ebenfalls diesen Wert.

§ 672 Gehen wir in Abb. 6.1 zu größeren Werten von  $\alpha$ , so gelangen wir in der Nähe von drei zur ersten Bifurkation. Ab hier hat die Folge keinen Grenzwert mehr sondern zwei

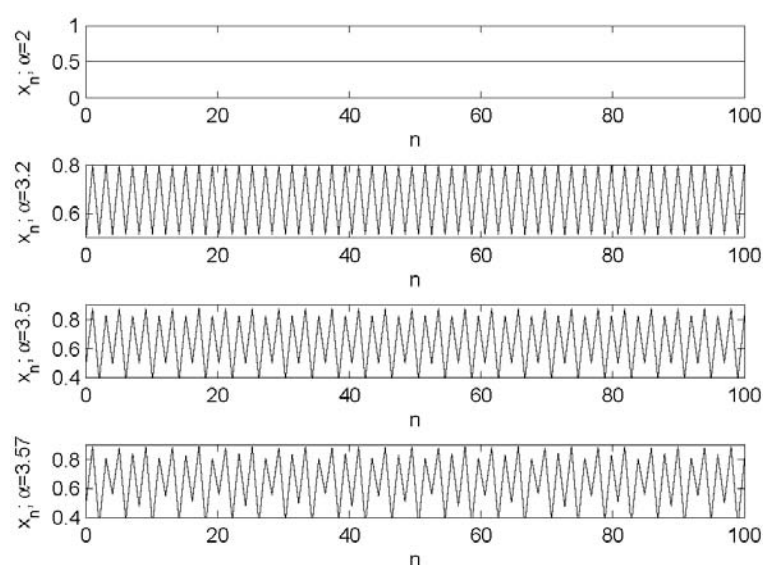


Abbildung 6.6: Die ersten 100 Glieder der durch die logistische Gleichung bestimmten Folge mit Anfangswert  $x_0 = 0.5$  für verschiedene  $\alpha$

Häufungspunkte zwischen denen sie hin- und her springt. Als Beispiel sind im zweiten Teilbild von Abb. 6.6 die ersten 100 Folgenglieder gezeigt. Hinter der zweiten Bifurkation ergeben sich für die Folge vier Häufungspunkte; das dritte Teilbild in Abb. 6.6 illustriert wieder den Verlauf.

§ 673 Das Schema setzt sich fort, bis sich bei  $\alpha = 4$  die volle logistische Gleichung ergibt: hier ist das Verhalten der Folge chaotisch, wie im unteren Teilbild in Abb. 6.6 angedeutet – mit einer Einschränkung: für  $x_0 = 0.5$  nehmen die höheren Glieder der Folge den Wert Null an. Die Null ist zwar ein Fixpunkt, allerdings konvergieren die Folgeglieder für andere Anfangswerte nicht gegen Null, da die Null keinem stabilen Gleichgewicht entspricht. Das wird auch im unteren Teilbild von Abb. 6.6 deutlich: die Folge kann, manchmal sogar für mehrere Folgeglieder in Folge sehr kleine Werte annehmen, um dann auf Werte knapp unterhalb von Eins zu springen.

**Zwischenrechnung 45** Spielen Sie mit der logistischen Folge. Untersuchen Sie die Abhängigkeit des Folgenverhaltens vom Anfangswert – jeweils durch explizite Betrachtung der Folgeglieder und mit Hilfe eines Feigenbaum-Diagramms.

### 6.2.3 Lorenz-Attraktor

§ 674 Der Schmetterlingseffekt (siehe auch Abschn. 9.6) ist ein auch in der populären Diskussion häufig verwendetes Beispiel für die Kopplung von Chaos und (Nicht-)Vorhersagbarkeit.<sup>7</sup> Er geht zurück auf Ed Lorenz [85], der sich am MIT mit numerischer Wettervorhersage befasst hat und dabei über die empfindliche Abhängigkeit der Lösung von den Anfangsbedingungen gestolpert ist.

§ 675 Lorenz ist dabei von einem sehr stark vereinfachten Satz von Gleichungen zur Modellierung der thermischen Konvektion (Benard Konvektion) ausgegangen. Lorenz hat das Geschwindigkeitsfeld  $\vec{v}(\vec{r}, t)$  und das Temperaturfeld  $T(\vec{r}, t)$  vorgegeben. Auf der Basis der Navier–Stokes-Gleichung, der Kontinuitätsgleichung und der Wärmeleitungsgleichung konnte Lorenz die folgenden dimensionslosen Gleichungen für die Trajektorie im Phasenraum aufstellen:

$$\dot{x}_1 = \sigma(x_2 - x_1),$$

<sup>7</sup>Eric Larson macht in seinem Roman Isaacs Sturm [78] über den Calveston zerstörenden Hurrikan ausführlich Gebrauch davon, Bradbury hat das Konzept wie im Eingangszitat angedeutet bereits verwendet, als es noch gar nicht bekannt war.

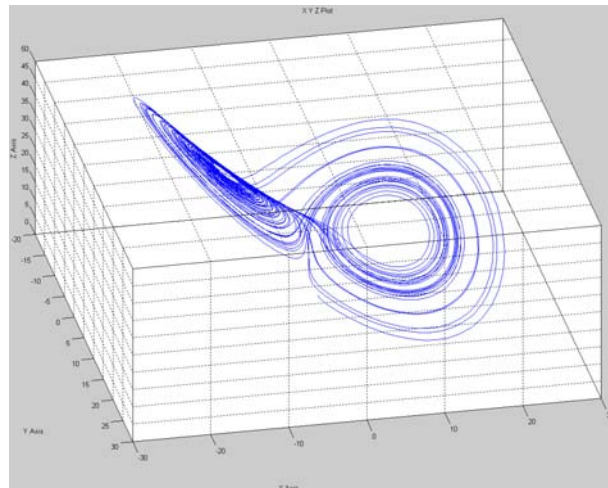


Abbildung 6.7: Lorenz-Attraktor

$$\begin{aligned}\dot{x}_2 &= (\varrho - x_3)x_1 - x_2, \\ \dot{x}_3 &= x_1x_2 - \beta x_3.\end{aligned}$$

In diesen Lorenz Gleichungen sind die Variablen  $x_1$ ,  $x_2$  und  $x_3$  mit der Stromfunktion und der Temperaturvariation verbunden;  $\sigma$  ist die Prandtl-Zahl,  $\beta$  ein geometrischer Parameter, der die Größe der Konvektionsrollen beschreibt, und  $\varrho$  ist die normierte Rayleigh-Zahl, die dem Temperaturgradienten proportional ist.

§ 676 Das System ist dissipativ: alle Trajektorien werden daher über lange Zeiträume einem Gleichgewichtszustand zustreben. Aber auch auf kurzen Zeiträumen ist das System interessant: die Trajektorie ist, wie in Abb. 6.7 auf einen kleinen Teil des Phasenraums beschränkt, in diesem jedoch erratisch. Verfolgt man die Trajektorie, so erkennt man, dass diese erst etliche Schleifen in einem Teil der Abbildung macht und dann für mehrere Durchläufe in den anderen Teil der Abbildung springt. Außerdem ist die Trajektorie (und insbesondere der Wechsel der Bahnen zwischen den beiden Teilen der Abbildung) stark von den Anfangsbedingungen abhängig.

§ 677 Die Lorenz-Gleichungen (und damit auch der Lorenz-Attraktor) sind insofern ein interessantes Modellsystem, als dass sie sich auf viele andere physikalische Probleme übertragen lassen; Beispiele sind in [75] gegeben.

### 6.2.4 Attraktoren in SIMULINK

§ 678 Die logistische Gleichung (ebenso wie die die Julia- und Mandelbrotmenge beschreibenden Gleichungen) sind Regeln zur Bildung von Folgengliedern – untersucht wird nur die Konvergenz der Folge für bestimmte Parameter. Damit besteht bei der Darstellung derartiger Probleme in SIMULINK ein großes Risiko, im Modell eine algebraische Schleife zu erzeugen.

§ 679 Ein interessantes Paket von SIMULINK-Modellen zur Darstellung von Attraktoren ist als `attractors` von Giampiero Campa<sup>8</sup> zusammen gefasst worden und kann unter <http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=7939&objectType=file> herunter geladen werden; Abb. 6.7 ist ein Beispiel für ein mit diesem SIMULINK-Modell erzeugtes Ergebnis.

<sup>8</sup>Für den Fall, dass Ihnen die Behandlung von Matrizen in SIMULINK etwas zu aufwendig ist, können die vom gleichen Autor gebastelten und unter <http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=266&objectType=FILE> zu findenden Blöcke gegebenenfalls hilfreich sein.

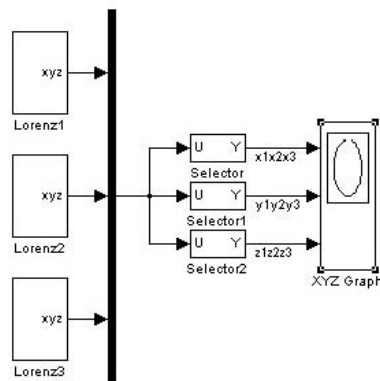


Abbildung 6.8: SIMULINK-Modell zur Darstellung des Lorenz-Attraktors  
<http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=266&objectType=FILE>

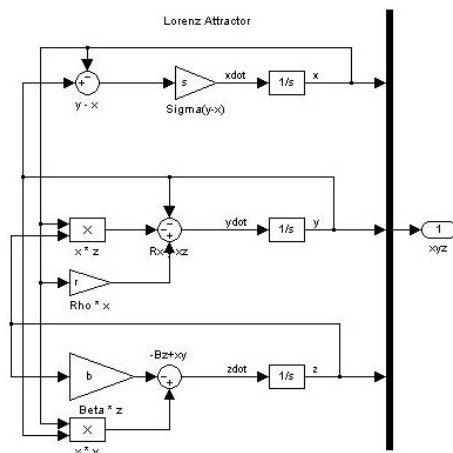


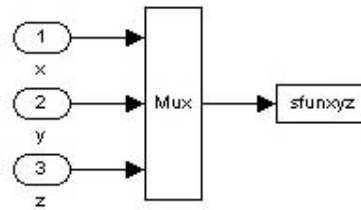
Abbildung 6.9: Unterprogramm aus Abb. 6.8  
<http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=266&objectType=FILE>

§ 680 Als Beispiel sei hier das Modell zur Darstellung des Lorenz-Attraktors betrachtet. Das Modell erlaubt es, dem Lorenz-Attraktor bei seinem Wachstum zu zusehen. Abbildung 6.8 gibt einen Überblick über das Modell: die linken drei Blöcke **Lorenz1** bis **Lorenz 3** sind maskierte Unter Systeme und lösen die jeweiligen Systemgleichungen. Der vektorielle Output wird über einen Multiplexer in einen entsprechend langen Vektor geschrieben. Aus diesem Vektor werden mit Hilfe der **Selector**-Blöcke drei Komponentensätze ausgewählt und auf das XYZ-Oszilloskop gegeben. Während der **Selector** ein konventioneller, unter der Rubrik Signal Routing zu findender SIMULINK-Block ist, ist der XYZ-Graph eine selbst geschriebene s-Function. Die Lösung erfolgt mit Hilfe von `ode45` in den Default-Einstellungen.

§ 681 Betrachten wir die Modellteile einzeln. Die eigentlichen Gleichungen werden in den Blöcken **Lorenz** gelöst. Diese Blöcke sind jeweils maskierte Unter Systeme, in den zu den Blöcken gehörigen Karteikarten (Doppelklick auf den Block) lassen sich neben den Parametern  $\sigma$ ,  $\rho$  und  $\beta$  auch die Anfangsbedingungen eintragen. In allen drei maskierten Unter Systemen sind die Default-Werte von  $\sigma = 10$ ,  $\beta = 8/3$  und  $\rho = 28$  eingetragen. Die drei Unter Systeme sind in ihrem Aufbau identisch (sie lösen die Lorenz Gleichungen) und unterscheiden sich lediglich in den Anfangsbedingungen. Um einen Lorenz-Attraktor zu betrachten wäre ein Block mit direkter Ausgabe auf das XYZ-Oszilloskop ausreichend; die Verwendung von drei Blöcken erlaubt es, die Abhängigkeit von verschiedenen Anfangsbedingungen zu zeigen.

§ 682 Abbildung 6.9 zeigt einen Blick unter die Maske des maskierten Unter Systems. Dieser lässt sich durch Rechtsklick auf den Block und Auswahl des Menüpunktes Look under Mask erreichen. Die Lorenz-Gleichungen sind explizit über die einzelnen Komponenten dargestellt. Die einzigen benötigten Blöcke sind die Integratoren, die Gain-Faktoren und die Summationsblöcke sowie der Produkt-Block. Die Verknüpfung der Gleichungen erfolgt bei

Abbildung 6.10:  
 3D-Oszilloskop als S-Function  
<http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=266&objectType=FILE>



der Rückführung des Signals – daher erfordern die Verbindungen links etwas Konzentration. Im rechten Teil werden die Lösungen der einzelnen skalaren Gleichungen des Systems über den Multiplexer zu einem Vektor zusammengefasst. Dieser wird über drei parallele Selektoren in drei Dreier-Vektoren zerpfückt, die jeweils die  $x$ , die  $y$  und die  $z$ -Komponenten für die drei Anfangsbedingungen enthalten.

§ 683 Die 3D-Darstellung des Lorenz-Attraktors erfolgt mit Hilfe eines selbst gebastelten XYZ-Oszilloskop. Als Eingangsgröße dienen jeweils drei Dreiervektoren: der erste enthält die  $x$ -Komponente der Trajektorie für die drei Anfangsbedingungen, der zweite die  $y$ - und der dritte die entsprechende  $z$ -Komponente. Ein Blick unter die Maske (siehe Abb. 6.10) erläutert die Struktur: die drei Vektoren werden wieder über einen Multiplexer zu einem Vektor kombiniert und dann mit Hilfe einer s-Function darstellen.

#### s-Function in SIMULINK

§ 684 Der Block s-Function ermöglicht die Erstellung einer vom Nutzer definierten Funktion: **S-Function**: User-definable block. Blocks can be written in C, M (level-1), Fortran, and Ada and must conform to S-function standards. The variables  $t$ ,  $x$ ,  $u$ , and  $flag$  are automatically passed to the S-function by Simulink. You can specify additional parameters in the 'S-function parameters' field. If the S-function block requires additional source files for the Real-Time Workshop build process, specify the filenames in the 'S-function modules' field. Enter the filenames only; do not use extensions or full pathnames, e.g., enter 'src src1', not 'src.c src1.c'.

Im Gegensatz zu den Blöcken **Fcn** und **MatLab Fcn** handelt es sich hierbei nicht um die Definition einer mathematischen Funktion sondern letztendlich um ein Unterprogramm oder eine Methode. Diese muss nicht zwingend in MATLAB geschrieben sein sondern kann auch in C, Fortran oder Ada erstellt sein. SIMULINK stellt auch einen S-Function Builder zur Verfügung, der das Schreiben derartiger Funktionen erleichtert.

## 6.3 Variationen über ein einzelnes Pendel

§ 685 In diesem Abschnitt werden wir Variationen über ein einzelnes Fadenpendel betrachten: im Feder-Fadenpendel wird der Faden durch eine Feder ersetzt, im Fadenpendel mit Rolle ändert sich die Länge des Fadens in Abhängigkeit von der Bewegung des Pendels. Beide Systeme sind mechanisch einfach, erlauben es jedoch, auf das Poincare-Portrait als wichtige Form der Darstellung chaotischer Systeme hinzuweisen. Die Anregung für beide Modelle stammt aus [159].

### 6.3.1 Das Feder-Fadenpendel

§ 686 Das Feder-Fadenpendel ist eine Variante des Fadenpendels mit veränderlicher Fadenlänge für beliebige Auslenkungen (also der nichtlinearen Bewegungsgleichung). Hier ist der Faden durch eine Feder ersetzt. Im Gegensatz zur vorgegebenen Änderung der Fadenlänge als  $l(t)$  ändert sich hier die Federlänge in Abhängigkeit von der Auslenkung, d.h. vom Ort, an dem sich die Masse befindet.

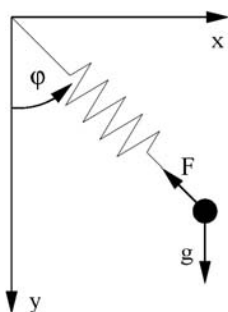


Abbildung 6.11:  
Fadenpendel mit  
Feder statt Faden

### Feder statt Faden

§ 687 Betrachten wir nun statt des gewöhnlichen Fadenpendels aus Abschn. 1.1.5 ein Fadenpendel, dessen Faden durch eine Feder mit der Ruhelänge  $l_0$  und einer Federkonstante  $k_0$  entsprechend einer Frequenz der freien Schwingung von  $\omega_0$  für die gegebene Pendelmasse  $m$ . Die Geometrie kann auch als Näherung für eine Masse an einem Gummiband dienen. Wie beim normalen nicht-linearen Fadenpendel erlauben wir Überschläge.

§ 688 Der Ursprung des Koordinatensystems liegt in der Aufhängung des Pendels, vgl. Abb. 6.11. Die momentane Länge  $l(t)$  der Feder ist durch den Ort  $(x(t), y(t))$  des Massenpunktes gegeben:  $l = \sqrt{x^2 + y^2}$ . Damit wird von der Feder eine Kraft  $F = k_0(l - l_0) = k_0(\sqrt{x^2 + y^2} - l_0)$  auf die Masse ausgeübt mit den Kraftkomponenten  $-Fx/l$  und  $-yF/l$  in kartesischen Koordinaten. Da die Gravitationskraft nur entlang der  $y$ -Achse wirkt, ergibt sich für die Bewegungsgleichung in kartesischen Koordinaten:

$$\ddot{x} + \omega_0^2 x \left(1 - \frac{l_0}{\sqrt{x^2 + y^2}}\right) = 0 \quad \text{und} \quad \ddot{y} + \omega_0^2 y \left(1 - \frac{l_0}{\sqrt{x^2 + y^2}}\right) - g = 0.$$

Alternativ lässt sich die Bewegung in Polarkoordinaten mit der Auslenkung  $\varphi$  und der Pendellänge  $l$  schreiben als

$$\ddot{l} = l\dot{\varphi}^2 - \omega_0^2(l - l_0) + g \cos \varphi \quad \text{und} \quad l\ddot{\varphi} = -2\dot{l}\dot{\varphi} - g \sin \varphi.$$

§ 689 Drücken wir die Länge in Einheiten der Ruhelänge  $l_0$  der unausgelenkten Feder und die Zeit in Einheiten von  $1/\omega_0$  aus, so ergibt sich als dimensionslose Gleichung in kartesischen Koordinaten

$$\ddot{x} + x \left(1 - \frac{1}{\sqrt{x^2 + y^2}}\right) = 0 \quad \text{und} \quad \ddot{y} + y \left(1 - \frac{1}{\sqrt{x^2 + y^2}}\right) - q = 0$$

bzw. in Polarkoordinaten

$$\ddot{l} = l\dot{\varphi}^2 - (l - 1) + q \cos \varphi \quad \text{und} \quad l\ddot{\varphi} = -2\dot{l}\dot{\varphi} - q \sin \varphi.$$

§ 690 Darin ist

$$q = \frac{g}{\omega_0^2 l}$$

der einzige dimensionslose Parameter des Systems.<sup>9</sup> Diese Größe beschreibt das Verhältnis der Quadrate der Frequenzen des Faden- und des Federpendels. Für ein Verhältnis in der Nähe

<sup>9</sup>Dimensionslose Größen und Gleichungen sind ein einfaches Verfahren, die Menge von Lösungen zu ordnen. Bereits im ersten Semester begegnen Sie Ihnen: in Abb. 1.2 betrachten wir das Resonanzverhalten bei einer erzwungenen Schwingung nicht in Abhängigkeit von der Frequenz der anregenden Schwingung sondern in Abhängigkeit von deren Frequenz in Einheiten der Frequenz der freien Schwingung. Damit ergibt sich Resonanz immer in der Nähe von Eins und für Werte kleiner/größer Eins kann man die Frequenz der Anregung beliebig kleiner/größer machen, ohne in das Risiko einer Resonanz zu laufen. Um eine Gleichung dimensionslos zu machen, dividiert man durch die charakteristischen Skalen: bei der Zeit sind dies in der Regel Schwingungsdauern (und damit Frequenzen) oder sonstige charakteristische Zeiten des Systems (z.B.

Abbildung 6.12: Die Bewegung des Feder-Fadenpendels aus Abb. 6.11 lässt sich mit Hilfe des dimensionslosen Parameters  $q$  charakterisieren: für kleine  $q$  ist die Lösung vom Fadenpendel dominiert, für große  $q$  vom Federpendel. In der Nähe von  $q = 1$  wird die Bewegung chaotisch. Anfangsbedingungen  $\dot{x} = \dot{y} = 0$  und  $x = 0.1$ ,  $y = -1$

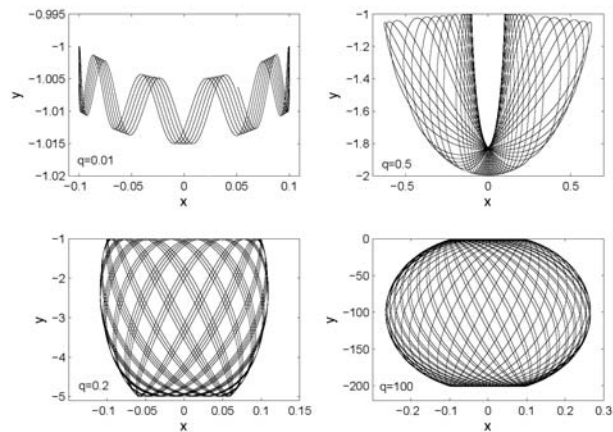
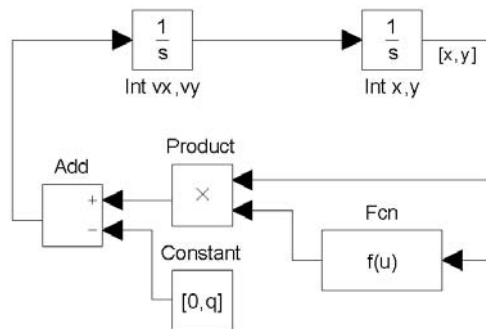


Abbildung 6.13: Modell des Feder-Fadenpendels in SIMULINK



von 1 sind im Gesamtsystem die Eigenschaften von Feder- und Fadenpendel zu ungefähr gleichen Anteilen vertreten – das Chaos sollte in diesem Bereich am größten werden. Ist dagegen  $q$  sehr klein oder sehr groß, so betrachten wir im wesentlichen das Federpendel mit einer kleinen überlagerten Störung durch das Fadenpendel bzw. umgekehrt, vgl. Abb. 6.12.

§ 691 Die beiden Differentialgleichungen für  $x$  und  $y$  (bzw. für  $l$  und  $\varphi$ ) sind gekoppelt, eine einfache vektorielle Schreibweise bietet sich jedoch nicht offensichtlich an. Wir können das Modell in SIMULINK aber dennoch vektoriell aufbauen, vgl. Abb. 6.13. Lediglich in der Funktion  $f(u)$  werden die beiden Komponenten ‘gemischt’ nicht wie konventionell mit Hilfe der Matrix sondern durch explizite Vereinbarung der Funktion als  $1/\sqrt{u(1)*u(1)+u(2)*u(2)}-1$ . Dieser Faktor wird im Block Produkt mit jeder Komponente des Vektors  $[x, y]$  multipliziert; der Parameter  $q$  wird als Vektor definiert, da er nur zu  $y$ -Komponente zu addieren ist.

§ 692 Für  $q$  in der Nähe von 1 sind beide Schwingungsformen ungefähr gleichberechtigt. Dass dort chaotische Bewegungen vorherrschen, ist nicht verwunderlich und bereits in Abb. 6.12 angedeutet. Mit zunehmender Gesamtenergie werden die Bewegungen noch komplexer, vgl. Abb. 6.14. In der Abbildung ist auch zu erkennen, dass chaotische Bewegungen keinesfalls auf  $q \sim 1$  beschränkt sind sondern auch bei kleineren und größeren Werten auftreten.

Zerfallskonstanten); bei der Länge dividiert man durch die charakteristische räumliche Skala, z.B. eine mittlere freie Weglänge oder die Wellenlänge/Wellenzahl.

In der Hydrodynamik begegnet uns recht früh die Reynolds-Zahl als eine dimensionslose Größe. Wir verwenden sie, um zu entscheiden, ob eine Strömung laminar ist oder turbulent. Physikalisch ist die Reynolds-Zahl das Verhältnis aus Trägheits- und Reibungskraft: deutlich unterhalb einer kritischen Reynoldszahl gilt die Strömung als laminar, deutlich darüber als turbulent. Das hier definierte Verhältnis  $q$  hat eine ähnliche Bedeutung als kritische Größe.

Dimensionslose Gleichungen, kritische Größen und Ähnlichkeitszahlen wie die Reynoldszahl sind zwar insbesondere in der Hydromechanik gebräuchlich; es handelt sich aber um allgemeine Verfahren zur Charakterisierung von Systemen und zur Loslösung von der Betrachtung einzelner Lösungen individueller Systeme. Stattdessen werden allgemeine Lösungen in Einheiten der das System charakterisierenden Parameter gesucht.



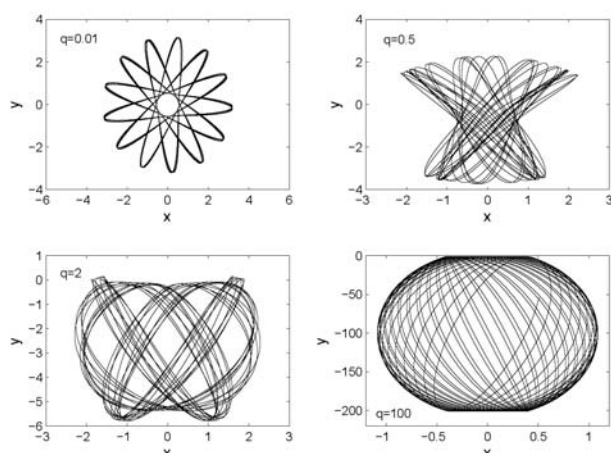


Abbildung 6.14: Wie Abb. 6.12 jedoch mit Anfangsbedingungen  $\dot{x} = 1$ ,  $\dot{y} = 2$ ,  $x = 0.1$  und  $y = -1$

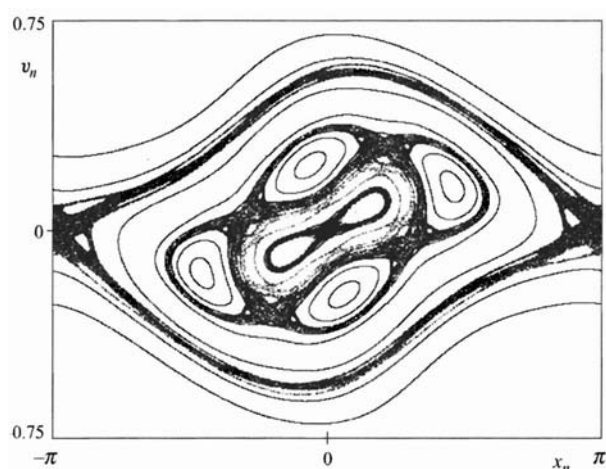


Abbildung 6.15: Poincaré Portrait mit  $x_n = \varphi_n$  und  $v_n = \dot{\varphi}_n$  für ein Feder-Fadenpendel für 11 verschiedene Anfangsbedingungen unter der Annahme einer konstanten (dimensionslosen) Gesamtenergie  $\epsilon = \dot{l}^2/2 + l^2\dot{\varphi}^2/2 - ql \cos \varphi + (l-1)^2/2 = 0.195$  für  $q = 0.07$  [159]

§ 693 Abbildung 6.15 zeigt einen Poincaré Schnitt ( $\rightarrow$  Abschn. ??) für  $q = 0.07$ , d.h. den Bereich, in dem die Eigenschaften des Fadenpendels dominieren. Dargestellt ist die Winkelgeschwindigkeit  $v_n = \dot{\varphi}_n$  gegen die Auslenkung im Winkel,  $x_n = \varphi_n$ ; jedoch nicht, wie in der konventionellen Bahn im Phasenraum für alle Punkte der Trajektorie sondern nur für die, bei denen die Länge des Pendels gleich der Ruhelänge ist:  $l = l_0$ . Daher die Bezeichnung Poincaré Schnitt: aus der Bewegung im vierdimensionalen Phasenraum werden nur die Punkte verwendet, die in der Ebene mit  $l = l_0$  liegen, d.h. die Punkte, in denen die Trajektorie diese Ebene schneidet.

§ 694 Der Poincaré Schnitt in Abb. 6.15 ist für elf verschiedene Anfangsbedingungen bestimmt, für die alle jeweils die (dimensionslose) Gesamtenergie

$$\epsilon = \frac{\dot{l}^2}{2} + \frac{l^2\dot{\varphi}^2}{2} - ql \cos \varphi + \frac{(l-1)^2}{2}$$

eine Konstante, hier  $E_{\text{ges}} = 0.195$ , ist.

§ 695 Abbildung 6.16 zeigt Poincaré Schnitte für die gleichen Trajektorien wie in Abb. 6.15 jedoch für  $\varphi = 0$ : daher ist  $x_n = l_n$  und  $v_n = \dot{l}_n$ . Beide Abbildungen zeigen, dass elliptische Domänen (d.h. periodische Bewegungen) und chaotische Domänen ineinander eingelagert sind.

**Zwischenrechnung 46** Erweitern Sie das Modell aus Abb. 6.13 derart, dass Sie Poincaré

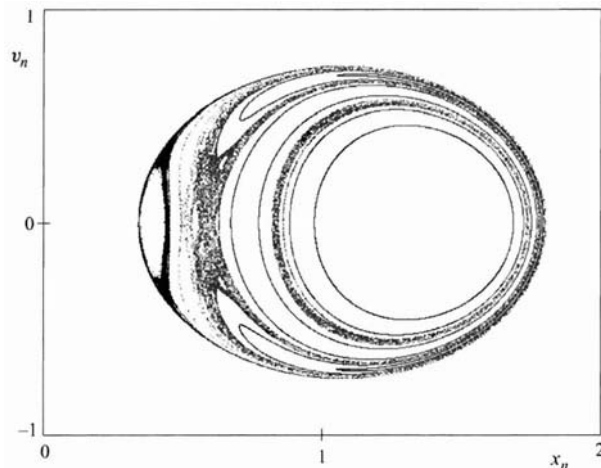


Abbildung 6.16: Wie Abb. 6.15 jedoch Schnitt für  $x_n = l$  und  $v_n = \dot{l}_n$ . Die Domänen für unterschiedliche Anfangsbedingungen treten hier in einer anderen Reihenfolge als in Abb. 6.15 auf [159]

Abbildung 6.17: Fadenpendel an einer Rolle mit Gegenmasse. Das Fadenpendel pendelt, die Gegenmasse kann sich nur in der Vertikalen bewegen

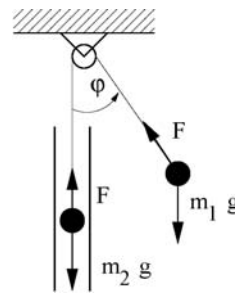


Diagramme für andere Werte von  $q$  erstellen können ebenso wie für andere Gesamtenergien.<sup>10</sup> Ergeben sich qualitative Unterschiede, für die Fälle, in denen die Gesamtenergie für einen Überschlag zu gering ist und denen, in denen ein Überschlag erfolgen kann?

### 6.3.2 Fadenpendel an Rolle

§ 696 Betrachten wir ein System aus zwei Massen  $m_1$  und  $m_2$ , die über einen Faden der Länge  $L$  verbunden sind. Der Faden ist über eine feste Rolle (vernachlässigbarer Radius und vernachlässigbare Masse) geführt. Die Masse  $m_2$  kann nur in der Vertikalen bewegt werden,  $m_1$  dagegen kann auch in einer vertikalen Ebene schwingen, vgl. Abb. 6.17. Die Position dieser schwingenden Masse ist durch die momentane Auslenkung  $\varphi$  und die momentane Länge  $l$  gemessen von der Rolle zur Masse gegeben. Da der Faden eine konstante Länge hat, ist die Lage der zweiten Masse damit eindeutig bestimmt.

§ 697 Dieses System lässt sich am einfachsten in einem mit der Masse  $m_1$  korotierenden Bezugssystem betrachten. Dessen momentane Winkelgeschwindigkeit ist  $\dot{\varphi}$ ; daher wirken auch Trägheitskräfte auf die Masse. Die Zentrifugalkraft  $m_1 l \dot{\varphi}^2$  weist radial entlang des Fadens nach außen, ebenso wie die Komponente  $m_1 g \cos \varphi$  der Gewichtskraft. Die Bewegungsgleichung entlang des Fadens wird damit

$$m_1 \ddot{l} = m_1 g \cos \varphi + m_1 l \dot{\varphi}^2 - K \quad (6.2)$$

mit  $K$  als der Seilspannung. Die Bewegung senkrecht zum Faden ist bestimmt durch die zweite Trägheitskraft, die Corioliskraft  $2m_1 l \dot{\varphi}$ , sowie die Komponente  $-m_1 g \sin \varphi$  der Gewichtskraft:

$$m_1 l \ddot{\varphi} = -m_1 g \sin \varphi - 2m_1 l \dot{\varphi} .$$

<sup>10</sup>Versuchen sie die Erweiterung mit Hilfe der Nullstellendetektion – dann treffen Sie den Nulldurchgang wirklich. Alternativ können Sie als Annäherung in der Ausgabedatei natürlich Nulldurchgänge über (a) Vorzeichenwechsel oder (b) den Abstand von der Nulllinie abschätzen. Um zu spielen und ein Gefühl für die Schnitte zu kriegen ist auch der letztere Ansatz ausreichend.

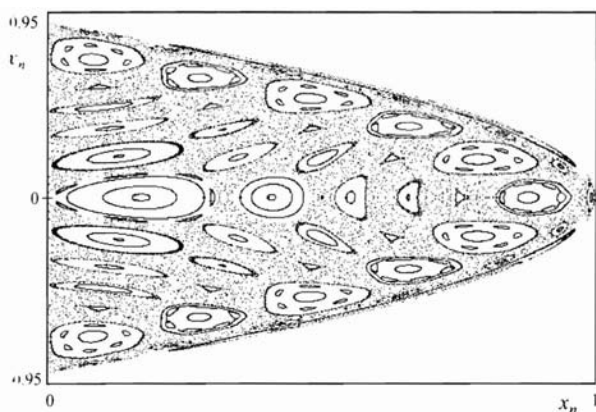


Abbildung 6.18: Poincaré Schnitt für das Pendel mit Rolle aus Abb. 6.17 mit  $x_n = l_n$  und  $v_n = \dot{l}_n$  für  $\mu = 0.3$  und eine dimensionslose Gesamtenergie  $\varepsilon = -0.3$  [159]

§ 698 Die Bewegung der zweiten Masse ist bestimmt durch die Gravitationskraft und die Seilspannung, die gleichzeitig die Kopplung mit der Bewegung der ersten Masse bildet:

$$m_2 \ddot{l} = K - m_2 g .$$

Einsetzen in (6.2) vereinfacht letztere Gleichung zu

$$m_1 \ddot{l} = m_1 g \cos \varphi + m_1 l \dot{\varphi}^2 - m_2 \ddot{l} - m_2 g .$$

§ 699 Division durch die Gesamtmasse liefert für die Bewegungsgleichungen entlang des und senkrecht zum Faden

$$\ddot{l} = \mu g \cos \varphi + \mu l \dot{\varphi}^2 - (1 - \mu)g \quad \text{und} \quad l \ddot{\varphi} = -g \sin \varphi - 2\dot{l}\dot{\varphi}$$

mit der relativen Masse

$$\mu = \frac{m_1}{m_1 + m_2} .$$

§ 700 Mit der Fadenlänge  $L$  als Längenskala und der Periode  $\sqrt{L/g}$  als Zeitskala ergeben sich die dimensionslosen Gleichungen

$$\ddot{l} = \mu g \cos \varphi + \mu l \cot^2 \varphi - (1 - \mu)g \quad \text{und} \quad l \ddot{\varphi} = -\sin \varphi - 2\dot{l}\dot{\varphi} .$$

Die relative Masse bildet den dimensionslosen Parameter.

§ 701 Für eine relative Masse  $\mu = 0.3$  ist in Abb. 6.18 ein Poincaré Schnitt mit  $x_n = l_n$  und  $v_n = \dot{l}_n$  für eine dimensionslose Gesamtenergie

$$\varepsilon = \frac{\dot{l}^2}{2} + \frac{\mu l^2 \dot{\varphi}^2}{2} - \mu l \cos \varphi - (1 - \mu)(1 - l)$$

mit  $\varepsilon = -0.3$  Auch in diesem Beispiel sind elliptische und chaotische Regionen ineinander eingelagert.

**Zwischenrechnung 47** Erstellen Sie das entsprechende Modell in SIMULINK und untersuchen Sie die Bewegung für verschiedene relative Massen  $\mu$ . Charakterisieren Sie die typischen Bewegungen (und begründen Sie diese anschaulich und formal aus den Bewegungsgleichungen). Konstruieren Sie auch dieses Modell so, dass Sie Poincaré Schnitte erstellen können.

### 6.3.3 Duffing-Oszillator

§ 702 Der Duffing-Oszillator ist ein Standard-System in der Darstellung von Chaos. Formal unterscheidet sich der Duffing-Oszillator nur durch einen nichtlinearen  $\beta x^3$ -Term vom angetriebenen harmonischen Oszillator:

$$\ddot{x} + \gamma \dot{x} + \omega_0^2 x + \beta x^3 = f \cos \Omega t . \quad (6.3)$$

Dieser zusätzliche Term führt zu einem völlig anderen Verhalten im Phasenraum.

Abbildung 6.19: Mechanisches Modell zum Duffing-Oszillator: die Masse kann sich nur entlang der  $y$ -Achse bewegen

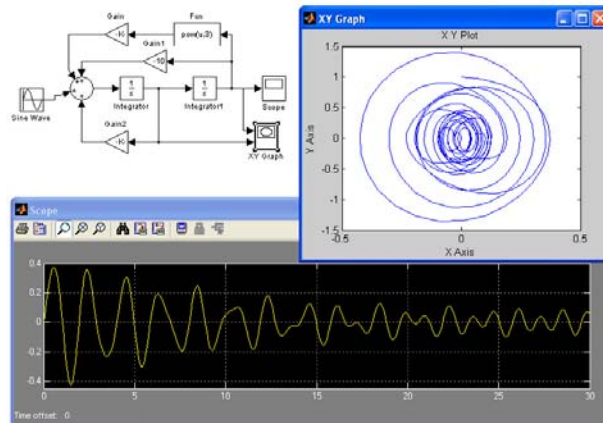
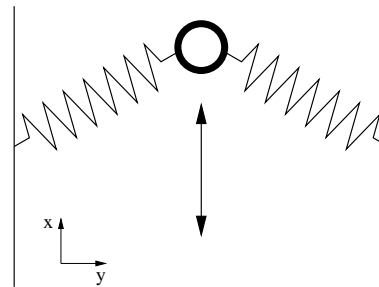


Abbildung 6.20: Duffing-Oszillator in SIMULINK

§ 703 Ein einfaches mechanisches Modell für den Duffing-Oszillator können wir, zumindest näherungsweise, mit Hilfe einer entlang einer Linie beweglichen Masse  $m$  und zwei Federn realisieren, vgl. Abb. 6.19. Verwenden wir konventionelle lineare Federn, so ist die rücktreibende Kraft bestimmt durch  $k\sqrt{x_i^2 + y_i^2}$  mit  $(x_i, y_i)$  als den Koordinaten der Masse relativ zum Befestigungspunkt der Feder. Dieser Ausdruck lässt sich nach Taylor entwickeln, so dass sich neben dem linearen Term ein kubischer ergibt – entsprechend (6.3).

§ 704 Das SIMULINK-Modell für den Duffing-Oszillator lässt sich unmittelbar aus dem des harmonischen Oszillators entwickeln, lediglich ein Gain-Block muss für den kubischen Term der rücktreibenden Kraft ergänzt werden, vgl. Abb. 6.20. Ein Beispiel für eine Lösung in Zeit und Phasenraum ist ebenfalls in der Abbildung gezeigt.

§ 705 Die Struktur möglicher Lösungen wird beim Duffing-Oszillator unübersichtlich, da (6.3) fünf freie Parameter enthält: die Frequenz  $\omega_0$  der freien Schwingung, die Dämpfung  $\gamma$ , die Frequenz  $\Omega$  der anregenden Kraft sowie deren Amplitude  $F$  und den Faktor  $\beta$  vor dem kubischen Term, der auch den Unterschied zur angeregten Schwingung bestimmt. Allerdings hat  $\omega_0^2$  nur in der Abb. 6.19 gegebenen anschaulichen Interpretation des Duffing-Oszillators die Bedeutung einer Frequenz – in (6.3) darf  $\omega_0^2$  auch negative Werte annehmen. In diesem Fall ergibt sich statt des parabolischen Potentials ein  $\omega$ -förmiges, d.h. das Potential wird bi-stabil.

§ 706 Die mit dem  $\omega$ -Potential verbundenen bistabilen Zustände werden in Abb. 6.21 deutlich: die beiden Zeitserien unterscheiden sich nur in den Anfangsbedingungen. Nach dem um  $x = 0$  annähernd symmetrischen Einschwingen schwingt die rote Kurve um das eine Minimum im  $\omega$ -Potential, die blaue Kurve dagegen um das andere. Variation der Anfangsbedingungen für den Duffing-Oszillator mit diesen Parametern führt für große Zeiten immer wieder zu Schwingungen um eine dieser beiden Gleichgewichtslagen.

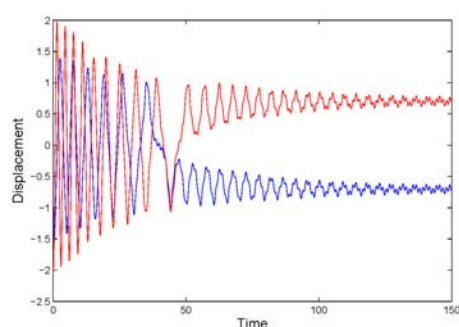


Abbildung 6.21: In Abhängigkeit von den Anfangsbedingungen ergeben sich zwei unterschiedliche Lösungen: der Duffing-Oszillator ist bistabil

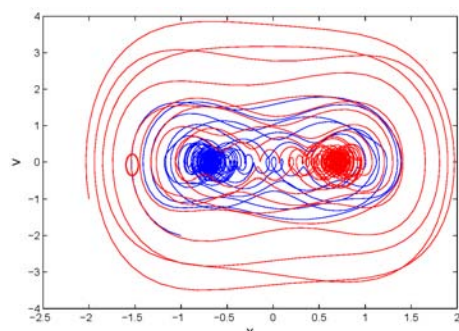


Abbildung 6.22: Wie Abb. 6.21, jedoch im Phasenraum aufgetragen. Die Sensitivität gegenüber den Anfangsbedingungen wird deutlich, da sich die Trajektorien in Richtung auf die beiden bistabilen Zustände im Phasenraum fast berühren können

§ 707 Die Sensitivität des Duffing-Oszillators gegenüber den Anfangsbedingungen wird aus Abb. 6.22 deutlich: hier sind die Phasenraumdiagramme für die beiden Anfangsbedingungen aus Abb. 6.21 gezeigt. Das Einschwingen um die beiden Gleichgewichtspunkte wird deutlich. Auch zeigt sich, z.B. im eingekreisten Bereich, dass sich die Trajektorien der beiden Oszillatoren sehr nahe kommen können: hier entscheidet eine kleine Variation in den Anfangsbedingungen, um welche der beiden Gleichgewichtslagen später geschwungen wird.

**Verständnisfrage 66** In Abb. 6.22 schneiden sich die Phasenraumtrajektorien – oder scheinen sich zumindest zu schneiden. Dürfen Sie das oder dürfen Sie das nicht? Welche Bedeutung hätte ein Schnitt von Trajektorien im Phasenraum anschaulich? Darf eine Trajektorie sich eigentlich selbst schneiden? und welche Bedeutung hätte das?

§ 708 Da ein Verständnis des Duffing-Oszillators über den gesamten Parameter-Bereich bisher nicht erreicht wurde, verwendet man in der Regel eine vereinfachte Gleichung, den Ueda-Oszillator:

$$\ddot{x} + r\dot{x} + x^3 = f \cos(t) .$$

Dieser hängt nur von den beiden Parametern  $r$  und  $f$  ab. Abbildung 6.23 zeigt das Verhalten dieses Oszillators im Parameterraum. In den mit römischen Zahlen markierten Bereichen finden sich periodische Attraktoren mit Periode Eins. In den mit  $m/n$  markierten Bereichen finden sich subharmonische oder ultrasubharmonische Lösungen.<sup>11</sup> In den schraffierten Bereichen sind die Bewegungen chaotisch, allerdings können sich auch Inseln periodischer Bewegung finden.

## 6.4 Gekoppelte Federpendel

§ 709 Wenn ein einzelnes Pendel, zumindest wenn es sich um ein Fadenpendel handelt und/oder die Geometrie etwas anspruchsvoller wird, bereits in Chaos führen kann, sollte

<sup>11</sup>Eine ultrasubharmonische Bewegung der Ordnung  $m/n$  hat eine Frequenz, die das  $m/n$ -fache der anregenden Frequenz ist.

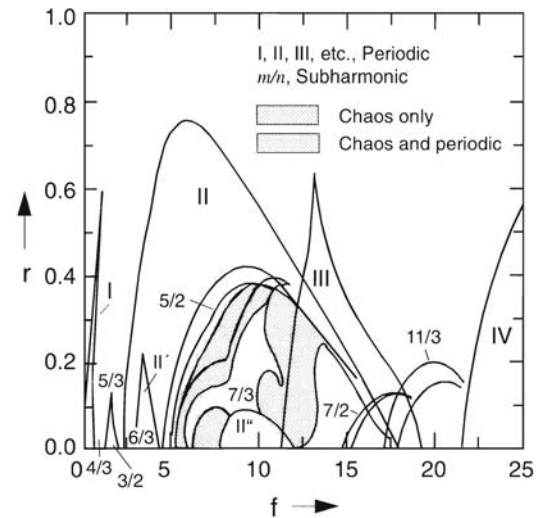


Abbildung 6.23: Parameter-Raum für den Ueda-Oszillator – qualitativ unterschiedliches Verhalten ergibt sich in den einzelnen Bereichen [75]

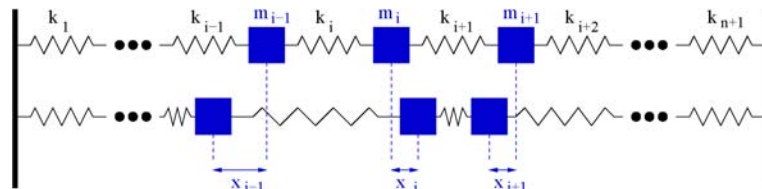


Abbildung 6.24: Lineare Anordnung gekoppelter Pendel

dies für gekoppelte Pendel noch einfacher sein. Um die Situation einfach zu halten, betrachten wir Federpendel. Sind diese mit linearen Federn gekoppelt, so passiert nichts Besonderes (Abschn. 6.4.1). Führen wir statt der linearen Kopplung jedoch eine nichtlineare ein, so entsteht ein neues Phänomen, ein Soliton (Abschn. 6.4.2). Und die Kopplung von Federpendeln durch Federn führt zum Fermi–Pasta–Ulam (FPU) Problem (Abschn. 6.4.3).

### 6.4.1 Lineare Geometrie linearer Pendel

§ 710 Wenn der Übergang von einem Pendel auf zwei gekoppelte Pendel in SIMULINK gemeistert ist, ist die Erweiterung auf ein System von Pendeln nicht mehr schwierig: entsprechend der Zahl der Pendel wird dazu nur der Vektor der Orte und Geschwindigkeiten und die den Gain enthaltende Matrix vergrößert. Damit haben wir ein einfaches Beispiel, an dem wir das Verhalten eines komplexen Systems studieren können.

§ 711 Betrachten wir den allgemeinsten Fall mit  $n$  Massen  $m_i$ , gekoppelt durch  $n + 1$  Federn mit den Federkonstanten  $k_i$ , siehe Abb. 6.24. Auf eine einzelne Masse  $m_i$  wirken die von den Federn  $k_i$  und  $k_{i+1}$  ausgeübten Kräfte, jeweils bestimmt durch die Federkonstante und die Längenänderungen dieser Federn; letztere bestimmt durch die Auslenkungen  $\Delta x_{i-1}$ ,  $\Delta x_i$  und  $\Delta x_{i+1}$  der Massen  $m_{i-1}$ ,  $m_i$  und  $m_{i+1}$  aus der jeweiligen Ruhelage:

$$F_{\text{links}} = k_i(x_{i-1} - x_i) \quad \text{und} \quad F_{\text{rechts}} = k_{i+1}(x_{i+1} - x_i).$$

Am linken bzw. rechten Rand reduzieren sich die Kräfte auf  $F_{\text{links,R}} = -k_1 x_1$  und  $F_{\text{rechts,R}} = -k_{n+1} x_n$ . Die Gesamtkraft auf die  $x_i$ -te Masse und damit die Bewegungsgleichung ist

$$m_i \ddot{x}_i = k_i x_{i-1} - (k_{i+1} - k_i) x_i + k_{i+1} x_{i+1} \quad \forall i \in [2, n-1]$$

oder

$$\ddot{x}_i = \frac{k_i}{m_i} x_{i-1} - \frac{(k_{i+1} - k_i)}{m_i} x_i + \frac{k_{i+1}}{m_i} x_{i+1} \quad \forall i \in [2, n-1].$$

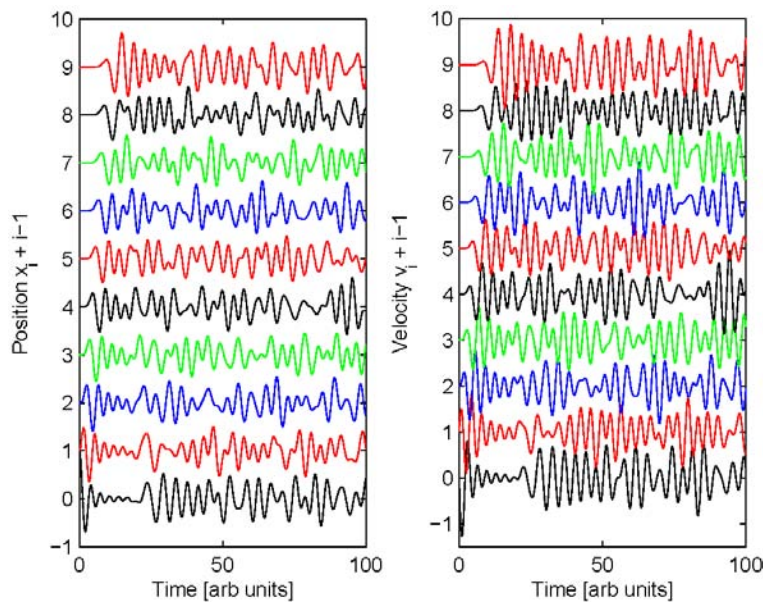


Abbildung 6.25: Weg-Zeit- (links) und Geschwindigkeits-Zeit-Diagramme (rechts) für 10 gekoppelte Pendel; Positionen und Orte jeweils um 1 verschoben, das linke Pendel (untere Kurve) wird anfangs um eine Einheit nach rechts ausgelenkt, alle anderen Massen befinden sich in der Ruhelage. Alle Massen haben eine verschwindende Anfangsgeschwindigkeit.

§ 712 In Matrix-Schreibweise ergibt sich damit als Bewegungsgleichung

$$\ddot{\vec{x}} = \begin{pmatrix} -\frac{k_2+k_1}{m_1} & \frac{k_2}{m_2} & 0 & 0 & \dots & 0 & 0 \\ \frac{k_2}{m_2} & -\frac{k_3+k_2}{m_2} & \frac{k_3}{m_2} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & \frac{k_i}{m_i} & -\frac{k_{i+1}+k_i}{m_i} & \frac{k_{i+1}}{m_i} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \frac{k_n}{m_n} & -\frac{k_{n+1}+k_n}{m_n} \end{pmatrix} \vec{x}.$$

Darin sind in der ersten und letzten Zeile die Randbedingungen bereits berücksichtigt; für eine spezielle Lösung müssen zusätzlich Anfangsbedingungen vorgegeben werden.

§ 713 Wie im Falle des Doppelpendels müssen für eine analytische Lösung die  $n$  Eigenwerte der Matrix, entsprechend den Werten für  $\omega_0$ , bestimmt werden. Für die Lösung in SIMULINK bietet sich dringen die Steuerung der Simulation aus einem m-File heraus an, da sie Matrix sonst zu unübersichtlich wird.

### Alles identisch

§ 714 Im einfachsten Fall sind alle Massen und Federkonstanten identisch, d.h.  $m_i = m$  und  $k_i = k \forall i$ . Dann reduziert sich die Bewegungsgleichung mit der Abkürzung  $\omega_0^2 = k/m$  auf

$$\ddot{\vec{x}} = \begin{pmatrix} -2\omega_0^2 & \omega_0^2 & 0 & 0 & \dots & 0 & 0 \\ \omega_0^2 & -2\omega_0^2 & \omega_0^2 & 0 & \dots & 0 & 0 \\ 0 & \omega_0^2 & -2\omega_0^2 & \omega_0^2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \omega_0^2 & -2\omega_0^2 \end{pmatrix} \vec{x}.$$

§ 715 Für die ersten Versuche verwenden wir eine überschaubare Zahl von zehn Massen. Abbildung 6.25 zeigt die Weg-Zeit- und die Geschwindigkeits-Zeit-Diagramme für die einzelnen Pendelmassen; unten beginnend mit der linken Masse  $m_1$ . Anfangs ist nur diese Masse um eine Einheit nach rechts ausgelenkt, alle anderen Massen befinden sich in der Ruhelage; alle Massen haben eine Anfangsgeschwindigkeit von Null. Für kleine Zeiten ist in beiden Diagrammen zu erkennen, dass sich die der Auslenkung des linken Pendels entsprechende

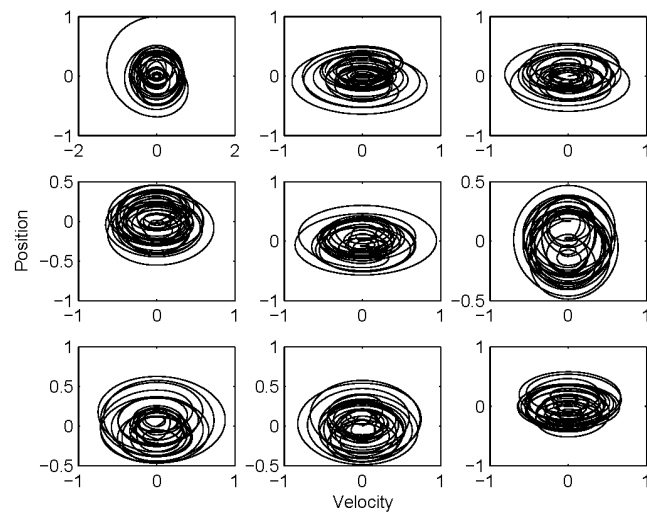


Abbildung 6.26: Darstellung der Bewegung im Phasenraum für die linken 9 Massen des Zehnerpendels; Zeitraum wie in Abb. 6.25

Störung langsam nach rechts ausbreitet – Auslenkungen und Geschwindigkeiten sind bei der rechten Masse noch für einen relativ langen Zeitraum gleich Null.

§ 716 Der mit der Ausbreitung der Störung verbundene Energietransport bewirkt, dass am Ausgangspunkt, d.h. für die ersten Massen am linken Rand, die kinetische und potentielle Energie kleiner werden – ohne jedoch zu verschwinden. Zu dieser Zeit werden kinetische und potentielle Energie bei der rechten Masse maximal – was zu einer Reflektion der Energie nach links führt. Da jedoch nicht die vollständige Anfangsenergie von der linken zur rechten Masse transportiert wurde, sind deren Auslenkungen und Geschwindigkeiten kleiner als die anfänglichen Werte links. Da in dem System keine mechanische Energie in andere Energieformen umgewandelt wird, müssen die anderen Massen und Federn einen Teil der fehlenden mechanischen Energie aufgenommen haben. Aufgrund dieses unvollständigen Energietransports verteilt sich die mechanische Energie im Laufe der Zeit mehr oder weniger gleichmäßig auf die einzelnen Massen: deren Bewegung wird gleichsam thermalisiert.

**Verständnisfrage 67** Warum schwingt die Störung nicht einfach zwischen dem linken Rand hin- und her? Fällt Ihnen eine Konstruktion ein, bei der eine derartige Oszillation auftreten könnte? Wie muss diese sich physikalisch von den gekoppelten Federn unterscheiden?

§ 717 Diese Thermalisierung wird bei der Darstellung im Phasenraum noch deutlicher. In Abb. 6.26 ist die Trajektorie im Phasenraum für alle Massen (mit Ausnahme der rechten) dargestellt für den auch in Abb. 6.25 betrachteten Zeitraum; in Abb. 6.27 wurde ein zehnmal so langer Zeitraum verwendet.

§ 718 Beim kurzen Zeitraum in Abb. 6.26 wird wieder, wie bereits beim Doppelpendel deutlich, dass sich im Phasenraum keine geschlossenen Trajektorien bilden, d.h. die Bewegung ist chaotisch. Die Phasenraumtrajektorien sind recht ähnlich, lediglich die Trajektorie der Masse  $m_1$  bildet eine Ausnahme: da am Anfang die Gesamtenergie in der Auslenkung von Masse  $m_1$  steckt, befindet sich diese am größten Abstand vom Ursprung – keine andere der Massen erreicht jemals einen derartig großen Abstand. Selbst Masse  $m_1$  erreicht diesen nicht wieder, da die Energie sich über die einzelnen Massen verteilt.

§ 719 In der den längeren Zeitraum umfassenden Darstellung der Phasenraumtrajektorien in Abb. 6.27 wird diese Thermalisierung noch deutlicher: die Phasenraumbahnen der einzelnen Massen sind im Hinblick auf den von ihnen eingenommenen Bereich ununterscheidbar geworden – wieder mit Ausnahme von Masse  $m_1$ , in deren Phasenraumbahn die Anfangsenergie für den bereits erwähnten Ausreißer sorgt.



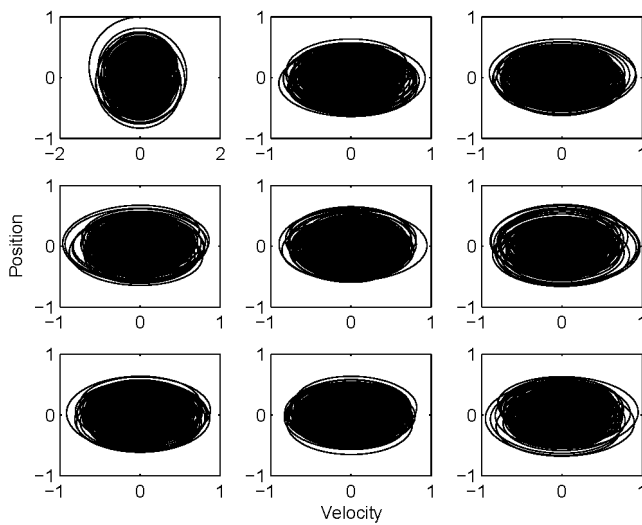


Abbildung 6.27: Darstellung der Bewegung im Phasenraum für die linken 9 Massen des Zehnerpendels; Zeitraum zehnfaches dessen in Abb. 6.25

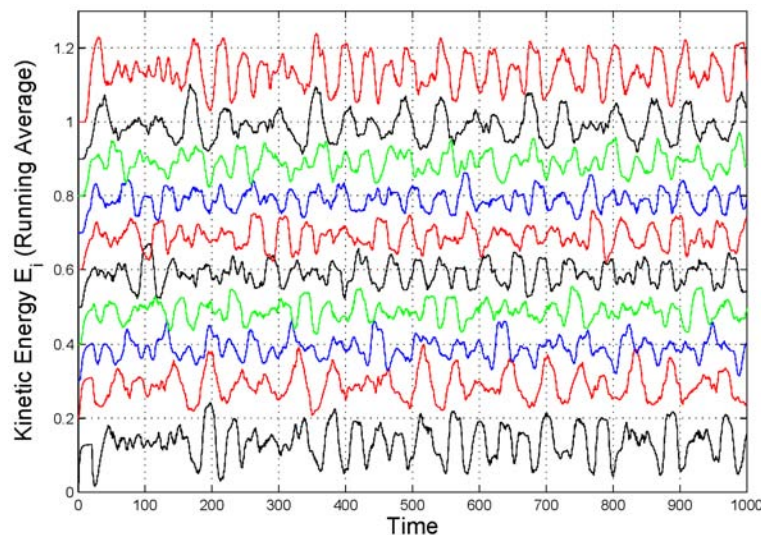


Abbildung 6.28: Gleitendes Mittel der kinetischen Energie der einzelnen Massenpunkte in der Pendellinie

§ 720 Eine alternative Darstellung zur Thermalisierung kann die kinetische Energie der einzelnen Massen verwenden, wie in Abb. 6.28 angedeutet. Auch dort wurde der in Abb. 6.27 verwendete lange Zeitraum betrachtet – um nicht durch die Änderungen der kinetischen Energie während jeder einzelnen Schwingung abgelenkt zu werden, wurde jeweils ein gleitendes Mittel über 1000 Datenpunkte (entspricht 20 Zeiteinheiten) gebildet. Wie aus der Diskussion der Phasenraumbahnen zu erwarten, ist die kinetische Energie am Anfang in den beiden äußeren Massen maximal, anschließend fluktuiert die kinetische Energie für jede einzelne Masse zwar deutlich, der Maximalwert wird jedoch nicht wieder erreicht.

**Verständnisfrage 68** Die kinetische Energie ist nur ein Teil der mechanischen Energie, der andere Teil steckt in der potentiellen Energie der gespannten bzw. komprimierten Federn. Bei der thermischen Bewegung in einem Gas ist die kinetische Energie der einzelnen Gasmoleküle kein konzeptionelles Problem – wie aber sieht es mit der potentiellen Energie aus? Gibt es dort ein Äquivalent, so dass wir bei den gekoppelten Federpendeln von Thermalisierung sprechen können oder ist der Ausdruck nicht angebracht?

§ 721 Beim Übergang zu einer größeren Zahl von gekoppelten Pendeln ist die bisherige Form der Darstellung nicht besonders hilfreich: einige zehn oder gar Hunderte bzw. Tausende von

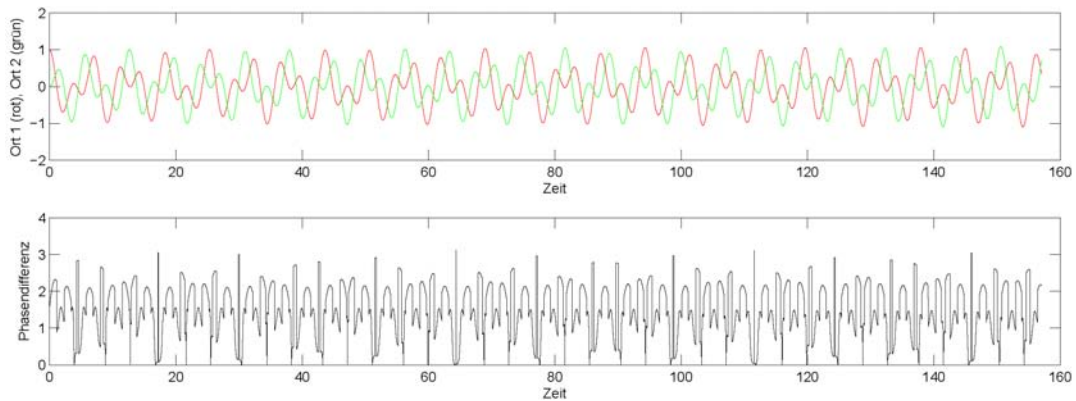


Abbildung 6.29:

Pendeln lassen sich nicht mit Hilfe individueller Orts–Zeit-Plots oder Phasenraumbahnen analysieren. Genauso, wie die Betrachtung der Bahnen der einzelnen Atome/Moleküle eines Gases nicht mehr sinnvoll ist, benötigen wir auch hier statistische Beschreibungsmöglichkeiten.

§ 722 In der einfachsten Form können wir uns auf die Betrachtung der Verteilung (bzw. der zeitlichen Entwicklung der Verteilung) der bisher betrachteten Größen beschränken, d.h. Auslenkungen und Geschwindigkeiten, bzw. aus letzterer abgeleitet, kinetische Energien der einzelnen Massen. In Abhängigkeit von der physikalischen Fragestellung lassen sich jedoch auch Verteilungen bzw. die zeitlichen Entwicklungen von Verteilungen anderer Parameter des Systems betrachten. Wollen wir z.B. der Frage der Synchronisation (siehe Abschn. 9.5) nachgehen, so wäre eine Untersuchung der Verteilung der Phasen sinnvoll. Und Toda's Soliton aus Abschn. 6.4.2 ließe sich aus einer Verteilung der Abstände benachbarter Massen identifizieren.

§ 723 Abbildung 2.11 illustriert die Idee für das einfache Doppelpendel aus Abb. 1.18. Im oberen Teilbild sind die Orte der beiden Massen gegeben: der optische Eindruck lässt eher auf Chaos als auf Synchronisation schließen. Im unteren Teilbild ist die Phasendifferenz gegeben – Hinweise auf Synchronisation lassen sich nicht finden, die Phasendifferenz variiert mit der Zeit. Auch eine schwache Synchronisation mit langsam auseinander driftenden Phasen ist nicht zu erkennen: in diesem Fall würde sich die Phasendifferenz linear mit der Zeit entwickeln.

**Zwischenrechnung 48** Führen Sie die Untersuchung auf Synchronisation für die bisher betrachtete Kette aus 10 Massen durch.

**Verständnisfrage 69** Und wie sieht es bei einer zweidimensionalen Geometrie aus, d.h. jedes Pendel ist mit vier benachbarten Pendeln verbunden?

### Unterschiedliche lineare Federn

§ 724 Die Verwendung unterschiedlicher Federkonstanten/Massen verändert die Details der Bewegung, nicht jedoch die wesentlichen Merkmale: Energie wird zwischen den Massen und Federn verlagert, es kommt zu einer Thermalisierung.

**Zwischenrechnung 49** Führen Sie die entsprechenden Simulationen in SIMULINK durch. Überprüfen Sie auch, im Vorgriff auf das Fermi–Pasta–Ulam problem in Abschn. 6.4.3 das Langzeitverhalten.

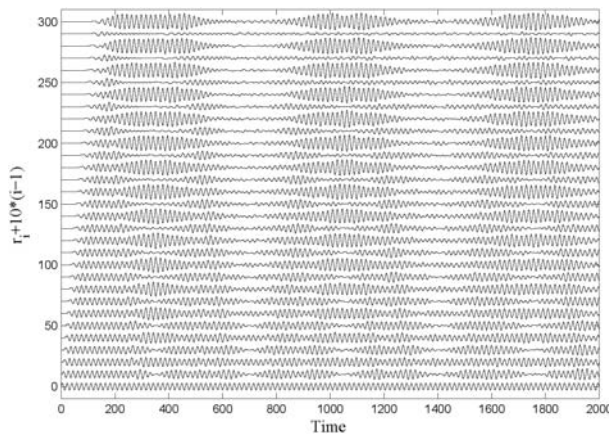


Abbildung 6.30: Vereinfachtes Toda Modell

### 6.4.2 Nicht-lineare Version: Toda's Soliton

§ 725 Verwenden nun wir in der Pendelkette statt der dem Hook'schen Gesetz gehorchenden linearen Federn solche mit nicht-linearen Eigenschaften – allgemein bezeichnet als FPU Lattices nach einer Arbeit von Fermi, Pasta und Ulam [28]. In den 1960 studierte Morikazu Toda [161, 162] ein System aus Federn und Massen entsprechend dem in Abb. 6.24 – diese Konfiguration wird daher auch als Toda-Lattice bezeichnet. Die Massen sind alle identische Einheitsmassen, die nichtlinearen Federn werden charakterisiert durch ihre potentielle Energie  $U(r)$  mit  $r$  als der Änderung des Abstandes benachbarter Massen vom Ruhezustand (entsprechend einem Minimum in der Energie der Feder). Die von einer Feder ausgeübte Kraft ist daher  $F = -dU/dr$ . Mit  $x_n$  als der Auslenkung der  $n$ -ten Masse aus ihrer Gleichgewichtslage ergibt sich als Bewegungsgleichung analog zu der bei linearen Federn verwendeten

$$m_n \ddot{x}_n = U'(x_{n+1} - x_n) - U'(x_n - x_{n-1}).$$

Mit  $r_n = x_{n+1} - x_n$  wird die Gleichung zu

$$m_n \ddot{r}_n = \left( \frac{dU(r_{n+1})}{dr_{n+1}} - \frac{dU(r_n)}{dr_n} \right) - \left( \frac{dU(r_n)}{dr_n} - \frac{dU(r_{n-1})}{dr_{n-1}} \right).$$

Mit Einheitsmassen wird diese Bewegungsgleichung zu

$$\ddot{r}_n = \frac{dU(r_{n+1})}{dr_{n+1}} - 2 \frac{dU(r_n)}{dr_n} + \frac{dU(r_{n-1})}{dr_{n-1}}.$$

§ 726 Eine exakte analytische Lösung ergibt sich für das von Toda vorgeschlagene Federpotential

$$U(r) = \frac{a}{b} (e^{-br} + br - 1)$$

mit  $a, b > 0$ . Mit diesem Potential nimmt die Bewegungsgleichung folgende Form an

$$\ddot{r}_n = a(2e^{-br_n} - e^{-br_{n+1}} - e^{-br_{n-1}}).$$

§ 727 Im Grenzfall  $b \rightarrow 0$  (aber trotzdem  $ab$  endlich), reduziert sich die Bewegungsgleichung auf eine lineare Differential-Differenzgleichung

$$\ddot{r}_n = ab(r_{n+1} - 2r_n + r_{n-1}).$$

Die Lösungen dieser Gleichungen sind langwellige Wellen mit einer Geschwindigkeit von  $\sqrt{ab}$  Massenpunkten pro Zeiteinheit.

§ 728 Der Grenzfall  $b \rightarrow 0$  lässt sich im Modell in der gleichen einfachen Schleife wie der harmonische Oszillator darstellen; daher ist das SIMULINK-Modell hier nicht noch einmal explizit angegeben. Die Gain-Matrix ist ebenfalls einfach. Abb. 6.30 zeigt die Lösungen für  $r$

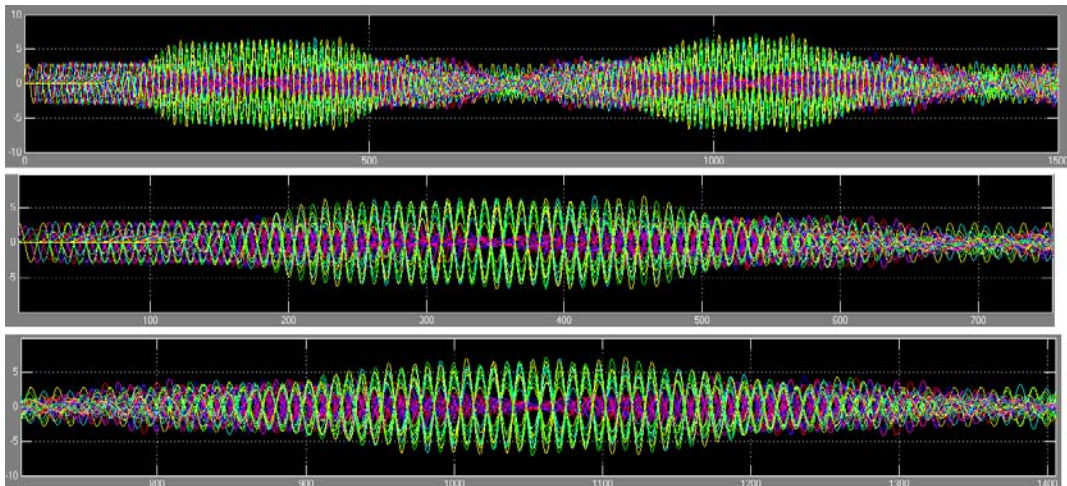


Abbildung 6.31: Vereinfachtes Toda Modell II: Übersicht (*oben*), Einschwingen (*Mitte*) und eine Periode (*unten*)

für eine Kette von 31 Pendeln; als Anfangsbedingung bewegt sich die linke Masse (unterste Kurve) aus der Ruhelage nach rechts. Unabhängig von den Details und deren Interpretation ist ein Unterschied offensichtlich: im Vergleich zu den gekoppelten harmonischen Oszillatoren in Abb. 6.25 zeigen sich in der Abbildung Strukturen, die auf ein geordnetes Verhalten schließen lassen. Auch bei der gemeinsamen Darstellen aller  $r(t)$ -Kurven auf dem Oszilloskop wird diese Struktur deutlich, wie in Abb. 6.31 zu erkennen.

**Verständnisfrage 70** Ist der Vergleich von Abb. 6.30 und Abb. 6.25 überhaupt korrekt? Oder zeigen beide Abbildungen etwas unterschiedliches? Wenn ja, wie lassen sie sich in einander überführen? Ändert sich dadurch die Aussage des voran gegangenen Paragraphen?

§ 729 Ist der Parameter  $b$  nicht klein, so lässt sich mit einer inversen Streu-Transformation (IST) das Differentialgleichungssystem exakt lösen (z.B. [138, 162]); das sich ergebende Soliton hat die Form

$$r_{n,s} = -\frac{1}{b} \log \left[ 1 + \sinh^2 \kappa \operatorname{sech}^2 \left( \kappa n \pm \sqrt{ab} \sinh \kappa \right) \right]$$

mit  $\kappa$  als einem Amplitudenparameter. Wie überlich deuten die beiden Vorzeichen an, dass sich das Soliton in positiver und negativer Richtung ausbreiten kann. Dieses Soliton entspricht einer Kompressionswelle. Deren Ausbreitungsgeschwindigkeit ist

$$v = \sqrt{ab} \left( \frac{\sinh \kappa}{\kappa} \right) > \sqrt{ab}$$

und damit sowohl größer als die Ausbreitungsgeschwindigkeit einer Welle kleiner Amplitude als auch größer als die der sich aus der Korteweg–de Vries Gleichung ergebenden Solitonen (wie z.B. Russell's Wasserwelle).

**Zwischenrechnung 50** Erstellen Sie ein Modell von Toda's Soliton in SIMULINK. Überprüfen Sie das Modell mit Hilfe der analytischen Lösung. Untersuchen Sie andere Potentiale. Lässt sich eine Bedingung an das Potential für das Auftreten von Solitonen abschätzen?

### 6.4.3 Fermi–Pasta–Ulam Problem: MANIAC's Test

§ 730 MANIAC I (Mathematical Analyzer Numerical Integrator and Computer Model I) war einer der ersten Universalrechner.<sup>12</sup> Neben frühen Monte Carlo Simulationen wurde MANIAC

<sup>12</sup>Wikipedia ([http://de.wikipedia.org/wiki/MANIAC\\_I](http://de.wikipedia.org/wiki/MANIAC_I)) gibt die folgende Information: 'Der MANIAC I (Mathematical Analyzer Numerical Integrator and Computer Model I) war ein früher Universalrechner, der

auch für andere Untersuchungen, z.B. des genetischen Codes, eingesetzt. Eine dieser anderen Fragestellungen betraf die schwingende Saite (bzw. ein System gekoppelter Oszillatoren), allerdings nicht in der linearen sondern in der nicht-linearen Version.

§ **731** Das Projekt wurde auf Anregung Fermis von ihm, Ulam und Pasta ausgeführt. Die gesamte Schwingungsenergie wurde anfangs in eine Schwingungsmode gesetzt, wie erwartet verteilte sich diese Energie im Laufe der Zeit auf benachbarte Schwingungsmoden bis sich ein Gleichgewicht eingestellt hat. Die nichtlineare Saite verhielt sich wie erwartet – “But the completely unexpected happened one day when a typical problem was being computed. Owing to a very lively and distracting discussion, the computations continued beyond the usual cutoff. The results were so strange and mysterious that everyone around was quick to assume that the computer, the traditional whipping boy, had gone awry. The vibrational energy had returned to its initial state, within a few percent! The rest is history—nonlinear studies were shown to have some fascinating aspects. The ideas of soliton theory emerged and there was an outpouring of papers that became a minor industry. Today this classic work is known as the FPU problem.” [1]

§ **732** Das Grundgitter zum FPU ist bereits durch Toda’s Soliton in § 725 gegeben – Toda ist nur ein Spezialfall des Fermi–Pasta–Ulam-Problems. Das FPU-Problem unterscheidet sich in einer wesentlichen Hinsicht vom Duffing-Oszillator: der Duffing-Oszillator enthält nicht nur ein  $\omega$ -Potential (oder anschaulicher einen nicht-linearen Term in der rücktreibenden Kraft) sondern auch Reibung und ist damit dissipativ. Das FPU-Problem dagegen enthält ein Potential, das sicherlich nicht zu einer linearen rücktreibenden Kraft führt, aber keine Reibung. Damit ist das FPU-Problem nicht dissipativ und damit ein Hamilton’sches System.

### Einschub: Hamilton-Systeme

§ **733** In einem Hamilton-System mit  $N$  Freiheitsgraden lässt sich die Dynamik mit Hilfe der Hamilton-Funktion  $H(\vec{p}, \vec{q}, t)$  beschreiben. Dabei sind  $\vec{q} = (q_1, \dots, q_N)$  und  $\vec{p} = (p_1, \dots, p_N)$  die kanonischen generalisierten Koordinaten und Impulse. Die Hamilton’schen Bewegungsgleichungen

$$\dot{p}_i = -\frac{\partial H}{\partial q_i}, \quad \dot{q}_i = \frac{\partial H}{\partial p_i}, \quad i = 1, \dots, N$$

erzeugen Trajektorien  $(\vec{p}(t), \vec{q}(t))$  im  $2N$ -dimensionalen Phasenraum.

§ **734** Die Hamilton-Funktion erzeugt also einen Strom im Phasenraum. Da das System nicht dissipativ ist, bleibt das Phasenraumvolumen

$$dV = dp_1 \dots dp_N dq_1 \dots dq_N$$

ebenso erhalten wie die Phasenraumfläche

$$d\vec{p} \cdot d\vec{q} = \sum_{i=1}^N dp_i dq_i .$$

---

unter der Leitung von Nicholas Metropolis Anfang der fünfziger Jahre für das Los Alamos National Laboratory gebaut wurde. Der auf der Von-Neumann-Architektur basierende Rechner wurde bei der University of California gebaut und kostete mit Ausgabegeräten 298.000 US-Dollar.

MANIAC benötigte eine Standfläche von 1,85 Quadratmetern und war knapp 2 Meter hoch. Der Energiebedarf des Rechners betrug 35 Kilowatt. 2400 Elektronenröhren und 500 Kristalldioden übernahmen die Schaltungen, die Befehle wurden entweder per Lochstreifen oder über Magnetband eingelesen. Die Speicherung erfolgte entweder elektrostatisch (Kapazität: 1024 Worte) oder auf einer Magnettrommel (Kapazität: 10.000 Worte). Die Durchführung einer Addition dauerte 80 Millisekunden, Multiplikations- und Divisionschritte nahmen eine Sekunde in Anspruch. Die Ausgabe erfolgte entweder über einen Anelex-Drucker, einen Fernschreiber, auf Papierband oder einen Magnetstreifen.

MANIAC I lief zum ersten Mal erfolgreich im März 1952, er wurde dann zur Berechnung von Nuklearwaffen eingesetzt, bis er 1957 durch seinen Nachfolger MANIAC II ersetzt wurde. Der Großrechner wurde dann an die University of New Mexico übergeben, wo er weiterverwendet wurde.’

§ **735** Ein einfaches System mit zwei Freiheitsgraden ist die Bewegung eines Teilchens in einem zweidimensionalen Potential  $V(x, y)$ . Die Hamilton-Funktion ist

$$H(p_x, p_y, x, y) = \frac{1}{2m} (p_x^2 + p_y^2) + V(x, y).$$

Daraus ergeben sich die Bewegungsgleichungen

$$\dot{p}_x = -\frac{\partial V}{\partial x}, \quad \dot{p}_y = -\frac{\partial V}{\partial y}, \quad \dot{x} = \frac{p_x}{m} \quad \text{und} \quad \dot{y} = \frac{p_y}{m}.$$

Diese Bewegungsgleichungen beschreiben die Entwicklung des Systems im vier-dimensionalen Phasenraum. Auf Grund der Energieerhaltung ist die Trajektorie des Systems auf eine dreidimensionale Mannigfaltigkeit  $H(p_x, p_y, x, y) = E$  beschränkt – diese wird auch als Energiefläche bezeichnet.

### FPU continued

§ **736** Die Hamilton-Funktion für das FPU-Problem für  $N$  Massenpunkte mit Einheitsmasse ist gegeben als

$$H(p, q) = \sum_{k=1}^N \left[ \frac{1}{2} p_k^2 + \frac{1}{2} (q_{k+1} - q_k)^2 + \frac{1}{3} \alpha (q_{k+1} - q_k)^3 + \frac{1}{4} \beta (q_{k+1} - q_k)^4 \right].$$

Die letzten beiden Terme geben die nichtlinearen Anteile der Kraft; ihre relativen Stärken werden durch die Parameter  $\alpha$  und  $\beta$  bestimmt. Die Randbedingungen können feste Enden ( $q_1 = q_{N+1} = 0$ ) oder periodisch ( $q_{N+1} = q_1$ ) sein.<sup>13</sup>

§ **737** In der den anfänglichen numerischen Versuchen wurde die Anfangsenergie in die niedrigste Mode des linearisierten Systems gesteckt – das ist der halbe Sinus, dem wir auch bei der schwingenden Seite begegnet sind. Bei niedriger Energie wird diese eine quasi-Mode des nichtlinearen Systems. Die  $N$  unabhängigen Moden  $Q_k$  des harmonischen Systems mit festen Rändern sind

$$q_i = \sqrt{2N+1} \sum_{k=1}^N Q_k \sin\left(\frac{k\pi}{N+1}\right).$$

Die zugehörigen harmonischen Frequenzen sind

$$\omega_k = 2 \sin\left(\frac{k\pi}{2N+2}\right).$$

Die ersten Untersuchungen zeigten, wie auch in Abb. 6.30 und 6.31 zu erkennen, Wiederholungen anstelle der gleichmäßigen thermodynamischen Verteilung der Energie auf die Moden. Diese Muster werden anschaulich als Ergebnis von Resonanzen zwischen verschiedenen Kombinationen von Moden erklärt.

## Literatur

§ **738** Eine sehr übersichtliche Einführung gibt Smith [150] – das Büchlein ist sehr praktisch, um die wesentlichen Begriffe sortiert zu kriegen. Eine gute aber immer noch einfach zu verstehende formale Einführung geben Korsch et al. [75]: zwölf verschiedene Projekte zum Chaos können mit Hilfe von C++-Programmen genauer untersucht werden. Einfache Systeme werden auch in [159] betrachtet: dort fehlt zwar die Rechnerunterstützung, dafür werden die mit Chaos und dynamischen Systemen verbundenen Konzepte besser eingeführt.

<sup>13</sup>In den Originalarbeiten wurden feste Ränder verwendet, so dass das ganze System einer diskretisierten nichtlinearen Feder entspricht. Außerdem wurde entweder  $\alpha$  oder  $\beta$  gleich Null gesetzt, d.h. es wurde wie beim Duffing Oszillator nur ein nichtlinearer Anteil verwendet. Da die meisten Varianten mit  $\alpha = 0$  gerechnet wurden, wird das betrachtete Problem auch als FPU- $\beta$  Modell bezeichnet. Letzteres kann einfach normiert werden mit dem einzigen signifikanten Parameter  $\beta E$ , wobei  $H(p, q) = E$  die Gesamtenergie ist. Das FPU- $\beta$  Modell hat später zur Entwicklung der ersten Modelle von Solitonen beigetragen.

§ **739** Die geometrische Annäherung an das Chaos in Form von Fraktalen ist zum einen in den Klassikern von Feigenbaum [26, 27] und Mandelbrot [91] dargestellt. Von letzterem gibt es auch eine interessante Anwendung auf Finanzmärkte [92]. Eine umfassende und allgemeinverständliche Einführung in den Zusammenhang zwischen Fraktalen und Chaos geben Peitgen et al. [117] oder in der deutschsprachigen Taschenbuchversion [118, 119]; als Bildband ist Peitgen und Richter [116] sicherlich der Klassiker.

§ **740** Pendelliege und andere Variationen über Toda's Soliton und das Fermi–Pasta–Ulam Problem werden u.a. diskutiert in Berman und Izrailev [6], Dauxois et al. [20], Gallavotti [30] und Lichtenberg et al. [82]; Dauxois [19] zeigt in seinem historischen Rückblick auf das Problem außerdem den von Mary Tsingou entworfenen Algorithmus zur Codierung des Problems – wenn das Bild etwas besser lesbar wäre, könnte man es direkt in SIMULINK übersetzen.

## Fragen

**Frage 24** Erklären Sie anschaulich und mit Hilfe von Beispielen, was deterministisches Chaos ist. Welche Zutaten benötigen Sie für ein chaotisches System?

**Frage 25** Was ist ein Attraktor? geben Sie Beispiele.

**Frage 26** Was ist ein Fraktal; was bedeutet eine fraktale Dimension?

**Frage 27** Was bedeutet Selbstähnlichkeit?

**Frage 28** Erläutern Sie den Zusammenhang zwischen Chaos, Attraktor und Fraktal.

**Frage 29** Wie kann man Chaos darstellen?

**Frage 30** Muss ein chaotisches System dissipativ sein?

**Frage 31** Was ist ein Soliton?

## SIMULINK-Spielereien

**SIMULINK-Aufgabe 20** Das bereits im Zusammenhang mit Abb. 6.5 betrachtete Magnetpendel bewegt sich über der  $xy$ -Ebene. Die Bewegungsgleichung ist bei  $N$  Magneten an den Positionen  $(x_i, y_i)$  gegeben durch

$$\ddot{x} + \omega_0^2 x + \alpha \dot{x} = \sum_{i=1}^N \frac{\gamma_i (x_i - x)}{D_i^3} \quad \text{und} \quad \ddot{y} + \omega_0^2 y + \alpha \dot{y} = \sum_{i=1}^N \frac{\gamma_i (y_i - y)}{D_i^3}. \quad (6.4)$$

Darin ist der Reibungskoeffizient  $\alpha > 0$  und  $\omega_0 = \sqrt{g/l}$  die Frequenz der freien Schwingung; die anziehende Kraft des  $i$ -ten Magneten ist unter der Annahme punktförmiger magnetischer Monopole in Anlehnung an das Coulomb-Gesetz gegeben durch  $\gamma_i/D_i^2$  mit  $\gamma_i$  als Maß für die Magnetfeldstärke und

$$D_i = \sqrt{(x_i - x)^2 + (y_i - y)^2 + d^2}$$

als dem Abstand zwischen dem Pendelkörper und dem Magneten. Konstruieren Sie das entsprechende Modell und untersuchen Sie das Modellverhalten für  $\gamma_i = 1$  und verschiedene Werte von  $\alpha$ ,  $\omega_0$  und  $d$ .

**SIMULINK-Aufgabe 21** Statt des normalen Federpendels haben Sie ein Gummipendel zur Verfügung: die Feder mit der linearen rückstellenden Kraft ist ersetzt durch ein Gummiband (Bungee-Seil?) mit einer nichtlinearen rückstellenden Kraft. Besorgen Sie sich einen Zusammenhang zwischen rückstellender Kraft und Auslenkung (Internet oder experimentell, z.B. Packriemen) und erstellen Sie ein entsprechendes Modell in SIMULINK. Wie verhält sich das Modell im Phasenraum. Kommt es zu einer chaotischen Bewegung oder nicht? Wie sieht das Potential aus?

**SIMULINK-Aufgabe 22** Spielen Sie mit dem Modell eines Duffing-Oszillators in SIMULINK. Untersuchen Sie das Verhalten für verschiedene Parameter-Kombinationen. Nehmen Sie den reduzierten Duffing (Ueda-Oszillator) und Abb. 6.23 als Orientierungshilfe.

## Kleinprojekte

### FPU-Problem

§ 741 Da das Fermi–Past–Ulam Problem gerade seinen fünfzigsten Geburtstag feiert, gibt es eine Vielzahl von aktuellen Veröffentlichungen und Rückblicken dazu. Sie haben also genug Literatur, um sich im Rahmen eines Kleinprojektes ein Modell des FPU-Problems in SIMULINK zu erstellen und die frühen Untersuchungen daran nachzuvollziehen.

### Synchronisation von van der Pol Oszillatoren

§ 742 In Kapitel 7 werden wir das Hodgkins–Huxley-Modell des Neurons als Beispiel für ein Diffusions–Reaktions-Modell kennen lernen. Ein einfacheres Modell eines Neurons kann mit einem einfachen Integrate–and–Fire Oszillator erzeugt werden. Der van der Pol Oszillator (siehe auch § 930) ist ein einfaches Beispiel. Erstellen Sie das entsprechende Modell in SIMULINK. Da van der Pol Oszillatoren sehr leicht synchronisieren [113], spielen Sie mit einer Kette oder einem Gitter dieser Oszillatoren und versuchen Sie, diese zur Synchronisation zu bringen. Wie stellen Sie Synchronisation am einfachsten dar? Strogatz [156] liefert schöne anschauliche Beispiele und Hinweise dazu; das Paper vom gleichen Autor [157] ist etwas formaler.

### Chua-Oszillator

§ 743 Eher eine MATLAB-Aufgabe: auch elektrische Schwingkreise können nichtlinear sein. Der Chua-Oszillator wird wie der Lorenz-Attraktor durch einen Satz von drei gekoppelten Differentialgleichungen mit vergleichbarer Struktur beschrieben. Sie können die entsprechenden Programme mit einer sehr ausführlichen Dokumentation unter <http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=3541&objectType=File> herunterladen und damit experimentieren. Versuchen Sie, entsprechend der Abb. 6.23 Ordnung in die verschiedenen, im Programm gegebenen Parametersätze zu bringen: wann verhält sich der Oszillator periodisch, wann chaotisch, wann ist es nicht vorherzusagen. Ein etwas anspruchsvollerer Artikel zum Chua-Oszillator stammt von Petras [120] – ein SIMULINK-Modell wird dort ebenfalls skizziert.

### Etwas exotisch: Quaternionische Fraktale

§ 744 Fraktale wie die Mandelbrot- und die Julia-Menge werden in 2D-Abbildungen dargestellt. Wie erzeugt man Fraktale im 3D? Eine in vielen Programmen geläufige Variante ist die Darstellung der Iterationsschritte nicht als Farbkodierung sondern als dritte Dimension. Einen ganz anderen Ansatz hat Christen [14] in seiner Matura-Arbeit gewählt: er wählt Quaternionen statt komplexer Zahlen als Basis und gelangt damit automatisch in höhere Dimensionen. Arbeiten Sie sich in das Thema ein, erstellen Sie ein entsprechendes Modell und spielen Sie damit. Was macht man eigentlich sinnvollerweise mit der vierten Dimension?



# Kapitel 7

## Diffusions–Reaktions-Modelle

Nonlinear science gives us the insight to accept things we cannot calculate, the ability to calculate things we can and the wisdom to know the difference.  
Scott [139]

§ 745 Dieses Kapitel enthält keine systematische Einführung in nichtlineare Systeme; letztere finden Sie in einfacher Darstellung in Scott [139], in etwas anspruchsvollere Darstellung ebenfalls bei Scott [138]. Im Rahmen dieses Kapitels werden wir als ein Beispiel für nicht-lineare Systeme und deren Fähigkeit zur Emergenz bzw. Selbstorganisation Diffusions–Reaktions-Modelle betrachten. Anwendungsbeispiele sind die Ausbildung von Hirnarealen, Jäger–Beute Modelle, die Anlagerung auf Oberflächen und die Muster auf Tierhäuten.

**Lernziele:** nach Durcharbeiten dieses Kapitels (und eventuell der Begriffsliste in Anhang 9) sollten Sie in der Lage sein,

- Begriffe wie Selbstorganisation und Emergenz zu erläutern und Beispiele zu geben.
- die Struktur und allgemeine Anwendbarkeit von Diffusions–Reaktions-Modellen zu erläutern.
- komplexere Modelle in SIMULINK auch unter Verwendung mehrere, gegebenenfalls umfangreicher und verschachtelter Untersysteme zu entwerfen und zu kontrollieren.
- die Fähigkeit von SIMULINK mit Hilfe von selbst gestrickten Funktionen (m-Files und s-Files) zu erweitern.
- partielle Differentialgleichungen in SIMULINK hinein zu mogeln.

### 7.1 Das Gleichungssystem

§ 746 Das Diffusions–Reaktions-Modell ist ein Musterbeispiel für ein nichtlineares System: zum einen zeigt es Selbstorganisation, zum anderen hat es vielfältige Anwendungen. Ein Beispiel, die Belousov–Zhabotinsky-Reaktion haben wir bereits in der Einleitung kennen gelernt (u.a. Abb. 1.26) und erläutert, dass ein entsprechendes System durch zwei konkurrierende Prozesse, Diffusion und Reaktion, sowie die entsprechenden Spezies und die zugehörigen Reaktions- bzw. Ausbreitungsparameter bestimmt wird.

§ 747 Das allgemeine Diffusions–Reaktions-Modell kann in kompakter Schreibweise dargestellt werden als

$$\frac{\partial \vec{c}}{\partial t} = D \Delta \vec{c} + \vec{F}(\vec{c}) . \quad (7.1)$$

Darin gibt  $\vec{c}$  die Konzentrationen der verschiedenen beteiligten Spezies  $c_i$ , der Diffusionstensor  $D$  auf seiner Diagonalen<sup>1</sup> die Diffusionskoeffizienten  $D_i$  der entsprechenden Spezies, und die vektorwertige Funktion  $\vec{F}(\vec{c})$  die Reaktionen und Verluste der einzelnen Spezies. Die Funktion  $\vec{F}$  koppelt die einzelnen Gleichungen des Gleichungssystems (und damit die einzelnen Komponenten des Vektors  $\vec{c}$ ), da die Senke in der Spezies  $c_i$  eine Quelle für die Spezies  $c_{i+1}$  sein kann oder umgekehrt.

§ 748 Die Form der Reaktionsfunktion hängt von der Art der chemischen Reaktion ab und führt zu unterschiedlichen Anwendungen und numerischen Verfahren. Die wichtigsten Typen chemischer Reaktionen sind

- Reaktion 1. Ordnung mit der Reaktionsrate

$$r = k_r c,$$

d.h. die Reaktion ist linear von der Konzentration  $c$  abhängig.

- Reaktion 2. Ordnung mit der Reaktionsrate

$$r = K_R c^2 \quad \text{oder} \quad r = k_r c_A c_B,$$

d.h. die Reaktionsrate hängt quadratisch von der Konzentration  $c$  ab oder sie hängt vom Produkt der Konzentrationen der beiden beteiligten Substanzen ab.

- Michaelis–Menten-Reaktionsrate

$$r = \frac{r_{\max} S}{k_m + S}$$

Die Michaelis–Menten [100] (manchmal auch als Michaelis–Menten–Henri [41]) Kinetik lässt sich zur Beschreibung der Kinetik einer Vielzahl von Enzymen anwenden und gilt daher als der Grundstein der Enzymkinetik. Auch bildet sie einen guten Anknüpfungspunkt zu Kapitel 6: es handelt sich um eine fraktale Kinetik [74, 134, 180].

§ 749 Zwei spezielle Lösungen der Diffusions–Reaktions-Modells sind von besonderem Interesse: oszillierende Lösungen, wie z.B. zur Beschreibung der Belousov–Zhabotinsky-Reaktion oder der Reizleitung im Hodgkins–Huxley Neuron erforderlich, und stationäre Lösungen, wie z.B. bei den Turing-Mustern erwünscht. Jäger–Beute-Modelle können (quasi-)oszillierende Lösungen haben, ebenso gut aber auch chaotische.

## 7.2 Anwendungen

§ 750 Die Diffusions–Reaktions-Gleichung in der allgemeinen Form (7.1) ist auf Grund der vielfältigen Anwendungsmöglichkeiten so interessant. Diese unterscheiden sich in der Zahl der beteiligten Komponenten sowie der Kopplung zwischen diesen, d.h. der Zahl und numerischen Beschreibung der einzelnen Reaktionen im Vektor  $\vec{F}(\vec{c})$  – für ein komplexeres biochemisches Beispiel siehe z.B. [175]. Dieser lässt sich in der Regel nur als Funktion der Konzentrationen darstellen, da eine Schreibweise über eine Matrix und den Vektor der Konzentrationen nur für die Beschreibung linearer Reaktionen möglich wäre.

<sup>1</sup>Die Elemente außerhalb der Diagonalen verschwinden, da die Diffusion als isotrop angenommen wird. Zur Erinnerung: die Diffusionsgleichung ( $\rightarrow$  Abschn. 9.1.1) ist gegeben als

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial x^2} \quad \text{bzw.} \quad \frac{\partial c}{\partial t} = D \Delta c$$

mit  $D$  als dem Diffusionskoeffizienten. Diese Darstellung gilt nur für isotrope Diffusion mit einem vom Ort unabhängigen Diffusionskoeffizienten; bei anisotroper Diffusion wird der Diffusionskoeffizient durch einen Diffusionstensor ersetzt:  $\dot{c} = -D\Delta c$ . Ist der Diffusionskoeffizient ferner vom Ort abhängig, so verschwindet die räumliche Ableitung des Diffusionskoeffizienten nicht und die bisher betrachtete vereinfachte Form muss ersetzt werden durch die direkt aus der Kontinuitätsgleichung abgeleitete Form der Diffusionsgleichung:  $\dot{c} = -\nabla D \nabla c$ .

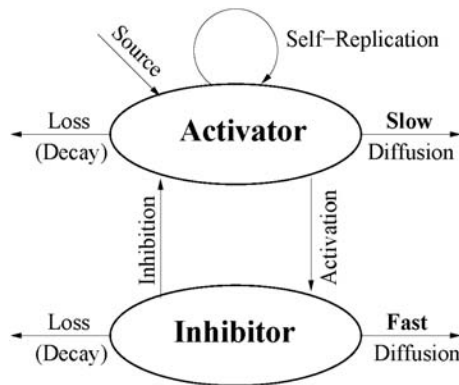


Abbildung 7.1: Schematische Darstellung des Diffusions-Reaktions-Modells zur Erklärung von Turing-Mustern

### 7.2.1 Turing-Muster

§ 751 Als erstes Beispiel wollen wir uns dem Modell zuwenden, das Turing [165] in den frühen 1950ern zur Erklärung biologischer Muster vorgeschlagen hat. Turing hat die Musterbildung als eine Wechselwirkung elementarer Prozesse beschrieben mit dem entscheidenden Ansatz, dass stabile Prozesse sich derartig überlagern können, dass Instabilitäten entstehen. Damit hat Turing am Chaos gerührt – auch wenn er diesen Begriff im Zusammenhang mit der Musterbildung nicht verwendet hat.<sup>2</sup> Auch fehlte Turing zu der Zeit die experimentelle Unterstützung: die Belousov-Zhabotinsky-Reaktion war noch nicht allgemein bekannt. Heute wird das Diffusions-Reaktions-Modell in verschiedenen Wissenschaftsbereichen angewendet – einigen werden wir in den folgenden Unterabschnitten begegnen, eine Übersicht geben [80] und [79], die Anwendung auf biologische Probleme ist z.B. in [98, 97, 107] gegeben.

§ 752 Turings Überlegung startet von einem Aktivator, der mit geringer Reichweite diffundiert, und einem Inhibitor, der über einen weiten Bereich diffundiert. Die beiden chemischen Spezies unterscheiden sich also durch ihre Beweglichkeit im Raum: der Aktivator  $A$  bleibt stets in der Nähe seiner Eintragsstelle während sich der Inhibitor  $H$  über ein weites Volumen verteilt. Die Produktion des Aktivators nimmt mit steigender Konzentration des Aktivators zu, nimmt jedoch in Gegenwart des Inhibitors ab. Die Produktion des Inhibitors hängt zwar von der Konzentration des Aktivators ab, nicht jedoch von der des Inhibitors: der Inhibitor kann sich nicht selbst vermehren. Die zeitliche Änderung in der lokalen Konzentration des Aktivators ist:

$$\frac{\partial A}{\partial t} = \frac{k_{A,1}A}{H} - k_{A,2} + k_{A,1}k_{A,3} + D_1 \frac{\partial^2 A}{\partial x^2} . \quad (7.2)$$

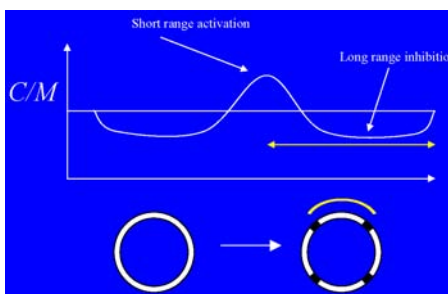
Darin sind die einzelnen Terme von links nach rechts: Produktion des Aktivators aus dem existierenden Aktivator mit einer Produktionsrate  $k_{A,1}$ , Zerfall des Aktivators mit einer Rate  $k_{A,2}$ , Grundproduktion des Aktivators (erfordert keine Anwesenheit des Aktivators) mit einer Produktionsrate  $k_{A,3}$ , sowie die Diffusion des Aktivators mit dem zugehörigen Diffusionskoeffizienten  $D_A$ .

§ 753 Die lokale Änderung des Inhibitors ist beschrieben durch

$$\frac{\partial H}{\partial t} = k_{H,1}A^2 - k_{H,2}H + k_{H,3} + D_H \frac{\partial^2 H}{\partial x^2} . \quad (7.3)$$

<sup>2</sup>Turing und seine Gedanken zur Berechenbarkeit der Welt sind hinreichend bekannt – wahrscheinlich war Chaos für ihn noch kein Begriff im Sinne des heutigen deterministischen Chaos. Und mit Determinismus hat er ohnehin seine Probleme gehabt: mehr als ein Jahrzehnt vor der Beschreibung des Schmetterlingseffekts durch Lorenz [85] hat Turing [164] etwas sehr ähnliches formuliert: “The system of the ‘universe as a whole’ is such that quite small errors in the initial conditions can have an overwhelming effect at a later time. The displacement of a single electron by a billionth of a centimeter at one moment might make the difference between a man being killed by an avalanche a year later, or escaping.”

Abbildung 7.2: Relative Längenskalen bei Turing Mustern und Verteilung der relativen Konzentrationen [158]



Die Produktion des Inhibitors  $H$  erfolgt in einer Reaktion 2. Ordnung mit einer Produktionsrate  $k_{H,1}$ . Zusätzlich zerfällt der Inhibitor mit einer Zerfallsrate  $k_{H,2}$  und wird, ohne Beteiligung des Aktivators, mit einer Produktionsrate  $k_{h,3}$  erzeugt. Die räumliche Ausbreitung ist durch den Diffusionskoeffizienten  $D_H$  bestimmt. Beide Populationen mit ihren Quellen und Senken sind schematisch in Abb. 7.1 dargestellt.

§ 754 Stabile räumliche Muster entstehen unter der bereits gegebenen Voraussetzung der unterschiedlichen räumlichen Beweglichkeiten,  $D_A < D_H$  für  $k_{A,1} \approx k_{A,2} \approx k_{H,2}$ ,  $k_{H,3} \approx k_{A,1}k_{A,3}$  sowie  $k_{H,1} \ll k_{a;1}$ . Beispiele und Erläuterungen zu verschiedenen Mustern gibt [146].

## 7.2.2 Kannibalen und Missionare oder Jäger–Beute Modelle

§ 755 Von Turing selbst stammt eine anschaulichere Formulierung für die beiden Gleichungen, wie sie auch bei Swinton [158] gegeben wird: wir betrachten, wie in den beiden oben gegebenen Gleichungen, ein eindimensionales Modell, in diesem Fall einen langen Strand. Zwei Populationen existieren an diesem Strand: Kannibalen  $C$  und Missionare  $M$ . Diese haben die folgenden Eigenschaften:

- Missionare  $M$  werden rekrutiert und leben zölibatär bis zu ihrem Tod. Für Missionare gibt es demnach eine Produktionsrate, die unabhängig von der Population der Missionare ist, sowie eine Verlustrate.
- Kannibalen sterben ebenfalls, können allerdings Kinder haben. Daher ist ihre Produktionsrate von der Größe ihrer Population abhängig.
- Wenn Missionare und Kannibalen zusammen treffen, werden Kannibalen zu Missionaren konvertiert.

Die zugehörigen Reaktionsgleichungen sind

$$\frac{dM}{dt} = a - M + M^2C \quad \text{und} \quad \frac{dC}{dt} = b - MC^2.$$

Wenn beide Populationen gemischt werden, steuern sie auf ein Gleichgewicht zu. In diesem Modell sind die Kannibalen ein Aktivator, die Missionare ein Inhibitor.

§ 756 Interessanter wird das Modell, wenn sich die beiden Populationen am Strand ausbreiten können. In beiden Fällen erfolgt die Ausbreitung diffusiv, jedoch bewegen sich die Missionare schneller. Die zugehörigen Diffusions–Reaktions–Gleichungen sind

$$\frac{\partial M}{\partial t} = a - M + M^2C + d_M \frac{\partial^2 M}{\partial x^2} \quad \text{und} \quad \frac{\partial C}{\partial t} = b - MC^2 + d_C \frac{\partial^2 C}{\partial x^2}. \quad (7.4)$$

Ein lokaler Überschuss in Kannibalen wird dann automatisch eine größere Menge Kannibalen produzieren, allerdings auch eine größere Zahl Missionare. Damit wird sich ein Muster aus hohem  $C/M$  über kleine Bereiche (kurze Reichweite des Aktivators) und niedrigem  $C/M$  über größere Bereiche (große Reichweite des Inhibitors) ausbilden, siehe Abb. 7.2. Erlaubt man nicht einen unendlichen linearen Strand sondern hat wie bei einer realen Insel periodische Randbedingungen, so ergibt sich, wie in der unteren Zeile angedeutet, ein periodisches Muster.

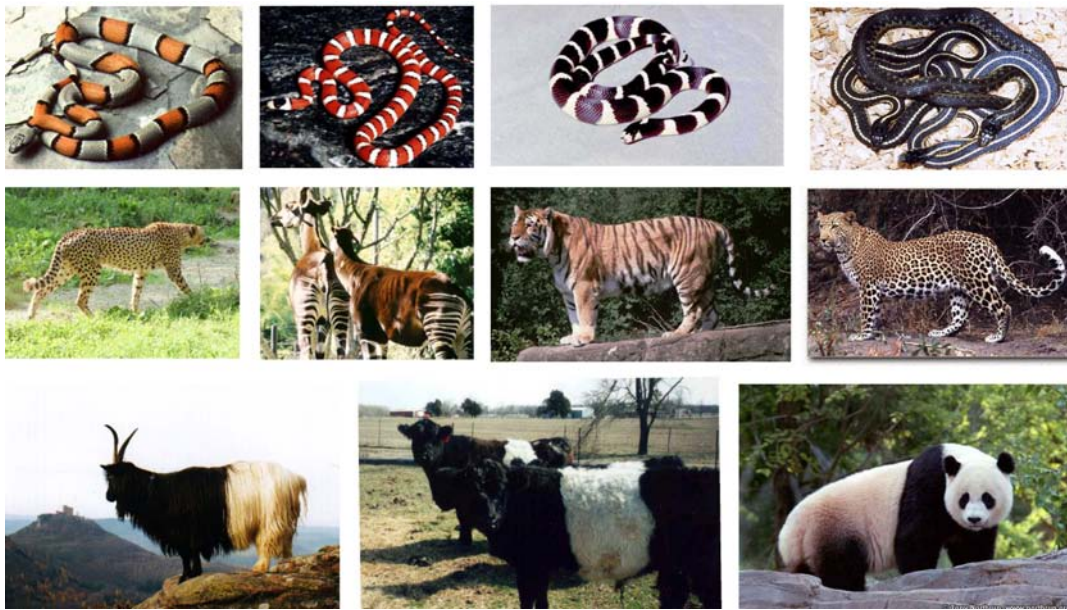


Abbildung 7.3: Beispiele für Muster auf Tierhäuten mit verschiedenen Anzahlen stehender Wellen und unterschiedlichen Geometrien. *Obere Reihe*: verschiedenen Schlangen mit Streifenmuster, *Mittlere Reihe*: Kombination von geflecktem Körper und gestreiften Beinen/Schwanz bei Cheetah, Okapi, Tiger und Leopard (letzterer mit geflecktem Schwanz), *untere Reihe*: Walliser Schwarzkopfschaf als Beispiel für  $\cos(x)$  sowie Galloway und Riesenpanda als Beispiele für  $\cos(2x)$  [146]

§ 757 Von Turing's Missionaren und Kanniblen ist es nur ein kleiner Schritt zur allgemeineren Anwendung des Diffusions-Reaktions-Gleichungssystems in der Form eine Jäger-Beute Modells. Wie bei den Missionaren und Kannibalen lesen wir dabei lediglich die Konzentrationen als Populationsdichten. Ein Beispiel für ein Räuber-Beute-Modell auf dieser Basis ist von Scheffer [135] zur Erklärung der Dichteoszillationen aus Zoo- und Phytoplankton vorgeschlagen worden. Der Übergang von stationären Strukturen zu Oszillationen ist nur eine Frage der Parameter – womit wir auch den Übergang zur Belousov-Zhabotinsky-Reaktion gefunden haben.

### 7.2.3 Hirnareale

§ 758 Die Spezialisierung der einzelnen Hirnareale ist, wie in § 108 angedeutet, ebenfalls ein Problem der Musterbildung. Auch hier wird von einem Aktivator, der seine eigene Reproduktion stimuliert, und einem Inhibitor, der schneller diffundiert, ausgegangen. Einen kurzen Überblick gibt [173], eine genauere Darstellung findet sich in dem Übersichtsartikel von Meinhardt [97].

§ 759 In der Großhirnrinde sind die Turing-Muster jedoch nicht nur im Hinblick auf Strukturbildung sondern auch für die Erklärung der zumindest bei einigen Tieren beobachteten Reparaturfähigkeit interessant: un den Diffusions-Reaktions-Modellen ist ein kritischer Parameter die Größe des morphogenen Feldes, über die das Muster geformt wird, im Vergleich zum Diffusionskoeffizienten als einer Art Maß für den 'Bewegungsradius' der Spezies/Moleküle. Ist das morphogene Feld sehr viel kleiner als die Diffusionslänge, so ergeben sich periodische Muster. Ist die Größe des Feldes dagegen mit der Diffusionslänge vergleichbar, so bildet sich nur ein einzelner Gradient des Morphogens aus. Eine Verkleinerung des Feldes ist, zumindest in Gegenwart eines schwachen Gradienten im Aktivator, mit einer Wiederherstellung

des ursprünglichen Gradienten verbunden – dieser Prozess scheint kritisch für die Regenerationsfähigkeit zu sein.

#### 7.2.4 Turing-Muster erweitert

§ 760 Die Bedeutung der relativen Längenskalen (und damit der Geometrie) zeigt sich bereits in den Tierhäuten: im eindimensionalen Fall bildet sich eine Welle und damit eine periodische Struktur aus. Wird die Linie geschlossen (Kreis in Abb. 7.2), so liegen periodische Randbedingungen vor und es ergibt sich ein periodisches Muster.

§ 761 Auch beim Übergang zu zweidimensionalen Geometrien gelten diese Regeln: periodische Muster und der Einfluss der Randbedingungen. Basierend auf den Arbeiten von Turing hat Murray [107] aus dem Verhältnis der verschiedenen Reaktionskonstanten und den Randbedingungen (z.B. Rechteck oder periodisch wie auf einem Zylinder) einige grundlegende Vorhersagen formuliert:

- Schlangen haben immer gestreifte Muster (Ringe), aber keine Flecken.
- Es gibt keine Tiere mit gestreiftem Körper aber geflecktem Schwanz während es Tiere mit geflecktem Körper und gestreiftem Schwanz gibt.<sup>3</sup>

Derartige Vorhersagen sind für eine Theorie natürlich praktisch, weil es viel Testmaterial gibt (siehe Abb. 7.3). Und sollten Sie einmal eine neue Tierspezies entdecken: gefleckter Körper plus gestreifter Schwanz ist zwar nett aber für die tiefer liegende Wissenschaft langweilig – erst die umgekehrte Variante wäre wirklich interessant.

#### 7.2.5 Selbstorganisation auf Oberflächen

§ 762 Auch Strukturen in nicht-biologischen Systemen lassen sich mit einem Diffusions-Reaktions-Modell beschreiben. Die Belousov-Zhabotinsky-Reaktion ist ein Beispiel für eine oszillierende Lösung. Aber auch stationäre Strukturen lassen sich auf diese Weise beschreiben. Von besonderem Interesse ist dabei die Selbstorganisation von Atomen und/oder Molekülen auf Oberflächen. Übersichten dazu sind in [80] und [177] gegeben, ein ausführlich diskutiertes Beispiel in [181]; für den Spezialfall von Oberflächen mit Terrassen-Struktur siehe [130]. Eine Laudatio auf den mit ähnlichen Themen befassten Träger des Chemie-Nobelpreises von 2007 gibt [71].

#### 7.2.6 Hodgkins-Huxley-Modell eines Neurons

§ 763 Ein Diffusions-Reaktions-Modell findet im zentralen Nervensystem nicht nur in der Organisation Anwendung sondern auch in der Beschreibung eines einzelnen Neurons. Das Hodgkins-Huxley-Modell [43] ist ein auf Untersuchungen am Tintenfisch (giant squid) basierendes empirisches Modell zur Beschreibung der Ausbreitung des Aktionspotentials entlang des Axons. Darin wird der Nervenimpuls als eine nichtlineare wandernde Welle beschrieben, die durch die nichtlineare Reaktion Anregung und die diffusive Ausbreitung des Potentials bestimmt ist. Das Modell ist u.a. in [24, 33, 137, 139] ausführlich beschrieben.

§ 764 Das Membranpotential  $V$  ist bestimmt durch die Diffusions-Reaktions-Gleichung

$$\frac{\partial V}{\partial t} = \frac{1}{rc} \frac{\partial^2 V}{\partial x^2} - \frac{i_{\text{ion}}}{c}. \quad (7.5)$$

Darin ist  $r$  der Widerstand/Länge des Axons und  $c$  dessen Kapazität pro Länge. Die Ionenstromdichte  $i_{\text{ion}}$  ist eine nichtlineare Funktion des Potentials  $V$  und der Zeit  $t$ . Der Ionenstrom

<sup>3</sup>Beispiele sind Zebra und Tiger für den gestreiften Körper kombiniert mit gestreiftem Schwanz; Leopard für gefleckten Körper und gefleckten Schwanz; Giraffe, Genette/Ginsterkatze und Gepard mit geflecktem Körper und gestreiftem Schwanz.

wird getragen durch die durch die Membran gehenden, von Natrium und Kalium getragenen Ionenströme sowie einen Leckstrom:

$$i_{\text{ion}} = i_K + i_{\text{Na}} + i_L .$$

§ **765** In der Beschreibung dieses Ionenstroms wird das Modell etwas unübersichtlich. Die Ströme sind jeweils bestimmt durch eine für die einzelnen Komponenten spezifische Leitfähigkeit und die Potentialdifferenz über die Membran sowie Aktivierungen  $m$  und  $n$  und eine Hemmung  $h$ :

$$i_{\text{Na}} = \bar{g}_{\text{Na}} m^3 h (V - V_{\text{Na}}), \quad i_K = \bar{g}_K n^4 (V - V_K) \quad \text{und} \quad i_L = g_L (V - V_L) .$$

Darin sind die Nernst Potentiale für die einzelnen Ionen gegeben zu  $V_{\text{Na}} = -115$  mV,  $V_K = +12$  mV und  $V_L = -10.613$  mV.<sup>4</sup> Im Gegensatz zur Stromdichte in einem Draht berücksichtigen die Aktivierungen bzw. die Hemmung, dass ein Ionenkanal offen sein muss, damit ein Strom fließen kann. Diese Faktoren können daher als Maß für die Zahl der zur Verfügung stehenden Ionenkanäle interpretiert werden. Diese sind durch Reaktionen 1. Ordnung beschrieben:

$$\begin{aligned} \dot{m} &= \alpha_m (1 - m) - \beta_m m \\ \dot{n} &= \alpha_n (1 - n) - \beta_n n \quad \text{und} \\ \dot{h} &= \alpha_h (1 - h) - \beta_h h . \end{aligned}$$

Die Raten  $\alpha$  und  $\beta$  für das Öffnen oder Schließen der Kanäle sind empirische Funktionen der Spannung für eine Temperatur von  $6.3^{\circ\text{C}}$  [137]:

$$\begin{aligned} \alpha_m &= \frac{0.1 (25 - V)}{\exp\{(25 - V)/10\} - 1}, & \beta_m &= 4 \exp(-V/18), \\ \alpha_n &= \frac{0.01 (10 - V)}{\exp\{(10 - V)/10\} - 1}, & \beta_n &= 0.125 \exp(-V/80), \\ \alpha_h &= 0.07 \exp(-V/20), & \beta_h &= \frac{1}{\exp\{(30 - V)/10\} + 1} . \end{aligned}$$

§ **766** Dieses Modell unterscheidet sich von den anderen Beispielen für die Reaktions-Diffusions-Gleichung in zwei Punkten:

- es ist einfacher in dem Sinne, dass nur eine partielle DGL betrachtet wird, da nur eine Spezies, das Aktionspotential, interessiert.
- die Gleichung ist komplizierter, da Ströme selbst durch Gleichungen beschrieben werden, in denen Koeffizienten durch DGLs bestimmt werden.

Gerade der letzte Punkt, die DGL innerhalb der DGL, macht dieses Modell zu einem attraktiven Kandidaten für die Behandlung in SIMULINK.

§ **767** Eine recht gute Zusammenfassung des Modells, seiner formalen Behandlung und der Dinge, die man damit machen kann, liefert (leider anonym) <http://www.shifz.org/papers/EEG-Simulation.pdf>.

## 7.3 Lösungsverfahren

§ **768** Für die Behandlung von Selbstorganisation und Musterbildung in diffusiv-reaktiven Systemen gibt es zwei verschiedenen Ansätze. Turing's anschaulicher Beschreibung mit Hilfe von Kannibalen und Missionaren legt die Beschreibung des Problems mit Hilfe zellulärer Automaten nahe. Da sich letztere einfach in Java (oder anderen Sprachen wie Logo) implementieren lassen, erfolgt z.B. die Beschreibung der Belousov-Zhabotinsky-Reaktion meist eher mit einem zellulären Automat als mit Hilfe der Differentialgleichung.

<sup>4</sup>Die Genauigkeit im Potential für den Leckstrom erscheint auf den ersten Blick völlig übertrieben: sie wird nur benötigt, um  $V = 0$  für einen stabilen Gleichgewichtszustand fixieren zu können [138].

<sup>5</sup>Große Denker müssen offenbar einen kühlen Kopf bewahren – wahrscheinlich hat es ein Tintenfisch zumindest in tieferem Wasser deutlich einfacher.

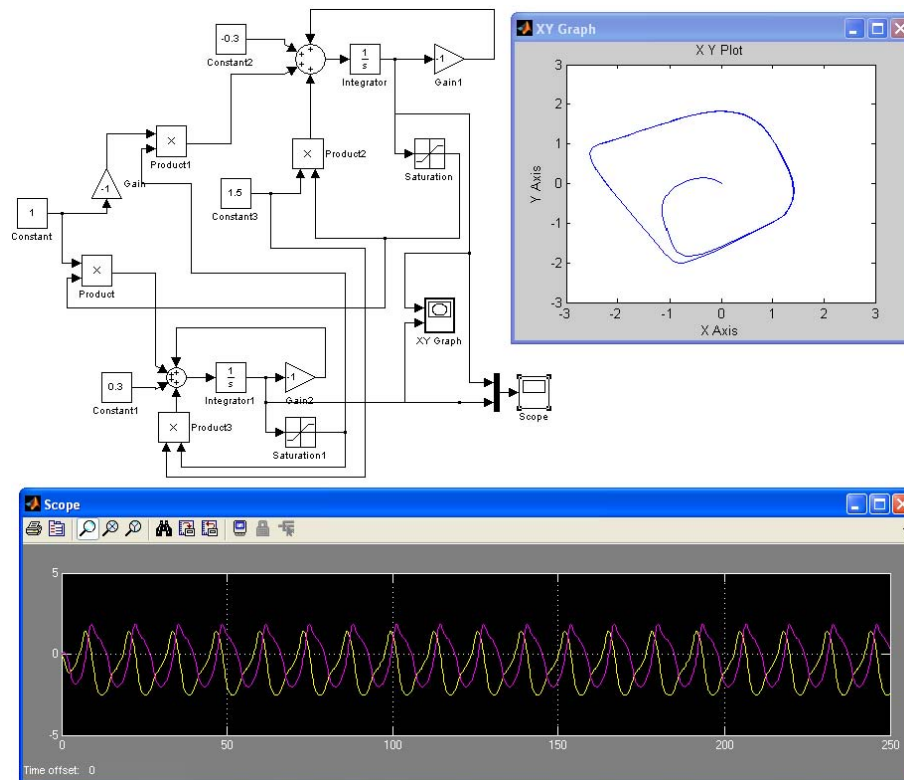


Abbildung 7.4: Central Pattern Generator CPG, auf Basis von [83]

§ 769 Turing selbst hat sich natürlich auf ganz klassische Weise an die Lösung der Diffusions-Reaktions-Gleichungen gemacht: Fourier Transformation liefert die stabilen Moden. Auf Turings Ideen, allerdings in MATLAB integriert, bauen auch Craster und Sassi [17] auf – der technical report der beiden ist damit auch eine interessante Einführung in die numerische Behandlung von Differentialgleichungen mit Hilfe von Spektralverfahren. Sonst gibt es keine analytischen Lösungsmöglichkeiten – auch die Verwendung dimensionsloser Variablen wie in [181] führt nur zu einer Vereinfachung der numerischen Behandlung, nicht aber zu einer analytischen Lösung.

§ 770 Die einzelnen Beispiele für Anwendungen eines Diffusions-Reaktions-Modells unterscheiden sich in ihrer Form jeweils etwas. Wir werden hier, ähnlich dem Vorgehen bei den Beispielen, mit dem einfachsten Modell anfangen und dieses jeweils erweitern.

§ 771 Allen Modell ist jedoch gemein, dass wir partielle DGLs zu behandeln haben: wie wir es vom Separationsansatz her kennen und auch kurz im Zusammenhang mit der Laplace Transformation in § 446 angesprochen haben, reduzieren wir die partielle Differentialgleichung auf gewöhnliche Differentialgleichungen. In diesem Falle verwenden wir, wie von der Laplace Transformation bekannt, eine äußere Zeitschleife, die die Entwicklung der Lösung beschreiben soll. Innerhalb dieser Zeitschleife lösen wir dann für jeden Zeitschritt den räumlichen Transport.

§ 772 Andere Lösungsverfahren wie Spektralverfahren, zellulare Automaten, Finite Elemente Methoden oder Monte Carlo Verfahren existieren selbstverständlich, werden aber im Rahmen dieses Skripts nicht diskutiert: die Verfahren lassen sich zwar auch in MATLAB sinnvoll verwenden, ihre Übertragung auf SIMULINK ist aber nicht mehr hilfreich.



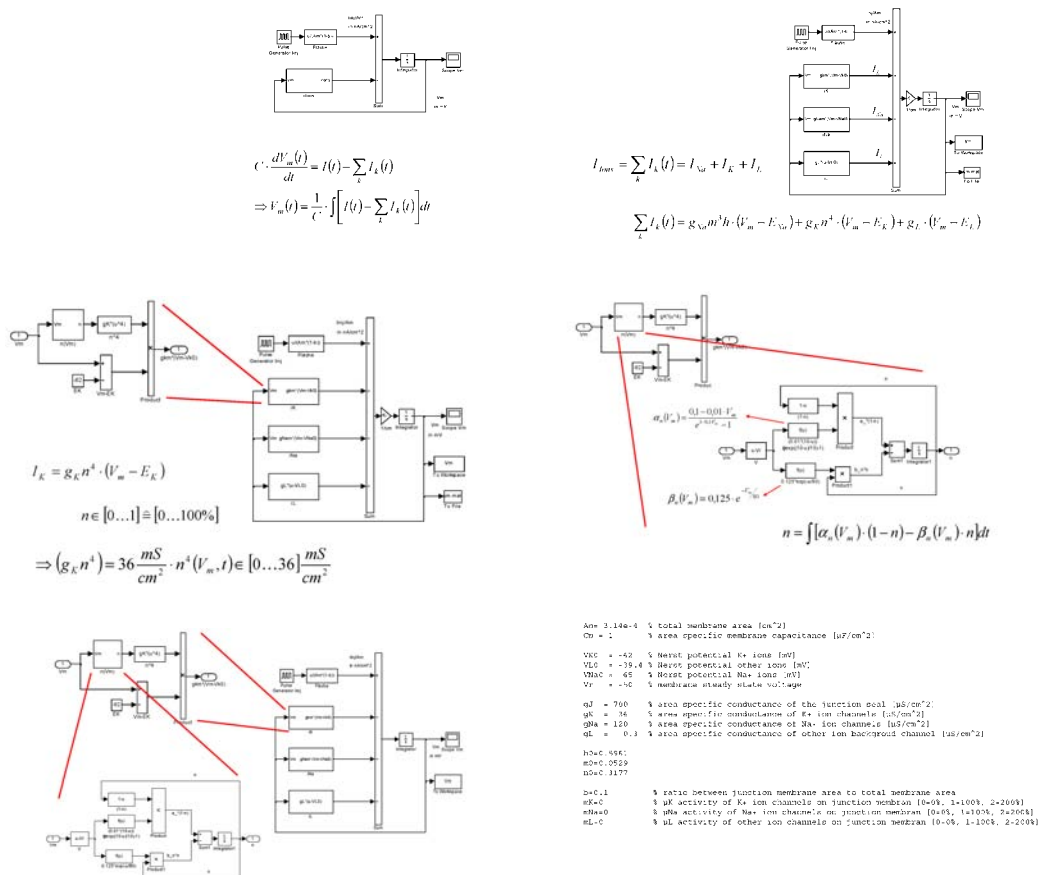


Abbildung 7.5: Hodgkins-Huxley in SIMULINK; aus Eggert et al. [24]

### 7.3.1 Vorübung: Central Pattern Generator CPG

§ 773 Der Central Pattern Generator CPG hat, wie der Name nahe legt, etwas mit der Bildung von Mustern zu tun. Hierbei handelt es sich jedoch nicht um räumliche Muster wie bei Turing sondern um zeitliche: CPGs sind neuronale Netzwerke, die einen rhythmischen Output produzieren können, ohne dass es einen rhythmischen Input oder ein zentrales rhythmisches Element gibt. Derartige Netze sind die Grundlage der meisten rhythmischen (und unbewussten) Bewegungsmuster (für eine Übersicht siehe [49]); der Sinusknoten im Herzen ist eine ähnlicher Taktgeber.

§ 774 Ein recht einfache Satz von Modellgleichungen inklusive des entsprechenden SIMULINK-Modells wird von [83] gegeben – wenn auch mit dem Ziel, dieses System mit Hilfe eines programmierbaren Logik-Bausteins zu realisieren. Das Gleichungssystem

$$\begin{aligned} \dot{x}_1 &= -x_1 + \alpha y_1 + \beta y_2 + i_1 \\ \dot{x}_2 &= -x_2 + \alpha y_1 + \beta y_2 + i_2 \end{aligned}$$

mit  $i_i = \text{const}$  und  $y_i = f(x_i)$  mit  $f(x) = 0.5 * (\text{abs}(x + 1) - \text{abs}(x - 1))$  ist natürlich wesentlich einfacher als ein ein Diffusions-Reaktions-System, da der Diffusionsterm fehlt. Die Letztendlich handelt es sich bei den Gleichungen nur um zwei gekoppelte gewöhnliche Differentialgleichungen mit den vier freien Parametern  $\alpha$ ,  $\beta$ ,  $i_1$  und  $i_2$ .

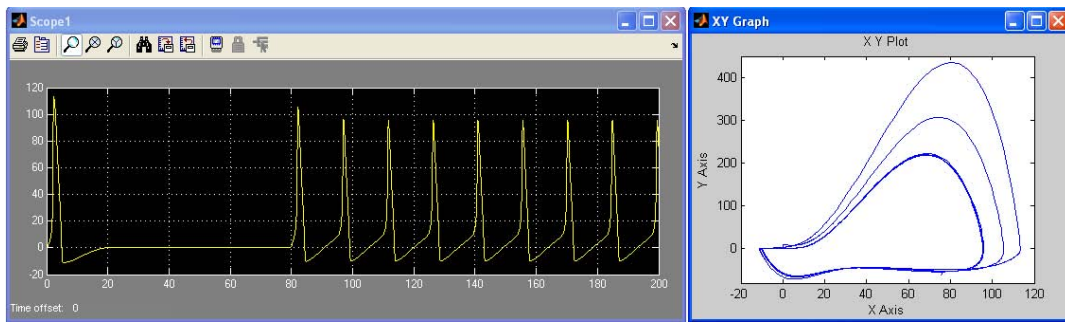


Abbildung 7.6: Aktionspotentiale eines Hodgkins–Huxley Neurons gegen die Zeit (*links*) und im Phasenraum (*rechts*)

§ 775 Abbildung 7.4 zeigt die von [83] vorgeschlagene SIMULINK-Implementierung des CPG-Modells in einer ziemlich direkten Umsetzung des Gleichungssystems. Die einzigen nicht-arithmetischen Elemente sind pro Gleichung der Integrator und der Sättigungsblock (mit den Grenzen  $-1$  bzw.  $+1$ ). Diese bestimmen das Verhalten des Systems. Die freien Parameter sind, wie leicht zu erkennen, gewählt als  $\alpha = 1.5$ ,  $\beta = 1$ ,  $i_1 = 0.3$  und  $i_2 = -0.3$ . Im Phasenraum erkennt man den limit cycle, d.h. das System strebt schnell in eine stabile Oszillation – auch wenn diese sicherlich nicht harmonisch ist, wie durch die Zeitverläufe deutlich wird.

**Zwischenrechnung 51** Basteln Sie eine kompaktere mathematische Formulierung für den CPG – und ebenso das kompaktere SIMULINK-Modell.

### 7.3.2 Hodgkins–Huxley

§ 776 Jetzt wird es etwas unübersichtlicher: versuchen wir, das Hodgkins–Huxley-Modell in SIMULINK zu implementieren. Hier können wir auf ein bestehendes Modell von Murat Saglam zurück greifen (<http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=19669&objectType=file>) oder auf die Arbeiten von Issing [59] (für das Modell <http://mikro.ee.tu-berlin.de/forschung/bioelectronic/myobip/downloads/model.zip>) oder die Vorlesung von Eggert et al. [24]. Genauso wie beim CPG geht es hier nicht um stabile Muster sondern um Oszillationen – in diesem Fall um integrate and fire Oszillationen wie beim van der Pol Oszillator. Wie beim CPG sollte sich im Phasenraum allerdings ein limit cycle einstellen.

§ 777 In Abb. 7.5 wird ausgehend von der Präsentation von Eggert et al. [24] sehr anschaulich der Aufbau des SIMULINK Modells erläutert. Im linken oberen Teilbild ist die äußere Schleife gegeben: diese enthält den Integrator und entspricht der Zeitschleife in einer konventionellen Finite Differenzen Codierung. Der Pulsgenerator im linken oberen Teil erzeugt externe Stimuli, der Block links unten ist ein Subsystem, in dem der entsprechende Ionenstrom bestimmt wird. Da sich dieser aus drei Spezies zusammensetzt, werden entsprechend drei Blöcke benötigt, wie im rechten oberen Teilbild von Abb. 7.5 gezeigt.

§ 778 Im linken mittleren Teilbild ist die obere Schicht des den Kalium-Ionenstrom  $I_K$  repräsentierenden Untersystems gezeigt. Dieses enthält neben einigen einfachen arithmetischen Operationen und der Definition einer Konstanten ein Untersystem (rechtes Teilbild in der mittleren Reihe), in dem die Differentialgleichung für die zugehörige Reaktionsrate integriert wird. Die untere Zeile fasst im linken Teil noch einmal die Elemente des Modells zusammen; im rechten Teil sind die Steuerparameter gegeben.

§ 779 Abbildung 7.6 zeigt das Aktionspotential für ein derartiges Hodgkins–Huxley-Neuron in der Zeitdomäne (*links*) und im Phasenraum. Das Feuern ist im Zeitverlauf deutlich zu erkennen, ebenso der sich einstellende limit cycle im Phasenraum.

Verständnisfrage **71** Und wo verbirgt sich der diffusive Term?

## Fragen

**Frage 32** Erläutern Sie kurz die inhaltlichen und formalen Grundprinzipien von Diffusion. Wie kommt man zu eine Diffusionsgleichung? Geben Sie anschauliche Beispiele für Diffusion.

## Kurzprojekte

§ **780** Die Projekte sind etwas härter und gehen teilweise an die Grenze zur Bachelor-Arbeit. Nehmen Sie das als Motivation – und sehen Sie darin eventuell sogar Ihre Chance für das Studium.

### Netze von Hodgkin–Huxley Neuronen

§ **781** Ein ähnliches SIMULINK-Modell für das Hodgkin–Huxley Neuron ist zusammen mit einer übersichtlichen Erklärung und Hinweisen für die Modellierung mehrere gekoppelter Neuronen in [24] gegeben. Bauen Sie sich ein kleines neuronales Gitter und untersuchen Sie dieses z.B. auf Synchronization. Können Sie die HH-Neuronen auch verwenden, um ein neuronales Gitter wie in der Informatik als Lernmodell gebräuchlich zu konstruieren? Für die Kopplung von Neuronen ist auch [33] hilfreich Trappenberg [163] und O’Reilly and Munakata [110] geben viele Hinweise zur Kopplung von Neuronen – Trappenberg verwendet dabei auf MATLAB.

### Jäger–Beute Modell

§ **782** Weniger empfindlich auf die Wahl der Parameter als die Turing Muster sind Jäger–Beute Modelle: bei Turing-Mustern möchte man stabile Muster wie auf den Tierhäuten haben während ein Jäger–beute Modell keinen stationären Zustand erreichen muss sondern auch oszillieren oder chaotisch sein darf. Ein Beispiel für verschiedene MATLAB-Implementationen finden Sie unter [http://www.uoguelph.ca/~mgarvie/PredPrey\\_files/fd1d.html](http://www.uoguelph.ca/~mgarvie/PredPrey_files/fd1d.html) für den eindimensionalen und unter [http://www.uoguelph.ca/~mgarvie/PredPrey\\_files/fd2d.html](http://www.uoguelph.ca/~mgarvie/PredPrey_files/fd2d.html) für den zwei-dimensionalen Fall; auch [17] bietet Beispiele und Code in MATLAB und Fortran. Implementieren Sie ein entsprechendes Modell in sc SimuLink. Überlegen Sie sich, wie sie geschickt die Möglichkeiten von SIMULINK ausnutzen können; die offensichtliche Lösung, eine s-Function oder ein m-file mit der konventionellen finiten Differenzen Lösung zu basteln kann es irgendwie nicht sein.

### Turing Muster in SIMULINK

§ **783** Eigentlich nichts anderes als das Jäger–Beute Modell: allerdings sollten die Parameter so gewählt werden, dass ein stabiles Muster entsteht.

**Teil IV**

**Diskrete Systeme**

## Etwas Spieltheorie

Der Mensch lebt nicht nur sein persönliches Leben als Einzelwesen, sondern, bewusst oder unbewusst, auch das seiner Epoche und Zeitgenossenschaft, und solle er die allgemeinen und unpersönlichen Grundlagen seiner Existenz auch als unbedingt gegeben und selbstverständlich betrachten und von dem Einfall, Kritik daran zu üben, so weit entfernt sein, wie der gute Hans Castorp es wirklich war, so ist doch sehr wohl möglich, dass er sein sittliches Wohlbefinden durch die Mängel vage beeinträchtigt fühlt. Dem einzelnen Menschen mögen mancherlei persönliche Ziele, Zwecke, Hoffnungen, Aussichten vor Augen schweben, aus denen er der Impuls zu hoher Anstrengung und Tätigkeit schöpft; wenn das Unpersönliche um ihn her, die Zeit selbst der Hoffnungen und Aussichten bei aller äußeren Regsamkeit im Grunde entbehrt, wenn sie sich ihm als hoffnungslos, aussichtslos und ratlos heimlich zu erkennen gibt und er bewusst oder unbewusst gestellten, aber doch irgendwie gestellten Frage nach seinem letzten, mehr als persönlichen, unbedingten Sinn aller Anstrengung und Tätigkeit ein hohles Schweigen entgegensetzt, so wird gerade in Fällen redlicheren Menschentums eine gewisse lähmende Wirkung solches Sachverhalts fast unausbleiblich sein, die sich auf dem Weg über das Seelisch-Sittliche geradezu auf das physische und organische Teil des Individuums erstrecken mag. Zu bedeutender, das Maß des schlechthin gebotenen überschreitener Leistung aufgelegt zu sein, ohne dass die Zeit auf die Frage Wozu? eine befriedigende Antwort wüsste, dazu gehört entweder eine sittliche Einsamkeit und Unmittelbarkeit, die schon selten vorkommt und heroischer Natur ist, oder eine sehr robuste Vitalität.

Thomas Mann, Zauberberg

§ 784 Nein – es geht nicht um die Theorie des Glücksspiels, auch wenn sich das vorangegangene Kapitel mit dem Zufall beschäftigt hat. Spieltheorie wird eigentlich eher in den Wirtschafts- und Sozialwissenschaften verwendet, so z.B. in der Diskussion im Strategien zur Reduktion des anthropogenen Treibhauseffekts. Allerdings lassen sich viele Strategieplanungen ebenfalls spieltheoretisch beschreiben (z.B. die Mittelvergabe an Universitäten, die Steuerung von Wissenschaft durch diese Mittelvergabe usw.). Und im Bezug auf SIMULINK bietet die Spieltheorie noch einmal eine andere Möglichkeit, mit diskreten Systemen zu arbeiten: es finden mehrfache Wechselwirkungen statt (mehrere Runden) und diese erfolgen nach festen Regeln.

§ 785 In diesem Kapitel werden die Grundbegriffe der Spieltheorie am Beispiel des Gefangenendilemmas (und logisch isomorpher Spiele) sowie der Gemeindewiese beschrieben. Anschließend wird eine Kopplung eines Klimamodells (beschrieben durch seine Impulsantwort) zur Bestimmung eines optimalen Emissionspfades eingeführt; die Aufteilung des letzteren

auf verschiedene Akteure wird unter spieltheoretischen Gesichtspunkten diskutiert. Am Ende des Abschnitts sind die Ergebnisse der studentischen Projekte zum Gefangenendilemma sowie einige der von den Studierenden entwickelten SIMULINK-Modelle zusammen gefasst.

**Lernziele:** nach Durcharbeiten dieses Kapitels sollen Sie in der Lage sein,

- einige relevante Begriffe der Spieltheorie zu erläutern, wie z.B. Nullsummenspiel, Nash Gleichgewicht.
- einfache Strategien so zu formulieren, dass sich die Regeln in SIMULINK implementieren lassen und die Strategien gegeneinander spielen können.
- die logischen Blöcken in SIMULINK einzusetzen.
- eine Simulation mit diskreten Zuständen durchzuführen.

## 8.1 Gefangene, Kühe und Stichlinge

§ 786 Die Spieltheorie liefert eine formale Beschreibung von Strategien zur Findung von Entscheidungen in Systemen mit mehreren Beteiligten/Akteuren. Die Beschreibung solcher Situationen mit Hilfe der Spieltheorie ist jedoch nur dann sinnvoll, wenn diese Entscheidungen mehrfach getroffen werden müssen, d.h. wir betrachten wiederholte Spielrunden und damit lange Zeiträume. Interessant werden diese sich wiederholenden Runden dadurch, dass die Entscheidung, die den persönlichen Vorteil maximiert, für die Gesamtheit der Beteiligten suboptimal ist. Umgekehrt ist die Entscheidung, die für die Gesamtheit optimal ist, für den Einzelnen suboptimal. Oder ganz einfach: Verzicht aufs Auto mag fürs Klima (und damit für die Gesamtheit der Spieler) besser sein, für mich ist es aber mit Einschränkungen verbunden.<sup>1</sup>

§ 787 Zur Entscheidungsfindung in einem sich wiederholenden Spiel kann jeder Spieler eine persönliche Strategie entwickeln. Diese lassen sich in zwei verschiedenen Formen unterteilen. Bei einer reinen Strategie legt der Spieler sein Vorgehen derart fest, dass in einer vorgegebenen Situation nach diesem Prinzip immer der gleiche Schritt erfolgt – er macht sich für den Gegner berechenbar. Bei einer gemischten Strategie dagegen schreibt der Spieler jeder Handlungsmöglichkeit eine Wahrscheinlichkeit zu; auf der Grundlage dieser Wahrscheinlichkeiten wird gehandelt, die Entscheidung selbst wird vom Zufall bestimmt. Die reine Strategie ist der Grenzfall der gemischten.

§ 788 Wikipedia [http://en.wikipedia.org/wiki/Game\\_theory](http://en.wikipedia.org/wiki/Game_theory) liefert eine über meine enge Auffassung von Spieltheorie weit hinaus gehende Definition:

Game theory is a branch of applied mathematics that is often used in the context of economics. It studies strategic interactions between agents. In strategic games, agents choose strategies which will maximize their return, given the strategies the other agents choose. The essential feature is that it provides a formal modeling approach to social situations in which decision makers interact with other agents. Game theory extends the simpler optimization approach developed in neoclassical economics.

The field of game theory came into being with the 1944 classic *Theory of Games and Economic Behavior* by John von Neumann and Oskar Morgenstern. A major center for the development of game theory was RAND Corporation where it helped to define nuclear strategies.

Game theory has played, and continues to play a large role in the social sciences, and is now also used in many diverse academic fields. Beginning in the 1970s, game theory has been applied to animal behavior, including evolutionary theory. Many games, especially the prisoner's dilemma, are used to illustrate ideas in political science and ethics. Game theory has recently drawn attention from computer scientists because of its use in artificial intelligence and cybernetics.

<sup>1</sup>In der Diskussion um die Einschränkung der Folgen eines anthropogenen Klimawandels haben sich daher auch verschiedenen spieltheoretische Ansätze ausgebildet; die ersten davon gehen auf Hasselmann [39, 37, 38] zurück.

In addition to its academic interest, game theory has received attention in popular culture. A Nobel Prize-winning game theorist, John Nash, was the subject of the 1998 biography by Sylvia Nasar and the 2001 film *A Beautiful Mind*. Game theory was also a theme in the 1983 film *WarGames*. Several game shows have adopted game theoretic situations, including *Friend or Foe?* and to some extent *Survivor*. The character Jack Bristow on the television show *Alias* is one of the few fictional game theorists in popular culture.[1]

Although some game theoretic analyses appear similar to decision theory, game theory studies decisions made in an environment in which players interact. In other words, game theory studies choice of optimal behavior when costs and benefits of each option depend upon the choices of other individuals.

### 8.1.1 Hofstaedter's Einmillionendollarspiel

§ 789 Douglas Hofstaedter, vielleicht eher bekannt durch 'Gödel, Escher, Bach' und andere Bücher [45, 46, 47] hat für seine Kolumne in der Zeitschrift *Scientific American* ein recht einfaches Spiel entwickelt, um die Grundkonzepte der Spieltheorie zu erläutern. Dazu entwickelt er folgende Spielsituation: der *Scientific American* verlost unter seinen Lesern einen Preis von 1 Mio \$ – allerdings hängt der Preis von der Zahl  $N$  der Bewerber ab: es gewinnt nur einer, der aber  $1/N$  Mio \$. Die Leser sind aufgefordert, eine sinnvolle Strategie zu finden – wobei sich die Frage stellt, was eine sinnvolle Strategie ist und für wen diese Strategie sinnvoll ist.

§ 790 Die Situation unterscheidet sich sowohl von einem typischen Spiel wie Skat oder Knobel als auch von einer typischen Auslosung. Normalerweise verringern bei letztere Mitbewerber zwar die Chance des Einzelnen auf den Gewinn, nicht aber den Gewinn; bei ersterem ist der Gewinn des einen Spielers gleich den Verlusten der anderen. Stattdessen ist diese Spielsituation ein gutes Beispiel für den Konflikt zwischen den Interessen des Einzelnen und denen der (Spieler-)Gemeinschaft.

§ 791 Dieser Interessenkonflikt wird deutlich, wenn wir die Frage nach dem Ziel einmal aus Sicht des einzelnen Spielers, einmal aus Sicht der Spielergemeinschaft stellen. Der spieltheoretische Background ist ein einfacher: es geht nicht um den Gewinn des einen als den Verlust des anderen (Nullsummenspiel) sondern um das Auffinden einer Strategie im gemeinsamen Kampf um optimale Ressourcen.

§ 792 Im Kopf des Einzelspielers bildet sich (ungefähr) die folgende Gedankenkette:

- (1) Ich möchte die Million gewinnen, also muss ich spielen.
- (2) Aber mit jedem zusätzlichen Spieler sinken meine Chancen und der Gewinn wird kleiner.
- (3) Also sollte ich nicht spielen, da sonst ein Großteil des Geldes, nämlich der  $(N - 1)/N$ te Teil, beim Verlag verbleibt.
- (4) Aber wenn alle so argumentieren, gewinnt kein Spieler sondern der Verlag und um das zu verhindern, sollte ich spielen.

Um die Ressourcen (den Gewinn) nicht ganz verfallen zu lassen, kommt der Einzelspieler zu dem Schluss, dass auch ein kleiner Gewinn besser ist als wenn das Geld beim Verlag verbleibt.

§ 793 In der Spielergemeinschaft dagegen bildet sich dagegen die folgende Gedankenkette aus:

- (1) Wir haben ein gemeinsames Ziel: die Million (die Ressourcen) muss gewonnen werden.
- (2) Diese kann aber nur dann gewonnen werden, wenn es genau einen Teilnehmer gibt: keiner Teilnehmer macht den Verlag zum Gewinner, mehrere schmälern den Gewinn für die Spielergemeinschaft und machen damit den Verlag zum Gewinner.

Damit hat die Spielergemeinschaft ein klares Ziel: genau einer soll spielen. Aber wer? Schließlich ist der eine ja auch der Gewinner. Die Spielergemeinschaft kann z.B. auf folgendes Verfahren kommen: die  $N$  Interessenten zur Teilnahme am Spiel haben jeweils einen  $n$ -seitigen Würfel und nur Spieler, die die 1 würfeln, dürfen mitspielen. In diesem Fall hat die Spielergemeinschaft eine optimale gemischte Strategie gefunden: sie enthält Zufallsentscheidungen und maximiert den Gewinn für die Gesamtheit der Spieler. Die Spielergemeinschaft hätte sich auch auf eine andere Strategie zurück ziehen können: schreibe alle Zahlen von 1 bis  $N$  auf Lose und lasse jeden Interessierten ein Los ziehen – dann hätte die Spielergemeinschaft eine Art Vorauslosung vorgenommen, die sich vom eigentlichen Spiel nur dadurch unterscheidet, dass der Nachteil der Gewinnverkleinerung bei mehreren Spielern entfällt, da diese erst in einer zweiten Runde (eben beim Verlag) wirksam wird.

§ 794 Experimentell ist das Spiel leider nie durchgeführt worden – da wollte offenbar keiner die Ressourcen opfern. Stattdessen wurde das Spiel in veränderter Form veröffentlicht: jeder Teilnehmer konnte in einem einzigen Brief angeben, wie oft er teilnehmen wollte. Das Ergebnis:

1	2	3	4	5	6	7	8	9	10	100	1000
1133	31	16	8	16	0	9	1	1	49	61	46

sowie ‘Spielverderber’ bis hin zu  $10^{100}$ . Etwas irritierend ist die große Zahl von Teilnehmern, die mehrfach teilnehmen würden; entweder nach dem Motto ‘bevor jemand anders viel gewinnt, versuch ich zumindest wenig zu gewinnen’ oder weil sie das Spielprinzip nicht verstanden haben.

### 8.1.2 Nullsummenspiele

**Definition 14** *Beim Nullsummenspiel ist der Gewinn des einen Spielers der Verlust des anderen.*

§ 795 Die meisten Gesellschaftsspiele sind Nullsummenspiele. Das einfachste Beispiel ist ein klassisches Spiel, der Münzwurf: bei Kopf gibt A 10 Cent an B, bei Zahl umgekehrt. Aber von Doppelkopf über Mensch-Ärgere-Dich-nicht und Schach bis hin zu Go ist stets der Gewinn des einen gleichbedeutend mit dem Verlust des anderen Spielers, d.h. es gibt Gewinner und Verlierer.

§ 796 Nullsummenspiele sind jedoch nicht auf klassische Spiele im alltäglichen Sinn beschränkt. Begrenzte Ressourcen führen auf Nullsummenspiele. Bei gedeckelten öffentlichen Haushalten sind auch ein Hochschuloptimierungskonzept und die leistungsorientierte Mittelvergabe Nullsummenspiele: wenn Osnabrück die Physik behält, muss halt eben Clausthal auf die Physik verzichten. Nullsummenspiele haben im Gegensatz zu dem oben beschriebenen Millionenspiel und zu Situationen, in denen der in § 786 beschriebene Zielkonflikt auftreten kann, eine Einschränkung: bei Nullsummenspielen kann Kooperation keinen Gewinn für die Spielergemeinschaft erreichen! Clausthal und Osnabrück möchten beide ihre Physik behalten und können sich beide gut darstellen, sie können sich u.U. sogar als eine gute Gemeinschaft verkaufen: da aber das Land die Zahl der Physikstudienplätze auf Grund mangelnder Nachfrage reduzieren will, ist einer von den beiden zuviel. Im Prinzip kann man in dieser Situation auch durch Münzwurf, also das klassische Nullsummenspiel, zu einer Entscheidung kommen.

### 8.1.3 Das Gefangenendilemma – Einzelspiel

§ 797 Das Gefangenendilemma ist ein klassisches Beispiel für ein Nicht-Nullsummenspiel mit einem Zielkonflikt. Es ist gleichzeitig auch das Musterbeispiel für die Bedeutung von Kooperation [2, 127] – und illustriert ein Problem der Kooperation besonders deutlich: Kooperation ist bei Vertrauen leichter möglich als in einer Situation des Misstrauens oder einfach des fehlenden Vertrauens im neutralen Sinne von nicht-Kennen.



§ 798 In der von Mérö [99] gegebenen Form liest sich das Gefangenendilemma folgendermaßen:

Die Polizei fängt ein langgesuchtes Duo, dem ein schweres Verbrechen angelastet wird. Die Polizei hat keine direkten Beweise für die Schuld der beiden Männer und kann ihnen lediglich Fahren mit überhöhter Geschwindigkeit nachweisen. Der Untersuchungsrichter möchte den Fall abschließen und unterbreitet dazu jedem der Gefangenen, die er in getrennte Zellen legen ließ, den folgenden Vorschlag:

“Wenn Sie das Verbrechen gestehen und uns dadurch helfen, den Fall schnell zu klären, lasse ich Sie frei und wir vergessen die Sache mit dem zu schnellen Fahren. Ihren Komplizen lochen wir dann für 10 Jahre ein und die Sache ist für immer erledigt. Das Angebot gilt jedoch nur, falls Ihr Komplize das Verbrechen nicht gesteht und uns bei der Aufklärung also nicht hilft. Wenn er ebenfalls gesteht, ist ihr Geständnis nicht viel wert, weil wir dann ohnehin Bescheid wissen. In diesem Fall werden Sie beide zu je 10 Jahren Gefängnis verurteilt. Wenn keiner von Ihnen gesteht, ist das Ihre Sache, aber in jedem Fall wird diese fürchterliche Raserei streng geahndet werden, und Sie kommen beide für ein Jahr hinter Schloss und Riegel. Noch etwas: Ihrem Komplizen habe ich das gleiche gesagt wie Ihnen. Ich erwarte Ihre Antwort morgen früh um zehn Uhr – um elf Uhr können Sie frei sein.”

§ 799 Die Spielausgänge hängen von den voneinander unabhängigen Entscheidungen der beiden Gefangenen ab und lassen sich in einer Gewinnmatrix zusammen fassen:

	K1 gesteht	K1 gesteht nicht
K2 gesteht	<b>-5</b> , -5	<b>-10</b> , 0
K2 gesteht nicht	<b>0</b> , -10	<b>-1</b> , -1

Darin bedeuten negative Zahlen einen Verlust. Im Fettdruck gegeben ist das Ergebnis für den Gefangenen K1. Aus der Gewinnmatrix wird das Problem des Spiels, der Zielkonflikt der einzelnen Spieler, deutlich: das optimale Ergebnis für den Einzelnen (keine Haft) ist nicht verträglich mit dem optimalen Ergebnis für die Gemeinschaft (Gesamthaftzeit von 2 Jahren, verglichen mit einer Gesamthaftzeit von 10 Jahren in allen anderen Fällen). Das Gefangenendilemma ist also kein Nullsummenspiel: die Summe der Gewinne und Verluste ist ungleich Null. In diesem Fall geht es nur um Verlustminimierung; für den einzelnen Spieler oder für die Gruppe der Spieler.

§ 800 Wie findet in dieser Situation jeder der Gefangenen seine Strategie? Eine Kommunikation zwischen beiden ist nicht möglich, d.h. jeder Gefangene kann zwar über die Gedanken des anderen spekulieren, kann aber keinerlei Aussage von diesem erhalten. Nehmen wir ferner Voraussetzung an, dass es keine besondere emotionale Bindung zwischen beiden gibt, d.h. dass keiner von den beiden irgendeinen Grund zu der Annahme hat, dass der zweite Gefangene ein besonderes Interesse am Wohlergehen des ersten Gefangenen (oder umgekehrt) hat. Dann hängt jeder der Gefangenen ungefähr dem folgenden Gedankengang nach:

- (1) Singt mein Partner, so bekomme ich 5 Jahre, wenn ich auch singe, aber 10 Jahre, wenn ich leugne. Singt mein Partner aber nicht, so komme ich frei, wenn ich singe, aber ich kriege 5 Jahre, wenn ich leugne.
- (2) In beiden Fällen ist meine Strafe geringer, wenn ich singe, also singe ich.

Diese Logik diktiert beiden Gefangenen zu singen – auch wenn das Gesamtergebnis damit nicht optimal ist. Aber das Singen ist die Strategie, die unabhängig von der Strategie des anderen Spielers, den eigenen Gewinn maximiert (Nash-Gleichgewicht).

§ 801 Allerdings kommen den Gefangenen bei dem Versuch einer Bewertung des Spielausgangs weitere Gedanken:

- (3) Ich bin meinem Partner zwar nicht emotional verbunden, weiß aber, dass er ebenfalls intelligent ist. Mein Partner wird zu dem gleichen Schluss kommen, dass

Singen das Beste ist.

- (4) Und mein Partner wird ebenfalls erkennen, dass es nicht die optimale Lösung ist – und dass ich es ebenfalls weiß.  
 (5) Daher wird mein Partner nicht singen – und ich werde es auch nicht.

Dieser Schritt ist eigentlich schon ganz gut: damit ließe sich der Gesamtgewinn optimieren (bzw. in diesem Fall eher der Gesamtverlust minimieren), jedoch um den persönlichen Preis von einem Jahr Gefängnis.

§ 802 Und dann kommen die Hintergedanken und jeder Gefangene versteht sein Dilemma noch deutlicher:

- (6) Und wenn mein Partner weiter denkt und merkt, dass ich auf der Basis dieser Logik nicht singe – dann kann er durch singen seinen Gewinn optimieren.

Damit ist jeder der Gefangenen wieder in der gleichen Situation wie am Anfang: er hat zwar erkannt, was die beste Strategie für die beiden Gefangenen gemeinsam wäre und dass dieser Strategie ein Zielkonflikt innewohnt, hat aber keinen Grund, dem anderen ebenfalls diesem Zielkonflikt ausgesetzten Gefangenen zu trauen.

§ 803 Die möglichen Strategien im Gefangenendilemma lassen sich mit zwei Begriffen beschreiben: Kooperation oder Konkurrenz.

**Definition 15** Kooperation: *Der Spieler wählt den Zug, der den Gewinn für die Gemeinschaft maximiert – aber gleichzeitig das größte persönliche Risiko beinhaltet.*

**Definition 16** Konkurrenz: *Der Spieler wählt den Zug, der den größten eigenen Vorteil gestattet – auch wenn der Gesamtgewinn dabei suboptimal ist.*

§ 804 Führt man das Gefangenendilemma experimentell durch, so zeigt sich, dass ungefähr 40% der Probanden eine kooperative Strategie wählen. Oder umgekehrt: es wird häufiger eine konkurrierende als eine kooperierende Strategie gewählt. Wird das Spiel jedoch nicht nur einmal, sondern mehrfach in Folge gespielt, so steigt die Zahl der Kooperativen auf bis zu 60% (*die Zukunft wirft ihre Schatten voraus*, s.u.). Eine mögliche Begründung ist die Risikovermeidung (bloß keine 10 Jahre Knast). Wirklich interessant und auch breiter anwendbar wird das Gefangenendilemma erst bei mehrfachem Spielen – was auch anschaulich vorstellbar ist, da ein mehrfaches Spiel die Möglichkeit eröffnet, den anderen Gefangenen einzuschätzen und die Idee von Kooperation (oder Konkurrenz) zu entwickeln.

§ 805 Um die Folgerungen aus dem Gefangenendilemma auf andere Situationen übertragen zu können, müssen diese einen vergleichbaren Zielkonflikt bieten. Derartige Situationen bezeichnet man als isomorphe Spiele. Dabei bedeutet logisch isomorph, dass sich die Spielsituation durch eine ähnliche Gedankenkette wie beim Gefangenendilemma und eine ähnliche Gewinnmatrix beschreiben lässt. Das Wettrüsten<sup>2</sup> ist ein verwandtes Problem: beim Wettrüsten ist der Gewinn die militärische Macht, der Einsatz die mit der Rüstung verbundenen Kosten. Die Gewinnmatrix mit 1 als dem schlechtesten und 4 als dem besten Ergebnis ist der des Gefangenendilemmas vergleichbar:

	Blau rüstet wenig	Blau rüstet stark
Rot rüstet wenig	2, 2 billiges Gleichgewicht	1, 4 Überlegenheit
Rot rüstet stark	1, 4 Wehrlosigkeit	3, 3 teures Gleichgewicht

<sup>2</sup>Das Gefangenendilemma ist nicht das Ausgangsspiel. Das Ausgangsspiel war das Wettrüsten (daher wird RAND ja auch von Wikipedia im Zusammenhang mit der Spieltheorie genannt). Das Gefangenendilemma ist nur eine neutrale Illustration um auf das Problem von Kooperation versus Konkurrenz hinzuweisen ohne sofort die gesamte Rhetorik und den gesamten emotionalen Ballast des Kalten Krieges aufzurufen. Spieltheorie wurde in den späten 1970ern und frühen 1980ern vor dem Hintergrund des Nato-Doppelbeschlusses und Star Wars (Reagans geplante Raketenabwehr im All) mit dem Gefangenendilemma populär, z.B. [2, 127]. Letztendlich hat sich die Gewinnmatrix schon vor dem Ende der Sowjetunion bewährt: das billigere Gleichgewicht wurde z.B. schon durch SALT langsam angestrebt.

§ 806 Ein anderes Beispiel ist die Preisgestaltung an der Zapfsäule – im Gegensatz zum Wettrüsten ein Beispiel, das nichts an seiner Aktualität verloren hat. Dazu betrachten wir die folgende Situation: zwei benachbarte Tankstellen dürfen einmal im Monat (oder in einem anderen Zeittakt) ihre Preise neu festsetzen. Geringerer Preis heißt weniger Gewinn pro Liter aber möglicherweise mehr Kunden. Die Gewinnmatrix ist

	Blau senkt Preis	Blau senkt Preis nicht
Rot senkt Preis	0, 0 geringer Verlust	4, -3 zusätzlicher Gewinn
Rot senkt Preis nicht	-3, 4 großer Verlust	1, 1 normaler Gewinn

Das Spiel wiederholt sich in den Folgemonaten (bzw. Folgetakten), falls nicht einer oder beide Pleite gegangen sind. Hier bietet sich als Strategie Kooperation an – dadurch lassen sich die Spritpreise sogar ohne (verbotene) Preisabsprachen hoch halten.

§ 807 Die Tankstellen bzw. das Wettrüsten sind auf Grund der wiederholten Wechselwirkung – und damit der Möglichkeit, eine Strategie zu entwickeln und sich auf den anderen Spieler bzw. dessen Strategie einzustellen – für die praktische Anwendung wesentlich wichtiger als das in § 798 als isoliertes Ereignis auftretende Gefangenendilemma.

§ 808 Allerdings hat die Spieltheorie in ihrer Anwendung auf Entscheidungsfindungen auch Grenzen: als Schelling Spiel [3] wird ein Spiel bezeichnet, in dem es auf reine Koordination bzw. Koinzidenz ankommt. Es gibt zwei Spieler, die nicht mit einander kommunizieren dürfen. Ihnen wird folgende Aufgabe gestellt: ‘Vervollständigen Sie den folgenden Satz: Eine Münze wurde geworfen; sie zeigte...’. Wenn beide Spieler den Satz auf gleiche Weise vervollständigen, erhalten sie je einen Euro; wenn sie ihn auf unterschiedliche Weise vervollständigen, erhalten Sie kein Geld (verlieren aber auch nichts). Die Gewinnmatrix ist

	Blau Kopf	Blau Zahl
Rot Kopf	(1,1)	(0,0)
Rot Zahl	(0,0)	(1,1)

Wie im Gefangenendilemma ist Kommunikation zwischen den Spielern nicht möglich. Jedoch ist es im Schelling Spiel nicht einmal möglich, dem anderen Spieler Motive für eine bestimmte Wahl zu unterstellen: es gibt keinen Zug, der von vornherein zu schlechteren oder besseren Alternativen führen würde. Und es gibt daher auch keine Möglichkeit, dem anderen Spieler eine Strategie zu unterstellen. Daher sollten alle vier Felder der Gewinnmatrix mit gleicher Häufigkeit auftreten: die Wahrscheinlichkeit, dass beide Spieler gewinnen, beträgt also 50%. Experimente (für Details und Zitate siehe [3]) zeigen jedoch, dass mehr als 3/4 der Spiele gewonnen wurden. Ein genauerer Blick erklärt warum: 86% der Probanden ergänzen ‘Kopf’ und nicht ‘Zahl’, obwohl die Versuchsausgänge der Münze gleich wahrscheinlich sind. Der Grund ist ein einfacher: ‘Kopf oder Zahl’ ist die typische Frage beim Münzwurf – die umgekehrte Folge, ‘Zahl oder Kopf’ dagegen wird nicht gefragt, kommt uns eher sperrig und irritierend vor.<sup>3</sup> Explizit befragt würde niemand diese Reihenfolge als relevant für eine Spielstrategie empfinden – sie schleicht sich aber über das unbewusste in die real angewandte Strategie ein. Solche unbewussten Einflussgrößen lassen sich auch im Rahmen der Spieltheorie nicht – oder bestenfalls auf der Basis sehr unvollständiger Modelle und Annahmen – formalisieren.

### Gefangenendilemma mit mehreren Spielern: die Gemeindefiese

§ 809 Das Problem der Gemeindefiese ist eine Spielvariante, die die eigentlich bei wiederholter Wechselwirkung leicht zu realisierende Kooperation noch einmal genauer betrachtet. Kooperation bedeutet gewöhnlich Verzicht. Betrachten wir eine Versuchung, die zu einer

<sup>3</sup>Die Argumentation klingt plausibel – ich sollte Ihnen jedoch nicht verheimlichen, dass die experimentellen Untersuchungen im anglo-amerikanischen Sprachraum durchgeführt wurden. Allerdings ist auch dort die Reihenfolge ‘Head or Tail’ fest verdrahtet.

Katastrophe führt, wenn ihr jeder erliegt. Die Situation lässt sich durch das Problem der Gemeindefeld illustrieren:

Zehn Bauern eines Dorfes, die je eine Kuh haben, lassen alle zehn Kühe auf der Gemeindefeld grasen. Die Kühe werden schön fett, wobei die Weide mehr oder weniger kahl gefressen wird. Die Bauern werden reicher und manche können sich bald eine zweite Kuh leisten. Als der erste Bauer seine zweite Kuh auf die Gemeindefeld schickt, ist praktisch keine Veränderung zu beobachten. Vielleicht finden die Kühe etwas weniger Nahrung, vielleicht werden sie etwas weniger fett. Auch wenn der zweite und der dritte Bauer jeweils eine zweite Kuh auf die Weide schicken, gibt es noch keine großen Probleme. Obwohl die Kühe sichtbar schlanker geworden sind, ist jede von ihnen noch wohlgenährt und gesund. Als jedoch auch der siebte Bauer seine zweite Kuh kauft, leiden offensichtlich alle Kühe unter Hunger, und der Wert aller siebzehn Kühe zusammen erreicht nicht den der ursprünglichen zehn. Als schließlich alle zehn Bauern zwei Kühe haben, verhungern alle Kühe. Zunächst sind also zwei Kühe mehr wert als eine, deshalb ist es für jeden Bauern vorteilhaft, eine zweite Kuh zu kaufen – bis alle verhungern.

§ 810 Mit den Ergebnissen I für ‘ich habe eine sehr dünne Kuh’, II für ‘ich habe zwei sehr dünne Kühe’, III für ‘ich habe eine sehr fette Kuh’ und IV für ‘ich habe zwei sehr fette Kühe’ ergibt sich für die Gemeindefeld die folgende Gewinnmatrix:

	Mehrheit kauft zweite Kuh	
	ja	nein
Ich kaufe 2te Kuh	<b>2</b> , 2 (II)	<b>4</b> , 1 (IV)
Ich kaufe keine 2te Kuh	<b>1</b> , 4 (I)	<b>3</b> , 3 (III)

Die Gewinnmatrix zeigt wieder einen Zielkonflikt: für mich persönlich ist IV optimal, für die Gemeinschaft III. Der entscheidende Unterschied zum Zwei-Personen-Spiel wie dem Gefangenendilemma besteht darin, dass der Zielkonflikt zwar existiert, aber für einen einzelnen Spieler dieser Zielkonflikt immer dadurch rationalisiert werden kann, dass ja alles im Sinne der Gemeinschaft gut geht, wenn er selbst der einzige ist, der statt Kooperation Konkurrenz wählt.

§ 811 Ein interessante Aspekt an dieser Matrix (und damit an der Interpretation des Mehrpersonenspiels) ist die Tatsache, dass das Mehrpersonenspiel wieder auf zwei Akteure reduziert wird. Statt alle einzelnen Spieler zu betrachten, wird ein Akteur (eben der, der seine Strategie ermitteln/bewerten will) ausgewählt. Alle anderen Mitspieler werden als ein Akteur aufgefasst. Die Beschreibung ist eine völlig sinnvolle Reduktion: für den einzelnen Akteur ist nur relevant, wie sich die anderen Spieler als Gesamtheit verhalten – welcher der Spieler welchen Zug auswählt (bzw. wie viele Kühe platziert) ist irrelevant. Auf diese Weise lassen sich alle Situationen der Art ‘das Interesse des Einzelnen vs. das Interesse der Gruppe/Gesellschaft’ formalisieren.

#### 8.1.4 Iteriertes Gefangenendilemma

§ 812 Beim iterierten Gefangenendilemma entspricht der einzelne Spielzug dem gewöhnlichen Gefangenendilemma. Im Unterschied zu diesem ist nicht nach einem Zug Schluss, sondern es werden wiederholt Züge gemacht, wie z.B. beim Wettrüsten in § 805 und bei den Spritpreisen in § 806. Als Konsequenz lernt man die Strategie des anderen Spielers kennen und kann sich auf diese einstellen. Außerdem unterscheidet sich die Situation im Hinblick auf die Folgen einer Strategie: eine Strategie, die im Einzelspiel einen maximalen Gewinn liefert, kann im Wiederholungsfall zum Verlust führen!

§ 813 Wie beim normalen Gefangenendilemma gehen wir auch hier von der folgenden Annahme aus: entscheidend für die Gewinnmaximierung in einem Nicht-Nullsummenspiel ist die Kooperation. Beim iterierten Gefangenendilemma lassen sich auf Grund der häufigeren

Wechselwirkung verschiedene Strategien entwickeln, die die voran gegangenen Züge des Mitspielers berücksichtigen können. Einige typische Strategien sind:

- Altruist: kooperiere immer, egal was der Mitspieler tut. Diese Strategie ist für den Mitspieler schnell zu erkennen, erlaubt aber keine echte Wechselwirkung der Strategien.
- Egoist: konkurriere immer, egal was der Mitspieler tut. Auch diese Strategie ist für den Mitspieler schnell zu erkennen und auch sie erlaubt keine echte Wechselwirkung.
- alternierender Altruist und Egoist: Kooperation und Konkurrenz nach festem Schema (z.B. je zweimal oder 4 mal Kooperation, dann zweimal Konkurrenz, dann wieder 4 mal Kooperation usw.). Erlaubt ebenfalls keine echte Wechselwirkung.
- Tit for Tat (wie Du mir, so ich Dir): mache immer das, was der Mitspieler im Zug davor gemacht hat. Im Gegensatz zu den beiden voran gegangenen Strategien werden hier die früheren Aktionen des Mitspielers berücksichtigt. Führt dieser jedoch eine der beiden nicht-reaktiven Strategien, so ergibt sich im besten Fall eine Altruist–Altruist Kombination, im ungünstigsten dagegen eine Egoist–Egoist Kombination.
- Altruist mit verzeihender Strafe: konkurriert der Gegner, konkurriere zweimal, dann kooperiere wieder.
- Altruist mit stärkerer Strafe: Kooperation, konkurriert der Gegner, konkurriere zweimal, dann kooperiere wieder.
- Konkurrenz wird nicht verziehen: kooperiere bis der andere konkurriert. Dann konkurriere immer.

Viele weitere Strategien lassen sich auf ähnliche Weise erzeugen. Insbesondere können auch Strategien entwickelt werden, die viele zurück liegende Züge berücksichtigen. Letzteres ist z.B. erforderlich, wenn man versuchen möchte, die Strategie des Mitspielers zu verstehen und sich gegebenenfalls auf diese einzustellen.

§ **814** Als eine sehr erfolgreiche Strategie erweist sich Tit for Tat (TFT). Oder im Detail [127]: Kooperiere im ersten Schritt. In jedem folgenden Schritt tue das, was der Gegner in der vorigen Runde tat. Diese Strategie ist so einfach, dass auch Stichlinge TFT spielen wenn es darum geht aus vielen Fischen eine Schwarmformation zu erzeugen, die eine räuberische Makrele davon überzeugt, dass sie keinen Schwarm Stichlinge vor sich hat sondern einen nicht zum Verzehr geeigneten großen Fisch. Hier gehen die Stichlinge näher am Fressfeind ein erhöhtes Risiko ein, gefressen zu werden; wenn die anderen Stichlinge aber gut genug sind, den großen Fisch zu formen, wird gar nicht gefressen. Was immer noch besser ist, als zufällig vom in den diffusen Schwarm einfallenden Fressfeind erwischt zu werden.

§ **815** Außer Tit for Tat haben sich im Experiment noch verschiedene andere erfolgreiche Strategien heraus gestellt. Als Eigenschaften dieser erfolgreicher Strategien haben sich gezeigt:

- Freundlichkeit: greift nicht als erstes an.
- Nachsichtigkeit: kehrt zur Kooperation zurück, wenn der Gegner kooperativ war.
- Provozierbarkeit: reagiert auf Konkurrenz mit Konkurrenz.
- Reziprozität: Reaktion des Programms hängt weitgehend von der Strategie des Gegners ab.
- Verständlichkeit: Strategie ist deutlich genug, so dass sich ein Gegner darauf einstellen kann.

Am Ende des Kapitels werden wir verschiedene in den studentischen Projekten entwickelte und getestete Strategien betrachten.

## 8.2 Klimafolgenabminderung spieltheoretisch

§ 816 Als ein Beispiel für die Anwendung spieltheoretischer Ansätze in einem komplexen System mit vielen Spielern und eindeutigen Zielkonflikten soll hier der Versuch der Klimafolgenabminderung beschrieben werden. Der anthropogene Klimawandel<sup>4</sup> koppelt die verschiedenen globalen Akteure in dem Sinne, dass er nicht vor irgendwelchen Grenzen halt macht oder in irgendeiner Form ein früheres Wohlverhalten besonders belohnt oder früheres Fehlverhalten besonders sanktioniert. Überspitzt formuliert: die reichen Staaten auf der Nordhalbkugel tragen zwar am meisten zu den CO<sub>2</sub> Emissionen bei, aber die Malediven saufen ab.

§ 817 Dieser Abschnitt ist unabhängig von der Spieltheorie auch auf Grund der vereinfachten Darstellung des Klimamodells von Interesse: zwar wird mit dem normalen komplexen Modell gerechnet, um jedoch nicht für jedes Emissionsszenario einen neuen Run machen zu müssen, wird das Modell linearisiert. Dann lässt sich die Green'sche Funktion des Systems bestimmen, d.h. seine Impulsantwort. Diese entspricht der Übertragungsfunktion des Systems. Diese kann mit dem Emissionsszenario gefaltet werden, um die zugehörige Klimaänderung zu bestimmen.

### 8.2.1 Das Klimasystem spieltheoretisch

§ 818 Anthropogene Einflüsse auf das Klimasystem lassen sich spieltheoretisch mit folgenden Schlagworten beschreiben:

- das Problem der anthropogenen Klimaänderungen (und entsprechend deren Vermeidung) ist kein Nullsummenspiel: des einen Gewinn ist nicht des anderen Verlust sondern es wird die optimale Ausnutzung von Ressourcen angestrebt. Einerseits möchten alle Beteiligten geringe momentane Reduktionskosten und möglichst dicht am Buisness as usual arbeiten, andererseits möchten aber auch alle Beteiligten möglichst geringe Klimafolgekosten.
- das Problem möglicher Reduktionsszenarien ist ein iteriertes Problem: jeder Spieler kann seine Spielzüge neu anpassen und gegebenenfalls seine Spielstrategie wechseln.
- komplizierter als die bisher betrachteten Beispiele wird die Kombination aus Klima- und Wirtschaftssystemen dadurch, dass es sich um ein Mehrpersonenspiel handelt. Allerdings kann es für einige Fragestellungen auch auf ein einfaches Zwei-Personenspiel reduziert werden.

§ 819 Betrachten wir die Akteure in diesem Mehrpersonenspiel etwas genauer. Sie sind beide Bestandteil von zwei Systemen: die Akteure sind durch das globale Wirtschaftssystem und das Klimasystem gekoppelt. Beide Systeme werden von allen Akteuren gemeinsam erzeugt (bzw. die Akteure sind die Systeme) – die Gemeindewiese in § 809ff ist ein einfaches Beispiel. Sie werden aber individuell erfahren: des einen Untergang (Versinken der Malediven) kann des anderen Argrarfläche sein (endlich lassen sich die Weiten Sibiriens zum Getreideanbau nutzen). Auf Grund der unterschiedlichen wirtschaftlichen Abhängigkeiten und der verschiedenen Empfindlichkeiten gegenüber Klimaänderungen divergieren die Auffassungen der Akteure bezüglich der besten Vermeidungsstrategie. Der Nettoeffekt etwaiger Vermeidungsstrategien hängt ab von den Spielstrategien der einzelnen Akteure sowie den Allianzen zwischen ihnen.

§ 820 Spieltheoretisch lässt sich das Problem auf verschiedene Weisen anpacken. In einem single-actor case (Einpersonenspiel) ist die einzige Frage die der Optimierung des Emissionspfad: die Welt handelt wie ein Spieler und die einzige zu bestimmende Frage ist, wieviel CO<sub>2</sub> darf wann emittiert werden. Und außerdem: welche zusätzlichen Energiequellen stehen zur Verfügung oder müssen erschlossen werden bzw. welche Auswirkungen hat die Reduktion der CO<sub>2</sub>-Emission, was davon ist tolerabel und was muss aufgefangen werden.

<sup>4</sup>Im Rahmen dieses Skripts ist es nicht sinnvoll, differenziert über Klimawandel als eine Überlagerung aus natürlichem und anthropogenen Einflüssen nachzudenken; Hinweise auf Literatur dazu finden Sie z.B. in [64] und natürlich, wenn auch versteckter, in den IPCC-Berichten unter [www.ipcc.ch](http://www.ipcc.ch).

## GLOBAL ENVIRONMENT AND SOCIETY (GES) MODEL

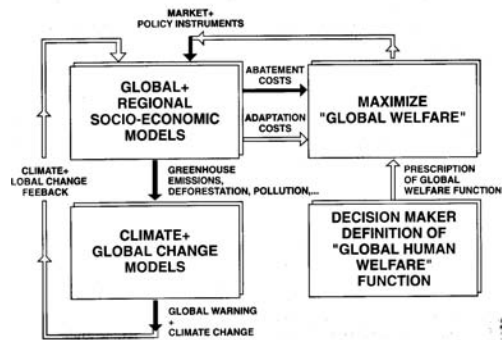


Abbildung 8.1: Definition des Global Environment and Society (GES) Model; dunkle Pfeile repräsentieren gut verstandene Wechselwirkungen, helle Pfeile weniger verstandene [39]

§ 821 In einem Szenario mit mehreren Mitspielern gibt es unterschiedliche Beschreibungen. Generell gilt im multi-actor case (Mehrpersonenspiel): die Beteiligten verfolgen unterschiedliche Strategien, wobei allerdings Gruppenbildung möglich ist. Gruppenbildung setzt vergleichbare Interessen und Möglichkeiten voraus; nahe liegende Gruppen sind die Industrienationen auf der einen, die Schwellenländer auf der anderen Seite und die Entwicklungsländer am Katzentisch.

§ 822 Egal ob zwei oder mehr Akteure bzw. Gruppen von Akteuren, die wesentlichen Strategien. In einer kooperativen Strategie verfolgen alle Beteiligten ein gemeinsames Ziel und sind bereit, es gemeinsam durchzusetzen. In diesem Fall reduziert sich das Spiel auf den single-actor case. Bei einer nicht-kooperativen Strategie dagegen überwiegen Egoismen: Ziele und Strategien der Akteure sind unterschiedlich.

### 8.2.2 Globales Modell Umwelt und Gesellschaft – Einpersonenspiel

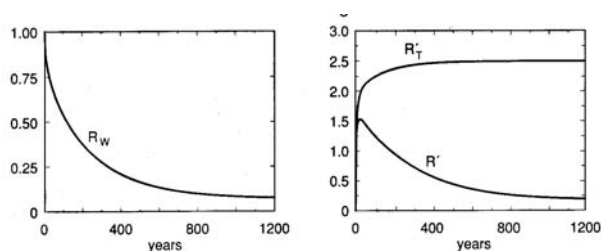
§ 823 Für die einfachste spieltheoretische Beschreibung möglicher Emissionsreduktionen erfolgt im Global Environment and Society (GES) ModelGlobal Environment and Society Model wie in Abb. 8.1 skizziert. Die einfachste, auch in der Abbildung angenommene Variante ist das Einpersonenspiel. Dieses Szenario ist nur unter einer Annahme sinnvoll: alle Beteiligten haben sich auf ein kooperatives Klimafolgen-Vermeidungsszenario geeignet. Das Ziel des Spielers (also aller Beteiligten) ist die Maximierung des globalen Wohlergehens GW – letzteres ist nicht nur gemeinsam definiert sondern auch akzeptiert. Dann lässt sich das Ziel als Optimierungsproblem auffassen: wie ist das Emissionsszenario zu gestalten, so dass das zeitliche Integral des GW maximiert wird?

§ 824 Zur formalen Behandlung dieses Optimierungsproblems benötigen wir eine Kombination aus Emissionsszenarios und Klimamodell.<sup>5</sup> Um die Klimafolgen eines gegebenen Emissionsszenarios zu bewerten, muss dieses in das Klimamodell eingesetzt werden.<sup>6</sup> In der Normalversion ist ein Klimamodell langsam. Damit eine größere Zahl von Emissionsszenarien betrachtet und miteinander verglichen werden kann, wird das verwendete Klimamodell linea-

<sup>5</sup>Für alle folgenden Betrachtungen macht es keinen Unterschied, dass die Arbeiten von Hasselmann auf dem 1996er Bericht des IPCC (Second Assessment Report [58]) aufbauen. Das Prinzip verändert sich auch mit den neueren IPCC-Berichten nicht (2001 und 2007, Third and Fourth Assessment Report, beide Berichte sind mit allen Bänden unter der Homepage des IPCC [www.ipcc.ch](http://www.ipcc.ch) zu finden. Für das eigentliche Modell ist nur der jeweils erste Band (Working Group I: The physical basis of climate change) interessant. Um die Motivation der einzelnen Spieler besser zu verstehen und die Auswirkungen von Klimaänderungen auf GW, lohnt auch ein Blick in Band II (Working Group II: Climate change impacts, adaption and vulnerability). Working Group III befasst sich im dritten Band mit den Möglichkeiten, Klimaänderungen zu reduzieren oder aufzufangen (Working Group III: Mitigation of Climate Change).

<sup>6</sup>Umgekehrt basieren natürlich auch alle Vorhersagen in den IPCC Berichten auf Emissionsszenarien – alle uns bekannten Vorhersagen hängen damit auch von Annahmen über wirtschaftliche Entwicklung und damit Entwicklung von CO<sub>2</sub> Emissionen ab.

Abbildung 8.2: Impulsantwort Klimasystem [39]: Green'sche Funktion  $R_\omega$  für die  $\text{CO}_2$ -Konzentration (*links*); Green'sche Funktion  $R_T$  der Temperatur sowie  $R$  als Gesamt-Response (*rechts*)



risiert: es werden nur kleine Änderungen betrachtet.<sup>7</sup> In diesem linearisierten Modell ist die Änderung des Klimavektor (Zustandsvektor)  $\vec{x}(t)$  auf Grund der Emission  $e(t)$  gegeben als

$$\vec{x}(t) = \int_{t_0}^t \vec{G}(t-t') e(t') dt$$

### Green'sche Funktion des Klimamodells

§ 825 Im linearisierten Modell lässt sich die Änderung des Klimavektors also durch die Faltung des Emissionsszenario mit der Impulsantwort des Klimasystems, beschrieben durch die Green'sche Funktion oder systemtheoretisch durch die Übertragungsfunktion, bestimmen. Damit lassen sich die Klimavektoren für verschiedene Emissionsszenarien bestimmen, ohne das Klimamodell jedesmal anwenden zu müssen.

§ 826 Zur Bestimmung der Green'schen Funktion muss außer dem globalen Zirkulationsmodell<sup>8</sup> (global circulation model, GCM) der  $\text{CO}_2$ -Kreislauf berücksichtigt werden. Daher ist die Green'sche Funktion zusammengesetzt aus den Green'schen Funktionen für beide Subsysteme:

$$G(t) = G_T(t) + \int_0^t G_T(t-t') \dot{\omega}(t') dt'$$

$$\text{mit } \omega(t) = \int_{t_0}^t G_\omega(t-t') e(t') dt$$

$$\text{und } T(t) = \int_{t_0}^t G_T(t-t') \dot{\omega}(t') dt' .$$

§ 827 Abbildung 8.2 zeigt im linken Teil die Green'sche Funktion  $R_\omega$  für die  $\text{CO}_2$ -Konzentration. Diese klingt, wie durch die Aufnahme von  $\text{CO}_2$  in Ozeane, in die Biosphäre aber auch in den Boden und den Einbau in Mineralien zu erwarten, mit der Zeit ab. Im rechten Teilbild ist  $R_T$  die Green'sche Funktion der Temperatur: das Klimasystem benötigt einige Zeit, bis sich nach einem  $\text{CO}_2$ -Eintrag ein neuer Gleichgewichtszustand und damit eine neue konstante Temperatur ausbildet. Beide Funktionen zusammen ergeben im rechten Teilbild  $R$  als Gesamt-Response. Ein  $\delta$ -Eintrag von  $\text{CO}_2$  hat einen maximalen Einfluss auf

<sup>7</sup>Dieser Ansatz ist keine so grobe Vereinfachung, wie sie auf den ersten Blick erscheint. Die alten Klimaszenarien gingen stets von einer Verdopplung der  $\text{CO}_2$  Konzentration aus. Dann wäre eine Linearisierung leichtfertig. Allerdings wurde die Verdopplung nur als ein Bezugswert verwendet und nicht transient gerechnet sondern für das sich unter den neuen Bedingungen einstellende Gleichgewicht. Da wir jedoch Emissionsszenarien betrachten wollen, benötigen wir einerseits ohnehin ein transientes Modell, andererseits betrachten wir aber auch keine großen sprunghaften Änderungen. Die Linearisierung scheint so lange ein plausibler Ansatz zu sein, wie die atmosphärische  $\text{CO}_2$  Konzentration das Doppelte ihres vorindustriellen Wertes nicht überschreitet und die Temperaturzunahme weniger als 3 K beträgt. Die Grenze, bis zu der eine Linearisierung noch sinnvoll erscheint, ist in den hier vorgestellten Ergebnissen jeweils mit angegeben.

<sup>8</sup>Ein modernes GCM wie ECHAM5 <http://www.mpimet.mpg.de/en/wissenschaft/modelle/echam.html> oder HAMMONIA <http://www.mpimet.mpg.de/en/wissenschaft/ueberblick/atmosphaere-im-erdsystem/modellierung-der-mittleren-und-hohen-atmosphaere/hammonia.html?0=> enthält neben der Dynamik und Chemie von Atmosphäre und Ozean (allerdings in zwei separaten, miteinander verwobenen Modellen) auch globale Kreisläufe von Spurengasen sowie Beschreibungen von Kryo- und Hydrosphäre, Boden und eine Topologie.



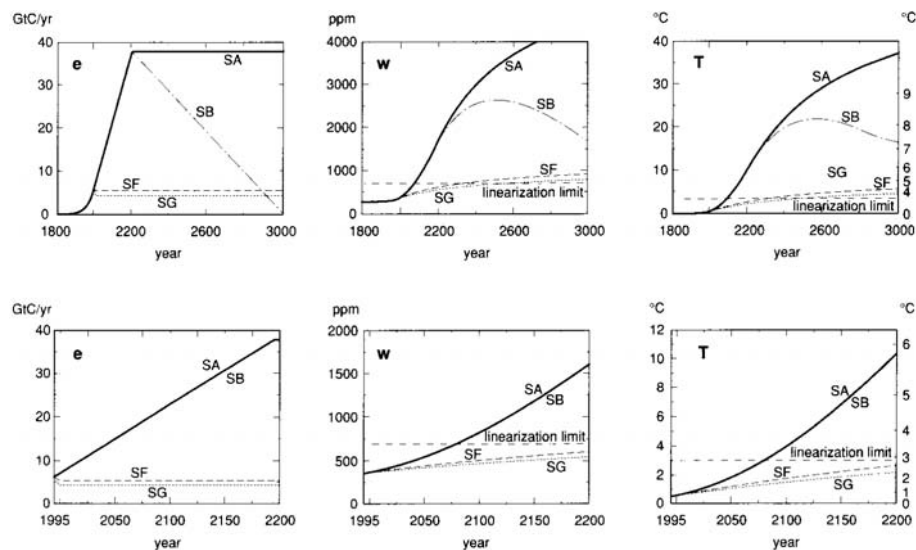


Abbildung 8.3: Emissionsszenarios (*links*), sich daraus ergebende  $\text{CO}_2$ -Konzentration in der Atmosphäre (Mitte) sowie daraus resultierende Temperaturänderung (*rechts*) für vier verschiedenen Szenarien im Zeitraum 1800–3000 (*oben*) und 1995–2200 (*unten*) [39]. Die horizontale Linie gibt den Bereich an, bis zu dem die Linearisierung des Klimamodells noch sinnvolle Ergebnisse liefert

die Temperatur nach ein bis zwei Dekaden und klingt danach ab: anfangs wird Zeit benötigt, bis sich die Atmosphäre auf die neue  $\text{CO}_2$ -Konzentration eingeepegelt hat, später nimmt das  $\text{CO}_2$  bereits wieder ab.

**Zwischenrechnung 52** Falls Sie selbst mit dem entsprechenden Modell spielen wollen; die zugehörigen Parameter sind in [39] gegeben, so dass sie die Response-Funktionen und damit die Temperaturänderungen für unterschiedliche Emissionspfade bestimmen lassen. Und da Sie ja den Umgang mit Übertragungsfunktionen in SIMULINK gelernt haben, sollte es auch kein Problem sein, die entsprechenden Modelle zu erstellen.

### Vorgegebene Szenarien – Response-Funktionen

§ 828 Im Rahmen des Skripts und der Vorlesung wollen wir keine eigenen Emissionsszenarien entwickeln sondern greifen auf die von [39] verwendeten zurück – letztere sind ähnlich den IPCC Szenarien. Das Standard-Szenario in der Vorhersage von Klimamodellen schreibt den in den letzten Dekaden beobachteten Trend des  $\text{CO}_2$ -Anstieges linear fort bis zum Jahr 2200. Anschließend ist die  $\text{CO}_2$ -Emission konstant. Dieses Szenario wird als Business As Usual (BAU) bezeichnet.

§ 829 Neben dem Business As Usual (BAU) Szenario, jeweils linkes Teilbild in Abb. 8.3 bezeichnet als SA, werden noch drei weitere Szenarios betrachtet: SB folgt BAU bis 2200 und fällt anschließend linear auf Null ab bis zum Jahre 3000. Im Szenario SF ist die Emission auf dem Niveau von 1990 eingefroren, in Szenario SG auf 80% von 1990.

§ 830 In allen vier Szenarien steigt die atmosphärische  $\text{CO}_2$ -Konzentration (mittleres Teilbild in Abb. 8.3) noch lange nach 2200 weiter an – entsprechend steigt natürlich auch die Temperatur (rechtes Teilbild in Abb. 8.3) weiterhin an. Selbst in den Kyoto-ähnlichen Szenarien (Emission eingefroren auf dem stand von 1990 bis 80% von 1990) steigt die Temperatur weiterhin an; sie bleibt allerdings bis nach dem Jahr 2200 unter dem Limit für die Linearisierung. Für die linear ansteigenden Szenarien BAU (und nach 2200 konstant oder linear abfallend) dagegen wird dieses Limit schon ca. 2075 überschritten – was damit den Vorhersagehorizont des Modells gibt.

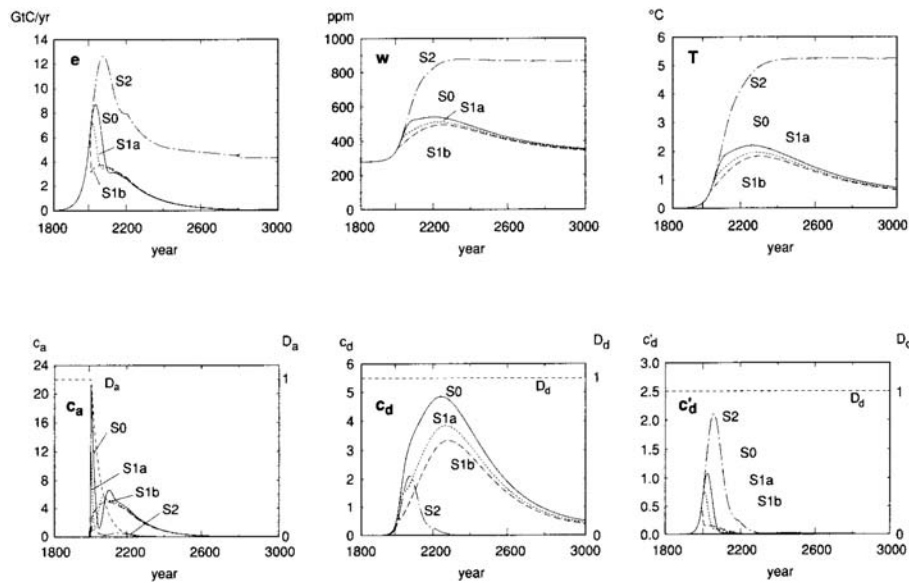


Abbildung 8.4: Sensitivität Kosten [39]

§ 831 Unabhängig von den im folgenden Unterabschnitt betrachteten Kostenmodellen kann auf der Basis dieses response bereits eine Einschränkung möglicher Szenarien vorgenommen werden, wenn für den Spieler bestimmte Ergebnisse (z.B. ein Temperaturanstieg von mehr als 5 K, ein Meeresspiegelanstieg von mehr als 30 cm) aus irgendwelchen nicht kostenbezogenen Erwägungen völlig inakzeptabel sind.

### Klimakostenmodell

§ 832 Für die verbliebenen Szenarien ist das Ziel die Maximierung von GW bzw. die Minimierung der mit dem Klimawandel bzw. der Verhinderung des Klimawandels verbundenen Kosten. Entsprechend setzen sich die Gesamtkosten  $C$  zusammen aus den Folgekosten  $C_d$  (für damage cost) und den Vermeidungskosten  $C_a$  (für abatement cost):

$$C = C_d + C_a .$$

Die Folgekosten lassen sich beschreiben als

$$C_d = \int_{t_0}^{\infty} c_d(T(t), \dot{T}(t), t) dt .$$

Die Folgekosten hängen zum einen explizit von der Zeit ab (z.B. allgemeine Kostenentwicklung), zum anderen von der Temperatur  $T(t)$  sowie ihrer Änderung  $\dot{T}(t)$ . Die erstere Abhängigkeit entsteht durch die direkten Folgen, z.B. die Finanzierung von Nahrungsmittelimporten auf Grund ausgefallener Ernten. Die Abhängigkeit von der Geschwindigkeit der Temperaturabhängigkeit ist vielfältig zu begründen: schnelle Temperaturänderungen erfordern schnelles Handeln (z.B. Erhöhung von Deichen) und damit gegebenenfalls höhere Kreditaufnahmen mit entsprechenden Kosten für Zinsen. Umgekehrt kann aber auch eine langsame Temperaturänderung zu höheren Folgekosten führen, da man erst eine einfache Abwehrmaßnahme durchführt, die dann bald durch eine besser angepasste ersetzt werden muss. Langsamere Temperaturänderungen können aber auch die Folgekosten verringern, wenn die Landwirtschaft es schafft, sich den sich verändernden Randbedingungen anzupassen ohne dass es zu großen Missernten bzw. Ernteaussfällen und den damit verbundenen Kosten kommt. Die kurze Erläuterung sollte ausreichen zu erklären, dass die Annahmen über die Folgekosten bereits eine Wissenschaft (oder ein Glaskugellesen) für sich sind.

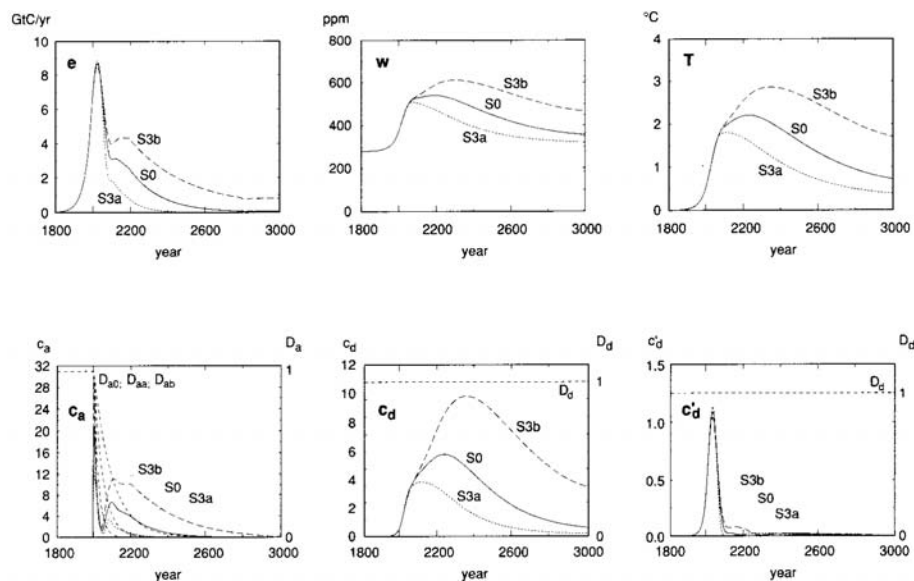


Abbildung 8.5: Vermeidungskosten für verschiedene Zeitkonstanten des Discount Terms [39]

§ **833** Folgekosten sind außerdem in einem anderen Parameter leicht zu variieren: die obere Integrationsgrenze ist hier auf  $\infty$  – alternativ ließe sich auch ein begrenzter Zeithorizont  $t_h$  verwenden. Der endliche Zeithorizont hat den Vorteil, dass das Integral konvergiert. Für  $t \rightarrow \infty$  lässt sich dies nur durch geeignete Discount Faktoren erreichen – daher ist in  $c_d$  und  $c_a$  jeweils die Zeit explizit enthalten. Sowohl die Kosten als auch die Discount-Faktoren sind inflations-angepasst zu berücksichtigen (oder, falls man den Blick in die Kristallkugel/den Teebeutel/die eingetrocknete Kaffeetasse scheut durch Annahme einer ungewöhnlich stationären Welt ohne Inflation – letzteres dann allerdings auch in allen Preisen).

§ **834** Die Vermeidungskosten sind auf ähnliche Weise beschrieben als

$$C_a = \int_{t_0}^{\infty} c_a(e(t), \dot{e}(t), \ddot{e}(t), t) dt .$$

Hier tritt eine zusätzliche Abhängigkeit von  $\ddot{e}(t)$  auf, d.h. der Beschleunigung der Emissionsrate. Das Auftreten dieser zusätzlichen Abhängigkeit lässt sich vielleicht an einem einfachen Beispiel deutlich machen: die Reduktion der Emissionsrate durch Sparmaßnahmen ist ein im wesentlichen linearer Effekt – statt 60 W Glühbirne eine 8 W Sparlampe. Das ist etwas teurer, die Kosten steigen entsprechend mit zunehmender Emissionsrate. Weniger individuelle Reduktionsmaßnahmen können z.B. der Bau eines AKW (oder eines Windparks) sein. Dies ist ein langfristiges Projekt mit entsprechend gestreckter Finanzierung – wenn es aber beschleunigt werden muss, da sich die Emissionsrate sonst immer stärker erhöhen würde (also beschleunigt würde), entstehen zusätzliche Kosten.

§ **835** Die Koeffizienten in den Folge- bzw. Vermeidungskosten hängen von wirtschaftlichen Parametern ab. Damit hat aber auch das gesamte Szenario eine Sensitivität gegenüber wirtschaftlichen Parametern. Hier verlassen wir den Bereich, in dem wir uns als Physiker wohlfühlen – ich möchte aber weniger, dass Sie die einzelnen wirtschaftlichen Szenarien bewerten und diskutieren als vielmehr dass wir diese hinnehmen um zu verstehen, wie man mit Hilfe eines Klimamodells und etwas Spieltheorie Möglichkeiten in der Reduktion von treibhauswirksamen Gasen (und damit in einer Begrenzung der Folgen eines sich ändernden Klimas) entwickeln kann.

§ **836** Beginnen wir mit einem sehr optimistischen Fall in Abb. 8.4. In der oberen Zeile sind jeweils  $\text{CO}_2$  Emission,  $\text{CO}_2$ -Konzentration sowie die sich daraus ergebende Temperatur

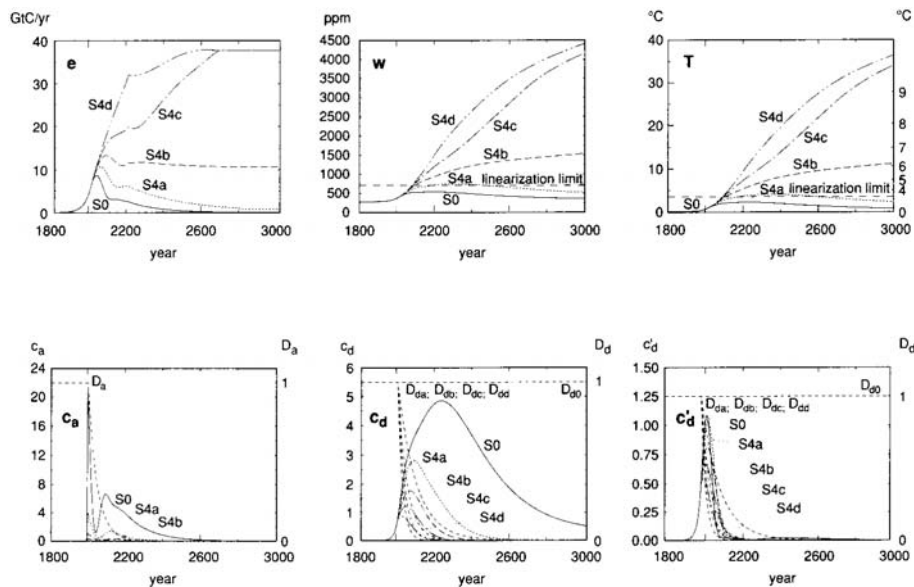


Abbildung 8.6: Abhängigkeit der Schadenskostenfunktion vom Discount [39]

gegeben, in der unteren Zeile die (spezifischen) Vermeidungskosten  $c_a$ , die spezifischen Folgekosten  $c_d$  sowie den Beitrag  $c'_d$  der Folgekosten, der sich aus der Rate der Berücksichtigung der Rate der Temperatur ergibt. Das Referenzszenario S0 ist für eine reduzierte konstante Emission bestimmt, entsprechend SG in Abb. 8.3. Die Kurven S1 sind für das gleiche Szenario mit einem reduzierten (a) bzw. einem verschwindenden (b) Trägheitsterm bestimmt; in S2 wurde für gleiches Emissionsszenario die Rate der Temperaturänderung nicht weiter im Schadensterm berücksichtigt.

§ 837 Der Vergleich von S1a und S1b zeigt, dass der langfristige Einfluss des Trägheitsterms gering ist. Außerdem stammt der wesentliche Anteil des Folgekostenterm aus der Temperaturänderung selbst und weniger aus der Rate der Temperaturänderung, wie ein Vergleich von  $c_d$  und  $c'_d$  nahe legt.

§ 838 Im Zusammenhang mit der Integrationsgrenze in den Folge- und Vermeidungskosten haben wir bereits auf die explizite Zeitabhängigkeit hingewiesen, die die Einführung eines Discounts erlaubt. Ein Discount ist für beide Kostenformen bei den Maßnahmen sinnvoll, die sich Ingenieur-technisch realisieren lassen, wie z.B. der Bau höherer Deiche. Vermeidungskosten kommen in der Regel aus diesem Bereich. Bei den Folgekosten ist die Einführung des Discount dagegen umstritten: für die technischen Maßnahmen lässt er sich einführen aber die Folgekosten für den Verlust an Biodiversität oder Landstrichen (untergegangene Malediven; verschwundene Alpengletscher<sup>9</sup>.)

§ 839 Um die Empfindlichkeit der Vermeidungskosten von der Zeitkonstante des Discount abschätzen zu können, wird angenommen, dass alle Kosten mit einem Discount behaftet werden können und dass dieser im Referenzszenario eine Zeitkonstante  $\tau_a$  von 50 Jahren hat (im Referenzszenario S0 wie aus Abb. 8.4); die Vergleichsszenarien gehen von einer

<sup>9</sup>Letztere bilden übrigens für die Schweizer Energiepolitik eine interessante Herausforderung: die Schweiz kann trotz hohen Lebensstandards zumindest bei der Energieerzeugung eine sehr gute Bilanz aufweisen: Wasserkraft und Kernkraft sind wesentliche Energieträger. Beide benötigen jedoch den Gletscher: nur durch ihn sind die Stauseen gefüllt und funktioniert das Kühlsystem der KKW's. Eine CO<sub>2</sub>-Zunahme und damit verbunden eine Temperaturerhöhung kann dadurch selbst bei geringerem Energieverbrauch zu einer Erhöhung der CO<sub>2</sub> Emission führen, da von den CO<sub>2</sub>-neutralen (naja?) Energieträgern auf fossile Brennstoffe ausgewichen werden müsste. Die Abhängigkeit regenerativer Energieträger von Klimaänderungen ist bisher meines Wissens nirgendwo richtig diskutiert.

Zeitkonstante des Discount von 25 Jahren (S3a) bzw. 50 Jahren (S3b) aus (siehe Abb. 8.5). Aus dem Vergleich der Kurven wird der starke Einfluss dieser Zeitkonstanten deutlich – und dass, wo gerade sie, wie bereits angedeutet, die am stärksten in den Modellen debattierte Größe ist (siehe auch die detailliertere Diskussion in [39] sowie die dort gegebenen Verweise).

§ 840 Auch die Schadenskosten lassen sich alle mit Hilfe eines Discounts beschreiben. Das Baseline Szenario S0 (durchgezogene Linien in Abb. 8.6) berücksichtigt keinen Discount bei den Klimafolgenkosten: alle anderen Szenarien berücksichtigen endliche Zeitkonstanten für den Discount von 100 Jahren (4a), 50 Jahren (4b), 35 Jahren (4c) und 25 Jahren (4d). Insbesondere in den letzten beiden Fällen steigen die CO<sub>2</sub> Konzentrationen und damit auch die Temperaturen sehr deutlich an – auf Werte, die die Lebensbedingungen in fast allen Regionen deutlich verändern werden.

§ 841 Aus dieser kurzen Beschreibung können wir einen wichtigen Schluss bezüglich der Sensitivität der Klimaänderungen ziehen: die sich unter Berücksichtigung eines wirtschaftlichen Szenarios ergebenden Klimaänderungen hängen sehr stark von den Zeitkonstanten der Vermeidungs- und Folgekosten – und insbesondere deren Verhältnis – ab! Und damit hängen auch alle anderen Aussagen des Modells davon ab.

## Ergebnisse

§ 842 Die Ergebnisse dieses spieltheoretisch noch nicht einmal besonders interessanten Modells sind eher etwas deprimierend – oder sie erlauben einen realistischen Blick auf die auf uns zukommenden Probleme.

- die Zeitskalen des CO<sub>2</sub>-Response sind wesentlich größer als der sozio-ökonomische Vorhersagehorizont. Daher sind alle Vorhersageversuche auf Annahmen im Bezug auf das Wirtschafts- und Gesellschaftssystem angewiesen. Zur Illustration: das Standard-Szenario geht von einem Business as usual über 200 Jahre aus. Allein im vergangenen Jahrhundert haben zwei Weltkriege, die Unabhängigkeit der Kolonien von den Mutterländern und das Bestreben nach Gleichberechtigung (Rasse, Religion, Geschlecht) das sozio-ökonomische System einigermaßen verändert.
- begrenzte Klimafolgen erfordern CO<sub>2</sub>-Emissionsreduktionen um mindestens 50% über wenige Jahrhunderte,
- die Reduktionsrate hängt stark von den Annahmen über die Kosten ab,
- klimatisch optimale Szenarien ergeben sich nur, wenn die Folgekosten wesentlich größer werden als die Vermeidungskosten,
- sozio-ökonomische Trägheit bewirkt, dass auch optimale Emissionszenarien noch 1–2 Dekaden wie BAU ansteigen dürfen,
- Energiesparmaßnahmen sind nicht effizient genug, Energieerzeugung muss CO<sub>2</sub>-frei erfolgen,
- Umstellungszeitskalen sind Jahrzehnte bis Jahrhunderte – in der Zeit steigt die Temperatur ohnehin noch,

Und noch eine Nachbemerkung: so pessimistisch die Auflistung klingt: sie ist noch ein optimistisches Szenario, da die anderen treibhauswirksamen Spurengase im Modell noch nicht einmal berücksichtigt sind.<sup>10</sup>

<sup>10</sup>Zur Zeit trägt CO<sub>2</sub> nur noch ungefähr die Hälfte zum anthropogenen Treibhauseffekt bei. Das zweitwichtigste treibhauswirksame Spurengas ist Methan. Dieses entsteht primär in Feuchtgebieten und in den Verdauungstrakten von Wiederkäuern. Damit ist es direkt mit der Nahrungsmittelproduktion (Reis und insbesondere immer mehr Rinder) verbunden. Zusätzlich wird es aber bei einer Erhöhung der Temperatur auch durch die aus den tauenden Permafrostböden gebildeten Feuchtgebiete in die Atmosphäre eingetragen – das klingt sehr nach einer Rückkopplung; und zwar eher einer Mit- als einer Gegenkopplung. Weitere Treibhausgase sind Lachgas (u.a. Düngung) und troposphärisches Ozon (Sommersmog).

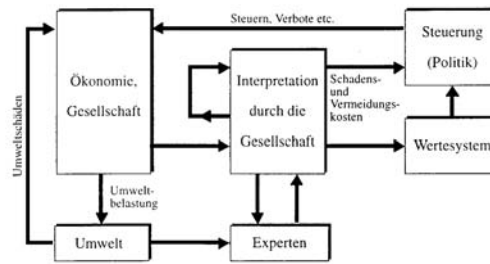


Abbildung 8.7: Erweitertes GES-Modell [153]

### Erweitertes Modell

§ 843 Stehr und von Storch (1995) haben ein erweitertes GES vorgestellt, vgl. Abb. 8.7. Im wesentlichen ist dabei die Zahl der Akteure bei der Definition des Ziels (also dieses schwammigen Begriffes ‘globales Wohlbefinden’) erweitert, auch wird den Experten eine größere Rolle eingeräumt. Diese bezieht sich insbesondere auf die Unterstützung bei der Interpretation sozio-ökonomischer und klimatischer Zusammenhänge.<sup>11</sup> Ein weiterer Punkt in diesem erweiterten Modell ist die Betonung von Wertesystemen – dieser Aspekt würde z.B. wie bereits in § 838 angedeutet, auch die Berücksichtigung nicht monetär zu betrachtender Klimafolgen erlauben. Eine kritische Diskussion dieses von Klima und Klimaänderung als soziales Konstrukt findet sich in [152].

### 8.2.3 Globales Modell Umwelt und Gesellschaft – Mehrpersonenspiel

§ 844 Im Rahmen des Mehrpersonenspiels brauchen wir keine Varianten betrachten, die auf Kooperation basieren: diese sind bereits durch das Einpersonenspiel beschrieben sind.<sup>12</sup> Für die verbleibenden nicht-kooperierenden Varianten ergeben sich zwei unterschiedliche Möglichkeiten:

- es gibt  $n$  gleichberechtigte (nicht-kooperierende) Akteure. Dabei ist gleichberechtigt im Sinne vergleichbaren Handlungsspielraums zu verstehen: auch wenn die Niederlande und die Malediven oder Bangladesch in vielleicht vergleichbarem Maße von einem mit einer Erwärmung einher gehenden Anstieg des Meeresspiegels betroffen sind, haben die Niederlande ganz andere technologische und finanzielle Möglichkeiten, die Folgen abzumindern und sich den neuen Randbedingungen anzupassen. Die gleichberechtigten Akteure erlauben eine gleichmäßige Verteilung von nicht nur Emissionsrechten sondern auch Vermeidungs- und Folgekosten; sie werden auch in ungefähr gleichem Maße (z.B. Pro-Kopf Verbrauch) an der Emission beteiligt sein.
- es gibt  $n$  Akteure mit (gruppenweise) unterschiedlichen Möglichkeiten. Aus dem ersten Beispiel wären wahrscheinlich Bangladesch und die Malediven in einer Gruppe, die Niederlande eher in einer anderen Gruppe anzusiedeln.

Die Einteilung der Akteure in diese beiden Arten von Gruppen erlaubt eine vollständige Beschreibung des wesentlich komplexeren Systems – allerdings um den Preis eines mehrstufigen Verfahrens. So wäre eine nahe liegende Einteilung ein Mehrpersonenspiel mit in der ersten Stufe drei Gruppen: (a) Industrienationen, (b) aufstrebende Staaten (wie Indien und China) und (c) klassische Entwicklungsländer. Diese Gruppen sind (naja vielleicht nur idealerweise)

<sup>11</sup>Wenn man sich das Hick-Hack der Experten (und Politiker) bei der Ausarbeitung der IPCC Berichte ansieht und bedenkt, dass diese unabhängigen Experten alle auch ihre Forschungsmittel haben wollen, sollte man die Vorteile einer stärkeren Einbindung von Experten nicht überbewerten.

<sup>12</sup>Das Einpersonenspiel legt den unter den Annahmen bezüglich Wirtschaft und wirtschaftlicher Entwicklung optimalen Emissionspfad fest. Da alle Beteiligten kooperieren, haben sie diesen Emissionspfad gemeinsam festgelegt und es geht nur noch um die Aufteilung der einzelnen Emissionsanteile an die Teilnehmer.

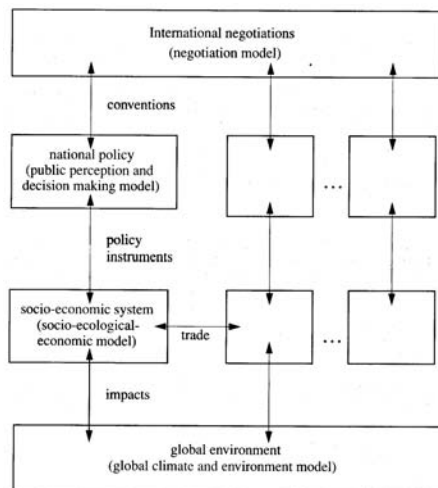


Abbildung 8.8: GES aus Abb. 8.1 im Mehrspielerszenario [38]

in sich relativ homogen und können unter dem Szenario gleich berechtigter – aber nicht notwendigerweise kooperierender – Akteure diskutiert werden. Der Prototyp für ein Spiel mit  $n$  gleichberechtigten Akteuren ist die Gemeindefläche.

§ 845 Für das Mehrspielermodell lassen sich prinzipiell zwei Ansätze wählen: (a) wir gehen davon aus, dass der optimale Emissionspfad  $e(t)$  bereits durch das Einspielermodell festgelegt ist; im Rahmen des Modells wird kein neuer Emissionspfad gesucht sondern für ein gegebenes  $e(t)$  eine geschickte Aufteilung der Emissionsanteile  $b_k(t)$  auf die Akteure derart, dass  $\sum b_k(t)e(t) = e(t)$ . (b) Der Emissionspfad ist nicht vorgegeben sondern entwickelt sich aus den jeweils durch die Wahl des eigenen Emissionspfades bestimmten Vermeidungskosten sowie die durch die kombinierten Emissionspfade aller Teilnehmer bedingten Folgekosten.<sup>13</sup>

§ 846 Im Mehrspielermodell sind alle Akteure durch das Klima gekoppelt. Sind die Akteure kooperativ, reduziert sich das Problem auf den Single-Actor. Sind sie nicht kooperativ, so lassen sich verschiedene Fälle durchspielen. So kann z.B. ein einzelner Vermeider betrachtet werden, wobei einzeln nicht zwingend im Sinne von einzelner Nation verstanden werden muss sondern auch eine zu einem einzelnen Akteur zusammen gefasste Gruppe von Staaten mit gemeinsamen Zielen und Möglichkeiten sein kann. Der allgemeinste Fall ist jedoch eine Ansammlung von  $n$  Individualisten.

§ 847 Als wesentliches Ergebnis werden wir sehen, dass Lösungen für nicht-kooperative Strategien oder teilweise kooperative Strategien (Kartell-ähnlich) der kooperativen Strategie unterlegen sind. Die optimale Lösung ergibt sich im Einpersonenspiel.

### Gleichberechtigte nicht-kooperative Akteure

§ 848 Das Grundproblem beim Spiel mehrerer Akteure oder Gruppen von Akteuren ist in beiden Varianten, gleichberechtigt oder nicht, identisch: jeder Akteur ist bestrebt sein Wohlbefinden GW zu optimieren. In diesem Spiel sind die Akteure allerdings keine Einzelpersonen sondern Staaten oder Staatengruppen. Daher ist bereits die Definition von Wohlbefinden bzw. maximalem Wohlbefinden innerhalb eines Akteurs nicht trivial. So werden die zur Herbeiführung des (maximalen) Wohlbefindens benötigten politische Entscheidungen z.B. beein-

<sup>13</sup>Unsere Akteure verfügen über unbegrenzte finanzielle Ressourcen – die Komplikation, dass z.B. die Niederlande ihre Vermeidungskosten nicht mehr finanzieren können, da sie plötzlich alles Geld in den Deichbau stecken müssen, ist nicht vorgesehen.

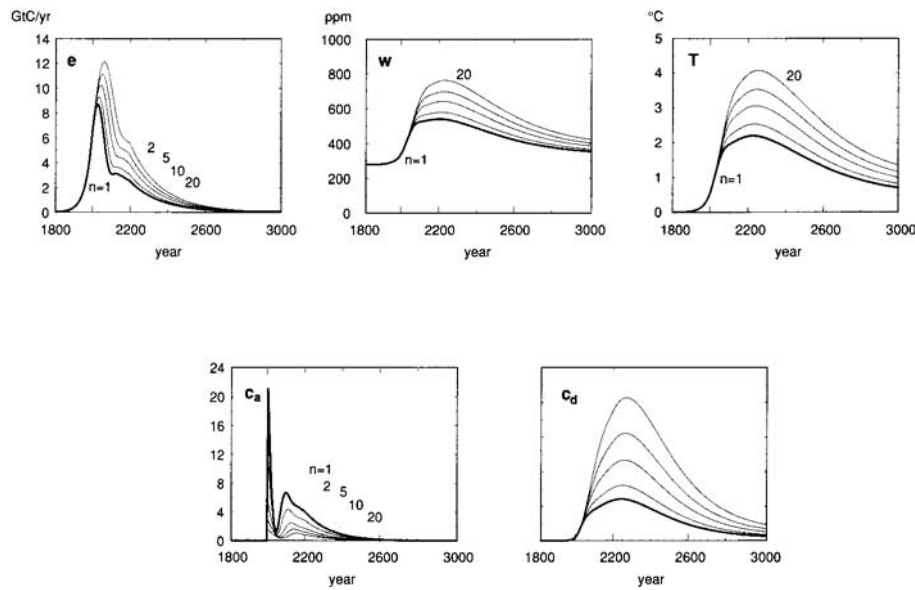


Abbildung 8.9: Nash Gleichgewicht mit wenigen Akteuren [38]; gegeben sind jeweils die Emission  $e$ , die resultierende  $\text{CO}_2$ -Konzentration  $w$ , die Temperaturänderung  $T$  sowie die Vermeidungskosten  $c_a$  und die Folgekosten  $c_d$

flusst durch lokale Interessen(vertretungen)<sup>14</sup> beeinflusst, durch globale Verflechtungen<sup>15</sup> und damit letztendlich gekoppelte Akteure sowie durch weitere politische Randbedingungen, wie die Förderung von alternativen Energien oder die Akzeptanz der Kernenergie – und natürlich auch den Export von Technologie und/oder Know-How.

§ 849 Die dafür erforderliche Erweiterung vom Einpersonen- auf das Mehrpersonenspiel verändert das in Abb. 8.1 gegebene Modell zu dem in Abb. 8.8 gezeigten. Für jeden einzelnen Akteur ergibt sich im wesentlichen wieder das aus dem Einpersonenspiel bekannte GES – der wesentliche Unterschied ist die Kopplung der Akteure über (a) das gemeinsam erfahren Klimasystem bzw. dessen Änderungen (globale Kopplung), (b) Handel (jeweils bilateral oder multi-lateral, nicht jedoch zwingend global) sowie (c) internationale Vereinbarungen (hoffentlich global oder zumindest nahezu global)

§ 850 Da wir das Mehrpersonenspiel als nicht-kooperativ definiert haben, wollen die Akteure den eigenen Profit ohne Rücksicht auf andere Akteure maximieren. In diesem Fall ist die Lösung ein Gleichgewicht (Nash-Gleichgewicht) in Form eines Satzes von Strategien unter denen jeder Spieler seinen Gewinn unter der Randbedingung der Strategien der anderen Spieler maximieren kann. Allerdings ist der Gewinn für die Gemeinschaft geringer als beim kooperativen Szenario. **Aber:** Nash-Gleichgewicht ist im Gegensatz zum kooperativen Szenario stabil.

§ 851 Die  $n$  nicht-kooperierenden Akteure haben also ein gemeinsames Ziel: das Nash-Gleichgewicht zu erreichen. Dieses soll erreicht werden unter Maximierung der eigenen Emis-

<sup>14</sup>Mögliches Szenario: Zwar lässt sich auf Grund der globalen Erwärmung im Alten Land die Apfelsorte Cox nicht mehr anbauen, dafür aber der Braeburn. Da sich dieser einfacher lagern und behandeln lässt, ist der Obstbauer nicht einmal unglücklich und daher entsprechend weniger motiviert, hohe Vermeidungskosten zu tragen. Der Kollege Seilbahnbetreiber im Schwarzwald oder in den bayrischen Alpen dagegen sieht seine Verdienstmöglichkeiten bei einer Klimaerwärmung im wahrsten Sinne des Wortes dahin schmelzen und hat eine höhere Motivation, sich an Vermeidungskosten zu beteiligen.

<sup>15</sup>Dieser Aspekt ist sicherlich heute im Zuge der Globalisierung wesentlich wichtiger als noch vor zehn Jahren: was nützt einer Nation ihre boomende Export-Industrie, wenn auf Grund der  $\text{CO}_2$  Emissionen und des damit verbundenen Temperaturanstiegs die potentiellen Importeure gerade untergehen.



sionsfunktion  $e_i$  bei gleichzeitiger Minimierung der Kostenfunktion  $C_i$ . Da wir gleichberechtigte Akteure betrachten, ergibt sich eine symmetrische Situation:  $e = \sum_{i=1}^n e_i = ne_i$  mit  $e$  als dem optimalen Emissionsszenario aus dem Einspielermodell.

§ 852 Im Gegensatz zum Einspielermodell hat das Mehrspielermodell den Vorteil, dass Abweichungen eines Akteurs weniger Gewicht haben als dies beim Einspielermodell der Fall ist: der Gewichtungsfaktor für jeden einzelnen Akteur ist  $1/n$ , d.h. selbst wenn einer der Akteure seine 'erlaubte' CO<sub>2</sub> Emission um den Faktor zwei überschreitet, ist die Gesamtabweichung vom optimalen Emissionsszenario gering. Mit zunehmender Zahl der Akteure beinhaltet diese 'Unauffälligkeit' aber auch ein Risiko: alle drücken sich um die Vermeidungskosten (Analogie: Gemeindewiese). Letztere Version wird als Freerider Lösung bezeichnet.

§ 853 Beginnen wir mit einem Nash-Gleichgewicht in einem Modell mit wenigen Akteuren. Abbildung 8.9 zeigt Emission  $e$ , resultierende CO<sub>2</sub>-Konzentration  $w$ , Temperaturänderung  $T$  sowie Vermeidungs-  $c_a$  und Folgekosten  $c_d$  für verschiedene 'Spieler'-Anzahlen von 1 bis 20. Der Fall  $n = 1$  entspricht einem Akteur (bzw. einer Menge kooperierender Akteure) und gibt damit das optimale Emissionsszenario. Bei einer größeren Zahl von Akteuren folgen alle dem gleichen Verhaltensmuster.

§ 854 Aus Abb. 8.9 lässt sich erkennen, dass die CO<sub>2</sub>-Emission mit der Zahl der Akteure gegenüber dem Einspielermodell ansteigt: jeder Akteur trägt weniger Vermeidungskosten bei und emittiert stattdessen mehr. Da dies zu einem Anstieg der CO<sub>2</sub> Emission führt, ist die Temperaturänderung entsprechend größer und die Folgekosten steigen deutlich. Der Anstieg in der CO<sub>2</sub> Emission ist noch moderat (kleiner als  $\sqrt{n}$ ); auch stellt sich ein vergleichbarer Emissionsverlauf ein.

§ 855 Vergleicht man die Folge- und die Vermeidungskosten in Abhängigkeit von  $n$ , so wird deutlich, dass die geringe absolute Abnahme der Vermeidungskosten durch deutlich höhere Folgekosten mehr als aufgeessen wird – allerdings kommen die Folgekosten halt eben erst in der Zukunft.

### Ein einzelner Vermeider

§ 856 Betrachten wir jetzt einen einzelnen Vermeider in den  $n$  gleichberechtigten aber nicht-kooperativen Akteuren. Dieser Akteur bewertet die Folgen des Klimawandels hoch und ist daher zu hohen Vermeidungskosten bereit. Die anderen  $(n - 1)$ -Akteure dagegen leben weiterhin in ihrem Nash-Gleichgewicht. Das Szenario lässt sich auch für  $m$  kooperative Klimafolgenvermeider verwenden, da letztere als ein Akteur betrachtet werden können. Lediglich das Verhältnis der Möglichkeiten von Vermeider zu Nicht-Vermeider ist wichtig.

§ 857 Die in Abb. 8.10 sind nicht wirklich überraschend: ein einzelner Vermeider kann die Klimafolgenkosten nicht reduzieren. Daher macht Reduktion für ihn keinen Sinn. Allerdings hat der Vermeider bei  $n \leq 10$  noch genug Einfluss auf BAUs, um insgesamt eine reduzierte Emissionen zu bewirken. Das Ergebnis sollte nicht unterschätzt werden: wenn einige wenige der Groß-Emittenten auf Vermeidung setzen würden, wären wir zwar immer noch weit von einem optimalen Emissionspfad entfernt – es würde sich aber dennoch bereits ein merkbarer positiver Effekt ergeben.

### Handelnde Akteure

§ 858 Die Situation verändert sich jedoch, wenn wir handelnde Akteure betrachten. Dann erfolgt die Kopplung der Akteure nicht nur durch das Klima sondern auch durch Handel – im Zuge der Globalisierung sicherlich keine schlechte Annahme. Auf Grund der Kopplung durch den Handel betrachtet jeder Akteur neben seiner Profitmaximierung auch die möglichen (Re-)Aktionen der anderen Akteure: alle Wirtschafts- und Klimavariablen eines Akteurs werden von allen anderen Akteuren beeinflusst.

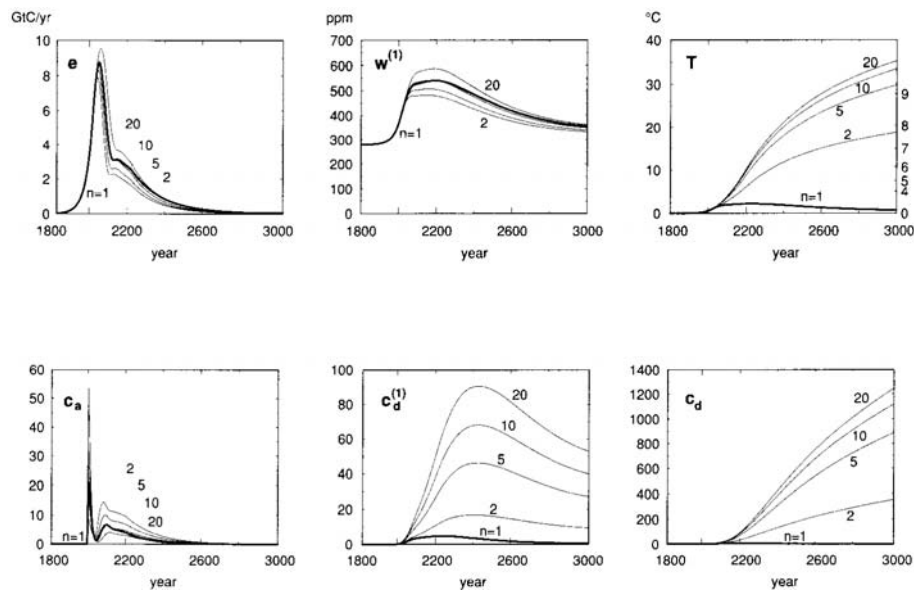


Abbildung 8.10: Nash Gleichgewicht und ein einzelner Vermeider in Gegenwart von  $n - 1$  gleichberechtigten, nicht-kooperativen Akteuren [38]

§ 859 Und die Einflussmöglichkeiten sind vielfältig, z.B. (a) Ächtung/Abbruch der Geschäftsbeziehungen auf Grund unterschiedlicher Auffassungen bezüglich Klimawandel und Reduktion der CO<sub>2</sub> Emission, (b) die bisher produzierten und exportierten Waren werden von den Importeuren auf Grund des veränderten Klimas nicht mehr benötigt (Sibirien baut sein Getreide selbst an und möchte statt der früher so gerne gekauften Heizlüfter lieber Kühlschränke), (c) der Lebensstandard bei einem wichtigen Importeur sinkt auf Grund der hohen Klimafolgekosten so stark, dass zwar noch Interesse an den Produkten besteht, aber kein Geld mehr da ist, (d) benötigte Rohstoffe/Agrarprodukte können nicht mehr importiert werden, da sie auf Grund des Klimawandels im Erzeugerland nicht mehr verfügbar sind, (e) Angebot und Nachfrage verschieben sich auf dem Weltmarkt (nachdem ganz Sibirien in ein Kornfeld verwandelt wurde, ist der Preis für Roggen soweit gesunken, dass der mittlere Westen der USA kollektiv Konkurs angemeldet hat).

§ 860 Diese kurzen (und teilweise etwas flapsigen) Beispiele zeigen, dass eine Berücksichtigung der Beziehungen unter den Akteuren nicht einfach ist. Eine besonders interessante Variante der handelnden Akteure sind jedoch Konsument und Produzent fossiler Brennstoffe. Deren Kopplung ist besonders direkt, da des einen Gewinn direkt des anderen Verlust ist. Außerdem tut sich an dieser Stelle eine wichtige Regulierungsmöglichkeit auf: Produzenten können über Preise in die Emissionszenarien eingreifen.<sup>16</sup>

§ 861 Abbildung 8.11 zeigt verschiedenen Szenarien für einen Preis–Verbrauchs–Parameter  $\beta$ , der Produzenten und Konsumenten koppelt:  $\beta = 0$  entspricht hohen Preisen, so dass Vermeidungsknoten leichter getätigt werden, während  $\beta = 100$  niedrigen Preisen entspricht (die Produzenten geben einen Discount, da auf Grund der zunehmenden Zahl von Vermeidern nicht mehr so viele fossile Brennstoffe abgenommen werden).  $\beta = 0$  ist so eingestellt, dass das bekannte Einspieler-Modell den optimalen Emissionspfad liefert; in allen anderen Varianten

<sup>16</sup>Die Erfahrung machen wir in Europa z.Z. mit den (allerdings auch auf Grund von China's Energie- und Rohstoffbedarf) stark gestiegenen Energiekosten. Plötzlich lässt sich, zusammen mit einem ungewöhnlich warmen Winter, einem neuen IPCC-Bericht (der eigentlich nur altbekanntes mit etwas veränderten Zahlen enthält) und etwas politischer Werbung Energiesparen wieder leichter verkaufen. Die Lehre hätten wir auch schon aus dem Kalten Krieg und dem Wettrüsten ziehen können: offenbar ist der Geldbeutel ein wesentlich besseres Regulierungsmittel als der Verstand.

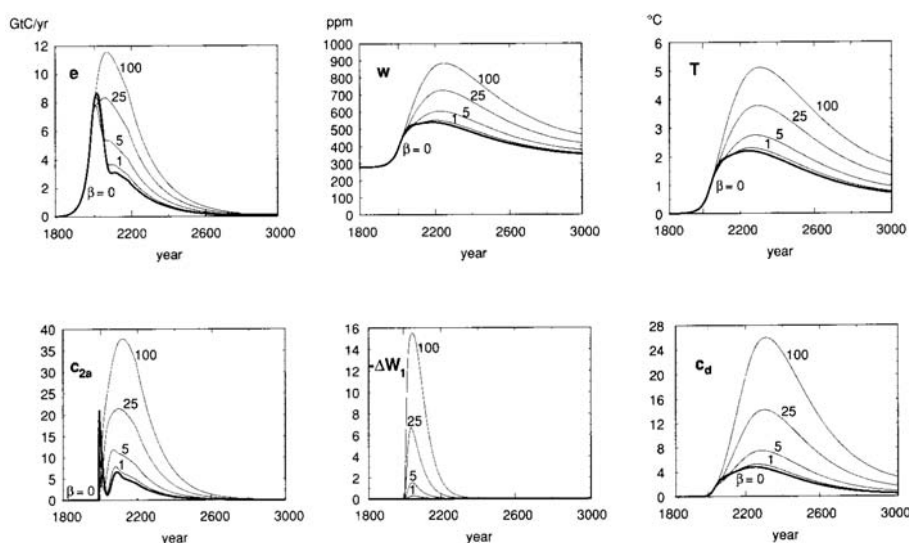


Abbildung 8.11: Handelnde Akteure: Konsument und Produzent fossiler Brennstoffe;  $\beta$  ist ein Preis-Verbrauchs-Parameter [38]

führen die niedrigen Energiekosten zu höheren Emissionen, also auch höheren Temperaturen und damit letztendlich höheren Vermeidungskosten. Die Preise fossiler Brennstoffe sind also ein starker Regelmechanismus.<sup>17</sup>

### Zusammenfassung Multi-Actor

§ 862 Die hier gezeigten Ergebnisse aus einem Mehrspieler-Szenario lassen sich wie folgt zusammen fassen:

- die Veränderung in den Folgen von Vermeidungsmaßnahmen in nicht-kooperativen Strategien ist nicht so stark wie aus dem Freerider-Szenario erwartet.
- bei kleinen Zahlen unabhängiger Akteure ändert sich das Szenario gegenüber dem optimalen nur geringfügig.
- einzelne Vermeider haben einen auf Grund nichtlinearer Effekte einen großen Einfluss.
- Preisnachlässe der Erzeugerländer fossiler Brennstoffe können Vermeidungsmaßnahmen außer Gefecht setzen.

Genauere Diskussionen finden sich in [37, 38] sowie auf anschauliche Weise auch in [153].

## 8.3 Strategien auswählen mit SIMULINK

§ 863 Im Wintersemester 2007/08 wurde der Leistungsnachweis für diese Vorlesung von den Studierenden jeweils in einem Projekt zum Gefangenendilemma erbracht. Beteiligt haben sich daran Florian Bertram, Andreas Buschermöhle, David Hamjediers, Oliver Hoefert, Benjamin Kloster, Alexander Niemer und Jens Poppenburg. Im Rahmen dieser Projekte wurden gängige Strategien vorgestellt, die Umsetzung in SIMULINK vorgenommen und die Strategien mussten sich (zumindest teilweise) gegeneinander bewähren.

<sup>17</sup>Die Aussage ist nicht ganz so trivial, wie sie einem gefrusteten Autofahrer vielleicht erscheinen mag: die Erhöhung von Spritpreisen (insbesondere die systematische wie bei Mehrwertsteuererhöhung) führt jedes mal zum Aufschrei und zum Versprechen, jetzt weniger fahren zu wollen. Aber meistens beruhigt sich das nach einiger Zeit. Ob die globale Wirtschaft ähnliche psychologische Züge zeigt?

### 8.3.1 Strategien

§ 864 Die gefundenen Strategien umfassen die bereits aus § 813 bekannten und einige sehr interessante zusätzliche. Auch zwei selbst entwickelte Strategien treten auf. Insgesamt traten an:

- Strategien ohne Berücksichtigung des Partners
  - **kind** (**always cooperate**): Kooperiere immer (kann leicht ausgenutzt werden)
  - **random**: zufälliger Zug
  - **nasty** (**always defect**): Konkurriere immer (bringt nur etwas, wenn der Gegner kooperiert); entsteht aus **spite falls** Gegner früh verrät
  - **per-kind**: periodisch koop-koop-verrat
  - **per-nasty**: periodisch verrat-verrat-koop
- Variationen über Tit for Tat
  - **tit for tat**: wie Du mir, so ich Dir, Oder: Auge um Auge, Zahn um Zahn; Problem Vendetta [73]
  - **tit for two tat**: bei Konkurrenz würfeln
  - **suspicious tit for tat**: beginnt mit Verrat
  - **prober**: beginnt kooperieren-verraten-verraten und spielt bei verzeihender Gegenstrategie verraten, sonst TIT FOR TAT
  - **forgiving tit for tat**: Verrat kann mit vorgegebener Wahrscheinlichkeit in Kooperation umgewandelt werden
  - **verratendes tit for tat**: Kooperation kann mit gewisser Wahrscheinlichkeit in Verrat umgewandelt werden
  - **forgetful tit for tat**: mit geringer Wahrscheinlichkeit wird letzter Zug des Gegners vergessen/mißdeutet
  - **omega tit for tat**: kooperiert im ersten Zug, dann **tit for tat** mit Langfrist Analyse (gegebenenfalls Verrat, Unterbrechung Vendetta) [126, 149]
  - **aggressive tit for tat**: analysiert die letzten 7 Züge
  - **negation**: **inverses tit for tat** (braucht einen Altruisten als Mitspieler)
- Weitere Strategien
  - **pavlov**: koop, dann koop falls beide in Vorrunde gleich sonst Verrat
  - **spite** (**grim**, **groll**): kooperiert bis erster Verrat, dann immer Konkurrenz
  - **modified grim**: überprüft 2 Züge (nur wenn beide Verrat wird selbst verraten)
  - **punisher**: kooperiere bis Gegner verrät; dann verraten, bis die durch den Verrat erzielten Gewinne des Gegenspielers aufgebraucht sind
  - **vengeance**: koop, bei Verrat zweimal Verrat (einer sofort, der andere bei nächster Kooperation des Gegners)
  - **gradual**: koop, bei Verrat „konk-koop-koop“, bei nächstem Verrat „konk-konk-koop-koop“
  - **go by majority** (Entscheidung gemäß Mehrheit): Langzeitgedächtnis; wenn Verrat Gegner überwiegt, Verrat – sonst koop (Vorschlag Buschermöhle)
  - **tit for tat memory last five** (letzte Fünf): kooperativ, Verratswahrscheinlichkeit aus 5 Zügen (Vorschlag Buschermöhle)
  - **master-slave**; **bauernopfer**: Turnier-Variante aus zwei Strategien; mit vielen Bauern auch als Southampton-Strategie bezeichnet: die Bauern reißen die anderen am Turnier beteiligten Strategien in die Verlustzone – dann kann der Master relativ schlecht spielen und gewinnt trotzdem.

### 8.3.2 Beispiele für die SIMULINK-Lösung

§ 865 Die wichtigsten Attribute in den meisten SIMULINK-Realisierungen waren die Verzögerung (irgendwo muss Tit for Tat ja sein Gedächtnis her kriegen), die Lookup-Table für die Gewinnmatrix und eine Sammlung von Blöcken, die die einzelnen Strategien beherbergen. Als Beispiel ist in Abb. 8.12 ein Grundmodell (links oben) zusammen mit verschiedenen Blöcken

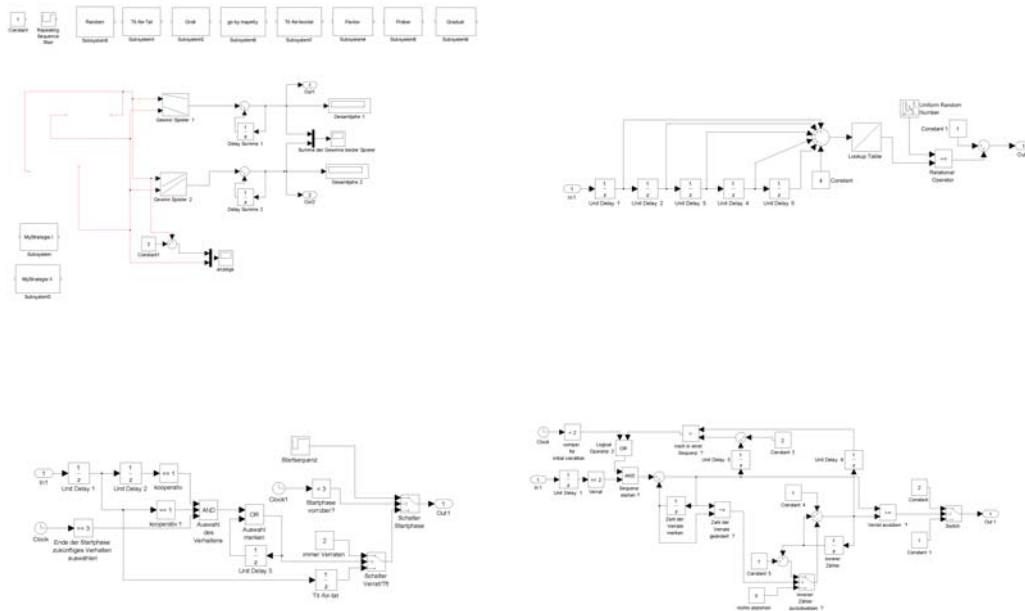


Abbildung 8.12: Gefangenendilemma: Master (*oben links*) und Beispiele für die durch Untersysteme realisierten Strategien: *tit for tat memory last five* (*oben rechts*), *prober* (*unten rechts*) und *gradual* (*unten rechts*) (Modell von Andreas Buschermöhle im Rahmen des Projekts erstellt)

für diverse Strategien gegeben. Aus dem Gesamtgewinn lässt sich der Gesamterfolg der Strategie ablesen, aus den Einzelgewinnen bzw. deren Verhältnis lässt sich erkennen, ob eine aggressive Strategie eine kooperative ausnutzt oder ob beide Strategien vergleichbaren Nutzen aus dem gemeinsamen Spiel ziehen.

### 8.3.3 Und welche Strategie ist die Beste?

§ 866 Es wurden nicht alle Strategien gegeneinander aufgezett sondern jeweils im Rahmen einer einzelnen Hausarbeit eine Teilmenge der Strategien. Insgesamt zeigt sich beim Vergleich:

- die Strategien waren alle recht zahm; es fehlte der richtige Bösewicht;
- Gutes Abschneiden lässt sich wie folgt erreichen:
  - Beginne kooperativ (nicht als erster verraten) → freundliche Strategie
  - Erkenne eine Vendetta (Strategie mit Gedächtnis):
    - \* indirekt z.B. in fTFT und TFfT
    - \* Direkt in Omega TFT oder mGrim
    - \* Erkenne Zufälligkeit (Omega TFT)

§ 867 Beim Vergleich der verschiedenen hier aufgeführten Strategien schneidet OTFT am besten ab: sie liefert den höchsten eigenen Gewinn und lässt sich gleichzeitig vom Gegenspieler nicht ausnutzen. Andere tit-for-tat Varianten liefern ebenfalls gute Ergebnisse. Insgesamt bestätigt sich das bereits im einführenden Abschnitt gesagte: grundsätzlich ist es besser, zu kooperieren – außer eine feindselige Strategie versucht, kooperationswillige Strategien auszunutzen. Die meisten besonders nachsichtigen Strategien liefern zwar im Spiel mit kooperativen Strategien bessere Ergebnisse; dafür lassen sie sich von einer konkurrierenden Strategie

aber auch um so besser ausnehmen. Tit-for-tat ist zumindest auch in aggressiven settings noch robust genug.

## Literaturhinweise

§ 868 Die Standardwerke über Spieltheorie unter sozialwissenschaftlichen Gesichtspunkten, d.h. der Frage nach Kooperation oder Konkurrenz, sind Rapoport [127] und Axelrod [2]; stärker auf die wirtschaftswissenschaftlichen Hintergründe geht Rasmusen [128] ein. Eine gute Mischung der verschiedenen Aspekte (und einfach gut zu lesen) ist Binmore [7]. Neuere Entwicklung sind, teilweise auch kritisch, diskutiert in Bacharach [3] sowie Hargreaves Heap and Varoufakis [35]. Und falls Sie bei Spiel eher an Glücks- oder Strategiespiele denken – Packel [112] bietet eine kurze und interessante Einführung.

## Fragen

**Frage 33** Erläutern Sie den Begriff des Nullsummenspiels.

**Frage 34** Erläutern Sie das Gefangenendilemma.

## Kurzprojekt in SIMULINK

§ 869 Das Modell zur Bestimmung des optimalen Emissionspfades haben Sie möglicherweise bereits in Zwischenrechnung 52 erstellt. Versuchen Sie jetzt, einen ganz einfachen spieltheoretischen Ansatz darum zu bauen, um einmal ein derartig komplexes Regelungssystem simuliert zu haben.

**Teil V**  
**Anhang**

## Einige wichtige Begriffe und Konzepte

§ 870 In diesem Abschnitt sind, sehr zusammenhanglos und auch nicht vollständig, einige im Zusammenhang mit nicht-linearen Systemen auftretende Begriffe und Konzepte erläutert. Wikipedia hätte vieles durchaus besser erklären und vor allen Dingen auch weiter führende Hinweise geben können – also scheuen Sie sich nicht, dort nach vertiefenden Informationen (und Links) zu suchen.

### 9.1 Diffusions–Reaktions-Modell

§ 871 Die Anwendung von Diffusions–Reaktions-Modellen ist in Abschn. 7.1 diskutiert; in diesem Abschnitt werden einige physikalische Grundbegriffe kurz wiederholt.

§ 872 In seiner allgemeinsten Form lässt sich ein Diffusions–Reaktions-Modell formal darstellen als

$$\frac{\partial \vec{c}}{\partial t} = \mathbb{D} \nabla^2 \vec{c} + \vec{F}(\vec{c}) . \quad (9.1)$$

Darin ist  $\vec{c}$  der Vektor der (gesuchten) Konzentrationen  $c_i$ ; die beiden Terme auf der rechten Seite beschreiben die diffusive Ausbreitung der Substanzen mit dem Diffusionstensor  $\mathbb{D}$  und die Reaktionen der verschiedenen Substanzen miteinander.

§ 873 Gleichung (9.1) ist eine spezielle Form der Kontinuitätsgleichung für die einzelnen Substanzen  $i$ : die Änderung der Konzentration  $c_i$  der  $i$ ten Substanz innerhalb eines Volumenelements (linke Seite) ist gegeben durch den Fluss dieser Substanz über die Umrandung des Volumenelements sowie die Quellen/Senken dieser Substanz innerhalb des Volumenelements.

§ 874 Gleichung 9.1 ist die vektorielle Form der einfachen skalaren Kontinuitätsgleichung einer Eigenschaft  $\varepsilon$  in einem Volumenelement:

$$\frac{\partial \varepsilon}{\partial t} + \nabla \cdot \vec{C}(\varepsilon) + S(\varepsilon)$$

mit  $\vec{C}(\varepsilon)$  als dem Fluss von  $\varepsilon$  über die Umrandung des Volumenelements und  $S(\varepsilon)$  als den Quellen und Senken von  $\varepsilon$  innerhalb der Volumenelements.

#### 9.1.1 Diffusion

§ 875 Im Rahmen des Diffusions–Reaktions-Modells erfolgt der Fluss durch die Umrandung nur durch Diffusion. Betrachten wir nur den diffusiven Teil, so ist der Fluss  $\vec{C}$  über die



Umrandung die einzige Quelle der Eigenschaft  $\varepsilon$  in dem Volumenelement und die Kontinuitätsgleichung liefert

$$\frac{\partial \varepsilon}{\partial t} + \nabla \cdot \vec{C}(\varepsilon) = \frac{\partial \varepsilon}{\partial t} + \nabla(-\mathbf{D}\nabla\varepsilon) = 0.$$

mit  $\vec{C}(\varepsilon) = -\mathbf{D}\nabla\varepsilon$  als dem Diffusionsstrom. Darin ist  $\mathbf{D}$  der Diffusionstensor.

§ 876 In einem homogenen Medium hängt der Diffusionstensor nicht von der räumlichen Koordinate ab, d.h. er kann vor die Divergenz gezogen werden und die Diffusionsgleichung wird

$$\frac{\partial \varepsilon}{\partial t} = \nabla(\mathbf{D}\nabla\varepsilon) = \mathbf{D} \Delta \varepsilon.$$

Ist die Diffusion ferner isotrop, so reduziert sich der Diffusionstensor auf einen skalaren Diffusionskoeffizienten  $D$  und die Diffusionsgleichung wird

$$\frac{\partial \varepsilon}{\partial t} = \nabla(D\nabla\varepsilon) = D \Delta \varepsilon.$$

§ 877 Der Diffusionstensor in (9.1) ist kein Hinweis auf anisotrope Diffusion sondern entsteht daraus, dass die einzelnen Substanzen unterschiedliche Diffusionskoeffizienten haben:

$$\begin{aligned} \dot{c}_1 &= D_1 \nabla^2 c_1 \\ \dot{c}_2 &= D_2 \nabla^2 c_2 \\ \vdots &= \vdots \\ \dot{c}_n &= D_n \nabla^2 c_n \end{aligned} \Rightarrow \begin{pmatrix} \dot{c}_1 \\ \dot{c}_2 \\ \vdots \\ \dot{c}_n \end{pmatrix} = \begin{pmatrix} D_1 & 0 & \dots & 0 \\ 0 & D_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 \dots & D_n & \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \dots \\ c_n \end{pmatrix} \Rightarrow \dot{\vec{c}} = \mathbf{D}\vec{c}.$$

§ 878 Reine Diffusion ist ein sehr selten auftretender Prozess: in flüssigen und gasförmigen Medien .... Dispersion oder Wirbeldiffusion

### 9.1.2 Reaktion

§ 879 Für die Reaktionen gibt es verschiedenen Ansätze in Abhängigkeit davon, welche Prozesse durch das Diffusions-Reaktionsmodell beschrieben werden sollen. Für den einfachen Turing'schen Ansatz mit Kannibalen der Konzentration  $C$  und Missionaren der Konzentration  $M$  ist das Gleichungssystem (ohne den Diffusionsteil)

$$\frac{\partial C}{\partial t} = b - MC^2 \quad \text{und} \quad \frac{\partial M}{\partial t} = a - M - M^2C.$$

Hierbei handelt es sich um zwei gekoppelte gewöhnliche Differentialgleichungen. Da die Gleichungen nicht linear sind, entfällt eine abkürzende Matrixschreibweise.

## 9.2 Emergenz

§ 880 Der Begriff der Emergenz wird in sehr vielen unterschiedlichen Zusammenhängen verwendet und hat entsprechend viele Bedeutungen. Für die schwächste Form können wir die Skulpturen 'The Prisoners' von Michelangelo Buonarotti (vgl. Abb. 9.1) verwenden: diese zeigen menschliche Figuren, die damit kämpfen, sich aus ihrem steinernen Gefängnis zu befreien.

§ 881 Michelangelo Buonarotti hat diese Skulpturen zur Veranschaulichung des künstlerischen Schaffungsprozesses erzeugt. In dieser einfachsten Interpretation beinhaltet Emergenz hier kein Geheimnis: der Bildhauer entfernt den nicht benötigten Stein und legt damit die Figuren frei. Die im Stein Gefangenen 'erscheinen' – diese Emergenz erfolgt in so elementarem Sinne, dass sie nach van Gulick [169] als specific value emergence bezeichnet wird mit der Definition

Abbildung 9.1: The Prisoner [http://www.bc.edu/bc\\_org/avp/cas/fnart/art/ren\\_italy/sculpture/10\\_97\\_5\\_25.jpg](http://www.bc.edu/bc_org/avp/cas/fnart/art/ren_italy/sculpture/10_97_5_25.jpg)



SPECIFIC VALUE EMERGENCE: the whole and its parts have features of the *same kind*, but have different *specific s btypes* or *values* of that kind. For example, a bronze statue has a given mass as does each of the molecular parts of which it is composed, but the mass of the whole is different from that of any of its material parts.

Über diesen sehr eingeschränkten Sinn hinaus definiert van Gulick [169] auch den Begriff der modest emergence

MODEST EMERGENCE: the whole has features that are *different in kind* from those of its parts (or alternately that *could* be had by its parts). For example, a piece of cloth might be purple in hue even though none of the molecules that make up its surface could be said to be purple. Or a mouse might be alive even if none of its parts (or at least none of its subcellular parts) were alive.

Modest emergence zeigt sich in einem Spektrum von Schattierungen in Abhängigkeit von dem Grad des Unterschieds zwischen dem Phänomen und der Basis, aus der es entsteht. Die kohärenten Strukturen in den nichtlinearen Systemen liefern viele Beispiele.

§ 882 Die von van Gulick eingeführten Begriffe haben nichts mit den Konzepten von starker und schwacher Form der Emergenz zu tun. Die schwache Form der Emergenz geht von der Annahme der nur vorübergehenden Nichterklärbarkeit der Eigenschaften emergenter Systeme auf Grundlage der Beschreibung ihrer Elemente aus; die starke Form der Emergenz dagegen geht von einer prinzipiellen Nichtbeschreibbarkeit aus.

§ 883 Modest Emergence tritt in vielen physikalischen Systemen auf. Die Solitonen der Korteweg–de Vries Gleichung sind unabhängige dynamische Entitäten, deren Geschwindigkeit und Form sich aus den Anfangsbedingungen ergibt. Weniger modest aber dafür robustere Emergenz zeigt sich in den Nervenpulsen, die sich in verschiedenen nichtlinearen Modellen (z.B. Hodgkins–Huxley Neuron) ergeben. Obwohl auch diese sich auf einem einheitlichen System mit konstanter Geschwindigkeit und unter Wahrung der Form ausbreiten, unterscheiden sie sich fundamental von Solitonen: ein Nervenimpuls erhält die Energie nicht sondern lebt in einem Zusammenspiel von Freisetzung und Dissipation von Energie.

§ 884 Noch robuster (und weniger modest) ist die Emergenz des Lebens sowie noch einen Schritt weiter die des Bewusstseins. Die Aspekte von Emergenz an diesem Ende der Skala fasst van Gulick [169] als radical emergence zusammen:

RADICAL EMERGENCE: the whole has features that are both different in kind from those had by its parts, and of a kind whose nature and existence is not necessitated by the feature of its parts, their mode of combination and the law-like regularities governing the features of its parts.

§ 885 Weitere Literatur: als Einführung eignet sich [48] oder [62], wesentlich anspruchsvoller (und formaler) ist [138]. Ebenfalls sehr formal, aber mit dem Schwerpunkt auf Neurowissenschaften ist [137]. Andere Literatur, ebenfalls mit der Frage nach dem Bewusstsein, sind [12, 70, 136, 169].

## 9.3 Morphogenese, biologische

§ 886 Eines der fundamentalen Probleme in der Entwicklungsbiologie ist das Verständnis der Entwicklung von Muster und Strukturen: wie entwickelt sich aus einem anfangs nahezu homogenen Haufen sich teilender Zellen des Embryos der weite Bereich von Mustern und Strukturen im lebenden Tier?<sup>1</sup> Wie gelingt die Ausbildung so unterschiedlicher funktionaler Elemente wie Haare, Federn und Drüsen auf der Haut? Wie bilden sich die Muster auf Tierhäuten oder sie schillernden Farben und Muster auf Schmetterlingsflügeln?

§ 887 Auch wenn die Gene eine wichtige Rolle dabei spielen, die Genetik sagt nichts über die die Muster und Strukturen produzierenden Mechanismen, die Morphogenese, aus. Die Bewegung und Umsortierung von Gewebe ist der Schlüsselprozess in nahezu allen Mechanismen der Morphogenese; sie resultiert aus einem komplexen Spiel mechanischer, chemischer und elektrischer Wechselwirkungen. Trotz der großen Fortschritte in molekularer Biologie und Genetik sind diese Prozesse und insbesondere das Wechselspiel zwischen ihnen kaum verstanden. Stattdessen besteht hier die Gefahr in die Methoden des 19. Jahrhunderts zurück zu verfallen und die Biologie wieder als eine klassifizierende Wissenschaft zu betrachten und sich in das Anfertigen endloser Listen zu flüchten. Bereits 1917 wies Thompson [160] auf dieses Problem hin. Und versuchte ihm dadurch zu begegnen, dass er die ersten Theorien dafür präsentierte, wie sich bestimmte Formen entwickeln anstatt sie nur nach damaliger Sitte zu klassifizieren.

§ 888 ...

§ 889 Weitere Literatur: eine grundlegende Veröffentlichung ist Turing's Paper [165], gute Bücher sind Murray [107] und [104]. Und ein schönes Buch Thompson [160]. Das Problem der Strukturbildung im zentralen Nervensystem ist z.B. in [173] dargestellt.

## 9.4 Solitonen

§ 890 In diesem Abschnitt wird das Konzept des Solitons am Beispiel einer Wasserwelle eingeführt. Auch wenn es hier besonders anschaulich wird, Solitonen sind keinesfalls nur auf Wasserwellen beschränkt. In Festkörpern lässt sich ihre Ausbildung z.B. durch Toda's Lattice (Abschn. 6.4.2) beschreiben. Optischen Solitonen in Lichtwellenleitern wurden 1973 theoretisch vorausgesagt und 1980 erstmals experimentell nachgewiesen (s.u.).

### 9.4.1 Tsunamis

§ 891 Tsunamis werden durch Seebeben ausgelöst. Ein Seebeben kann durch Anheben der lokalen Meeresoberfläche einen horizontalen Druckgradienten auslösen und damit einen Wellenberg oder einen kurzen Wellenzug über die Ozeanoberfläche senden. Da sich diese Prozesse im tiefen Wasser abspielen, ist Reibung am Boden zu vernachlässigen. Die innere Reibung ist ebenfalls vernachlässigbar, so dass wir die Welle als Oberflächenwelle beschreiben können.

<sup>1</sup>Eng damit verbunden ist natürlich auch die ethische Frage, ab wann ein menschlicher Embryo als Mensch bzw. bis wann er als Zellhaufen gilt.



Abbildung 9.2: Nachbildung von Russell's Soliton <http://www.ma.hw.ac.uk/solitons/press.html>

§ 892 Unter diesen idealen Voraussetzungen breitet sich der Wellenberg bzw. Wellenzug ungestört und unter Wahrung seiner Form über Tausende von Kilometern über die Wasseroberfläche aus. Selbst Wellen mit großer Wellenhöhe werden dabei aufgrund der langen Wellenlängen kaum wahr genommen und können unbemerkt unter einem Schiff hinweg gleiten.

§ 893 Die Formstabilität dieser Wellen ist eine bemerkenswerte Eigenschaft, da bei einer Ausbreitung über Tausende von Kilometern die Vernachlässigung der inneren Reibung problematischer ist als bei einer eher lokalen Welle. Diese 'Konstanz' ist das Merkmal eines speziellen Wellentyps, der als Soliton bezeichnet wird. Da sich Solitonen wie Teilchen und nicht wie Wellen verhalten, ist eine Ausbreitung mit Geschwindigkeiten größer als der Signalgeschwindigkeit des Mediums kein Problem: Solitonen in Luft können Überschallgeschwindigkeit annehmen. Auch die Solitonen auf dem Ozean breiten sich schneller als gewöhnliche Wasserwellen aus – daher die kurzen Vorwarnzeiten.

#### 9.4.2 Parry und die voreilige Kanone

§ 894 Die Tatsache, dass Solitonen sowohl stabil als auch supersonisch sein können, erklärt sie Beobachtungen von W.E. Parry, einem gefeierten britischen Entdeckers des frühen 19. Jahrhunderts – u.a. auf der Suche nach der Nordwest-Passage. In einem zu seinen Journalen veröffentlichten Anhang bemerkt C. Fisher [114]

On one day, and one day only, February 9, 1822, the officer's word of command "fire" was several times heard distinctly both by Captain Parry and myself about one beat of the chronometer [nearly half a second] after the report of the gun.

Dies ist der erste (bekannte) schriftliche Report der Beobachtung, dass der Schall einer feuernden Kanone früher gehört werden kann als das Kommando zum Abfeuern der Kanone.<sup>2</sup>

#### 9.4.3 Russel und Korteweg–de Vries

§ 895 Solitonen sind langlebige Wellen, die nicht dissipieren und ihre Form beibehalten. In vielen Aspekten verhalten sich Solitonen eher wie Teilchen als wie Wellen.

<sup>2</sup>Die unterschiedliche 'Behandlung' der beiden Schallsignale durch die Atmosphäre liegt an ihren unterschiedlichen Eigenschaften: das Kommando ist eine normale Schallwelle, d.h. ein Signal kleiner Amplitude, bei dem sich die Atmosphäre reversibel verändert und die Ausbreitungsgeschwindigkeit durch die Schallgeschwindigkeit limitiert ist. Die konventionelle Schallwelle ist ein einfaches lineares Phänomen. Das Feuern der Kanone dagegen erzeugt eine Stoßwelle – eine so schnelle und kräftige Kompression der Atmosphäre, dass sich deren Eigenschaften irreversibel verändern. Die Stoß- oder Explosionswelle ist daher ein nichtlineares Problem und hat nichts mit der konventionellen Schallwelle zu tun. Also gibt es keinen Grund, identische Ausbreitungsgeschwindigkeiten anzunehmen.

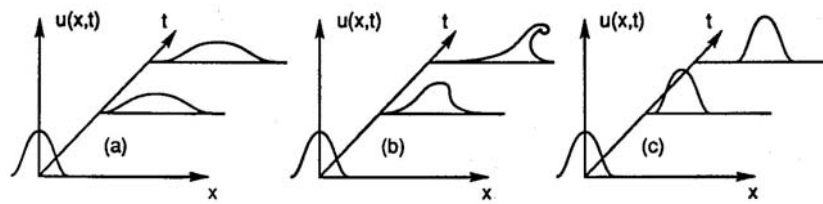


Abbildung 9.3: a) Wasserwellen laufen in flachem Wasser aufgrund der Dispersion auseinander. (b) Nicht-lineare Effekte erzeugen ein Aufsteilen der führenden Flanke der Welle. (c) Wenn sich Dispersion und nicht-lineare Effekte genau die Waage halten, entsteht ein Soliton [108]

§ 896 Die erste schriftlich überlieferte Beschreibung eines Solitons stammt von J.S. Russell [131] (in der Übersetzung in [96]; die moderne experimentelle Version ist in Abb. 9.2 gezeigt):

Ich beobachtete die Bewegung eines Schiffes, das von zwei Pferden durch schnell durch einen engen Kanal gezogen wurde. Das Schiff stoppte plötzlich, nicht jedoch die Masse des Wassers, die es in Bewegung gesetzt hatte. Diese sammelte sich am Bug des Schiffes in einem Zustand lebhaftester Bewegung, ließ ihn dann plötzlich hinter sich, rollte vorwärts mit großer Geschwindigkeit, wobei sie die Form einer großen einzelnen Erhebung annahm, eines abgerundeten, glatten und wohldefinierten Wasserbuckels, der seinen Weg fortsetzte durch den Kanal, anscheinend ohne Änderung der Form oder der Geschwindigkeit. Ich folgte ihr zu Pferde und überholte sie, wobei sie immer noch vorwärts rollte mit einer Geschwindigkeit von etwa acht oder neun Meilen pro Stunde und ihre ursprüngliche Gestalt von etwa 30 Fuß Länge und anderthalb Fuß Höhe beibehielt. Die Höhe nahm allmählich ab und nach einer Jagd von ein oder zwei Meilen verlor ich sie in den Windungen des Kanals.

§ 897 Während ihrer Ausbreitung verhalten sich Solitonen also eher wie Teilchen als wie gewöhnliche Wellen. Sie entstehen aus einem Zusammenspiel von Dispersion und Nichtlinearität. Betrachten wir dazu das in Abb. 9.3 dargestellte Beispiel zur Entstehung eines Solitons in Wasser. In Teil (a) der Abbildung ist ein zerfließendes Wellenpaket gezeigt, z.B. eine Wasserwelle in tiefem Wasser (d.h. die Wassertiefe muss mindestens gleiche der Wellenlänge sein). Aufgrund der Dispersion bewegen sich die einzelnen Wellenkomponenten mit unterschiedlicher Phasengeschwindigkeit und der Wellenberg zerfließt. In flachem Wasser, z.B. bei Auflaufen auf den Strand, ist die Situation anders (Teil (b) der Abbildung): die vordere Flanke der Welle wird gebremst während die hinteren Teile noch relativ ungestört weiter laufen. Damit steilt sich die Welle auf. Diese Aufsteilung kann so weit gehen, dass die Welle sich überschlägt. In Teil (c) der Abbildung herrscht ein Gleichgewicht aus beiden Effekten (z.B. in einem Kanal mit einer Tiefe in der Größenordnung der Wellenlänge): das Zerfließen aufgrund der Dispersion wird gerade kompensiert durch das Aufsteilen der Welle aufgrund der nichtlinearen Effekte und es bildet sich ein stabiles Soliton aus, das sich entlang des Kanals unter Beibehaltung seiner Form ausbreitet, vgl. die Beschreibung von Russell. Dies geschieht derart, dass die schnelleren Frequenzkomponenten des Wellenpakets in langsamere umgewandelt werden und umgekehrt – wobei die Bildung des Solitons voraussetzt, dass sich in dieser Umwandlung ein dynamisches Gleichgewicht ausbildet.

§ 898 Russell bemerkte ferner einen Zusammenhang zwischen Höhe und Ausbreitungsgeschwindigkeit der Welle: Wasserwellen mit hoher Amplitude breiten sich schneller aus als niedrigere Wellen. Russell hat seine Beobachtungen korrekt mit denen Parry's in Beziehung gesetzt und aus letzteren korrekt die Höhe der Erdatmosphäre abgeschätzt [132].

§ 899 Mathematisch wird ein Soliton durch eine nichtlineare partielle Differentialgleichung beschrieben, die außer der Dispersionsrelation noch ein nichtlineares Glied enthält. Die ers-

te Beschreibung erfolgte 1895 durch Korteweg und de Vries in der nach ihnen benannten Korteweg–de Vries Gleichung, hier in einer modifizierten Form:

$$\frac{\partial u}{\partial t} - 6u \frac{\partial u}{\partial x} + \frac{\partial^3 u}{\partial x^3} = 0. \quad (9.2)$$

Der zweite Term der Gleichung beschreibt die konvektiven/advektiven Effekte  $\vec{u} \cdot \nabla \vec{u}$ , die ein Aufsteilende der Welle bewirken, der dritte Term ergibt sich aus der Dispersion. Diese Gleichung erlaubt eine relativ genaue Beschreibung von recht langwelligen Wasserwellen und auch von den von Russell beschriebenen solitären Wellen in flachen Kanälen. Für eine Anwendung der Korteweg–de Vries Gleichung sei auf [108] verwiesen. Eine Einführung in die Behandlung der entstehenden nicht-linearen Differentialgleichungen mit Hilfe der Inversen Streumethode zusammen mit Beispielen für Solitonen in vielen Bereichen der Physik geben [96]; auch Scott [138, 139] geht ausführlich auf Solitonen ein.

§ 900 Die wesentlichen Merkmale von Solitonen lassen sich zusammenfassen als

- Solitonen entstehen aus dem Zusammenspiel von Nichtlinearität und Dispersion.
- Solitonen sind räumlich lokalisiert.
- Solitonen breiten sich mit (im allgemeinen von der Wellenform abhängiger) konstanter Geschwindigkeit aus, und
- Solitonen sind stabil gegenüber Stößen untereinander.

#### 9.4.4 Aufsteilen im Flachwasser

§ 901 Der zerstörerische Aspekt des Tsuanmis entsteht erst, wenn der Wellenzug in flaches Wasser gelangt. Hier gelten die gleichen Kriterien für das Aufsteilen wie bei einer normalen Welle, die auf den Strand läuft.

§ 902 Typische Parameter eines Tsuanmis sind Wellenlängen im Bereich von Hunderten von Kilometern bei Wellenhöhen im Bereich von 1 m.<sup>3</sup> Tsuanmis entstehen im freien Ozean, so dass sich auf Grund der großen Wassertiefe wegen  $c = \sqrt{gh}$  sehr hohe Ausbreitungsgeschwindigkeiten ergeben, die 700 km/h oder 20 m/s überschreiten können. Beim Auflaufen in flacheres Wasser bleibt der Energiefluss in der Welle konstant, allerdings verringert sich die Geschwindigkeit deutlich. Daher steilt sich die Welle auf, Wellenhöhe von über 30 m im Küstenbereich wurden beobachtet. Entsprechend kann sich ein Tsunami bei einem flachen Landprofil über etliche Kilometer in das Hinterland ausbreiten.

§ 903 Der Hauptentstehungsort von Tsuanmis ist der Pazifik, da sich dort die notwendige Kombination von Erdbeben (als Wellenauslöser) und hinreichend tiefem Wasser (zur Erreichung der hohen Wellengeschwindigkeiten) ergibt; Abb. 9.4 zeigt als Beispiel im oberen Teil eine Simulation für den Tsunami vom 26.12.2004 – das globale Bild der Ausbreitung gibt es unter <http://www.noaanews.noaa.gov/video/tsunami-worldpropagation2004.mov>.

§ 904 Aufgrund der großen Gefährdung küstennaher Bereiche hat sich im Pazifik ein Warnsystem für Tsuanmis etabliert.<sup>4</sup> Trotz der Erfolge dieses Systems muss man sich darüber im Klaren sein, dass das Warnsystem nur sehr kurzfristig warnen kann: Warnzeiten von 1 h setzen voraus, dass sich das Seebeben in einem Abstand von mindestens 750 km von der zu warnenden Küstenregion ereignet hat, selbst für die kurze Warnzeit von 10 min ist noch ein Abstand von über 100 km erforderlich.

<sup>3</sup>Daher wird die Welle auf freier See auch nicht bemerkt sondern geht einfach unter einem Schiff durch. Je nach Relief und zur Verfügung stehender Zeit kann die Flucht auf das Wasser hinaus eine sinnvolle Option sein.

<sup>4</sup>Ok, im wesentlichen von Japan und Hawaii getragen, so dass die Warnung für Indonesien und andere im Dezember 2004 nicht mehr rechtzeitig möglich war.

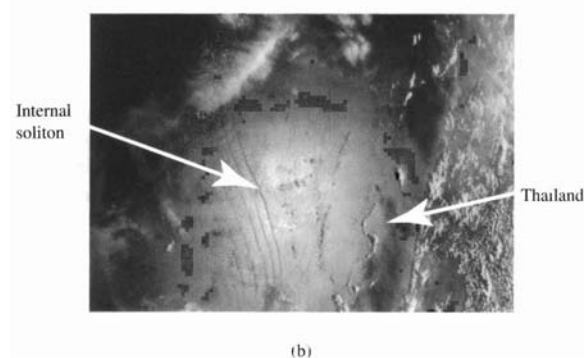
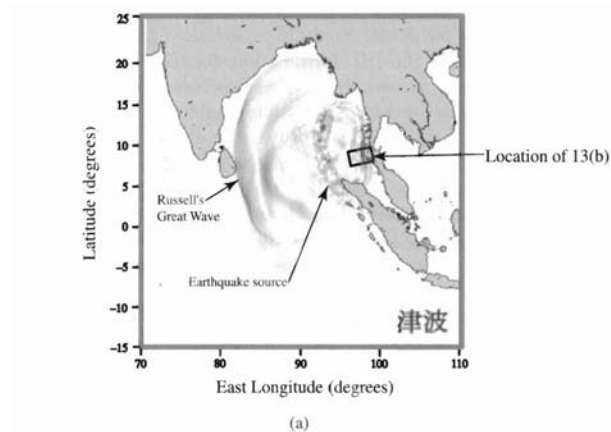


Abbildung 9.4: (a) Computer Simulation des Tsunami vom 26.12.2004 ca. 1:50 h nach dem Start (siehe auch <http://staff.aist.go.jp/kenji.satake/animation.html> für eine animierte Version); (b) Apollo-Soyuz Aufnahme der Andaman See mit Evidenz für interne Wellen; die Lage dieses Ausschnitts ist durch das Rechteck im oberen Teilbild angedeutet [139]

### 9.4.5 Die mathematische Formulierung

§ 905 Unabhängig von irgendeiner physikalischen Realisierung lässt sich ein Soliton als eine lokalisierte Lösung einer nichtlinearen partiellen Differential- oder einer Differential-Differenzgleichung beschreiben, wobei die Lösung bei der Kollision mit anderen derartigen Lösungen ihre Integrität bewahrt.<sup>5</sup> In der Regel bleibt die Energie in derartigen Systemen erhalten, es sind Hamilton-Systeme. Die Lokalisierung der Lösung kann sich, wie oben beschrieben, aus einem dynamischen Gleichgewicht von nichtlinearen und dispersiven Effekten ergeben (nicht-topologische Solitonen) oder durch topologische Randbedingungen. Eine gute Übersicht über die verschiedenen Solitonen und ihre mathematische Behandlung gibt [129].

#### Nicht-topologische Solitonen

§ 906 Die wahrscheinlich bekannteste Gleichung, die Solitonen als Lösungen zulässt, ist die bereits erwähnte und in einem Spezialfall beschriebene Korteweg-de Vries Gleichung. Diese existiert in verschiedenen Varianten, da sie von Korteweg und de Vries als mathematisches Modell eingeführt wurde, das zwar das von Russell beobachtete physikalische Phänomen beschreiben sollte, selbst aber nicht die zu Grunde liegenden physikalischen Prozesse enthält. Eine gebräuchliche Form der KdV Gleichung ist

$$u_t + \alpha u u_x + u_{xxx} = 0 \quad (9.3)$$

mit einem quadratischen nichtlinearen Term ( $\alpha u u_x$ ), der durch den dispersiven Term ( $u_{xxx}$ ) ausbalanciert wird, und den Lösungen

$$u = a \operatorname{sech}^2 \left( \sqrt{\frac{\alpha a}{12}} \left[ x - \frac{\alpha a}{3} t \right] \right). \quad (9.4)$$

<sup>5</sup>Das ist nur eine andere Formulierung dafür, dass Solitonen wie Teilchen kollidieren.

Diese beschreiben Beschreiben eine solitäre Welle mit der Amplitude  $a$ . Die Lösung hat ferner die Eigenschaft, dass Form und Geschwindigkeit bei Kollisionen beibehalten wird; die Welle verhält sich wie ein Teilchen, ein Soliton.

§ 907 Gleichung (9.3) ist eine einfache Annäherung. Ist der quadratische nichtlineare Term nur klein, so kann es sinnvoll sein, den kubischen nichtlinearen Term zusätzlich zu berücksichtigen:

$$u_t + \alpha uu_x + \beta z^2 u_x + u_{xxx} = 0.$$

Die Gleichung wird als erweiterte KdV Gleichung bezeichnet – mit  $\alpha = 0$  und  $\beta = 6$  ergibt sich die in (9.2) gegebene modifizierte KdV Gleichung mit solitären Wellen als Lösungen

$$u(x, t) = u_0 + \frac{2\nu^2}{u_0 + \sigma\lambda \cosh[2\nu(x - \mu t)]} \quad (9.5)$$

mit  $\lambda > |u_0|$ ,  $\nu = \sqrt{\lambda^2 - u_0^2}$ ,  $\sigma = \pm 1$ ,  $\mu = 2u_0^2 + 4\lambda^2$  sowie

$$u(x, t) = u_0 - \frac{4u_0}{1 + 4u_0^2(x - 6u_0^2 t)^2}.$$

Diese Lösungen streben für  $t \rightarrow \infty$  gegen  $u_0$ . Für  $u_0 \rightarrow 0$  ergibt sich aus (9.5) ein Soliton

$$u = 2\sigma\lambda \operatorname{sech}[2\lambda(x - 4\lambda^2 t)].$$

Diese Lösung unterscheidet sich von (9.4) insofern, als dass die Form  $(u - u_0)$  der Welle für  $|x| \rightarrow \infty$  algebraisch verschwindet; das sich ergebende Soliton wird als algebraisches oder rationales Soliton bezeichnet.

§ 908 Für verschiedenen Terme und Parameterkombinationen lassen sich die verschiedensten Formen von Solitonen erzeugen – alle brav mit den Namen ihrer Beschreiber versehen und für verschiedene, teilweise sehr spezielle Anwendungen gedacht.

### Einhüllende Solitonen

§ 909 Eine völlig andere Art von Solitonen ergibt sich, wenn wir die Ausbreitung einer ebenen, modulierten Welle betrachten, bei der die Dispersion von der Amplitude abhängt,  $\omega = \omega(k, |u|^2)$ . Taylor Entwicklung um die Frequenz  $\omega_0$  und die Wellenzahl  $k_0$  der Trägerwelle liefert

$$\omega - \omega_0 = \left(\frac{\partial\omega}{\partial k}\right)_0 (k - k_0) + \frac{1}{2} \left(\frac{\partial^2\omega}{\partial k^2}\right)_0 (k - k_0)^2 + \left(\frac{\partial\omega}{\partial |u|^2}\right)_0 |u|^2 \dots$$

Ersetzen wir  $\omega - \omega_0$  durch  $i\partial/\partial t$  und  $k - k_0$  durch  $i\partial/\partial x$ , d.h. führen wir eine Fourier Transformation durch, und wenden die Operatoren auf  $u$  an, so ergibt sich die nichtlineare Schrödinger (NLS) Gleichung

$$i \left[ \frac{\partial u}{\partial t} + \left(\frac{\partial\omega}{\partial k}\right)_0 \frac{\partial u}{\partial x} \right] + \frac{1}{2} \left(\frac{\partial^2\omega}{\partial k^2}\right)_0 |u|^2 u = 0.$$

Die Welle ist instabil gegenüber Modulation für

$$\left(\frac{\partial^2\omega}{\partial k^2}\right)_0 \left(\frac{\partial\omega}{\partial |u|^2}\right)_0 < 0,$$

ansonsten ist sie stabil.

§ 910 Mit etwas Skalierung und Variablentransformation lässt sich die NLS Gleichung umschreiben in die Standardform

$$iu_t + u_{xx} + 2\varepsilon |u|^2 u = 0$$

mit  $\varepsilon = \pm 1$ . Für  $\varepsilon = +1$  wird dies zur fokussierenden NLS (FNLS) Gleichung mit einer einhüllenden Lösung (envelope solution)

$$u = K e^{i(\tilde{k}x - \tilde{\omega}t + \vartheta)} \operatorname{sech}(Kx - \Omega t + \sigma).$$



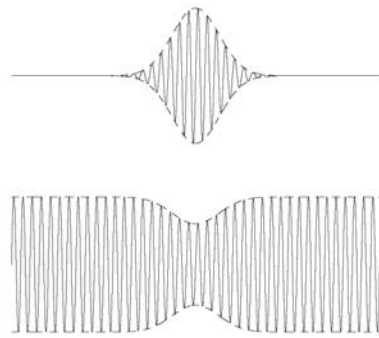


Abbildung 9.5: Helles (oben) und dunkles (unten) Soliton

Darin sind  $\vartheta$  und  $\sigma$  beliebige Konstanten und  $(K + i\tilde{k})^2 - (\Omega + i\tilde{\omega}) = 0$ . Diese Lösung entspricht in der nichtlinearen Optik dem hellen Soliton, vgl. oberes Teilbild in Abb. 9.5: nimmt die Phasengeschwindigkeit mit zunehmender Amplitude zu(ab), so führen die nichtlinearen Effekte zu einer Abnahme(Zunahme) der Wellenzahl in der vorderen Hälfte der Hüllkurve und zu einer Zunahme (Abnahme) in deren hinteren Hälfte. Damit wird die Hüllkurve zusammengesprengt und ein Soliton bildet sich in diesem dynamischen Gleichgewicht zwischen Dispersion und der durch nichtlineare Effekte bewirkten Kompression.

§ 911 Für  $\varepsilon = -1$  ergibt sich die defokussierende NLS (DNLS) Gleichung mit dem dunklen Soliton (unterer Teil von Abb. 9.5)

$$|u|^2 = u_0^2 [1 - a^2 \operatorname{sech}(u_0 a (x - vt))]$$

als Lösung. Dieses bildet eine Intensitätsdelle in einem ansonsten unendlich ausgedehnten Wellenzug. Für den Spezialfall  $a = 1$  wird das dunkle Soliton zum schwarzen Soliton, für  $a < 1$  zum grauen.

§ 912 helle optische Lösungen sind für die Nachrichtentechnik interessant: da ein Puls bei der Ausbreitung entlang einer Glasfaser durch Dispersion aufgeweitet wird, ist ein ‘Sicherheitsabstand’ zwischen aufeinanderfolgenden Pulsen erforderlich, d.h. die Übertragungsrate ist eingeschränkt. Bereits in den 1970ern schlugen daher Hasagawa und Tappert [36] die Ausnutzung nichtlinearer Änderungen der dielektrischen Konstante innerhalb der Faser zur Ausbildung von Solitonen vor; der experimentelle Nachweis erfolgte 1980 durch Mollenauer et al. [103].

### Topologische Solitonen

§ 913 Während die bisherigen Solitonen aus dem dynamischen Gleichgewicht von Dispersion und nichtlinearen Effekten entstehen, entstehen topologische Solitonen durch eine topologische Randbedingung. So, wie die KdV (und in geringerem Maße die NLS) Gleichung das mathematische Mustersystem zur Beschreibung nichttopologischer Solitonen liefert, ist die sine-Gordon (SG) Gleichung

$$u_{tt} - u_{xx} + m^2 \sin u = 0 .$$

§ 914 Die wesentlichen Eigenschaften der sine-Gordon Gleichung sind Invarianz gegenüber Lorentz-Transformationen und die Existenz mehrfacher Grundzustände in der Energie. Letzteres ist bei der Ausbildung von Solitonen hilfreich, da das Feld auf beiden Seiten einer isolierten Welle unterschiedliche asymptotische Werte annimmt. Dieses verhindert die Dispersion der Welle. Da die Begründung der entspricht, dass ein in den Randbedingungen am Ende eines Bandes enthaltener Twist nicht entfernt werden kann. Auf Grund dieser Analogie zur Topologie wird das sich ergebende Soliton als topologisches Soliton bezeichnet.

§ 915 Die Voraussetzung dafür, dass ein Wellenfeld topologische Solitonen enthält, ist die Existenz mehrfacher Grundzustände mit einer geeigneten Topology. Das entsprechende Soliton lässt sich beschreiben durch

$$u = 4 \arctan \left[ \exp \left\{ \pm m \frac{x - vt - x_0}{\sqrt{1 - v^2}} \right\} \right]$$

Dieses Soliton entspricht einem Twist in der einen Grundzustand  $2n\pi$  mit einem anderen Grundzustand  $(2n \pm 2)\pi$  verbindenden Feldkonfiguration. Die Lösung mit dem  $+$  wird als Kink oder Soliton, die mit dem  $-$  als Antikink oder Antisoliton bezeichnet. Soliton und Antisoliton haben teilweise ähnliche Eigenschaften wie geladenen Elementarteilchen: so bewegen sich in einem aus zwei Solitonen bestehenden Anfangszustand diese im Laufe der Zeit von einander weg. Daher wird der Twist eines Solitons oder Antisolitons auch als topologische Ladung bezeichnet.

### Lattice Solitonen

§ 916 Toda's Lattice Solitonen sind das Musterbeispiel für Solitonen, die sich nicht in einem kontinuierlichen System ausbilden sondern auf einem ein- oder mehrdimensionalen Gitter. Daher auch die Bezeichnung Lattice Soliton.

### Boomeronen

§ 917 Die Tsunamis erzeugenden Solitonen im Ozean entstehen in Folge von Seebeben im freien Ozean. Diese finden in der Regel in einer Wassertiefe von einigen Hundert oder gar einigen Tausend Metern statt. Dennoch sind die entstehenden Solitonen Flachwasserwellen,<sup>6</sup> da ihre Wellenlänge so lang ist. Ist der Boden nicht mehr eben, so führt dessen Relief zu einem variablen Koeffizienten in der KdV Gleichung [61]. Wird ein Soliton dabei reflektiert (es tritt von einer Seite in den entsprechenden Bereich ein und verlässt es anschließend mit entgegengesetzter Geschwindigkeit auf wieder in dieser Richtung), so wird es, in Anlehnung an den Boomerang, als Boomeron bezeichnet. Beginnt das Soliton dagegen zwischen zwei Punkten zu oszillieren ohne von diesen entweichen zu können, so wird es als Trappion bezeichnet.

## 9.5 Synchronisation

§ 918 Im klassischen Sinne bezeichnet Synchronisation die Anpassung der Rhythmen sich selbst-erhaltender periodischer Oszillationen durch ihre schwache Wechselwirkung untereinander. Formal kann Synchronization durch Phase Locking und Frequency Entrainment beschrieben werden. Moderne Konzepte betrachten verschiedenen Formen der Synchronisation, die z.B. einen Master und einen Sklaven beinhalten können oder sich nur auf die Phase beziehen.<sup>7</sup>

§ 919 Die erste (bekannte) Aufzeichnung einer Synchronisation stammt von Huygens [55] (für eine moderne experimentelle Realisierung siehe Abb. 9.6)

... it is quite worth noting that when we suspended two clocks so constructed from two hooks imbedded in the same wooden beam, the motions of each pendulum in opposite swings were so much in agreement that they never receded the least bit

<sup>6</sup>Für die Definition, ob eine gegebene Welle im tiefen oder im flachen Wasser betrachtet wird, ist nur das Verhältnis aus der Wellenlänge und der Wassertiefe maßgeblich: ... Physikalisch bedeutet die Unterscheidung zwischen Wellen im tiefen und im flachen Wasser nur, dass in letzterem die Welle durch die Eigenschaften des Meeresbodens beeinflusst wird, in ersterem jedoch nicht. Diese kann man daher als eine Welle auf einem unendlich tiefen Gewässer idealisieren. Die Flachwasserwelle dagegen wird z.B. durch Reibung beeinflusst, was zum Abbremsen des Wellenfusses und damit zum Brechen derselben bei Auflaufen an auf den Strand führt.

<sup>7</sup>Und dann gibt es noch die ganzen technischen Aspekte von Synchronization wie zwischen Notebook und PDA. Aber die interessieren hier nicht.



Abbildung 9.6: Moderne Version für Huygen's synchronisierte Uhren (<http://osiris.sunderland.ac.uk/ncaf/htm/meet.htm>)

from each other and the sound of each was always heard simultaneously. Further, if this agreement was disturbed by some interference, it reestablished itself in a short time. For a long time I was amazed at this unexpected result, but after a careful examination finally found that the cause of this is due to the motion of the beam, even though this is hardly perceptible. The cause is that the oscillations of the pendula, in proportion to their weight, communicate some motion to the clocks. This motion, impressed onto the beam, necessarily has the effect of making the pendula come to a state of exactly contrary swings if it happened that they moved otherwise at first, and from this finally the motion of the beam completely ceases. But this cause is not sufficiently powerful unless the opposite motions of the clocks are exactly equal and uniform.

§ 920 Synchronisation ist (zumindest im Rahmen dieser Vorlesung) in drei unterschiedlichen Bereichen von Interesse: bei der angetriebenen Schwingung findet eine Synchronisation zwischen einem Oszillator und einem externen Antrieb statt. Bei zwei gekoppelten Pendeln findet die Synchronisation zwischen zwei Oszillatoren statt – im Gegensatz zur erzwungenen Schwingung sind diese gleich berechtigt. Die gegenseitige Synchronisation in einer großen Population von Oszillatoren ist die Erweiterung der beiden gekoppelten Oszillatoren.

### 9.5.1 Zum Einstieg: erzwungene Schwingung

§ 921 Formal ist die wesentliche Eigenschaft zur Betrachtung einer Synchronisation die Phase  $\varphi$ . Mathematisch lässt sie sich als die Parametrisierung der Bewegung eines autonomen, zeitkontinuierlichen Systems im Phasenraum für den stabilen Grenzfall beschreiben. Die Phase kann daher monoton mit der Zeit anwachsen,  $\dot{\varphi} = \omega_0$  mit  $\omega_0$  als der Frequenz der freien Schwingung des Oszillators.

§ 922 Die Phase ist im indifferenten Gleichgewicht (neutrale Stabilität): Störungen werden weder gedämpft noch wachsen sie an. Das bedeutet jedoch auch, dass eine kleine äußere Störung (z.B. ein äußerer Zwang mit einer anderen Frequenz oder die Kopplung an einen anderen Oszillator) im Laufe der Zeit zu großen Abweichungen in der Phase führen kann. Ein Oszillator mit der freien Frequenz  $\omega_0$  und einem externen periodischen Antrieb mit Frequenz  $\omega$  und Amplitude  $\varepsilon$  kann dann beschrieben werden als

$$\dot{\varphi} = \omega_0 + \varepsilon Q(\varphi, \omega t) .$$

Darin ist  $Q$  eine Kopplungsfunktion, die in beiden Argumenten periodisch mit Periode  $2\pi$  ist; ihre Details hängen von den Eigenschaften der freien Schwingung und des externen Antriebs ab.

§ 923 In der Nähe der Resonanz, d.h. für  $\omega \approx \omega_0$  enthält die Kopplungsfunktion  $Q$  schnelle oszillierende and langsam variierende Terme. Letztere können angenähert werden als  $q(\varphi - \omega t)$ . Im Mittel über eine Schwingung lässt sich die Dynamik der Phase damit schreiben als

$$\frac{d\Delta\varphi}{dt} = -(\omega - \omega_0) + \varepsilon q(\Delta\varphi) \quad (9.6)$$

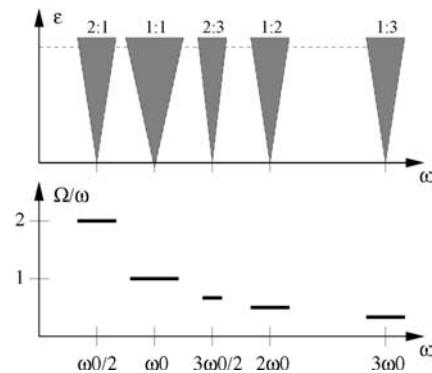


Abbildung 9.7: Arnold's Tongue (oben) und Devil's Staircase (unten)

mit  $\Delta\varphi = \varphi - \omega t$  als der Differenz zwischen der Phase des Oszillators und der der antreibenden Kraft. Die Funktion  $q$  ist periodisch mit Periode  $2\pi$ , im einfachsten Fall lässt sich  $q$  als Sinus schreiben. Gleichung 9.6 wird als Adler Gleichung bezeichnet.

§ 924 Im Parameterraum des externen Antriebs ergeben sich im Bereich  $\varepsilon q_{\min} < \omega - \omega_0 < \varepsilon q_{\max}$  stabile stationäre Lösungen der Adler Gleichung (9.6). Diese Lösungen entsprechen dem Phase Locking, d.h. die Phase  $\varphi$  der erzwungenen Schwingung folgt der des Forcings derart, dass  $\varphi - \omega t = \text{const}$ , und dem Frequency Entrainment, d.h. die Frequenz  $\Omega = \dot{\varphi}$  des Oszillators entspricht exakt der Frequenz  $\omega$  der anregenden Kraft.

§ 925 Synchronisation kann auch für Resonanzen höherer Ordnung auftreten: dann entspricht die Frequenz der erzwungenen Schwingung nicht exakt der der Anregung sondern die beiden stehen in einem ganzrationalen Verhältnis:  $n\omega \approx m\omega_0$ . Auch in diesem Fall wird die Dynamik der Phasendifferenz  $\Delta\varphi = m\varphi - n\omega t$  durch eine Gleichung ähnlich der Adler Gleichung (9.6) beschrieben:

$$\frac{d\Delta\varphi}{dt} = -(n\omega - m\omega_0) + \varepsilon\tilde{q}(\Delta\varphi).$$

Ein synchrones Regime entspricht einem perfekten Entrainment der Frequenz des Oszillators bei einem rationalen Vielfachen der Anregungsfrequenz,  $\Omega = n\omega/m$ , sowie einem Phase Locking  $m\varphi = n\omega t + \text{const}$ .

§ 926 Die allgemeine Darstellung erfolgt in der  $\omega, \varepsilon$ -Ebene durch ein System von dreieckigen Synchronisations Bereichen (Arnold's Zungen, Arnold's Tongue, oberes Teilbild in Abb. 9.7), deren Spitze die  $\omega$ -Achse jeweils bei rationalen Vielfachen der Frequenz der freien Schwingung berührt. Ein ähnliches Bild ergibt sich auch für ein nur moderates Forcing, allerdings behalten die Synchronisationsbereiche dabei nicht die dreieckige Form. Der Devil's Staircase im unteren Teilbild von Abb. 9.7 besteht aus horizontalen Plateaus bei allen möglichen rationalen Frequenzverhältnissen: für eine feste Amplitude  $\varepsilon$  der Anregung und eine variable Anregungsfrequenz  $\omega$  ergeben sich unterschiedliche Intervalle der Synchronisation. Hier ist die Bewegung periodisch, dazwischen dagegen quasi-periodisch.

§ 927 Die erzwungene Schwingung ist von diversen mechanischen Pendeln und dem Schwingkreis bekannt, aber wahrscheinlich nicht gerade das Phänomen, welches man unter der Rubrik Synchronisation betrachten würde. Das wesentliche Merkmal der Synchronisation in diesem Fall ist der externe 'Taktgeber'. Derartigen Taktgebern begegnen wir im Alltag häufig: die konventionell Funkuhr ist ein einfaches technisches Beispiel für eine Synchronisation. Biologischer Rhythmen werden teilweise ebenfalls synchronisiert: der Sinusknoten (Purkinje Fasern) synchronisiert die Herzzellen und damit den Herzschlag – wird er unzuverlässig, so lässt sich diese Funktion an einen externen Taktgeber, den Herzschrittmacher, delegieren. Fehlerhafte Synchronisation von Gehirnzellen lässt sich ebenfalls mit Hilfe eines Schrittmachers regulieren

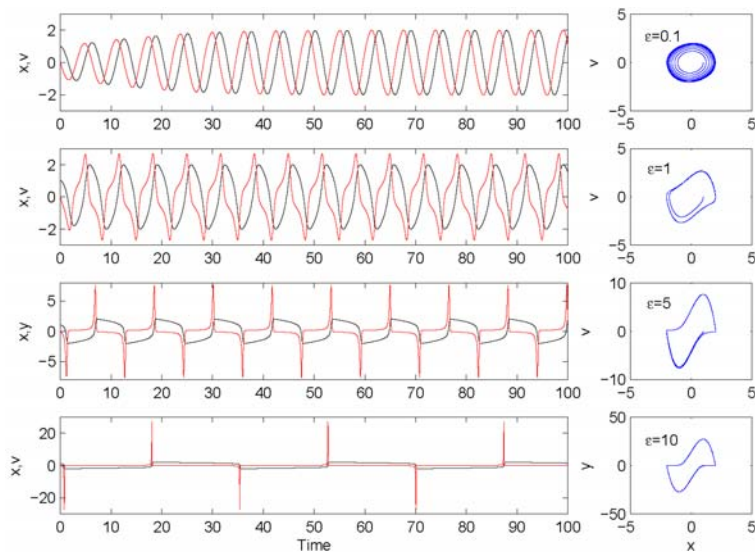


Abbildung 9.8:  
 Lösungen von van der Pol's Gleichung für verschiedene Werte des Parameters  $\varepsilon$ : für kleine Werte ergibt sich eine harmonische Schwingung, für große eine relaxierende Oszillation

(... bei Parkinson). Auch auf der Ebene des Gesamtorganismus werden verschiedenen biologische Funktionen (Herzfrequenz, Körpertemperatur, Hormonausschüttung, Schlaf-Wach-Rhythmen) durch einen externen Taktgeber synchronisiert: alle quasi-circadianen Rhythmen werden durch das Sonnenlicht synchronisiert. Sperrt man Menschen in Höhlen oder sonstige künstliche Settings, in denen sie keinerlei Informationen über den externen Tagesverlauf erhalten, so fehlt der externe Taktgeber und es bilden sich, in der Regel länger dauernde, Schlaf-Wach-Rhythmen aus. Viele dieser biologischen Beispiele für Synchronisation sind im Detail in [174] beschrieben.

### 9.5.2 Zwei gekoppelte Oszillatoren

§ 928 Die Synchronisation zwischen zwei gekoppelten Oszillatoren kann auf gleiche Weise beschrieben werden. Eine schwache Kopplung betrifft lediglich die Phasen  $\varphi_1$  und  $\varphi_2$  der beiden Oszillatoren:

$$\dot{\varphi}_1 = \omega_1 + \varepsilon Q_1(\varphi_1, \varphi_2) \quad \text{und} \quad \dot{\varphi}_2 = \omega_2 + \varepsilon Q_2(\varphi_2, \varphi_1).$$

Durch Mittelung erhält man für die Phasendifferenz  $\Delta\varphi = \varphi_2 - \varphi_1$  einen Ausdruck ähnlich dem der Adler Gleichung 9.6.

§ 929 In diesem Fall bedeutet Synchronisation das von Huygens Beschriebene: zwei nicht-identische Oszillatoren beginnen mit der gleichen Frequenz (oder allgemeiner mit einem rationalen Frequenzverhältnis) zu oszillieren. Die gemeinsame Frequenz liegt üblicherweise zwischen  $\omega_1$  und  $\omega_2$ .

§ 930 Das Phase Locking ebenso wie das Frequency Locking beinhaltet keine Beschränkungen bezüglich Amplitude und Wellenform; die Oszillationen können auch Relaxation (integrate and fire) oder quasi-harmonisch sein. Harmonische (oder quasi-harmonische) Oszillationen erwarten wir insbesondere in Systemen, die sich mit der Hamilton'schen Dynamik beschreiben lassen, d.h. in denen Energieerhaltung gilt. Ein typisches dissipatives System mit einer limit-cycle Oszillation ist durch van der Pol's Gleichung [167]

$$\ddot{x} - \varepsilon(1 - x^2)\dot{x} + \omega^2 x = 0$$

beschrieben. In Abhängigkeit vom Parameter  $\varepsilon$  variiert die Schwingung: für kleine  $\varepsilon$  ergibt sich eine harmonische und Energie erhaltende Bewegung  $x(t) = \sin(\omega t + \varphi_0)$ ; für große  $\varepsilon$  ergeben sich blockierende oder relaxierende Schwingungsformen. Die Wellenform eines relaxierenden Oszillators dagegen besteht aus abwechselnd schnellen und langsamen Bewegungen,

ähnlich den Spikes und langsamen Erholung bei einem feuernenden Neuron und den stick-slip Oszillationen bei Bewegung mit Reibung.

**Zwischenrechnung 53** Mit van der Pol können Sie in Abwandlung des Modells für die gewöhnliche gedämpfte Schwingung selbst ein wenig herum spielen – das ist auch im Hinblick auf Stabilität und Schrittweite im numerischen Verfahren interessant.

§ 931 Genauso wie der harmonische Oszillator lässt sich ein van der Pol Oszillator durch eine externe Anregung synchronisieren. Dies ist um so leichter möglich, je dichter die anregende Frequenz an der der freien Schwingung des van der Pol (VDP) Oszillators, je größer  $\varepsilon$  und je stärker die Kopplung. Auch hier bilden sich wieder die aus Abb. 9.7 bekannten Arnold'schen Zungen.

### 9.5.3 Eine Population gekoppelter Oszillatoren

§ 932 Die Synchronisation einer großen Zahl gekoppelter Oszillatoren lässt sich als ein Nicht-Gleichgewichts-Phasenübergang beschreiben mit dem mittleren oszillierenden Feld als Ordnungsparameter. Die alltäglichen Beispiele sind die gleichzeitigen Lichtblitze von Glühwürmchen oder der synchronisierte Applaus in einem großen Publikum.

§ 933 Formal schwieriger zu handhaben sind die Populationen teilweise gekoppelter Oszillatoren, wie z.B. bei der Erklärung von Aktivitätsmustern in der Großhirnrinde: hier synchronisiert nicht die gesamte Neuronenpopulation sondern die Musterkodierung erfolgt eben gerade dadurch, dass eine Teilpopulation sich synchronisiert. Je nach relativem Anteil dieser Population ist die Beschreibung als Phasenübergang möglich oder es sollte lieber die Kopplung einer größeren Zahl einzelner Oszillatoren beschrieben werden.

§ 934 Formal lässt sich die Synchronisation einer Zahl von  $N$  Oszillatoren mit geringfügig unterschiedlichen Frequenzen  $\omega_i$  ihrer freien Schwingung schreiben als

$$\dot{\varphi} = \omega_i + \sum_{j=1}^N \Gamma_{ij}(\varphi_i - \varphi_j) \quad i = 1, 2, \dots, N$$

mit  $\Gamma$  als einer in  $2\pi$  periodischen Funktion. Diese Betrachtung lässt sich auch auf Oszillatoren ausweiten, die ein kontinuierliches Feld bilden.

§ 935 Weitere Literatur: eine anschauliche Einführung gibt [156], eine weitergehende [121]. Technische Systeme werden in [8] betrachtet, chemische in [77], biologische in [174] und physiologische in [32]. Die Anwendung auf lebende Systeme (und damit chaotische Synchronisation) wird in [105] betrachtet.

## 9.6 Schmetterlingseffekt

But can modeling have an impact on society? Can modeling really allow or predictions? Isn't there Lorenz' butterfly effect<sup>8</sup> [85]: "Does the flap of a butterfly's wing in Brazil set off a tornado in Texas?"

While Lorenz made the experience of the small error in initial conditions, Alan Turing [164] had made a similar statement about two decades earlier out of reasoning about computers and in particular the later so-called Turing test:

<sup>8</sup>The tale goes that Lorenz discovered this effect after a computer breakdown in his lab at MIT. Lorenz was a pioneer in numerical weather forecast. One day, he restarted a run after a computer breakdown with the last saved data set as initial conditions for the run's continuation. When he compared the results to the results from a complete run, obvious differences appeared. He concluded that the rounding errors made as the saved data were fed back into the program were large enough to affect the program's output strongly. To illustrate that such a small deviation can make a strong effect, he created the above statement.

The system of the ‘universe as a whole’ is such that quite small errors in the initial conditions can have an overwhelming effect at a later time. The displacement of a single electron by a billionth of a centimeter at one moment might make the difference between a man being killed by an avalanche a year later, or escaping.

Turing’s statement is even more fatal than Lorentz’ because it retracts to the quantum mechanical level. If the world model or the general model of the universe including details down to the atomic and sub-atomic level is the target, Turing’s objections most likely are correct. However, whether a model allows for a prediction or not, depends on the question and the model. A dike reeve, for instance, wants to know whether a certain stretch of dike will hold or break – it does not matter in detail where and how it will break.<sup>9</sup> Thus if we judge a model in its ability to reproduce the intimate details of the process under study, most models probably will fail. But if we ask the more general question, the model, at least if it is a good working model, probably will give a reasonable answer. The same is true for climate models. The large number of different climate models also can be used to confirm the trends in the results: although all models lead to different predictions about the exact temperature evolution until 2100 under the constraint of increased carbon dioxide levels, all models agree on the warming effect and all models lead to predictions in a rather small range of temperature changes, see e.g. [58]. And this agreement despite the differences in the models suggests that it might be time to think about measures to take even if models are not perfect. This problem is discussed in detail in Pollack’s book *Uncertain science ... uncertain world* [123].

In sum, although models do not predict the future in all detail, good models give quite reasonable ideas about it. For instance, weather forecast is well advanced and gives pretty reliable predictions for the general weather pattern over Germany for about five days in advance – also it always almost fails to tell me whether I will need an umbrella on my way to the mensa or not. The path of hurricane Katrina has been predicted fairly well – as had been the inability of New Orleans’ dike system to withstand such a hurricane. Tsunami prediction in the Pacific works reasonably well and is greatly appreciated by the surrounding countries. And even the rather crude models two decades ago made reasonable predictions about the fall out following the Chernobyl accident and the burning oil fields in the first Iraqi war.

---

<sup>9</sup>Probably nobody in New Orleans cared about which part of the dike broke – the fact that some dike broke was fatal enough.

# Kapitel 10

## Nützliche Ergänzungen

### 10.1 Formelzeichen und Abkürzungen

#### 10.1.1 Formelzeichen

$\beta$	Reibungskoeffizient
$\beta$	Separationskonstante
$\gamma$	Reibungskoeffizient/Masse, $\gamma = \beta/2m$
$\gamma$	Integrationskonstante bei DGL
$\varepsilon$	irgendeine Eigenschaft
$\varphi$	Phase
$\lambda$	Eigenwert
$\lambda$	thermische Leitfähigkeit
$\omega_0$	Frequenz der freien Schwingung ( $\omega_0^2 = k/m$ )
$\omega$	(kreis-)Frequenz
$\Omega$	Frequenz (der antreibenden Kraft)
$\rho$	Dichte
$A$	Amplitude
$A^c$	komplexwertige Amplitude
$\vec{B}$	magnetische Flussdichte
$c$	Lichtgeschwindigkeit
$C$	Kapazität
$C$	Klimakosten
$C_a$	Vermeidungskosten
$C_d$	Klimafolgekosten
$\vec{C}(\varepsilon)$	Fluss der Eigenschaft $\varepsilon$
$D$	Diffusionstensor
$D$	Diffusionskoeffizient
$e$	Emissionsfunktion
$f$	Frequenz
$f_A$	Abtastfrequenz
$f_A$	Dichte der antreibenden Kraft, $f_A = F_A/m$
$F_A$	antreibende Kraft
$g$	Gravitationsbeschleunigung
$I$	Strom (konstant)
$i$	Strom (variabel, $i(t)$ )
$\vec{j}$	Stromdichte
$k$	Federkonstante



$L$	Induktivität
$m$	Masse
$Q$	Ladung (konstant)
$Q$	Quellfunktion (thermisch)
$q$	Ladung (variabel, $q(t)$ )
$R$	Widerstand
$R_\omega$	Impulsantwort $\text{CO}_2$ Konzentration
$R_T$	Impulsantwort Temperatur
$R$	Impulsantwort Atmosphäre
$S(\varepsilon)$	Quellen und Senken der Eigenschaft $\varepsilon$
$t$	Zeit
$T$	Temperatur
$T$	Periode, Schwingungsdauern
$T_A$	Abtastintervall
$U$	Spannung (konstant)
$u$	Spannung (variabel, $u(t)$ )
$u$	bulk velocity (mittlere Geschwindigkeit eines kont. Mediums)
$v$	Geschwindigkeit
$v_A$	Alfvén-Geschwindigkeit
$x$	Ort
$z$	Ort (komplex)

### 10.1.2 Abkürzungen

BAU	Business as usual
DCT	discrete Cosinus transform, diskrete Kosinus Transformation
DFT	diskrete Fourier Transformation
DGL	Differentialgleichung
DNLS	defokussierende nichtlineare Schrödinger Gleichung
EDA	Electronic Design Automation
FFT	fast Fourier transform, schnelle Fourier Transformation
FIR	finite impulse response
FNLS	fokussierende nichtlineare Schrödinger Gleichung
FPU	Fermi–Pasta–Ulam Problem
FT	Fourier Transformation
GCM	Global Circulation Model
GES	Global Environment and Society (GES) Model
GW	Global Welfare (globales Wohlergehen)
IIR	infinite impulse response
IPCC	Intergovernmental Panel on Climate Change, <a href="http://www.ipcc.ch">www.ipcc.ch</a>
KdV	Korteweg–de Vries
IST	Inverse Streu-Transformatio
LTI	linear time invariant
NLS	nichtlineare Schrödingergleichung
ODE	ordinary differential equation
PDE	partial differential equation
PDGL	partielle Differentialgleichung
SG	Sine–Gordon Gleichung
TFT	Tit for Tat
VDP	van der Pol

## 10.2 Numerische Integration

§ 936 Numerische Verfahren zur Lösung gewöhnlicher Differentialgleichungen werden auf konventionelle Verfahren zur numerischen Integration zurück geführt. Eine homogene DGL erster Ordnung lässt sich analytisch mit Hilfe des Separationsansatzes lösen:

$$\dot{x} = f(x(t)) \Rightarrow \int_t^{t+\Delta t} dx = \int_t^{t+\Delta t} f(x(t)) dt \Rightarrow x(t + \Delta t) = x(t) + \int_t^{t+\Delta t} f(x(t)) dt .$$

§ 937 Für das numerische Verfahren wird die DGL diskretisiert: die Differentialgleichung wird in eine Differenzgleichung überführt:<sup>1</sup>

$$\frac{\Delta x}{\Delta t} = \frac{x_n - x_{n-1}}{\Delta t} = f(x, t) .$$

Der dabei entstehende Fehler wird als Diskretisierungsfehler bezeichnet. Er wird, ebenso wie die Stabilität eines numerischen Schemas, durch die Schrittweite  $h$  bestimmt:

$$h = \Delta x = \frac{b - a}{M}$$

mit dem Integrationsintervall  $[a, b]$  und  $M$  als der Zahl der Schritte.

§ 938 Differentialgleichungen  $n^{\text{ter}}$  Ordnung werden durch Substitution in  $n$  DGLs erster Ordnung überführt; vgl. Abschn. 1.1.2.

### 10.2.1 Einfache Verfahren mit fester Schrittweite

§ 939 Verfahren fester Schrittweite sind die traditionellen Verfahren, das typische Beispiel ist das Euler Verfahren. Sie sind einfach (allerdings auch ungenau), auch die Fehler und Ungenauigkeiten sind leicht abzuschätzen. Die hier vorgestellten Verfahren haben alle ihre Paten in den numerischen Integrationsverfahren:

- Gesucht:

$$I = \int_a^b f(x) dx .$$

Unabhängig vom Verfahren wird das Integrationsintervall  $[a, b]$  kann in  $M$  Schritte der Länge  $\Delta x$  eingeteilt:  $\Delta x = (b - a)/M$ .

- bei der Mittelpunktsformel wird jeweils der Funktionswert in der Intervallmitte zur Bestimmung des Flächenelements verwendet:

$$I_{\text{MP}} = \sum_{k=1}^M f(x_{mk}) \Delta x = \sum_{k=1}^M f\left(\frac{x_{k-1} + x_k}{2}\right) \Delta x .$$

- bei der Trapezformel wird das Flächenstück als durch die Funktionswerte an den Intervallgrenzen bestimmtes Trapez angenähert:

$$I_{\text{T}} = \sum_{k=1}^M \frac{1}{2} (f(x_k) + f(x_{k-1})) \Delta x .$$

- beim Simpson-Verfahren wird die Funktion durch ein Polynom angenähert, so dass sich für das Flächenstück ergibt

$$I_{\text{Simpson}} = \frac{1}{6} \sum_{k=1}^M (f(x_{k-1}) + 4f(x_{mk}) + f(x_k)) \Delta x .$$

<sup>1</sup>Die einfachere Interpretation der Differenzgleichung geht von der Differenzgleichung aus und verzichtet auf die Grenzwertbildung im Übergang vom Differenzen- zum Differentialquotienten.

### Euler Verfahren

§ 940 Das Euler Vorwärts Verfahren ist das einfachste Verfahren: aus dem bekannten Anfangswert  $x_0(t_0)$  wird mit Hilfe der bekannten Steigung  $f(t_0, x_0)$  der Funktionswert  $x_1$  am Ende des ersten Integrations schrittes berechnet:

$$x_1 = x_0 + \Delta t f(t_0, x_0) .$$

Dabei wird die Steigung im Anfangspunkt des Intervalls als für das gesamte Intervall  $\Delta t$  konstant angenommen. Der so erhaltene Funktionswert am Ende des Intervalls ist gleichzeitig der Anfangswert für das folgende Intervall; das Euler Vorwärts Verfahren ist damit gegeben als

$$x(t_{i+1}) = x(t_i) + \Delta t f(t_i, x_i) \quad \text{mit} \quad x(t_0) = x_0 .$$

Da in diesem Verfahren der Wert  $x(t_{i+1})$  nur vom vorangegangenen Wert  $x(t_i)$  abhängt, handelt es sich um ein explizites Verfahren.

§ 941 Alternativ können wir auch die (noch nicht bekannte) Steigung am Ende des Integrationsintervalls verwenden:

$$x(t_{i+1}) = x(t_i) + \Delta t f(t_{i+1}, x_{i+1})$$

mit  $x(t_0) = x_0$  und  $t_i = t_0 + i\Delta t$ . Dieses Verfahren wird als Rückwärts Methode bezeichnet. Im Gegensatz zum Vorwärts Verfahren hängt der Wert  $x(t_{i+1})$  nicht nur vom vorher bestimmten Wert  $x(t_i)$  ab sondern auch vom zu bestimmenden Wert  $f(t_{i+1}, x_{i+1})$ . Das Verfahren ist daher ein implizites.

§ 942 Die näherungsweise Berechnung erfolgt im sogenannten modifizierten Euler Verfahren in zwei Schritten: (1) im Prädikatorschritt wird der Wert an der Stelle  $i+1$  mit dem Vorwärts-Verfahren berechnet:

$$x^P(t_{i+1}) = x(t_i) + \Delta t f(t_i, x_i) .$$

Im Korrektorschritt wird dieser Wert zur Bestimmung der Steigung verwendet und damit der eigentlich gesuchte Wert an der Stelle  $i+1$  bestimmt:

$$x(t_{i+1}) = x(t_i) + \Delta t f(t_{i+1}, x_{i+1}^P) .$$

Das modifizierte Euler Verfahren ist noch ein explizites Verfahren. Ein echtes implizites Verfahren entsteht durch Iteration.

### Crank–Nicolson Verfahren

§ 943 Das Crank–Nicolson Verfahren kombiniert die Vorwärts und Rückwärts Methoden aus dem Euler Verfahren:

$$x(t_{i+1}) = x(t_i) + \frac{\Delta t}{2} (f(t_i, x_i) + f(t_{i+1}, x_{i+1}))$$

mit  $x(t_0) = x_0$  und  $t_i = t_0 + i\Delta t$ . Durch die Kombination der beiden Euler Verfahren ist das Crank–Nicolson Verfahren ein kombiniertes explizites/implizites Verfahren.

### Leap-Frog

§ 944 Bei den bisherigen Verfahren, egal ob explizit oder implizit, wurde die Steigung jeweils an einer Intervallgrenze (untere beim expliziten, obere beim impliziten Verfahren) bestimmt über den gesamten Integrations schritt als konstant angenommen. Das Leapfrog Verfahren verwendet die Steigung in der Intervallmitte, diese wird wieder über den gesamten Integrations schritt als konstant betrachtet. Dazu werden zwei Euler Verfahren parallel verwendet: eins an den Stützstellen der Integrationsintervalle, das andere in deren Mitte. Diese werden abwechselnd iteriert:

$$\begin{aligned} x_{i+\frac{1}{2}} &= x_{i-\frac{1}{2}} + f(t_i, x_i)\Delta t & \text{mit} & \quad x_{-\frac{1}{2}} = x_0 - \frac{1}{2}f(t_0, x_0)\Delta t \\ x_{i+1} &= x_i + f(t_{i+\frac{1}{2}}, x_{i+\frac{1}{2}})\Delta t & \text{mit} & \quad x_0 = x_0 . \end{aligned}$$

### Runge–Kutta

§ 945 Das Runge–Kutta Verfahren 4. Ordnung ist ein sehr genaues Rechenverfahren. Wie beim Euler Verfahren wird die Lösungskurve in jedem Teilintervall  $\Delta t$  durch eine Gerade angenähert. Allerdings wird die Steigung dieser Geraden nicht am Anfang oder wie beim Leapfrog Verfahren in der Mitte des Intervalls bestimmt sondern es wird eine mittlere Steigung angenommen, die sich aus den Steigungen in den beiden Randpunkten und in der Intervallmitte zusammensetzt:

$$x(t_i) = x_i = x_{i-1} + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

mit

$$\begin{aligned} k_1 &= \Delta t f(t_{i-1}, x_{i-1}), & k_2 &= \Delta t f\left(t_{i-\frac{1}{2}}, x_{i-1} + \frac{k_1}{2}\right), \\ k_3 &= \Delta t f\left(t_{i-\frac{1}{2}}, x_{i-1} + \frac{k_2}{2}\right), & k_4 &= \Delta t f(t_{i-1} + \Delta t, x_{i-1} + k_3). \end{aligned}$$

Dabei müssen die Wichtungsfaktoren  $k_i$  für jeden Rechenschritt  $\Delta t$  neu berechnet werden.

## 10.2.2 Stabilität und Fehler

### Diskretisierung

§ 946 Diskretisierung kann formal über die Taylor Entwicklung des Differentialquotienten hergeleitet werden. Dazu wird die Ausgangsfunktion an der Stelle  $x_i$  entwickelt:

$$u_{k+1} = f(x_{k+1}) = f(x_k + h) = f(x_k) + h f'(x_k) + \frac{h^2}{2!} f''(x_k) + \frac{h^3}{3!} f'''(x_k) + \mathcal{O}(h^4).$$

Mit  $f(x_k) = u_k$  lässt sich dies schreiben als

$$\frac{u_{k+1} - u_k}{h} = f'(x_k) + \frac{h}{2!} f''(x_k) + \frac{h^2}{3!} f'''(x_k) + \mathcal{O}(h^4). \quad (10.1)$$

Für  $h \rightarrow 0$  ist die linke Seite die Definition der Ableitung und kann daher als finite Differenzen Annäherung an die Ableitung  $f'(x_k)$  gelesen werden. Der führende Term des Fehlers ist  $h f''(x_k)/2! = \mathcal{O}(h)$ ; er ist von erster Ordnung in  $h$ . Daher wird eine derartige Diskretisierung auch als “genau von erster Ordnung” bezeichnet. Das Schema wird als Vorwärts Schema bezeichnet; es entspricht der Art von Diskretisierung, die wir auch anschaulich verwendet haben.

§ 947 Ersetzen wir  $h$  durch  $-h$ , so ergibt sich ein Rückwärts Schema. Taylor Entwicklung liefert

$$u_{k-1} = f(x_{k-1}) = f(x_k - h) = f(x_k) - h f'(x_k) + \frac{h^2}{2!} f''(x_k) - \frac{h^3}{3!} f'''(x_k) + \mathcal{O}(h^4)$$

und damit als Annäherung an die Ableitung

$$\frac{u_k - u_{k-1}}{h} = f'(x_k) - \frac{h}{2!} f''(x_k) + \frac{h^2}{3!} f'''(x_k) + \mathcal{O}(h^4). \quad (10.2)$$

Auch dieses Verfahren ist einseitig und von erster Ordnung, nur eben rückwärts gerichtet.

§ 948 Der Mittelwert dieser beiden Schemata liefert eine zentrale Differenz

$$\frac{u_{k+1} - u_{k-1}}{2h} = f'(x_k) + \frac{h^2}{3!} f'''(x_k) + \mathcal{O}(h^4).$$

Hier heben sich die Terme mit der zweiten Ableitung  $f''(x_k)$  heraus, der Fehler beginnt erst mit dem  $h^2$ -Term. Das Verfahren ist daher von zweiter Ordnung und damit bei gleicher Schrittweite  $h$  genauer als das Vorwärts oder das Rückwärts Schema.

§ 949 Die Differenz aus Vorwärts Schema (10.1) und Rückwärts Schema (10.2) gibt eine Annäherung an die zweite Ableitung:

$$\frac{u_{k+1} - 2u_k + u_{k-1}}{h^2} = f''(x_k) + \frac{h^2}{4!} f'''(x_k) + \mathcal{O}(h^4). \quad (10.3)$$

Auch hierbei handelt es sich um ein zentriertes Schema mit einer Genauigkeit von zweiter Ordnung. Mit Hilfe dieser Diskretisierung lassen sich auch zweite Ableitungen für ein numerisches Schema aufbereiten ohne dass die DGL zweiter Ordnung in zwei DGLs erster Ordnung zerlegt werden muss.

### Konsistenz und Genauigkeit

§ 950 Der Fehler des numerischen Verfahrens entsteht durch die Vernachlässigung der höheren Terme der Taylor Entwicklung bei der Diskretisierung. Quantifiziert wird dieser Fehler durch den Truncation Error TE (Abbruchfehler):

**Definition 17** Der Truncation Error  $TE \varepsilon_i$  wird bestimmt, in dem die exakte Lösung  $\hat{u}_k$  an Stelle der angenäherten Lösung in die Differenzgleichung eingesetzt wird.

§ 951 Zur Zerfallsgleichung

$$\frac{dN}{dt} = \dot{N} = -\lambda N$$

gehört die Differenzgleichung

$$\frac{N_{k+1} - N_k}{\Delta t} + \lambda N_k = 0.$$

Der Truncation Error ist mit  $\hat{N}_k$  als exakter Lösung an Stelle der Lösung  $N$  des numerischen Schemas

$$\varepsilon_k = \frac{\hat{N}_{k+1} - \hat{N}_k}{\Delta t} + \lambda \hat{N}_k \neq 0.$$

Taylor Entwicklung der Ausgangsgleichung (die Differenzgleichung kann nicht entwickelt werden, da sie nur an diskreten Stützstellen definiert ist) liefert

$$\varepsilon_k = \left( \frac{\hat{N}_{k+1} - \hat{N}_k}{\Delta t} + \lambda \hat{N}_k \right) + \frac{\Delta t}{2} \frac{d^2 \hat{N}_k}{dt^2} + \mathcal{O}((\Delta t)^2).$$

Da die exakte Lösung die DGL erfüllt, verschwinden die Terme in der Klammer. Der verbliebenen Truncation Error ist

$$\varepsilon_k = \frac{\Delta t}{2} \frac{d^2 \hat{N}_k}{dt^2} + \mathcal{O}((\Delta t)^2)$$

und damit von erster Ordnung un  $\Delta t$ .

§ 952 In den Abbruchfehler geht die gewählte Schrittweite ein, ebenso wie die zweite Ableitung der exakten Lösung, d.h. die Änderung der Steigung der gesuchten Funktion. Das ist anschaulich verständlich: im Idealfall einer konstanten Steigung im Intervall  $\Delta t$  liefert das Euler'sche Streckenzugverfahren für den Wert am Intervallende nicht nur eine Näherung sondern den exakten Wert. Dann ist  $\varepsilon_k = 0$  weil  $\ddot{N}_k = 0$ . Je geringer die Änderung der Steigung in  $\Delta t$  ist, um so stärker nähert sich der Wert am Intervallende an die exakte Lösung an, d.h.  $\varepsilon_k \sim \ddot{N}$  ist zumindest plausibel. Die Abhängigkeit von  $\Delta t$  ist ebenfalls anschaulich: je kleiner der Rechenschritt, um so früher korrigieren wir die fälschliche Annahme  $\dot{N} = \text{const}$  des numerischen Verfahrens.

§ 953 Die Qualität eines numerischen Differenzschemas hängt von der Größe des Abbruchfehlers  $\varepsilon_k$  ab. Manchmal ist  $\varepsilon_k$  bei der Lösung einer bestimmten DGL klein: allerdings nicht, weil das verwendete Schema so gut ist, sondern weil die Eigenschaften oder Parameter der DGL so sind, dass der Fehler des Schemas durch teilweise kompensiert wird. Dann ist das Schema für andere DGLs oder Parameter jedoch nicht geeignet; es ist in seinem Verhalten nicht konsistent sondern von Parametern der DGL abhängig. In diesem Fall ist  $\varepsilon_k$  sehr variable. Die Veränderung von  $\varepsilon_k$  bei Variation der Schrittweite kann daher als Test für die Qualität eines numerischen Verfahrens verwendet werden und führt auf den Begriff des konsistenten Schemas:

**Definition 18** Ein Differenzschema wird als konsistent bezeichnet, wenn  $\varepsilon_k \rightarrow 0$  für  $\Delta t \rightarrow 0$  für alle Gitterpunkte  $k$  im Integrationsintervall.

### Stabilität

§ 954 Konsistenz und Genauigkeit sind wichtige Eigenschaften: sie beschreiben, ob ein Differenzschema konvergiert oder nicht. Dieser Zusammenhang wird durch das Lax'sche Äquivalenztheorem beschrieben:

**Satz 1** Gegeben ist ein wohl definiertes Anfangswertproblem und ein Differenzschema, dass die Bedingung der Konsistenz erfüllt. Dann ist Stabilität sowohl eine notwendige als auch eine hinreichende Bedingung für Konvergenz.

§ 955 Dabei bedeutet Konvergenz dass die Differenz zwischen der exakten Lösung  $\hat{u}_k(t)$  und der angenäherten Lösung  $u_k$  gegen Null strebt wenn der Diskretisierungsschritt gegen Null strebt:

$$|\hat{u}_k(t) - u_k| \rightarrow 0 \quad \text{wenn} \quad \Delta t \rightarrow 0.$$

### Anwendung dieser Begriffe auf die diskutierten Schemata

§ 956 Die Euler Vorwärts Methode ist, wie bereits bei der Diskretisierung angemerkt, ein Vorwärtsverfahren, da der Wert am Intervallende jeweils aus dem am Intervallanfang ermittelt wird. Wie bei der Diskretisierung bereits angemerkt, ist dies ein Verfahren von erster Ordnung in  $\Delta t$ ; auch der Truncation Error ist von erster Ordnung in  $\Delta t$ .

§ 957 Die Euler Rückwärts Methode ist als Rückwärtsverfahren aus Sicht der Diskretisierung ebenfalls von erster Ordnung in  $\Delta t$ . Das Verfahren ist zweistufig, insbesondere wird im Prädikatorschritt ein Zwischenwert berechnet. Beide Schritte wären einzeln nur von erster Ordnung in  $\Delta t$ . In der Kombination heben sich jedoch der Zwischenwert und die erste Ordnung des  $\Delta t$  heraus, so dass der sich ergebende Truncation Error von zweiter Ordnung in  $\Delta t$  ist.

§ 958 Die von uns betrachtete Variante des Runge–Kutta Verfahrens heißt so, da sie von vierter Ordnung in  $\Delta t$  ist. Sie können dies zeigen, indem Sie den Truncation Error für jeden der Schritte aufstellen und dann die Gleichungen zusammen fügen: die Fehler geringerer Ordnung als vier heben sich heraus.

**Zwischenrechnung 54** Letzteres ist Quälkram, aber machen Sie es trotzdem.

### 10.2.3 Verfahren mit variabler Schrittweite – Idee

§ 959 Die Idee für ein Verfahren mit variabler Schrittweite ist relativ einfach. Beim Euler Verfahren wird die Steigung im Anfangspunkt der Zeitintervalls bestimmt und dann für das gesamte Intervall als konstant angenommen. Dieser letzte Punkt, die Steigung ist im gesamten Zeitintervall konstant (oder stückweise linear zusammen gesetzt wie bei Runge–Kutta) wird in allen numerischen Verfahren gemacht; in den impliziten ebenso wie in den expliziten.

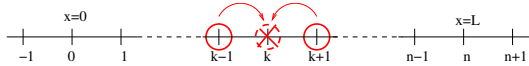


Abbildung 10.1: Zentrale Differenz: der Funktionswert im  $k$ -ten Gitterpunkt bestimmt sich aus den beiden benachbarten Funktionswerten

§ 960 Die Annahme einer konstanten Steigung ist korrekt, falls die Ableitung wirklich eine Konstante ist: dann wäre die Ausgangsfunktion eine Gerade und diese können wir problemlos mit einer im gesamten zu betrachtenden Intervall konstanten Steigung rekonstruieren. Das numerische Verfahren käme also mit einem Zeitschritt aus.

§ 961 Verfahren mit variabler Schrittweite werden als adaptive Verfahren bezeichnet, weil sie versuchen, entsprechend der Änderung der Steigung die Schrittweite anzupassen: schnelle Änderungen der Schrittweite erfordern kleine Zeitschritte, da sonst der durch die Annahme konstanter Schrittweite gemachte Fehler zu groß wird. Ändert sich die Schrittweite mit der Zeit dagegen nur geringfügig, so ist der durch die Annahme konstanter Schrittweite gemachte Fehler auch nur gering.

### 10.2.4 Zentrale Differenz

§ 962 Da sich eine DGL  $n$ ter Ordnung in  $n$  DGLs erster Ordnung zerlegen lässt, ist für die Lösung einer gewöhnlichen Differentialgleichung die Diskretisierung einer ersten Ableitung ausreichend. Bei einer partiellen Differentialgleichung, z.B. der Diffusionsgleichung, treten die erste zeitliche und die zweite räumliche Ableitung auf. Eine Separation von zeitlichem und räumlichem Anteil und anschließende Zerlegung der DGL zweiter Ordnung des räumlichen Teils in zwei DGLs erster Ordnung ist nur für eingeschränkte Geometrien machbar. Außerdem ist die Gleichung nur für  $\delta$ -Injektionen analytisch lösbar – räumlich oder zeitlich ausgedehnte Injektionen führen automatisch auf eine Faltung.

§ 963 Für die numerische Lösung des Diffusionsterms (oder der Wärmeleitungsgleichung) ist daher eine Diskretisierung der zweiten Ableitung wie in (10.3) sinnvoll: bei dieser zentralen Differenz werden die beiden benachbarten Gitterpunkte zur Bestimmung des zentralen Gitterpunktes heran gezogen. Das Verfahren ist damit implizit.

§ 964 Auch physikalisch ist die Verwendung des impliziten Verfahrens unter Berücksichtigung beider Nachbarpunkte befriedigender als ein normalerweise bei der Diskretisierung der zeitlichen Ableitung verwendetes explizites Verfahren: während man bei der Zeit das explizite Verfahren noch anschaulich dadurch begründen kann, dass die Zeit in eine Richtung fortschreitet und damit der folgende Zeitpunkt nur von den vorhergehenden beeinflusst wird, gibt es im Raum in der Regel keine Vorzugsrichtung. Stattdessen wird eine physikalische Größe in einem Raumpunkt von der entsprechenden Größe in den benachbarten Raumpunkten beeinflusst. Im eindimensionalen Fall wird dies genau durch die zentrale Differenz abgebildet.

#### Eindimensionaler Fall

§ 965 Beschränken wir uns auf eine Raumdimension, so ist die zentrale Differenz (10.3), siehe auch Abb. 10.1, zur Diskretisierung ausreichend.

#### Stationärer Fall

§ 966 Im stationären Zustand verschwindet die zeitliche Ableitung und die partielle Differentialgleichung reduziert sich auf eine gewöhnliche Differentialgleichung zweiter Ordnung:

$$\frac{\partial^2 u}{\partial x^2} = 0.$$

Mit (10.3) ergibt sich aus dieser Differentialgleichung die Differenzengleichung

$$\frac{u_{k+1} - 2u_k + u_{k-1}}{(\Delta(x))^2} = -\frac{Q}{\lambda}$$

mit  $Q$  als einer Wärmequelle, die an der Stelle  $k$  auszuwerten ist, und  $\lambda$  als der thermischen Leitfähigkeit. Die Wärmequelle benötigen wir, um die Randbedingungen stationär zu halten – mit der rechten Seite identisch Null würde das numerische Schema eine recht triviale Lösung liefern.

§ 967 Die Differential- und damit auch die Differenzgleichung sind auf einer eindimensionalen Geometrie definiert, z.B. für einen Stab der Länge  $L$ , der für die Diskretisierung in  $N = L/\Delta x$  Intervalle der Länge  $\Delta x$  geteilt wird. Damit gibt es  $N + 1$  Gitterpunkte  $x_0 \dots x_N$ . An den inneren Gitterpunkten lässt sich die Differenzgleichung erfüllen, am ersten und letzten Gitterpunkt,  $x_0$  und  $x_N$  dagegen ist das Differenzschema nicht zu lösen, da die zentrale Differenz dann auf Gitterpunkte außerhalb des Stabes zurück greifen müsste. Der Einfluss der äußeren Gitterpunkte lässt sich durch die Randbedingungen eliminieren.

### Zeitabhängige Lösung

§ 968 Die zeitabhängige Lösung ist etwas weniger trivial. Die Differentialgleichung

$$\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2}$$

wird diskretisiert als

$$\frac{u_k^{l+1} - u_k^l}{\Delta t} = D \frac{u_{k+1}^{l+1} - 2u_k^{l+1} + u_{k-1}^{l+1}}{(\Delta x)^2}. \quad (10.4)$$

Umsortieren nach den einzelnen Gitterpunkten liefert

$$\frac{D}{(\Delta x)^2} u_{k-1}^{l+1} - \left( \frac{1}{\Delta t} + \frac{2D}{(\Delta x)^2} \right) u_k^{l+1} + \frac{D}{(\Delta x)^2} u_{k+1}^{l+1} = -\frac{1}{\Delta t} u_k^l$$

bzw. in Kurzform

$$A u_{k-1}^{l+1} + B u_k^{l+1} + C u_{k+1}^{l+1} = E u_k^l$$

mit den Abkürzungen

$$A = C = \frac{D}{(\Delta x)^2}, \quad C = -\left( \frac{1}{\Delta t} + \frac{2D}{(\Delta x)^2} \right) \quad \text{und} \quad E = -\frac{1}{\Delta t}. \quad (10.5)$$

§ 969 Diese Gleichung gilt für alle inneren Punkte, lediglich an den Randpunkten  $u_0$  und  $u_n$  müssen die Randbedingungen berücksichtigt werden um die (nicht existenten) äußeren Punkte  $u_{-1}$  und  $u_{n+1}$  zu eliminieren. Um einen Fluss über die Umrandung (Neumann'sche Randbedingungen) zu beschreiben, werden die entsprechenden Punkte benötigt; bei Dirichlet'schen Randbedingungen dagegen

§ 970 Das System der Differenzgleichungen lässt sich in Matrixschreibweise zusammen fassen als

$$\begin{bmatrix} B_{\text{in}} & C_{\text{in}} & 0 & 0 & \dots & 0 & & \\ A & B & C & 0 & \dots & 0 & 0 & \\ 0 & A & B & C & \dots & 0 & 0 & \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \\ 0 & 0 & 0 & 0 & \dots & A_{\text{out}} & B_{\text{out}} & \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ c_n \end{bmatrix} = E \begin{bmatrix} c_0^- \\ c_1^- \\ c_2^- \\ \vdots \\ c_n^- \end{bmatrix}$$

mit den Abkürzungen wie in (10.5).



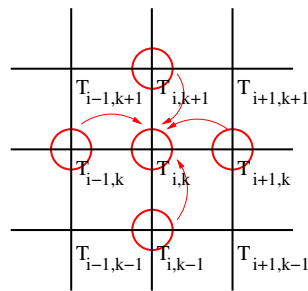


Abbildung 10.2: Zentrale Differenz im 2D

### Zweidimensionaler Fall

§ 971 Im zweidimensionalen Fall hat jeder Gitterpunkt nicht nur die beiden Nachbarn entlang der einen Koordinatenachse sondern auch zwei Nachbarn entlang der anderen Raumdimension. Daher wird die physikalische Größe in einem Gitterpunkt von ihren Werten in den vier benachbarten Gitterpunkten bestimmt, wie in Abb. 10.2 angedeutet. Die Diskretisierung der DGL

$$\frac{\partial u}{\partial t} = D \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

erfolgt wie im eindimensionalen Fall, jedoch wird der räumliche Term durch zwei partielle Ableitungen diskretisiert:

$$\frac{u_{j,k}^{l+1} - u_{j,k}^l}{\Delta t} = D \left( \frac{u_{j,k+1}^{l+1} - 2u_{j,k}^{l+1} + u_{j,k-1}^{l+1}}{(\Delta y)^2} + \frac{u_{j+1,k}^{l+1} - 2u_{j,k}^{l+1} + u_{j-1,k}^{l+1}}{(\Delta x)^2} \right).$$

Normalerweise ist  $\Delta x = \Delta y$ , so dass sich die Gleichung vereinfachen lässt zu

$$\frac{u_{j,k}^{l+1} - u_{j,k}^l}{\Delta t} = D \frac{u_{j+1,k}^{l+1} + u_{j,k+1}^{l+1} - 4u_{j,k}^{l+1} + u_{j,k-1}^{l+1} + u_{j-1,k}^{l+1}}{(\Delta x)^2}. \quad (10.6)$$

Das weitere Verfahren läuft entsprechend: es muss nach dem Gitterpunkt  $u_{j,k}$  aufgelöst werden und die Gleichung in einer geeigneten Kurzform geschrieben werden.

§ 972 Formal hat das Verfahren den Nachteil, dass wir statt des Vektors eine Matrix von Gitterpunkten haben (eben den 2D) und entsprechend müsste die Koeffizientenmatrix durch ein dreidimensionales Gebilde ersetzt werden. Bevor wir hier nach neuen mathematischen Verfahren suchen, können wir die 2D-Version auch auf die Matrixschreibweise der 1D-Version zurück führen. Dazu wird das zweidimensionale räumliche Gitter in einen Vektor überführt: die einzelnen Spalten werden einfach untereinander geschrieben. Oder explizit für ein einfaches  $3 \times 3$ -Gitter:

$$\begin{pmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{pmatrix} = \left( \begin{pmatrix} \vec{u}_{i1} \\ \vec{u}_{i2} \\ \vec{u}_{i3} \end{pmatrix} \right) \Rightarrow \begin{pmatrix} \vec{u}_{i1} \\ \vec{u}_{i2} \\ \vec{u}_{i3} \end{pmatrix} = \begin{pmatrix} u_{11} \\ u_{21} \\ u_{31} \\ u_{12} \\ u_{22} \\ u_{32} \\ u_{13} \\ u_{23} \\ u_{33} \end{pmatrix}.$$

Aus dem  $3 \times 3$ -Gitter wird entsprechend ein  $9 \times 1$  Vektor. Das zugehörige  $3 \times 3 \times 3$ -Gebilde der Koeffizienten wird entsprechend in eine  $9 \times 9$ -Matrix überführt. Diese ist wieder wie im eindimensionalen Fall auf der Haupt- und den beiden benachbarten Nebendiagonalen von Null verschieden (entsprechend dem Transport in einer Dimension) sowie auf zwei weiteren Nebendiagonalen (entsprechend dem Transport in der anderen Dimension). Die Matrix ist also wie im eindimensionalen Fall dünn besetzt (sparse matrix).

§ 973 Damit lässt sich der zweidimensionale Fall völlig analog zum eindimensionalen behandeln. Allerdings wird bei einem  $n \times n$ -Gitter die Koeffizientenmatrix mit einer Dimension  $n^2 \times n^2$  schnell sehr groß – und der Rechenaufwand entsprechend. Ein typisches Hilfsmittel zum Umgang mit großen, dünn besetzten Matrizen sind das Jacobi-Verfahren und der Gauß–Seidel-Algorithmus.

## 10.3 Code-Beispiele aus dem Text

### 10.3.1 Euler-Vorwärts für das Federpendel

§ 974 Eine Codierung des Euler-Vorwärts Verfahrens für die freie Schwingung aus Abschn. 1.1.2 in MATLAB könnte folgende Form annehmen:

```

01 % Harmonischer Oszillator
02 clear;clf
03 %%%%%%% Vereinbarungsteil %%%%%%%%%%%%%%%
04 k=1;m=1; %Systemparameter (Eingabe)
05 omega2=k/m; %Systemparameter (für Rechnung)
06 n=1000;deltt=2*pi/(sqrt(omega2)*n); %Zahl Rechenschritte während einer Periode
07 t=[0:deltt:5*pi/sqrt(omega2)]; %Festlegung Rechendauer (statt Abbruchbedingung)
08 x=zeros(1,length(t));v=x; %Festlegung Ausgabevektoren
09 x(1)=0.; v(1)=1.; %Anfangsbedingungen
10 %%%%%%% Beginn Schleife %%%%%%%%%%%%%%%
11 for i=2:length(t)
12     v(i)=v(i-1)-deltt*omega2*x(i-1); %Lösung zweite DGL
13     x(i)=x(i-1)+deltt*v(i-1); %Lösung erste DGL
14 end
15 %%%%%%% Ende Schleife %%%%%%%%%%%%%%%
16 %%%%%%% Darstellungsteil %%%%%%%%%%%%%%%
17 plot(t,x,'r');hold on;plot(t,v,'g');hold off;
18 xlabel('Zeit','FontSize',14);
19 ylabel('Ort x (rot), v (grün)','FontSize',14);
20 title('Harm.Oszillator (Euler)','FontSize',18);

```

§ 975 In Zeile 4 bis 6 werden die Eingabeparameter (Federkonstante  $k$  und Masse  $m$ , Zeile 4) in den Systemparameter  $\omega_0^2 = k/m$  bzw. die charakteristische Zeit des Systems  $\tau = 2\pi/\omega$  umgewandelt (Zeile 5) und daraus die für eine feste Zahl  $n$  von Rechenschritten pro Periode erforderliche Zeitauflösung `deltt` bestimmt (Zeile 6). In Zeile 7 wird der zugehörige Vektor der Zeitachse erzeugt derart, dass 2.5 Schwingungen in das Simulationsintervall passen. Durch die Festlegung der Zeitachse ist gleichzeitig auch die Simulation begrenzt, so dass keine Abbruchbedingung erforderlich ist.<sup>2</sup> In Zeile 8 werden die Ausgabevektoren für Ort  $x$  und Geschwindigkeit  $v$  als Nullvektoren vordefiniert; sonst rechnet MATLAB sich in der Schleife tot. Zeile 9 enthält die Eingabe der Anfangsbedingungen. Diese werden an die erste Stelle der Ausgabevektoren eingetragen.

§ 976 Die eigentliche Lösung der DGL erfolgt in der Schleife von Zeile 10 bis 15: in Zeile 12 wird (1.5) gelöst, in Zeile 13 (1.5). In Zeile 12 wird dabei die Beschleunigung bei der Lösung der Gleichung mit bestimmt, entsprechend dem linken Flussdiagramm in Abb. 1.3. Für den rechten Teil von Abb. 1.3 müsste Zeile 12 in zwei Zeilen aufgebrochen werden:

```

12a acc = -omega2*x(i-1)
12b v(i)=v(i-1)+deltt*acc

```

<sup>2</sup>Alternativ hätte man die Skalierung von Zeit und Zeitschritt auch derart wählen können, das in Einheiten der Zeitkonstante des Systems gearbeitet wird – der wesentliche Unterschied ist die Skalierung der Zeitachse: in dieser Form ist die Zeit in absoluten Einheiten gegeben, so dass für unterschiedliche Kombinationen von  $k$  und  $m$  jeweils unterschiedliche Zeitfenster betrachtet werden. Im anderen Fall ist das Zeitfenster in Einheiten der Zeitkonstante fix, d.h. die Darstellung entspricht der mit dimensionslosen Variablen.

Die folgenden Zeilen 16 bis 20 dienen der Darstellung der Ergebnisse (Ausgabe), siehe auch Abb. 1.4.

§ 977 Für die erzwungene Schwingung können wir den obigen MATLAB-Code nahezu unverändert übernehmen. Dabei gehen wir von der Variante mit der aufgespaltenen Zeile 12 aus (siehe § 976). Hier muss Zeile 12a erweitert werden zu

```
12a acc = -omega2*x(i-1) - beta*v(i-1) + fext(i-1)
```

Zur Ausführung dieses Codes muss die Dämpfungskonstante `beta` vereinbart werden (sinnvollerweise zusammen mit den anderen Systemparametern in Zeile 4). Außerdem muss die äußere Kraft `fext` definiert werden. Dies sollte nach der Definition der Zeitachse in Zeile 7 erfolgen, z.B. als Zeile 7a mit

```
07a fext=fampl*cos(Omega*t)
```

wobei die Amplitude, bzw. genauer die Amplitude pro Masse `fampl`, der anregenden Kraft und die Anregungsfrequenz `Omega` ebenfalls in Zeile 4 vereinbart werden können.

## 10.4 Nullstellenbestimmung: Newton–Raphson

§ 978 Gesucht ist die Nullstelle  $f(x) = 0$  der Funktion  $f(x)$ . Zwar haben wir eine grobe Vorstellung, wo die Nullstelle liegen muss; wir haben auf Grund der Struktur der Funktion jedoch keine Möglichkeit, eine exakte Lösung zu finden. Eine einfache Möglichkeit ist Intervallschachtelung; wir bestimmen Funktionswerte für Argumente größer und kleiner als unser Verdachtswert. Wechselt das Vorzeichen der Funktionswerte, so machen wir das Intervall kleiner, überprüfen den Vorzeichenwechsel und so weiter, bis wir die Nullstelle mit gewünschter Genauigkeit eingeschachtelt haben. Das Verfahren ist anschaulich aber aufwendig, da für jeden Schritt der Funktionswert an zwei Stellen (den beiden Intervallgrenzen) bestimmt werden muss. Außerdem scheitert das Verfahren bereits bei einer so einfachen Funktion wie  $x^2$ : diese hat zwar bei  $x = 0$  eine Nullstelle, es findet jedoch kein Vorzeichenwechsel statt. Intervallschachtelung versagt also, wenn die Nullstelle gleichzeitig Extremum der Funktion ist.

§ 979 Allgemeiner ist die Newton–Raphson Methode. Sie basiert auf einer Iteration der Form

$$a_{n+1} = a_n - \frac{f(a_n)}{f'(a_n)} \quad (10.7)$$

mit  $f'$  als der Ableitung von  $f$ . Der letzte Term ist das Verhältnis aus Funktionswert und Ableitung an der Stelle  $a_n$ , d.h. er gibt an, wie weit der Schnittpunkt zwischen der Geraden durch  $(a_n, f(a_n))$  mit Steigung  $f'(a_n)$  und der Abszisse von  $a_n$  entfernt ist, vgl. Abb. 10.3. Auf diese Weise entsteht eine Folge von Werten  $a_1, a_2, a_3, \dots$ , die sich immer weiter der gesuchten Nullstelle annähert.<sup>3</sup>

§ 980 Für einige Funktionen lässt sich die Folge der  $a_i$  explizit angeben. So ergibt sich für das Polynom  $(x - 1)^2 = 0$  mit der Ableitung  $2(x - 1)$  aus dem Bildungsgesetz der Newton–Raphson Methode eine Folge, deren Glieder nach der Rekursionsformel

$$a_{n+1} = a_n - \frac{(a_n - 1)^2}{2(a_n - 1)} = \frac{a_n + 1}{2}$$

bestimmt werden. Mit einem geratenen Anfangswert  $a_1 = \frac{1}{2}$  ergibt sich die Folge

$$\frac{1}{2}, \frac{3}{4}, \frac{7}{8}, \frac{15}{16}, \dots, \frac{2^n - 1}{2^n}, \dots$$

Diese Sequenz entwickelt sich in Richtung auf den erwarteten Wert  $x = 1$ . Daher kann man sagen, dass diese Folge für  $n \rightarrow \infty$  gegen 1 konvergiert. Auch für alle anderen Anfangswerte  $a_1$  konvergiert diese Folge gegen 1, es kann nur etwas länger dauern.

<sup>3</sup>Das gilt in unserem Beispiel, da die Funktion glatt, d.h. stetig und differenzierbar ist. Damit haben wir auch die Voraussetzung für die Anwendbarkeit der Methode gegeben.

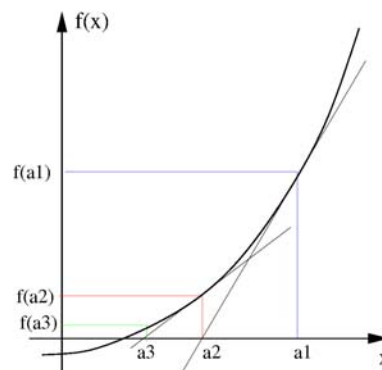


Abbildung 10.3: Geometrische Konstruktion zur Newton-Raphson Methode

**Zwischenrechnung 55** Den Anfangswert für die Newton-Raphson Methode haben wir willkürlich auf  $1/2$  gesetzt. Wie ändert sich das Ergebnis, wenn wir stattdessen z.B.  $a_1 = -1/2$  oder  $a_1 = 10$  verwenden. Erklären Sie das Ergebnis!

## 10.5 Etwas (historischer) Hintergrund

### 10.5.1 Block- und Flussdiagramme

§ 981 Bereits in der Einführung in Abschn. 1.1 sind wir den Begriffen Block- und Flussdiagramm begegnet. Eine Unterscheidung zwischen den beiden Begriffen ist nicht unbedingt erforderlich: zwar wird in der Informatik klassischerweise der Begriff Flussdiagramm verwendet, in der Systemdynamik werden jedoch die Begriffe Fluss- und Blockdiagramm gleichermaßen verwendet.

§ 982 So gibt Wikipedia (<http://de.wikipedia.org/wiki/Flussdiagramm>) die folgende Erläuterung:

Das Wort Flussdiagramm bezeichnet

- eine in der Informatik verwendete Darstellungsform von Ablaufplänen in Computerprogrammen (<http://de.wikipedia.org/wiki/Programmablaufplan>)
- eine in der Systemdynamik ([http://de.wikipedia.org/wiki/System\\_Dynamics](http://de.wikipedia.org/wiki/System_Dynamics)) verwendete Systematik zur Darstellung von kausalen Zusammenhängen
- ein Sankey-Diagramm (<http://de.wikipedia.org/wiki/Sankey-Diagramm>) wird auch als Massenflussdiagramm bezeichnet.

§ 983 Im englischen Sprachraum wird der Begriff flowchart ebenfalls für beide Anwendungen genutzt – dort bemüht sich Wikipedia (<http://en.wikipedia.org/wiki/Flowchart>) nicht einmal um obige sprachliche Feinheiten sondern gibt die Definition: ‘A flowchart is a schematic representation of an algorithm or a process.’

### 10.5.2 Flussdiagramme in der Informatik

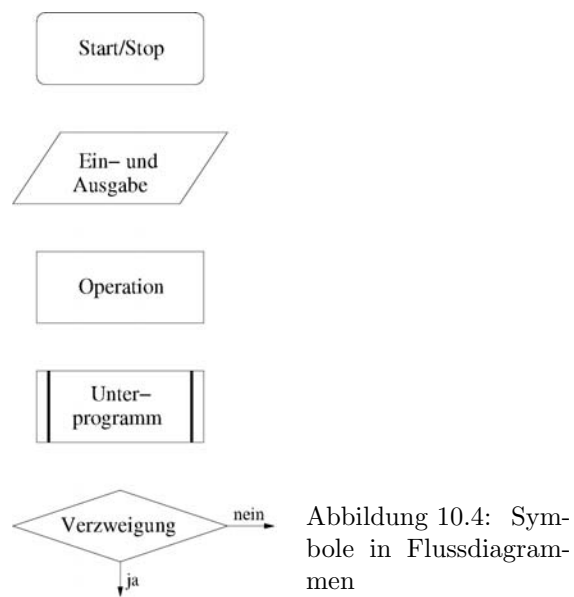
§ 984 Die wichtigsten, in Flussdiagrammen verwendeten Symbole, sind in Abb. 10.4 zusammen gefasst.

## 10.6 Lösungen ausgewählter Aufgaben

Aufgabe 4:

$$\frac{16}{\pi^2} \left( \cos \frac{\pi x}{4} + \frac{1}{3^2} \cos \frac{3\pi x}{4} + \frac{1}{5^2} \cos \frac{5\pi x}{4} \dots \right)$$

Aufgabe 6  $x(t) = -7e^t + 4e^{2t} + 4te^{2t}$



Aufgabe 7  $x = 5e^{-t} + 3e^{4t}$ ,  $y = 5e^{-t} - 2e^{4t}$

Aufgabe 8  $y(x) = W_0/(24E)(l^3x - 2lx^3 + x^4) = W_0/(24E)x(l-x)(l^2 + lx - x^2)$

# Abbildungsverzeichnis

1.1	Lösungen gedämpfte Schwingung . . . . .	15
1.2	Erzwungene Schwingung . . . . .	17
1.3	Flussdiagramm freie Schwingung . . . . .	19
1.4	Freie Schwingung (numerische Lösungen) . . . . .	19
1.5	Erzwungene Schwingung (numerische Lösungen) . . . . .	20
1.6	Freie Schwingung in SIMULINK . . . . .	22
1.7	Freie Schwingung in SIMULINK II . . . . .	23
1.8	Erzwungene Schwingung in SIMULINK . . . . .	24
1.9	Erzwungene Schwingung in SIMULINK . . . . .	24
1.10	Erzwungene Schwingung mit <b>Transfer Function</b> . . . . .	25
1.11	Erzwungene Schwingung mit State Space . . . . .	26
1.12	Fadenpendel nichtlinear (große Winkel) . . . . .	29
1.13	Fadenpendel im Phasenraum . . . . .	30
1.14	Vorhersageprobleme: Übergang Schwingung – Rotation . . . . .	30
1.15	DGL zweiter Ordnung . . . . .	31
1.16	Quiver-Plot zum Modell in Abb. 1.15 . . . . .	32
1.17	Ein Seil gleitet reibungsfrei über eine Tischkante in SIMULINK . . . . .	34
1.18	Gekoppelte Federpendel . . . . .	35
1.19	Gekoppelte Pendel (numerische Lösung) . . . . .	37
1.20	Gekoppelte Pendel in SIMULINK . . . . .	37
1.21	Gekoppelte Pendel im Phasenraum . . . . .	38
1.22	Elektrischer Schwingkreis . . . . .	40
1.23	Spezialisierung verschiedener Regionen der Großhirnrinde . . . . .	41
1.24	Purkinje-Zellen . . . . .	42
1.25	Barnsley Farn, Muschel mit Sierpinski Dreiecken und Koch'sche Schneeflocke . . . . .	43
1.26	Belousov-Zhabotinsky-Reaktion . . . . .	44
1.27	Selbstorganisation . . . . .	45
1.28	Humunculus . . . . .	46
1.29	Pendel: schwingende U-Röhre, Feder-Faden-Pendel, Schwingungstaucher, doppelschiefe Ebene . . . . .	50
2.1	Lego® und Anspruch . . . . .	53
2.2	Einfaches Modell in SIMULINK: Tiefpass . . . . .	54
2.3	Bandpass in SIMULINK . . . . .	55
2.4	Bandpass: Signalverlauf für verschiedenen Eingangsfrequenzen . . . . .	55
2.5	Gruppen von Blöcken in SIMULINK . . . . .	56
2.6	Kontinuierliche Signale (SIMULINK) . . . . .	58
2.7	Parameter des Integrators . . . . .	59
2.8	Integration einer Rampe . . . . .	60

2.9	Blöcke für mathematische Operationen in SIMULINK . . . . .	62
2.10	Blöcke zur Signaldarstellung in SIMULINK . . . . .	64
2.11	Scope mit mehr als einem Eingang . . . . .	64
2.12	Blöcke zur Darstellung externer Antriebe in SIMULINK . . . . .	65
2.13	Simulationskontrolle in SIMULINK . . . . .	66
2.14	Parameter Gain . . . . .	68
2.15	Parameter Integrator . . . . .	69
2.16	Gedämpfte Schwingung . . . . .	70
2.17	Freie Schwingung in SIMULINK: Überprüfung der Lösung . . . . .	71
2.18	Kran als Fadenpendel . . . . .	72
2.19	SIMULINK Modell zum Kran . . . . .	74
2.20	Auslenkung des Containers . . . . .	75
2.21	Doppelpendel im Phasenraum . . . . .	77
2.22	Lissajous Figur . . . . .	78
2.23	Gekoppelte Federpendel und Untersysteme: Geometrie . . . . .	79
2.24	Subsystem für eine einzelne Masse . . . . .	79
2.25	Gekoppelte Pendel und Untersysteme . . . . .	80
2.26	Simulationsergebnis Dreierpendel . . . . .	80
2.27	Editor für ein maskiertes Subsystem 1 . . . . .	81
2.28	Editor für ein maskiertes Subsystem 2 . . . . .	82
2.29	Editor für ein maskiertes Subsystem 4 . . . . .	83
2.30	Editor für ein maskiertes Subsystem 3 . . . . .	83
2.31	Editor für ein maskiertes Subsystem 5 . . . . .	84
2.32	System aus Abb. 2.23 mit maskierten Untersystemen . . . . .	85
2.33	Bedingtes Untersystem (Enabled Subsystem) . . . . .	85
2.34	Bedingtes Untersystem II . . . . .	86
2.35	Triggered Subsystem . . . . .	86
2.36	Raketenstart: Main . . . . .	88
2.37	Raketenstart: Subsystem . . . . .	89
2.38	Raketenstart: Ergebnisse . . . . .	90
2.39	Schrittweite ODE 45 . . . . .	92
2.40	Simulationskontrolle . . . . .	94
2.41	Schrittweiten verschiedene ODEs . . . . .	97
2.42	Springender Ball in SIMULINK . . . . .	99
2.43	Modell eines elastischen Balls . . . . .	99
2.44	Elastischer Ball in SIMULINK . . . . .	100
2.45	Zener-Diode . . . . .	101
2.46	Schmitt Trigger . . . . .	101
2.47	Simulationskontrolle: Im- und Exporte . . . . .	102
2.48	Simulationskontrolle: Optimierung . . . . .	103
2.49	Simulationskontrolle: Diagnostik . . . . .	104
2.50	DGL erster Ordnung mit algebraischer Schleife . . . . .	104
2.51	Wie Abb. 2.6.6 aber ohne algebraische Schleife . . . . .	105
2.52	Memory Block zum Aufbrechen einer algebraischen Schleife . . . . .	106
2.53	Bewegung zweier Wagen . . . . .	106
2.54	Algebraische Schleife . . . . .	107
2.55	Wie Abb. 2.54 aber mit Memory Block . . . . .	107
3.1	Fourier Reihe . . . . .	115
3.2	Periodische Funktion . . . . .	116
3.3	Basisfunktionen der Fourier Transformation . . . . .	116
3.4	$f(t) = \cos^2(t)$ als Beispiel für Fourier Zerlegung . . . . .	117
3.5	Fourier Reihe für $x^2$ . . . . .	120
3.6	Langsam und schnell veränderliche Funktion und ihr Fourier Spektrum . . . . .	122

3.7	Fourier Reihe eines Rechtecks . . . . .	122
3.8	Gibbs'sches Phänomen . . . . .	123
3.9	$\sin(x)/x$ . . . . .	123
3.10	Lorentz Funktion als Fourier Transformiert . . . . .	124
3.11	Diskrete Fourier Transformation Winkelfunktionen . . . . .	130
3.12	Fourierkoeffizienten Winkelfunktionen . . . . .	131
3.13	Sonnenfleckenzyklus . . . . .	133
3.14	Rechteck und FFT . . . . .	135
3.15	Reduziertes Rechteck und FFT . . . . .	135
3.16	Echtes Rechteck und FFT . . . . .	135
3.17	Länge Datensatz und Genauigkeit Fourier . . . . .	136
3.18	Verrauschtes Rechteck und FFT . . . . .	138
3.19	Multiplikation im Frequenzbereich statt Faltungsintegral im Zeitbereich . . . . .	139
3.20	Leistungsdichtespektrum . . . . .	140
3.21	Beispiel Leistungsdichtespektrum . . . . .	140
4.1	RC-Netzwerke . . . . .	151
4.2	Harmonischer Oszillator . . . . .	156
4.3	Tiefpass: Ein- und Ausgangssignal für unterschiedliche Frequenzen und Signalformen; Bode-Diagramm . . . . .	158
4.4	Systeme in Serie . . . . .	164
4.5	Parallele Systeme . . . . .	165
4.6	Rückkopplung und offener Regelkreis . . . . .	165
4.7	Standardsregelkreis . . . . .	165
5.1	Analoge Repräsentation eines digitalen Signals . . . . .	171
5.2	Signalkette . . . . .	172
5.3	Unterabtastung . . . . .	172
5.4	Typische Filter . . . . .	173
5.5	Hoch- und Tiefpass . . . . .	176
5.6	Filter zweiter Ordnung . . . . .	177
5.7	Serienschwingkreis in LTSpice . . . . .	178
5.8	Butterworth Filter (Schaltung) . . . . .	179
5.9	Butterworth Filter (Übertragungsfunktion) . . . . .	179
5.10	Tschebyscheff Polynome . . . . .	180
5.11	Tschebyscheff Filter . . . . .	181
5.12	Gleitendes Mittel als Tiefpass . . . . .	184
5.13	Gleitendes Mittel: Zeit und Support . . . . .	185
5.14	Gleitendes Mittel: Ramdpunkte . . . . .	186
5.15	Gleitendes Mittel in SIMULINK . . . . .	187
5.16	Blockdiagramm digitaler Filter . . . . .	188
5.17	Blockdiagramm IIR Filter . . . . .	189
5.18	Butterworth Filter . . . . .	190
6.1	Feigenbaum Attraktor . . . . .	197
6.2	Apfelmännchen . . . . .	197
6.3	Küstenlänge . . . . .	198
6.4	Schneeflocken . . . . .	199
6.5	Attraktor-Bereiche beim Magnetpendel . . . . .	202
6.6	Logistische Gleichung . . . . .	203
6.7	Lorenz-Attraktor . . . . .	204
6.8	SIMULINK-Modell zur Darstellung des Lorenz-Attraktors . . . . .	205
6.9	Unterprogramm aus Abb. 6.8 . . . . .	205
6.10	3D-Oszilloskop als S-Function . . . . .	206
6.11	Fadenpendel mit Feder . . . . .	207



6.12	Feder-Fadenpendel: Typen der Bewegung	208
6.13	Feder-Fadenpendel in SIMULINK	208
6.14	Feder-Fadenpendel: Typen der Bewegung II	209
6.15	Poincaré Portrait Feder-Fadenpendel I	209
6.16	Poincaré Portrait Feder-Fadenpendel II	210
6.17	Fadenpendel an einer Rolle	210
6.18	Fadenpendel mit Rolle: Poincaré Schnitt	211
6.19	Mechanisches Modell zum Duffing-Oszillator	212
6.20	Duffing-Oszillator in SIMULINK	212
6.21	Bistabiler Duffing Oszillator	213
6.22	Duffing Oszillator: Sensitivität gegen Anfangsbedingungen	213
6.23	Parameter-Raum für den Ueda-Oszillator	214
6.24	Lineare Anordnung gekoppelter Pendel	214
6.25	Weg-Zeit- und Geschwindigkeits-Zeit-Diagramme für 10 gekoppelte Pendel	215
6.26	Darstellung Phasenraum	216
6.27	Darstellung Phasenraum	217
6.28	Kinetische Energie in der Pendelkette	217
6.29	Doppelpendel: Orte und Phasendifferenz	218
6.30	Vereinfachtes Toda Modell	219
6.31	Vereinfachtes Toda Modell II	220
7.1	Diffusions-Reaktions-Modell schematisch	227
7.2	Relative Längenskalen bei Turing Mustern	228
7.3	Muster auf Tierhäuten	229
7.4	Central Pattern Generator	232
7.5	Hodgkins-Huxley in SIMULINK	233
7.6	Aktionspotential eines Hudgkins-Huxley Neurons	234
8.1	Global Environment and Society (GES) Model	247
8.2	Impulsantwort Klimasystem	248
8.3	Emissionsszenarios	249
8.4	Sensitivität Kosten	250
8.5	Discount und Vermeidungskosten	251
8.6	Schadenskostenfunktion und Discount	252
8.7	Erweiterts GES-Modell	254
8.8	GES im Mehrspielerszeanrio	255
8.9	Nash-Gleichgewicht	256
8.10	Nash Gleichgewicht und einzelner Vermeider	258
8.11	Konsument-Produzent fossiler Brennstoffe	259
8.12	Gefangenendilemma	261
9.1	The Prisoner	266
9.2	Russel's Soliton modern	268
9.3	Soliton aus Dispersion und Nichtlinearität	269
9.4	Tsunami 2004	271
9.5	Bright and Dark Soliton	273
9.6	Moderne Variante Huygen's synchronisierte Uhren	275
9.7	Arnold's Tongue und Devil's Staircase	276
9.8	Van Pol's Gleichung	277
10.1	Zentrale Differenz	287
10.2	Zentrale Differenz im 2D	289
10.3	Newton-Raphson Methode	292
10.4	Symbole in Flussdiagrammen	293

# Tabellenverzeichnis

3.1	Fourier Transformierte wichtiger Funktionen . . . . .	125
3.2	Laplace Transformierte wichtiger Funktionen . . . . .	142

# Literaturverzeichnis

- [1] H.L. Anderson: Scientific Uses of the MANIAC, *J. Stat. Phys.* **43** Nos. 5/6, 1986 221
- [2] R. Axelrod: *The evolution of cooperation*, New York, Basic Books, 1984 240, 242, 262
- [3] M. Bacharach: *Beyond individual choice – teams and frames in game theory*, Princeton University Press, 2006 243, 262
- [4] M. Barnsley: *Fractals everywhere*, Academic Press, 1985 42
- [5] B.P. Belousov: A periodic reaction and its mechanism (in russ.), Compilation of Abstracts on Radiation Medicine, 147:145, 1959. Auf deutsch: B. P. Belousov: Eine periodische Reaktion und ihr Mechanismus. In: L. Kuhnert , U. Niedersen (Hrsg.): Selbstorganisation chemischer Strukturen. Verlag Harri Klein, Frankfurt/Main 2001, S. 73-82 44
- [6] G.P. Berman and F.M. Izrailev: Ther Fermi–Pasta–Ulam problem: 50 years of progress, 2008, <http://arxiv.org/abs/nlin/0411062v3> 223
- [7] K. Binmore: *Fun and Games: a text von game theory*, D.C. Heath, Lexington, MA 262
- [8] I.I. Blekham: *Synchronization in science and technology*, ASME Press, New York, 1988 278
- [9] Bronstein 122, 125, 142, 145
- [10] T. Butz: *Fouriertransformation für Fußgänger*, Teubner, 1998 123, 132
- [11] S. Campbell, J.-P. Chancelier, and R. Nikoukhah: *Modeling and Simulation in Scilab/Scicos*, Springer, 2006 49
- [12] D. Chalmers: *The conscious mind: in search of a fundamental theory*, Oxford University Press, 1996 267
- [13] T. Chang and D. Chang: Enhancing Learning Experience with Dynamic Simulation, <http://www.ni.com/pdf/academic/us/journals/lv02.23.pdf>
- [14] P. Christen: *was sind quaternionische Fraktale?*, Matura-Arbeit, 2006, <http://pascal.christenhome.ch/doku/Matura.v2.pdf> 224
- [15] J. W. Cooley und J. W. Tukey: An algorithm for the machine calculation of complex Fourier series, *Math. Comput.* **19**, 297–301, 1965 132
- [16] A.P. Cracknell und L.W.B. Hayes: *Introduction to remote sensing*, Taylor und Francis, 1991 193
- [17] R.V. Craster and R. Sassi: Spectral algorithms for reaction–diffusion equations, Technical report, Note de Polo, No. 99, Uni Mailand, 2006, [http://www.crema.unimi.it/Biblioteca/Note\\_pdf/124.pdf](http://www.crema.unimi.it/Biblioteca/Note_pdf/124.pdf), code unter <http://www.dti.unimi.it/~sassi/NotePolo99.Code.zip> 232, 235
- [18] J.B. Dabney and T.L. Harman: *Mastering SimuLink*, Pearson, 2004 49, 106, 108
- [19] T. Dauxois: Fermi, Pasta, Ulam and a mysterious lady, 2008, <http://aps.arxiv.org/abs/0801.1590v1> 223
- [20] T. Dauxois, M. Peyrard and S. Ruffo: The Ferm–Pasta–Ulam numerical experiment: history and pedagogical perspectives, 2005, <http://arxiv.org/abs/nlin/0501053v2> 223

- [21] A.K. Dixit and B.J. Nalebuff: *Spieltheorie für Einsteiger*, Schäffer-Poeschel, Darmstadt, 1995
- [22] G. Doetsch: *Anleitung zum praktischen Gebrauch der Laplace Transformation*, Oldenbourg, 1956 145
- [23] C.L. Epstein: *Introduction to the Mathematics of Medical Imaging*, Cambridge University Press, 2007 192
- [24] T. Eggert, E. Schneider and S. Glasauer: *Introduction to Computational Neuroscience*, bccn, <http://www.nefo.med.uni-muenchen.de/~vorlesung/>, insbesondere <http://www.nefo.med.uni-muenchen.de/~vorlesung/neuron.pdf> 230, 233, 234, 235
- [25] M.V. Evans, S.M. Dowd, E.M. Kenyon, M.F. Hughes, and H.A. El-Masri: A physiologically based pharmacokinetic model for intravenous and ingested dimethylarsenic acid in mice, *Toxic. Sci.* 104, (2)250–260, 2008, doi:10.1093/toxsci/kfn080, <http://toxsci.oxfordjournals.org/cgi/reprint/104/2/250.pdf> sowie zusätzliches Material unter <http://toxsci.oxfordjournals.org/cgi/data/kfn080/DC1/1> 109
- [26] M.J. Feigenbaum: *Universality in complex discrete dynamical systems*, Los Alamos Theoretical Division Annual Report, 98-102, 1977 42, 196, 223
- [27] M.J. Feigenbaum: Some characterizations of strange sets, *J. Stat. Phys.* 46, 919–924, 1987 42, 196, 223
- [28] E. Fermi, J.R. Pasta and S.M. Ulam: A study of nonlinear problems, Los Alamos Sci. Lab. rep. LA-1940, 1955, <http://www.physics.utah.edu/~detar/physcs6720/handouts/fpu/FermiCollectedPapers1965.pdf> 219
- [29] Y. Fisher (Hrsg.): *Fractal image compression*, Springer, 1996 43, 201
- [30] G. Gallavotti (ed): *The Fermi–Pasta–Ulam Problem*, Springer Lecture Notes in Physics 728, Springer, 2008 223
- [31] M.R. Garvie: Finite difference schemes for reaction-diffusion equations modeling predator–prey interactions in MATLAB, *Bull. Math. Biol.* 69, 931–956, 2007, DOI 10.1007/s11538-006-9062-3, <http://www.uoguelph.ca/~mgarvie/paper7.pdf>, Code unter [http://www.uoguelph.ca/~mgarvie/PredPrey\\_files/pred\\_preym\\_sim.html](http://www.uoguelph.ca/~mgarvie/PredPrey_files/pred_preym_sim.html)
- [32] L. Glass: Synchronization and rhythmic processes in physiology, *Nature*, 410, 277–284, 2001 278
- [33] H. Haken: *Brain Dynamics*, Springer, 2008 230, 235
- [34] H. Handels: *Medizinische Bildverarbeitung*, Teubner, 2000 192, 193
- [35] S.P. Hargreaves Heap and Y. Varoufakis: *Game theory – a critical text*, Routledge, 2004 262
- [36] A. Hasagawa and F. Tappert: Transmission of stationary nonlinear optical pulses in dispersive dielectric fibers, I. anomalous dispersion, II. normal dispersion, *Appl. Phys. Lett.*, 23, 142–144, 171–172, 1973 273
- [37] Hasselmann, K. and Hasselmann, S.: Multi-Actor Optimization of Greenhouse-Gas Emission Paths Using Coupled Integral Climate Response and Economic Models’, in *Proc. Potsdam Symp. ‘Earth System Analysis: Integrating Science for Sustainability’*, 1994, 1996 238, 259
- [38] Hasselmann, K., Hasselmann, S., Giering, R., Ocana, V., and von Storch, H.: *Optimization of CO<sub>2</sub> Emissions Using Coupled Integral Climate Response and Simplified Cost Models. A Sensitivity Study*, MPI Rep. 192, also IIASA workshop series (in press), 1996 238, 255, 256, 258, 259
- [39] K. Hasselmann, S. Hasselmann, R. Giering, V. Ocana and H. von Storch: Sensitivity study of optimal CO<sub>2</sub> emission paths using a simplified structural integrated assessment model (SIAM). *Clim. Change* 37(2), 345–386, 1997 238, 247, 248, 249, 250, 251, 252, 253
- [40] H. Häberle: *Informationstheoretische Grundlagen der Digitaltechnik*, VDI-Bildungswerk, 1966 171
- [41] V. Henri: *Lois Generales de l’Action des Distases*, Hermann, Paris, 1903 226
- [42] E. Herter und W. Lörcher: *Nachrichtentechnik: Übertragung - Vermittlung - Verarbeitung*, Hanser, München, 1992 171

- [43] A.L. Hodgkin and A.F. Huxley: A quantitative description of membrane current and its application to conduction and excitation in nerve, *J. Physiology* **117**, 500–544, 1952 230
- [44] J. Hoffmann and U. Brunner: *MatLab & Tools*, Addison-Wesley, 2002 49, 99, 100
- [45] D.R. Hofstadter: *Gödel, Escher, Bach – an eternal golden braid*, Vintage, 1980 239
- [46] D.R. Hofstadter: *Metamagicum – fragen nach der Essenz von Geist und Struktur*, Klett-Cotta, 1988 239
- [47] D.r. Hofstadter: *Fluid concepts and creative analogies*, Basic Books, 1995 239
- [48] J.H. Holland: *Emergence – from chaos to order*, Oxford University Press, Oxford, 1998 267
- [49] S.L. Hooper: *Central Pattern Generators*, Embryonic ELS <http://crab-lab.zool.ohiou.edu/hooper/cpg.pdf>, 2000 233
- [50] M. Horn und N. Dourdoumas: *Regelungstechnik*, Pearson, 2004 168
- [51] M.E. Hosea and L.F. Shampine: Analysis and implementation of TR-BDF2, *Appl. Num. Math.* **20**, (1-2)21–37, 1996 95
- [52] Ying-Yu Huang, Oliver Rinner, Patrik Hedinger, Shih-Chii Liu, and Stephan C.F. Neuhauss: Oculomotor Instabilities in Zebrafish Mutant belladonna: A Behavioral Model for Congenital Nystagmus Caused by Axonal Misrouting, *J. Neurosci.* **26** (39):9873–9880, 2006, [https://www.zora.uzh.ch/356/1/Huang\\_JN\\_2006.pdf](https://www.zora.uzh.ch/356/1/Huang_JN_2006.pdf)
- [53] B.R. Hunt, R.L. Lipsman, and J.M. Rosenberg: *A guide to MatLab for beginners and experienced users*, Cambridge University Press, 2<sup>nd</sup>ed, 2006 49, 72, 74, 75, 109
- [54] K. Hutter (Hrsg.): *Dynamik umweltrelevanter Systeme*, Springer, 1991
- [55] Ch. Huygens: *Horologium Oscillatorium*, Muguet, Paris, 1673; englisch as *The pendulum clock*, Iowa State University Press, Ames, 1986 274
- [56] D.M. Imboden and S. Koch: *Systemanalyse – Einführung in die mathematische Modellierung natürlicher Systeme*, Springer, 2003
- [57] P. Imkeller und J.-S. von Storch (eds.): *Stochastic climate models*, Birkhäuser, 2001
- [58] IPCC (Intergovernmental Panel on Climate Change): *Climate Change 2007* mit allen seinen Teilbänden (und den älteren Bänden) unter <http://www.ipcc.ch/ipccreports/assessments-reports.htm> 247, 279
- [59] M. Issing: *Messung der elektrischen Aktivität biologischer Zellen mit Feld-Effekt-Transistoren*, TU Berlin, 2000, unter <http://mikro.ee.tu-berlin.de/forschung/bioelectronic/myobip/download.html> 234
- [60] B. Jähne: *Digitale Bildverarbeitung*, Springer, 2002 192
- [61] R.S. Johnson: Water waves and Korteweg–de Vries equations, *J. Fluid Mech.* **97**, 701–719, 1980 274
- [62] S. Johnson: *emergence*, Penguin Books, 2002 13, 45, 47, 267
- [63] M.-B. Kallenrode: *Rechenmethoden der Physik*, Springer, 2. Aufl., 2005 17, 95
- [64] M.-B. Kallenrode: *Physik der Atmosphäre*, Vorlesungsskript, Uni Osnabrück, Mai 2006, <http://www.sotere.uni-osnabrueck.de/Lehre/skript/atmo-master.pdf> 246
- [65] M.-B. Kallenrode: *Modeling transport*, Vorlesungsskript, Uni Osnabrück, Januar 2007, <http://www.sotere.uni-osnabrueck.de/Lehre/skript/transport-master.pdf> 17, 28, 38
- [66] M.-B. Kallenrode: *Mathematik für Physiker (2)*, Vorlesungsskript, Uni Osnabrück, November 2007, [http://www.sotere.uni-osnabrueck.de/Lehre/skript/mathemaster\\_small.pdf](http://www.sotere.uni-osnabrueck.de/Lehre/skript/mathemaster_small.pdf) 17, 95
- [67] E.R. Kandel, J.H. Schwartz and T.M. Jessell: *Principles of neural science*, 2<sup>nd</sup> ed., Prentice Hall, 1991 46
- [68] M. Kauke: *Kooperative Intelligenz*, Spektrum Akademischer Verlag, 1998
- [69] J. Kennedy and R.C. Eberhart: *Swarm intelligence*, Morgan Kaufmann Publ. 2001
- [70] J. Kim: *Mind in a physical world*, MIT Press, Cambridge, MA, 1999 267
- [71] H. Knözinger and M. Scheffler: Expedition zwischen Physik und Chemie, *Phys. J.* **6**, (12)27, 2007 230

- [72] H. von Koch: Sur une courbe continue sans tangente, obtenue par une construction géométrique élémentaire, *Arkiv för Matematik* **1**, 681-704, 1904 **42**, **200**
- [73] P. Kollock: An Eye for an Eye Leaves Everyone Blind: Cooperation and Accounting Systems, *Am. Sociol. Rev.* **58** (6)768–786, 1993 **260**
- [74] R. Kopelman: fractal reaction kinetics, *Science* **241**, 1620–1626, 1988 **226**
- [75] H.J. Korsch, H.-J. Jodl and T. Hartmann: *Chaos*, Springer, 2008 **204**, **214**, **222**
- [76] D. Kreß: *Theoretische Grundlagen der Signal- und Informationsübertragung*, Vieweg, Braunschweig, 1977 **171**
- [77] Y. Kuramoto: *Chemical oscillations, waves and turbulence*, S pringer, Berlin, 1984 **278**
- [78] E. Larson: *Isaacs Sturm*, Fischer, 2000 **203**
- [79] T. Leppänen: *Computational studies of pattern formation in turing systems*, Dissertation, Univ. Helsinki, 2004; <http://lib.tkk.fi/Diss/2004/isbn9512273969/isbn9512273969.pdf> **227**
- [80] T. Leppänen, The theory of Turing pattern formation, Imperial College Press, 2004. [http://www.lce.hut.fi/////research/polymer/turing\\_review.pdf](http://www.lce.hut.fi/////research/polymer/turing_review.pdf) **227**, **230**
- [81] K. Libbrecht and P. Rasmussen: *Snowflake*, Colin Bexter, 2003; viels davon auch unter <http://www.snowcrystals.com/> zu finden, insbesondere die Bilder **199**
- [82] A.J. Lichtenberg, R. Livi, M. Pettini and S. Ruffo: Dynamics of Oscillator Chains, in *Lect. Notes Phys.* **728**, 21–121, 2008, DOI 10.1007/978-3-540-72995-2 **2** **223**
- [83] J.J. Lin: *FPGA Implementation of Central Pattern Generator*, [http://nonlinear.eecs.berkeley.edu/raptor/Publications/James.FPGADocumentation\\_Fall105.pdf](http://nonlinear.eecs.berkeley.edu/raptor/Publications/James.FPGADocumentation_Fall105.pdf) **232**, **233**, **234**
- [84] E. Löffler: *Geographie und Fernerkundung*, Teubner, 1994
- [85] Lorenz, E.: Deterministic Nonperiodic Flow, *J. Atmospheric Sciences* **20**, März 1963 (hier ist der Schmetterling zwar noch eine Möwe, aber sie/er entwickelt sich) **195**, **203**, **227**, **278**
- [86] N. Luhmann: *Einführung in die Systemtheorie*, Carl Auer, Heidelberg, 2006 **39**
- [87] H. Lutz und W. Wendt: *Taschenbuch der Regelungstechnik*, Harri Deutsch, 2003 **49**, **87**, **168**
- [88] D.P. Maki and M. Thompson: *Mathematica modeling and computer simulation*, Brooks/Cole – Thomson Learning, belmont, CA, 2006
- [89] F. Malik: *Unternehmenspolitik und Corporate Governance – Wie Organisationen sich selbst organisieren*, Campus, 2008 **39**
- [90] F. Malik: *Systemisches Management, Evolution und Selbstorganisation*, Haupt, 2004 **39**
- [91] B. B. Mandelbrot: *The fractal geometry of nature*, Freeman, 1983 **42**, **201**, **223**
- [92] B.B. Mandelbrot and R.L. Hudson: *Fraktale und Finanzen – Märkte zwischen Risiko, Rendite und Ruin*, Piper, 2007 **223**
- [93] The mathworks: *SimuLink 6*, [http://www.mathworks.com/access/helpdesk/help/pdf\\_doc/simulink/sl\\_using.pdf](http://www.mathworks.com/access/helpdesk/help/pdf_doc/simulink/sl_using.pdf), vers. 2007a, März 2007 **109**
- [94] J.H. McClellan, R.W. Schafer and M.A. Yoder: *Signal processing first*, Pearson, 2003 **145**, **192**
- [95] B. Meffert und O. Hochmuth: *Werkzeuge der Signalverarbeitung*, Pearson, 2004 **145**, **192**
- [96] R. Meinel, G. Neugebauer, and H. Steudel: *Solitonen*, Akademie Verlag Berlin, 1991 **269**, **270**
- [97] H. Meinhardt: *Models of biological pattern formation*, Academic Press, New York, 1982 **227**, **229**
- [98] H. Meinhardt: *The algorithmic beauty of sea shells*, Springer, 1998 **42**, **43**, **227**
- [99] L. Mérö: *Die Logik der Unvernunft*, rororo, 2003 **241**
- [100] L. Michaelis and M. Menten: Die Kinetik der Invertinwirkung, *Biochem. Z.* **48**, 333–369, 1913 **226**
- [101] A.S. Mikhailov and V. Calenbuhr: *From Cells to Societies*, Springer, 2002
- [102] R. Morris, L. Tarassenko and M. Kenward (eds): *Cognitive Systems*, Elsevier, 2006 **305**

- [103] L.F. Mollenauer, R.H. Stolen and J.P. Gordon: Experimental observations of picosecond pulse narrowing and solitons in optical fibers, *Phys. Rev. Lett.* **45**, 1095–1098, 1980 273
- [104] E. Mosekilde and O.G. Mouritsen (eds): *Modelling of the dynamics of biological systems – Nonlinear phenomena and pattern formation*, Springer, 1995 267
- [105] E. Mosekilde, Yu Maistrenko and D. Postnov: *Chaotic synchronization: applications to living systems*, World Scientific, Singapore, 2002 278
- [106] R. Münch: *Die akademische Elite*, Suhrkamp, 2007 47
- [107] J.D. Murray: *Mathematical biology*, Springer, Berlin, 1983 227, 230, 267
- [108] S. Nettel: *Wave physics*, Springer, Berlin, 1992 269, 270
- [109] H. Nyquist: Certain Topics in Telegraph Transmission Theory, *Trans. Amer. Inst. Elect. Eng.* **47**, 617-644, 1928 131
- [110] R.C. O'Reilly and Y. Munakata: *Computational explorations in cognitive neuroscience*, MIT Press, Cambridge, 2000 235
- [111] M. Osterloh und B. Frey: Die Krankheit der Wissenschaft, *Forschung und Lehre* **11/07**, 670–673, 2007 47
- [112] E. Packel: *The mathematics of games and gambling*, The Mathematical Association of America, 2006 262
- [113] U. Parlitz, A. Pikovsky, M. Rosenblum and J. Kurths: Schwingungen im Gleichtakt, *Phys. J.* **5**, (10)33, 2006, . . . 224
- [114] W.E. Parry: Three voyages for the discovery of the Northwest passage from the Atlantic to the Pacific and narrative of an attempt to reach the North Pole, Vol. 1, Harper & Bro, New York, 1840 268
- [115] T. Parsons: *The social system*, 1951 149
- [116] H.-O. Peitgen und P. H. Richter: *The beauty of fractals – images of complex dynamical systems*, Springer, 1986 42, 201, 223
- [117] H.-O. Peitgen, H. Jürgens and D. Saupe: *Chaos and Fractals – new frontiers of science*, Springer, 1992; Inhalt im wesentlichen zweibändig als deutsche Fassung in [118] und [119] 42, 223
- [118] H.-O. Peitgen, H. Jürgens und D. Saupe: *Bausteine des Chaos – Fraktale*, rororo 1998 201, 223, 303
- [119] H.-O. Peitgen, H. Jürgens und D. Saupe: *Chaos – Bausteine der Ordnung*, rororo 1998 201, 223, 303
- [120] I. petras: Method for simulation of the fractional order chaotic system, *Acta Mont. Slov.* **11**, (4)273-277, 2006, <http://www.doaj.org/doaj?func=fulltext&passMe=http://actamont.tuke.sk/pdf/2006/n4/7petras.pdf> 224
- [121] A. Pikovsky, M. Rosenblum and J. Kurths: *Synchronization: a universal concept in nonlinear science*, Cambridge University Press, Cambridge 2001 278
- [122] S. Pinker: *The language instinct*, William Morrow & Co, 1994 47, 52
- [123] Pollack: *Uncertain science – uncertain world*, ... 279
- [124] A.D. Poularkis and Z.M. Ramadan: *Adaptive Filtering Primer with MATLAB*, Taylor and Francis, 2006 192
- [125] W.H. Press, A.S.A. Teukolsky, W.T. Vetterling and B.P. Flannery: *Numerical Recipes 3<sup>d</sup> Edition*, Cambridge University Press, 2007; ältere online Versionen (Fortran, C) unter <http://www.nr.com/oldverswitcher.html> 134, 137
- [126] Prisoner's Dilemmas: The iterated prisoner's dilemma competition: celebrating the 20th anniversary, <http://www.prisoners-dilemma.com/>, 30.12.2007 260
- [127] A. Rapoport: *Kämpfe, Spiele und Debatten*, Verlag Darmstädter Blätter, 1976 240, 242, 245, 262
- [128] E. Rasmussen: *Games and information*, Blackwell, Cambridge MA, 1994 262
- [129] M. Remoissenet: *Waves called solitons*, Springer, 1999 271
- [130] H. Riedrich: *Nukleation auf vikinalen Oberflächen*, BA thesis, Osnabrück, 2006 230
- [131] J.S. Russell: *Report on waves*, British Association Reports, 1844 269
- [132] J.S. Russell: *The wave of translation in the oceans of water, air and ether*, trübner, London, 1855 269

- [133] F.F. Sabins, Jr.: *Remote sensing – Principles and interpretation*, Freeman, 1996 192, 193
- [134] M.A. Savageau: michaelis–Menton mechanism reconsidered: implications of fractal kinetics, *J. Theor. Biol.* 176, 115–124, 1995 226
- [135] M. Scheffer: Fish and nutrient interplay determines algal biomass: a minimal model, *OIKOS* 62, 271–282, 1991 229
- [136] A.C. Scott: *Stairways to the mind: the controversial new science of conscious*, Springer, Berlin, 1995 267
- [137] A. Scott: *Neuroscience – A mathematical primer*, Springer, New York, 2002 230, 231, 267
- [138] A. Scott: *Nonlinear Science – Emergence and dynamics of coherent structures*, Oxford, 2003 47, 220, 225, 231, 267, 270
- [139] A. Scott: *The nonlinear universe – chaos, emergence, life*, Springer, 2007 225, 230, 270, 271
- [140] A.F. Seila, V. Ceric, and P. Tadikamalla: *Applied simulation modeling*, Brooks/Cole – Thomson Learning, Belmont, CA, 2003
- [141] L.F. Shampine and M.W. Reichelt: *The MATLAB ODE Suite*, The MathWorks, INc., 1996, [http://www.mathworks.com/access/helpdesk/help/pdf\\_doc/otherdocs/ode\\_suite.pdf](http://www.mathworks.com/access/helpdesk/help/pdf_doc/otherdocs/ode_suite.pdf) 95
- [142] L.F. Shampine, I. Gladwell, and S. Thompson: *Solving ODEs with MATLAB*, Cambridge University Press, 2003 95
- [143] C.E. Shannon: *A mathematical theory of communication*, Bell Syst. Tech. J. 27, 379-423 und 623-656, 1948 171
- [144] C.E. Shannon and W. Weaver: *The mathematical theory of information*, Univers. Ill. Press, Urbana, 1949; deutsch *Mathematische Grundlagen der Informationstheorie*, Oldenbourg, München, 1976 171
- [145] C. E. Shannon: Communication in the Presence of Noise, *Proc. IRE* 37, No. 1, Jan. 1949 132
- [146] Jumping Shi: Turing patterns in animal coats, <http://www.resnet.wm.edu/~jxshix/Mathbiology-1.ppt> 228, 229
- [147] W. Sierpinski: Sur une courbe cantorienne dont tout point est un point de ramification, *C.R. Acad. Paris* 160, 302, 1915 42
- [148] W. Sierpinsky: Sur une courbe cantorienne qui contient une image biunivoque et continue de toute courbe donnée, *C.R. Acad. Paris* 162, 629–632, 1916 42
- [149] W. Slany and W. Kienreich: On some winning strategies for the Iterated Prisoner's Dilemma or Mr. Nice Guy and the Cosa Nostra, arXiv:cs/0609017v1, 2006. 260
- [150] L. Smith: *Chaos – A Very Short Introduction*, Oxford University Press, Oxford, 2007 222
- [151] M.R. Spiegel: *Laplace Transformationen – Theorie und Anwendung*, McGraw-Hill, 1977 145
- [152] Stehr, N. and von Storch, H.: The Social Construct of Climate and Climate Change, *Clim. Res.* 5, 99–110, 1995 254
- [153] N. Stehr und H. v. Storch: *Klima, Wetter, Mensch*, Beck, 1997 254, 259
- [154] Stoecker, H.: *Taschenbuch mathematischer Formeln und moderner Verfahren*, 4te Auflage, Harri Deutsch, 1999 122, 125, 142, 145
- [155] R.S. Strichartz: *Differential equations on fractals – a tutorial*, Princeton University Press, 2006 201
- [156] S. Strogatz: *sync – the emerging science of spontaneous order*, Penguin Books, 2003 47, 224, 278
- [157] S. Strogatz: From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators, *Physica D* 143, 1–20, 2000 224
- [158] J. Swinton: Watching the daisies grow: Turing and Fibonacci phyllotaxis, Talk given at Turing 2004, [www.swintons.net/jonathan/turing.htm](http://www.swintons.net/jonathan/turing.htm) 228



- [159] T. Tél and M. Gruis: *Chaotic Dynamics – An introduction based on classical mechanics*, Cambridge, 2006 202, 206, 209, 210, 211, 222
- [160] D’Arcy Wentworth Thompson: *Über Wachstum und Form*, Die Andere Bibliothek, 2006 267
- [161] M. Toda: Vibration of a chain with nonlinear interactions, *J. Phys. Soc. Japan* **22**, 431–436, 1967 219
- [162] M. Toda: *Theory of nonlinear lattices*, Springer, Berlin, 1981 219, 220
- [163] T.P. Trappenberg: *Fundamentals of computational neuroscience*, Oxford, 2002 235
- [164] A. Turing: Computing machinery and intelligence, *Mind* **59**, 433, 1950 227, 278
- [165] A. Turing: The Chemical Basis of Morphogenesis, *Phil. Trans. R. Soc. Lond.*, **B237**, 37-72, 1952 44, 227, 267
- [166] J. Tyndall: *The forms of water in clouds and rivers, ice and glaciers*, Appleton and Co, New York, 1872 147
- [167] B. van der Pol: The nonlinear theory of electric oscillations, *Proc. IRE* **22**, 1051–1086, 1934 277
- [168] E. van Groesen and J. Molenaar: *Continuum modeling in the physical sciences*, SIAM, Philadelphia, 2007
- [169] R. van Gulick: Reduction, emergence and other recent options on the mind/body problem, *J. Consc. Stud.* **8** (9-19)1–34, 2001 265, 266, 267
- [170] J. Wainwright and M. Mulligan: *Environmental modeling – finding simplicity in complexity*, Wiley and Sons, 2004
- [171] J.H. Weaver: *Theory of discrete and continuous Fourier analysis*, John Wiley & Sons, 1989 121
- [172] U. Wilensky: *NetLogo Fire model*, <http://ccl.northwestern.edu/netlogo/models/Fire>, Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, 1997
- [173] D. Willshaw: Self-Organisation in the nervous system, in [102], p. 5–33 47, 229, 267
- [174] A.T. Winfree: *The geometry of biological time*, Springer, Berlin, 2000 277, 278
- [175] C.C. Winterbourn, M.B. Hampton, J.H. Livesay and A.J. Kettle: Modeling the reaction of superoxide and myeloperoxidase in the neutrophil phagosome, *J. Biol. Chem.* **281**, 39860–39869, 2006, DOI 10.1074/jbc.M605898200, <http://www.jbc.org/cgi/reprint/281/52/39860> und zusätzliches Material unter <http://www.jbc.org/cgi/content/full/M605898200/DC1> 226
- [176] I.A. Witten, A. Moffat and T.C. Bell: *Managing Gigabytes*, 2<sup>nd</sup>ed., Morgan and Kaufman, 1999
- [177] D.J. Wollkind and L.E. Stephenson: Chemical Turing pattern formation analysis: comparison of theory with experiment, *SIAM J. Appl. Math* **61**, 387–431; <http://epubs.siam.org/sam-bin/dbq/article/32621> 230
- [178] Woltering, M.: *Chaos und Turbulenz in der Belousov-Zhabotinsky-Reaktion*, Diss., Dortmund 2002, <http://eldorado.uni-dortmund.de:8080/FB2/ls7/forschung/2002/Woltering/maWolteringunt.pdf> 44
- [179] Wupper: *Elektronische Schaltungen 1*, Springer, Berlin, 1996 168
- [180] F. Xu and H. Ding: A new kinetic model for heterogenous (or spatially confinde) enzymatic catalysis: contributions from the fractal and jamming (overcrowding) effects, *Apl. Catal. A. Gen.* **317**, 70–81, 2007 226
- [181] Xin-She Yang: Turing pattern formation for catalytic reaction-diffusion systems in engineering application, *Modeling Simul. Mater. Sci. Eng.* **11**. 321, 2003; <http://www.iop.org/EJ/abstract/0965-0393/11/3/305/> 230, 232
- [182] A.M. Zhabotinsky: Periodic processes of malonic acid oxidation in a liquid phase (in russ.), *Biofizika*, 9:306–311, 1964. Auf deutsch: A. M. Zhabotinsky: Eine periodische Oxydationsreaktion in flüssiger Phase. In: L. Kuhnert, U. Niedersen (Hrsg.): Selbstorganisation chemischer Strukturen. Verlag Harri Klein, Frankfurt/Main 2001, S. 83-89 44

# Index

AbsTol, 93  
Absolute Tolerance, 97  
Checkbox, 82  
Clock, 74, 88  
CloseFcn, 108  
Dead Zone, 100  
Decimation, 93  
Derivative, 57  
Discrete, 95  
Divide, 61  
DstWorkspace, 94  
Enabled Subsystem, 85  
Enabled and Triggered Subsystem, 85  
Fcn, 31, 88  
FinalStateName, 93  
FixedStep, 93  
From File, 65  
From Workspace, 65  
Gain, 21, 62, 189  
IC, 99  
InitFcn, 108  
Initial step Size, 96  
InitialState, 93  
InitialStep, 93  
Integrator, 58  
Math Functions, 88  
Math Function, 63  
Max Step Size, 96  
MaxOrder, 93  
MaxRows, 93  
MaxStep, 93  
Memory Block, 106  
Mux, 23  
ODE113, 95, 98  
ODE14x, 95  
ODE15S, 95  
ODE15s, 93  
ODE1, 95  
ODE23S, 95  
ODE23TB, 95  
ODE23T, 95  
ODE23, 95, 97  
ODE2, 95  
ODE3, 95  
ODE45, 92, 95, 97  
ODE4, 95  
ODE5, 95  
OutputPoints, 93  
Outputvariables, 93  
PoastLoadFcn, 108  
Popup, 82  
PostSaveFcn, 108  
Power Spectral Density, 134, 139  
PreLoadFcn, 108  
PreSaveFcn, 108  
Refine, 93  
RelTol, 93  
Relative Tolerance, 97  
Relay, 101  
Scope, 21, 64  
Selector, 205  
Solver, 93  
SrcWorkspace, 94  
StartFcn, 108  
State Space, 26, 27, 60, 76, 181  
StopFcn, 108  
Sum, 63, 189  
Switch, 99  
To File, 65  
To Workspace, 65  
Trace, 94  
Transfer Fcn, 60  
Triggered Subsystem, 85, 86  
Triggered and Enabled Subsystem, 86  
Trigonometric Function, 63  
Unit Delay, 188, 189  
Weighting Moving Average, 187  
ZeroCross, 94  
butter, 190  
filter2, 170  
filter, 170, 186  
find\_system, 91  
gcs, 91  
get\_param, 91

- initial condition, 99
- ode5, 32
- out, 31
- set\_param, 91
- simget, 94
- simplot, 92
- simset, 32, 93
- sim, 89, 92
- sum, 23
  
- abatement cost, 250
- Abschneidefrequenz, 131
- Abstraktion, 32
- Abtastfrequenz, 172
- Abtastintervall, 121
- Abtasttheorem, 123, 132
- AD-Wandler, 172
- Additional Sinks, 65
- Adler Gleichung, 276, 277
- Aktionspotential, 230
- Albedo, 164
- algebraische Schleife, 99, 104, 196
- Altruist, 245
- Amplitudenspektrum, 118
- Analogfilterung, 173
- Anfangsbedingung, 59, 60
- Anfangswert, 149
- Anfangszustand, 156
- Antisoliton, 274
- Apfelmännchen, 42, 104, 198
- Arnold's Zunge, 276
- Attraktor, 196, 197, 201
  - limit cycle, 202
  - seltsam, 202
- Ausgangsgröße, 148, 149, 151, 155–157
- Ausgangsmatrix, 27
- Ausgangsvariablen, 92
- Autokorrelation, 128, 132, 140
  
- Bandbegrenzung, 172
- Bandbreitenbegrenzung, 173
- bandbreitenlimitiert, 117
- Bandpass, 177
- Barnsley Farn, 42, 43
- Bartlett window, 137
- basin of attraction, 201
- Basis, 184
- Basisfunktion, 118
- Basisvektor, 118
- Belousov–Zhabotinsky-Reaktion, 42, 44, 225, 226, 229, 231
- Benard Konvektion, 203
- Bewusstsein, 266
- Beziehungsgeflecht, 39–41
- Bibliothek, 84
- Bibliotheksblock, 84
- Bifurkation, 196, 197, 201, 202
- Bildfunktion, 141
- Bildraum, 141
- Block
  - maskiert, 84
  - maskiertes Untersystem, 81
- Block Library, 84
- Blockdiagramm, 18, 20, 21, 36, 189, 292
- Bode-Diagramm, 158, 168
- Bogacki–Shampine Verfahren, 95
- Boomeron, 274
- Buisness As Usual, 249
- butterfly effect, 278
- Butterworth Filter, 174, 178
  - vierte Ordnung, 190
  
- Callback, 82, 108
- Cauer Topologie, 178, 180
- Central Pattern Generator, 233
- Chaos, 195, 196
- Chernobyl, 279
- Chua-Oszillator, 224
- Cooley–Tukey-Algorithmus, 134
- CPG, 233, 234
- Crank–Nicolson Verfahren, 283
  
- damage cost, 250
- Dead Code, 103
- Dead Zone, 101
- Decimation, 102
- Delta Funktion, 124, 125
- Demultiplexer, 38
- Devil's Staircase, 276
- DFT, 191
- Differentialgleichung
  - steif, 95
- Differentiationsregel, 142
- Differenzgleichung, 183
- Diffusion, 264
  - Wirbeld., 265
- Diffusions–Reaktions-Modell, 225, 227, 230, 232, 264
- Diffusionsgleichung, 226, 265
- Diffusionskoeffizient, 226, 265
- Diffusionsstrom, 265
- Diffusionstensor, 226, 265
- Dimension
  - fraktal, 200, 202
  - Hausdorff, 200
- Discontinuities Block Library, 100
- Discrete Transfer Function, 190
- Diskretisierung, 70, 199, 284, 285
  - zweite Ableitung, 285
- Diskretisierungsfehler, 70, 282

- Dispersion, 265  
 DNLS, 273  
 Dorman–Prince, 32  
 Dorman–Prince Verfahren, 95  
 Dormand–Prince Verfahren, 95  
 Drehmatrix, 156  
 Druck—Temperatur–Rückkopplung, 164  
 Duffing-Oszillator, 197, 211, 212  
 Durchlassbereich, 174
- ECHAM5, 248  
 Egoist, 245  
 Eigenfunktion, 161, 163, 166, 167  
   Tiefpass, 166  
 Eigenvektor, 35, 167  
 Eigenwert, 14, 15, 35  
 Eigenwertgleichung, 35, 166  
 Eingabevektor, 60  
 Eingangs–Ausgangs-Relation, 156, 162  
 Eingangsfunktion  
   harmonisch, 157  
 Eingangsgröße, 148, 149, 151, 155  
 Eingangsmatrix, 27  
 Einheitsverzögerung, 188  
 Einpersonenspiel, 246, 247  
 Eis–Albedo-Feedback, 164  
 emergence  
   modest, 266  
   radical, 266  
   specific value, 265  
 Emergenz, 47, 196, 225, 265  
   schwache Form, 266  
   starke Form, 266  
 Entersystem, 79  
 Enzymkinetik, 226  
 Erhaltungsgrößen, 71  
 Euler Rückwärts Verfahren, 283  
 Euler Verfahren, 95  
   Korrektorschritt, 283  
   modifiziertes, 283  
   Prädikatorschritt, 283  
   rückwärts, 283, 286  
   vorwärts, 283, 286  
 Euler Vorwärts, 18  
 Euler Vorwärts Verfahren, 283  
 explizites Verfahren, 283  
 explizites/implizites Verfahren, 283  
 Exponentialansatz, 14, 15, 35, 154  
 Exponentialfunktion  
   Matrix als Argument, 154
- Fadenpendel, 13, 28, 72, 202  
 Faltung, 126, 127, 132, 149, 153, 155  
 Faltungssatz, 113  
 Fast Fourier Transform, 132  
 Federpendel, 13, 95  
   gekoppelte, 34  
 Feigenbaum, 196  
 Feigenbaum Attraktor, 42, 197  
 Fermi–Pasta–Ulam, 214, 221  
 FFT, 132, 133, 172  
 Filter, 27, 119  
   analog, 170, 172, 182  
   Butterworth, 178  
   Differenzgleichung, 183  
   digital, 170, 172, 182, 183  
     Ordnung, 183  
   Echtzeit, 183  
   Glättung, 182  
   gleitendes Mittel, 182  
   kausal, 183  
   nicht-kausal, 183  
   Ordnung, 183  
   rekursiv, 183, 189  
   Tschebyscheff, 180  
 Filter-Kern, 187  
 Finite Impulse Response, 182  
 FIR, 182  
 Fixpunkt, 201  
 flowchart, 292  
 Flussdiagramm, 18, 20, 36, 292  
 FNLS, 272  
 Folgekosten, 250, 251  
 Fourier, 148  
 Fourier Analyse, 48, 113  
 Fourier Koeffizient, 117–119, 121  
 Fourier Reihe, 113, 115, 118, 119, 121, 122  
 Fourier Transformation, 48, 112, 113, 119, 121, 140, 141, 148, 175, 177, 272  
   diskret, 114, 129, 191  
   kontinuierliche, 113  
 FPU, 221  
 FPU Lattice, 219  
 Fraktal, 42, 43, 196, 198, 200  
   quaternionisch, 224  
 fraktale Dimension, 200  
 fraktale Kinetik, 226  
 Freerider, 257  
 Frequency Entrainment, 274, 276  
 Frequency Locking, 277  
 Frequenz  
   komplex, 141  
 Frequenzgang, 166, 167  
 Frequenzspektrum, 122, 172  
 Freundlichkeit, 245

- Gauß Funktion, 124  
 GCM, 248  
 Gefangenendilemma, 240–242  
     iteriertes, 244  
 Gegenkopplung, 164  
 Gemeindewiese, 244, 246, 255, 257  
 Generator, 200, 201  
 GES, 247  
 Gesamtsystem, 34  
 Gewichtsfunktion, 126, 156, 157, 161, 163  
 Gewinnmatrix, 241  
 Gibbs'sches Phänomen, 123  
 Glättung, 182  
 Gleichgewichtspunkt, 201  
 Gleichstromanteil, 116, 117, 130  
 gleitendes Mittel, 182  
 global circulation model, 248  
 Global Environment and Society Model, 247  
 Green'sche Funktion, 148, 246, 248  
 Grenzfall  
     aperiodisch, 15, 16  
 Grenzfrequenz, 138, 159, 174, 175  
  
 Hamilton-System, 221  
 HAMMONIA, 248  
 Hann window, 137  
 Hausdorff Dimension, 200  
 Heun's Verfahren, 95  
 Hochpass, 48, 138, 174–176  
 Hodgkins–Huxley, 226, 230, 234, 266  
 Humunculus, 47  
 Hyperzeichen, 171  
 Hysterese, 101  
  
 Icon, 84  
 IIR, 183, 189  
 implizites Verfahren, 283  
 Impulsantwort, 48, 148, 149, 188, 246, 248  
 Impulsfunktion, 140  
 Information, 120, 171, 172  
 Informationsgehalt, 120  
 Initiator, 200, 201  
 Integraltransformation, 22, 112  
 integrate and fire, 277  
 Integrator, 21, 23, 138  
 Interessenkonflikt, 239  
 Intervallschachtelung, 291  
 Iraqi war, 279  
 Iteration, 291  
  
 Jäger–Beute Modell, 229  
 JPEG, 123  
 Julia Menge, 198  
  
 Küstenlänge, 198, 199  
  
 Katrina, 279  
 Kausalität, 150  
 Kern, 131  
 Kernel, 131, 186, 187  
 Klima  
     Folgekosten, 250  
     Vermeidungskosten, 250  
 Klimasystem, 14  
 Klimavektor, 248  
 Koch'sche Schneeflocke, 43, 200  
 Konkurrenz, 242, 245, 247, 254  
 Konsistenz, 286  
 Konvergenz, 286  
 Konvergenzabszisse, 141  
 Kooperation, 240, 242–244, 247, 254  
 Kopplungsfunktion, 275  
 Kopplungsterm, 34, 36  
 Korrektorschritt, 283  
 Korteweg–de Vries Gleichung, 220, 266, 270, 271  
     erweitert, 272  
     modifizierte, 270  
 Kreuzkorrelation, 127, 132, 140  
 Kriechfall, 15  
  
 Lagrange Funktion, 73  
 Laplace Transformation, 22, 25, 27, 48, 76, 112, 140, 148, 153, 155, 170, 175, 177, 191  
     Differentiationsregel, 142  
     Faltungsregel, 141  
     Tabelle, 142  
 Laplace Transformierte, 141  
 Laplace'scher Dämon, 112  
 Lax'sches Äquivalenztheorem, 286  
 leakage, 137  
 Leapfrog Verfahren, 283  
 Leben, 266  
 Leistungsdichtespektrum, 134, 139  
 Library, 84  
 Library Browser, 56  
 limit cycle, 201, 202, 234  
 Limit Output, 98  
 Linearität, 150, 151  
 Lissajous-Figur, 78  
 logisch isomorph, 242  
 logistische Gleichung, 202  
 Lorentz Funktion, 124, 125  
 Lorenz, 278  
 Lorenz Gleichungen, 204  
 Lorenz' butterfly, 278  
 LTI-System, 60, 147  
  
 Magnetpendel, 202, 223  
 Mandelbrot, 201

- Mandelbrot Menge, 198  
 MANIAC, 220  
 Masken-Editor, 81
  - Dokumentation, 83
  - Icon, 84
  - Initialisierung, 82
  - Parameter, 82
 maskierter Block, 84  
 maskiertes Untersystem, 81, 83  
 Matrizen
  - SimuLink, 204
 Mehrpersonenspiel, 246, 247  
 Michaelis–Menten–Reaktionsrate, 226  
 Mitkopplung, 164  
 Mittel
  - gleitendes, 183, 184
 Mittelpunktsformel, 282  
 modest emergence, 266  
 Morphogenese, 267  
 multi-actor case, 247  
 Multiplexer, 23  
 Muster, 39  
  
 Nachricht, 171  
 Nachsichtigkeit, 245  
 Nash–Gleichgewicht, 241, 256, 257  
 Neuron, 41
  - bipolar, 41
  - multipolar, 41
 Newton–Raphson Methode, 105, 291, 292  
 NLS, 272  
 Nullstelle, 163, 191, 291  
 Nullsummenspiel, 239–241, 244, 246  
 Nyquist Frequenz, 131  
  
 Optimierung, 103  
 Optimierungsproblem, 247  
 Originalfunktion, 141  
 Oszillator
  - chemisch, 42
  - relaxierend, 277
  - stick–slip, 278
 Output–Input–Relation, 48  
  
 Parallelschwingkreis, 40  
 Parameter
  - Übergabe aus Workspace, 80
  - Untersystemmaske, 82
 Parry, 268  
 Parseval’s Theorem, 127  
 Periodogramm, 134  
 Phase, 275  
 Phase Locking, 276, 277  
 Phase Locking, 274  
 Phasenraum, 201  
  
 Phasenspektrum, 118  
 Poincare–Portrait, 196  
 Poisson Integral, 149  
 Pol, 163, 191  
 Polstelle, 167  
 Prädikatorschritt, 283  
 Projektion
  - gefiltert, 119
 Provozierbarkeit, 245  
 Punkt–Attraktor, 201  
 Purkinje Fasern, 276  
 Purkinje–Zellen, 41  
  
 Rückkopplung, 104, 105, 148, 164
  - negativ, 164
  - positiv, 164
 Rückwärts Schema, 284  
 radical emergence, 266  
 rapid application development, 22  
 rapid prototyping, 22  
 Rauschen, 137  
 Rauschunterdrückung, 173  
 RC–Hochpass, 168  
 Reaktion
  - chemisch, 226
 Reaktions–Diffusions–System, 44  
 Rechteckfunktion, 115, 122  
 Referenzblock, 84  
 Regelkreis
  - Offener, 165
  - Standard, 166
 Rekursionsformel, 291  
 Relaxation, 277  
 Resolvente, 76, 160, 162, 163, 166  
 Resonanz, 275  
 Resonanzkatastrophe, 16  
 Reynolds–Zahl, 208  
 Reziprozität, 245  
 Runge–Kutta Verfahren, 95, 284, 286  
 Russell, 220, 269  
  
 s–Function, 205, 206  
 Sägezahnfunktion, 115  
 Schelling Spiel, 243  
 Schleife
  - algebraisch, 99, 204
 Schmetterlingseffekt, 195, 203  
 Schmetterlingseffekt, 227  
 Schmitt–Trigger, 101  
 Schneeflocke, 199
  - Koch’sche, 200
 Schrödinger Gleichung
  - nichtlinear, 272
    - defokussierend, 273
    - fokussierend, 272

- Schwingfall, 15  
 Schwingung  
   erzwungene, 16, 17  
   freie, 14, 17, 21, 114  
 Selbstähnlichkeit, 42, 197–200  
 Selbstorganisation, 28, 42, 44, 47, 225  
 Selbstorganisaton, 42  
 Serienschwingkreis, 40, 156  
 show state port, 99  
 si, 121, 123  
 Siebkette, 40  
 Sierpinski Dreieck, 42, 43, 200  
 Signal, 148, 171  
   diskret, 190  
 SIMO, 61, 187  
 Simpson-Verfahren, 282  
 sinc, 121, 123  
 sine cardinal, 121, 123  
 Single input single output SISO, 26  
 single-actor case, 246, 247  
 Single-input multiple-output, 61  
 Sinks, 64  
 Sinusknoten, 276  
 SISO, 26, 187  
 Skalierungsgesetz, 200  
 Skalierungssatz, 132, 134  
 Soliton, 196, 214, 220, 221, 267, 268, 270  
   algebraisch, 272  
   Anti-, 274  
   dunkles, 273  
   helles, 273  
   nicht-topologisch, 271  
   rational, 272  
   topologisch, 273  
 Solver, 92, 94  
 Spaltfunktion, 123, 124  
 specific value emergence, 265  
 Sperrbereich, 174  
 Sperrfrequenz, 174  
 Sperrkreis, 40  
 Spiel  
   isomorph, 242  
   iteriertes, 246  
 Sprungantwort, 48, 140, 148, 163  
 Sprungfunktion, 122, 140  
 Standardregelkreis, 166  
 State Port, 99  
 Stichling, 245  
 stick-slip Oszillation, 278  
 Strategie, 239  
   gemischte, 238  
   reine, 238  
 Struktur, 39, 40  
 Subsystem, 79, 234  
 Subsystems-Block, 79  
 Summationspunkt, 63  
 Superposition, 15, 153  
 Superpositionsprinzip, 150, 151  
 Support, 184  
 Switch, 100  
 Switch-Block, 100  
 synchrones Regime, 276  
 Synchronisation, 218, 274, 277  
 System, 39, 148  
   angeregt, 153  
   Definition, 39  
   dynamisch, 40  
   frei, 153  
   freies, 154  
   kausales, 150  
   lineares, 151  
   Parallelschaltung, 165  
   selbst-organisierend, 42  
   Serienschaltung, 164  
   statisch, 40  
 Systemantwort, 161  
 Systemdynamik, 292  
 Systemeigenschaften, 151  
 Systemelement, 39–41  
 Systemgleichung, 157, 162  
 Systemgleichungen, 13  
 Systemgröße, 148, 149  
 Systemmatrix, 27, 151, 157, 160, 162  
   Eigenvektor, 167  
 Systemmodul, 149  
 Systemparameter, 21  
 Systemtheorie, 48, 149  
   Grundaufgabe, 149  
 Systemvariable, 48, 148  
 Taylor Entwicklung, 284, 285  
 Teilsystem, 149  
 Testfunktion, 148  
 Tiefpass, 48, 54, 138, 144, 149, 161, 162, 166–  
   168, 174, 176, 182–184  
   Übertragungsfunktion, 161  
   zweite Ordnung, 168, 177  
 Tit for Tat, 245  
 Toda, 219  
 Toda-Lattice, 219  
 Transferfunktion, 54, 55  
 Transformation, 153  
 Transformationsmatrix, 152  
 Transitionsmatrix, 154, 156, 157, 160, 161,  
   163  
 Trapezformel, 282  
 Trappon, 274  
 Truncation Error, 285  
 Tschebyscheff Filter, 180, 181

- Tsunami, 267  
 Turing, 44, 278  
 Turing's avalanche, 279  
 Turing's Lawine, 227  
 Turing-Muster, 44  
  
 Überraschungswert, 120  
 Übertragbarkeit, 32  
 Übertragungsfunktion, 13, 25, 27, 28, 48, 54,  
     57, 60, 61, 76, 126, 127, 147, 148,  
     158, 161–164, 166, 167, 170, 173, 175,  
     190, 246, 248  
     allgemeine, 166  
     Bildbereich, 161, 162  
     diskretes System, 190  
     faktoriert, 163  
     Frequenzgang, 167  
     Nullstelle, 163, 167  
     offener Regelkreis, 165  
     Parallelschaltung, 165  
     Pol, 163  
     Polstelle, 167  
     rückgekoppelte Systeme, 165  
     Serienschaltung, 164  
     Standardregelkreis, 166  
     Struktur, 163  
 Ueda-Oszillator, 213  
 unit act, 149  
 Unterabtastung, 172  
 Untersystem, 79–81, 148  
     maskiert, 81, 83, 205  
 User-Defined Functions, 88  
  
 van der Pol, 234  
 van der Pol Oszillator, 224  
 van der Pol's Gleichung, 277  
 Variable  
     dimensionslos, 14, 290  
     lokal, 83  
     maskiertes Untersystem, 83  
 Vermeidungskosten, 250, 251  
 Verschiebungssatz, 131, 151  
 Verständlichkeit, 245  
 Vorwärts Schema, 284  
  
 Wadi-Modell, 46  
 Welch window, 137  
 Wettrüsten, 242  
 Widerstandsverhältnis, 152  
 Windowing, 137  
 windowing, 137  
 Wirbeldiffusion, 265  
  
 z-Transformation, 191  
 Zeichen, 171  
  
 zeitinvariant, 150  
 Zeitinvarianz, 151  
 Zeitkonstante, 152  
 Zeitserie, 114  
 zellulärer Automat, 44  
 zellulärer Automat, 231  
 Zener-Diode, 100  
 zentrale Differenz, 284  
 zero crossings control, 98  
 Zero Padding, 132, 137, 139, 185  
 Zielkonflikt, 240–242, 244  
 Zustand, 149  
 Zustandsgröße, 151  
 Zustandsvariable, 92, 149, 152, 153  
 Zustandsvektor, 26, 60, 149, 151, 248