

UNIVERSITÄT
BAYREUTH

Modellprädiktive Regelung für nichtlineare evolutionäre partielle Differentialgleichungen

Theorie

Diplomarbeit

von

Stefan Trenz

FAKULTÄT FÜR MATHEMATIK, PHYSIK UND
INFORMATIK

MATHEMATISCHES INSTITUT

Datum: 29. September 2008

Aufgabenstellung / Betreuung:
Prof. Dr. L. Grüne

Danksagung

An dieser Stelle möchte ich mich recht herzlich bei Herrn Prof. Dr. Grüne für die stets ausgezeichnete und freundliche Betreuung und Beratung, vor allem in den vergangenen Monaten während der Entstehung dieser Arbeit, bedanken.

Ein herzliches Dankeschön gebührt des Weiteren Karl Worthmann und Jürgen Pannek für deren endlose Geduld und Hilfestellung bei der Einarbeitung in die modellprädiktive Regelung und in das zur Verfügung gestellte Programm, sowie für viele hilfreiche und motivierende Anregungen, Ratschläge und Diskussionen.

Ganz besonderer Dank gilt meinen lieben Eltern, die mir dieses Studium überhaupt erst ermöglicht und mich während dieser Zeit stets geduldig und verständisvoll unterstützt haben.

Inhaltsverzeichnis

| | |
|---|-----------|
| Tabellenverzeichnis | IV |
| Abbildungsverzeichnis | V |
| 1 Einleitung | 1 |
| 1.1 Motivation und inhaltlicher Aufbau | 1 |
| 1.2 Anmerkung zur Konzeption der Arbeit | 3 |
| 2 Partielle Differentialgleichungen | 5 |
| 2.1 Motivation | 5 |
| 2.2 Grundbegriffe und Definitionen | 7 |
| 2.2.1 Gewöhnliche und partielle Differentialgleichungen | 7 |
| 2.2.2 Algebraische Eigenschaften | 8 |
| 2.2.3 Ein gut gestelltes (well-posed) Problem | 10 |
| 2.3 Klassifikation partieller Differentialgleichungen | 11 |
| 2.4 Transformation von Differentialgleichungssystemen höherer Ordnung | 13 |
| 2.5 Anfangs- und Randbedingungen | 15 |
| 2.5.1 Anfangsbedingungen | 15 |
| 2.5.2 Randbedingungen | 16 |
| 2.5.2.1 Dirichlet-Randbedingung | 17 |
| 2.5.2.2 Neumann-Randbedingung | 17 |
| 2.5.2.3 Cauchy-Randbedingung | 18 |
| 2.6 Kontrollmöglichkeiten der Systeme | 19 |
| 2.6.1 Randsteuerung | 19 |
| 2.6.2 Verteilte Steuerung | 20 |
| 3 Modellprädiktive Regelung von Kontrollsystemen | 23 |
| 3.1 Einführung in die modellprädiktive Regelung | 23 |
| 3.2 Mathematische Kontrolltheorie | 24 |
| 3.2.1 Unterscheidung von Steuerung und Regelung | 25 |
| 3.2.2 Stabilität dynamischer Systeme | 26 |
| 3.3 Kontrollsysteme | 29 |
| 3.3.1 Definition eines Kontrollsystems | 29 |
| 3.3.2 Existenz und Eindeutigkeit | 30 |
| 3.3.3 Asymptotische Kontrollierbarkeit und Feedback-Stabilisierung | 32 |
| 3.4 Übergang von zeitkontinuierlichen zu zeitdiskreten Systemen | 33 |
| 3.4.1 Das Prinzip der Abtastung | 33 |
| 3.4.2 Stabilisierung mittels Abtast-Feedback | 34 |
| 3.4.3 Abtastung und Ljapunov-Funktionen | 36 |

| | | |
|----------|---|-----------|
| 3.5 | Optimale Steuerung | 37 |
| 3.5.1 | Bellman'sches Optimalitätsprinzip | 38 |
| 3.5.2 | Klassen von Lösungsansätzen für Optimalsteuerungsprobleme | 39 |
| 3.5.2.1 | Globale Verfahren | 39 |
| 3.5.2.2 | Lokale Verfahren | 40 |
| 3.6 | Die modellprädiktive Regelung | 41 |
| 3.6.1 | MPC Verfahren im Allgemeinen | 41 |
| 3.6.2 | Methodik und Strategie | 42 |
| 3.7 | Das nichtlineare MPC-Problem | 42 |
| 3.7.1 | Grundlegende Komponenten und Definitionen | 44 |
| 3.7.2 | Formulierung des NMPC-Problems | 47 |
| 3.7.3 | Notwendigkeit eines endlichen Horizonts für NMPC | 47 |
| 3.8 | Analyse der MPC-Schemata | 48 |
| 3.8.1 | Asymptotische Kontrollierbarkeit | 49 |
| 3.8.2 | Erste Resultate | 49 |
| 3.8.3 | Asymptotische Stabilität | 51 |
| 3.8.4 | Wahl einer geeigneten Kostenfunktion l | 53 |
| 4 | Diskretisierungen | 57 |
| 4.1 | Definitionen | 57 |
| 4.1.1 | Gitterdiskretisierung | 57 |
| 4.1.2 | Semidiskretisierung | 61 |
| 4.1.2.1 | Zeitliche Diskretisierung | 61 |
| 4.1.2.2 | Räumliche Diskretisierung | 62 |
| 4.2 | Numerische Differentiation | 63 |
| 4.2.1 | Taylor-Entwicklung | 63 |
| 4.2.2 | Finite-Differenzen-Methode | 65 |
| 4.2.2.1 | Differenzenapproximationen erster Ordnung | 67 |
| 4.2.2.2 | Differenzenapproximationen zweiter Ordnung | 69 |
| 4.2.2.3 | Differenzenapproximation der Neumann-Randbedingung | 69 |
| 4.2.3 | Upwind-Methode | 70 |
| 4.2.4 | Stabilität bezüglich der Finite-Differenzen-Methode | 72 |
| 4.3 | Methode der Liniendiskretisierung (Linienmethode) | 74 |
| 4.3.1 | Definition | 74 |
| 4.3.2 | Vertikale Linienmethode | 75 |
| 4.3.3 | Horizontale Linienmethode | 76 |
| 4.3.4 | Alternative Methoden | 77 |
| 4.3.4.1 | Vorwärts-Euler-Zeitdiskretisierung | 77 |
| 4.3.4.2 | Leapfrog (Bocksprung) | 78 |
| 4.3.4.3 | Lax-Friedrich | 78 |
| 4.3.5 | Stabilität bzgl. der Linienmethode | 78 |
| 4.4 | Konsistenz und Konvergenz der Methoden | 79 |
| 5 | Lösungsräume partieller Differentialgleichungen | 83 |
| 5.1 | Die Sobolev-Räume | 85 |
| 5.2 | Sobolev-Räume unter Einbeziehung der Zeit: Der $W(0, T)$ | 89 |

| | | |
|--------------|---|----------------|
| 6 | Vorstellung der betrachteten Systeme und ihre Diskretisierung | 95 |
| 6.1 | Die Reaktions-Advektions-Diffusions-Gleichung | 97 |
| 6.2 | Die Wellengleichung | 98 |
| 6.3 | Diskretisierung der Systeme | 99 |
| 6.3.1 | Dirichlet-Randbedingung bei verteilter Steuerung | 100 |
| 6.3.2 | Dirichlet-Randsteuerung | 100 |
| 6.3.3 | Neumann-Randbedingung bei verteilter Steuerung | 102 |
| 6.3.4 | Neumann-Randsteuerung | 104 |
| 7 | Auswertung und numerische Ergebnisse | 107 |
| 7.1 | Allgemeine Voraussetzungen und Bedingungen | 107 |
| 7.2 | Numerische Auswertung der Reaktions-Advektions-Diffusions-Gleichung . . | 111 |
| 7.2.1 | Ausgangssystem und -parameter | 111 |
| 7.2.2 | Dirichlet-Randbedingung | 112 |
| 7.2.2.1 | Verteilte Steuerung mit L^2 -Norm im Zielfunktional | 113 |
| 7.2.2.2 | Verteilte Steuerung mit H^1 -Norm im Zielfunktional | 116 |
| 7.2.2.3 | Randsteuerung mit L^2 -Norm im Zielfunktional | 117 |
| 7.2.2.4 | Randsteuerung mit H^1 -Norm im Zielfunktional | 119 |
| 7.2.2.5 | Änderung der Anfangswerte: Zickzack-Pofil | 123 |
| 7.2.2.6 | Erhöhung der Anzahl der Diskretisierungspunkte | 124 |
| 7.2.3 | Neumann-Randbedingung | 124 |
| 7.2.3.1 | Verteilte Steuerung mit L^2 -Norm im Zielfunktional | 125 |
| 7.2.3.2 | Verteilte Steuerung mit H^1 -Norm im Zielfunktional | 127 |
| 7.2.3.3 | Randsteuerung mit L^2 -Norm im Zielfunktional | 129 |
| 7.2.3.4 | Randsteuerung mit H^1 -Norm im Zielfunktional | 130 |
| 7.2.4 | Zusammenfassung der Ergebnisse | 132 |
| 7.3 | Numerische Auswertung der Wellengleichung | 133 |
| 7.3.1 | Ausgangssystem und -parameter | 134 |
| 7.3.2 | Vergleich der SQP-Verfahren | 137 |
| 7.3.2.1 | Vergleich anhand des zeitlichen Verlaufs | 137 |
| 7.3.2.2 | Vergleich anhand der Kosten | 144 |
| 7.3.2.3 | Vergleich anhand der Rechenzeit | 147 |
| 7.3.2.4 | Zusammenfassung der Ergebnisse | 150 |
| 7.3.3 | Vergleich der beiden Normen H^1 und L^2 | 151 |
| 7.3.3.1 | Dirichlet-Randbedingung mit verteilter Steuerung | 152 |
| 7.3.3.2 | Dirichlet-Randbedingung mit Randsteuerung | 153 |
| 7.3.3.3 | Neumann-Randbedingung mit verteilter Steuerung | 154 |
| 7.3.3.4 | Neumann-Randbedingung mit Randsteuerung | 155 |
| 7.3.3.5 | Zusammenfassung der Ergebnisse | 156 |
| 7.3.4 | Angabe minimaler stabilisierender Horizonte | 157 |
| 7.3.4.1 | Dirichlet-Randbedingung, verteilte Steuerung, L^2 -Norm | 159 |
| 7.3.4.2 | Dirichlet-Randbedingung, verteilte Steuerung, H^1 -Norm | 164 |
| 7.3.4.3 | Dirichlet-Randbedingung, Randsteuerung, L^2 -Norm | 166 |
| 7.3.4.4 | Dirichlet-Randbedingung, Randsteuerung, H^1 -Norm | 170 |
| 7.3.4.5 | Einseitige Neumann-Randbedingung, verteilte Steuerung | 175 |
| 7.3.4.6 | Zweiseitige Neumann-Randbedingung, verteilte Steuerung | 178 |
| 7.3.4.7 | Neumann-Randbedingung, Randsteuerung | 180 |

| | |
|--|------------|
| 7.3.4.8 Zusammenfassung der Ergebnisse | 183 |
| A Anhang | 185 |
| A.1 Dokumentation des Quelltextes für die Wellengleichung | 185 |
| A.2 MATLAB-Codes zur graphischen Darstellung der Resultate | 188 |
| A.3 Inhalt der Daten-CD-ROM | 190 |
| Literaturverzeichnis | 196 |

Tabellenverzeichnis

| | | |
|------|--|-----|
| 2.1 | Beispiele für PDEs aus der Physik | 8 |
| 7.2 | Ausgangsparameter für die Auswertung der ReAdDiff | 111 |
| 7.3 | Ausgangsparameter für die Auswertung der Wellengleichung | 134 |
| 7.4 | Parameterbelegung in Variante 1 | 135 |
| 7.5 | Parameterbelegung in Variante 2 | 135 |
| 7.6 | Parameterbelegung in Variante 3 | 135 |
| 7.7 | Parameterbelegung in Variante 4 | 135 |
| 7.8 | Bereiche der praktischen Stabilität: Dirichlet-RB, DC, L^2 -Norm, Variante 1 | 162 |
| 7.9 | Bereiche der praktischen Stabilität: Dirichlet-RB, BC, L^2 -Norm, Variante 1 | 169 |
| 7.10 | Bereiche der praktischen Stabilität: Dirichlet-RB, BC, H^1 -Norm, Variante 4 | 174 |
| 7.11 | Übersicht der Horizonte bezüglich Stabilisierung | 183 |
| A.1 | Übersicht der bereitgestellten MATLAB-Codes. | 189 |
| A.2 | Inhalt der beiliegenden Daten-CD-ROM. | 190 |

Abbildungsverzeichnis

| | | |
|------|--|-----|
| 1.1 | Grundlegende Bausteine. | 2 |
| 2.1 | Neumann-Randbedingungen | 18 |
| 2.2 | Randsteuerung | 20 |
| 2.3 | Verteilte Steuerung | 21 |
| 3.1 | Stabilisierung eines Stabes mit der Fingerspitze. | 25 |
| 3.2 | Schematische Darstellung von Steuerung und Regelung | 26 |
| 3.3 | Analoges und zugehöriges abgetastetes Signal | 34 |
| 3.4 | Die Lösungen zu Startpunkten innerhalb des Einzugsbereichs verlaufen nicht exakt in das Gleichgewicht x^* , sondern in eine ϵ -Umgebung von x^* | 35 |
| 3.5 | Struktur eines MPC-Verfahrens. | 41 |
| 3.6 | Ablauf der <i>receding horizon strategy</i> | 43 |
| 4.1 | Diskretisierung eines Rechtecks | 59 |
| 4.2 | Diskretisierung eines 2-dimensionalen Gebiets | 59 |
| 4.3 | Augangssystem | 61 |
| 4.4 | Zeitliche Diskretisierung | 62 |
| 4.5 | Augangssystem | 62 |
| 4.6 | Räumliche Diskretisierung | 63 |
| 4.7 | Sekantensteigung | 67 |
| 4.8 | Zentrale Differenzenformel | 68 |
| 4.9 | Normalenableitung | 70 |
| 4.10 | Vertikale Linienmethode | 75 |
| 4.11 | Horizontale Linienmethode | 76 |
| 6.1 | Auslenkung der Massepunkte | 98 |
| 6.2 | Eindimensionale Wellengleichung | 99 |
| 7.1 | Veranschaulichung der verwendeten Startprofile. | 108 |
| 7.2 | Anliegen der unterschiedlichen Steuerungen an den M Diskretisierungspunkten (dargestellt durch \times). | 109 |
| 7.3 | Dirichlet-RB, DC, L^2 -Norm | 114 |
| 7.4 | Modifikation von λ bei Dirichlet-RB mit DC, L^2 -Norm | 115 |
| 7.5 | Dirichlet-RB, DC, H^1 -Norm, $N = 3$, $t_f = 1.0$ | 116 |
| 7.6 | Dirichlet-RB, BC, L^2 -Norm, $N = 3$ und $\lambda = 10^{-5}$ | 118 |
| 7.7 | Dirichlet-RB, BC, L^2 -Norm, $N = 11$ und Endzeit $t_f = 3.5$ | 118 |
| 7.8 | Dirichlet-RB, BC, H^1 -Norm, $N = 5$ | 120 |
| 7.9 | Dirichlet-RB, BC, H^1 -Norm, $N = 17$ und Endzeit $t_f = 1.25$ | 121 |
| 7.10 | Dirichlet-RB, BC, H^1 -Norm, $N = 10$ und $\nu = 0.35$ | 122 |

| | | |
|------|---|-----|
| 7.11 | Dirichlet-RB, BC, H^1 -Norm, $N = 20$ und $\omega = 0.01$ | 122 |
| 7.12 | Dirichlet-RB, BC, L^2 -Norm, $N = 12$, "Zickzack-Profil", Endzeit $t_f = 2$ | 123 |
| 7.13 | Neumann-RB, DC, L^2 -Norm | 126 |
| 7.14 | Neumann-RB, DC, H^1 -Norm, $N = 10$ | 128 |
| 7.15 | Neumann-RB, BC, L^2 -Norm, $N = 11$ | 129 |
| 7.16 | Neumann-RB, BC, H^1 -Norm, $N = 10$ | 131 |
| 7.17 | Simulation des Lösungsverhaltens der Wellengleichung im ungesteuerten Fall | 133 |
| 7.18 | Dirichlet-RB, DC, L^2 -Norm, Variante 1, $N = 2$ | 138 |
| 7.19 | Dirichlet-RB, DC, L^2 -Norm, Variante 2, $N = 5$ | 139 |
| 7.20 | Dirichlet-RB, DC, L^2 -Norm, Variante 3, $N = 10$ | 140 |
| 7.21 | Dirichlet-RB, DC, H^1 -Norm, Variante 1, $N = 2$ | 141 |
| 7.22 | Dirichlet-RB, DC, H^1 -Norm, Variante 2, $N = 5$ | 142 |
| 7.23 | Dirichlet-RB, DC, H^1 -Norm, Variante 3, $N = 10$ | 143 |
| 7.24 | Kostenvergleich bei Dirichlet-RB, DC, L^2 -Norm | 145 |
| 7.25 | Kostenvergleich bei Dirichlet-RB, DC, H^1 -Norm | 146 |
| 7.26 | Rechenzeitvergleich bei Dirichlet-RB, DC, L^2 -Norm | 148 |
| 7.27 | Rechenzeitvergleich bei Dirichlet-RB, DC, H^1 -Norm | 149 |
| 7.28 | Normenvergleich: Dirichlet-RB, DC, $N = 5$, Variante 1 | 152 |
| 7.29 | Normenvergleich: Dirichlet-RB, BC, $N = 20$, Variante 1 | 153 |
| 7.30 | Normenvergleich: Neumann-RB, DC, $N = 10$, Variante 2 | 154 |
| 7.31 | Normenvergleich: Neumann-RB, BC, $N = 20$, Variante 2 | 155 |
| 7.32 | Veranschaulichung des ϵ -Schlauchs. | 157 |
| 7.33 | Dirichlet-RB, DC, L^2 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 1 | 160 |
| 7.34 | Dirichlet-RB, DC, L^2 -Norm, $N = 10$, Variante 1 | 160 |
| 7.35 | Überprüfung auf praktische Stabilität, Dirichlet-RB, DC, L^2 -Norm, Variante 1 mit vergrößerter Endzeit $t_f = 120$ | 161 |
| 7.36 | Dirichlet-RB, DC, L^2 -Norm | 163 |
| 7.37 | Dirichlet-RB, DC, H^1 -Norm | 165 |
| 7.38 | Vergleich der Variante 1 ($N = 3$) mit Variante 2 ($N = 2$): Dirichlet-RB, DC, H^1 -Norm | 165 |
| 7.39 | Dirichlet-RB, BC, L^2 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 1 | 167 |
| 7.40 | Dirichlet-RB, BC, L^2 -Norm, $t_f = 120$ | 167 |
| 7.41 | Dirichlet-RB, BC, L^2 -Norm | 168 |
| 7.42 | Dirichlet-RB, BC, L^2 -Norm: Vergleich der betragsmäßig gemittelten Abweichungen von der 0. | 169 |
| 7.43 | Dirichlet-RB, BC, H^1 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 1 | 171 |
| 7.44 | Dirichlet-RB, BC, H^1 -Norm, Variante 1 | 171 |
| 7.45 | Dirichlet-RB, BC, H^1 -Norm, $N = 10$ | 172 |
| 7.46 | Dirichlet-RB, BC, H^1 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 4 | 173 |
| 7.47 | Dirichlet-RB, BC, H^1 -Norm, $N = 20$, Variante 4 | 174 |
| 7.48 | Neumann-RB, DC, $N = 10$, L^2 -Norm | 176 |
| 7.49 | Neumann-RB einseitig, DC, H^1 -Norm, Variante 1 | 176 |
| 7.50 | Neumann-RB einseitig, DC, H^1 -Norm, Variante 2 | 177 |
| 7.51 | Neumann-RB zweiseitig, DC, H^1 -Norm, Variante 1 | 179 |
| 7.52 | Neumann-RB zweiseitig, DC, H^1 -Norm, Variante 2 | 179 |
| 7.53 | Neumann-RB, BC, L^2 -Norm | 181 |
| 7.54 | Neumann-RB, BC, H^1 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 1 | 182 |

7.55 Neumann-RB, BC, H^1 -Norm, $N = 20$ 182

1 Einleitung

1.1 Motivation und inhaltlicher Aufbau

In dieser Arbeit beschäftigen wir uns mit der *modellprädiktiven Regelung nichtlinearer evolutionärer Systeme*, die in Form von *partiellen Differentialgleichungen* beschrieben werden. Ein “evolutionäres System” charakterisiert sich dabei durch die Eigenschaft, dass es sich im zeitlichen Verlauf weiterentwickelt (lat. *evolvere*) und dadurch kontinuierlichen Veränderungen unterliegt. Diese Systeme lassen sich in fast allen Bereichen des alltäglichen Lebens in Form von Abläufen, Vorgängen oder Prozessen (z.B. zur Beschreibung von Wachstumsprozessen in Natur und Wirtschaft oder von chemischen und physikalischen Vorgängen) wiederfinden. Die partiellen Differentialgleichungen stellen dabei ein probates Mittel dar, um diese Systeme möglichst genau zu modellieren und mathematisch zu formulieren. Will man auf diese Systeme Einfluss nehmen, zum Beispiel in Form von *Kontrollen*, so erweist sich hierfür die *modellprädiktive Regelung* als eine bewährte Methode.

Ziel dieser Arbeit ist eine grundlegende theoretische Einführung in die genannten einzelnen Bereiche, sowie die praktische Anwendung eines modellprädiktiven Regelungsansatzes auf zwei nichtlineare evolutionäre Systeme: Zum einen auf die parabolische *Reaktions-Advektions-Diffusions-Gleichung* und zum anderen auf die hyperbolische *Wellengleichung*. Dies geschieht in Form zweier Auswertungen, in denen versucht wird, mittels der modellprädiktiven Regelung das jeweilige System unter verschiedenen *Randbedingungen*, *Steuerungsarten* und *Systemmodifikationen* in ein *Null-Profil* zu stabilisieren. Dazu wurde von JÜRGEN PANNEK ein Programm¹ in Form einer C++-Implementierung der modellprädiktiven Regelung mit *1-Schritt-Regler* zur Verfügung gestellt.

Um die Auswertungen durchzuführen, kommt folgendes Vorgehen zur Anwendung:

1. Das auszuwertende System liegt in Form einer partiellen Differentialgleichung vor.
2. Die partielle Differentialgleichung wird mittels *Finite-Differenzen-Methode* diskretisiert und dadurch in ein *System gewöhnlicher Differentialgleichungen* überführt.
3. Das erhaltene System gewöhnlicher Differentialgleichungen wird unter Berücksichtigung der verschiedenen Randbedingungen und Steuerungsarten in das Programm implementiert.
4. Die Auswertung erfolgt unter dem Aspekt der Auswirkungen verschiedener Modifikationen und Variationen des Systems auf dessen Stabilisierungsverhalten.

¹Das Programm lag zur Zeit der Verwendung in einer Beta-Version vor und wurde auch währenddessen kontinuierlich weiterentwickelt und verbessert. Für diesbezügliche Fragen möchten wir an dieser Stelle auf den Autor verweisen. Email: juergen.pannek@uni-bayreuth.de, Homepage: <http://num.math.uni-bayreuth.de/~jpannek/>.

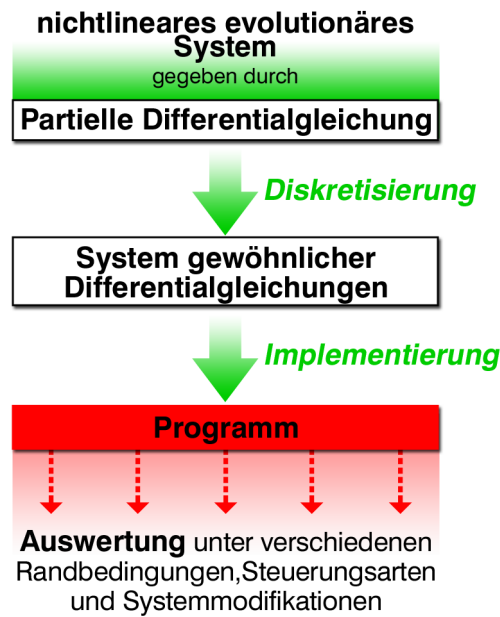


Abbildung 1.1: Grundlegende Bausteine.

Das verwendete Programm berechnet durch Lösen eines *nichtlinearen modellprädiktiven Kontrollproblems* eine stückweise konstante Steuerfolge, um ein gewünschtes Langzeitverhalten des Systems (hier: Stabilisierung in die Null) zu erreichen. Dieses Kontrollproblem besteht aus einer Anzahl von *Optimalsteuerungsproblemen*. In diesem Zusammenhang spielt die Beschaffenheit der *Zielfunktionale* eine wichtige Rolle in Bezug auf die angestrebte Stabilisierung. Dazu wird diesbezüglich die *Kostenfunktion*, die das jeweilige Zielfunktional definiert, aufgrund der darin verwendeten *Normen* unterschieden und das sich daraus ergebende Stabilisierungsverhalten untersucht.

Die Arbeit gliedert sich in sechs Hauptkapitel, von denen die ersten vier (Kapitel 2 bis 5) die theoretischen Grundlagen und Einführungen in die eingangs erwähnten Bereiche beinhalten. Das sechste Kapitel bildet anschließend die Schnittstelle zum praxisbezogenen Teil dieser Arbeit, der im siebten Kapitel in Form der Auswertungen behandelt wird.

Im zweiten Kapitel wird auf die in vielen Bereichen gegebene Notwendigkeit der partiellen Differentialgleichungen eingegangen und ihre algebraischen Eigenschaften und verschiedenen Klassifikationen definiert. Außerdem werden die in der Auswertung verwendeten Randbedingungen und Steuermöglichkeiten genau betrachtet. Die darin formulierten Definitionen sind an die Arbeiten von G. BÄRWOLFF [5], C.-D. MUNZ und T. WESTERMANN [39] und C. BÜSKENS [8] angelehnt. Ein Beispiel für die Steuerungsarten wird aus dem Werk von F. TRÖLTZSCH [52] entnommen.

In Kapitel 3 wird zunächst eine allgemeine Einführung in die modellprädiktive Regelung sowie Einblicke in die mathematische Kontrolltheorie gegeben. Anschließend wird der Begriff eines Kontrollsystems konkret gefasst und der Übergang von zeitkontinuierlichen zu zeitdiskreten Systemen mittels des Prinzips der Abtastung formuliert. Diesen Abschnitten liegen die Arbeiten von L. GRÜNE [25, 26] zugrunde. Im nachfolgenden Abschnitt wird das Problem der optimalen Steuerung und zugehörige Klassen von Lösungsansätzen nach F. TRÖLTZSCH [51], R. VINTER [53] und C. BÜSKENS [9] vorgestellt. Nach einer Einführung in die Methodik und Strategie des modellprädiktiven Regelungsverfahrens wird

das nichtlineare modellprädiktive Regelungsproblem formuliert. Dabei orientiert sich das Kapitel an den Arbeiten von R. FINDEISEN [16], E. F. CAMACHO und C. BORDONS [10] sowie an J. A. ROSSITER [43]. Abschließend wird eine Analyse der MPC-Schemata bezüglich der Stabilität in Anlehnung an die Arbeit von L. GRÜNE [24] durchgeführt, mit der Intention, Richtlinien für die Wahl einer geeigneten Kostenfunktion zu erhalten.

Das vierte Kapitel beschäftigt sich mit Methoden, die zur Überführung der partiellen Differentialgleichungen in Systeme gewöhnlicher Differentialgleichungen dienen. Dazu werden zu Beginn verschiedene dimensionale Gitterdiskretisierungen aus dem Werk von C.-D. MUNZ und T. WESTERMANN [39] eingeführt, um dann die Semidiskretisierungen sowohl räumlich als auch zeitlich anhand eines Beispiels genauer zu betrachten. Ein wichtiges Hilfsmittel zur Lösungsfindung hierbei ist zum einen die Finite-Differenzen-Methode, die Ableitungen entsprechend ihrer Ordnung approximiert, und zum anderen die Linienmethode, die sich wiederum in die vertikale und die horizontale Linienmethode untergliedern lässt. Hierzu liegen vornehmlich die Werke von C. BÜSKENS [8] und H. P. LANGTANGEN [34] zugrunde. Außerdem wird die Upwind-Methode vorgestellt, die Stabilität der einzelnen Methoden kurz untersucht und alternative Methoden angegeben. Hierfür sei hauptsächlich auf die Arbeiten von C.-D. MUNZ und T. WESTERMANN [39] und R. J. LEVEQUE [35] verwiesen.

Im fünften Kapitel wird ein kurzer Einblick in die die funktionalanalytischen Grundlagen der Lösungsräume partieller Differentialgleichungen gegeben. Hauptaugenmerk liegt hierbei auf der Einführung der in den Kostenfunktionen verwendeten Normen. Dieses Kapitel basiert vornehmlich auf den Arbeiten von F. TRÖLTZSCH [51], L. C. EVANS [15] und K. THEISSEN [48].

In Kapitel 6 werden die der Auswertung zugrunde liegenden Systeme in Form der Reaktions-Advektions-Diffusions-Gleichung und der Wellengleichung vorgestellt und anschließend mittels der in Kapitel 4 eingeführten Finite-Differenzen-Methode diskretisiert. Hierbei wird bezüglich der in Kapitel 2 betrachteten Randbedingungen und der Arten der anliegenden Steuerungen unterschieden, so dass sich je nach Bedingung an den Rand und je nach Art der Steuerung eine unterschiedliche Diskretisierung des Systems ergibt.

Den Abschluss dieser Arbeit bildet Kapitel 7. Darin werden verschiedene Auswertungen der beiden Systeme unter den Gesichtspunkten der verwendeten Randbedingungen, Steuerungsarten, Normen und Systemmodifikationen vorgenommen und bezüglich ihrer Auswirkungen auf die Stabilisierung und den dazugehörigen minimalen Stabilisierungshorizont analysiert.

Anhang A enthält neben einer kurzen Dokumentation des zur Auswertung der Wellengleichung verwendeten Quelltextes `wellengleichung.cpp` eine Beschreibung der zur graphischen Veranschaulichung beigefügten MATLAB-Codes, sowie eine Inhaltsangabe der beiliegenden Daten-CD-ROM.

1.2 Anmerkung zur Konzeption der Arbeit

Das zugrundeliegende Thema *Modellprädiktive Regelung für nichtlineare evolutionäre partielle Differentialgleichungen* wurde zu einer umfassenden Behandlung in die zwei Bereiche *Numerik* und *Theorie* aufgeteilt, die jeweils in einer eigenständigen Diplomarbeit von Annika Grötsch beziehungsweise Stefan Trenz bearbeitet wurden: Die die Numerik be-

1 *Einleitung*

treffenden Kapitel 2, 4 und 6 wurden von Annika Grötsch, die die Theorie betreffenden Kapitel 3 und 5 sowie die theoretischen Grundlagen in Kapitel 6 von Stefan Trenz verfasst. Sämtliche Auswertungen sowie die dafür benötigten Implementationen in Kapitel 7 wurden gemeinsam durchgeführt, die daraus erhaltenen Resultate gemeinsam diskutiert und präsentiert. Die zur graphischen Veranschaulichung der Ergebnisse verwendeten MATLAB-Codes wurden von Stefan Trenz geschrieben.

Zur besseren Lesbarkeit wurden die beiden Diplomarbeiten zu der hier vorliegenden Arbeit zusammengefasst.

2 Partielle Differentialgleichungen

2.1 Motivation

Im ersten Kapitel dieser Arbeit werden die partiellen Differentialgleichungen, ihre genaue Definition, ihre Klassifikationen und verschiedene Modifikationen wie Anfangsbedingungen, Randbedingungen und Kontrollmöglichkeiten genauer betrachtet.

Zunächst stellt sich dabei folgende Frage:

„Was ist eine partielle Differentialgleichung?“

Dazu wollen wir den Begriff in Anlehnung an die Dissertation von J. M. CHARTON [11] in seine Bestandteile zerlegen:

- **Gleichung:**

Eine *Gleichung* setzt mathematische Größen (bekannt oder unbekannt, abhängig oder unabhängig) zueinander in Beziehung, wobei jeweils zwei Terme durch ein Gleichheitszeichen verknüpft werden, z.B.

$$x_1 + 2 = 5 + x_2, \text{ mit } \mathbb{D} = \mathbb{R}.$$

Dabei liefert eine Auswertung beider Seiten dasselbe Element der zugrundeliegenden Definitionsmenge.

- **Differentialgleichung:**

Eine *Differentialgleichung* ist eine Erweiterung der allgemeinen Gleichung, so dass neben den Funktionswerten auch Ableitungen einer gesuchten Funktion auftreten:

$$x_1' = x_2(x_1 + 2), \text{ mit } \mathbb{D} = \mathbb{R}.$$

- **Partielle Differentialgleichung:**

Bei *partiellen Differentialgleichungen* ist die Lösung im Gegensatz zu den gewöhnlichen Differentialgleichungen, bei denen die gesuchte Lösung nur von einer Variable abhängt, entweder von mehreren Raumdimensionen oder von einer Raumdimension und der Zeit oder von mehreren Raumdimensionen und der Zeit abhängig. Das heißt, dass die Variable y von mindestens zwei unabhängigen Variablen abhängt, wie z.B.

$$\frac{\partial y(x, t)}{\partial x} = \frac{\partial y(x, t)}{\partial t}.$$

2 Partielle Differentialgleichungen

In vielen Bereichen der Ingenieur-, Natur-, Sozial- und Wirtschaftswissenschaften, aber auch der Medizin und Technik sind partielle Differentialgleichungen unverzichtbar, da die Vorgänge in den einzelnen Bereichen nicht durch eine Koordinate darstellbar sind und somit die gewöhnlichen Differentialgleichungen nicht mehr ausreichen, diese genügend genau zu beschreiben. Beispiele, bei denen die partiellen Differentialgleichungen ihre Anwendung finden (siehe [5, 8]), sind:

- Wettervorhersage
- Halbleiterbauelemente
- Strömungsmechanik
- Portfoliomodelle
- Chemotherapeutische Vorgänge
- Chemische Reaktionen in der Verfahrenstechnik
- Crash-Simulationen
- Gerichtsmedizin (z.B. zur Bestimmung des Todeszeitpunktes bei Verbrechen)

Diese vielseitige Funktionalität zeigt, dass die partiellen Differentialgleichungen ein äußerst wichtiges Anwendungsgebiet darstellen und eines der herausforderndsten Forschungsgebiete in der Mathematik sind. In einigen wenigen Fällen lassen sich die Probleme analytisch lösen, meist ist man aber auf numerische Lösungsmethoden angewiesen (vgl. Kapitel 4). Im Vergleich zu den gewöhnlichen Differentialgleichungen existiert jedoch kein allgemein gültiges Verfahren zur numerischen Lösung.

Jede PDE stellt ein eigenes Forschungsgebiet dar.

Die unterschiedlichen Verfahren zielen zudem darauf ab, nicht irgendeine Lösung zu finden, sondern eine, die außerdem noch bestimmte Zusatzbedingungen erfüllen muss. Dies können zum Beispiel Vorgaben auf den Rändern des Gebietes (Randbedingungen) oder Vorgaben für den Anfangszeitpunkt (Anfangsbedingungen) sein (vgl. [5]).

Wir werden uns in dieser Arbeit vornehmlich mit Differentialgleichungen zweiter Ordnung beschäftigen, wobei die Ordnung immer durch den Grad der höchsten Ableitung in der Gleichung gegeben ist. Bei der Analyse von Systemen interessiert es uns, wie sich die Eigenschaften des Systems bei laufender Zeit t verändern, wie zum Beispiel Geschwindigkeit oder Beschleunigung. Außerdem interessiert uns die räumlichen Verteilung der Systemeigenschaften, wie zum Beispiel Temperaturschleppungsgeschwindigkeit. Deshalb betrachten wir als Veränderliche in unseren Systemen nur die Zeit t und die Raumvariable x . Die allgemeinste Form einer partiellen Differentialgleichung zweiter Ordnung im \mathbb{R}^n sieht wie folgt aus (vgl. [39]):

$$\boxed{F(x, t, y, y_x, y_t, y_{xt}, y_{xx}, y_{tt}) = 0} \quad (2.1)$$

mit folgender Kurzschreibweise:

$$y_x = \frac{\partial y}{\partial x}, \quad y_t = \frac{\partial y}{\partial t}, \quad y_{xt} = \frac{\partial^2 y}{\partial x \partial t}, \quad y_{xx} = \frac{\partial^2 y}{\partial x^2}, \quad y_{tt} = \frac{\partial^2 y}{\partial t^2} \quad (2.2)$$

Im Folgenden werden wir für den Begriff *partielle Differentialgleichung* die englische Abkürzung PDE (*partial differential equation*) verwenden.

2.2 Grundbegriffe und Definitionen

2.2.1 Gewöhnliche und partielle Differentialgleichungen

Zuerst wollen wir in diesem Abschnitt den Begriff der gewöhnlichen Differentialgleichungen definieren, um dann zu den partiellen Differentialgleichungen überzugehen (dazu vgl. F. TRÖLTZSCH [52]).

Wie bereits erwähnt, kommen bei **gewöhnlichen** Differentialgleichungen nur Ableitungen nach einer Variable vor.

Definition 2.1 (Gewöhnliche Differentialgleichung)

Gegeben sei eine Funktion

$$F(x, z_0, \dots, z_m) : (a, b) \times \mathbb{R}^{m+1} \rightarrow \mathbb{R}$$

und ein betrachtetes Gebiet $\Omega = (a, b)$ mit $a, b \in \mathbb{R}$.

Gesucht ist eine Funktion $y = y(x)$ einer reellen Veränderlichen $x \in (a, b)$, sodass die Gleichung

$$F(x, y(x), y'(x), \dots, y^{(m)}(x)) = 0, \quad x \in (a, b) \quad (2.3)$$

erfüllt ist.

Sie wird *gewöhnliche Differentialgleichung m-ter Ordnung* genannt.

Dies gilt nur, wenn $m \geq 1$ gilt und $y^{(m)}$ auftritt. \square

Bei **partiellen** Differentialgleichungen dagegen suchen wir eine Funktion von mehreren Veränderlichen

$$y = y(x_1, \dots, x_n), \quad y : \Omega \rightarrow \mathbb{R}, \quad \Omega \subset \mathbb{R}^n.$$

Für die Definition fassen wir die Variablen x_1, \dots, x_n zu einem Vektor x zusammen, mit $x \in \Omega$. Außerdem treten jetzt partielle Ableitungen auf, d.h. Ableitungen nach einem oder mehreren x_i mit $i \in \{1, \dots, n\}$, wie zum Beispiel

$$\frac{\partial y}{\partial x_1}, \quad \text{oder} \quad \frac{\partial^2 y}{\partial x_2 \partial x_3}$$

(vgl. für diese Schreibweise (2.2) und für eine genaue Definition der partiellen Ableitungen siehe [17]).

2 Partielle Differentialgleichungen

Definition 2.2 (Partielle Differentialgleichung)

Gegeben sei ein Gebiet $\Omega \subset \mathbb{R}^n$ und eine Funktion

$$F(x, z_0, \dots, z_k) : \Omega \times \mathbb{R}^{k+1} \rightarrow \mathbb{R}.$$

Eine Gleichung der Form

$$F(x, y(x), D_x^{\alpha_1} y(x), \dots, D_x^{\alpha_k} y(x)) = 0, \quad x \in \Omega \quad (2.4)$$

heißt *partielle Differentialgleichung in n reellen Veränderlichen*.

Dabei zeigt der Differentialoperator $D_x^\alpha = (D_{x_1}^{\alpha_1}, \dots, D_{x_n}^{\alpha_n})$ an, wie oft nach x_i partiell abgeleitet wird. Außerdem sind $\alpha_1, \dots, \alpha_n$ Multiindizes, also jeweils Vektoren der Dimension n .

Wir sagen PDE m -ter Ordnung, wenn

$$|\alpha_i| \leq m, \quad i = 1, \dots, k, \quad \text{und} \quad \exists \alpha_j : |\alpha_j| = m$$

gilt.

Die Definition gilt nur, wenn $D_x^{\alpha_j} y$ auftritt. □

Stehen zusätzlich noch Anfangswerte zur Verfügung, so sprechen wir von einem *Anfangswertproblem*, analog von einem *Randwertproblem*.

Weitere Definitionen bezüglich der PDEs sind im Skript von C. BÜSKENS [8] und der Arbeit von D. J. EVANS [14] zu finden. In dieser Allgemeinheit ist das Problem, eine Lösung für eine PDE zu finden, meist unlösbar, da unklar ist, ob überhaupt eine Lösung existiert, und wenn, ob diese Lösung eindeutig ist. Es gibt aber auch relativ einfache PDEs aus der Physik. Die nachfolgende Tabelle zeigt einige Beispiele aus der Dissertation von J. M. CHARTON [11]:

| PDE | Bezeichnung |
|-----------------------------|-------------------------|
| $y_t + cy_x = 0$ | Einfacher Transport |
| $y_t + yy_x = 0$ | Stoßwelle |
| $y_{tt} - y_{xx} - y^3 = 0$ | Welle mit Rückkoppelung |
| $y_t + yy_x + y_{xxx} = 0$ | Dispersionswelle |
| $y_{tt} + y_{xxxx} = 0$ | Schwingender Stab |
| $y_t - iy_{xx} = 0$ | Quantenmechanik |

Tabelle 2.1: Beispiele für PDEs aus der Physik

2.2.2 Algebraische Eigenschaften

Eine Möglichkeit, PDEs in Klassen einzuteilen, liegt in der Aufteilung nach algebraischen Eigenschaften. In der nächsten Definition, die in Anlehnung an F. TRÖLTZSCH [52] gewählt wurde, schreiben wir zur Vereinfachung nur D^α anstelle von D_x^α .

Definition 2.3 (Algebraische Eigenschaften von PDEs)

1. Seien die Funktionen a_α und f gegeben. Dann heißt eine PDE der Form

$$\sum_{|\alpha| \leq k} a_\alpha(x) D^\alpha y = f(x) \tag{2.5}$$

linear.

2. Seien die Funktionen a_α und \tilde{F} gegeben. Wenn alle in \tilde{F} stehenden Ableitungen die Ordnung kleiner als k besitzen, heißt eine PDE k -ter Ordnung der Form

$$\sum_{|\alpha|=k} a_\alpha(x) D^\alpha y + \tilde{F}(x, D^{\alpha^1} y, \dots, D^{\alpha^l} y) = 0 \tag{2.6}$$

semilinear.

3. Ein weiterer Typ ist die *quasilineare* PDE, in der nur die höchsten Ableitungen linear vorkommen. Seien dazu wieder die Funktionen a_α und \tilde{F} gegeben. Eine PDE hat somit folgende Form:

$$\sum_{|\alpha|=k} a_\alpha(x, D^{\beta_1 y}, \dots, D^{\beta_p y}) D^\alpha y + \tilde{F}(x, D^{\alpha^1} y, \dots, D^{\alpha^l} y) = 0 \tag{2.7}$$

mit $|\alpha_1|, \dots, |\alpha_l|, |\beta_1|, \dots, |\beta_p| \leq k - 1$

4. In allen anderen Fällen heißt die PDE *nichtlinear*. □

Bemerkung 2.4

Sowohl der Differentialoperator als auch der Multiindex gelten wie in Definition 2.2.

Mit $|\alpha| = \alpha_1 + \dots + \alpha_n$ wird die Differentiationsordnung angegeben. ○

Wir wollen zu jeder algebraischen Eigenschaft noch ein kurzes Beispiel angeben:

Beispiel 2.5

1. Lineare PDE zweiter Ordnung:

$$x_1^3 D_2 D_3 y + D_1 y = 0$$

2. Semilineare PDE zweiter Ordnung:

$$x_1^3 D_2 D_3 y + (D_1 y)^2 = 0$$

3. Quasilineare PDE zweiter Ordnung:

$$x_1^3 (D_2 D_3 y) (D_1 y)^2 = 0$$

4. Nichtlineare PDE zweiter Ordnung:

$$(D_1 y)^2 + (D_2 y)^2 = 0 \tag{•}$$

Weitere analoge Definitionen mit detaillierten Beispielen sind in der Arbeit von G. BÄRWOLFF [5] und C. GROSSMANN und H.-G. ROSS [23] zu finden. Bisher haben wir die Definition von PDEs und ihre jeweiligen algebraischen Eigenschaften kennengelernt.

Es stellt sich die Frage, ob unser vorliegendes Problem überhaupt „gut gestellt“ ist. Dies werden wir im nächsten Abschnitt in Anlehnung an C. BÜSKENS [8] und H. P. LANGTANGEN [34] betrachten.

2.2.3 Ein gut gestelltes (well-posed) Problem

Definition 2.6 (well-posed problem)

Eine PDE mit Nebenbedingungen, in unserem Fall Anfangs- und Randbedingungen¹ heißt *gut* oder *sachgemäß gestellt*, falls folgende drei Bedingungen erfüllt sind:

1. Eine Lösung existiert.
2. Die Lösung ist eindeutig.
3. Die Lösung hängt stetig von ihren Nebenbedingungen ab.

Sonst nennen wir das Problem *schlecht gestellt*. □

Bemerkung 2.7

Die dritte Bedingung bedeutet, dass kleine Veränderungen in den Nebenbedingungen oder in den Koeffizienten der PDE nur zu kleinen Veränderungen in der Lösung führen dürfen. ○

Die zugrundeliegenden Probleme müssen offensichtlich „gut gestellt“ sein, um numerische Lösungen zu bestimmen. Außerdem können bei Nichteinhaltung der 3. Bedingung auf Grund der endlichen Genauigkeit in den arithmetischen Operationen bereits kleinste Rundungsfehler die Lösung grundlegend verändern.

Wenn wir, wie im späteren Verlauf gezeigt, ein „gut gestelltes“ Problem mittels numerischer Methoden diskretisieren wollen, müssen wir sicherstellen, dass auch das diskrete Problem „gut gestellt“ ist. Das bedeutet, dass

1. die diskrete Lösung existiert,
2. die diskrete Lösung eindeutig ist und
3. die diskrete Lösung stetig von der approximierten Darstellung der Nebenbedingungen abhängt.

Auch aus diesem Grund suchen wir eine diskrete Lösung, die nahe an der Lösung des kontinuierlichen Problems ist.

¹Auf diese gehen wir später noch genau ein (vgl. dazu Abschnitt 2.5)

2.3 Klassifikation partieller Differentialgleichungen

Nachdem wir bereits die Aufteilung von PDEs nach algebraischen Eigenschaften betrachtet haben, wollen wir jetzt eine sehr allgemein gültige Klassifikation definieren und spezifische Beispiele dazu betrachten. Diese Klassifikation findet sich in den meisten Arbeiten über PDEs, wir haben uns hier an den Werken von C.-D. MUNZ und T. WESTERMANN [39], D. J. EVANS [14] und R. J. LEVEQUE [35] orientiert. Zwar fallen nicht alle PDEs in eine dieser Klassen, aber viele Gleichungen in der Praxis tun dies. Diese Klassen von Gleichungen beschreiben verschiedene Arten von Vorgängen, zeigen unterschiedliches Verhalten und benötigen verschiedene numerische Methoden für ihre Lösung.

Zur Klassifikation betrachten wir die semilinearen partiellen Differentialgleichungen zweiter Ordnung mit konstanten Koeffizienten für unsere Funktion $y(x, t)$

$$A(x, t)y_{xx} + B(x, t)y_{xt} + C(x, t)y_{tt} = F(x, t, y, y_x, y_t) \quad (2.8)$$

mit gegebenen Funktionen A, B, C und F , die von der Raumvariable x und der Zeit t abhängen. Dabei wird vorausgesetzt, dass die Lösung mindestens 2-mal stetig differenzierbar ist.

Definition 2.8 (Klassifikation)

Die Klassifikation ist abhängig vom Vorzeichen der Diskriminante. Eine PDE heißt also

1. *elliptisch*, wenn $B^2 - 4AC < 0 \quad \forall(x, t)$,
2. *parabolisch*, wenn $B^2 - 4AC = 0 \quad \forall(x, t)$ und
3. *hyperbolisch*, wenn $B^2 - 4AC > 0 \quad \forall(x, t)$. □

Bemerkung 2.9

- Bei Umformulierung des mathematischen Modells oder bei Wahl eines anderen Koordinatensystems ändert sich der Typ der Klassifikation nicht. Es wird auch nach Modifikation noch das gleiche Problem beschrieben.
- Zur Bestimmung des Typs sind nur Koeffizienten der höchsten Ableitung in der PDE ausschlaggebend. Das heißt, dass im Beispiel einer PDE zweiter Ordnung die rechte Seite nur von x, t, y selbst, y_x und y_t , also höchstens von ersten Ableitungen abhängen darf.
- Diese Definition ist nur, wie in (2.8), für PDEs mit konstanten Koeffizienten anwendbar. Ist dies nicht der Fall, können wir den Typ nur in Abhängigkeit von x und t lokal in einem Punkt (\bar{x}, \bar{t}) bestimmen, analog zu den Bedingungen in Definition 2.8. ○

Wir betrachten nun die einzelnen Klassen der PDEs anhand ihrer Anwendungsgebiete und spezifischer Beispiele:

Beispiel 2.10

1. Die elliptischen PDEs beschreiben stationäre, d.h. zeitunabhängige, Gleichgewichtszustände in physikalischen Bereichen, wie z.B. Festigkeitslehre, Magnetostatik, Strömungsmechanik, Thermodynamik und Wärmeverteilung.

Ein typisches Beispiel mit $A = C = 1$ und $B = 0$ ist die **Poisson-Gleichung**:

$$\Delta y = h$$

mit $h = h(x_1, x_2, y, y_{x_1}, y_{x_2})$ und dem Laplace-Operator $\Delta y = y_{x_1 x_1} + y_{x_2 x_2}$, die ihre Anwendung in der Elektrostatik oder auch in der Berechnung von gravitativen Potentialen findet.

2. In der Realität und deswegen auch in unseren späteren Auswertungen (siehe Kapitel 7) kommen zeitunabhängige Probleme relativ selten vor, weswegen die Zeit t bei den parabolischen PDEs als weiterer Parameter hinzukommt. Diese nennen wir auch instationäre PDEs. Es werden dadurch zum Beispiel Dissipation² und Reibungseinflüsse beschrieben. Als Standardbeispiel in dieser Klasse dient die **Wärmeleitungsgleichung** mit $A = \kappa > 0$ und $C = 0$. In einer Raumdimension hat sie die Form

$$y_t = \kappa y_{xx}$$

und in mehreren Raumdimensionen

$$y_t = \kappa \Delta y$$

mit der Temperaturleitzahl $\kappa > 0$. Für eine eindeutige Lösung sind Anfangs- und Randbedingungen notwendig, was letztendlich zu einem Problem der instationären Wärmeleitung führt, einem so genannten Anfangs-Randwert-Problem.

Lässt man die Zeit t gegen unendlich laufen, erhalten wir wieder die oben genannte Poisson-Gleichung, die somit eine Wärmeleitung unabhängig von der Zeit t darstellt, eine so genannte stationäre Wärmeleitungsgleichung.

3. Mit den hyperbolischen PDEs werden vor allem Wellen-, Transport- und Ausbreitungsvorgänge dargestellt. Auch in diesem Fall finden wir rein instabile Vorgänge. Bei dieser Klasse sind Anfangsbedingungen für die Existenz einer eindeutigen Lösung unverzichtbar. Im Vergleich zu den parabolischen PDEs benötigt man neben dem Funktionswert auch dessen Ableitung nach der Zeit t . Natürlich kann auch hier das physikalische Problem ein Anfangs-Randwert-Problem sein, wodurch am Rand noch Verträglichkeitsbedingungen³ auftreten. In einer Raumdimension hat die **Wel-**

²Zerstreuung

³Physikalische Probleme treten meist in endlichen Gebieten auf, so dass Randwerte nicht beliebig vorgeschrieben werden können. Dies bedeutet, dass die Randwerte mit der Wellenausbreitung verträglich sein müssen.

lengleichung die Form

$$y_{tt} - c^2 y_{xx} = 0$$

und in mehreren Raumdimensionen

$$y_{tt} - c^2 \Delta y = 0$$

mit der Wellengeschwindigkeit $c > 0$. •

Wir wollen zum Schluss die wichtigsten Punkte dieses Abschnitts noch einmal zusammenfassen:

- Die verschiedenen Typen der PDEs beschreiben unterschiedliche physikalische Vorgänge.
- Die Lösungen haben verschiedene Eigenschaften.
- Kenntnis der Klassifizierung ist wesentlich für die Auswahl der zu verwendenden numerischen Approximation.
- Klassifizierung gibt Hinweise auf die vorzugebenden Anfangs- und Randbedingungen für ein gut gestelltes Problem (siehe Definition 2.6).

2.4 Transformation von Differentialgleichungssystemen höherer Ordnung

Da wir meist PDEs zweiter Ordnung gegeben haben, müssen wir, um das für unsere Auswertung zur Verfügung gestellte Programm von JÜRGEN PANNEK zu verwenden, die PDE zunächst in ein System von gewöhnlichen Differentialgleichungen umformen. Dies geschieht mittels Diskretisierung (siehe Kapitel 4). Zuvor führen wir noch eine Reduktion der Ordnung durch, um die PDE zweiter Ordnung in ein Differentialgleichungssystem erster Ordnung zu transformieren und somit die Implementation letztendlich durchführen zu können. Außerdem können wir uns durch diese Reduktion auf die Betrachtung von Differentialgleichungssysteme erster Ordnung beschränken. Dies lässt sich analog zu den gewöhnlichen Differentialgleichungen durchführen, wobei die Typeneinteilung nicht verloren geht (vgl. [17]).

2 Partielle Differentialgleichungen

Definition 2.11 (Reduktion auf ein System erster Ordnung)

Um eine Differentialgleichung n -ter Ordnung auf ein System von Differentialgleichungen erster Ordnung zu reduzieren, sei eine Differentialgleichung

$$y^n = f(x, y, y', \dots, y^{(n-1)})$$

und eine stetige Funktion f , die auf einer Teilmenge $G \subset \mathbb{R}^{n+1}$ definiert ist, gegeben. Dann ist

$$y'_0 = y_1$$

$$y'_1 = y_2$$

...

$$y'_{n-2} = y_{n-1}$$

$$y'_{n-1} = f(x, y_0, y_1, \dots, y_{n-1})$$

das auf ein System von Differentialgleichungen erster Ordnung transformierte System. \square

Bemerkung 2.12

- Die Lösungen des nicht-reduzierten und des reduzierten Differentialgleichungssystems stehen dabei in eindeutiger Beziehung zueinander.
- Um eine Lösung für ein Differentialgleichungssystem höherer Ordnung zu finden, ist es ausreichend, das auf erste Ordnung reduzierte Differentialgleichungssystem zu lösen. \circ

Wir wollen dies anhand der Poisson-Gleichung und der Wellengleichung (nach [8]) darstellen:

Beispiel 2.13 (Poisson-Gleichung)

Die Poisson-Gleichung lautet:

$$y_{x_1x_1} + y_{x_2x_2} = f(x_1, x_2)$$

Wir setzen

$$v := y_{x_1}, \quad w := y_{x_2} \quad \Rightarrow \quad v_{x_2} = y_{x_1x_2} = y_{x_2x_1} = w_{x_1},$$

erhalten folgende Gleichung

$$y_{x_1x_1} + y_{x_2x_2} = v_{x_1} + w_{x_2}$$

und somit das umgeformte System

$$v_{x_1} + w_{x_2} = f(x_1, x_2)$$

$$v_{x_2} - w_{x_1} = 0. \quad \bullet$$

Es kann auch nur ein Teil der höheren Ableitungen in ein System erster Ableitungen umgeschrieben werden. Anhand der Wellengleichung soll dies dargestellt werden:

Beispiel 2.14 (Wellengleichung)

Die Wellengleichung ist gegeben durch

$$y_{tt} - c^2 y_{xx} = 0$$

Wir setzen

$$v := y_t.$$

Dadurch erhalten wir folgendes transformiertes System:

$$y_t - v = 0$$

$$v_t - c^2 y_{xx} = 0. \quad \bullet$$

2.5 Anfangs- und Randbedingungen

Da die allgemeine Lösung einer PDE fast immer von einer beliebigen Funktion abhängt, sind zusätzliche Bedingungen an das System erforderlich. Diese sind abhängig vom Typ der PDE zu wählen und sind notwendig, da wir nicht nur daran interessiert sind, irgendeine Lösung zu finden, sondern eine, die für das vorliegende „gut gestellte“ Problem eindeutig ist. Wir werden jetzt Anfangs- und Randbedingungen (vgl. unter anderem die Abreit von F. TRÖLTZSCH [52], C.-D. MUNZ und T. WESTERMANN [39] und P. FURLAN [20]) und ihre Auswirkungen auf die PDE genauer betrachten, da diese Nebenbedingungen das eindeutige Verhalten der vollständigen Lösung definieren. Die Randbedingungen sind in Anlehnung an das Skript von C. BÜSKENS [8] definiert.

2.5.1 Anfangsbedingungen

Da viele PDEs ohne jegliche Nebenbedingungen unendlich viele Lösungen besitzen, haben diese Bedingungen die Aufgabe, eine eindeutige Auswahl aus den berechneten Lösungen zu treffen oder eine leere Lösungsmenge zu finden. Die erste Nebenbedingung, die wir betrachten, ist das Bereitstellen von Anfangsbedingungen, die den Anfangszustand eines physikalischen Systems beschreiben.

Wenn die PDE eine zeitliche Entwicklung wiedergibt, legen Anfangsbedingungen fest, in welchem Zustand die Bewegung (z.B. Wellenbewegung) beginnt. Möchte man zum Beispiel eine Auslenkung einer schwingenden Saite $y_{tt} = c^2 y_{xx}$ an der Stelle x zum Zeitpunkt t angeben, dann muss man folgende zwei Bedingungen vorgeben, um eine eindeutige Lösung zu erhalten:

- Anfangslage $y(x, 0) = 0$ zur Zeit $t = 0$ beschreibt keinerlei Auslenkung aus der Ruhelage und

- Anfangsgeschwindigkeit $\frac{\partial y}{\partial t}(x, 0) = 0$ bedeutet, dass die Saite sich zu Beginn in Ruhe befindet.

Erst mit diesen Voraussetzungen ist das Problem vollständig beschrieben.

2.5.2 Randbedingungen

Bei anderen Problemen, wie zum Beispiel der Wärmeleitungsgleichung, reicht es aus anzugeben, welchen Funktionswert die gesuchte Lösung zu Beginn hat und nicht zusätzlich noch Bedingungen an die Ableitung bei $t = 0$ zu stellen. Meist sind so genannte Randbedingungen notwendig, um ein Problem vollständig zu beschreiben. Bei der Wärmeleitungsgleichung $y_t = \kappa y_{xx}$ zum Beispiel geschieht dies mittels

- wesentlicher Randbedingungen

$$y(x, t) = h(x, t) \quad \forall (x, t) \in \Gamma \times (0, T)$$

mit h als bekannter Randtemperatur und

- natürlicher Randbedingungen

$$y_\nu(x, t) = h(x, t) \quad \forall (x, t) \in \Gamma \times (0, T)$$

mit h als Wärmefluss am Rand oder

$$y_\nu(x, t) = \alpha(h(x, t) - y(x, t)) \quad \forall (x, t) \in \Gamma \times (0, T)$$

mit h als Außentemperatur und α als Wärmeübergangszahl.

Wir werden jetzt die zweite Art von Nebenbedingungen genauer betrachten, die Bedingungen an den Rand. Dabei unterscheiden wir Dirichlet-, Neumann- und Cauchy-Randbedingungen.

Zur Definition dieser drei Randbedingungen setzen wir folgende **Voraussetzungen** fest:

Sei $\Omega \subset \mathbb{R}^n$ ein beschränktes Gebiet und werde durch mehrere Raumkurven berandet. Sei Γ die Vereinigung sämtlicher Randkurven. Der Rand bestehe aus stückweise stetig differenzierbaren Kurven, auf denen die vom Gebiet Ω ins Äußere zeigende Normalenableitung ν erklärt werden kann.

Es seien Γ_1 , Γ_2 und Γ_3 disjunkte Randteile, wobei auch die leere Menge erlaubt ist mit

$$\Gamma_1 \cup \Gamma_2 \cup \Gamma_3 = \Gamma \tag{2.9}$$

und f_1 , f_2 , f_3 und α seien gegebene Funktionen auf den betreffenden Randteilen.

2.5.2.1 Dirichlet-Randbedingung

Die Dirichlet-Randbedingung, benannt nach PETER GUSTAV LEJEUNE DIRICHLET⁴, setzt zur Berechnung der Lösungsfunktion Werte auf dem jeweiligen Rand des Definitionsbereichs fest, die angenommen werden sollen.

Definition 2.15 (Dirichlet-Randbedingung)

Es gelten die gemachten Voraussetzungen. Dann heißt die Randbedingung der Form

$$y(x, t) = f_1(x, t) \quad \forall (x, t) \in \Gamma_1 \quad (2.10)$$

Dirichlet-Randbedingung. □

Bemerkung 2.16

Auf einem Gebiet $\Omega = (0, N)$ könnte das folgendermaßen aussehen:

$$y(0, t) = \sigma_1, \quad y(N, t) = \sigma_2 \quad (2.11)$$

mit gegebenen reellen Zahlen σ_1 und σ_2 für die Funktionswerte der Lösung auf dem Rand $\partial\Omega = \Gamma$. ○

Wenn nur Dirichlet-Randbedingungen bei einer Problemstellung auftreten, sprechen wir auch von einer *Dirichlet-Randwertaufgabe*.

2.5.2.2 Neumann-Randbedingung

Zum Beispiel bei elektrostatischen Problemen benötigen wir andere Randbedingungen. So verschwindet an einem optimalen Leiter das elektrische Feld senkrecht zum Rand und mathematisch die Normalenableitung des Potentials y . Die so genannte Neumann-Randbedingung, benannt nach CARL GOTTFRIED NEUMANN⁵, setzt Werte auf dem Rand des Definitionsbereichs für die Normalenableitung der Lösung fest.

Definition 2.17 (Neumann-Randbedingung)

Es gelten die gemachten Voraussetzungen und $\frac{\partial y}{\partial \nu}$ bezeichne die Richtungsableitung des nach außen gerichteten Normalenvektors⁶ von Γ_2 . Dann heißt die Randbedingung der Form

$$\frac{\partial y}{\partial \nu}(x, t) = f_2(x, t) \quad \forall (x, t) \in \Gamma_2 \quad (2.12)$$

Neumann-Randbedingung. □

Bemerkung 2.18

Auf einem Gebiet $\Omega = (0, N)$ könnte das folgendermaßen aussehen:

$$\frac{\partial y}{\partial \nu}(0, t) = \sigma_3, \quad \frac{\partial y}{\partial \nu}(N, t) = \sigma_4 \quad (2.13)$$

mit gegebenen reellen Zahlen σ_3 und σ_4 für die Normalenableitung der Lösung auf dem Rand $\partial\Omega = \Gamma$. ○

⁴geboren 13. Februar 1805 in Düren und gestorben 5. Mai 1859 in Göttingen

⁵geboren 7. Mai 1832 in Königsberg (Preußen) und gestorben 27. März 1925 in Leipzig

⁶siehe Abschnitt 4.2.2.3

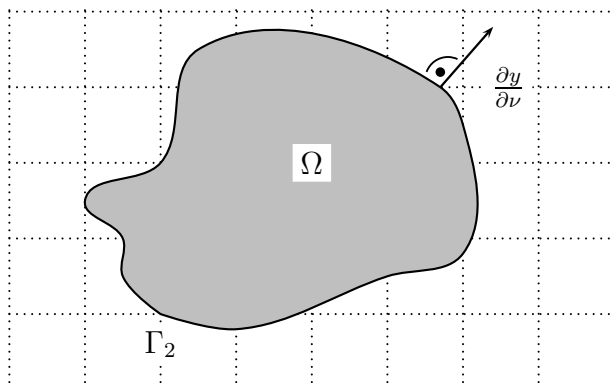


Abbildung 2.1: Neumann-Randbedingungen

Wenn nur Neumann-Randbedingungen bei einer Problemstellung auftreten, sprechen wir auch von einer *Neumann-Randwertaufgabe*.

2.5.2.3 Cauchy-Randbedingung

Zur vollständigen Darstellung der Randbedingungen sei noch die Cauchy-Randbedingung genannt, die die Dirichlet- und Neumann-Randbedingung kombiniert.

Definition 2.19 (Cauchy-Randbedingung)

Es gelten die gemachten Voraussetzungen und $\frac{\partial y}{\partial \nu}$ bezeichne die Richtungsableitung des nach außen gerichteten Normalvektors von Γ_3 . Dann heißt die Randbedingung der Form

$$\frac{\partial y}{\partial \nu}(x, t) + \alpha y(x, t) = f_3(x, t) \quad \forall (x, t) \in \Gamma_3 \tag{2.14}$$

Cauchy-Randbedingung. □

Ein Beispiel hierfür ist ein isolierter Stab mit Länge l (d.h. Intervall $[0, l]$) mit eingetauchtem rechten Ende in einem Wasserreservoir der Temperatur $f(t)$. Es kommt zu einem Wärmeaustausch zwischen Stabende und Wasserreservoir, der mathematisch mittels der Cauchy-Randbedingung dargestellt werden kann:

$$\frac{\partial y}{\partial \nu}(l, t) = \alpha[y(l, t) - f(t)] \quad \text{mit } \alpha > 0 \tag{2.15}$$

Das bedeutet, dass sich der Wärmezufluss am Stabende proportional zur Temperaturdifferenz verhält.

Mit der Schreibweise aus der Definition:

$$\frac{\partial y}{\partial \nu}(l, t) + \alpha y(l, t) = \alpha f(t). \tag{2.16}$$

2.6 Kontrollmöglichkeiten der Systeme

Am Schluss des Kapitels über PDEs betrachten wir die Kontrollmöglichkeiten unserer gegebenen Systeme. Diese so genannten Kontrollsysteme, die als Variablen t und x besitzen, hängen außerdem von einer Funktion u ab, die von x und t abhängen kann. Wir werden hier in diesem Abschnitt allgemein den Begriff der Steuerung für u verwenden und meinen damit eine Einflussnahme von außen auf unser System.

Die Einflussnahme geschieht als Erstes

- mit Hilfe einer **Randsteuerung**, die nur an den Rändern des Gebiets Ω anliegt und somit nur “indirekt” vom Rand aus auf das Innere des Gebiets wirken kann

und als Zweites

- mit einer **verteilten Steuerung**, die “direkte” Einwirkung auf das Innere unseres Gebiets Ω hat.

Normalerweise wird die Steuerung in Abhängigkeit der Zeit t betrachtet. Im Folgenden wollen wir zur Vereinfachung aber die einfachste Art von PDEs wählen, die elliptischen PDEs, die nicht von der Variable t abhängen. Wir stellen also zwei motivierende Beispiele in Anlehnung an das Werk von F. TRÖLTZSCH [51] Abschnitt 1.2.1 für diese beiden Steuermöglichkeiten dar.

2.6.1 Randsteuerung

Es sei ein Ortsgebiet $\Omega \subset \mathbb{R}^3$ mit Rand Γ gegeben, das einen Körper darstellt, der sowohl geheizt als auch gekühlt werden kann.

Am Rand legen wir eine zeitlich konstante Temperatur $u = u(x)$ in Abhängigkeit des Ortes $x \in \Omega$ an.

Das Ziel soll sein, die bestmögliche Annäherung einer vorgegebenen Temperaturverteilung $y_\Omega = y_\Omega(x)$ in Ω zu bestimmen. Eine Möglichkeit, das System mit Zielfunktional⁷ zu beschreiben, ist folgende:

$$\min_u J(y, u) := \frac{1}{2} \int_{\Omega} (y(x) - y_\Omega(x))^2 dx + \frac{\lambda}{2} \int_{\Gamma} u(x)^2 ds(x) \quad (2.17)$$

mit den Nebenbedingungen

$$-\Delta y = 0 \quad \text{in } \Omega$$

$$\frac{\partial y}{\partial \nu} = \alpha(u - y) \quad \text{auf } \Gamma$$

⁷Das Zielfunktional ist das wesentliche Hilfsmittel für ein Optimalsteuerungsproblem, das in unseren Auswertungen minimiert werden soll (näheres hierzu in Abschnitt 3.5, Definition 3.32)

$$u_a(x) \leq u(x) \leq u_b(x).$$

Graphisch können wir uns die Randsteuerung bei einem elliptischen Problem wie in Abbildung 2.2 vorstellen.

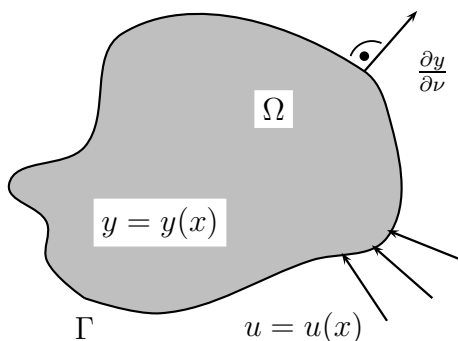


Abbildung 2.2: Randsteuerung

Die zwei ersten Nebenbedingungen stellen die Zustandsgleichung dar, die letzte Nebenbedingung beschreibt eine punktweise Beschränkung an die Steuerung aufgrund von beschränkten Kapazitäten. λ sei ein Gewicht für die Energiekosten der Steuerung u und α die Wärmeübergangszahl von Ω in das Medium hinein.

Gesucht ist also bei dieser Art von Steuerung ein optimales $u = u(x)$ mit Zustand $y = y(x)$.

2.6.2 Verteilte Steuerung

Ganz analog kann u als Temperaturquelle in Ω wirken, zum Beispiel beim Aufheizen eines Körpers Ω durch elektromagnetische Induktion oder Mikrowelle. Dabei sei die Randtemperatur mit Null vorgegeben.

Analog stellen wir wieder unser System auf:

$$\min_u J(y, u) := \frac{1}{2} \int_{\Omega} (y(x) - y_{\Omega}(x))^2 dx + \frac{\lambda}{2} \int_{\Omega} u(x)^2 ds(x) \quad (2.18)$$

mit den Nebenbedingungen

$$-\Delta y = \beta u \quad \text{in } \Omega$$

$$y = 0 \quad \text{auf } \Gamma$$

$$u_a(x) \leq u(x) \leq u_b(x) \quad \text{in } \Omega.$$

Auch hier können wir uns die verteilte Steuerung anhand der Abbildung 2.3 einfacher vorstellen, wobei $\beta = \beta(x)$ auf ganzem Ω oder nur auf einem Teilgebiet $\Omega_c \subset \Omega$ vorgegeben ist, also entweder auf dem kompletten Gebiet wirkt oder nur auf einem Teil.

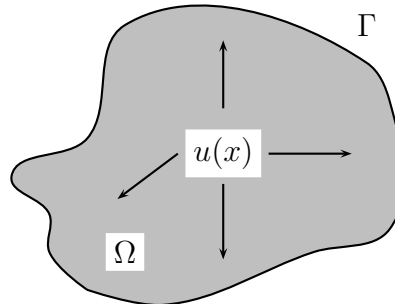


Abbildung 2.3: Verteilte Steuerung

Es ist allerdings realistischer aufgrund einer wahrscheinlich vorhandenen Außentemperatur die Randtemperatur $y_a \neq 0$ anzunehmen. Dies ändert die zweite Nebenbedingung in

$$\frac{\partial y}{\partial \nu} = \alpha(y_a - y) \quad \text{auf } \Gamma. \quad (2.19)$$

Die Aufgabenstellung kann auch analog zur Randsteuerung aufgestellt werden, nur mit verteilter Steuerung u .

Diese Art von Steuerung hat natürlich größere Möglichkeiten, auf ein System einzuwirken und es in den gewünschten Zustand zu bringen, wie wir auch später sehen werden. Probleme mit Randsteuerung hingegen sind um einiges schwerer zu beeinflussen, vor allem da die Auswirkungen nach innen sich sehr unterschiedlich je nach „Art“ des Gebiets verhalten können.

3 Modellprädiktive Regelung von Kontrollsystemen

3.1 Einführung in die modellprädiktive Regelung

Die (modellbasierte) prädiktive Regelung oder auch Regelung auf einem sich bewegendem Horizont¹ ist eine etablierte Methode auf dem Gebiet der optimalen Kontrolle von linearen und nichtlinearen Systemen. Ihr Ursprung liegt in den späten siebziger Jahren als Methodik in der Prozesskontrolle, hier vor allem in der Chemie- und verfahrenstechnischen Industrie. Sie ist seitdem Gegenstand aktiver Forschungsarbeit und stetiger Weiterentwicklung.

Der Begriff “Modellprädiktive Regelung” bezeichnet nicht etwa eine einzige spezielle Kontrollstrategie, sondern umfasst vielmehr eine mittlerweile breite Palette an Kontrollmethoden, die in den letzten Jahrzehnten entstanden sind. Zu den bekannten linearen Ansätzen der modellprädiktiven Regelung zählen z.B.

- Identification and Command (IDCOM): ein heuristischer modellprädiktiver Kontrollansatz nach RICHLET et al. [42], 1978,
- Dynamic Matrix Control (DMC): ein besonders mächtiger Algorithmus für multiple input multiple output (MIMO) Kontrollsysteme nach CUTLER und RAMAKER [13], 1980,
- Generalised Predictive Control (GPC) nach CLARKE et al. [12], 1987,

um nur einige aus den mehr als 2200 Verfahren umfassenden und von chemischer Industrie bis hin zu Raumfahrt reichenden Anwendungsgebieten zu nennen (vgl. hierzu auch [10] und [43]).

Basieren die genannten Methoden noch alle auf linearen Modellen, begann man Anfang der neunziger Jahre den Fokus vermehrt auf nichtlineare modellprädiktive Kontrolle (NMPC) zu legen. Dabei besteht hier das praktische Interesse vor allem in der Tatsache, dass an heutige Prozesse höhere Anforderungen und Bedingungen z.B. seitens umwelt- und sicherheitstechnischer Gesichtspunkte gestellt werden, und dies stets unter der Vorgabe einer hohen Leistungsfähigkeit der angewandten Methoden. Dies ist oft nur unter Einbeziehung der Nichtlinearität zu bewerkstelligen, wie sie in den meisten natürlichen Abläufen und Verhaltensmustern in der Natur gegeben sind.

Doch was genau ist die Grundidee von MPC? Dazu betrachte man das natürliche mensch-

¹Im Englischen verwendet man die Bezeichnungen *model predictive control (MPC)*, *moving horizon control (MHC)* oder *receding horizon control (RHC)*.

liche Verhalten innerhalb eines Vorgangs oder Prozesses in der Realität: Der Mensch analysiert intern für sich ein einfaches Abbild oder Modell des realen Prozesses und selektiert dann für das Erreichen des gesetzten Zieles eine Anzahl von Handlungen, von denen er denkt, sie führen ihn auf effektivstem Wege dorthin. Er “optimiert” also anhand möglicher Einflussgrößen sein Verhalten, um das angestrebte Ziel zu erreichen. Oftmals treten im zeitlichen Verlauf unvorhergesehene Störungen oder Änderungen auf, so dass zu diesen Zeitpunkten auf die Störung durch erneute Selektion und Optimierung reagiert werden muss. Genau dieses Vorgehen stellt das Grundkonzept aller modellprädiktiven Regelungsansätze dar (siehe [43]):

1. Das Regelungsgesetz basiert auf vorhergesagtem Verhalten.
2. Die Verlaufsprognose des Systems wird mittels eines Prozessmodells berechnet.
3. Die zu bestimmten Zeitpunkten stattfindenden “kontrollierenden Eingriffe” in das System werden durch Optimierung einer vorher festgelegten Gütefunktion für das vorhergesagte Verhalten ermittelt.
4. Das Verschieben des Horizonts: Der betrachtete Zeitabschnitt wird um eine bestimmte Anzahl an Zeitschritten in die Zukunft verschoben.

Ein abschließendes Beispiel soll diesen Vergleich verdeutlichen. Dazu betrachte man den Prozess des Autofahrens, wobei der Autofahrer einem MPC-Regler, der Straßenverlauf einer Referenztrajektorie, die charakteristischen Eigenschaften des Fahrzeugs dem Modell, und Lenkung und Pedale den Steuermöglichkeiten entsprechen. Der Autofahrer wird, den Blick stets auf die Straße gerichtet, den Straßenverlauf einige hundert Meter voraus einsehen und somit potenzielle Gefahren antizipieren können. Während sich das Auto auf der Straße bewegt, ändert sich auch der einsehbare Straßenverlauf in Fahrtrichtung, so dass immer die nächsten hundert Meter der Strecke gesehen werden können. Dies entspricht im Wesentlichen dem Verschieben des Horizonts, d.h. die Sichtweite des Fahrers verschiebt sich in Fahrtrichtung in derselben Geschwindigkeit, mit der das Auto fährt. Somit erhält der Fahrer ständig neue Informationen, die er dazu verwendet, seine Entscheidungen bezüglich der Kontrolle zu aktualisieren.

3.2 Mathematische Kontrolltheorie

Die mathematische Kontrolltheorie — oder korrekter ausgedrückt: mathematische *Steu-
rungstheorie* oder *Regelungstheorie* — stellt das mathematische Gegenstück der In-
genieursdisziplin *Regelungstechnik* dar und bildet sozusagen eine Art “Schnittstelle” der Be-
reiche Mathematik, Ingenieurwissenschaften, Wirtschaftswissenschaften und Informatik
(vgl. [21, 37, 38]).

Unter dem Begriff der Kontrolle versteht man die gezielte Beeinflussung eines dynami-
schen Systems, das durch gewöhnliche oder partielle Differentialgleichungen beschrieben
ist, um es zu einem gewünschten Verhalten zu veranlassen, wie z.B. es in einem Gleichge-
wichtspunkt zu stabilisieren oder es entlang einer vorgegebenen Trajektorie verlaufen zu

lassen. Es geht also darum, frei wählbare Funktionen innerhalb des Differentialgleichungssystems so zu bestimmen, dass resultierende Lösungen vorgeschriebene oder bevorzugte Eigenschaften haben (vgl. [56]). Dies lässt sich sehr schön an folgendem Beispiel veranschaulichen:

Beispiel 3.1 (Balancieren eines Stabes auf der Fingerspitze)

Eines der zweifellos bekanntesten und einfachsten “Kontrollprobleme” ist das Balancieren eines Stabes auf der Fingerspitze, wobei der Stab das dynamische System darstellt. Der aufgerichtete Stab entspricht einem instabilen Gleichgewicht des Systems.

Um den Stab in der aufgerichteten Position zu halten, wird man nun versuchen, einem Umkippen des Stabes z.B. nach vorne durch eine gleichgerichtete Bewegung des Fingers entgegenzuwirken. Man versucht also das System mittels Bewegungen der Hand dahingehend zu kontrollieren, dass der Stab in dem gewünschten Zustand (hier: im Gleichgewicht in aufgerichteter Position) verbleibt. Eine graphische Veranschaulichung dieses Vorgangs findet sich in Abbildung 3.1. •

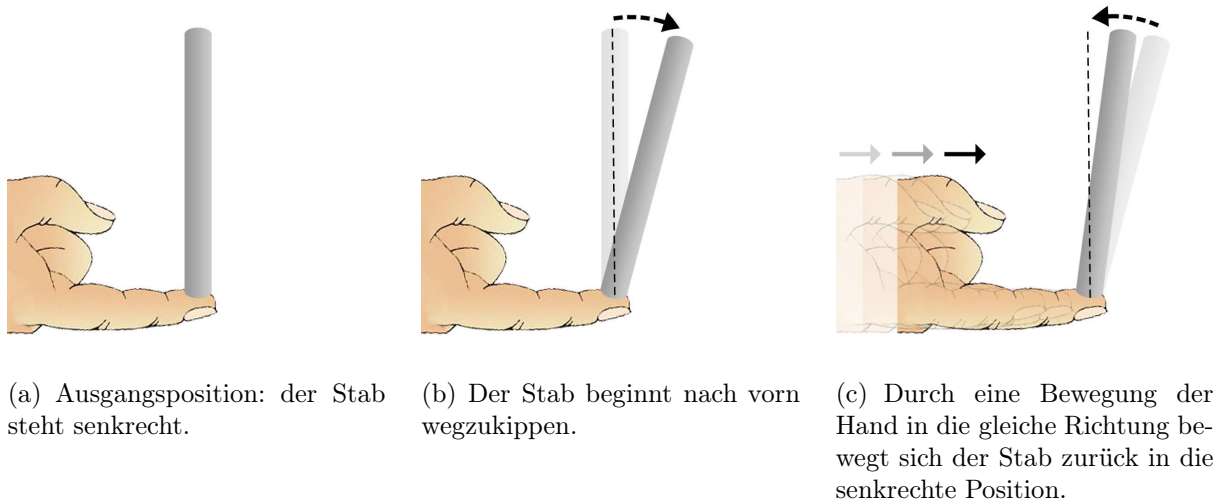


Abbildung 3.1: Stabilisierung eines Stabes mit der Fingerspitze.

Für weitere theoretische Einblicke und Grundlagen sowie interessante und anschauliche Beispiele, z.B. das des “Inversen Pendels” oder des “Fliehkraftreglers”, sei auf [3, 25, 26, 54] verwiesen.

3.2.1 Unterscheidung von Steuerung und Regelung

Wie zu Beginn bereits erwähnt, wird im Deutschen der Begriff der Kontrolle übergeordnet verwendet, indem er umfassend für die Begriffe *Steuerung* und *Regelung* steht. Im Englischen hingegen unterscheidet man zwischen *open loop control* (Steuerung) und *closed loop control* (Regelung). Die Hauptunterschiede sollen in den beiden folgenden Definitionen (vgl. dazu [32]) verdeutlicht werden. Die Begriffe der Steuerung und der Regelung sind in der Deutschen Industrienorm DIN 19226² festgelegt.

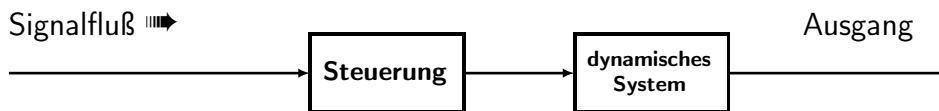
²vgl. Deutsches Institut für Normung e. V., Berlin

Definition 3.2 (Steuerung)

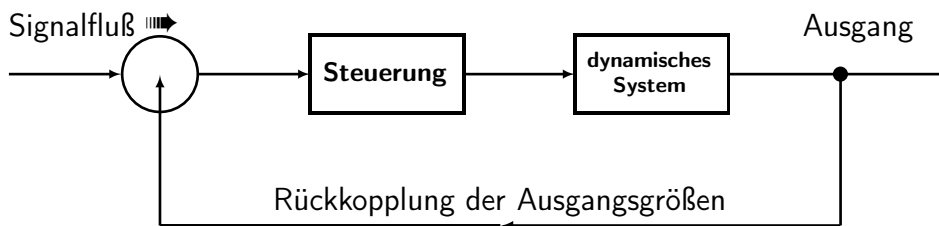
Als Steuerung versteht man die Beeinflussung eines Systems oder eines Prozesses durch die Berechnung von Stellgrößen in Abhängigkeit der Eingangsgrößen und der Zustandsgrößen. Die Ausgangsgrößen werden nicht abgefragt und damit nicht berücksichtigt (*offener* Steuerkreis), Störgrößen werden nicht erkannt. Steuerungen können somit nicht auf auftretende Störungen innerhalb des Prozesses reagieren. Eine graphische Veranschaulichung findet sich in Abbildung 3.2(a). □

Definition 3.3 (Regelung)

Die Regelung eines Systems beschreibt einen Vorgang indem die Ausgangsgrößen des Systems abgefragt und rückgekoppelt — d.h. wieder in das System zurückgeführt — werden und damit Einfluss auf die Stellgrößen nehmen (*geschlossener* Regelkreis). Störgrößen werden erkannt, so dass innerhalb des Prozesses auf auftretende Störungen reagiert werden kann. Eine graphische Veranschaulichung findet sich in Abbildung 3.2(b). □



(a) **Steuerung:** Offene Wirkungskette.



(b) **Regelung:** Geschlossene Wirkungskette.

Abbildung 3.2: Schematische Darstellung von Steuerung und Regelung

Bemerkung 3.4

Die entscheidenden Merkmale einer Steuerung im Vergleich zu einer Regelung sind somit das Fehlen einer fortlaufenden Rückkopplung der Ausgangsgrößen auf die Eingangsgrößen und die Tatsache, dass auf Störungen nicht reagiert werden kann. Das hat zur Folge, dass zur Kontrolle von vor allem dynamischen evolutionären Systemen und Prozessen auf Methoden der Regelung (wie eben die modellprädiktive Regelung) gesetzt wird. ○

Tiefere Einblicke in das Gebiet der Regelungstechnik geben unter anderem [19, 22, 50].

3.2.2 Stabilität dynamischer Systeme

Zu einer der grundlegenden Aufgaben der Kontrolltheorie zählt das Analysieren bzw. das Lösen von Stabilisierungsproblemen, da auf Stabilität als eine wesentliche Eigenschaft dynamischer Systeme sowohl in theoretischen als auch in praktischen Anwendungen (z.B.

bei Kraftfahrzeugen das ‘‘Elektronische Stabilitatsprogramm’’, kurz: ESP) nicht verzichtet werden kann (vgl. [26, 30]). Dabei versteht man unter Stabilitat im Allgemeinen, das Zuebewegen der Systemzustande auf einen Sollwert und anschlieendes Verbleiben an oder innerhalb einer Umgebung von diesem Punkt.

Zur Definition der verschiedenen Stabilitatskonzepte werden vorab noch einige grundlegende Begriffe und Definitionen benotigt, fur die wir in diesem Abschnitt die Existenz und Eindeutigkeit einer Losung φ unseres Systems o.B.d.A. voraussetzen, den Nachweis dazu aber in Abschnitt 3.3.2 sowohl fur diskrete als auch kontinuierliche Systeme erbringen.

Das Standardmodell der von uns betrachteten dynamischen Systeme besteht aus einer endlichen Anzahl gewohnlicher Differentialgleichungen erster Ordnung

$$\dot{x}(t) := \frac{d}{dt}x(t) = f(x(t)), \quad (3.1)$$

welche wir mittels des Verfahrens der *Diskretisierung* unserer das System definierenden partiellen Differentialgleichung erhalten (siehe hierzu Kapitel ?? Diskretisierung).

Ein oft angestrebtes Verhalten eines solchen Systems ist dessen Stabilisierung in eine *Ruhelage* oder in ein *Gleichgewicht*.

Definition 3.5 (Gleichgewicht)

Ein Punkt $x^* \in \mathbb{R}$ heit *Gleichgewicht*, *Ruhelage* oder *Equilibrium* der gewohnlichen Differentialgleichung (3.1), falls fur die zugehorige Losung

$$\varphi(t, x^*) = x^* \quad \forall t \in \mathbb{R}$$

erfullt ist, oder aquivalent dazu

$$f(x^*) = 0$$

gilt. □

Bemerkung 3.6

Man kann o.B.d.A. fur das Gleichgewicht $x^* = 0$ annehmen, da im Falle $x^* \neq 0$ das transformierte System $\tilde{f}(x) = f(x + x^*)$ mit $\tilde{f}(0) = f(0 + x^*) = x^*$ betrachtet werden kann. ◦

Im Folgenden sei x^* immer ein Gleichgewicht. Eine besonders fur nichtlineare Systeme verwendete Technik ist die Charakterisierung von Stabilitat mittels *Vergleichsfunktionen*.

Definition 3.7 (Vergleichsfunktionen)

Es werden die folgenden Klassen von *Vergleichsfunktionen* definiert:

- $\mathcal{K} \quad := \quad \{\alpha : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+ \mid \text{stetig, streng monoton wachsend, } \alpha(0) = 0\}$
- $\mathcal{K}_\infty \quad := \quad \{\alpha \in \mathcal{K} \mid \alpha \text{ ist unbeschrankt}\}$
- $\mathcal{L} \quad := \quad \{\gamma : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+ \mid \text{stetig, streng monoton fallend, } \lim_{r \rightarrow \infty} \gamma(r) = 0\}$
- $\mathcal{KL} \quad := \quad \{\beta : \mathbb{R}_0^+ \times \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+ \mid \text{stetig, } \beta(r, \cdot) \in \mathcal{L} \forall r > 0, \beta(\cdot, t) \in \mathcal{K} \forall t \geq 0\}$
- $\mathcal{KL}_0 \quad := \quad \{\beta : \mathbb{R}_0^+ \times \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+ \mid \text{stetig, } \beta(r, \cdot) \in \mathcal{L} \forall r > 0, \beta(\cdot, t) \in \mathcal{K}_\infty \text{ oder } \beta(\cdot, t) \equiv 0 \forall t > 0\}$

□

Bemerkung 3.8

Die Klasse \mathcal{KL}_0 stellt eine Erweiterung der Klasse \mathcal{KL} dar. Es sind neben den \mathcal{KL} -Funktionen auch solche enthalten, die im ersten Argument zunächst aus der Klasse \mathcal{K}_∞ sind, aber nun auch für bestimmte r identisch Null sein können. ◦

Es lassen sich nun die für die hier behandelten Systeme interessanten Stabilitätskonzepte definieren.

Definition 3.9 (stabil, asymptotisch stabil, exponentiell stabil)

Sei $x^* = 0 \in \mathbb{R}$ ein Gleichgewicht der Differentialgleichung (3.1).

1. x^* heißt *stabil*, falls eine Umgebung N von x^* und eine Funktion $\alpha \in \mathcal{K}$ existieren, so dass gilt

$$\|\varphi(t, x_0)\| \leq \alpha(\|x_0\|) \quad \forall x_0 \in N, t \geq 0. \tag{3.2}$$

2. x^* heißt *instabil*, falls 1. nicht gilt.
3. x^* heißt (*lokal*) *asymptotisch stabil*, falls eine Umgebung N von x^* und eine Funktion $\beta \in \mathcal{KL}$ existieren, so dass gilt

$$\|\varphi(t, x_0)\| \leq \beta(\|x_0\|, t) \quad \forall x_0 \in N, t \geq 0. \tag{3.3}$$

4. x^* heißt (*lokal*) *exponentiell stabil*, falls Konstanten $C, \sigma > 0$ existieren, so dass 3. mit $\beta(r, t) \leq C e^{-\sigma t} r$ gilt.

Die Stabilität in 3. und 4. nennt man *global*, falls dort $N = \mathbb{R}^n$ gilt. □

Ein weiteres wichtiges Hilfsmittel für die Stabilitätsuntersuchungen von Differentialgleichungen erhält man mittels verallgemeinerten Abstandsfunktionen, den sogenannten *Ljapunov-Funktionen*, benannt nach dem russischen Mathematiker und Physiker ALEXANDER M. LJAPUNOV (1857 – 1918). Mit ihnen ist es möglich eine hinreichende und notwendige Bedingung für die asymptotische Stabilität anzugeben.

Es wird hierfür im Anschluss nur die Definition und der Satz als eines der wichtigsten Resultate angegeben. Eine detaillierte Beschreibung der Zwischenschritte und ausführliche Beweisführung findet sich im Skript zur Vorlesung “Mathematische Kontrolltheorie II: Nichtlineare Systeme” von L. GRÜNE (siehe [26]).

Definition 3.10 (Ljapunov-Funktion)

Betrachte eine Differentialgleichung (3.1) mit $f(0) = 0$, O sei eine offene Umgebung der Null. Eine stetige, auf $O \setminus \{0\}$ stetig differenzierbare Funktion $V : O \rightarrow \mathbb{R}_0^+$ heißt *lokale Ljapunov-Funktion*, falls Funktionen $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$ und eine stetige Funktion $W : O \rightarrow \mathbb{R}$ existieren, so dass die Ungleichungen

$$W(x) > 0 \tag{3.4}$$

$$\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|) \tag{3.5}$$

$$DV(x) f(x) \leq -W(x) \tag{3.6}$$

für alle $x \in O \setminus \{0\}$ gelten.

Das Paar (V, W) heißt *Ljapunov-Paar*.

Erfüllen V und W diese Bedingungen für $O = \mathbb{R}^n$, dann nennt man die Funktion V eine *globale Ljapunov-Funktion*. □

Lemma 3.11

Es existiert für jedes Ljapunov-Paar (V, W) und jede abgeschlossene Teilmenge $D \subseteq O$ der offenen Umgebung O der Null eine global Lipschitz-stetige Funktion $g : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$, so dass $(V, g(V))$ ebenfalls ein Ljapunov-Paar ist. \diamond

Damit kommt man zu einem der Hauptresultate der mathematischen Kontrolltheorie:

Satz 3.12 (Ljapunov-Funktion \iff Asymptotische Stabilität)

Betrachte eine Differentialgleichung (3.1) mit $f(0) = 0$. Es existiert genau dann eine lokale (bzw. globale) Ljapunov-Funktion V im Sinne der Definition 3.10, wenn das Gleichgewicht $x^* = 0$ lokal (bzw. global) asymptotisch stabil ist. Die Vergleichsfunktion $\beta \in \mathcal{KL}$ ist dabei gegeben durch

$$\beta(r, t) = \alpha_1^{-1}(\mu(t, \alpha_2(r))), \quad (3.7)$$

mit μ als Lösung des eindimensionalen Anfangswertproblems

$$\frac{d}{dt}\mu(t, r) = -g(\mu(t, r)), \quad \mu(0, r) = r \quad (3.8)$$

und Funktion g aus Lemma 3.11. \diamond

Beweis: Dieser Satz stellt sozusagen die Kombination der Hauptresultate “Ljapunov-Funktion \implies Asymptotische Stabilität” und dessen Umkehrung aus [26], Kapitel 2: Stabilität nichtlinearer Differentialgleichungen, Satz 2.10 und Satz 2.13, dar. Die Beweisführungen hierzu erfolgen wiederum unter Verwendung weiterer Hilfsresultate, so dass für einen ausführlichen Beweis des obigen Satzes auf die genannte Literatur verwiesen wird. \blacklozenge

3.3 Kontrollsysteme

Nach einem Einblick in das Fachgebiet der mathematischen Kontrolltheorie widmet sich dieser Abschnitt nun konkreter der in Bezug auf das Lösen der Stabilisierungsproblematik zugrundeliegenden Technik: der Analyse von *Kontrollsystemen*. Dazu wird zunächst der Begriff des Kontrollsystems definiert und anschließend die Stabilitätsdefinitionen aus dem vorhergehenden Abschnitt auf Kontrollsysteme verallgemeinert.

3.3.1 Definition eines Kontrollsystems

Kontrollsysteme dienen dazu, das Verhalten von Systemen³ und Prozessen in der Realität mathematisch zu beschreiben. Dabei handelt es sich um dynamische Systeme in *kontinuierlicher* oder *diskreter* Zeit, die von einem *Kontrollparameter* $u \in \mathbb{R}^m$ abhängen. Dieser

³Der Astrophysiker STEVEN KLEIN definiert den Begriff “System” folgendermaßen: “Ein System ist ein synergisches Produkt aus einer Vielzahl von Komponenten.” (Lexikon der Paranomalie III, Kapitel 20, S. 596, Zeile 5ff). Diese prägnante und treffende Beschreibung verdeutlicht sehr gut den Umfang, den ein System als Beschreibung der Gesamtheit verschiedener Vorgänge in einem natürlichen Prozess einnimmt und somit auch die damit verbundene Schwierigkeit, diese einzelnen, in Wechselwirkung zueinander stehenden Bestandteile in eine mathematische Form zu fassen.

kann sich in Abhängigkeit der Zeit und/oder vom Zustand des Systems ändern, um ein gewünschtes Verhalten der Lösungen zu erhalten (vgl. hierzu open loop und closed loop Kontrolle in Definition 3.2 und Definition 3.3).

Die Kontrollsysteme werden bezüglich ihrer Dynamik in *lineare* und *nichtlineare* Systeme unterschieden. Da der Hauptaugenmerk dieser Arbeit auf den nichtlinearen Kontrollsystemen liegt, werden nur diese formal definiert. Der Vollständigkeit halber wird für eine ausführliche Behandlung und Definition linearer Systeme auf die Vorlesung “Mathematische Kontrolltheorie I: Lineare Systeme” von L. GRÜNE [25] verwiesen. Wir orientieren uns in diesem Abschnitt vornehmlich an [26].

Definition 3.13 (Zeitkontinuierliches Kontrollsystem)

Seien $m, n \in \mathbb{N}$ natürliche Zahlen, die Abbildung $f : \mathbb{R}^n \times U \rightarrow \mathbb{R}^n$ ein *parameterabhängiges stetiges Vektorfeld*, die Menge $U \subset \mathbb{R}^m$ der *Kontrollwertebereich* von u und $\mathcal{U} := \{u : \mathbb{R} \rightarrow U \mid u \text{ zulässig}\}$ die Bezeichnung für den *Raum der zulässigen Kontrollfunktionen*.

Dann heißt das System von n Differentialgleichungen 1. Ordnung

$$\dot{x}(t) := \frac{d}{dt}x(t) = f(x(t), u(t)), \quad t \in \mathbb{R} \tag{3.9}$$

ein *Kontrollsystem in kontinuierlicher Zeit* $\mathbb{T} = \mathbb{R}$ mit *Kontrolle* $u(t)$ und *Zustand* $x(t) \in \mathbb{R}^n$ zum Zeitpunkt $t \in \mathbb{T}$. □

Definition 3.14 (Zeitdiskretes Kontrollsystem)

Seien $m, n \in \mathbb{N}$ natürliche Zahlen, sei $f_h : \mathbb{R}^n \times U \rightarrow \mathbb{R}^n$ eine stetige Abbildung, die Menge $U \subset \mathbb{R}^m$ der *Kontrollwertebereich* von u und $\mathcal{U} := \{u : \mathbb{R} \rightarrow U \mid u \text{ zulässig}\}$ die Bezeichnung für den *Raum der zulässigen Kontrollfunktionen*.

Dann heißt das System gegeben durch die Differenzengleichung

$$x(t + h) = f_h(x(t), u(t)) \tag{3.10}$$

ein *Kontrollsystem in diskreter Zeit* $\mathbb{T} = h\mathbb{N} = \{hk \mid k \in \mathbb{N}\}$ für ein $h \in \mathbb{R}, h > 0$, mit *Kontrolle* $u(t)$ und *Zustand* $x(t) \in \mathbb{R}^n$ zum Zeitpunkt $t \in \mathbb{T}$. □

Im folgenden Abschnitt werden wir Eigenschaften “zulässiger” Klassen von Funktionen festlegen.

3.3.2 Existenz und Eindeutigkeit

Die Existenz und Eindeutigkeit einer Lösung im zeitdiskreten Fall zeigt man leicht mittels Induktion: Dazu betrachten wir die folgende Funktionenmenge

$$\mathcal{U}_h := \{u_h : h\mathbb{Z} \rightarrow U \mid u_h \text{ ist zulässig}\}. \tag{3.11}$$

Die *Lösung* von (3.10) ist somit eine Funktion

$$x_h : h\mathbb{N}_0 \times \mathbb{R}^n \times \mathcal{U}_h \rightarrow \mathbb{R}^n$$

für jeden Anfangswert $x_0 \in \mathbb{R}^n$ und jede Funktion $u_h \in \mathcal{U}_h$, für die

$$x_h(0, x_0, u_h) = x_0 \quad \text{und} \tag{3.12}$$

$$x_h(t + h, x_0, u_h) = f_h(x_h(t, x_0, u_h), u_h(t)) \tag{3.13}$$

gilt.

Bemerkung 3.15

Bei der Wahl des Kontrollfunktionenraumes \mathcal{U} ist zu beachten, dass die Existenz und Eindeutigkeit der Lösungen von (3.9) gegeben ist. Des Weiteren sollte die Menge der zulässigen Kontrollfunktionen nicht zu klein gewählt werden, da zum Beispiel die Einschränkung auf die Menge der stetigen Funktionen für viele Anwendungen zu “eng” ist. Ein geläufiges Beispiel einer unstetigen Steuerung ist die sogenannte *bang-bang* Steuerung (vgl. [47, 51]): Charakteristisch für diese ist der sprunghafte Wechsel zwischen zwei Zuständen (vergleiche z.B. *Heaviside-Funktion*, benannt nach dem britischen Mathematiker und Physiker OLIVER HEAVISIDE, 1850 – 1925, vgl. [6]). Man findet sie häufig bei beschränkten optimalen Steuerprozessen. \circ

Für den folgenden Satz benötigen wir noch einige Begriffe.

Definition 3.16 (stückweise konstant, messbar, essentiell beschränkt)

Sei $I = [a, b] \in \mathbb{R}$ ein abgeschlossenes Intervall.

- (i) Eine Funktion $g : I \rightarrow \mathbb{R}^m$ heißt *stückweise konstant*, falls eine Zerlegung von I in endlich viele Teilintervalle I_j , $j = 1, \dots, n$ existiert, so dass g auf I_j konstant ist für alle $j = 1, \dots, n$.
- (ii) Eine Funktion $g : I \rightarrow \mathbb{R}^m$ wird (*Lebesgue-*) *messbar* genannt, falls eine Folge von stückweise konstanten Funktionen $g_i : I \rightarrow \mathbb{R}^m$, $i \in \mathbb{N}$, existiert mit $\lim_{i \rightarrow \infty} g_i(x) = g(x)$ für fast alle $x \in I$.
- (iii) Eine Funktion $g : \mathbb{R} \rightarrow \mathbb{R}^m$ heißt (*Lebesgue-*) *messbar*, falls für jedes abgeschlossene Teilintervall $I = [a, b] \subset \mathbb{R}$ die Einschränkung $g|_I$ messbar im Sinne von (ii) ist.
- (iv) Eine messbare Funktion $g : \mathbb{R} \rightarrow \mathbb{R}^m$ bezeichnet man als *lokal essentiell beschränkt*, falls für jedes kompakte Intervall $I \subset \mathbb{R}$ eine Konstante $C \in \mathbb{R}$, $C > 0$ existiert, so dass $\|u(t)\| \leq C$ ist für fast alle $x \in I$. \square

Satz 3.17 (Satz von Caratheodory)

Gegeben sei ein kontinuierliches Kontrollsystem (3.9) mit folgenden Eigenschaften:

- (1) Der Raum der Kontrollfunktionen ist gegeben durch

$$\mathcal{U} := \{u : \mathbb{R} \rightarrow U \mid u \text{ ist messbar und lokal essentiell beschränkt}\}. \quad (3.14)$$

- (2) Das Vektorfeld $f : \mathbb{R}^n \times U \rightarrow \mathbb{R}^n$ ist stetig.

- (3) Für jedes $R \in \mathbb{R}$, $R > 0$ existiert eine Konstante $M_R \in \mathbb{R}$, $M_R > 0$, so dass die Abschätzung

$$\|f(x, u)\| \leq M_R \quad (3.15)$$

für alle $x \in \mathbb{R}^n$ und alle $u \in U$ mit $\|x\| \leq R$ und $\|u\| \leq R$ erfüllt ist.

- (4) Für jedes $R \in \mathbb{R}$, $R > 0$ existiert eine Konstante $L_R \in \mathbb{R}$, $L_R > 0$, so dass die Abschätzung

$$\|f(x_1, u) - f(x_2, u)\| \leq L_R \|x_1 - x_2\| \quad (3.16)$$

für alle $x_1, x_2 \in \mathbb{R}^n$ und alle $u \in U$ mit $\|x_1\| \leq R$, $\|x_2\| \leq R$ und $\|u\| \leq R$ erfüllt ist.

Dann existiert für jeden Punkt $x_0 \in \mathbb{R}^n$ und jede Kontrollfunktion $u \in \mathcal{U}$ ein maximales Intervall $I = (\tau_{min}, \tau_{max}) \subset \mathbb{R}$ mit $\tau_{max} > 0 > \tau_{min}$, auf dem eine eindeutige und absolut stetige Funktion $x(t)$ existiert, die die Integralgleichung

$$x(t) = x_0 + \int_0^t f(x(\tau), u(\tau)) d\tau \quad (3.17)$$

für alle $t \in I$ erfüllt. ◇

Beweis: Siehe [26], Satz 1.5. ◆

Definition 3.18

Wir bezeichnen im Weiteren die eindeutige Funktion $x(t)$ aus Satz 3.17 mit $\varphi(t, x_0, u)$ und nennen sie die *Lösung* des kontinuierlichen Kontrollsystems (3.9) mit Anfangswert $x_0 \in \mathbb{R}^n$ und Kontrollfunktion $u \in \mathcal{U}$. □

Bemerkung 3.19

Wir rechtfertigen diese Bezeichnung mit der Begründung, dass $\varphi(t, x, u) \in \mathbb{R}^n$ die Differentialgleichung (3.9) für fast alle $t \in I$ erfüllt, da $\varphi(t, x, u)$ aufgrund seiner absoluten Stetigkeit für fast alle $t \in I$ nach t differenzierbar ist. ○

3.3.3 Asymptotische Kontrollierbarkeit und Feedback-Stabilisierung

In diesem Abschnitt sollen nun die bereits kennengelernten Konzepte der Stabilität auf Kontrollsysteme verallgemeinert werden. Dazu betrachte man das Kontrollsystem (3.9)

$$\dot{x}(t) = f(x(t), u(t))$$

mit einer Kontrollfunktion $u(t)$. Diese kann, wie bereits zu Beginn erwähnt, auf die folgenden zwei grundlegenden Kontrollarten Einfluss auf das System nehmen:

1. Einflussnahme als *Steuerung* oder *open-loop control*: dabei ist $u \in \mathcal{U} = L_\infty(\mathcal{R}, U)$ eine explizit zeitvariante Funktion.
2. Einflussnahme als *Regelung* oder *closed-loop control*: $u(t)$ wird mittels einer Abbildung $F : \mathbb{R}^n \rightarrow U$ in Abhängigkeit des aktuellen Zustands als $u(t) = F(x(t))$ gewählt.

Diese Unterscheidung schlägt sich nieder in der folgenden Definition der entsprechenden Stabilitätsbegriffe für Kontrollsysteme. Beiden Definitionen gemein sei das Gleichgewicht $x^* = 0$ des Kontrollsystems (3.9) für ein $u \in \mathcal{U}$.

Definition 3.20 (Asymptotische Kontrollierbarkeit)

Man nennt das Gleichgewicht $x^* = 0$ (lokal) *asymptotisch kontrollierbar*, falls eine offene Umgebung N von x^* und Funktionen $\beta \in \mathcal{KL}$ und $\gamma \in C(\mathbb{R}_0^+, \mathbb{R}_0^+)$ existieren, so dass gilt:

- zu jedem $x \in N$ existiert eine Kontrollfunktion $u \in \mathcal{U}$ mit $\|u\|_\infty \leq \gamma(\|x\|)$
- $\|\varphi(t, x_0, u)\| \leq \beta(\|x_0\|, t)$ für alle $x_0 \in N$ und $t \geq 0$. □

Definition 3.21 (Lipschitz-stetig Feedback-stabilisierbar)

Man nennt das Gleichgewicht $x^* = 0$ (lokal) *Lipschitz-stetig Feedback-stabilisierbar*, falls eine offene Umgebung N von x^* , eine Funktion $\beta \in \mathcal{KL}$ sowie eine stetige Feedback-Abbildung $F : \mathbb{R}^n \rightarrow U$ mit lokal Lipschitz-stetigem Vektorfeld $f(x, F(x))$ in x_0 existieren, so dass gilt:

$$\|\varphi(t, x_0, F)\| \leq \beta(\|x_0\|, t)$$

für alle $t \geq 0$ und $x_0 \in N$. □

Bemerkung 3.22

Die Kontrollierbarkeit bzw. Stabilisierbarkeit aus obiger Definition heißt *global*, falls $N = \mathbb{R}^n$ und *exponentiell*, falls die Funktion $\beta(r, t) = Ce^{-\sigma t}$ für Konstanten $C, \sigma > 0$ gewählt werden kann. ◦

3.4 Übergang von zeitkontinuierlichen zu zeitdiskreten Systemen

Betrachtet man die verschiedenen Gebiete, die heutzutage mathematischen Untersuchungen unterliegen (z.B. Finanzmärkte, Entwicklung der Wirtschaft, Pharmazie, Naturereignisse usw.) so stellt man fest, dass die meisten der zugehörigen Prozesse durch ein zeitkontinuierliches System beschrieben werden. Die mathematischen Auswertungen, Berechnungen oder Bearbeitungen dieser Prozesse geschehen aber unter der Verwendung von digitalen Rechnern, was eine Art “Digitalisierung des Systems” voraussetzt und einen Übergang vom zeitkontinuierlichen System in ein *zeitdiskretes System* notwendig macht. Diesen Übergang bewerkstelligt man zumeist mittels der Methode der *Abtastung*, auch *Sampling* genannt.

3.4.1 Das Prinzip der Abtastung

Beim *Abtasten* oder *Sampeln* (aus dem Englischen: *to sample* - abfragen, probieren, als Muster dienen für) wird das System nur zu bestimmten Zeiten ausgewertet. Wie lange die Dauer zwischen zwei Auswertungspunkten ist, wird durch die *Abtastperiode* oder *Samplingperiode* T festgelegt, die im Falle der Verwendung von digitalen Rechnern durch deren Taktrate $T^* > 0$ nach unten beschränkt ist. Dies bedeutet, das kontinuierliche System wird nur an den Zeitpunkten $0, T, 2T, 3T, \dots$ ausgewertet und man erhält letztendlich eine zeitlich diskretisierte Variante des Systems.

Diese Technik wird vor allem in signalverarbeitenden Bereichen, wie z.B. der *mp3-Kodierung von Wave-Signalen*, eingesetzt. Die Abbildung 3.3 soll dies veranschaulichen.

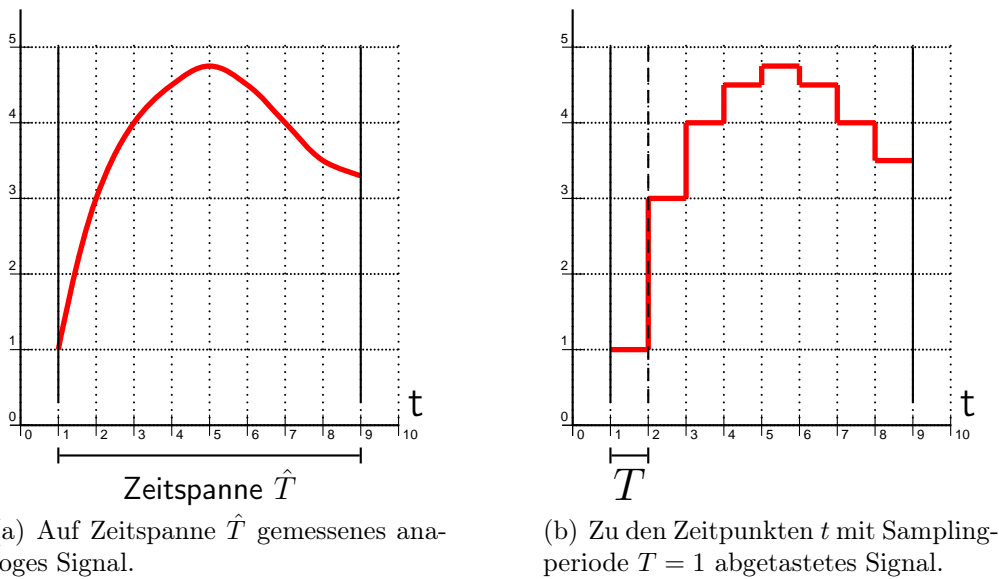


Abbildung 3.3: Analoges und zugehöriges abgetastetes Signal

Bemerkung 3.23

Ein weiterer bedeutender Aspekt der Verwendung Feedback-geregelter Kontrollsysteme ist die Tatsache, dass die Existenz von *stetigen* stabilisierenden Feedbacks nicht immer garantiert werden kann (vgl. [26], Kapitel 7). Will man das Feedback-Stabilisierungsproblem aber dennoch lösen, ist eine Verwendung von *unstetigen* Feedbacks unumgänglich, so dass auch ein für diesen Fall greifendes Lösungskonzept vorhanden sein muss. \circ

3.4.2 Stabilisierung mittels Abtast-Feedback

Im vorhergehenden Abschnitt wurde bereits erwähnt, dass das System nur an den diskreten Zeitpunkten $0, T, 2T, 3T, \dots$ für die Abtastzeit T ausgewertet wird. Ebenso werden bei einem Abtast-Feedback nur zu diesen Zeitpunkten die Kontrollen $u(t)$ berechnet und dann als konstanter Wert über den Intervallen $[kT, (k + 1)T]$ der Länge T implementiert. Das bedeutet, dass je zwischen zwei Abtast-Zeitpunkten kT und $(k + 1)T$ die Kontrolle $u(t)$ als ein *konstantes Feedback* $F(x(t))$ vorliegt, das für die Dauer eines Abtast-Intervalls sich nicht mehr ändert. Damit sind die Stabilitätseigenschaften und die kontrollierende Einflussnahme auf das System stark von der gegebenen Abtastzeit T abhängig.

Definition 3.24 (Abtastlösung)

Betrachte ein Kontrollsystem (3.9). Sei $F : \mathbb{R}^n \rightarrow U$ eine beliebige Abbildung, die die Abschätzung $\|F(x)\| \leq \delta(x)$ für eine stetige Funktion $\delta : \mathbb{R}^n \rightarrow \mathbb{R}$ und alle $x \in \mathbb{R}^n$ erfüllt. Zu einer gegebenen *Abtastperiode* (auch *Abtastzeit* oder *Sampling-Periode*) $T > 0$ definieren wir die *Abtastlösung* oder *Sampling-Lösung* des Anfangswertproblems

$$\begin{aligned} \dot{x}(t) &= f(x(t), F(x(t))) \\ x(0) &= x_0 \end{aligned}$$

für $t \geq 0$ induktiv mittels

$$\varphi_T(t, x_0, F) := \varphi(t - iT, x_i, F(x_i)) \quad \forall t \in [iT, (i + 1)T). \tag{3.18}$$

□

Bemerkung 3.25

Sei nun eine Abtastperiode $T > 0$ gegeben und sei n die Bezeichnung für einen diskreten Zeitpunkt, der der kontinuierlichen Zeit $t = nT$ entspricht. Dann erhalten wir unser *Systemmodell in diskreter Zeit*

$$x(n + 1) = f(x(n), u(n))$$

gemäß Definition 3.14 durch das *Abtastsystem*, indem man

$$f(x(n), u(n)) := \varphi_T(T, x(n), \tilde{u}) \quad \text{mit } \tilde{u}(t) \equiv u(n) = \tilde{u}(0) \text{ für } t \in [0, T)$$

entsprechend Definition 3.24 setzt. ◦

Wir wollen nun einen weiteren an die Stabilisierung mittels Abtastung angepassten Stabilitätsbegriff einführen, der uns vor allem in Bezug auf die bei einigen Auswertungen gemachte Beobachtung eines “nur tendenziell erkennbaren” asymptotisch stabilen Systemverhaltens nützlich sein wird. Dennoch wird im Weiteren aus Gründen der Einfachheit das “unkompliziertere” Konzept der asymptotischen Stabilität Verwendung finden.

Definition 3.26 (semiglobal praktisch asymptotisch stabil)

Betrachte ein Kontrollsystem (3.9) mit $f(0, 0) = 0$. Man sagt, dass eine Familie von Feedbacks $F_T : \mathbb{R}^n \rightarrow U$ für $T \in (0, T^*]$ das Gleichgewicht $x^* = 0$ des Abtastsystems *semiglobal praktisch asymptotisch stabilisiert*, falls eine Funktion $\beta \in \mathcal{KL}$ existiert, so dass für je zwei Konstanten $R > \epsilon > 0$ ein $T_0 > 0$ existiert, so dass für alle $T \in (0, T_0]$ die Abtastlösungen $\varphi_T(t, x_0, F_T)$ die Abschätzung

$$\|\varphi_T(t, x_0, F_T)\| \leq \max\{\beta(\|x_0\|, t), \epsilon\} \tag{3.19}$$

für alle $t \geq 0$ und alle Anfangswerte $x_0 \in \mathbb{R}^n$ mit $\|x_0\| \leq R$ erfüllen. ◻

Bei der *semiglobalen praktischen asymptotischen Stabilität* bezieht sich die Bezeichnung “semiglobal” auf die den *Einzugsbereich* definierende Konstante R und die Bezeichnung “praktisch” auf die eine *Umgebung* um das Gleichgewicht definierende Konstante ϵ (vgl. Abbildung 3.4).

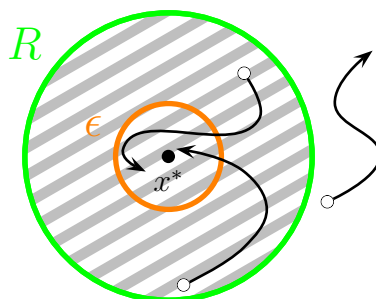


Abbildung 3.4: Die Lösungen zu Startpunkten innerhalb des Einzugsbereichs verlaufen nicht exakt in das Gleichgewicht x^* , sondern in eine ϵ -Umgebung von x^* .

Bemerkung 3.27

Bei Stabilisierung mittels Abtastung ist die (semiglobal) praktisch asymptotische Stabilität tatsächlich das Beste, das man für entlang der Lösungen stückweise konstante Feedbacks erwarten kann. Da ein vorgegebenes Feedback F nicht kontinuierlich ausgewertet wird, kann dies zu Einbußen in der Kontrollgenauigkeit führen: Zum einen nahe dem Gleichgewicht $x^* = 0$, da hier eine präzise Steuerung verlangt wird, zum anderen weit entfernt, da hier die Dynamik des Systems sehr schnell sein kann.

Beide Fälle benötigen ausreichend Informationen über den dortigen Verlauf der Lösungen des Systems und ziehen somit ein häufiges Messen und Auswerten des Feedbacks nach sich. Das wiederum bedeutet, dass die Abtastzeit T kleiner gewählt werden muss, was letztendlich zu Ungenauigkeiten in den Auswertungen führen kann. \circ

3.4.3 Abtastung und Ljapunov-Funktionen

Auch für den Fall der Abtastung können die Ljapunov-Funktionen ein sehr nützliches Hilfsmittel zur Konstruktion stabilisierender Feedbacks im Sinne von Definition 3.26 darstellen:

Definition 3.28 (Abtast-Ljapunov-Funktionen)

Betrachte ein Kontrollsystem (3.9) mit $f(0, 0) = 0$. Eine Familie von stetigen Funktionen $V_T : \mathbb{R}^n \rightarrow \mathbb{R}_0^+$ für $T \in (0, T^*]$ heißt *semiglobale praktische Familie von (Abtast-)Ljapunov-Funktionen*, falls Funktionen $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$, ein $C > 0$ und eine stetige Funktion $W : \mathbb{R}^n \rightarrow \mathbb{R}$ existieren, so dass die Ungleichungen

$$W(x) > 0, \tag{3.20}$$

$$\alpha_1(\|x\|) \leq V_T(x) \leq \alpha_2(\|x\|) \tag{3.21}$$

für alle $T \in (0, T^*]$ und alle $x \in \mathbb{R}^n \setminus \{0\}$ erfüllt sind und für alle Konstanten $C_2 > C_1 > 0$ ein $T_0 > 0$ existiert, so dass die Ungleichung

$$\inf_{u \in U_C} V_T(\varphi(T, x, u)) \leq \max\{V_T(x) - TW(x), C_1\} \tag{3.22}$$

für alle $x \in \mathbb{R}^n$ mit $V_T(x) \leq C_2$ und alle $T \in (0, T_0]$ gilt.

Das Infimum wird dabei über $U_C := \{u \in U \mid \|u\| \leq C\}$ gebildet. \square

Bemerkung 3.29

Das u in Ungleichung (3.22) ist keine messbare Kontrollfunktion aus \mathcal{U}_C , sondern ein konstanter Kontrollwert aus U_C . Das Infimum darin ist tatsächlich ein Minimum, da die Lösung $\varphi(T, x, u)$ stetig von $u \in U_C$ abhängt. \circ

Wie man nun aus den Ljapunov-Funktionen V_T stabilisierende Feedbacks im Sinne der Definition 3.26 erhalten kann, zeigt folgender Satz:

Satz 3.30

Betrachte ein Kontrollsystem (3.9) mit $f(0, 0) = 0$. Sei V_T für $T \in (0, T^*]$ eine semiglobale praktische Familie von Ljapunov-Funktionen. Dann ist

$$F_T(x) := \arg \min_{u \in U_C} V_T(\varphi(T, x, u))$$

eine Familie von *semiglobal praktisch asymptotisch stabilisierenden Feedbacks* im Sinne von Definition 3.26. \diamond

Beweis: Siehe hierzu [26], Satz 7.5. ◆

Bemerkung 3.31

Oft wird man die Familie von Ljapunov-Kontrollfunktionen V_T nicht explizit berechnen können oder wollen, um daraus wie im Satz beschrieben die Familie von Feedbacks F_T zu bestimmen. Vielmehr wird man in der Praxis versuchen, ein F_T zu berechnen und dann sicherzustellen, dass dazu ein passendes V_T existiert. Dies wiederum entspricht auch dem Vorgehen bei MPC-Schemata. ○

3.5 Optimale Steuerung

Um ein Kontrollsystem in einen gewünschten Endzustand zu überführen, kann es für eine dafür geeignete Kontrollfolge viele unterschiedliche Lösungen geben. Will man z.B. ein Auto auf gerader Strecke vom Punkt A in den Punkt B bewegen, könnte man das Auto bis zu einem Zeitpunkt stark beschleunigen und anschließend ausrollen lassen, so dass es im Punkt B stehen bleibt. Eine andere Lösung wäre das Auto mit konstanter Geschwindigkeit in den Punkt B zu bewegen und anschließend dort zu bremsen. Beide Lösungen führen zum gewünschten Endzustand, so dass beide akzeptabel erscheinen.

In der Praxis ist es aber zumeist der Fall, dass man das Erreichen eines Ziels unter dem Aspekt der Wirtschaftlichkeit betrachtet und z.B. daran interessiert ist, welche der beiden Lösungen einen möglichst schnell oder möglichst energiesparend ans Ziel bringt. Dieser Problematik widmet sich das Gebiet der *optimalen Steuerung*.

Dieses Kapitel orientiert sich inhaltlich weitestgehend an den Werken von F. TRÖLTZSCH [51] und R. VINTER [53], sowie an der Arbeit von C. BÜSKENS [9].

Optimale Steuerprozesse bewerten die Lösungen von Kontrollsystemen anhand eines Optimalitätskriteriums, des sogenannten *Zielfunktional*s. Wie ein solches Zielfunktional beschaffen ist, hängt von dem angestrebten Endzustand ab. Zwei bekannte Beispiele hierfür sind:

- *zeitoptimale Steuerung*: Erreichen des Endzustands in möglichst kurzer Zeit.
- *energieoptimale Steuerung*: Erreichen des Endzustands unter möglichst geringem Steuereinsatz.

Die Aufgabe der optimalen Steuerung ist demzufolge die Bestimmung einer (bezüglich der verwendeten Zielfunktion) optimalen Kontrolle unter den gegebenen Anforderungen.

Zu Beginn wollen wir den Begriff des Zielfunktional formal fassen.

Definition 3.32 (Kostenfunktion, Zielfunktional)

Sei $g : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}_0^+$ eine stetige nichtnegative *Kostenfunktion* und $u \in \mathcal{U}$ eine Kontrollfunktion.

Das *Zielfunktional* (oder auch *Kostenfunktional*) wird damit als

$$J(x_0, u) = \int_0^\infty g(x(t, x_0, u), u(t)) dt \quad (3.23)$$

definiert. □

Damit lässt sich nun das Optimalsteuerungsproblem mit zugehöriger optimaler Wertefunktion definieren (siehe auch [25]).

Definition 3.33 (Optimalsteuerungsproblem, Optimale Wertefunktion)

Das *Optimalsteuerungsproblem* ist gegeben durch

$$\text{Minimiere } J(x_0, u) \quad (3.24)$$

$$\text{u.d.N. } \dot{x}(t) = f(x(t), u(t)) \quad (3.25)$$

$$u(t) \in \mathcal{U} \quad (3.26)$$

$$x_0 \in \mathbb{R}^n \quad (3.27)$$

Die Funktion

$$V(x_0) := \inf_{u \in \mathcal{U}} J(x_0, u) \quad (3.28)$$

heißt *optimale Wertefunktion* des optimalen Steuerungsproblems.

Das Paar

$$(x^*, u^*) \text{ mit } J(x^*, u^*) = V(x^*) \quad (3.29)$$

heißt *optimales Paar*. □

In den folgenden Abschnitten widmen wir uns nun dem Lösen des genannten Optimalsteuerungsproblems. Dazu formulieren wir zunächst eine wichtige Aussage in Bezug auf optimale Trajektorien und klassifizieren im Anschluss verschiedene Lösungsansätze für dieses Problem.

3.5.1 Bellman'sches Optimalitätsprinzip

Das *Optimalitätsprinzip von Bellman*, benannt nach dem US-amerikanischen Mathematiker R. BELLMAN (1920 – 1984), stellt ein grundlegendes Prinzip der Lösung von optimalen Steuerungsproblemen dar und beschreibt eine Charakterisierung der Wertefunktion V . Des Weiteren bildet es die Basis vieler Methoden der dynamischen Programmierung, weshalb es auch als *Prinzip der dynamischen Programmierung* bezeichnet wird.

Die Kernaussage dabei ist, dass die Endstücke optimaler Trajektorien wieder optimale Trajektorien sind, d.h. dass bei Vorliegen einer optimalen Steuerung zu jedem Zeitpunkt optimal gesteuert wird.

Satz 3.34 (Bellman'sches Optimalitätsprinzip)

1. Für die optimale Wertefunktion $V(x_0)$ gilt für jedes $\tau > 0$

$$V(x_0) = \inf_{u \in \mathcal{U}} \left\{ \int_0^\tau g(x(t, x_0, u), u(t)) dt + V(x(\tau, x_0, u)) \right\} \quad (3.30)$$

2. Für ein optimales Paar (x^*, u^*) gilt für jedes $\tau > 0$

$$V(x^*) = \int_0^\tau g(x(t, x^*, u^*), u(t)) dt + V(x(\tau, x^*, u^*)) \quad (3.31)$$

◇

Ein ausführlicher Beweis des Satzes findet sich in [25], Satz 5.2.

3.5.2 Klassen von Lösungsansätzen für Optimalsteuerungsprobleme

Aufgrund des großen Umfangs und der weitreichenden und vielfältigen Forschungen auf dem Gebiet der Optimalsteuerung sind eindeutige Zuordnungen und Klassifizierungen z.B. von Methoden, Verfahren oder Charakteristiken zu bestimmten Teilgebieten wegen der oftmals mehr oder weniger stark verschwimmenden Grenzen sehr schwer zu realisieren. So verhält es sich auch bezüglich der Lösungsansätze von Optimalsteuerungsproblemen, da diese je nach betrachtetem Teilgebiet unterschiedlich gegliedert und zugeordnet werden. Infolgedessen stellte Herr Prof. Grüne eine für diese Arbeit sinnvolle Klassifizierung bereit, die im Nachfolgenden Verwendung findet.

Es gibt zwei grundsätzlich verschiedene Klassen von Lösungsansätzen zur optimalen Steuerung, die wir hier als "globale" und "lokale" Verfahren unterscheiden wollen.

3.5.2.1 Globale Verfahren

Die globalen Verfahren berechnen eine optimale Lösung für alle Anfangszustände $x_0 \in X$ auf einmal, wobei $X \subset \mathbb{X}$ eine Teilmenge des Zustandsraumes \mathbb{X} ist.

Dies kann auf zwei Weisen geschehen:

- Berechnung der optimalen Wertefunktion $V(x)$ für alle $x \in X$, oder
- Berechnung der optimalen Kontrolle in Feedbackform $F(x)$ für alle $x \in X$.

Bei den klassischen globalen Verfahren finden vor allem Methoden der *dynamischen Programmierung* Verwendung. Dies kann analytisch über die Herleitung einer bestimmten Gleichung geschehen, deren Lösung den optimalen Wert ergibt (siehe Hamilton-Jacobi- oder Riccati-Gleichungen bei z.B. linear-quadratischen Problemen), oder auch numerisch auf verschiedenen Zerlegungen von X geschehen.

3.5.2.2 Lokale Verfahren

Die lokalen Verfahren berechnen für einen gegebenen Anfangszustand $x_0 \in \mathbb{X}$ die zugehörige optimale Trajektorie. Sie sind also "lokal" in dem Sinne, dass sie das Optimalsteuerungsproblem jeweils nur für einen Anfangszustand lösen.

Des Weiteren findet bei den lokalen Verfahren eine weitere Unterscheidung in *indirekte* und *direkte Methoden* statt:

Indirekte Methoden verwenden zumeist Varianten des Pontryagin'schen Maximumprinzips (benannt nach dem russischen Mathematiker LEV S. PONTRYAGIN, 1908 – 1988). Dabei liefert ein abstraktes Optimalitätskriterium die hinreichenden Bedingungen des Maximumprinzips in Form eines Randwertproblems, das analytisch oder anhand einer anschließenden Diskretisierung auch numerisch gelöst wird. Ebenfalls zur Anwendung kommen hierbei Methoden der dynamischen Programmierung, die ebenfalls ein abstraktes Optimalitätskriterium liefern.

Direkte Methoden diskretisieren zunächst das Problem und formen dadurch das Optimalsteuerungsproblem in ein *statisches Optimierungsproblem* um. Dieses löst man anschließend mit geeigneten numerischen Verfahren wie z.B. SQP-Verfahren. Der Vorteil des direkten Diskretisierens ist, dass damit keine Vorkenntnisse der Lösungsstruktur und keine Schätzungen bezüglich kritischer Punkte notwendig sind. Des Weiteren kann auf die Berechnung der adjungierten Differentialgleichung verzichtet werden.

Bemerkung 3.35

Das direkte Verfahren findet Verwendung in der für die Auswertung verwendeten Implementierung eines MPC-Verfahrens von J. PANNEK.

Dies hat vor allem den Grund, dass die Anwendung der indirekten Verfahren auf große komplexe Prozesse aufgrund der benötigten komplizierten mathematischen Beschreibungen und Formulierungen stark begrenzt ist, während die direkten Verfahren durch ihre einfachen Berechnungen und der hohen Effizienz beim Lösen großer Probleme diesbezüglich klar im Vorteil sind. ◦

3.6 Die modellprädiktive Regelung

Bei der modellprädiktiven Regelung handelt es sich nicht etwa um eine einzelne Methode, sondern vielmehr um einen Oberbegriff, der eine *Sammlung von Methoden* zur Kontrolle komplexer Prozesse beschreibt. Im Kern basieren all diese Methoden auf einem bestimmten Vorgehen, das nun im Folgenden näher beschrieben und erläutert werden soll (vgl. [10]).

3.6.1 MPC Verfahren im Allgemeinen

Beim modellprädiktiven Regelungsansatz wird der zu regelnde Prozess oder das zu regelnde System durch ein (möglichst einfaches⁴) diskretes dynamisches Modell dargestellt. Anhand dieses Modells werden die zukünftigen Zustände des Prozesses in Abhängigkeit der Eingangsgrößen auf einem bestimmten *Zeithorizont* berechnet. Basierend auf dieser Vorhersage (oder *Prädiktion*) wird die Berechnung von optimalen Eingangsgrößen (unter Berücksichtigung verschiedener Eingangs-, Ausgangs- bzw. End- und Zustandsbeschränkungen) ermöglicht. Die Optimierung der Eingangsgrößen — hier entsprechen diese einer *Kontrollfolge* — erfolgt bezüglich einer geeigneten Gütefunktion (*optimales Steuerungsproblem*).

Anschließend wird der Horizont um eine bestimmte Anzahl an Zeitschritten verschoben (*receding horizon strategy*) und die Optimierung auf dem neuen Zeitabschnitt wiederholt.

Eine Veranschaulichung des beschriebenen Ablaufs liefert das Blockdiagramm (entnommen aus [10], S.4) in Abbildung 3.5.

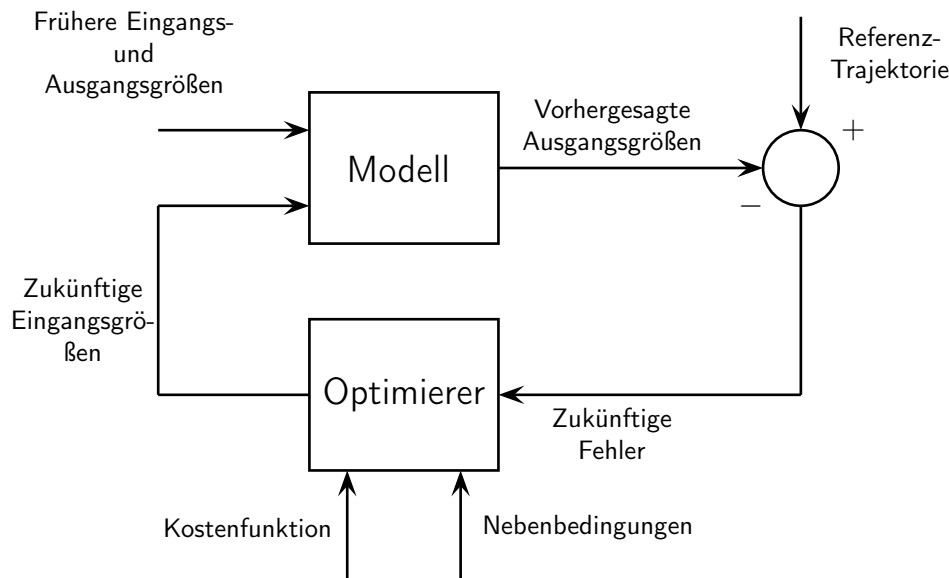


Abbildung 3.5: Struktur eines MPC-Verfahrens.

⁴Im Englischen *fit for purpose*: Es sollte das einfachste Modell verwendet werden, das ausreichend exakte Vorhersagen für den Systemverlauf liefert.

3.6.2 Methodik und Strategie

Wir wollen nun auf die charakteristischen Schritte, die der Methodik der verschiedenen Regelungsansätze zugrunde liegen, im Einzelnen etwas genauer eingehen und graphisch veranschaulichen.

Dabei lässt sich die folgende Strategie erkennen (vgl. [10]):

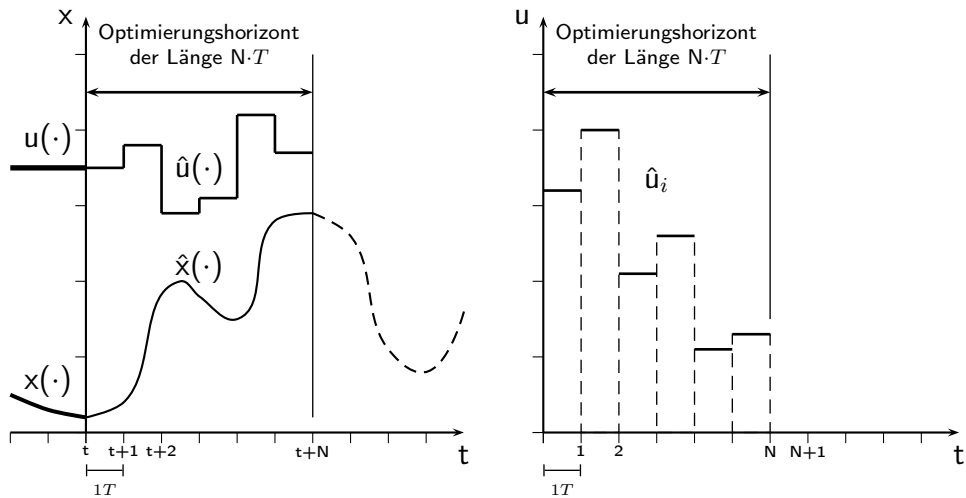
1. Das (zukünftige) Systemverhalten wird für eine dem Optimierungshorizont N entsprechende Anzahl an Zeitschritten unter Verwendung des Systemmodells vorhergesagt. Diese vorhergesagten Ausgangsgrößen hängen dabei
 - von den zum aktuellen Zeitpunkt bekannten Werten (sowohl Eingangs- wie auch Ausgangsgrößen) und
 - von den zukünftigen Kontrollgrößenab.
2. Die zukünftigen Kontrollgrößen werden durch das Lösen eines optimalen Steuerungsproblems berechnet. Diese Kontrollwerte sind insofern optimal, als dass sie versuchen, den Verlauf der Lösung möglichst nahe entlang der Referenztrajektorie zu halten.
3. Es wird die erste Kontrollgröße aus der berechneten optimalen Kontrollfolge angewendet, die restlichen Kontrollgrößen werden verworfen⁵.
4. Der Optimierungshorizont N wird verschoben und der Durchlauf für den neuen Zeitabschnitt wiederholt.

Dieser Vorgang wird in Abbildung 3.6 noch einmal bildlich dargestellt.

3.7 Das nichtlineare MPC-Problem

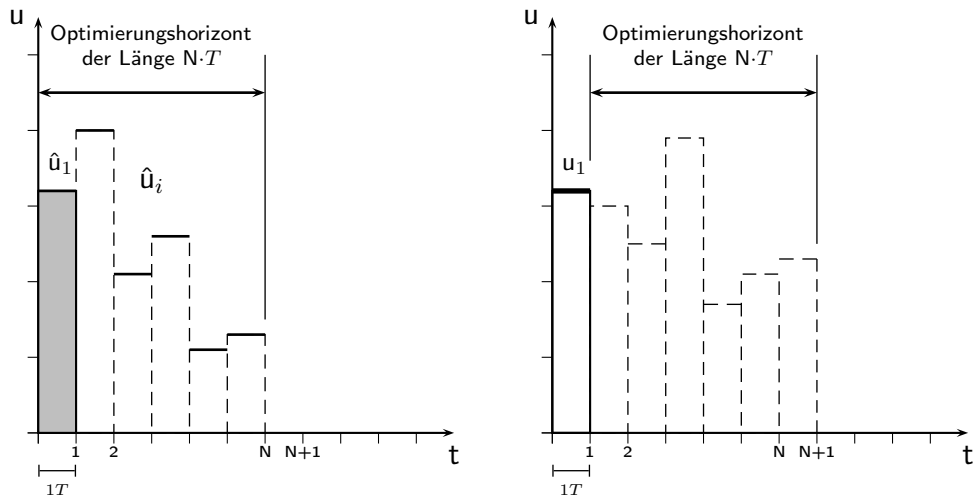
In diesem Kapitel soll näher auf die elementaren Bestandteile des MPC-Verfahrens für die in dieser Arbeit behandelten nichtlinearen diskreten Systeme eingegangen und das daraus resultierende *nichtlineare modellprädiktive Regelungsproblem* formuliert werden. Dabei spielt neben dem Systemmodell und der Zielfunktion auch das sogenannte *m-Schritt Feedback* eine wichtige Rolle, womit sich der nun folgende Abschnitt beschäftigt (vgl. hierzu [9, 24, 48]).

⁵Wieviele der berechneten Kontrollgrößen tatsächlich implementiert bzw. verworfen werden, hängt vom sogenannten *m-Schritt Regler* ab, vgl. Definition 3.40.



(a) Das zukünftige Systemverhalten $\hat{x}(\cdot)$ wird bezüglich der zukünftigen Kontrollen $\hat{u}(\cdot)$ und dem bisherigen Systemverlauf $x(\cdot)$ auf dem Horizont der Länge $N \cdot T$ vorhergesagt. Dabei entspricht T der festgelegten Abtastzeit.

(b) Die zukünftigen Kontrollgrößen $\hat{u}_i \in U$ ($i = 0, \dots, N - 1$) werden durch das Lösen des optimalen Steuerungsproblems berechnet.



(c) Es wird nur die erste aus der Optimierung resultierende Kontrolle \hat{u}_1 implementiert. Die restlichen Kontrollen \hat{u}_i für $i = 1, \dots, N - 1$ werden verworfen.

(d) Anschließend wird der Horizont um $1 \cdot T$ verschoben und das Prozedere auf dem neuen Zeitabschnitt wiederholt.

Abbildung 3.6: Ablauf der *receding horizon strategy*.

3.7.1 Grundlegende Komponenten und Definitionen

Wir wollen in diesem Abschnitt die grundlegenden Begriffe und Bestandteile des MPC-Verfahrens angeben bzw. gegebenenfalls herleiten. Diese bilden nicht nur die Grundlage für die anschließende Formulierung des nichtlinearen MPC-Problems, sondern ebenso für die im nächsten Abschnitt behandelte Analyse von MPC-Schemata.

Im Folgenden bezeichnet $n \in \mathbb{N}_0$ die diskreten Zeitpunkte, $x(n) \in \mathbb{X}$ die Zustandsvariablen aus dem Zustandsraum, sowie $u(n) \in U$ die Kontrollvariablen aus dem Kontrollwertebereich. Die Lösungstrajektorie zu einem $u \in \mathcal{U}$ bezeichnen wir mit $x_u(n)$. Der Zustandsraum \mathbb{X} ist ein beliebiger metrischer Raum und kann somit von einer endlichen Menge bis hin zu einem unendlich dimensionalen Raum reichen.

□ Das Systemmodell

Das nichtlineare Systemmodell in diskreter Zeit ist gegeben durch

$$x(n+1) = f(x(n), u(n)), \quad x(0) = x_0. \quad (3.32)$$

Die Modellbildung wird hier anhand des Zustandsraums \mathbb{X} (*state space model*) vorgenommen, was u. a. den Vorteil mit sich bringt, dass das Modell auf einfache Weise ebenfalls für Systeme mehrerer Variablen verwendet werden kann. Erläuterungen dazu sowie andere verbreitete Prozessmodellierungen wie *impulse response model*, *step response model* oder *transfer function model* finden sich in [10], Kapitel 2.

Das benötigte Abtast-System erhält man durch das Festlegen der Abtastzeit $T > 0$ und durch Setzen von

$$f(x, u) := \varphi(T, x, \tilde{u}) \quad (3.33)$$

mit $\tilde{u}(t) \equiv u$. Dabei entspricht $\varphi(t, x_0, \tilde{u})$ der Lösung der kontrollierten Differentialgleichung $\dot{x}(t) = g(x(t), \tilde{u}(t))$ zum Startwert x_0 .

Damit gilt: Für jede Kontrollfunktion $u \in \mathcal{U}$ in diskreter Zeit genügen die Lösungstrajektorien x_u von (3.32) und (3.33) der Gleichung

$$x_u(n) = \varphi(nT, x_0, \tilde{u})$$

für die auf den Intervallen $[nT, (n+1)T)$ stückweise konstanten Kontrollfunktionen in kontinuierlicher Zeit $\tilde{u} : \mathbb{R} \rightarrow U$ mit $\tilde{u}|_{[nT, (n+1)T)} \equiv u(n)$.

Bemerkung 3.36

Wir erreichen somit durch diese Konstruktion, dass die diskrete Zeit n mit der kontinuierlichen Zeit $t = nT$ übereinstimmt. Damit genügt es also für die Kontrolle der Systeme auf T stückweise konstante Abtast-Feedbacks zu berechnen. ◦

Unser Ziel ist es nun ein Gesetz für die Feedback-Kontrolle zu finden, die die Kosten auf unendlichem Horizont

$$J_\infty(x_0, u) = \sum_{n=0}^{\infty} l(x_u(n), u(n)) \quad (3.34)$$

mit den laufenden Kosten $l : X \times U \rightarrow \mathbb{R}_{0+}$ minimiert. Die *optimale Wertefunktion* für dieses Problem lautet

$$V_\infty(x_0) = \inf_{u \in \mathcal{U}} J_\infty(x_0, u).$$

Allerdings kann man V_∞ im Allgemeinen nicht angeben, da man das Infimum über den Raum der Kontrollfunktionen aufgrund seiner Unendlichkeit nicht berechnen kann. Denn wäre V_∞ bekannt, so wäre es leicht unter Verwendung von *Bellman's Optimalitätsprinzip* (vgl. [24], S.4, und [25], Kapitel 5) ein optimales Feedback anzugeben.

□ Die Zielfunktion

Aus diesem Grund versuchen wir mit dem Ansatz des Verschiebens des Horizonts näherungsweise optimale Kontrollen zu berechnen, indem wir das *Funktional auf endlichem Horizont*

$$J_N(x_0, u) = \sum_{n=0}^{N-1} l(x_u(n), u(n)) \quad \text{für } N \in \mathbb{N}_0 \quad (3.35)$$

mit den *laufenden Kosten* $l : X \times U \rightarrow \mathbb{R}_0^+$ als Zielfunktion minimieren. Die optimale Wertefunktion hierzu lautet

$$V_N(x_0) = \inf_{u \in \mathcal{U}} J_N(x_0, u). \quad (3.36)$$

Bemerkung 3.37

Es sei angemerkt, dass es sich hierbei um den einfachsten der MPC- bzw. RHC-Ansätze handelt, d.h. ohne Berücksichtigung von Endkosten und Endbeschränkungen. ◻

□ Die Feedback-Kontrolle

Es stellt sich die Frage, wie eine entsprechende Feedback-Kontrolle aussieht. Dazu zunächst die folgende allgemeine Definition.

Definition 3.38 (*m*-Schritt Feedback)

Für $m \geq 1$ ist ein *m-Schritt Feedback* eine Abbildung $\mu : X \times \{0, \dots, m-1\} \rightarrow U$, die nach folgender Regel angewendet wird:

$$x_\mu(n+1) = f(x_\mu(n), \mu(x_\mu([n]_m), n - [n]_m)), \quad x_\mu(0) = x_0. \quad (3.37)$$

Dabei entspricht $[n]_m$ dem größtmöglichen Produkt km , $k \in \mathbb{Z}$, mit $km \leq n$. ◻

Mittels des Feedbacks μ erhält man also eine Kontrollfolge von m Kontrollwerten u für ein System, indem man das Feedback an den diskreten Zeitpunkten $0, m, 2m, \dots$ berechnet und anschließend die erhaltene Kontrollfolge in den m Schritten bis zur nächsten Auswertung anwendet. Wir wollen dies an einem Beispiel veranschaulichen:

Beispiel 3.39 (Berechnung einer Kontrollfolge mittels Feedback)

Gegeben sei ein m -Schritt Feedback nach Definition 3.38. Dann gilt für die Berechnung der Kontrollwerte u an den diskreten Zeitpunkten $0, m, 2m, \dots$ mit

$$x(i) = f(x_\mu(i-1), \mu(x(i \bmod m), i \bmod m - 1))$$

für alle $i \in \mathbb{N}$:

$$\begin{aligned} u(0) &= \mu(x_0, 0) \quad \text{mit } x_0 = x_\mu(0) \\ u(i) &= \mu(x_0, i) \quad \text{für } i = 1, \dots, m-2 \\ &\vdots \\ u(m-1) &= \mu(x_0, m-1) \\ u(m) &= \mu(x(m), 0) \\ u(m+1) &= \mu(x(m), 1) \\ u(m+2) &= \mu(x(m), 2) \\ &\vdots \\ u(2m) &= \mu(x(2m), 0) \\ &\vdots \end{aligned}$$

Es werden die ersten m Kontrollwerte $u(0), \dots, u(m-1)$ zum Startpunkt x_0 berechnet und die daraus erhaltene Kontrollfolge angewendet, um anschließend die nächsten m Kontrollwerte $u(m), \dots, u(2m-1)$ zu erhalten. Dazu benötigt man die bereits berechnete Kontrolle $u(m-1)$, da sonst $x(m)$ für die Berechnung dieser Kontrollwerte nicht angegeben werden kann. •

Jetzt muss man nur noch dafür sorgen, dass die Kontrollfolge das Kostenfunktional minimiert. Dies erreichen wir dadurch, dass wir das Feedback in Abhängigkeit von N und m und über eine optimale Kontrollfolge bezüglich unserer optimalen Wertefunktion (3.36) auf endlichem Horizont definieren:

Definition 3.40 (m -Schritt MPC-Feedback)

Sei u^* eine minimierende Kontrollfolge für (3.35) und Startwert x_0 . Das m -Schritt MPC-Feedback wird definiert durch

$$\mu_{N,m}(x_0, n) = u^*(n), \quad n = 0, \dots, m-1.$$

Dabei ist hier N der *Optimierungshorizont* und m der *Kontrollhorizont*. □

Bemerkung 3.41

Für $m = 1$ erhält man den Fall des *1-Schritt-Reglers* wie in Abbildung 3.6 dargestellt. Dieser liegt auch der für die Auswertungen verwendeten Implementierung des MPC-Verfahrens von J. PANNEK zugrunde. ○

3.7.2 Formulierung des NMPC-Problems

Wir wollen nun das Problem der optimalen Kontrolle eines Systems mit MPC formulieren, indem wir zusammenfassend die Resultate aus dem vorhergehenden Abschnitt verwenden.

Gesucht wird eine das Zielfunktional über endlichem Horizont J_N minimierende Kontrollfolge μ_N aus der Menge der zulässigen Kontrollfunktionen \mathcal{U} mittels dem Verfahren der modellprädiktiven Regelung mit 1-Schritt-Regler. Das dafür zu lösende Problem lässt sich wie folgt definieren:

Definition 3.42 (Nichtlineares modellprädiktives Kontrollproblem)

Wir nennen die durch

$$\text{Finde } \mu_N^j = \arg \min_u J_N(x_0, u) \quad (3.38)$$

$$\text{u.d.N. } J_N(x_0, u) := \sum_{n=0}^{N-1} l(x(n), u(n)) \quad (3.39)$$

$$x(n+1) = f(x(n), u(n)) \quad (3.40)$$

$$x(0) = x_0^j \quad (3.41)$$

$$u(n) \in U \quad \text{für } n \in [0, N-1] \quad (3.42)$$

$$x(n) \in \mathbb{X} \quad \text{für } n \in [0, N] \quad (3.43)$$

mit gegebenem x_0^0 und $x_0^j := f(x_0^{j-1}, \mu_N^{j-1}(0))$

gegebenen optimalen Steuerungsprobleme ein *nichtlineares modellprädiktives Kontrollproblem (NMPC-Problem)*. \square

Es werden also für den Erhalt einer geeigneten Kontrollfolge nicht ein einziges, sondern mehrere Optimalsteuerungsprobleme gelöst. Die dadurch erhaltenen Lösungen liefern uns schließlich unsere Lösung für das NMPC-Problem in folgendem Sinne:

$$\mu_N := (\mu_N^j(0))_{j \in \mathbb{N}}.$$

Die Nichtlinearität rührt von dem zugrundeliegenden nichtlinearen Systemmodell in diskreter Zeit in (3.40) her.

Bemerkung 3.43

Man beachte, dass es sich bei den dem NMPC-Problem zugrunde liegenden Optimierungsproblemen um open loop Probleme handelt, also somit der aktuelle Zustand *nicht* in die Berechnung der Kontrollen mit einfließt. Allerdings wird durch das Anwenden der ersten berechneten optimalen Kontrolle und dem Verschieben des Optimierungshorizonts sozusagen ein closed loop System “simuliert”. \circ

3.7.3 Notwendigkeit eines endlichen Horizonts für NMPC

H. VOIT folgert in [54] mittels des *Bellman’schen Optimalitätsprinzips*, dass für einen unendlichen Optimierungshorizont $N = \infty$ die Lösungen des open loop Systems den

Lösungen des closed loop Systems unter NMPC-Kontrolle entsprechen. Dies konkretisiert er in dem folgenden Satz⁶, der besagt:

Angenommen f aus Definition 2.14 (entspricht hier Definition 3.42, Anm. d. A.) ist zweimal stetig differenzierbar und die 0 ist ein Gleichgewicht von f . Wenn das open loop Problem für $n = 0$ zulässig ist, d.h. es gibt eine optimale Kontrollfolge, dann ist das closed loop NMPC Kontroll Problem asymptotisch stabil.

Man erhält also asymptotische Stabilität, wenn man den Optimierungshorizont N gegen Unendlich gehen lässt. Dieses Problem ist in der Praxis allerdings nur für *lineare* Regelungskreise lösbar, da hierfür sehr effiziente Algorithmen zum Lösen von dabei auftretenden sehr großen quadratischen Optimierungsproblemen existieren. Für die in dieser Arbeit behandelten *nichtlinearen* Systeme sind derart große Probleme entweder nur unter sehr hohem Rechenaufwand (und den damit verbundenen langen Rechenzeiten), oder gar nicht zu lösen.

Aus diesem Grund können für nichtlineare MPC-Probleme nur *endliche* Optimierungshorizonte $N < \infty$ verwendet werden.

3.8 Analyse der MPC-Schemata

Unser Ziel ist es, mittels eines modellprädiktiven Regelungsansatzes dahingehend Einfluss auf das zugrunde liegende System zu nehmen, so dass wir für dieses Stabilität erhalten. Wie wir bereits festgestellt haben, ist hierfür die Wahl des *Optimierungshorizonts* N besonders entscheidend: Dieser soll aus kosten- und rechentechnischen Gründen nach Möglichkeit klein sein.

Um dies zu erreichen, spielt die *Zielfunktion* J_N und, mit dieser ganz entscheidend, die Wahl der *Funktion der laufenden Kosten* l eine tragende Rolle.

Wie so ein Entwurf von l für ein stabilisierendes MPC-Schema mit möglichst kleinem Horizont aussieht, wollen wir in diesem Kapitel erarbeiten (vgl. hierzu [24]).

Dazu verwenden wir in erster Linie die bereits eingeführten Begriffe und Definitionen aus Abschnitt 3.7.1. Um Abschätzungen für die *Suboptimalität* von Feedback-geregelten Systemen und für den *Grad an Suboptimalität* des m -Schritt MPC-Feedbacks $\mu_{N,m}$ zu erhalten, wird noch eine weitere Wertefunktion benötigt:

Definition 3.44

Für auf unendlichem Horizont mittels einem Feedback μ geregelte Systeme definieren wir eine Wertefunktion

$$V_{\infty}^{\mu}(x_0) := \sum_{n=0}^{\infty} l(x_{\mu}(n), \mu(x_{\mu}([n]_m), n - [n]_m)). \quad (3.44)$$

Dabei ist $x_{\mu}(n)$ die zugehörige Lösungstrajektorie. □

⁶siehe [54], S.22, Satz 2.5

3.8.1 Asymptotische Kontrollierbarkeit

Wir wollen nun den Begriff der asymptotischen Kontrollierbarkeit auf das Verfahren der modellprädiktiven Regelung übertragen. Dazu werden wir die Kontrollierbarkeitsannahme nicht bezüglich der Lösungstrajektorie x_u , sondern bezüglich der laufenden Kosten $l(x_u, n)$ entlang der Trajektorie formulieren.

Des Weiteren definieren wir

$$l^*(x) := \min_{u \in \mathcal{U}} l(x, u)$$

als die *minimalen laufenden Kosten* bezüglich einem Kontrollwert $u \in \mathcal{U}$.

Definition 3.45 (Asymptotische Kontrollierbarkeit)

Sei x^* ein Gleichgewicht mit einer offenen Umgebung X . Angenommen es gibt ein $\beta \in \mathcal{KL}_0$, so dass für alle $x \in X$ eine Kontrollfunktion $u_x \in \mathcal{U}$ existiert, so dass gilt

$$l(x_{u_x}(n), u_x(n)) \leq \beta(l^*(x), n) \quad \forall n \in \mathbb{N}_0. \quad (3.45)$$

Dann heißt das Gleichgewicht x^* *asymptotisch kontrollierbar*. □

Bemerkung 3.46

Spezialfälle für $\beta \in \mathcal{KL}_0$ sind

- *exponentielle Kontrollierbarkeit:*

$$\beta(r, n) = C\sigma^n r \quad (3.46)$$

für reelle Konstanten $C \geq 1$ und $\sigma \in (0, 1)$.

- *Kontrollierbarkeit in endlicher Zeit:*

$$\beta(r, n) = c_n r \quad (3.47)$$

für eine reelle Folge $(c_n)_{n \in \mathbb{N}_0}$ mit $c_n \geq 0$ und $c_n = 0$ für alle $n \geq n_0$. ◦

Abschließend definieren wir noch eine vereinfachende Schreibweise B_N für die Summe über die Vergleichsfunktionen $\beta \in \mathcal{KL}_0$

$$B_N(r) := \sum_{n=0}^{N-1} \beta(r, n) \quad (3.48)$$

für beliebiges $r \geq 0$ und $N \geq 1$.

3.8.2 Erste Resultate

Unter Verwendung der Definitionen und Begriffe aus den beiden vorhergehenden Abschnitten wollen wir nun erste Resultate formulieren, die uns beim Entwurf einer geeigneten Kostenfunktion l von Nutzen sein werden.

Es sollen hier nur die dafür elementaren Aussagen genannt werden, ohne dabei auf die ausführlichen Hinführungen oder Beweise einzugehen. Diese können in [24] oder [54] nachgelesen werden.

Zu Anfang wird eine Abschätzung für die Qualität von $V_\infty^{\tilde{\mu}}$ angegeben:

Lemma 3.47

Betrachte das m -Schritt Feedback $\tilde{\mu} : X \times \{0, \dots, m-1\} \rightarrow U$, die zugehörige Lösung $x_{\tilde{\mu}}(k)$ mit $x_{\tilde{\mu}}(0) = x_0$ und eine Funktion $\tilde{V} : X \rightarrow \mathbb{R}_0^+$, die für ein $\alpha \in (0, 1]$ und für alle $x_0 \in X$ der folgenden Ungleichung genügt:

$$\tilde{V}(x_0) \geq \tilde{V}(x_{\tilde{\mu}}(m)) + \alpha \sum_{k=0}^{m-1} l(x_{\tilde{\mu}}(k), \tilde{\mu}(x_0, k)). \quad (3.49)$$

Dann gilt die folgende Abschätzung

$$\alpha V_{\infty}(x) \leq \alpha V_{\infty}^{\tilde{\mu}}(x) \leq \tilde{V}(x) \quad (3.50)$$

für alle $x \in X$. ◇

Beweis: Siehe hierzu [24], Proposition 2.4. ◆

Man hat also mit $\tilde{V}(x)$ eine obere Schranke für $V_{\infty}^{\tilde{\mu}}(x)$ und damit für $V_{\infty}(x)$, deren Güte von $\alpha \in (0, 1]$ abhängt. Dieses α lässt sich mittels eines Optimierungsproblems berechnen.

Die nächsten Aussagen sind Resultate aus der asymptotischen Kontrollierbarkeit unter der folgenden Annahme:

Annahme 3.48

Für ein $\beta \in \mathcal{KL}_0$ und für alle $x \in X$ ist das System asymptotisch kontrollierbar gemäß Definition 3.45.

Wir formulieren einen weiteren Hilfssatz:

Lemma 3.49

Es gelte Annahme 3.48. Wir betrachten einen Optimierungshorizont $N \geq 1$, $m \in \{1, \dots, N-1\}$, eine Folge $\lambda_n > 0$, $n = 0, \dots, N-1$ und einen Wert $\nu > 0$. Es sei $x_0 \in X$ und es existiere eine optimale Kontrollfolge $u^* \in \mathcal{U}$ für (3.36) mit Optimierungshorizont N , so dass

$$\lambda_n = l(x_{u^*}(n), u^*(n)), \quad n = 0, \dots, N-1$$

und $\nu = V_N(x_{u^*}(m))$ gilt. Dann gelten die Ungleichungen

$$\sum_{n=k}^{N-1} \lambda_n \leq B_{N-k}(\lambda_k), \quad k = 0, \dots, N-2 \quad (3.51)$$

$$\nu \leq \sum_{n=0}^{j-1} \lambda_{n+m} + B_{N-j}(\lambda_{j+m}), \quad j = 0, \dots, N-m-1 \quad (3.52)$$

◇

Beweis: Siehe hierzu [24], Proposition 4.1. ◆

Damit lässt sich nun Lemma 3.47 für das m -Schritt MPC-Feedback $\mu_{N,m}$ formulieren:

Satz 3.50

Betrachte $\beta \in \mathcal{KL}_0$, $N \geq 1$, $m \in \{1, \dots, N-1\}$. Wir nehmen an, dass alle Folgen $\lambda_n > 0$, $n = 0, \dots, N-1$ und alle Werte $\nu > 0$, die den Ungleichungen (3.51) und (3.52) genügen, die folgende Ungleichung

$$\sum_{n=0}^{N-1} \lambda_n - \nu \geq \alpha \sum_{n=0}^{m-1} \lambda_n \quad (3.53)$$

für ein $\alpha \in (0, 1]$ erfüllen.

Dann sind für alle optimalen Steuerungsprobleme (3.32) und (3.36), für die die Annahme 3.48 gilt, alle Voraussetzungen von Lemma 3.47 für das m -Schritt MPC-Feedback $\mu_{N,m}$ erfüllt. Des Weiteren gilt die Ungleichung

$$\alpha V_\infty(x) \leq \alpha V_\infty^{\mu_{N,m}}(x) \leq V_N(x) \quad (3.54)$$

für alle $x \in X$. ◇

Beweis: Siehe hierzu [24], Theorem 4.2. ◆

Wir erhalten damit eine Aussage, unter welchen Voraussetzungen ein optimales Steuerungsproblem die Suboptimalitätsabschätzung erfüllt.

Um sicher zu gehen, dass all diese Voraussetzungen an λ_n und ν erfüllt werden, verwendet man die Ungleichungen (3.51) und (3.52) zusammen mit den Nichtnegativitätsbedingungen $\lambda_n > 0$ und $\nu > 0$ als Nebenbedingungen sowie die Ungleichung (3.53) als Zielfunktion und erhält damit das folgende nichtlineare Optimierungsproblem:

Problem 3.51

Sei $\beta \in \mathcal{KL}_0$ gegeben, $N \geq 1$ und $m \in \{1, \dots, N-1\}$. Berechne

$$\alpha := \inf_{\lambda_0, \dots, \lambda_{N-1}, \nu} \frac{\sum_{n=0}^{N-1} \lambda_n - \nu}{\sum_{n=0}^{m-1} \lambda_n}$$

unter den Nebenbedingungen

- $\lambda_0, \dots, \lambda_{N-1} > 0$
- $\nu > 0$
- $\sum_{n=k}^{N-1} \lambda_n - B_{N-k}(\lambda_k) \leq 0 \quad k = 0, \dots, N-2$
- $\nu - \sum_{n=0}^{j-1} \lambda_{n+m} - B_{N-j}(\lambda_{j+m}) \leq 0 \quad j = 0, \dots, N-m-1$

Ist der Zielfunktionswert α aus dem halboffenen Intervall $(0, 1]$, so sind sämtliche Voraussetzungen des Satz 3.50 erfüllt. Damit gelten auch die Abschätzungen zur Suboptimalität.

3.8.3 Asymptotische Stabilität

Wir wissen nun zwar, dass für einen aus Problem 3.51 resultierenden Zielfunktionswert α aus $(0, 1]$ die Abschätzungen aus Satz 3.50 gelten, jedoch nicht, ob das System asymptotisch stabil ist. Unter welchen Voraussetzungen dies der Fall ist, erreichen wir durch eine Untersuchung der asymptotischen Stabilität des Kerns von l^* und formulieren dazu eine weitere Annahme:

Annahme 3.52

Es existiert eine abgeschlossene Menge $A \subset X$ mit folgenden Eigenschaften:

1. Für alle $x \in A$ existiert ein $u \in U$ mit:

- $f(x, u) \in A$
- $l(x, u) = 0$

Das bedeutet, dass wir in der Menge A bleiben, ohne Kosten zu verursachen.

2. Es existieren Vergleichsfunktionen $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$, so dass die Ungleichung

$$\alpha_1(\|x\|_A) \leq l^*(x) \leq \alpha_2(\|x\|_A) \quad (3.55)$$

für alle $x \in X$ mit $\|x\|_A := \min_{y \in A} \|x - y\|$ gilt.

Wir formulieren nun das Hauptresultat bezüglich der asymptotischen Stabilität im folgenden Satz.

Satz 3.53

Sei $\beta \in \mathcal{KL}_0$ gegeben, $N \geq 1$, sowie $m \in \{1, \dots, N - 1\}$. Wir nehmen an, dass das Optimierungsproblem 3.51 einen optimalen Zielfunktionswert α aus $(0, 1]$ besitzt. Das m -Schritt MPC-Feedback $\mu_{N,m}$ stabilisiert die Menge A asymptotisch für jedes Optimalsteuerungs-Problem (3.32) und (3.36), das die Annahmen 3.48 und 3.52 erfüllt.

Des Weiteren ist V_N eine entsprechende m -Schritt *Ljapunov-Funktion*, für die

$$V_N(x_{\mu_{N,m}}(m)) \leq V_N(x) - \alpha V_m(x) \quad (3.56)$$

gilt.

V_N ist dabei gemäß Definition 3.28 für eine durch das Abtastsystem festgesetzte Abtastperiode⁷ $T > 0$ und mit $W \triangleq \alpha V_m(x)$ definiert. Außerdem ist hier die Ungleichung (3.22) für $C_1 = 0$ erfüllt. \diamond

Beweis: Siehe hierzu [24], Theorem 5.2. \blacklozenge

Man hat mit diesem Satz nun gezeigt, dass das System asymptotisch stabilisiert werden kann, sofern das nichtlineare Optimierungsproblem 3.51 einen Zielfunktionswert $\alpha \in (0, 1]$ besitzt.

Allerdings schließt diese Formulierung nicht aus, dass Stabilität bereits für kleinere Optimierungshorizonte N gilt.

⁷Für den auf den Abtastzeitpunkt t_i folgenden Zeitpunkt $t_{i+1} = t_i + T$ schreiben wir $n_{i+1} = n_i + 1$ mit $t_i = n_i$. Man kann sozusagen für die Definition von V_N gemäß Definition 3.28 in dieser den Zeitparameter T als 1 "interpretieren", so dass er in der Definition als Faktor vor W nicht in Erscheinung tritt und $TW \triangleq 1 \cdot W \triangleq W \triangleq \alpha V_m(x)$ identifiziert werden kann.

3.8.4 Wahl einer geeigneten Kostenfunktion l

Wir sind nun in der Lage unter Verwendung einer Vergleichsfunktion β aus Annahme 3.48 in Abschnitt 3.8.1 und der Resultate aus Abschnitt 3.8.2 durch Lösen des Problems 3.51 aus Abschnitt 3.8.3 für einen bevorzugten Wert $\alpha_0 \geq 0$ einen *minimalen Optimierungshorizont*

$$\hat{N} := \min\{N \in \mathbb{N} \mid \alpha > \alpha_0\} \quad (3.57)$$

zu berechnen, der asymptotische Stabilität liefert und im Falle von $\alpha_0 > 0$ eine Abschätzung der Güte

$$\alpha_0 V_\infty^{\mu_{\hat{N}}, m}(x) \leq V_{\hat{N}}(x) \quad (3.58)$$

zusichert.

Zusammenhang von α und l

Betrachtet man die Definition des optimalen Horizonts \hat{N} , so erhält man diesen, indem man das nichtlineare Optimierungsproblem 3.51 zu gegebenen β und α_0 für verschiedene Horizonte N auswertet und dann diese herausnimmt, die als Zielfunktionswert ein α liefern, für das $\alpha > \alpha_0$ gilt. Anschließend wird über dieser Menge minimiert und der resultierende Wert als der *minimale Optimierungshorizont* \hat{N} definiert.

Dabei beobachtet man, dass das α aus dem Problem 3.51 monoton fallend in β ist. Das lässt sich dadurch erklären, dass kleinere β engere Schranken in Problem 3.51 liefern und folglich dieses wiederum bedeutet, dass unser minimaler Horizont \hat{N} aus (3.57) in β monoton steigt.

Diese Tatsache hat eine sehr große Bedeutung in Bezug auf obere Schranken des minimalen Horizonts \hat{N} , da es in der Praxis kaum bis gar nicht möglich ist eine entsprechend der Annahme 3.48 geeignete enge Schranke β zu berechnen. Viel eher steht zu erwarten, dass man hierfür keine minimalen oberen Schranken erhält. Damit besagt diese Monotonie-Eigenschaft, dass jedes für solch ein β berechnetes \hat{N} eine obere Schranke für den tatsächlichen optimalen Horizont darstellt.

Das führt dazu, sich mit der Beschaffenheit von β aus Annahme 3.48 auseinanderzusetzen, was vor allem die charakteristischen Eigenschaften von β , wie z.B. *decay rate* und *overshoot*, betrifft.

Da sich nun die Annahme 3.48 auf die laufenden Kosten l entsprechend

$$l(x_{u_x}(n), u_x(n)) \leq \beta(l^*(x), n) \quad \forall n \in \mathbb{N}_0$$

bezieht, liefert uns eine diesbezügliche Untersuchung von β sozusagen “Richtlinien” für den Entwurf von l , die es uns ermöglichen, stabile MPC-Schemata mit kleinem Optimierungshorizont zu erhalten.

Eine solche Untersuchung des Zusammenhangs von β und des geschätzten minimalen Stabilisierungshorizont N wurde anhand numerischer Auswertungen von L. GRÜNE in [24], Abschnitt 6, durchgeführt.

Dazu wurde untersucht, wie N von einem durch β dargestellten Wert $\gamma > 0$ abhängt, für

den gilt

$$\sum_{n=0}^{\infty} \beta(r, n) \leq \gamma r. \quad (3.59)$$

Diesen Wert γ bezeichnet man als “angehäuften Überschuss” (*accumulated overshoot*).

Ergebnisse der numerischen Auswertung

Für die numerischen Untersuchungen wurde \hat{N} aus (3.57) für β aus (3.59), einen Kontrollhorizont m und zu verschiedenen Werten $\gamma = 1, 2, \dots, 50$ berechnet. Daraus erhielt man die folgenden interessanten Resultate:

- Für Kontrollhorizont $m = 1$ erhält man $\hat{N} \approx \gamma \log \gamma$.
- Für Kontrollhorizont $m = \lceil N/2 \rceil + 1$ erhält man $\hat{N} \approx 1.4\gamma$.
- Je kleiner γ gewählt wird, umso kleiner wird \hat{N} .

Aus den ersten beiden Beobachtungen lässt sich folgern, dass man für größeren Kontrollhorizont $m > 1$ Stabilität für viel kleinere N erhält, dies allerdings auf Kosten der Reaktionsmöglichkeiten auf Störungen (Zur Erinnerung: Die Berechnung der Kontrollfolge basierend auf den aktuellen Zustandsgrößen findet nur zu den Zeiten $0, m, 2m, \dots$ statt). Lässt sich jedoch andererseits ein großer Überschuss γ nicht vermeiden und liegen Einschränkungen der rechen-technischen Ressourcen vor, so besteht zumindest die Möglichkeit kleinere Horizonte N durch ein gezieltes Erhöhen von m zu erhalten und dabei trotzdem die Komplexität des Optimierungsproblems im Rahmen zu halten.

Für den dritten Punkt bieten sich verschiedene Wege an, das γ zu reduzieren:

So ist im Falle einer exponentiell fallenden Funktion $\beta(r, n) = C\sigma^n r$ interessant zu wissen, ob ein kleiner Überschuss (d.h. kleines C) oder ein schnelles Sinken (d.h. kleines σ) wichtiger und damit besser geeignet ist, um Stabilität für kleines \hat{N} zu garantieren. Gleiches gilt auch für den Fall der Kontrollierbarkeit in endlicher Zeit mit einer nicht streng monoton fallenden Funktion $\beta(r, n) = c_n r$.

Die numerischen Auswertungen haben in beiden Fällen ergeben, dass ein kleiner Überschuss deutlich wichtiger als ein schnelles Sinken ist. Daneben konnte aber noch eine weitere ausschlaggebende Eigenschaft beobachtet werden, die Auswirkung auf die Länge des stabilisierenden Horizonts N hat: Dabei handelt es sich um die minimale Zeit n_c , für die *Kontraktion* (d.h. $\beta(r, n) < r$ für alle $n \geq n_c$) eintritt.

Richtlinien für die Wahl von l

Fasst man die oben gemachten Ergebnisse und Resultate zusammen, lassen sich damit Richtlinien für die Wahl von l formulieren, um ein stabiles MPC-Schema mit kleinem Optimierungshorizont N zu erhalten:

- Wähle l derart, dass der Überschuss $\gamma = \sum_{n=0}^{\infty} \frac{\beta(r,n)}{r}$ so klein wie möglich ist.
- Im Falle der exponentiellen Kontrollierbarkeit $\beta(r, n) = C\sigma^n r$ ist eine Verminderung des Überschusses durch Reduzieren von C wirkungsvoller als durch Reduzieren von σ .
- Im Falle der Kontrollierbarkeit in endlicher Zeit $\beta(r, n) = c_n r$ ist eine Verminderung des Überschusses durch Reduzieren von c_n wirkungsvoller als durch Reduzieren der Zeit, um $l^*(x) = 0$ zu erreichen

Bemerkung 3.54

Es sei angemerkt, dass auch die Wahl von *Endgewichten* ω in der Zielfunktion und die Wahl des Kontrollhorizonts m einen erheblichen Einfluss darauf nehmen, ein stabiles MPC-Schema mit kleinem Optimierungshorizont N zu erhalten. Für Richtlinien zu einer geeigneten Wahl dieser Größen verweisen wir auf [24], Abschnitt 7. \circ

In [24], Abschnitt 7, wird die Wirksamkeit dieser Richtlinien anhand zweier Beispiele für ein endlich dimensionales und ein unendlich dimensionales System demonstriert. Gerade im unendlich dimensionalen Fall erweist es sich als sehr schwierig die Funktion β sowie die Kontrollfunktion u_x aus Annahme 3.48 explizit zu konstruieren. Dennoch ist eine Anwendung der erhaltenen Ergebnisse möglich, indem die benötigten Informationen über β aus der Struktur der Systemgleichung entnommen werden.

Zur Veranschaulichung dient hier als Beispiel die parabolische *Reaktions-Advektions-Diffusions-Gleichung* mit additiv hinzugefügter *verteilter Steuerung* u

$$y_t = -y_x + \nu y_{xx} + \mu y(y+1)(1-y) + u$$

auf einem Gebiet $\Omega = (0, 1)$ mit der zugehörigen Lösung $y = y(x, t)$. Die Intention dabei ist, das System unter festgelegten Parameterbelegungen sowie Rand- und Startbedingungen in der Null zu stabilisieren.

Um dieses Ziel zu erreichen, ergibt sich als sinnvolle Wahl der Kostenfunktion l das sogenannte *tracking type* Funktional

$$l(y(x, t), u(x, t)) = \|y(x, t)\|_{L^2(\Omega)}^2 + \lambda \|u(x, t)\|_{L^2(\Omega)}^2. \quad (3.60)$$

Auf der Suche nach einem minimalen stabilisierenden Horizont erweist sich diese Form von l im Zielfunktional in manchen Fällen als nicht zufriedenstellend, weshalb auch folgende Form Verwendung findet, bei der die Ableitung $\|y_x\|_{L^2(\Omega)}^2$ additiv hinzugefügt wurde:

$$l(y(x, t), u(x, t)) = \|y(x, t)\|_{L^2(\Omega)}^2 + \|y_x(x, t)\|_{L^2(\Omega)}^2 + \lambda \|u(x, t)\|_{L^2(\Omega)}^2. \quad (3.61)$$

Die numerische Auswertung der Reaktions-Advektions-Diffusions- sowie der Wellengleichung in Kapitel 7 wird unter Verwendung dieser beiden unsere Zielfunktionale definierenden Kostenfunktionen l vorgenommen, wobei wir diese bezüglich der in (3.60) und (3.61) verwendeten Normen L^2 und H^1 unterscheiden werden (siehe hierzu Kapitel 5).

4 Diskretisierungen

Sowohl gewöhnliche als auch partielle Differentialgleichungen lassen sich meist nicht analytisch lösen. Deswegen finden numerische Lösungsmethoden in vielen natur- und ingenieurwissenschaftlichen Bereichen ihre Anwendung.

In der Mikroelektronik zum Beispiel ist die numerische Simulation für die Entwicklung hochintegrierter Bauelemente unverzichtbar geworden (vgl. hierzu G. BÄRWOLFF [5]), aber auch verschiedenartige Tests im Windkanal oder im Weltraum werden anhand von numerischen Experimenten vorbereitet.

Dabei treten folgende drei elementare Näherungsverfahren für PDEs auf:

- Finite-Volumen-Methode
- Finite-Elemente-Methode
- Finite-Differenzen-Methode

In unserer Arbeit wird die Finite-Differenzen-Methode ihre Anwendung finden, um die später auszuwertenden PDEs mittels Differenzenapproximationen der ersten und zweiten Ableitung, sowohl nach der Zeit als auch nach dem Ort, in ein System gewöhnlicher Differentialgleichungen zu überführen. Durch diese Approximationen erhalten wir aus dem kontinuierlichen Problem ein endliches diskretes Problem, das dann mit einem digitalen Rechner gelöst werden kann.

Um die Finite-Differenzen-Methode anwenden zu können, benötigen wir nur an speziellen Punkten Näherungswerte für unsere jeweiligen Differenzenquotienten, die die Ableitungen ersetzen. Daher müssen wir ein so genanntes *Rechengitter* oder kurz Gitter über dem gegebenen Gebiet Ω einführen. Diese Zerlegung in kleine „Abschnitte“ macht eine digitale Verwendung möglich. Eine Variante der Finite-Differenzen-Methode stellen die Linienmethoden dar, die sehr flexibel und vielseitig anwendbar sind. Auch hier werden wir die verschiedenen orts- und zeitabhängigen Diskretisierungen genauer betrachten.

4.1 Definitionen

4.1.1 Gitterdiskretisierung

Zur numerischen Lösung von PDEs approximieren wir ihre partiellen Ableitungen. Dazu verwenden wir eine Diskretisierung, bei der Raum und/oder Zeit durch ein Rechengitter in endlich viele Abschnitte zerlegt wird/werden.

4 Diskretisierungen

Zur Motivation betrachten wir den eindimensionalen Fall nur in Richtung der örtlichen Variable, d.h. in x -Richtung, mit gesuchter Funktion

$$y = y(x) : \mathbb{R} \rightarrow \mathbb{R}. \quad (4.1)$$

Eine Diskretisierung für ein gegebenes Intervall $[a, b]$ ist:

$$x_i = a + i \cdot \Delta x = a + i \cdot h, \quad h = \frac{b - a}{M}, \quad i = 0, \dots, M, \quad M \in \mathbb{N}. \quad (4.2)$$

Dabei nennt man h die Diskretisierung und M den Diskretisierungsparameter, der die Anzahl der Intervalle in x -Richtung angibt. Wir betrachten die Gleichung also nur noch für $M + 1$ räumliche Diskretisierungspunkte. Auch die Lösung wird daher nur an isolierten Gitterpunkten berechnet, wobei die Werte, die dazwischen liegen, approximativ bestimmt werden können. Die Diskretisierung muss nicht wie in der Motivation auf einem äquidistanten¹ Gitter, sondern kann auch auf einem adaptiven² Gitter gewählt werden. Wenn große Fehler erwartet werden, kann somit das Gitter feiner gewählt werden. Meist wird jedoch die äquidistante Verteilung der Gitterpunkte bevorzugt, da die numerischen Verfahren und Algorithmen so eine sehr einfache Gestalt annehmen.

Wenn nun die Lösung nicht nur in x -Richtung, sondern auf einem Rechtecksgebiet gesucht wird, müssen wir das Konzept entsprechend erweitern (siehe [39]).

Zur Motivation betrachten wir wieder den einfachsten Fall: Als Rechtecksgebiet betrachten wir

$$R = [a, b] \times [c, d], \quad (4.3)$$

das in x - und t -Richtung äquidistant mit konstanten Schrittweiten Δx und Δt zerlegt wird. Diese lassen sich wie folgt berechnen:

$$\Delta x = \frac{b - a}{I} \quad \Delta t = \frac{b - a}{J} \quad (4.4)$$

mit Anzahl der Gitterintervalle I und J jeweils in x - und t -Richtung.

Die Menge der Gitterpunkte ist somit durch die inneren Gitterpunkte (x_i, t_j) gegeben mit:

$$x_i = a + i \cdot \Delta x \quad , \quad i = 1, \dots, I - 1 \quad (4.5)$$

$$t_j = c + j \cdot \Delta t \quad , \quad j = 1, \dots, J - 1 \quad (4.6)$$

Die Gitterpunkte am Rand sind:

$$(a, t_j), (b, t_j), (x_i, c), (x_i, d) \quad , \quad i = 0, \dots, I, j = 0, \dots, J. \quad (4.7)$$

Folgende Abbildung 4.1 zeigt diese Aufteilung mit $I = 6$ und $J = 4$.

¹Teilung in gleich große Intervalle

²Teilung in unterschiedlich große Intervalle

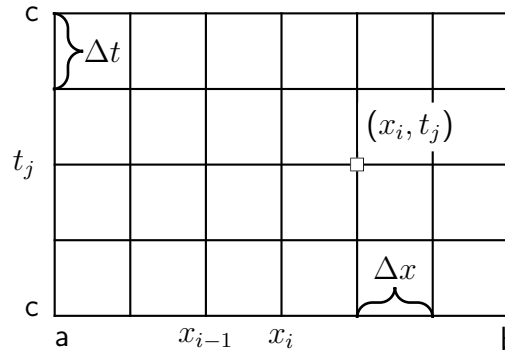


Abbildung 4.1: Diskretisierung eines Rechtecks

Wir wollen jetzt die Definition für eine Diskretisierung eines zweidimensionalen Gebiets angeben.

Definition 4.1 (Diskretisierung eines Gebiets)

Sei $\Omega \subset \mathbb{R}^2$ offen und beschränkt und $h > 0$ die Schrittweite. Außerdem sei Γ der Rand von Ω . Eine mögliche Diskretisierung von Ω ist im Inneren durch

$$\Omega_h := \{x \in \Omega : \exists k, l \in \mathbb{Z}, x_1 = kh, x_2 = lh\} \tag{4.8}$$

und am Rand durch

$$\Gamma_h := \bigcup_{k \in \mathbb{Z}} \partial\{x \in \Omega : x_1 = kh\} \cup \bigcup_{l \in \mathbb{Z}} \partial\{x \in \Omega : x_2 = lh\} \tag{4.9}$$

gegeben, wobei die Ränder relativ bezüglich der Geraden $\{x \in \mathbb{R}^2 : x_1 = kh\}$ bzw. $\{x \in \mathbb{R}^2 : x_2 = lh\}$ sind. □

Bemerkung 4.2

Diese Definition lässt sich auf höhere Dimensionen verallgemeinern. Wir benötigen aber in unseren Auswertungen nur den zweidimensionalen Fall, weswegen uns diese Definition genügt. ○

Auch hier wollen wir die Diskretisierung anhand der Abbildung 4.2 betrachten.

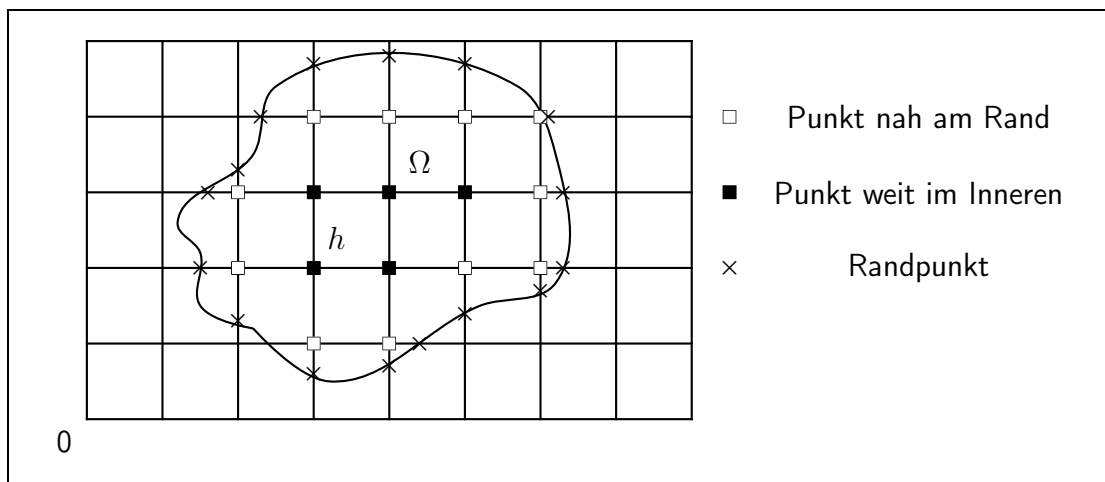


Abbildung 4.2: Diskretisierung eines 2-dimensionalen Gebiets

Bemerkung 4.3

In der Abbildung sind die aus der Definition 4.1 bekannten Geraden, sowie deren Punkte im Inneren und auf dem Rand zu sehen. Es ergibt sich folgende Inklusion:

$$\Omega_h \subset \Omega \quad , \quad \Gamma_h \subset \Gamma \quad (4.10)$$

○

Bemerkung 4.4

- Wir betrachten in unserem Zusammenhang nur **achsenparallele Gitterlinien**, die äquidistant sind, da diese eine sehr einfache Form eines Gitters darstellen und somit auch einfache Bezeichnungen und Nummerierungen besitzen. Die Differenzenverfahren, die wir in den nächsten Abschnitten kennen lernen werden, sind alle auf diese Gitter anwendbar. Der Grund liegt darin, dass durch die in Abbildung 4.1 zu sehenden Punkte (x_i, t_j) auch ihre anschließenden Nachbarpunkte (z.B. (x_{i+1}, t_j) oder (x_{i-1}, t_j)) bekannt sind, wenn der Punkt im Inneren liegt. Genau diese Nachbarn sind für die Berechnung zwingend notwendig. Ein Nachteil besteht darin, dass trotz feiner Gitterauflösung stufenweise approximiert wird, was eine Fehlerquelle darstellt.
- Eine weitere Möglichkeit der Gitterdiskretisierung ist das **randangepasste strukturierte Gitter**, das heißt die Gitterlinien passen sich dem Rand an. Diese Gitterstruktur wird bei Gebieten, die keine Rechteckstruktur, sondern einen krummlinigen Rand oder eine „einspringende Kante“ (vgl. [39]) besitzen, angewendet. Dabei wird das Gitter auf ein kartesisches Gitter durch eindeutige Transformation abgebildet, wodurch sich auch hier eine logische Struktur ergibt und jeder Punkt wieder einen anschließenden Nachbarpunkt besitzt. Deswegen können auch hier die Differenzenverfahren angewendet werden. Der Vorteil besteht darin, dass der Diskretisierungsfehler um einiges geringer als im Vergleich zu achsenparallelen Gitterlinien ist. Trotzdem wird diese Diskretisierung seltener verwendet, da sie vor allem bei komplizierten Geometrien viel Erfahrung und Zeit benötigt. Für nähere Informationen über die Erzeugung von randangepassten Gittern siehe C.-D. MUNZ und T. WESTERMANN [39].
- Mit einem **unstrukturierten Gitter** können noch sehr komplizierte Gebiete diskretisiert werden. Dies stellt zwar eine sehr variable Diskretisierung dar, jedoch gibt es hier keine Transformation auf ein kartesisches Gitter wie bei den randangepassten Gittern. Deswegen finden hier keine Differenzenverfahren Anwendung. Es werden Verfahren wie die Finite-Elemente- oder Finite-Volumen-Methode benötigt, auf die wir aber in dieser Arbeit nicht weiter eingehen werden. ○

Bemerkung 4.5

Da wir unser kontinuierliches System durch das Prinzip der Abtastung (siehe Abschnitt 3.4.1) nur an diskreten Zeitpunkten auswerten, erhalten wir letztendlich eine zeitlich diskrete Variante des betrachteten Systems. Daher ist für uns die Diskretisierung mittels eines achsenparallelen äquidistanten Gitters vollkommen ausreichend. Außerdem ist diese Möglichkeit der Gitterdiskretisierung am einfachsten und schnellsten zu realisieren. ○

4.1.2 Semidiskretisierung

Von einer Semidiskretisierung sprechen wir, wenn wir nicht alle unabhängigen Variablen, sondern nur in eine bestimmte Richtung diskretisieren, das heißt in unseren Auswertungen in Richtung der örtlichen Variable x , oder in Richtung der Zeit t .

Allgemein bedeutet das, dass wir nur nach einer ausgewählten Variable diskretisieren, um zum Beispiel mittels einer Linienmethode (siehe Abschnitt 4.3) zu approximieren. Dazu müssen alle vorkommenden partiellen Ableitungen der zu diskretisierenden Variable approximiert werden. Dies kann durch einen Differenzenquotienten geschehen (siehe Abschnitt 4.2.2). In den meisten Anwendungen, wie zum Beispiel der Wärmeleitungsgleichung, bleibt die Zeit t kontinuierlich, während nur die örtliche Variable diskretisiert wird.

4.1.2.1 Zeitliche Diskretisierung

Wir wollen zuerst die Diskretisierung nach der t -Variable anhand eines Beispiels graphisch darstellen, das jedoch nicht weiter analysiert wird. Dazu nehmen wir uns die sogenannte *Richardsgleichung*³ her, die u.a. die Wasserbewegung im Erdboden beschreibt. Wir betrachten hierbei die Bodenfeuchte Θ mit

$$\frac{\partial \Theta}{\partial t} = -\frac{\partial}{\partial z} \left(k(\Theta) \frac{\partial \psi}{\partial z} \right) + S, \quad (4.11)$$

wobei z die Tiefe, $k(\Theta)$ und $\psi(\Theta)$ bodenhydraulische Funktionen und S der Senkenterm zur Beschreibung der Wurzelwasseraufnahme bodenhydraulischer Funktionen⁴ sind.

Für weitere ausführliche Informationen siehe [29].

Ein beispielhafter Verlauf der Bodenfeuchte zeigt folgende Abbildung 4.3:

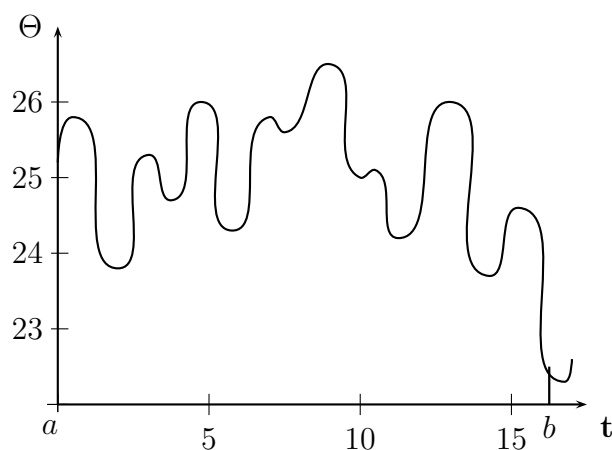


Abbildung 4.3: Ausgangssystem

Diskretisieren wir jetzt das Intervall $[a, b]$ nach der Zeit t mit $\Delta t = 1$, erhalten wir nur noch die diskreten Punkte, die in der Abbildung 4.4 mit rot markiert sind:

³nach *Lorenzo A. Richards*, 1904-1993

⁴Vergleiche hierzu <http://www.uni-potsdam.de/u/Geoökologie/institut/wasserhaushalt/download/practmodneu4.pdf>

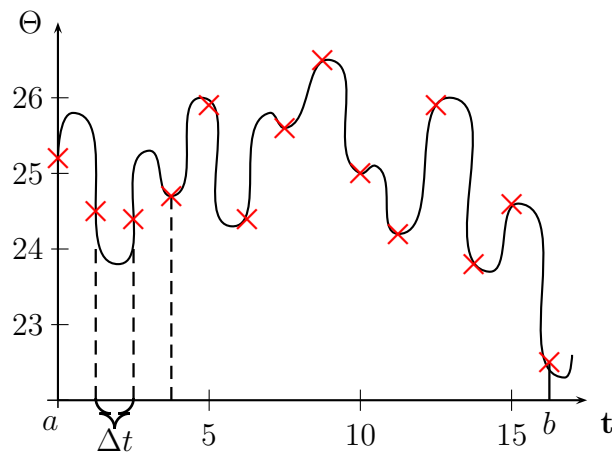


Abbildung 4.4: Zeitliche Diskretisierung

Aufgrund der vorgenommenen Diskretisierung können nun die Ableitungen durch die Finite-Differenzen-Methode approximiert werden. Diese Methode ist in Abschnitt 4.2.2 zu finden.

4.1.2.2 Räumliche Diskretisierung

Analog verläuft die Diskretisierung im Raum, nur dass jetzt die Bodenfeuchte Θ an diskreten Raumpunkten betrachtet wird. Die erste Abbildung 4.5 zeigt dabei anhand von Verfärbungen die verschiedenen Bodenfeuchtigkeiten eines betrachteten Erdbodens.

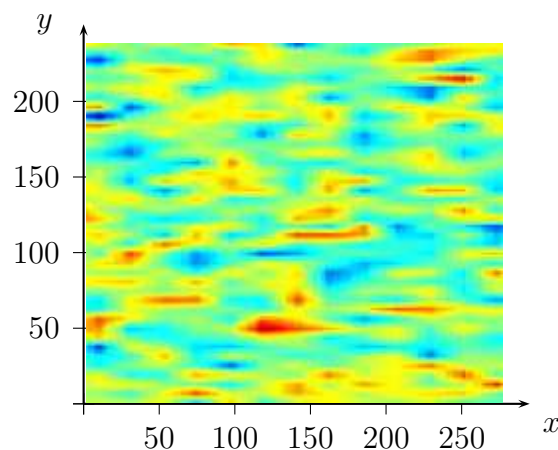


Abbildung 4.5: Ausgangssystem

Für die Diskretisierung nach einer Raumvariable wird die Bodenfeuchte nur an diskreten Punkten betrachtet, wie wir in nachfolgender Abbildung 4.6 erkennen können.

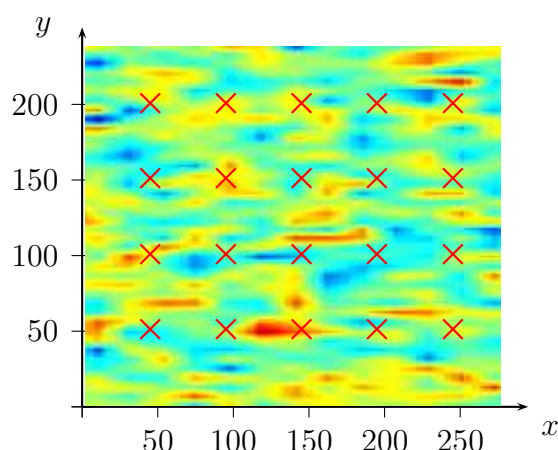


Abbildung 4.6: Räumliche Diskretisierung

Bemerkung 4.6

Wir werden in unseren Auswertungen der PDEs die räumliche Diskretisierung verwenden. Das aus dem Problem mit partiellen Ableitungen entstehende Anfangswertproblem gewöhnlicher Differentialgleichungen kann hierauf mit Hilfe numerischer Verfahren gelöst werden. Dazu wird das räumlich diskretisierte System nur an diskreten Zeitpunkten betrachtet. \circ

4.2 Numerische Differentiation

Die Differentiation ist ein konstruktives Verfahren, das bei explizit gegebenen Funktionen exakt per Hand oder mittels eines Computer-Algebra-Systems bestimmt werden kann. Da viele Funktionen jedoch nur durch Werte aus einer Wertetabelle gegeben sind, benötigen wir meist die numerische Differentiation. Aufgabe dieses Verfahrens ist es, die k -te Ableitung einer mindestens k -mal stetig differenzierbaren Funktion $y = y(x)$ ⁵ geeignet zu approximieren. Dies geschieht über so genannte Differenzenverfahren, bei denen die kontinuierlichen Ableitungen einer gegebenen PDE durch Näherungsausdrücke ersetzt werden. Dieses Verfahren werden wir im Folgenden herleiten und näher betrachten.

4.2.1 Taylor-Entwicklung

Wir wollen nun gute Approximationen für unsere auftretenden partiellen Ableitungen herleiten und verfolgen in diesem Kapitel den Ansatz über die Taylorreihe (vgl. [8, 39]). Voraussetzung dafür sind hinreichende Differenzierbarkeitseigenschaften an die Funktion $y = y(x)$.

⁵Wir betrachten hier vereinfacht nur den eindimensionalen Fall mit der Raumvariablen x . Analog können die Ergebnisse sowohl alleine in Abhängigkeit von der Zeit t als auch mit Hinzunahme der Zeit t ergänzt werden.

Dazu setzen wir

$$\Delta x := x_{i+1} - x_i, \quad , \quad i = 0, \dots, M - 1$$

und entwickeln $y(x)$ an den Stellen x_{i+1} :

$$y(x_{i+1}) = y(x_i) + \frac{\partial y}{\partial x}(x_i)\Delta x + \frac{1}{2}\frac{\partial^2 y}{\partial x^2}(x_i)(\Delta x)^2 + O(\|\Delta x\|^3). \quad (4.12)$$

Analog setzen wir

$$-\Delta x := x_{i-1} - x_i, \quad , \quad i = 1, \dots, M$$

und entwickeln $y(x)$ an den Stellen x_{i-1} :

$$y(x_{i-1}) = y(x_i) + \frac{\partial y}{\partial x}(x_i)(-\Delta x) + \frac{1}{2}\frac{\partial^2 y}{\partial x^2}(x_i)(-\Delta x)^2 + O(\|\Delta x\|^3). \quad (4.13)$$

Da wir uns für die erste Ableitung interessieren, subtrahieren wir die beiden Taylorreihen (4.13) und (4.12) voneinander und erhalten folgende Näherung für die erste Ableitung nach x :

$$y(x_{i+1}) - y(x_{i-1}) = \frac{\partial y}{\partial x}(x_i)(2 \cdot \Delta x) + O(\|\Delta x\|^3) \quad (4.14)$$

$$\Rightarrow \frac{\partial y}{\partial x}(x_i) = \frac{y(x_{i+1}) - y(x_{i-1}))}{2 \cdot \Delta x} + O(\|\Delta x\|^2). \quad (4.15)$$

Der Exponent in der O -Notation gibt die Ordnung der Differenzenformel an (Konsistenzfehler). Diese Approximation ist somit zweiter Ordnung und wird ohne den O -Term *zentraler Differenzenquotient zur Näherung der ersten Ableitung einer Funktion $y(x)$* genannt.

Bemerkung 4.7

Da die hergeleitete Approximation (4.15) nur für $i = 1, \dots, M-1$ definiert ist, wir aber auch die Approximationen am Rand benötigen, betrachten wir die Taylorreihen bei x_1 und x_2 bzw. analog bei x_{M-1} und x_{M-2} , um eine Approximation mit hinreichender Genauigkeit für $\frac{\partial y}{\partial x}(x_0)$ bzw. analog für $\frac{\partial y}{\partial x}(x_M)$ zu erhalten. \circ

Für den Fall $i = 0$ verknüpfen wir die beiden Taylorreihen

$$y(x_1) = y(x_0) + \frac{\partial y}{\partial x}(x_0)\Delta x + \frac{1}{2}\frac{\partial^2 y}{\partial x^2}(x_0)(\Delta x)^2 + O(\|\Delta x\|^3) \quad (4.16)$$

und

$$y(x_2) = y(x_0) + \frac{\partial y}{\partial x}(x_0)(2 \cdot \Delta x) + \frac{1}{2}\frac{\partial^2 y}{\partial x^2}(x_0)(2 \cdot \Delta x)^2 + O(\|\Delta x\|^3) \quad (4.17)$$

derart, dass die Terme zweiter Ordnung herausfallen und erhalten eine Approximation zweiter Ordnung:

$$\frac{\partial y}{\partial x}(x_0) = \frac{-3 \cdot y(x_0) + 4 \cdot y(x_1) - y(x_2)}{2 \cdot \Delta x} + O(\|\Delta x\|^2). \quad (4.18)$$

Analog erhalten wir für $i = M$ folgende Approximation zweiter Ordnung:

$$\frac{\partial y}{\partial x}(x_M) = \frac{3 \cdot y(x_M) - 4 \cdot y(x_{M-1}) + y(x_{M-2})}{2 \cdot \Delta x} + O(\|\Delta x\|^2). \quad (4.19)$$

Bemerkung 4.8

Die hergeleiteten Differenzenformeln gelten nur unter Vernachlässigung der Rundungsfehler. Man kann beobachten, dass auch bei Verkleinerung der Schrittweite Δx der Wert des Fehlers wieder steigen kann. Für weitere Informationen über die verschiedenen Fehler sei auf C. MUNZ [39] und G. BÄRWOLFF [5] verwiesen. \circ

Bemerkung 4.9

Analog zu der Bestimmung der Approximationen für die erste Ableitung kann die Herleitung der Differenzenformeln für Ableitungen höherer Ordnung mit Hilfe der entsprechenden Taylorreihen durchgeführt werden. \circ

4.2.2 Finite-Differenzen-Methode

Unser Ziel ist es, möglichst geschickt Lösungen von PDEs zu approximieren. Dies geschieht zum Beispiel, indem eine diskrete Approximation einer Funktion gefunden wird, die einer bestimmten Beziehung verschiedener Ableitungen auf einem gegebenen Gebiet Ω genügt. Zusätzlich müssen oft noch einige Rand- und Anfangsbedingungen erfüllt werden. In den meisten Fällen haben wir es mit sehr komplexen Problemen zu tun und sind nicht in der Lage, diese analytisch zu lösen. Selbst bei linearen Problemen ist dies oft der Fall.

Die Finite-Differenzen-Methode ist ein sehr schnelles und weit verbreitetes Verfahren, das auf lineare und nichtlineare Differentialgleichungen angewendet werden kann. Die Methode arbeitet mit Ersetzen der Ableitungen in den Differenzengleichungen durch finite Differenzenapproximationen. Dadurch erhalten wir ein sehr großes, aber finites algebraisches System von Gleichungen, das an Stelle der Differentialgleichung gelöst wird.

Die Finite-Differenzen-Methode (FDM) ersetzt formal das kontinuierliche Gebiet durch eine diskrete Menge von Gitterpunkten auf den Koordinatenlinien und führt Differenzenquotienten bezüglich dieser Gitterpunkte ein (siehe C. GROSSMANN und H.-G. ROSS [23], M. HANKE-BOURGEOIS [28] und R. J. LEVEQUE [35]). Nach H. P. LANGTANGEN [34] lässt sich unter Berücksichtigung der vorherigen Abschnitte die FDM in 6 Schritte unterteilen:

1. Das gegebene Gebiet (z.B. $\Omega = (0, 1)$) wird in eine bestimmte Anzahl an Bereichen (z.B. $[x_i, x_{i+1}]$, $i = 1, \dots, n - 1$) aufgeteilt. Dadurch erhalten wir eine diskrete Darstellung des Gebiets in Form eines Gitters mit Gitterpunkten.
2. Eine numerische Approximation der exakten Lösung $u(x_i)$ wird gesucht.
3. Die Ableitungen der Differentialgleichung werden durch finite Differenzenapproximationen ersetzt.
4. Die Differentialgleichung muss an den Gitterpunkten erfüllt sein.

5. Anfangsbedingungen müssen berücksichtigt werden, indem die Differenzengleichung an den entsprechenden Stellen durch die entsprechende numerische Approximation ersetzt wird.
6. Einfache Randbedingungen werden analog berücksichtigt. Bedingungen an Ableitungen müssen zusätzlich noch durch geeignete finite Differenzenapproximationen kombiniert mit der entsprechenden Differenzenapproximation der Differentialgleichung ersetzt werden.

Wenn wir diese Schritte durchführen, erhalten wir ein diskretes Problem, das in eine Matrix-Form gebracht werden kann.

Die Basisfrage, die sich jetzt stellt, ist folgende:

Wie können wir die Ableitungen einer gegebenen Funktion an bestimmten Gitterpunkten durch finite Differenzenquotienten approximieren, wenn wir nur Werte der Funktion selbst an diskreten Punkten zur Verfügung haben (vgl. Punkt 3 oben)?

An diese Fragestellung wollen wir im klassischen Sinne herangehen und erst einmal definieren, wie eine Ableitung einer Funktion an einem bestimmten Punkt mit einem digitalen Rechner numerisch berechnet werden kann (z.B. [18, 39]).

Definition 4.10 (Differentialquotient)

Sei $D \subset \mathbb{R}$ und $y : D \rightarrow \mathbb{R}$ eine Funktion. $y = y(x)$ heißt in einem Punkt $x \in D$ differenzierbar, falls der Grenzwert

$$\frac{\partial y}{\partial x} := \lim_{\substack{\zeta \rightarrow x \\ \zeta \in D \setminus \{x\}}} \frac{y(\zeta) - y(x)}{\zeta - x} \quad (4.20)$$

existiert. Insbesondere wird vorausgesetzt, dass es mindestens eine Folge $\zeta_n \in D \setminus \{x\}$ mit $\lim_{n \rightarrow \infty} \zeta_n = x$ gibt. Dies ist z.B. immer der Fall, wenn D ein Intervall ist, das aus mehr als einem Punkt besteht.

Der Grenzwert y_x wird *Differentialquotient* von y im Punkt x genannt. □

Bemerkung 4.11

Man kann den Differentialquotienten auch folgendermaßen schreiben:

$$\frac{\partial y}{\partial \bar{x}} := \lim_{\Delta \bar{x} \rightarrow 0} \frac{y(\bar{x} + \Delta x) - y(\bar{x})}{\Delta \bar{x}} \quad (4.21)$$

oder angepasst an die Schreibweise der Taylorentwicklung und $\Delta x = h$

$$\frac{\partial y}{\partial x}(x_i) \approx \frac{y(x_{i+1}) - y(x_i)}{h}. \quad (4.22)$$

○

Bemerkung 4.12

Für kleine $\Delta x \neq 0$ stellt der rechte Ausdruck der Gleichung eine Näherung für die erste Ableitung dar.

Geometrisch stellt die Gleichung (4.21) die Steigung der Tangente im Punkt $y(x)$ bzw. der Sekante durch die Funktionswerte bei x und $x + \Delta x$ dar (vgl. Abbildung 4.7). \circ

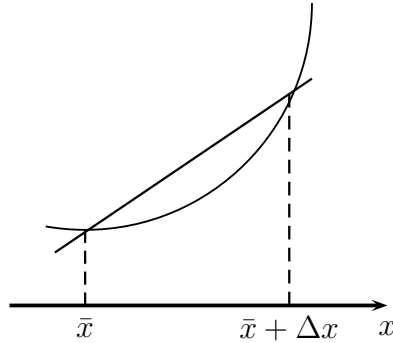


Abbildung 4.7: Sekantensteigung

Mit Hilfe dieser Definitionen und der Taylor-Entwicklung können wir jetzt die allgemeinen Differenzenquotienten erster Ordnung aufstellen. Wir schreiben y wieder in Abhängigkeit von x und t .

4.2.2.1 Differenzenapproximationen erster Ordnung

Die erste offensichtliche Wahl ist der *Vorwärts-Differenzenquotient*, der durch die Definition 4.10 sowie dem begrenzten Wert dieses Ausdrucks ($h \rightarrow 0$) motiviert wird. Für kleine Schrittweiten h erhalten wir also:

$$D_+y(\bar{x}, t) = y_x(x_i, t) \approx \frac{y(x_{i+1}, t) - y(x_i, t)}{h} \quad (4.23)$$

Bemerkung 4.13

$D_+y(\bar{x}, t)$ kann als Anstieg der Linieninterpolation y an den Punkten x_{i+1} und x_i angesehen werden.

Der obige Ausdruck ist eine einseitige Approximation, da y nur bei $x \geq x_{i+1}$ gilt. Aufgrund der Taylor-Entwicklung (mit vereinfachter Schreibweise ohne den Parameter t und $\Delta x = h$)

$$y(x_{i+1}) = y(x_i) + \frac{\partial y}{\partial x}(x_i)\Delta x + O(\|\Delta x\|^2) \quad (4.24)$$

erhalten wir hier eine Approximation mit einer Genauigkeit erster Ordnung. Dies bedeutet, dass die Größe des Fehlers ungefähr proportional zur Schrittweite ist. \circ

Analog wird eine weitere wichtige einseitige Approximation motiviert. Dieser so genannte *Rückwärts-Differenzenquotient* stellt den Anstieg der Linieninterpolation y diesmal an den Punkten x_i und x_{i-1} für kleine Schrittweiten h dar:

$$D_-y(\bar{x}, t) = y_x(x_i, t) \approx \frac{y(x_i, t) - y(x_{i-1}, t)}{h} \quad (4.25)$$

Bemerkung 4.14

Diese Approximation (4.25) von $y_x(x_i, t)$ besitzt wie die Formulierung (4.23) eine Genauigkeit erster Ordnung. Dies kann man auch an der Taylorreihe sehen:

$$y(x_{i-1}) = y(x_i) + \frac{\partial y}{\partial x}(x_i)(-\Delta x) + O(\|\Delta x\|^2) \quad (4.26)$$

Eine dritte Möglichkeit ist der *Zentrale-Differenzenquotient*, der bereits bei der Taylorentwicklung hergeleitet wurde. Vergleiche hierzu (4.15) für kleine Schrittweiten h :

$$D_0y(\bar{x}, t) = y_x(x_i, t) \approx \frac{y(x_{i+1}, t) - y(x_{i-1}, t)}{2h} \quad (4.27)$$

Bemerkung 4.15

Dies stellt den Anstieg der Linieninterpolation y an x_{i+1} und x_{i-1} dar und schließt den Vorteil der beiden einseitigen Approximationen ein. Wir erhalten eine Approximation mit Genauigkeit zweiter Ordnung (vgl. Gleichung (4.15)), das heißt, dass der Fehler jetzt proportional zu h^2 und daher sehr viel kleiner ist als der Fehler für eine Approximation erster Ordnung. Dies setzt allerdings ein ausreichend kleines h voraus. \circ

Der Zentrale-Differenzenquotient erster Ordnung besitzt folgende graphisch Darstellung, die der Darstellung der Sekantensteigung sehr ähnlich ist (vgl. Abbildung 4.8).

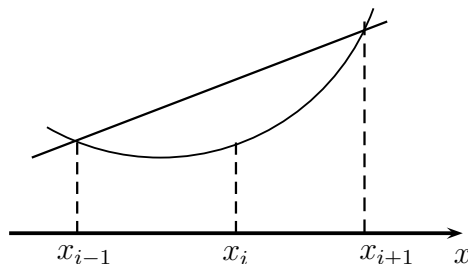


Abbildung 4.8: Zentrale Differenzenformel

4.2.2.2 Differenzenapproximationen zweiter Ordnung

Approximationen für zweite Ableitungen $y_{xx}(x_i, t)$ können in analoger Weise zum zentralen Differenzenquotienten für die erste Ableitung bestimmt werden. Dazu werden die beiden Taylorreihen (4.12) und (4.13) addiert. Die zentrale Approximation zweiter Ordnung ist somit für kleine Schrittweiten h gegeben durch:

$$D^2y(\bar{x}, t) = y_{xx}(x_i, t) \approx \frac{y(x_{i+1}, t) - 2y(x_i, t) + y(x_{i-1}, t))}{h^2} \quad (4.28)$$

Bemerkung 4.16

Da dies eine *symmetrische* zentrale Approximation ist, gilt auch hier, dass alle Terme ungerader Ordnung herausfallen. Symmetrisch bedeutet in diesem Zusammenhang, dass die in der Approximation verwendeten Punkte x_{i+1} und x_{i-1} gleich weit von x_i entfernt sind (vgl. (4.28)). Es gibt auch andere Möglichkeiten, diese Approximation zu erreichen:

- Berechnung der zweiten Ableitung der quadratisch polynomialen Interpolation $y(x)$ bei x_{i+1} , x_i und x_{i-1} oder
- mehrmalige Anwendung der Differenzenquotienten erster Ordnung ◦

Für weitere finite Differenzenapproximationen sei auf das Werk D. J. EVANS [14] und für allgemeine Informationen auf R. LI und Z. CHEN [36] verwiesen.

4.2.2.3 Differenzenapproximation der Neumann-Randbedingung

Die gesonderte Betrachtung der Neumann-Randbedingung ist bei randnahen Gitterpunkten zwingend. Bei inneren Gitterpunkten können oben genannte Approximationen verwendet werden. Zur Motivation gehen wir davon aus, dass der Rand mit der Gitterlinie, die parallel zur t -Achse (in der Abbildung nach oben) läuft, zusammenfällt und die äußere Normale in Richtung der positiven x -Achse (in der Abbildung nach rechts) zeigt (siehe Abbildung 4.9 in Anlehnung an C. BÜSKENS [8]).

Wir führen also einen Hilfsgitterpunkt ein, der den Wert y_E besitzt. Somit kann die Normalenableitung mittels des zentralen Differenzenquotienten approximiert werden:

$$\frac{\partial y}{\partial \nu}(x, t) \approx \frac{y_E(x, t) - y_W(x, t)}{2h} = 0 \quad (4.29)$$

und da $h \neq 0 \Rightarrow y_E(x, t) = y_W(x, t)$.

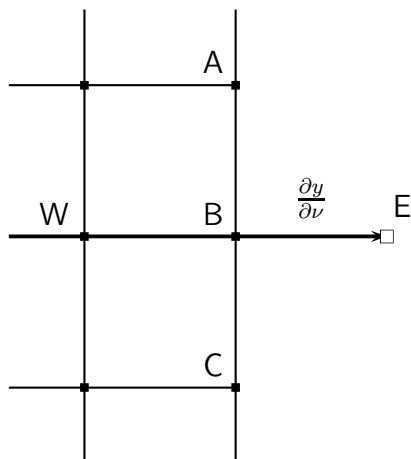


Abbildung 4.9: Normalenableitung

Bei dieser Darstellung verschwindet die Normalenableitung und ist somit gleich Null. Das heißt, dass $y(x, t)$ bezüglich des Randes symmetrisch ist. Aufgrund dieser Symmetrieeigenschaft ist es erlaubt, $y(x, t)$ über den Rand hinaus fortzusetzen.

Dieses Verschwinden kann man zum Beispiel am entsprechenden mathematischen Modell eines *idealen* Kühlschranks, d.h. mit vollständiger Isolierung, beobachten, da hier keine Temperatur nach außen abgegeben wird.

4.2.3 Upwind-Methode

Die Upwind-Methode findet vor allem bei Transportvorgängen, z.B. $y_t + ay_x = 0, a \in \mathbb{R}$, ihre Verwendung (vgl. [5, 28, 39]), da für eine stabile Diskretisierung einseitige Differenzenquotienten entgegen der Richtung des Flusses („upwind“) gewählt werden müssen. Diese nennt man auch CIR-Verfahren⁶. Die Richtung der Differenzenbildung wird dabei immer *stromauf* vorgenommen, das heißt, dass beim Vorwärts-Differenzenquotienten nur Informationen rechts von x_i und beim Rückwärts-Differenzenquotienten nur Informationen links von x_i verwendet werden. Anders ausgedrückt werden nur Informationen aus der „Vergangenheit“ benutzt, um die Approximation zu berechnen.

In unserem Beispiel entspricht $a > 0$ einem Fluss in x -Richtung nach rechts und $a < 0$ einem Fluss in x -Richtung nach links. Somit wählen wir folgende Differenzenquotienten:

$$a > 0 : D_- y(\bar{x}, t) = y_x(x_i, t) \approx \frac{y(x_i, t) - y(x_{i-1}, t)}{h} \tag{4.30}$$

$$a < 0 : D_+ y(\bar{x}, t) = y_x(x_i, t) \approx \frac{y(x_{i+1}, t) - y(x_i, t)}{h} \tag{4.31}$$

Bemerkung 4.17

Dies sind die beiden eingeführten einseitigen Differenzenapproximationen, da sie Daten nur auf einer Seite des Punktes x_i verwenden (siehe Abschnitt 4.2.2.1). ◦

⁶nach Courant, Isaacson und Rees

Es mag zunächst unklar sein, warum wir solche Approximationen verwenden, wenn doch zentrierte Approximationen genauer sind.

Dies erklärt sich aber dadurch, dass es für manche Transportgleichungen eine Asymmetrie innerhalb der Gleichungen gibt. Sie kommt aufgrund der Modellierung nach der Variable a , also der Geschwindigkeit, zustande. Wenn $a > 0$ gilt, bewegt sich die Lösung nach rechts, während, wenn $a < 0$ gilt, bewegt sie sich nach links. Es gibt Situationen, in denen es deswegen sinnvoll ist, diese Eigenschaft einer Asymmetrie zu kennen und entsprechend der Richtung einseitige Differenzen zu verwenden.

Für ausführliche Beispiele verweisen wir auf weitere Literatur, wie zum Beispiel die Dissertation von J. M. CHARTON [23] und insbesondere auf die Arbeit von H. P. LANGTANGEN [34].

Nun haben wir alle Bausteine, die wir für die beiden in der Auswertung betrachteten Systeme (Reaktions-Advektions-Diffusions-Gleichung und Wellengleichung) benötigen, eingeführt und definiert. Deshalb sei hier nur kurz angegeben, welche Differenzenquotienten wir in der Auswertung der Systeme verwenden. Für eine ausführliche Beschreibung mittels der Approximationen sei an dieser Stelle auf das Kapitel 6 verwiesen.

Bemerkung 4.18

Bei der Auswertung der Reaktions-Advektions-Diffusions-Gleichung

$$y_t = -y_x + \nu y_{xx} + \mu y(y+1)(1-y) + u \quad (4.32)$$

haben wir für die Diskretisierung der zweiten Ableitung y_{xx} die zentrale Differenzenapproximation (4.28)

$$y_{xx}(x_i, t) \approx \frac{y(x_{i+1}, t) - 2y(x_i, t) + y(x_{i-1}, t))}{h^2} \quad (4.33)$$

verwendet. Der Grund liegt hier in der größeren Genauigkeitsordnung im Vergleich zur einseitigen Approximation. Für die erste Ableitung verwenden wir entsprechend der Upwind-Methode (siehe Abschnitt 4.2.3) den Vorwärts-Differenzenquotienten (4.23)

$$y_x(x_i, t) \approx \frac{y(x_{i+1}, t) - y(x_i, t)}{h}, \quad (4.34)$$

da die Flussrichtung⁷ $a = -1$, also negativ ist und somit den Vorwärts-Differenzenquotienten erzwingt.

In der Wellengleichung

$$y_{tt} = c^2 \cdot y_{xx} \quad \text{mit } c > 0 \quad (4.35)$$

o

approximieren wir die zweite Ableitung y_{xx} ebenfalls durch den zentralen Differenzenquotienten (4.28).

⁷hier der Faktor vor der erste Ableitung y_x .

4.2.4 Stabilität bezüglich der Finite-Differenzen-Methode

Für die Ausführung von verschiedenen Verfahren und ihrer praktischen Berechnung spielt deren Stabilität eine zentrale Rolle. Um die Stabilität eines Verfahrens bzw. Näherungsverfahrens zu beurteilen, benötigt man Informationen über die Eigenschaften der Differentialgleichung und der Lösung. Für eine Instabilität der numerischen Lösung eines Anfangswertproblems (PDE) gibt es zwei mögliche Gründe:

- das Problem an sich ist bereits instabil oder
- das angewendete numerische Verfahren ist instabil.

Wir benötigen somit Informationen über das Problem, das die Grundvoraussetzung ein „gut gestelltes“ Problem (siehe Abschnitt 2.2.3) zu sein, erfüllt. Diese stetige Abhängigkeit von den gegebenen Daten wird bereits als „Stabilität des mathematischen Modells“ (C.-D. MUNZ und T. WESTERMANN [39]) angesehen (vgl. auch R. J. LEVEQUE [35]). Das heißt konkret, dass Fehler wie Rundungs- oder Diskretisierungsfehler nicht beliebig wachsen können. Trotzdem kann es zu sehr unterschiedlichem Verhalten der Lösungen kommen.

Wenn bei numerischer Simulation wachsende Störungen auftreten, nennt man dieses Verhalten *Instabilität des numerischen Verfahrens*. Diese Methode ist dann für praktische Anwendungen nicht sinnvoll. Somit sollte darauf geachtet werden, dass das Ausgangsproblem von vornherein stabil ist.

Bevor wir den Begriff der Stabilität definieren, wollen wir kurz anhand eines Beispiels zeigen, wie man von den Bausteinen der Finite-Differenzen-Methode zum linearen Gleichungssystem kommt. Wir betrachten die Differentialgleichung

$$y_{xx}(x_i, t) = f(x) \quad , \quad 0 < x < 1$$

$$y(0, t) = 0 \quad , \quad y(1, t) = 0$$

mit fest vorgegebener Funktion $f = f(x)$ und verwenden die Differenzenapproximation zweiter Ordnung (4.28)

$$y_{xx}(x_i, t) \approx \frac{y(x_{i+1}, t) - 2y(x_i, t) + y(x_{i-1}, t)}{h^2}.$$

Durch diese Ersetzung mit dem zentralen Differenzenquotienten erhalten wir eine Menge von algebraischen Gleichungen

$$\frac{y(x_{i+1}, t) - 2y(x_i, t) + y(x_{i-1}, t)}{h^2} = f(x_i) \quad \text{mit } j = 1, \dots, m. \quad (4.36)$$

Dabei ist zu beachten, dass hier die Werte $y(0, t) = 1$ und $y(1, t) = 1$ direkt in die

erste bzw. letzte Gleichung mit einfließen. Wir haben nun ein lineares System mit m Gleichungen für unsere m Unbekannten, das folgendermaßen geschrieben werden kann:

$$AY = F, \quad (4.37)$$

wobei Y der Vektor der Unbekannten $Y = (Y_1, \dots, Y_m)^T$ ist und

$$A = \frac{1}{h^2} \begin{pmatrix} -2 & 1 & & & & & \\ & 1 & -2 & 1 & & & \\ & & 1 & -2 & 1 & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & 1 & -2 & 1 \\ & & & & & 1 & -2 \end{pmatrix} \quad \text{und} \quad F = \begin{pmatrix} f(x_1) - \frac{1}{h^2} \\ f(x_2) \\ f(x_3) \\ \vdots \\ \vdots \\ f(x_m) - \frac{1}{h^2} \end{pmatrix}. \quad (4.38)$$

Analog können andere Systeme mit der Finite-Differenzen-Methode in ein lineares Gleichungssystem umgeschrieben werden.

Jetzt lässt sich kurz und prägnant die Stabilität definieren:

Definition 4.19 (Stabilität)

Angenommen, eine Finite-Differenzen-Methode für ein lineares Randwertproblem erzeugt eine Folge von Matrixgleichungen

$$A^h Y^h = F^h \quad (4.39)$$

mit der Schrittweite h .

Das Verfahren ist stabil, wenn $(A^h)^{-1}$ existiert für alle h genügend klein und eine Konstante C unabhängig von h , so dass

$$\|(A^h)^{-1}\| \leq C \quad \forall h < h_0 \quad (4.40)$$

gilt. □

Bemerkung 4.20

Da die Stabilität von der jeweiligen PDE abhängt, die betrachtet wird, wollen wir hier nur kurz und ohne Beweise auf die einzelnen Differenzenapproximationen erster Ordnung und somit auch auf die Upwind-Methode eingehen. Ausführliche Informationen finden sich in der Arbeit von R. J. LEVEQUE [35]. ◦

Vergleicht man die exakte Lösung eines Verfahrens mit seiner Näherungslösung kann es vor allem bei Transportgleichungen vorkommen, dass die Näherungslösung oszilliert, während die exakte Lösung einen völlig gleichmäßigen Verlauf aufweist. Dieses Verhalten ist nicht auf die Konsistenzordnung zurückzuführen, vielmehr hat man in diesen Fällen ein Stabilitätsproblem vor sich. Es können sich zum Beispiel Fehler aus einer ungenauen Approximation an einem Rand in das komplette Gebiet fortpflanzen. Erst bei sehr starker Verkleinerung des Gitters, können wir meist eine Verbesserung feststellen. Dies lässt sich

anhand der nicht mehr *diagonal dominanten*⁸ Tridiagonalmatrix A für festes h begründen. Das Problem lässt sich nur mit einer Verkleinerung der Schrittweite h beheben. Alternativ tritt auch bei Verwendung der einseitigen Differenzenquotienten eine Verbesserung ein, denn in diesem Fall bleibt die Tridiagonalmatrix A immer diagonal dominant. Deswegen erhalten wir auch bei exakter Verwendung der Upwind-Methode eine stabile Lösung.

4.3 Methode der Liniendiskretisierung (Linienmethode)

Wie zu Beginn bereits erwähnt ist die Linienmethode eine Semidiskretisierung. Dazu wollen wir die allgemeine Definition für Linienmethoden und im Anschluss eine Auswahl an unterschiedlichen Variationen betrachten.

4.3.1 Definition

Das Ziel der folgenden klassischen Linienmethoden ist es, eine Diskretisierung in x - oder t -Richtung einzuführen, sodass sich anschließend ein Anfangswertproblem für ein System gewöhnlicher Differentialgleichungen ergibt. Linienmethoden werden verwendet, indem zuerst nach einer Variable diskretisiert wird, um anschließend mit Hilfe eines Software Pakets für gewöhnliche Differentialgleichungen das Problem zu lösen. Es gibt Pakete, die speziell für den Gebrauch der Linienmethode entwickelt wurden. Wir geben nun die Definition der allgemeinen Linienmethode an, die als Orientierung für die nachfolgenden Definition der einzelnen Spezialisierungen dienen soll:

Definition 4.21 (Linienmethode)

Sei $x = (x_1, \dots, x_n)$ ein Vektor der unabhängigen Variablen einer PDE $L[y] = f(x)$ und $\bar{x} = (\bar{x}_1, \dots, \bar{x}_{\bar{n}})$, $0 \leq \bar{n} \leq n$ ein Vektor mit $\bar{x}_i \in \{x_1, \dots, x_n\}$, $i = 1, \dots, \bar{n}$ und $\bar{x}_i \neq \bar{x}_j$, $i \neq j$. Eine Approximation von $L[y] = f(x)$ durch ein System $L_{\bar{x}}[y] = \tilde{f}(x)$ mit einer geeignet gewählten Funktion \tilde{f} wird *Linienmethode* genannt. Das bedeutet, $L_{\bar{x}}[y]$ enthält nur noch partielle Ableitungen nach x_j mit $j = 1, \dots, \bar{n}$. \square

Wir werden uns in den Auswertungen auf die beiden Variablen x und t beschränken, d.h. die Definition 4.21 wird bei uns auf $\bar{x} = t$ eingeschränkt. Deswegen haben wir zwei Möglichkeiten unterschiedlicher Reihenfolge von Diskretisierungen nach Raum und Zeit. Wir werden hier in dieser Arbeit die *vertikale Linienmethode* verwenden. Eine andere Möglichkeit stellt die *horizontale Linienmethode* oder auch *Rothe-Methode* dar (vgl. hierfür die Arbeiten von C. BÜSKENS [8] und C. GROSSMANN und H.-G. ROSS [23]).

⁸Eine Matrix A ist diagonal dominant, wenn für alle Matrixeinträge gilt:

$$|a_{ij}| > \sum_{i=1}^n |a_{ij}|, \quad i = 1, \dots, n, \quad i \neq j.$$

4.3.2 Vertikale Linienmethode

Die vertikale Linienmethode ist eine Semidiskretisierung nach der örtlichen Variable bzw. den örtlichen Variablen. Die grundlegende Idee besteht also darin, die x -Richtung zu diskretisieren und die t -Richtung kontinuierlich zu belassen:

$$y_i(t) = y(x_i, t) \tag{4.41}$$

Beispiel 4.22

Ein System $y_t = y_{xx}$ wird durch eine Differenzenapproximation zweiter Ordnung auf ein System gewöhnlicher Differentialgleichungen transformiert:

$$\frac{\partial}{\partial t} y_i(t) = y_t(x_i, t) = \frac{y_{i+1}(t) - 2y_i(t) + y_{i-1}(t)}{\Delta x^2}, \quad i = 1, \dots, M - 1 \tag{4.42}$$

Bemerkung 4.23

Randbedingungen müssen am Rand nicht als Approximationen eingeführt werden, da die Lösung der PDE dort durch diese bereits bekannt ist. Die Randbedingungen fließen also direkt in das Problem ein. ◦

Wir wollen in Anlehnung an die Definition 4.21 die vertikale Linienmethode allgemein definieren:

Definition 4.24 (Vertikale Linienmethode)

Sei $L_{x_i}[y] = \tilde{f}(x)$ mittels einer Linienmethode aus $L[y] = f(x)$ entstanden.

Stellt die Veränderliche x_i die Zeit dar, also $x_i = t$, so wird die Approximation

$$L_{x_i}[y] = L_t[y] = \tilde{f}(x) \tag{4.43}$$

vertikale Linienmethode genannt. ◻

In der Abbildung 4.10 betrachten wir diese Methode mit zwei Raumdimensionen x_1 und x_2 und der Zeit t . Ein Gebiet wird mit vertikalen Linien in t -Richtung überzogen, so dass nur noch an Punkten auf diesen Linien approximiert werden muss. Das heißt, dass die beiden Raumdimensionen diskretisiert werden und nur die Zeit t kontinuierlich bleibt.

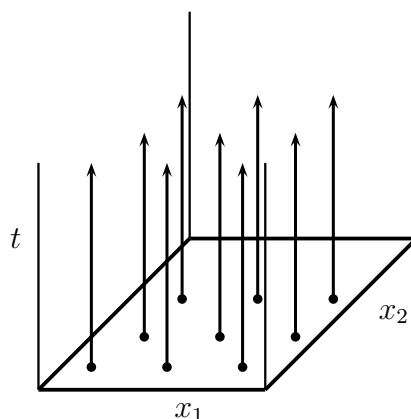


Abbildung 4.10: Vertikale Linienmethode

4.3.3 Horizontale Linienmethode

Analog zur vertikalen Linienmethode liegt bei der horizontalen Linienmethode die Idee zugrunde, die Variable t zu diskretisieren und gleichzeitig die örtliche Variable bzw. örtlichen Variablen kontinuierlich zu belassen:

$$y_i(t) = y(x, t_i) \tag{4.44}$$

Beispiel 4.25

Wir können das oben genannte Beispiel $y_t = y_{xx}$ wieder aufnehmen, verwenden hier jedoch eine Differenzenapproximation erster Ordnung, um ein System von gewöhnlichen Differentialgleichungen zu erhalten:

$$\frac{\partial^2}{\partial x^2} y_i(t) = y_{xx}(x, t_i) = \frac{y_i(x) - y_{i-1}(x)}{\Delta t}, \quad i = 1, \dots, N, \quad \Delta t = t_i - t_{i-1} \tag{4.45}$$

$N \in \mathbb{N}$ stellt die Anzahl der äquidistanten Zeitpunkte $t_i = i \cdot \Delta t = ik$ mit $k = \frac{1}{N}$ dar, analog zur Gitterdiskretisierung in x -Richtung. •

Bemerkung 4.26

Auch hier müssen keine zusätzlichen Approximationen am Rand eingeführt werden, da die Lösung der PDE dort durch die Randbedingungen bekannt ist. Die Randbedingungen fließen also wieder direkt in das Problem ein. ◦

Wir wollen analog zu 4.24 die horizontale Linienmethode allgemein definieren:

Definition 4.27 (horizontale Linienmethode)

Sei $L_{\bar{x}}[y] = \tilde{f}(x)$ mittels einer Linienmethode aus $L[y] = f(x)$ entstanden.

Tritt die Zeit im Originalproblem auf, also

$$\exists i \in \{1, \dots, n\} \quad \text{mit} \quad x_i = t, \tag{4.46}$$

nicht aber in der Transformation, also

$$\bar{x}_i \notin \{x_1, \dots, x_n\}, \tag{4.47}$$

so wird die Approximation $L_{\bar{x}}[y] = \tilde{f}(x)$ *horizontale Linienmethode* oder *Rothe-Methode* genannt. ◻

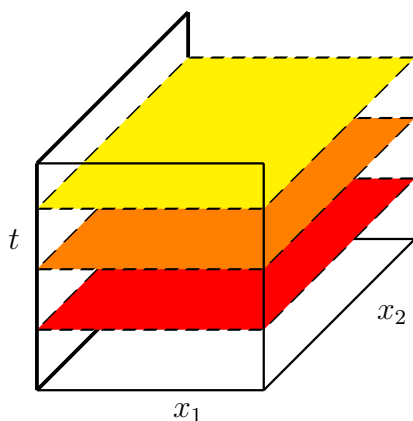


Abbildung 4.11: Horizontale Linienmethode

In der Abbildung 4.11 sehen wir, wie man sich diese Methode mit zwei Raumdimensionen x_1 und x_2 und der Zeit t vorzustellen hat. Ein Gebiet wird mit horizontalen Ebenen überzogen, da wir zwei Variablen festhalten. Bei dieser Linienmethode wird nur die Zeit t diskretisiert und die beiden Raumdimensionen x_1 und x_2 kontinuierlich gehalten. Es muss also eine partielle Differentialgleichung in zwei Raumdimensionen pro Zeitschritt gelöst werden.

Es gibt Linienmethoden, die weder der vertikalen noch der horizontalen Methode zuzuordnen sind, zum Beispiel bei elliptischen PDEs. Wenn wir Probleme mit mehreren Raumvariablen vorliegen haben, gelangen wir mit der horizontalen Linienmethode zu einer elliptischen PDE, die in jedem Zeitschritt gelöst werden muss.

Bemerkung 4.28

In Bezug auf die späteren Auswertungen ist für uns die vertikale Linienmethode entscheidend, da wir ausgehend von unseren betrachteten kontinuierlichen Systemen (Reaktions-Advektions-Diffusions-Gleichung und Wellengleichung) in einem ersten Schritt die Zeit t kontinuierlich wählen, während eine räumliche Diskretisierung in x -Richtung durchgeführt wird. Diese Diskretisierung, die dazu dient auf ein semidiskretes System zu kommen, basiert wiederum auf den Finite-Differenzen-Methoden, die im letzten Abschnitt 4.2.2 bereits dargestellt wurden. ◦

4.3.4 Alternative Methoden

Neben den nun bekannten Methoden gibt es viele alternative Linienmethoden, die wir nicht alle hier nennen wollen. Exemplarisch erwähnen wir noch drei Methoden⁹ in Anlehnung an R. J. LEVEQUE [35], werden aber nicht näher auf diese eingehen. Für weitere Informationen sei auf D. J. EVANS [14] und H. P. LANGTANGEN [34] verwiesen.

4.3.4.1 Vorwärts-Euler-Zeitdiskretisierung

Für das Beispiel

$$y_t + ay_x = 0 \quad \text{mit } a = \text{const.} \tag{4.48}$$

erhalten wir durch die Approximation

$$\frac{y_{i+1}(x) - y_i(x)}{\Delta t} = -a \frac{y_{i+1}(t) - y_{i-1}(t)}{2\Delta x} \tag{4.49}$$

die *Vorwärts-Euler-Zeitdiskretisierung* der Linienmethode:

$$y_{i+1}(x) = y_i(x) - \frac{a\Delta t}{2\Delta x} (y_{i+1}(t) - y_{i-1}(t)). \tag{4.50}$$

⁹Zeit- und ortsabhängige Diskretisierungen

4.3.4.2 Leapfrog (Bocksprung)

Eine bessere Zeitdiskretisierung in Bezug auf das obige Beispiel (4.48) ist die Verwendung von zentralen Approximationen für y_x und y_t . Dadurch erhalten wir mit

$$\frac{y_{i+1}(x) - y_{i-1}(x)}{2\Delta t} = -a \frac{y_{i+1}(t) - y_{i-1}(t)}{2\Delta x} \quad (4.51)$$

ein so genanntes „3-Zeitniveau-Schema“ (auch genannt *Leapfrog*):

$$y_{i+1}(x) = y_{i-1}(x) - \frac{a\Delta t}{\Delta x} (y_{i+1}(t) - y_{i-1}(t)). \quad (4.52)$$

Diese Methode ist vor allem für hyperbolische PDEs gut geeignet, ist aber instabil für parabolische PDEs (vgl. [14]). Sie besitzt aufgrund der zentralen Approximationen eine Genauigkeit zweiter Ordnung in Ort und Zeit.

4.3.4.3 Lax-Friedrich

Als letzte Variante wollen wir noch die *Lax-Friedrich-Methode* nennen, die für $y_i(x)$ einen Durchschnittswert $\frac{1}{2}(y_{i-1}(x) + y_{i+1}(x))$ verwendet. Somit erhalten wir analog zur Vorwärts-Euler-Zeitdiskretisierung:

$$y_{i+1}(x) = \frac{1}{2}(y_{i-1}(x) + y_{i+1}(x)) - \frac{a\Delta t}{2\Delta x} (y_{i+1}(t) - y_{i-1}(t)). \quad (4.53)$$

Da diese Methode geringe Genauigkeit hat, wird sie in der Praxis selten verwendet, jedoch dient sie dazu, einige Stabilitätseigenschaften zu zeigen.

4.3.5 Stabilität bzgl. der Linienmethode

Wie im vorherigen Abschnitt 4.2.4 über die Stabilität der Finite-Differenzen-Methoden, wollen wir hier nur kurz auf die wichtigsten Punkte in Bezug auf Stabilität der hier genannten Linienmethoden eingehen. Weitere Informationen und Analysen finden sich zum Beispiel in dem Skript von C. BÜSKENS [8], in den Arbeiten von D. J. EVANS [14], H. P. LANGTANGEN [34] und R. J. LEVEQUE [35]. Sowohl bei der vertikalen, als auch bei der horizontalen Linienmethode hängt die Stabilität der Verfahren von den gewählten Differenzenquotienten ab. Wir haben bereits im letzten Abschnitt gesehen, dass die Verwendung von einseitigen Approximationen weniger Schwierigkeiten macht als die Verwendung von zentralen Approximationen. Jedoch möchte man wegen der besseren Genauigkeitsordnung zentrale Approximationen verwenden. Aus diesem Grund muss überprüft werden, ob das System von gewöhnlichen Differentialgleichungen nach der Diskretisierung *steif*¹⁰ ist oder nicht, da durch diese Eigenschaft Stabilitätsverluste möglich sind. Der Begriff der Steifheit lässt sich nur sehr schwer mathematisch fassen: Steifheit beschreibt Vorgänge, deren Lösung sich aus zwei stark unterschiedlichen Anteilen zusammensetzen. Vor allem bei chemischen Reaktionen ist dieses Phänomen oft zu beobachten. Bei zu großer Schrittweite h können dadurch Probleme bezüglich der Stabilität auftreten. Stabilität bedeutet

¹⁰d.h. falls Eigenwerte der Matrix A betragsmäßig große negative Realteile besitzen.

in diesem Zusammenhang, dass ein lokaler Approximationsfehler durch das angewendete Verfahren nicht verstärkt wird. Auf der sicheren Seite ist man, wenn man ein „A-stabiles“-Verfahren verwendet:

Definition 4.29 (A-stabil)

Ein numerisches Verfahren heißt *A-stabil*, falls seine Anwendungen auf eine Differentialgleichung für jede Schrittweite $h > 0$ stets eine nicht wachsende Folge von Approximationen $\{y_i\}$ produziert.

$$|y_i| \leq |y_0| \forall i \tag{4.54}$$

□

Eine ausführliche Stabilitätsanalyse findet sich in [35], vor allem in Bezug auf die alternativen Methoden. Wir nennen nur die Ergebnisse in Bezug auf Beispiel (4.48):

- Das Euler-Verfahren ist instabil für jegliche Diskretisierung Δx oder Δt .
- Das Leapfrog-Verfahren ist stabil für $\frac{a\Delta t}{\Delta x} < 1$.
- Das Lax-Friedrich-Verfahren ist ebenfalls nur für $\frac{a\Delta t}{\Delta x} < 1$ stabil.

4.4 Konsistenz und Konvergenz der Methoden

Konsistenz, Konvergenz und auch Stabilität gehören zu den grundlegenden Qualitätskriterien für Näherungsverfahren (vgl. C.-D. MUNZ und T. WESTERMANN [39]). Die Konsistenz wird dabei wie bei den gewöhnlichen Differentialgleichungen über den lokalen Diskretisierungsfehler bestimmt, d.h. über die Differenz von Näherungswert und exakter Lösung im neuen Punkt. Dies entspricht dem Fehler, den man in einem Schritt macht. Dabei wird vorausgesetzt, dass der Wert der Näherungslösung im n -ten Schritt mit dem Wert der exakten Lösung übereinstimmt. Eventuell müssen nur kleine Anpassungen bezüglich der partiellen Ableitungen vorgenommen werden. Konsistenz sagt aus, dass bei gegen Null strebender Schrittweite der lokale Diskretisierungsfehler ebenfalls gegen Null geht. Um den Begriff der Konsistenz zu definieren, schreiben wir ein gegebenes Problem in der Form

$$Ly - f = 0 \tag{4.55}$$

mit dem Differentialoperator L , der eventuelle Anfangs- und Randbedingungen beinhaltet. Analog ist ein Näherungsverfahren mittels der Vorschrift

$$L_h y_h - f_h = 0 \tag{4.56}$$

mit der Approximation L_h , der Näherungslösung y_h und der Diskretisierung h gegeben. Wird die exakte Lösung in das Näherungsverfahren eingesetzt, ergibt sich ein Fehler, den man auch als Residuum bezeichnet.

Definition 4.30 (Konsistenz)

Ein numerisches Verfahren ist *konsistent* und besitzt die *Approximations-* oder *Konsistenzordnung* p , wenn für den lokalen Diskretisierungsfehler die Ungleichung

$$\|L_h u_h - f_h\| \leq C h^p \quad (4.57)$$

mit einer von h unabhängigen Konstanten C gilt. \square

Bei mehreren Diskretisierungsparametern müssen wir entsprechend, zum Beispiel mit den Raum- und Zeitschritten Δx und Δt erweitern zu

$$\|L_h u_h - f_h\| = O(\Delta x^p, \Delta t^q). \quad (4.58)$$

Wir sagen dann, dass das numerische Verfahren konsistent mit Ordnung p bzgl. der Raumapproximation und mit Ordnung q bzgl. der Zeitapproximation ist.

Bei Differenzenverfahren erhalten wir nur Näherungswerte an den einzelnen Gitterpunkten. Dies führt dazu, dass wir die Werte der exakten Lösung an diesen diskreten Punkten in die Näherungsformel einsetzen und somit die exakte Lösung auf das jeweilige Gitter einschränken.

Auch die Stabilität ist eine wichtige Eigenschaft für sinnvolle numerische Verfahren. Bereits bei den gewöhnlichen Differentialgleichungen ist bekannt, dass die Stabilitätsanforderungen problemabhängig sind. Besonders Verfahren, die auf steife Differentialgleichungen angewendet werden, benötigen starke Stabilitätsbedingungen. Bei den PDEs ist die Beachtung dieser Bedingungen noch entscheidender, da die Eigenschaften und Lösungsstrukturen um einiges vielfältiger sind.

Als letzte wichtige Eigenschaft eines Verfahrens betrachten wir die Konvergenz eines Näherungsverfahrens, also die Konvergenz der Näherungslösung gegen die exakte Lösung, wenn die Schrittweite h gegen 0 strebt.

Definition 4.31 (Konvergenz)

Sei e_h der globale Diskretisierungsfehler

$$e_h = v_h - y_h. \quad (4.59)$$

Dann bedeutet *Konvergenz*, dass die Ungleichung

$$\|e_h\| \leq C h^q \quad (4.60)$$

mit einer geeigneten Norm erfüllt wird. q wird Konvergenzordnung des Verfahrens genannt. \square

Die drei betrachteten Begriffe sind nicht unabhängig voneinander, sondern stehen in enger Beziehung. Die Konsistenz eines numerischen Verfahrens bewirkt einen kleinen lokalen Diskretisierungsfehler, wenn die Schrittweite klein ist. Außerdem heißt Stabilität eines Verfahrens, dass der Fehler zwischen Näherung und exakter Lösung nicht unbeschränkt wachsen kann. Für lineare Probleme gilt das fundamentale Äquivalenztheorem von Lax, das aussagt, dass die Stabilität eine notwendige und hinreichende Bedingung für die Konvergenz ist, falls wir ein konsistentes Diskretisierungsschema vorliegen haben. Für nichtlineare Probleme gilt dieses so genannte Lax-Theorem zwar nicht, aber Stabilität und Konsistenz sind auch hier wichtige Eigenschaften für ein vernünftig funktionierendes Verfahren.

Konsistenz + Stabilität = Konvergenz

Die Stabilität wird im folgenden Sinne betrachtet: Eine Methode heißt stabil, falls der Lösungsfehler, d.h. die betragsmäßige Differenz zwischen berechneter und exakter Lösung, begrenzt bleibt. Ein Beweis findet sich dazu unter anderem in der Arbeit J. W. THOMAS [49].

5 Lösungsräume partieller Differentialgleichungen

Wir wollen in diesem Kapitel einen kurzen Einblick in die funktionalanalytischen Grundlagen der Lösungsräume partieller Differentialgleichungen geben. Dabei spielen die sogenannten *Sobolev-Räume* eine entscheidende Rolle, die wir im folgenden Kapitel betrachten wollen. Des Weiteren legen wir den Fokus im Besonderen auf die jeweils für die verschiedenen Räume definierten *Normen*, vor allem auf die L^2 - und die H^1 -Norm. Diese finden bei der Definition der Kostenfunktion l unseres nichtlinearen modellprädiktiven Regelungsproblems 3.42 Verwendung und spielen damit eine große Rolle in Bezug auf die Stabilisierung des betrachteten Systems.

In diesem Kapitel orientieren wir uns vornehmlich an F. TRÖLTZSCH [51], L. C. EVANS [15] und K. THEISSEN [48], sowie in Auszügen an H. W. ALT [2], D. WERNER [55] und C. SIMADER [46].

Zu Beginn wollen wir einige benötigte Definitionen nennen. Dabei bezeichnet $C([a, b])$ den Raum aller auf $[a, b] \subset \mathbb{R}$ stetigen reellwertigen Funktionen und $C^k([a, b])$ den Raum aller auf $[a, b]$ k -mal stetig differenzierbaren reellwertigen Funktionen. Des Weiteren setzen wir den Begriff des *linearen Raums* (oder auch *Vektorraum*) als bekannt voraus.

Definition 5.1 (Norm, normierter Raum)

Sei X ein Vektorraum über dem Körper $\mathbb{K} = \mathbb{R}$ der reellen Zahlen. Eine Abbildung $\|\cdot\| : X \rightarrow \mathbb{R}$, die folgende Eigenschaften

- *Definitheit*: $\|f\| \geq 0$ für alle $f \in X$ und $\|f\| = 0 \iff f = 0$
- *Homogenität*: $\|\lambda f\| = |\lambda| \cdot \|f\|$ für alle $f \in X$ und $\lambda \in \mathbb{K}$
- *Dreiecksungleichung*: $\|f + g\| \leq \|f\| + \|g\|$ für alle $f, g \in X$

erfüllt, heißt *Norm auf X* , das Paar $(X, \|\cdot\|)$ heißt *normierter Raum*. □

Bemerkung 5.2

Der Raum \mathbb{R}^n , versehen mit der *Euklidischen Norm*

$$|x| = \left(\sum_{i=1}^n x_i^2 \right)^{1/2} \quad (5.1)$$

ist ein linearer normierter Raum.

Der Raum $C([a, b])$ aller auf $[a, b] \subset \mathbb{R}$ definierten und stetigen reellwertigen Funktionen wird mit der *Maximum-Norm*

$$\|x\|_{C([a,b])} = \max_{t \in [a,b]} |x(t)| \quad (5.2)$$

zu einem linearen normierten Raum. ◦

Definition 5.3 (vollständig)

Ein normierter Raum $(X, \|\cdot\|)$ heißt *vollständig*, wenn jede Cauchy-Folge $(f_n)_{n \in \mathbb{N}} \subset X$ einen Grenzwert $f \in X$ besitzt. □

Definition 5.4 (Banach-Raum)

Ein vollständiger normierter Raum $(X, \|\cdot\|)$ heißt *Banach-Raum*. □

Definition 5.5 (Skalarprodukt)

Sei X ein Vektorraum über \mathbb{R} . Eine Abbildung $\langle \cdot, \cdot \rangle : X \times X \rightarrow \mathbb{R}$ heißt *Skalarprodukt* oder *inneres Produkt*, falls

- $\langle x + z, y \rangle = \langle x, y \rangle + \langle z, y \rangle,$
- $\langle \lambda x, y \rangle = \lambda \langle x, y \rangle,$
- $\langle x, y \rangle = \langle y, x \rangle,$
- $\langle x, x \rangle \geq 0,$
- $\langle x, x \rangle = 0 \iff x = 0$

für alle $x, y, z \in X$ und $\lambda \in \mathbb{R}$ gilt. □

Definition 5.6 (Prä-Hilbert-Raum, Hilbert-Raum)

Sei H ein Vektorraum über \mathbb{R} , auf dem ein Skalarprodukt definiert ist. Das Paar $(H, \langle \cdot, \cdot \rangle)$ heißt dann *Prä-Hilbert-Raum*. Einen vollständigen reellen Prä-Hilbert-Raum nennt man einen *Hilbert-Raum*. □

Bemerkung 5.7

Bei *Hilbert-Räumen* handelt es sich um Banach-Räume, die als zusätzliche Struktur ein Skalarprodukt zulassen, das wiederum eine Norm induziert. Aufgrund “des hohen Grades an mathematischer Struktur” (z.B. bezüglich der Definition unendlichdimensionaler Räume und Operatoren) dienen sie als Basis für funktionalanalytische Untersuchungen, vor allem im Bereich der Lösungstheorie partieller Differentialgleichungen (vgl. hierzu [27,55]).◦

Definition 5.8 (Lebesgue-messbar)

Sei $I := [a, b] \subset \mathbb{R}$ ein abgeschlossenes Intervall. Eine Funktion $g : I \rightarrow \mathbb{R}^m$ heißt *Lebesgue-messbar*, falls eine Folge von stückweise konstanten Funktionen $g_i : I \rightarrow \mathbb{R}^m, i \in \mathbb{N}$, existiert mit $\lim_{i \rightarrow \infty} g_i(x) = g(x)$ für fast alle $x \in I$. □

Die folgende Schreibweise soll die Notationen bezüglich der Differentiation vereinfachen:

Definition 5.9 (Multiindex)

Ein *Multiindex* α ist ein Vektor nichtnegativer ganzer Zahlen $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{N}_0^n$, der durch seine Komponenten $\alpha_i (i \in \{1, \dots, n\})$ angibt, wie oft nach dem entsprechenden x_i differenziert wird. Dabei wird die folgende Kurzbezeichnung verwendet:

$$D^\alpha y(x) := D_1^{\alpha_1} \dots D_n^{\alpha_n} y(x) \tag{5.3}$$

Die *Ordnung* von α wird durch $|\alpha| = \alpha_1 + \dots + \alpha_n$ definiert. □

Im Weiteren sei der Vektor $\alpha \in \mathbb{N}_0^n$ stets ein Multiindex nach obiger Definition.

5.1 Die Sobolev-Räume

Zur Motivation der Sobolev-Räume

Die Sobolev-Räume, benannt nach dem russischen Mathematiker SERGEI L. SOBOLEV (1908 – 1989), haben sich als eine geeignete “räumliche” Grundlage bei der Anwendung funktionalanalytischer Methoden zum Erhalt brauchbarer Ergebnisse und Informationen bezüglich der Lösungstheorie partieller Differentialgleichungen (kurz: PDE, aus dem Englischen für partial differential equation) herausgestellt.

Ein Hauptziel dabei ist es, die Anwendungen und Methoden einer großen Klasse von PDEs zugänglich zu machen. Dazu wird eine Anzahl verschiedenartiger PDEs in *Operatoren* $A : X \rightarrow Y$ “umgeformt”, die sozusagen den “Strukturen” der partiellen Differentialgleichungen entsprechen, und die auf linearen Räumen angewendet werden. X und Y stellen dabei die geeigneten Funktionenräume dar.

Der Vorteil besteht darin, dass eine geeignete Interpretation des PDE-Problems in dieser Form dazu verwendet werden kann, allgemein gültige Grundsätze und Methoden der Funktionalanalysis zur Überprüfung der Lösbarkeit von verschiedenen Gleichungen bezüglich des Operators A anzuwenden. Hierbei stellt sich aber besonders die Angabe geeigneter Funktionenräume X und Y als schwierig heraus.

Die im Nachfolgenden behandelten Sobolev-Räume wurden nun so konzipiert, dass sie sich als die geeignete Wahl dieser Funktionenräume erweisen.

Man vergleiche hierzu [15], S. 239.

Wir betrachten im Folgenden ein beschränktes Gebiet $\Omega \subset \mathbb{R}^n$ mit Rand $\Gamma := \partial\Omega$ und bezeichnen mit $\bar{\Omega}$ den *Abschluss* von Ω .

Die Theorie partieller Differentialgleichungen erfordert Gebiete Ω mit einem hinreichend glatten Rand (vergleiche [51], Abschnitt 2.2.2): Während für ein Gebiet mit Dimension $n = 1$, dessen Rand nur aus zwei Punkten besteht, keine weitere Klassifizierung notwendig ist, betrachtet man für $n \geq 2$ nur Gebiete mit einem so genannten *Lipschitz-* oder *$C^{0,1}$ -Rand*, den wir folgendermaßen definieren:

Sei ein Gebiet $\Omega \subset \mathbb{R}^n$ gegeben. Falls ein $d \in \mathbb{N}$ und offene Mengen $M_1, \dots, M_d \subset \mathbb{R}^n$ existieren, so dass gilt:

- $\Gamma \subset \bigcup_{1 \leq i \leq d} M_i$ und
- jeder Teilrand $\Gamma \cap M_i$ ist als Graph einer Lipschitz-stetigen Funktion h_i darstellbar,

so heißt Ω ein *Lipschitz-Gebiet* (auch reguläres Gebiet) oder *Lipschitz-Rand*.

Vergleiche für diese Definition [48].

Außerdem definieren wir die L^p -Räume (oder auch *Lebesgue-Räume*, benannt nach dem französischen Mathematiker H. L. LEBESGUE, 1875 – 1941), die im folgenden Verlauf Verwendung finden. Bei diesen handelt es sich um spezielle *Banach-Räume*, die aus Räumen von bis zur p -ten Potenz integrierbaren Funktionen gebildet werden. L^p -Räume werden über das *Lebesgue-Integral* definiert, wobei das $p \in \mathbb{R}$ ein reeller Parameter ist. Für alle $1 \leq p \leq \infty$ ist ein L^p -Raum definiert.

Definition 5.10 (L^p -Raum oder Lebesgue-Raum)

Es sei $1 \leq p < \infty$. Der Raum aller messbaren Funktionen $y : \Omega \rightarrow \mathbb{R}$ mit der Eigenschaft

$$\int_{\Omega} |y(x)|^p dx < \infty \tag{5.4}$$

wird als L^p -Raum oder *Lebesgue-Raum* bezeichnet und mit folgender Norm versehen:

$$\|y\|_{L^p(\Omega)} = \left(\int_{\Omega} |y(x)|^p dx \right)^{1/p}. \tag{5.5}$$

Funktionen, die sich nur auf einer Menge vom Maß Null unterscheiden, werden als gleich angesehen und gehören einer gemeinsamen Äquivalenzklasse an. □

Wir betrachten nun den für uns interessanten Fall $p = 2$.

Beispiel 5.11 (Raum der quadratintegrierbaren Funktionen L^2)

Es sei $p = 2$. Der Raum aller messbaren Funktionen $y : \Omega \rightarrow \mathbb{R}$ mit der Eigenschaft

$$\int_{\Omega} |y(x)|^2 dx < \infty \tag{5.6}$$

und der durch das Skalarprodukt

$$\langle f, g \rangle_{L^2} = \int_{\Omega} f(x) g(x) dx \tag{5.7}$$

induzierten Norm

$$\|y\|_{L^2(\Omega)} = \left(\int_{\Omega} |y(x)|^2 dx \right)^{1/2} \tag{5.8}$$

wird als *Raum der quadratintegrierbaren Funktionen L^2* bezeichnet. •

Bemerkung 5.12

Die Ungleichung (5.6) stellt die Quadratintegrierbarkeit und damit die Wohldefiniertheit der Norm sicher. ◦

Bemerkung 5.13

Der L^2 ist ein Hilbert-Raum, falls Ω ein Hilbert-Raum ist. ◦

Nachdem wir nun die L^p -Räume für endliches p definiert haben, betrachten wir noch vollständigkeithalber den Spezialfall für unendliches p :

Definition 5.14 (Der Raum L^∞)

Mit $L^\infty(\Omega)$ bezeichnet man den Raum der *fast überall gleichmäßig beschränkten und messbaren Funktionen* $y : \Omega \rightarrow \mathbb{R}$, versehen mit der Norm

$$\|y\|_{L^\infty(\Omega)} = \operatorname{ess\,sup}_{x \in \Omega} |y(x)| := \inf_{|F|=0} \left(\sup_{x \in \Omega \setminus F} |y(x)| \right). \tag{5.9}$$

□

Bemerkung 5.15

Mit *ess sup* wird das essentielle oder wirkliche Maximum bzw. Supremum einer Funktion bezeichnet und damit z.B. *unwesentliche* Maxima ausgeschlossen, d.h. solche die sich bereits durch Herausnehmen einzelner, in gewissem Sinne "isolierter" Punkte ändern (vergleiche [51]). \circ

Abschließend wollen wir noch Räume stetiger Funktionen definieren, deren Eigenschaften bei der Einführung der Sobolev-Räume ausgenutzt werden. Dazu die folgende Definition:

Definition 5.16 (Träger)

Sei $v : \Omega \rightarrow \mathbb{R}$ eine Funktion. Man bezeichnet die Menge

$$\text{supp}(v) = \overline{\{x \in \Omega : v(x) \neq 0\}} \quad (5.10)$$

als *Träger (support)* von v . Dabei handelt es sich um die kleinste abgeschlossene Menge, außerhalb der v identisch verschwindet. \square

Definition 5.17 (Die Räume $C_c^k(\Omega)$, $C(\overline{\Omega})$ und $C^k(\overline{\Omega})$)

Mit $C_c^k(\Omega)$ wird der Raum aller k -mal stetig differenzierbaren Funktionen $v : \Omega \rightarrow \mathbb{R}$ mit kompaktem Träger in Ω bezeichnet.

Der Raum aller auf $\overline{\Omega}$ stetigen reellwertigen Funktionen wird mit $C(\overline{\Omega})$ bezeichnet, der Raum aller auf $\overline{\Omega}$ k -mal stetig partiell differenzierbaren reellwertigen Funktionen mit $C^k(\overline{\Omega})$.

Versieht man letztere mit den folgenden Normen, so werden diese zu Banach-Räumen:

- $\|y\|_{C(\overline{\Omega})} = \max_{x \in \overline{\Omega}} |y(x)|,$
- $\|y\|_{C^k(\overline{\Omega})} = \max_{x \in \overline{\Omega}} \sum_{|\alpha| \leq k} |D^\alpha y(x)|.$ \square

Bemerkung 5.18

Betrachtet man für $C_c^k(\Omega)$ den Fall $k = \infty$, dann haben die Funktionen aus C_c^∞ auf dem Rand Γ den Wert Null, so dass man im Falle der partiellen Integration verschwindende Randintegrale erhält. Diese sogenannten *Testfunktionen* stammen aus dem Raum

$$C_c^\infty(\Omega) := \left\{ v \in C^\infty(\Omega) \mid \text{supp}(v) := \overline{\{x \in \Omega \mid v(x) \neq 0\}} \subset \Omega \right\} \quad (5.11)$$

der Menge aller beliebig oft stetig differenzierbaren Funktionen $v : \Omega \rightarrow \mathbb{R}$ mit kompaktem Träger in Ω . \circ

Aufgrund dieser Eigenschaft der Testfunktionen sind wir nun in der Lage, den klassischen Ableitungsbegriff zu verallgemeinern und erhalten somit die Definition der *schwachen Ableitung*.

Definition 5.19 (schwache Ableitung der Ordnung α)

Sei $p \in [1, +\infty]$, $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{N}_0^n$ ein Multiindex und die Funktion $y \in L^p(\Omega)$. Weiter sei $\varphi : \Omega \rightarrow \mathbb{R}$ eine *Testfunktion* aus $C_c^\infty(\Omega)$.

Dann heißt $w_\alpha \in L^p(\Omega)$ eine *schwache Ableitung der Ordnung α* von y , geschrieben

$$D^\alpha y = w_\alpha,$$

wenn gilt

$$\int_{\Omega} y D^\alpha \varphi \, dx = (-1)^{|\alpha|} \int_{\Omega} w_\alpha \varphi \, dx \quad \forall \varphi \in C_c^\infty(\Omega). \quad (5.12)$$

\square

Das heißt, falls zu einer gegebenen Funktion y eine Funktion w_α existiert, die (5.12) für alle φ erfüllt, sagt man, dass $D^\alpha y = w_\alpha$ in schwachem Sinne gilt.

Wir wollen dies an folgendem Beispiel veranschaulichen:

Beispiel 5.20

Sei $n = 1$, $\Omega = (0, 2)$ und $y(x)$ folgendermaßen gegeben:

$$y(x) = \begin{cases} x & \text{für } 0 < x \leq 1 \\ 1 & \text{für } 1 \leq x < 2 \end{cases}$$

Wir definieren

$$w(x) = \begin{cases} 1 & \text{für } 0 < x \leq 1 \\ 0 & \text{für } 1 < x < 2 \end{cases}$$

und zeigen, dass $y' = w$ im schwachen Sinne gilt, indem wir nachweisen, dass

$$\int_0^2 y\varphi' dx = - \int_0^2 w\varphi dx$$

gilt. Dazu wählen wir ein beliebiges $\varphi \in C_c^\infty(\Omega)$ und rechnen nach

$$\begin{aligned} \int_0^2 y\varphi' dx &= \int_0^1 x\varphi' dx + \int_1^2 \varphi' dx \\ &= - \int_0^1 \varphi dx + \varphi(1) - \varphi(1) = - \int_0^2 w\varphi dx, \end{aligned}$$

wie gefordert. Vergleiche hierzu [15], S. 243. •

Bemerkung 5.21

Eine schwache Ableitung ist in der Funktionalanalysis eine Erweiterung des Begriffs der gewöhnlichen Ableitung und stellt die Möglichkeit dar, den nicht (stark) differenzierbaren Funktionen eine Ableitung zuzuordnen. Man vergleiche hierzu [40]. ◦

Damit sind wir nun in der Lage, den für die Lösungstheorie partieller Differentialgleichungen grundlegenden Funktionenraum zu definieren, dessen Elemente schwache Ableitungen verschiedener Ordnung besitzen und in einem L^p -Raum liegen:

Definition 5.22 (Sobolev-Raum)

Es sei $p \in [1, +\infty]$, $m \in \mathbb{N}$ und jeweils $D^\alpha y$ die schwache Ableitung. Der Vektorraum

$$W^{m,p}(\Omega) := \{y \in L^p(\Omega) \mid \forall \alpha \text{ mit } 1 \leq |\alpha| \leq m \exists D^\alpha y \in L^p(\Omega)\}$$

versehen mit der Norm

$$\|y\|_{W^{m,p}} := \begin{cases} \left(\sum_{|\alpha| \leq m} \|D^\alpha y\|_{L^p(\Omega)}^p \right)^{1/p} & \text{für } 1 \leq p < +\infty \\ \max_{|\alpha| \leq m} \|D^\alpha y\|_{L^\infty(\Omega)} & \text{für } p = +\infty \end{cases}$$

ist ein Banach-Raum und wird als *Sobolev-Raum* bezeichnet.

Er besteht aus allen lokal summierbaren Funktionen $y : \Omega \rightarrow \mathbb{R}$, so dass für jeden Multiindex α mit $|\alpha| \leq m$ die Ableitung $D^\alpha y$ in schwachem Sinne existiert und zu $L^p(\Omega)$ gehört. Dabei steht der erste der beiden Indizes für die Ableitungen nach x und der zweite Index für die Integrationsordnung. □

Bemerkung 5.23

Für den Fall $p = 2$ wird die folgende gebräuchliche Schreibweise verwendet:

$$H^m(\Omega) := W^{m,2}(\Omega). \quad \circ$$

Damit erhalten wir den für uns interessanten Fall $m = 1$:

$$\begin{aligned} H^1(\Omega) &= W^{1,2}(\Omega) = \\ &= \{y \in L^2(\Omega) \mid \exists D_i y \in L^2(\Omega) \text{ schwach für } i = 1, \dots, n\}, \end{aligned} \quad (5.13)$$

versehen mit der Norm

$$\|y\|_{H^1(\Omega)} = \left(\int_{\Omega} (y^2 + |\nabla y|^2) dx \right)^{1/2} \quad (5.14)$$

mit $|\nabla y|^2 = \sum_{i=1}^N (D_i y)^2$.

Bemerkung 5.24

Betrachtet man die Vervollständigung von $C_c^\infty(\Omega)$ in $W^{m,p}(\Omega)$, bezeichnet mit $W_0^{m,p}(\Omega)$ und versehen mit derselben Norm wie $W^{m,p}(\Omega)$, so erhält man analog zu Bemerkung 5.23 für den Fall $p = 2$

$$H_0^m(\Omega) := W_0^{m,2}(\Omega).$$

Des Weiteren findet für $m = 1$ der Raum $H_0^1(\Omega)$, bestehend aus den H^1 -Funktionen mit kompaktem Träger in Ω , als Lösungsraum im Besonderen von Problemen mit Dirichlet-Randbedingung und verteilter Steuerung Verwendung. ◦

5.2 Sobolev-Räume unter Einbeziehung der Zeit: Der $W(0, T)$

Beschreiben elliptische Differentialgleichungen *stationäre*, also zeitunabhängige physikalische Vorgänge, so kommen bei *instationären* (oder zeitabhängigen) Vorgängen die parabolischen und hyperbolischen Differentialgleichungen zum Tragen.

Deshalb betrachten wir nun eine Variante der Sobolev-Räume, deren enthaltene Funktionen die *Zeitkomponente* in Banach-Räume abbilden, und sich damit als extrem nützlich in Bezug auf die Konstruktion schwacher Lösungen für parabolische und hyperbolische partielle Differentialgleichungen erweisen (vgl. [15]).

Die dafür notwendigen Funktionen werden dazu auf einem *Ort-Zeit-Zylinder* $Q := \Omega \times (0, T)$ definiert und beschreiben auf dem Ortsgebiet die Dynamik der betrachteten Größen vom Anfangszeitpunkt $t_0 = 0$ bis zum Endzeitpunkt $T > 0$. Im Folgenden habe Ω einen $C^{1,1}$ -Rand Γ , der sich dahingehend vom $C^{0,1}$ -Rand unterscheidet, indem sozusagen die "Forderung der stetigen Differenzierbarkeit" am Rand hinzukommt (genauer gesagt: an die den Rand parametrisierenden Funktionen, vgl [51]). Dabei stellt der

$C^{1,1}$ -Rand hinreichende Glattheitsforderungen, die im \mathbb{R}^n für $n \geq 2$ zu Problemen führen können. Da wir uns in den Auswertungen aber auf den eindimensionalen Fall beschränken und damit beliebige Glattheitsbedingungen an den Rand stellen können (sozusagen der $C^{1,1}$ -Rand mit dem $C^{0,1}$ -Rand “zusammenfällt”), wird hier auf diese Problematik nicht weiter eingegangen.

Wir wollen nun die bereits kennengelernten Konzepte und Funktionenräume bezüglich des Ort-Zeit-Zylinders betrachten. So erhält man für den Raum der quadratintegrierbaren Funktionen auf $Q := \Omega \times (0, T)$:

$$L^2(Q) := \left\{ y : Q \rightarrow \mathbb{R} \mid \|y\|_{L^2(Q)}^2 < \infty \right\}$$

mit der zugehörigen Norm

$$\|y\|_{L^2(Q)}^2 := \left(\int_0^T \left(\int_{\Omega} y^2(x, t) dx \right) dt \right)^{1/2}.$$

Des Weiteren lässt sich auch der Begriff der schwachen Ableitung in Ω auf Q analog übertragen:

Definition 5.25 (schwache Ableitung der Ordnung α in Q)

Sei $p \in [1, +\infty]$, $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{N}_0^n$ ein Multiindex und die Funktion $y \in L^p(Q)$. Weiter sei $\varphi : Q \rightarrow \mathbb{R}$ eine Testfunktion aus $C_c^\infty(Q)$.

Dann heißt $w_\alpha \in L^p(Q)$ eine schwache Ableitung der Ordnung α von y in Q , geschrieben

$$D^\alpha y = w_\alpha,$$

wenn gilt

$$\iint_Q y(x, t) D^\alpha \varphi(x, t) dx dt = (-1)^{|\alpha|} \iint_Q w_\alpha(x, t) \varphi(x, t) dx dt \quad (5.15)$$

für alle $\varphi(x, t) \in C_c^\infty(Q)$. □

Damit erhalten wir für Q ein Analogon zum Raum $H^1(\Omega)$:

Definition 5.26

Wir bezeichnen mit $W_2^{1,0}(Q)$ den Raum aller (Äquivalenzklassen von) Funktionen $y \in L^2(Q)$, die alle schwachen Ableitungen erster Ordnung nach x_1, \dots, x_n im Raum $L^2(Q)$ besitzen, versehen mit der Norm

$$\|y\|_{W_2^{1,0}(Q)} = \left(\int_0^T \int_{\Omega} (y(x, t)^2 + |\nabla y(x, t)|^2) dx dt \right)^{1/2}$$

mit $\nabla := \nabla_x$ Gradient bezüglich x , und schreiben dafür kurz

$$W_2^{1,0}(Q) = \{ y \in L^2(Q) \mid \exists D_i y \in L^2(Q) \text{ schwach für } i = 1, \dots, n \}.$$

Dabei stehen die oberen beiden Indizes für die Ableitungen nach x und t und der untere Index für die Integrationsordnung. □

Bemerkung 5.27

In der Fachliteratur wird der Raum $W_2^{1,0}(Q)$ auch mit $H^{1,0}(Q)$ bezeichnet. ○

Die im Raum $W_2^{1,0}(Q)$ enthaltenen Funktionen werden auch als *abstrakte Funktionen* bezeichnet und dienen als grundlegendes Hilfsmittel zur Beschreibung von zeitabhängigen Gleichungen, den sogenannten *Evolutionsgleichungen* (vgl. [51]).

Definition 5.28 (Abstrakte Funktion)

Sei $[a, b] \subset \mathbb{R}$ ein kompaktes Intervall. Eine Abbildung von $[a, b]$ in einen Banach-Raum X wird als *abstrakte Funktion* bezeichnet. □

Je nach Wahl des Raumes X ergeben sich unterschiedliche Fälle, von denen wir im Folgenden den für uns interessantesten betrachten wollen.

Beispiel 5.29

Wir wählen für $X = H^1(\Omega)$.

Dann erhält man als Funktionswert $y(t)$ einer abstrakten Funktion $y : [a, b] \rightarrow H^1(\Omega)$ für jedes $t \in [a, b]$ ein Element aus $H^1(\Omega)$, d.h. jedem Zeitpunkt t wird eine Funktion $y(\cdot, t) \in H^1(\Omega)$ zugeordnet. Damit gilt für jedes feste t :

$$y(t) = y(\cdot, t) \quad \text{mit} \quad y(\cdot, t) = y(x, t), \quad x \in \Omega.$$

Vergleiche hierzu [51], Abschnitt 3.4.1, Beispiel (iii). •

Als nächstes wollen wir L^p -Räume abstrakter Funktionen einführen. Dazu benötigen wir vorher noch die Begriffe der Stetigkeit und der Messbarkeit abstrakter Funktionen.

Definition 5.30

Eine abstrakte Funktion heißt stetig im Punkt t , wenn aus $\tau \rightarrow t$ folgt $y(\tau) \rightarrow y(t)$ in X , d.h.

$$\lim_{\tau \rightarrow t} \|y(\tau) - y(t)\|_X = 0.$$

Der Raum aller stetigen Funktionen $y : [a, b] \rightarrow X$, versehen mit der Norm

$$\|y\|_{C([a,b],X)} = \max_{t \in [a,b]} \|y(t)\|_X,$$

wird mit $C([a, b], X)$ bezeichnet. □

Definition 5.31

Eine abstrakte Funktion $y : [a, b] \rightarrow X$ heißt *messbar*, wenn eine Folge $\{y_k\}_{k=1}^\infty$ von Treppenfunktionen $y_k : [a, b] \rightarrow X$ existiert, so dass gilt:

$$y(t) = \lim_{k \rightarrow \infty} y_k(t) \quad \text{fast überall auf dem Intervall } [a, b]. \quad \square$$

Eine abstrakte Funktion y bezeichnet man dabei dann als *Treppenfunktion*, wenn $[a, b]$ die Vereinigung endlich vieler Lebesgue-messbarer und paarweise disjunkter Mengen $M_i \subset [a, b]$ ist und endlich viele Elemente $y_i \in X$ existieren, so dass $y(t) = y_i \quad \forall t \in M_i$ gilt.

Definition 5.32 (Der L^p -Raum abstrakter Funktionen)

Sei $1 \leq p < \infty$. Der Raum aller (Äquivalenzklassen von) messbaren abstrakten Funktionen $y : [a, b] \rightarrow X$ mit der Eigenschaft

$$\int_a^b \|y(t)\|_X^p dt < \infty$$

versehen mit der Norm

$$\|y\|_{L^p(a,b;X)} := \left(\int_a^b \|y(t)\|_X^p dt \right)^{1/p}$$

wird bezeichnet mit $L^p(a, b; X)$. □

Betrachtet man den Raum $L^2(0, T; H^1(\Omega))$, dann kann man die enthaltenen abstrakten Funktionen auch gleichzeitig als reellwertige Funktionen der Variablen $x \in \Omega$ und $t \in [0, T]$ auffassen. Für jedes t ist y in Bezug auf x eine Funktion aus $H^1(\Omega)$. Betrachtet man die Normen in $W_2^{1,0}(Q)$ und $L^2(0, T; H^1(\Omega))$, so stellt man fest, dass diese gleich sind (vgl. [51], S. 115). Des Weiteren erhält man (indem man zeigt¹, dass jede Funktion $y \in W_2^{1,0}(Q)$ durch Abänderung auf einer Menge vom Maß Null zu einer Funktion aus $L^2(0, T; H^1(\Omega))$ wird und umgekehrt) sogar die Isomorphie

$$W_2^{1,0}(Q) \cong L^2(0, T; H^1(\Omega))$$

der beiden Räume.

Somit erhalten wir nun die folgende Definition.

Definition 5.33 (Der Raum $W(0, T)$)

Unter $W(0, T; H^1(\Omega))$ oder kurz $W(0, T)$ versteht man den Raum aller Funktionen $y \in L^2(0, T; H^1(\Omega))$ mit (distributioneller) Ableitung $y' \in L^2(0, T; H^1(\Omega)^*)$, versehen mit der Norm

$$\|y\|_{W(0,T)} := \left(\int_0^T \left(\|y(t)\|_{H^1(\Omega)}^2 + \|y'(t)\|_{H^1(\Omega)^*}^2 \right) dt \right)^{1/2},$$

in Kurzschreibweise:

$$W(0, T) := \{y \in L^2(0, T; H^1(\Omega)) \mid \exists y' \in L^2(0, T; H^1(\Omega)^*)\}$$

Dabei bezeichnet $H^1(\Omega)^*$ den *Dualraum* von $H^1(\Omega)$, für dessen Definition wir auf [46] verweisen. □

Bemerkung 5.34

Bei der Anwendung von abstrakten Funktionen auf eine parabolische Modellgleichung erhält man durch den Vergleich der berechneten Funktionale und durch die Tatsache, dass jede schwache Lösung y aus $W_2^{1,0}(Q)$ nach Abänderung auf einer Menge vom Maß Null als abstrakte Funktion in $L^2(0, T; H^1(\Omega))$ aufgefasst werden kann, dass die Ableitung y_t aus $L^2(0, T; H^1(\Omega)^*)$ ist.

¹Eine Beweiskonstruktion hierzu findet sich in E. HILLE und R. S. PHILLIPS, *Functional Analysis and Semigroups*. Amer. Math. Soc. Colloq. Publ., Providence, 1957.

Dies wurde für den allgemeinen Fall eines Raumes V untersucht, indem man die Ableitung im Sinne einer *vektorwertigen Distribution* \mathcal{T} definiert, dessen Ableitung ebenfalls als vektorwertige Distribution durch das *Bochner Integral* eingeführt wird, und damit entsprechende Funktionen identifiziert (siehe hierzu [51], 3.4.3). Letztendlich erhält man unter der Voraussetzung $w \in L^2(0, T; V)$ für die die Ableitung y' definierende abstrakte Funktion w sogar, dass man die Menge

$$\{y \in L^2(0, T; V) \mid y' \in L^2(0, T; V)\}$$

von Funktionen $y \in L^2(0, T; V)$ mit einer regulären Ableitung einführen kann. Da diese Konstruktion aber zu “eng” ist, wie die Anwendung auf die parabolische Modellgleichung gezeigt hat, ergibt sich, dass $y' = y_t$ ein Element aus dem größeren Raum $L^2(0, T; V^*)$ ist. Setzt man nun für $V = H^1(\Omega)$, so erhalten wir obige Definition.

Für eine ausführliche Herleitung dieser Aussagen verweisen wir auf das Buch von F. TRÖLTZSCH [51], Kapitel 3.4.1 bis 3.4.3. ◦

In der abschließenden Bemerkung wollen wir noch den Bezug zu der von uns verwendeten Kostenfunktion herstellen.

Bemerkung 5.35

Die Lösung unserer partiellen Differentialgleichung $y \in W(0, T)$ entspricht an jedem Abtastzeitpunkt $t = nT$ einer Funktion aus $H^1(\Omega)$. Des Weiteren entspricht $y(t, \cdot)$ einem “kontinuierlichen” Ortsprofil auf Ω , d.h. für jedes $x \in \Omega$ ist ein Funktionswert definiert, und stellt somit einen Kostenbestandteil in der Kostenfunktion unseres Optimalsteuerungsproblems dar. In diesem Zusammenhang liefert der $W(0, T)$ die Wohldefiniertheit der Norm $\|y(\cdot, t)\|_{H^1(\Omega)}$ und erlaubt dadurch eine Bewertung dieser Kosten.

Durch die für die Lösung mittels des MPC-Programms notwendige räumliche Diskretisierung der unserer Auswertung zugrunde liegenden evolutionären Systeme erhalten wir ein System gewöhnlicher Differentialgleichungen. Somit liegt nun nicht mehr ein kontinuierliches Ortsprofil zu den Abtastzeitpunkten t vor, sondern lediglich eine diskrete Variante dieses Profils. Für die Modellierung unserer Zielfunktion bedeutet dies, dass wir die H^1 -Norm in additiver Form zweier L^2 -Normen erhalten:

$$\|y(x, t)\|_{H^1(\Omega)} = \|y(x, t)\|_{L^2(\Omega)} + \|y_x(x, t)\|_{L^2(\Omega)} \tag{5.16}$$

Letztendlich erhält man damit, dass der Teil $\|y'(t)\|_{H^1(\Omega)^*}^2$ der Norm des $W(0, T)$ und damit der Dualraum für die von uns durchgeführten Auswertungen keine weitere Rolle spielt und dahingehend nicht weiter betrachtet wird. ◦

6 Vorstellung der betrachteten Systeme und ihre Diskretisierung

In diesem Kapitel wollen wir die in dieser Arbeit ausgewerteten Systemen vorstellen und deren Diskretisierungen angeben. Dabei handelt es sich zum einen um

- die parabolische **Reaktions-Advektions-Diffusions-Gleichung** und zum anderen um
- die hyperbolische **Wellengleichung**.

Beide Systeme werden auf dem eindimensionalen offenen Gebiet $\Omega = (0, 1)$ mit Rändern $\Gamma = \{0, 1\}$ betrachtet. Dazu zerlegen wir dieses Gebiet in ein äquidistantes Gitter mit Gitterpunkten x_i entsprechend Abschnitt 4.1.1:

$$x_i = (i - 1)h$$

mit $i = 1, \dots, M$ für $M \in \mathbb{N}$ der Anzahl der Diskretisierungspunkte und der Schrittweite $h = \Delta x := \frac{1}{M-1}$.

Wir schreiben kurz:

$$x_i = \frac{i - 1}{M - 1}, \quad i = 1, \dots, M.$$

Auf diesem Gitter werden die Systeme mittels der in Abschnitt 4.2.2 eingeführten *Finite-Differenzen-Methode* approximiert.

Wir betrachten die Reaktions-Advektions-Diffusions-Gleichung und die Wellengleichung unter bestimmten Anfangs- und Randbedingungen (vgl. Abschnitt 2.5):

$$\text{Dirichlet-Randbedingungen:} \quad y(0, t) = \sigma_1 \quad (6.1)$$

$$y(1, t) = \sigma_2 \quad (6.2)$$

$$\text{Neumann-Randbedingungen} \quad y_\nu(0, t) = \sigma_3 \quad (6.3)$$

$$y_\nu(1, t) = \sigma_4 \quad (6.4)$$

$$\text{mit vorgegebenen} \quad \sigma_1, \sigma_2, \sigma_3, \sigma_4 \in \mathbb{R}$$

Dabei können $\sigma_1, \sigma_2, \sigma_3$ und σ_4 entsprechend der Problemstellung gewählt werden.

$$\text{Anfangsbedingung} \quad y_0(x) := y(x, 0) = x_{\text{Startprofil}} \quad (6.5)$$

$$\text{mit} \quad x_{\text{Startprofil}} \in \mathbb{R}^M$$

Dabei entspricht $x_{\text{Startprofil}} := (x_1, \dots, x_M)$ mit $x_i \in \mathbb{R}$ dem durch das Startprofil vorgegebenen Wert am Diskretisierungspunkt i .

Bei der Betrachtung der Systeme unter Verwendung verschiedener Steuerungen (vgl. Abschnitt 2.6) und Randbedingungen (vgl. Abschnitt 2.5.2) müssen folgende vier Fälle bezüglich der Diskretisierung unterschieden werden:

- Dirichlet-Randbedingung mit verteilter Steuerung
- Dirichlet-Randsteuerung
- Neumann-Randbedingung mit verteilter Steuerung:
 - bei Reaktions-Advektions-Diffusions-Gleichung: nur einseitig
 - bei Wellengleichung: einseitig und zweiseitig
- Neumann-Randsteuerung

Dazu verwenden wir im Weiteren stets die folgenden Approximationen:

Für die inneren Punkte $x_i \in \Omega = (0, 1), i = 2, \dots, M-1$ berechnen wir die zweite Ableitung y_{xx} mit dem in (4.28) definierten *Zentral-Differenzenquotienten* zweiter Ordnung:

$$y_{xx}(x_i, t) = \frac{y(x_{i+1}, t) - 2y(x_i, t) + y(x_{i-1}, t)}{h^2}$$

Für die Reaktions-Advektions-Diffusions-Gleichung benötigen wir zusätzlich die Approximation der ersten Ableitung y_x und verwenden hierfür die *Upwind-Methode*. Wir erhalten somit, aufgrund des negativen Vorfaktors der erster Ableitung (siehe Abschnitt 4.2.3), den in (4.23) definierten *Vorwärts-Differenzenquotienten* erster Ordnung (vgl. hierzu Bemerkung 4.18):

$$y_x(x_i, t) = \frac{y(x_{i+1}, t) - y(x_i, t)}{h}$$

Auf dem Rand Γ müssen die Diskretisierungen je nach verwendeter Randbedingung gesondert betrachtet werden.

Wir verwenden die einfachere Schreibweise $y_i(t)$ an Stelle von $y(x_i, t)$. Bevor wir uns in den nächsten Abschnitten den Diskretisierungen der Systeme widmen, wollen wir zunächst die beiden betrachteten PDEs in den folgenden Abschnitten motivieren.

6.1 Die Reaktions-Advektions-Diffusions-Gleichung

Die *Reaktions-Advektions-Diffusions-Gleichung* (kurz: ReAdDiff) lässt sich zum Beispiel durch die Kombination zweier partieller Differentialgleichungen definieren, die unter anderem vornehmlich in Bereichen der Physik, der Chemie und der Biochemie zur Beschreibung und Modellierung von Vorgängen und Prozessen Verwendung finden. Es handelt sich hier um

- die **Reaktions-Diffusions-Gleichung** und
- die **Advektions-Diffusions-Gleichung**.

Eine Reaktions-Diffusions-Gleichung beschreibt das sogenannte *Erhaltungsprinzip* einer physikalischen Größe:

$$y_t(x, t) = \Delta y(x, t) + f(x, t, y(t), \nabla y(x, t)).$$

In Worten ausgedrückt: *Reaktion = Diffusion + Quellterm*. Dabei spielt es keine Rolle, ob $y(x, t)$ der Temperatur eines Körpers, einer Bevölkerungsdichte oder Ähnlichem entspricht. Vielmehr muß die Größe $y(x, t)$ demselben Gesetz genügen: Pro Zeiteinheit setzt sich die Produktionsrate zum einen zusammen aus den Teilchen, die in den Ort x diffundieren, zum anderen aus den Teilchen, die von einer Quelle erzeugt werden. f entspricht dabei der Quellergiebigkeit, die vor allem bei chemischen Reaktionen in der Regel neben Ort x und Zeit t auch von der Temperatur und der Konvektion abhängt. Eine exakte Formulierung von f ist dabei vom betrachteten Modell oder System abhängig und bietet somit eine breite Palette an Modellierungsmöglichkeiten (vergleiche [4]).

Die Advektions-Diffusions-Gleichung stellt eine allgemeine Schreibweise für *Strömungs-* oder *Transportgleichungen* dar:

$$y_t(x, t) + c y_x(x, t) = d \Delta y(x, t)$$

mit *Transportgeschwindigkeit* c und *Diffusionskonstante* d (vgl. [33]). Sowohl die Advektion (von lat.: Zufuhr, Verschiebung) als auch die Diffusion (von lat. Ausbreitung, Verstreuung) entsprechen dabei sogenannten Transportmechanismen oder Transportarten. Während die Advektion für eine gerichtete Verschiebung z.B. eines Stoffes in einem Medium zuständig ist, bewirkt die Diffusion eine ungerichtete Ausbreitung oder Bewegung des Stoffes. Damit lässt sich dann u. a. der Transport einer Fluideigenschaft aufgrund der Eigenbewegung eines Fluids beschreiben (z.B. bei Abwassereinleitung in einen See oder Fluss). Für eine ausführliche und anschauliche Beschreibung sowie weitere Erklärungen sei hierzu auf [1] verwiesen.

Für unsere Auswertungen verwenden wir die folgende Reaktions-Advektions-Diffusions-Gleichung

$$y_t = -y_x + \nu \Delta y + \mu y(y + 1)(1 - y) \tag{6.6}$$

mit $\nu, \mu, x \in \mathbb{R}, t \in (0, t_f)$ und einem nichtlinearem Term $y(y + 1)(1 - y)$. Wir setzen für $\nu = c$ und für $d = 1$.

6.2 Die Wellengleichung

Bei der *Wellengleichung* handelt es sich um eine lineare partielle Differentialgleichung zweiter Ordnung vom hyperbolischen Typ, deren Lösungen die Ausbreitung von *Wellen* modellieren.

Die Wellen selbst benötigen einen *Träger* (z.B. im eindimensionalen Fall eine Saite oder einen Stab, im zweidimensionalen eine geeignete (Ober-)Fläche, im dreidimensionalen ein ausgedehntes Medium wie Luft oder Wasser), dessen Bausteine eine Auslenkung aus ihrer Ruhelage erfahren. Während der Träger selbst an seinem Ort verbleibt, wandert das durch die Auslenkung erzeugte *Auslenkungsmuster* durch den Raum, welches wir als *Welle* identifizieren (siehe Abbildung 6.1). Das bedeutet, dass in Wellen kein Materietransport, sondern ein Energietransport stattfindet (vergleiche [7]).

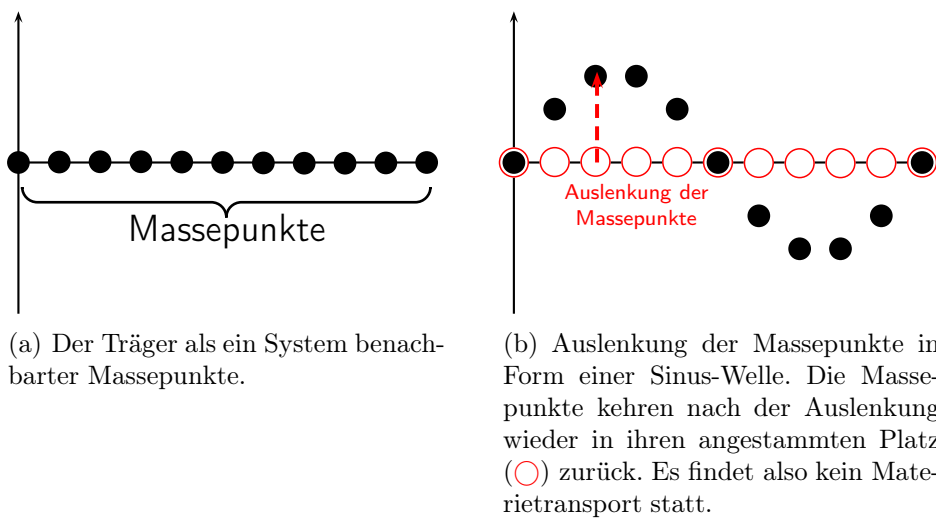


Abbildung 6.1: Auslenkung der Massepunkte

Stellt man sich den Träger als ein durch elastische Kräfte gekoppeltes System von benachbarten Raumelementen vor, so erhält man eine Wellengleichung, die im einfachsten homogenen Fall wie folgt definiert ist:

$$y_{tt} - c^2 \Delta y = 0 \quad \text{mit } c > 0, \quad x \in \mathbb{R}, \quad t \in (0, t_f), \quad (6.7)$$

mit der *Wellenausbreitungsgeschwindigkeit* $c > 0$ und dem Laplace-Operator Δ . Diese beschreibt die Auslenkung des Trägers y zur Zeit t am Ort x . Eine ausführliche Herleitung einer Wellengleichung für die Schallausbreitung sowie weitere Theorie zur Wellengleichung findet sich in [31].

Im anschließenden Beispiel wollen wir die Lösung einer eindimensionalen Wellengleichung genauer betrachten und anhand einer Grafik veranschaulichen.

Beispiel 6.1 (Lösung der Wellengleichung in einer Dimension)

Die homogene Wellengleichung in einer Dimension

$$c^2 \frac{\partial^2 y(x, t)}{\partial x^2} = \frac{\partial^2 y(x, t)}{\partial t^2}$$

besitzt eine allgemeine Lösung

$$y(x, t) = f(x + ct) + g(x - ct), \quad (6.8)$$

wobei $f(x)$ und $g(x)$ beliebige, zweimal differenzierbare Funktionen sind. Die Funktion f beschreibt dabei eine mit Geschwindigkeit c nach links, die Funktion g eine mit Geschwindigkeit c nach rechts laufende Welle. Diese lassen sich des Weiteren z.B. als Linearkombination von Sinus- oder auch Exponential-Funktionen schreiben, so dass man eine Lösung der Wellengleichung der folgenden Form erhält:

$$y(x, t) = A \sin(kx \pm \omega t + \phi)$$

bzw.

$$y(x, t) = A e^{-(kx + \omega t)^2} + e^{-(kx - \omega t)^2}$$

für eine Konstante $A \in \mathbb{R}$ und der zugehörigen *Dispersionsrelation* $\omega = k \cdot c$ mit *Kreiswellenzahl* k . Zur graphischen Veranschaulichung wählen wir das einfache Beispiel der Überlagerung zweier Wellen mit $k = 1$, so dass $\omega = c$ folgt und man damit

$$y(x, t) = e^{-(x+ct)^2} + e^{-(x-ct)^2}$$

erhält. Für eine Wellengeschwindigkeit von $c = 0.6$ ergibt sich daraus die mit MATLAB erstellte Abbildung 6.2 (vgl. hierzu [8]).

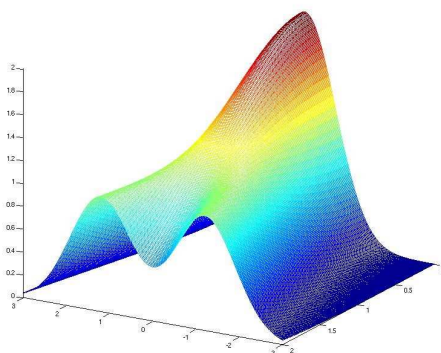


Abbildung 6.2: Eindimensionale Wellengleichung

Wir betrachten in unseren Auswertungen die eindimensionale Wellengleichung mit Wellengeschwindigkeit $c = 1$:

$$y_{tt} = y_{xx} \quad \text{mit } x \in \mathbb{R}, t \in (0, t_f). \quad (6.9)$$

6.3 Diskretisierung der Systeme

In diesem Abschnitt wollen wir nun die vorgestellten Systeme mittels der in Kapitel 4 behandelten Approximationen bezüglich der zu Beginn erwähnten Fälle diskretisieren.

6.3.1 Dirichlet-Randbedingung bei verteilter Steuerung

Wir verwenden die Dirichlet-Randbedingung (6.1) und (6.2) bei verteilter Steuerung $u(\cdot, t) := u_i(t) \in \Omega$ mit $i = 1, \dots, M$. Aufgrund dieser Randbedingung erhalten wir mittels Diskretisierung ein System von gewöhnlichen Differentialgleichungen nur an den inneren Punkten x_2 bis x_{M-1} . Dabei wird die jeweilige Steuerung jeweils additiv an die entsprechende Differentialgleichung angehängt:

Reaktions-Advektions-Diffusions-Gleichung:

$i = 2 :$

$$y_t(x_2, t) = -\frac{y_3(t) - y_2(t)}{h} + \nu \frac{y_3(t) - 2y_2(t) + \sigma_1}{h^2} + \mu(y_2(t)(y_2(t) + 1)(1 - y_2(t))) + u_1(t)$$

$i = 3, \dots, M - 2 :$

$$y_t(x_i, t) = -\frac{y_{i+1}(t) - y_i(t)}{h} + \nu \frac{y_{i+1}(t) - 2y_i(t) + y_{i-1}(t)}{h^2} + \mu(y_i(t)(y_i(t) + 1)(1 - y_i(t))) + u_{i-1}(t) \quad (6.10)$$

$i = M - 1 :$

$$y_t(x_{M-1}, t) = -\frac{\sigma_2 - y_{M-1}(t)}{h} + \nu \frac{\sigma_2 - 2y_{M-1}(t) + y_{M-2}(t)}{h^2} + \mu(y_{M-1}(t)(y_{M-1}(t) + 1)(1 - y_{M-1}(t))) + u_{M-2}(t)$$

Wellengleichung:

$i = 2 :$

$$y_{tt}(x_2, t) = \frac{\sigma_1 - 2y_2(t) + y_3(t)}{h^2} + u_1(t)$$

$i = 3, \dots, M - 2 :$

$$y_{tt}(x_i, t) = \frac{y_{i-1}(t) - 2y_i(t) + y_{i+1}(t)}{h^2} + u_{i-1}(t) \quad (6.11)$$

$i = M - 1 :$

$$y_{tt}(x_{M-1}, t) = \frac{y_{M-2}(t) - 2y_{M-1}(t) + \sigma_2}{h^2} + u_{M-2}(t)$$

6.3.2 Dirichlet-Randsteuerung

Bei einer Dirichlet-Randbedingung mit Randsteuerung setzen wir im Gegensatz zur verteilten Steuerung nun die Bedingung an den Rändern $x_1 = 0$ und $x_M = 1$ entsprechend

der dort anliegenden Steuerung:

$$y(0, t) := u_1(t) \quad \text{und} \quad y(1, t) := u_2(t). \quad (6.12)$$

Da bei Randsteuerung im Inneren nicht mehr gesteuert wird, fließen die Steuerungen nun **direkt** in die Approximationen ein und wir erhalten mit Randsteuerung ein System von gewöhnlichen Differentialgleichungen an den Stellen x_2 bis x_{M-1} :

Reaktions-Advektions-Diffusions-Gleichung:

$i = 2 :$

$$y_t(x_2, t) = -\frac{y_3(t) - y_2(t)}{h} + \nu \frac{y_3(t) - 2y_2(t) + u_1(t)}{h^2} + \mu(y_2(t)(y_2(t) + 1)(1 - y_2(t)))$$

$i = 3, \dots, M - 2 :$

$$y_t(x_i, t) = -\frac{y_{i+1}(t) - y_i(t)}{h} + \nu \frac{y_{i+1}(t) - 2y_i(t) + y_{i-1}(t)}{h^2} + \mu(y_i(t)(y_i(t) + 1)(1 - y_i(t))) \quad (6.13)$$

$i = M - 1 :$

$$y_t(x_{M-1}, t) = -\frac{u_2(t) - y_{M-1}(t)}{h} + \nu \frac{u_2(t) - 2y_{M-1}(t) + y_{M-2}(t)}{h^2} + \mu(y_{M-1}(t)(y_{M-1}(t) + 1)(1 - y_{M-1}(t)))$$

Wellengleichung:

$i = 2 :$

$$y_{tt}(x_2, t) = \frac{u_1(t) - 2y_2(t) + y_3(t)}{h^2}$$

$i = 3, \dots, M - 2 :$

$$y_{tt}(x_i, t) = \frac{y_{i-1}(t) - 2y_i(t) + y_{i+1}(t)}{h^2} \quad (6.14)$$

$i = M - 1 :$

$$y_{tt}(x_{M-1}, t) = \frac{y_{M-2}(t) - 2y_{M-1}(t) + u_2(t)}{h^2}$$

6.3.3 Neumann-Randbedingung bei verteilter Steuerung

Bei der einseitigen Neumann-Randbedingung (6.3) wird nun eine Bedingung an die Normalenableitung von y (siehe 4.2.2.3) am Randpunkt $x_1 = 0$ gestellt. Am rechten Randpunkt x_M verwenden wir die Dirichlet-Randbedingung (6.2).

Dadurch erhalten wir jetzt eine zusätzliche Differentialgleichung am linken Rand, zu deren Approximation mittels des Zentrale-Differenzenquotienten Punkte außerhalb des Intervalls $[0, 1]$ benötigt werden. Dieses Problem wird durch das Einsetzen der Neumann-Randbedingung in die Differentialgleichung am Rand behoben. Dazu approximieren wir die Normalenableitung mittels des Zentrale-Differenzenquotienten (4.27):

$$y_\nu(x_1, t) = \sigma_3 = \frac{1}{2h}(y_0(t) - y_2(t)).$$

Diese Gleichung wird nach $y_0(t)$ aufgelöst:

$$y_0(t) = 2h\sigma_3 + y_2(t). \quad (6.15)$$

Damit haben wir die Möglichkeit, die Differenzenapproximationen für y_{xx} aufzustellen und erhalten ein System von gewöhnlichen Differentialgleichungen, diesmal an den Stellen x_1 bis x_{M-1} , wobei die jeweilige Steuerung wie bei Dirichlet-Randbedingung jeweils additiv an die entsprechende Differentialgleichungen in Ω gehängt wird:

Reaktions-Advektions-Diffusions-Gleichung bei einseitiger Neumann-Randbedingung:

$i = 1 :$

$$y_t(x_1, t) = -\frac{y_2(t) - y_1(t)}{h} + \nu \frac{2y_2(t) - 2y_1(t) + 2h\sigma_3}{h^2} + \mu(y_1(t)(y_1(t) + 1)(1 - y_1(t)))$$

$i = 2, \dots, M - 2 :$

$$y_t(x_i, t) = -\frac{y_{i+1}(t) - y_i(t)}{h} + \nu \frac{y_{i+1}(t) - 2y_i(t) + y_{i-1}(t)}{h^2} + \mu(y_i(t)(y_i(t) + 1)(1 - y_i(t))) + u_{i-1}(t) \quad (6.16)$$

$i = M - 1 :$

$$y_t(x_{M-1}, t) = -\frac{\sigma_2 - y_{M-1}(t)}{h} + \nu \frac{\sigma_2 - 2y_{M-1}(t) + y_{M-2}(t)}{h^2} + \mu(y_{M-1}(t)(y_{M-1}(t) + 1)(1 - y_{M-1}(t))) + u_{M-2}(t)$$

Wellengleichung bei einseitiger Neumann-Randbedingung:

$$\begin{aligned}
 i = 1 : \\
 y_{tt}(x_1, t) &= \frac{2h\sigma_3 + 2y_2(t) - 2y_1(t)}{h^2} \\
 i = 2, \dots, M - 2 : \\
 y_{tt}(x_i, t) &= \frac{y_{i-1}(t) - 2y_i(t) + y_{i+1}(t)}{h^2} + u_{i-1}(t) \\
 i = M - 1 : \\
 y_{tt}(x_{M-1}, t) &= \frac{y_{M-2}(t) - 2y_{M-1}(t) + \sigma_2}{h^2} + u_{M-2}(t)
 \end{aligned} \tag{6.17}$$

Bei der zweiseitigen Neumann-Randbedingung (6.3) und (6.4) wird nun zusätzlich eine weitere Bedingung an die Normalenableitung von y (siehe 4.2.2.3) am rechten Randpunkt $x_M = 1$ gestellt.

Dadurch erhalten wir jetzt zwei zusätzliche Differentialgleichungen an den jeweiligen Rändern, zu deren Approximationen mittels des Zentrale-Differenzenquotienten wiederum Punkte außerhalb des Intervalls $[0, 1]$ benötigt werden.

Dazu approximieren wir die Normalenableitung analog für den rechten Rand mit Hilfe des Zentrale-Differenzenquotienten (4.27):

$$y_\nu(x_M, t) = \sigma_4 = \frac{1}{2h}(y_{M+1}(t) - y_{M-1}(t)).$$

Diese Gleichung wird nach $y_{M+1}(t)$ aufgelöst:

$$y_{M+1}(t) = 2h\sigma_4 + y_{M-1}(t). \tag{6.18}$$

Wir erhalten somit ein System von gewöhnlichen Differentialgleichungen an den Stellen x_1 bis x_M :

Wellengleichung bei zweiseitiger Neumann-Randbedingung:

$$\begin{aligned}
 i = 1 : \\
 y_{tt}(x_1, t) &= \frac{2h\sigma_3 + 2y_2(t) - 2y_1(t)}{h^2} \\
 i = 2, \dots, M - 1 : \\
 y_{tt}(x_i, t) &= \frac{y_{i-1}(t) - 2y_i(t) + y_{i+1}(t)}{h^2} + u_{i-1}(t) \\
 i = M : \\
 y_{tt}(x_M, t) &= \frac{2y_{M-1}(t) - 2y_M(t) + 2h\sigma_4}{h^2}
 \end{aligned} \tag{6.19}$$

6.3.4 Neumann-Randsteuerung

Wir betrachten jetzt die Randsteuerung mit einem Neumann-Rand. Im Gegensatz zur verteilten Steuerung wird die Normalenableitung an den Rändern $x_1 = 0$ und $x_M = 1$ entsprechend der dort anliegenden Steuerung gesetzt:

$$y_\nu(x_1, t) = u_1(t) \quad \text{und} \quad y_\nu(x_M, t) = u_2(t). \tag{6.20}$$

Somit erhalten wir mit dem Zentrale-Differenzenquotienten (4.27):

$$y_\nu(x_1, t) = u_1(t) = \frac{1}{2h}(y_0(t) - y_2(t))$$

und

$$y_\nu(x_M, t) = u_2(t) = \frac{1}{2h}(y_{M+1}(t) - y_{M-1}(t)).$$

Diese Gleichung wird nach $y_0(t)$ bzw. $y_{M+1}(t)$ aufgelöst:

$$y_0(t) = 2h \cdot u_1(t) + y_2(t) \tag{6.21}$$

bzw.

$$y_{M+1}(t) = 2h \cdot u_2(t) + y_{M-1}(t). \tag{6.22}$$

Damit können wir die Differenzenapproximationen für y_{xx} aufstellen und erhalten ein System von gewöhnlichen Differentialgleichungen an den Stellen x_1 bis x_M :

Reaktions-Advektions-Diffusions-Gleichung: $i = 1 :$

$$y_t(x_1, t) = -\frac{y_2(t) - y_1(t)}{h} + \nu \frac{y_2(t) - 2y_1(t) + 2h \cdot u_1(t)}{h^2} + \mu(y_1(t)(y_1(t) + 1)(1 - y_1(t)))$$

 $i = 2, \dots, M - 1 :$

$$y_t(x_i, t) = -\frac{y_{i+1}(t) - y_i(t)}{h} + \nu \frac{y_{i+1}(t) - 2y_i(t) + y_{i-1}(t)}{h^2} + \mu(y_i(t)(y_i(t) + 1)(1 - y_i(t))) \quad (6.23)$$

 $i = M :$

$$y_t(x_M, t) = -\frac{2hu_2(t) + y_{M-1}(t) - y_M(t)}{h} + \nu \frac{2hu_2(t) - 2y_M(t) + 2y_{M-1}(t)}{h^2} + \mu(y_{M-1}(t)(y_{M-1}(t) + 1)(1 - y_{M-1}(t)))$$

Wellengleichung: $i = 1 :$

$$y_{tt}(x_1, t) = \frac{2h \cdot u_1(t) + 2y_2(t) - 2y_1(t)}{h^2}$$

 $i = 2, \dots, M - 1 :$

$$y_{tt}(x_i, t) = \frac{y_{i-1}(t) - 2y_i(t) + y_{i+1}(t)}{h^2} \quad (6.24)$$

 $i = M :$

$$y_{tt}(x_M, t) = \frac{2y_{M-1}(t) - 2y_M(t) + 2h \cdot u_2(t)}{h^2}$$

7 Auswertung und numerische Ergebnisse

7.1 Allgemeine Voraussetzungen und Bedingungen

Die nachfolgenden Auswertungen wurde mittels einer Implementierung des *MPC-Verfahrens mit 1-Schritt-Regler* von JÜRGEN PANNEK durchgeführt. Die numerischen Ergebnisse basieren auf der Verwendung eines SQP-Verfahrens aus der *Numerical Algorithms Group Libraries* in der Programmiersprache C (kurz: NAGC¹), sowie dem SQP-Verfahren *NLPQLP²* (Nonlinear Programming with Non-Monotone and Distributed Line Search) in der Programmiersprache *Fortran* (kurz: SQPF) von Prof. Dr. KLAUS SCHITTKOWSKI zur Lösung der Optimalsteuerungsprobleme. Auf die sich bezüglich der L^2 - und H^1 -Norm unterscheidenden Zielfunktionale der Optimalsteuerungsprobleme wird in der jeweiligen Auswertung genauer eingegangen.

Das Ziel wird dabei sein, den kleinsten Horizont zur numerisch exakten Stabilisierung in die Null unter verschiedenen Anfangs- und Randbedingungen sowie bezüglich Unterscheidung in verteilte Steuerung und Randsteuerung zu finden. Dazu wurden Systeme unterschiedlicher Modifikationen für verschiedene Optimierungshorizonte N ausgewertet und anhand der daraus erhaltenen Resultate ein geeigneter Kandidat für einen minimalen stabilisierenden Horizont näher untersucht.

Für die gesamte Auswertung werden die Systeme auf einem **eindimensionalen Gebiet** $\Omega = (0, 1)$ mit **Rand** $\Gamma = \{0, 1\}$ betrachtet.

Bemerkung zu den Normen

Wir unterscheiden in der Auswertung bezüglich der das Zielfunktional unseres Optimalsteuerungsproblems definierenden Kostenfunktion verwendeten Normen folgendermaßen:

$$\mathbf{L^2\text{-Norm:}} \quad l(y(x, t), u(x, t)) = \|y(x, t)\|_{L^2(\Omega)}^2 + \lambda \|u(x, t)\|_{L^2(\hat{G})}^2$$

$$\mathbf{H^1\text{-Norm:}} \quad l(y(x, t), u(x, t)) = \|y(x, t)\|_{L^2(\Omega)}^2 + \|y_x(x, t)\|_{L^2(\Omega)}^2 + \lambda \|u(x, t)\|_{L^2(\hat{G})}^2$$

mit $\hat{G} = \Omega$ für verteilte bzw. $\hat{G} = \Gamma$ für Randsteuerung. Dabei entspricht $\lambda \in \mathbb{R}$ einer *Gewichtung* des Steuerungsaufwands.

¹Bei der NAGC Bibliothek handelt es sich um eine der umfangreichsten Sammlungen mathematischer und statistischer Algorithmen für C und C++. Eine ausführliche Dokumentation dazu findet sich unter <http://www.nag.co.uk/numeric/CL/CLdescription.asp>.

²Siehe hierzu <http://www.old.uni-bayreuth.de/departments/math/~kschittkowski/nlpqlp22.htm>, sowie [44, 45].

Bemerkung zu den Startwerten

Für das *Startprofil*, bestehend aus den Startwerten zum Zeitpunkt $t = 0$ an den Diskretisierungspunkten, verwenden wir folgende Wertebelegung:

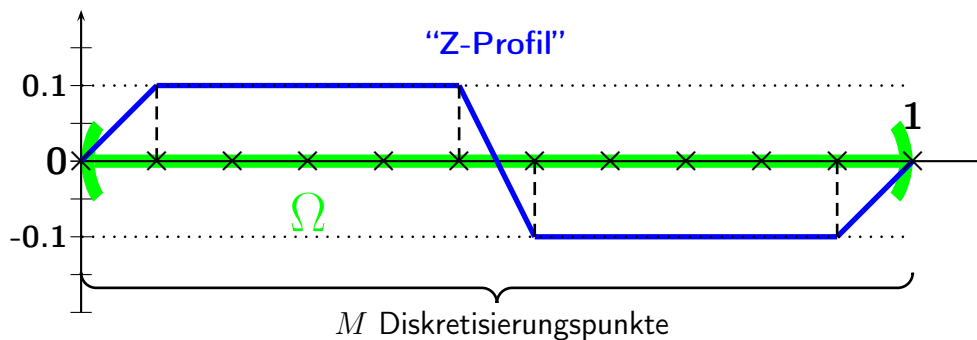
Sei M die Anzahl der *räumlichen Diskretisierungspunkte*, dann nennen wir im Weiteren das Startprofil $y_0(x_i)$, $i = 1, \dots, M$ mit

- $i \in \{1, M\}$ ist $y_0(x_i) = 0$ (Randpunkte)
- $i = 2, \dots, \lfloor M/2 \rfloor$ ist $y_0(x_i) = +0.1$,
- $i = \lceil M/2 \rceil, \dots, M - 1$ ist $y_0(x_i) = -0.1$

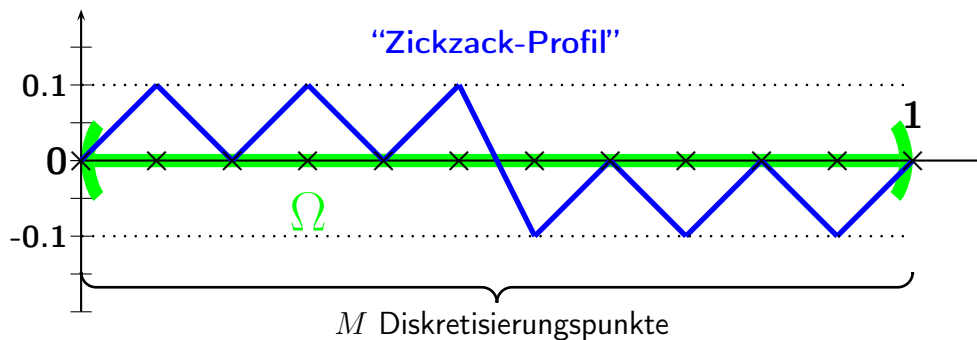
ein "Z-Profil" (siehe Abbildung 7.1 (a)).

Im späteren Verlauf der Auswertung findet eine Modifikation des Z-Profils Verwendung: das "Zickzack-Profil". Dazu wird, ausgehend vom Z-Profil, jeder zweite Wert auf Null gesetzt (siehe Abbildung 7.1(b)).

Des Weiteren bezeichnen wir als "Null-Profil" eine Wertebelegung mit Null an allen räumlichen Diskretisierungspunkten³ zu einem Zeitpunkt t .



(a) Z-Profil.



(b) Zickzack-Profil.

Abbildung 7.1: Veranschaulichung der verwendeten Startprofile.

³**Anmerkung:** Wenn wir von einer "Stabilisierung in die Null" sprechen, so ist exakt ausgedrückt die **Stabilisierung in das Null-Profil** gemeint

Bemerkung zur Startbelegung der Wellengeschwindigkeit $y_t(\cdot, 0)$

Die für die Implementation der Wellengleichung benötigte und in Abschnitt 2.4, Definition 2.11 eingeführte Reduktion der PDE auf ein System erster Ordnung erfordert für die Auswertung mittels des Programms eine Startbelegung der Wellengeschwindigkeit $y_t(\cdot, 0)$ zum Zeitpunkt $t = 0$. Diese wird für alle Diskretisierungspunkte mit dem Wert 0 initialisiert.

Bemerkung zu den Steuerungen

Bei *Randsteuerungen* wird ausschließlich an den Rändern (d.h. am ersten und am letzten Diskretisierungspunkt) des Gebiets Ω gesteuert, während bei einer *verteilten Steuerung* nur an den inneren Diskretisierungspunkten eine Steuerung anliegt (vergleiche hierzu Abbildung 7.2). Das bedeutet, dass man z.B. für $M = 25$ Diskretisierungspunkte somit 23 verteilte Steuerungen oder entsprechend 2 Randsteuerungen erhält.

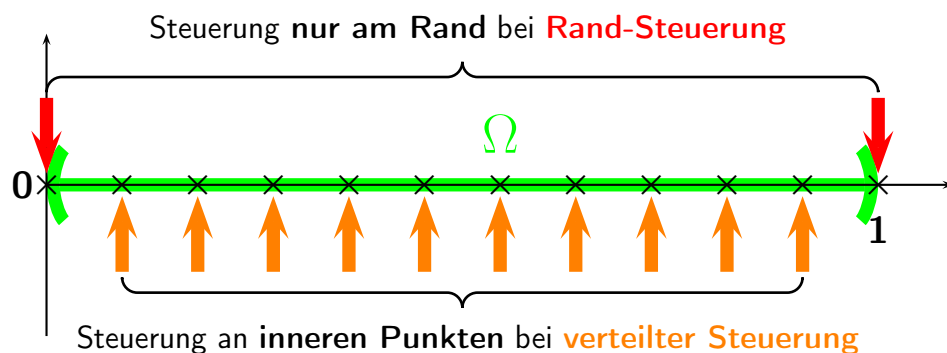


Abbildung 7.2: Anliegen der unterschiedlichen Steuerungen an den M Diskretisierungspunkten (dargestellt durch \times).

Bemerkung zu den SQP-Verfahren

Beim *Sequential Quadratic Programming* (kurz: SQP) handelt es sich um ein Verfahren zur Lösung nichtlinearer beschränkter Optimierungsprobleme. Dadurch, dass keine besonderen Anforderungen an die Struktur des Optimierungsproblems gestellt werden und dass eine schnelle lokale Konvergenz gegen einen stationären Punkt erreicht wird, eignet sich das SQP-Verfahren für die hier vorliegenden Optimalsteuerungsprobleme mit quadratischer Zielfunktion. Der SQP-Algorithmus geht dabei iterativ vor, indem er in jedem Iterationsschritt das nichtlineare Optimierungsproblem durch ein so genanntes quadratisches Optimierungsproblem approximiert und anschließend versucht, die daraus entstehende Folge von Teilproblemen zu lösen (vgl. [41]).

Bemerkung zur Notation

Es treten für Benennungen die folgenden Abkürzungen auf:

| | |
|----|---|
| RB | Randbedingung |
| DC | Verteilte Steuerung (engl. <i>distributed control</i>) |
| BC | Randsteuerung (engl. <i>boundary control</i>) |

| | |
|----------|---|
| DIR | Dirichlet |
| NEUM | Neumann-Randbedingung |
| NEUM1 | einseitige Neumann-Randbedingung |
| NEUM2 | zweiseitige Neumann-Randbedingung |
| ReAdDiff | Reaktions-Advektions-Diffusions-Gleichung |
| Welle | Wellengleichung |

7.2 Numerische Auswertung der Reaktions-Advektions-Diffusions-Gleichung

In den nachfolgenden Abschnitten wurde die *Reaktions-Advektions-Diffusions-Gleichung*

$$y_t = -y_x + \nu y_{xx} + \mu y(y + 1)(1 - y)$$

ausschließlich mittels des SQPF-Verfahrens unter Dirichlet- und einseitiger Neumann-Randbedingung ausgewertet und ihr Stabilisierungsverhalten sowohl für verteilte Steuerung als auch für Randsteuerung untersucht.

Wir möchten darauf hinweisen, dass diese Auswertung anfänglich als eine Art “Motivation” für die im zweiten Teil ausgewertete *Wellengleichung* gedacht war. Dabei wurde hier hauptsächlich nach dem Trial-and-Error-Prinzip vorgegangen, um einerseits ein “Gefühl” für die Auswirkungen von Änderungen der Parameterbelegungen und Systemmodifikationen zu erhalten, als auch programmtechnische Anpassungen und Verbesserungen vornehmen zu können. Da sich aber bereits ab Beginn dieser Auswertung einige aufschlussreiche und interessante Beobachtungen feststellen ließen, wollen wir nun die von uns erhaltenen Ergebnisse und Resultate in vollem Umfang präsentieren.

7.2.1 Ausgangssystem und -parameter

Zur Auswertung der Reaktions-Advektions-Diffusions-Gleichung wurde die folgende Belegung der Parameterwerte verwendet:

Ausgangsparameter

| | |
|--|-----------------------|
| Startprofil zum Zeitpunkt t_0 | Z-Profil |
| Startzeitpunkt t_0 | 0 |
| Angestrebtes Endprofil zum Zeitpunkt t_f | Null-Profil |
| Endzeitpunkt t_f | variiert |
| Anzahl Diskretisierungspunkte M | 25 |
| Abtastperiode T | 0.025 |
| μ | 10.0 |
| ν (Diffusionsfaktor) | 0.1 |
| λ (Gewichtung der Steuerung in der Zielfunktion) | 10^{-3} |
| Steuerbeschränkungen | $] -\infty, +\infty[$ |
| Zustandsbeschränkungen | $] -\infty, +\infty[$ |

Tabelle 7.2: Ausgangsparameter für die Auswertung der ReAdDiff

Auf Abweichungen bzw. Veränderungen der Parameter wird gegebenenfalls in den einzelnen Auswertungen hingewiesen.

Implementation der Normen

Die in der Zielfunktion auftretenden Normen wurden folgendermaßen implementiert:

- Normen im Inneren:

$$\|y(i, \cdot)\|_{L^2(\Omega)}^2 := \frac{1}{M-2} \sum_{j=2}^{M-1} y(i, j)^2 \quad (7.1)$$

$$\|y_x(i, \cdot)\|_{L^2(\Omega)}^2 := \frac{1}{M-2} \sum_{j=2}^{M-1} \left(\frac{y(i+1, j) - y(i, j)}{h} \right)^2 \quad (7.2)$$

$$\|u(i, \cdot)\|_{L^2(\Omega)}^2 := \frac{1}{M-2} \sum_{j=2}^{M-1} u(i, j)^2 \quad (7.3)$$

- Norm auf dem Rand:

$$\|u(i, \cdot)\|_{L^2(\Gamma)}^2 := \frac{1}{2} \sum_{j=1}^2 u(i, j)^2 \quad (7.4)$$

Diskretisierung des Systems

Für die in der Auswertung benötigte diskrete Formulierung der Reaktions-Advektions-Diffusions-Gleichung verweisen wir auf die Abschnitte 4.2.2 und 6.3.

7.2.2 Dirichlet-Randbedingung

Wir untersuchen das System unter der Dirichlet-Randbedingung (siehe Abschnitt 2.5.2.1) auf dem Gebiet $\Omega = (0, 1)$ mit Rand $\Gamma := \partial\Omega$:

$$y(0, t) = \sigma_1 = 0, \quad y(1, t) = \sigma_2 = 0$$

Bemerkung 7.1

Für die Diskretisierung unseres Systems unter Dirichlet-Randbedingung bei verteilter Steuerung vergleiche Abschnitt 6.3, Diskretisierung (6.10) sowie bei Randsteuerung Abschnitt 6.3, Diskretisierung (6.13).

7.2.2.1 Verteilte Steuerung mit L^2 -Norm im Zielfunktional

Das optimale Steuerungsproblem ist für Dirichlet-Randbedingung mit verteilter Steuerung und L^2 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ &\text{mit } l(y(\cdot, nT), u(\cdot, nT)) = \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2 \\ \\ \text{s.t.} \\ &u = y_t + y_x - \nu y_{xx} - \mu y(y+1)(1-y) \\ &y(x, 0) = y_0(x) \quad \text{für } x \in \Omega \\ &y(0, t) = 0 \\ &y(1, t) = 0 \\ &u(x, t) \in U \quad \text{für } x \in \Omega \\ &u(\cdot, t) = u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \\ \\ \text{mit} \\ &\Omega = (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}. \end{aligned}$$

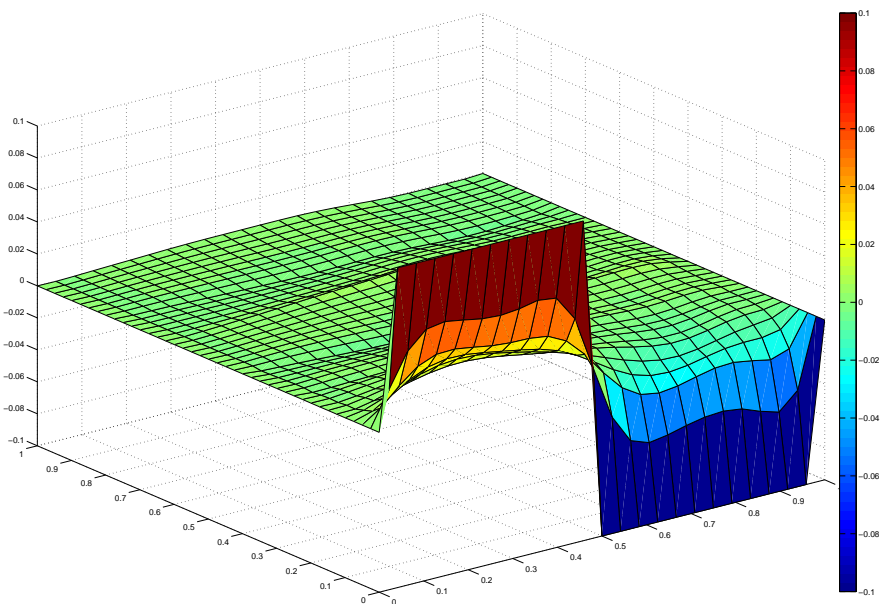
Bemerkung 7.2

In dieser Auswertung haben wir die mittels MATLAB berechneten Ergebnisse aus Kapitel 7 der Arbeit “Analysis and design of unconstrained nonlinear MPC schemes for finite and infinite dimensional systems” von L. GRÜNE [24] nachgebaut und verifiziert. Aufgrund einer im weiteren Verlauf unserer Auswertung verwendeten kleineren Gewichtung der Steuerung in der Zielfunktion $\lambda = 0.001$ (anstelle von $\lambda = 0.1$ entsprechend genannter Arbeit), erhalten wir nun wie erwartet kleinere stabilisierende Horizonte. \circ

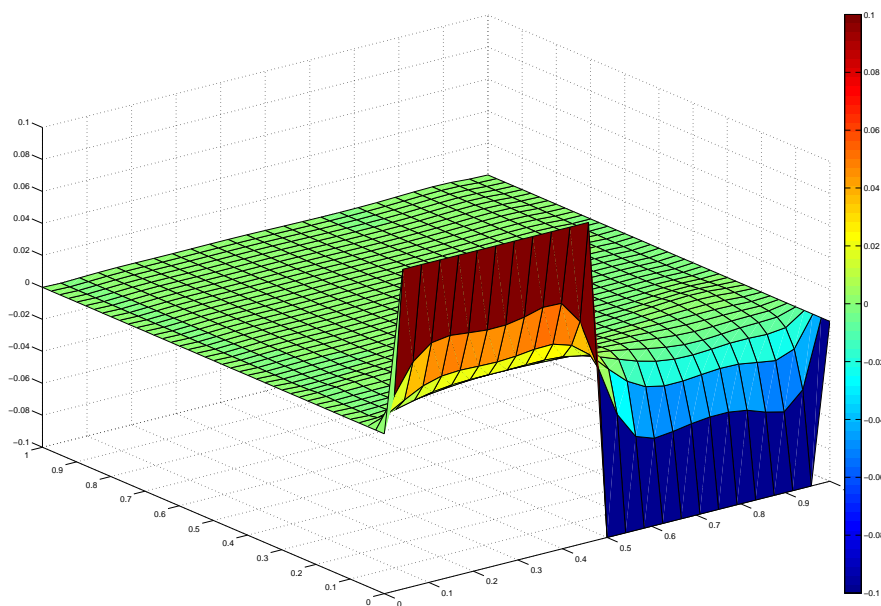
Beobachtungen

Bereits bei einem Stabilisierungshorizont von $N = 3$ für Endzeit $t_f = 1.0$ können wir eine “gute” Stabilisierung in die Null beobachten. Die im Verlauf erkennbaren Abweichungen von der Null in Form kleinerer “Wellenberge und -täler” beruhen vermutlich auf numerischen Rechenungenauigkeiten. Eine weitere Möglichkeit der Erklärung für diese ist, dass hier die Eigenschwingung des Systems angeregt wird. Man kann erkennen, dass sich der Verlauf des Systems spätestens zum Zeitpunkt $t = 0.25$ in einer Umgebung des angestrebten Null-Profiles aufhält (Abbildung 7.3(a)).

Der Vollständigkeit halber wollen wir auch den von uns berechneten Plot für $N = 11$ zeigen (siehe Abbildung 7.3(b)), zumal hier die bereits genannten Wellenberge minimal bis gar nicht mehr zu erkennen sind.



(a) Für $N = 3$ erhält man bereits eine gute Stabilisierung in einer Umgebung der Null.



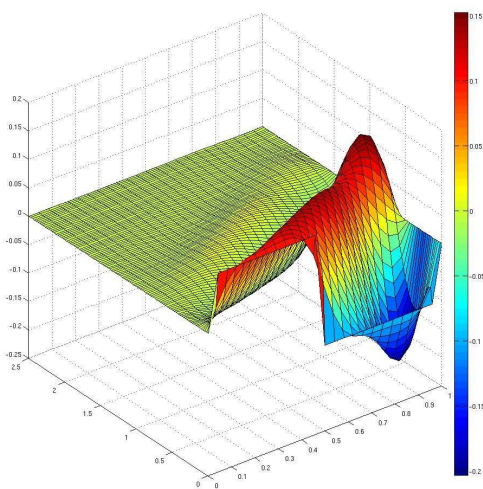
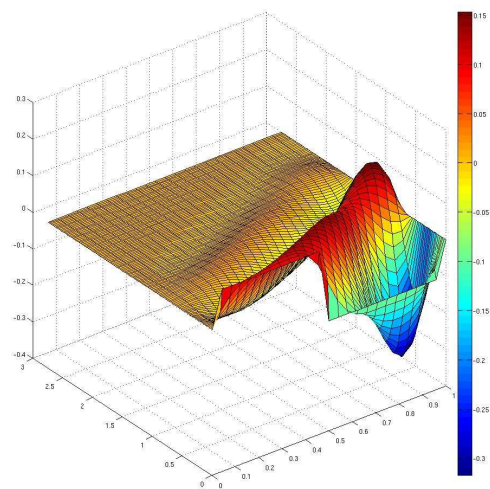
(b) Für $N = 11$ wird schon nach kurzer Zeit eine sehr gute Stabilisierung in die Null erreicht.

Abbildung 7.3: Dirichlet-RB, DC, L^2 -Norm

Modifikation: Heraufsetzen des Parameters λ

Aufgrund der erhaltenen Verbesserungen wird in Betracht gezogen das System dahingehend zu modifizieren, dass wir die Gewichtung λ der Steuerung in der Zielfunktion erhöhen. Damit soll untersucht werden, inwieweit sich eine ‐Verteuerung der Steuerung‐ auf den stabilisierenden Horizont auswirkt. Die folgenden beiden Falle werden untersucht:

- $\lambda = 0.1$: Man erhalt eine Stabilisierung ab einem Horizont $N = 8$ fur Endzeit $t_f = 2.5$, wie sich in Abbildung 7.4 (a) erkennen lasst.
- $\lambda = 0.5$: Bei weiterer Erhohung des λ -Werts auf 0.5 gelingt die Stabilisierung in die Null erst ab einem Horizont von $N = 14$ und einer Endzeit $t_f > 2.75$ (vgl. Abbildung 7.4 (b)).

(a) $\lambda = 0.1$, $N = 8$ und $t_f = 2.5$ (b) $\lambda = 0.5$, $N = 14$ und $t_f > 2.75$ Abbildung 7.4: Modifikation von λ bei Dirichlet-RB mit DC, L^2 -Norm

Fazit: Ein Heraufsetzen der Gewichtung λ um Faktor 100 auf $\lambda = 0.1$ bewirkt eine Erhohung des Stabilisierungshorizonts um einen Faktor von ≈ 2.6 auf $N = 8$. Eine Verteuerung der Steuerung im Zielfunktional durch eine Erhohung von λ um einen Faktor 500 auf $\lambda = 0.5$ hat eine Erhohung des Stabilisierungshorizonts um einen Faktor von ≈ 4.6 auf $N = 14$ zur Folge.

7.2.2.2 Verteilte Steuerung mit H^1 -Norm im Zielfunktional

Wir verwenden nun die H^1 -Norm in unserer Zielfunktion, um deren Auswirkung auf die Stabilisierung zu untersuchen.

Das optimale Steuerungsproblem ist für Dirichlet-Randbedingung mit verteilter Steuerung und H^1 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ \text{mit } l(y(\cdot, nT), u(\cdot, nT)) &= \\ &= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2 \end{aligned}$$

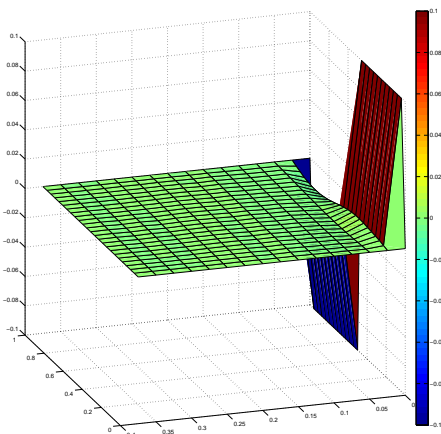
s.t.

$$\begin{aligned} u &= y_t + y_x - \nu y_{xx} - \mu y(y+1)(1-y) \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y(0, t) &= 0 \\ y(1, t) &= 0 \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \end{aligned}$$

mit

$$\Omega = (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}.$$

Beobachtungen



Wie für die L^2 -Norm erhalten wir als minimalen stabilisierenden Horizont $N = 3$, wobei eine Verbesserung bezüglich des Zeitpunkts t eintritt, ab dem der Verlauf in der Null angelangt ist: dieser beträgt hier nur $t = 0.1$ (siehe Abbildung 7.5).

Abbildung 7.5: Dirichlet-RB, DC, H^1 -Norm, $N = 3$, $t_f = 1.0$

7.2.2.3 Randsteuerung mit L^2 -Norm im Zielfunktional

Bei der Randsteuerung haben wir jetzt im Vergleich zur verteilten Steuerung nur noch zwei Steuerparameter zur Verfügung, von denen jeweils einer an einem Rand des Gebiets Einfluss auf das System nimmt.

Das optimale Steuerungsproblem ist für Dirichlet-Randbedingung mit Randsteuerung und L^2 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\min J_N(y_0(\cdot), u) = \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT))$$

mit $l(y(\cdot, nT), u(\cdot, nT)) = \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Gamma)}^2$

s.t.

$$\begin{aligned} 0 &= y_t + y_x - \nu y_{xx} - \mu y(y+1)(1-y) \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y(0, t) &= u(0, t) \\ y(1, t) &= u(1, t) \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \end{aligned}$$

mit

$$\Omega = (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}.$$

Beobachtungen

Für die Randsteuerung können die bei verteilter Steuerung erhaltenen minimalen Stabilisierungshorizonte mit den bisherigen Parameterwerten nicht verifiziert werden. Deshalb versuchen wir für Horizont $N = 3$ durch Herabsetzen der Gewichtung λ der Steuerung in der Zielfunktion eine Stabilisierung zu erreichen. Aber selbst für Werte von bis zu $\lambda = 10^{-7}$ stellt sich keine Stabilisierung in die Null ein, obwohl in einem solchen Fall die Steuerung in der Zielfunktion so gut wie "nichts mehr kostet" (vergleiche Abbildung 7.6). Vielmehr erhalten wir hier eine Stabilisierung in die 1.

Dies legt die Vermutung nahe, dass sich eine Stabilisierung hier nur über eine Vergrößerung des Optimierungshorizonts N erreichen lässt.

Aber auch für $\lambda = 10^{-4}$ und $N = 4, \dots, 10$ kann für eine Endzeit von $t_f > 10$ keine Stabilisierung in die Null erzielt werden. Eine weitere Erhöhung des Optimierungshorizonts N um 1 auf $N = 11$ bewirkt jedoch den Erhalt einer Stabilisierung für Endzeit $t_f = 3.5$ (vergleiche Abbildung 7.7).

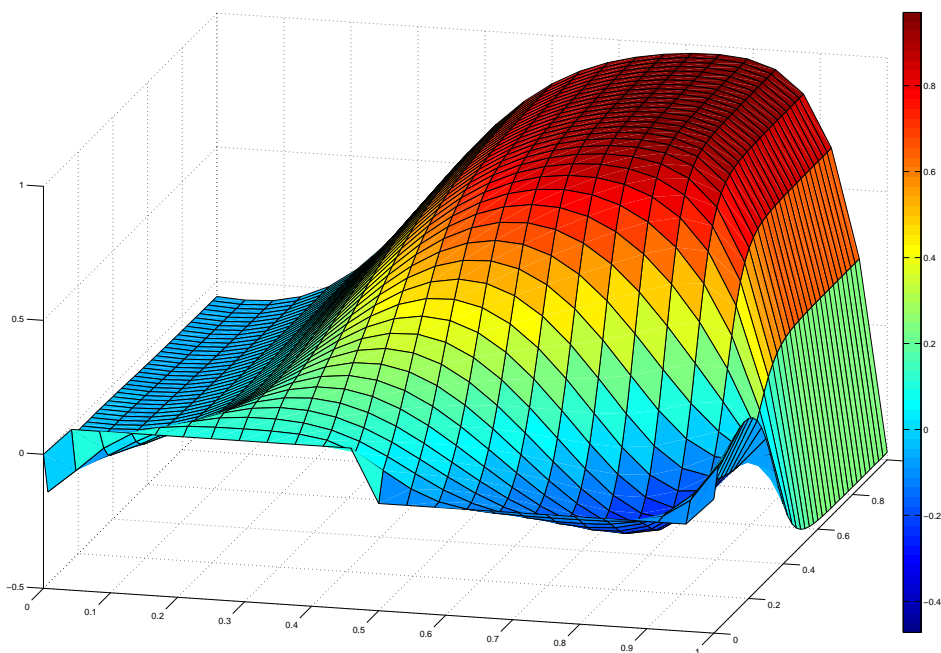


Abbildung 7.6: Dirichlet-RB, BC, L^2 -Norm, $N = 3$ und $\lambda = 10^{-5}$

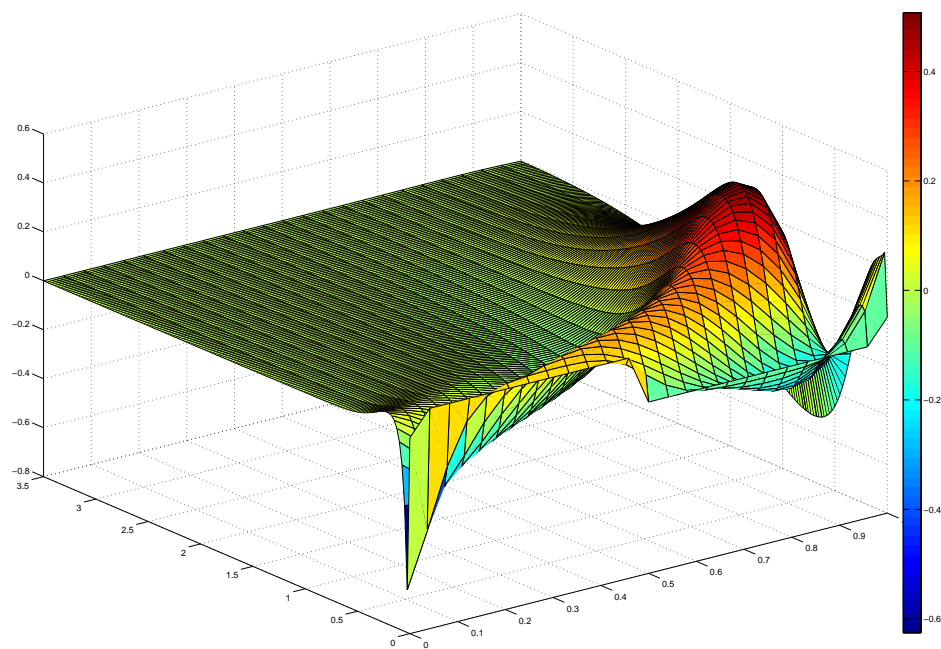


Abbildung 7.7: Dirichlet-RB, BC, L^2 -Norm, $N = 11$ und Endzeit $t_f = 3.5$

7.2.2.4 Randsteuerung mit H^1 -Norm im Zielfunktional

Nachdem die Randsteuerung mit der L^2 -Norm nur ab einem Horizont von $N = 11$ zu der gewünschten Stabilisierung in die Null führt, wir aber an einem kleineren Horizont interessiert sind, betrachten wir nun die Auswirkungen der H^1 -Norm im Zielfunktional. Das optimale Steuerungsproblem ist für Dirichlet-Randbedingung mit Randsteuerung und H^1 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ \text{mit } l(y(\cdot, nT), u(\cdot, nT)) &= \\ &= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Gamma)}^2 \\ \text{s.t.} \\ 0 &= y_t + y_x - \nu y_{xx} - \mu y(y+1)(1-y) \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y(0, t) &= u(0, t) \\ y(1, t) &= u(1, t) \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \\ \text{mit} \\ \Omega &= (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}. \end{aligned}$$

Beobachtungen

Laut dem zu Beginn erwähnten Paper von L. GRÜNE erhält man bei verteilter Steuerung, im Vergleich zur L^2 -Norm, für die H^1 -Norm eine Verbesserung der Stabilisierung in die Null. Es stellt sich jetzt die Frage, ob diese Verbesserung auch für eine Randsteuerung beobachtet werden kann. Dazu setzen wir $\lambda = 10^{-3}$ und den Stabilisierungshorizont auf $N = 2$. Dabei ist zu beobachten, dass sich selbst für eine große Endzeit $t_f > 2.5$ keine Stabilisierung in die Null einstellt. Analog zur L^2 -Norm bei Randsteuerung erhalten wir auch hier eine Stabilisierung in die 1.

Selbst bei Variation des Optimierungshorizonts ($2 < N \leq 16$) und der Gewichtung λ (z.B. $= 10^{-5}, \dots, 10^{-8}$) ergeben sich keine Veränderungen in Bezug auf die Stabilisierung in die Null (vergleiche Abbildung 7.8).

Dies lässt sich folgendermaßen begründen: Die Stabilisierung für den Fall der verteilten Steuerung aus dem Paper beruht auf der Erfüllung der Voraussetzung

$$\|u(n, \cdot)\|_{L^2(\Omega)}^2 \geq \|y_x(n, \cdot)\|_{L^2(\Omega)}^2, \quad (7.5)$$

d.h. die Steuerung u muß die Ableitung y_x kompensieren.

Diese Voraussetzung zu erfüllen gelingt in unserem Fall der Randsteuerung für $2 < N \leq 16$ jedoch nicht. Die L^2 -Norm von y_x ist stets größer als die L^2 -Norm der Steuerung u .

Die Steuerung u bewirkt in unserem System eine Veränderung von y . Damit einhergehend erhält man eine Änderung von y_x , da y_x der örtlichen Ableitung von y entspricht. Bezüglich der hier verwendeten Randsteuerung bedeutet dies, dass ein Steuern nach Null nur dann möglich ist, wenn die Norm der Ableitung $\|y_x\|_{L^2(\Omega)}^2$ zwischenzeitlich große Werte annimmt. Resultiert also eine betragsmäßig große Steuerung u in einer betragsmäßig großen Veränderung von y und damit in einem großen Wert $\|y_x\|_{L^2(\Omega)}^2$, so nimmt auch der Wert der zu minimierenden Zielfunktion zu.

Da unser System eine Minimierung der Kosten (d.h. hier: Minimierung der Abweichung von y vom Wert Null, Minimierung der Ableitung y_x und Minimierung der Steuerung u) zum Ziel hat, erhalten wir nur kleine Kontrollen u , die aber nicht ausreichen, wie angestrebt in die Null zu stabilisieren. Vielmehr bewegt sich die Lösung in ein stabiles Gleichgewicht bei der 1, da kein Anreiz besteht von dieser Ruhelage abzuweichen, zumal Steuerung u und Ableitung y_x hier identisch Null sind. Die Kosten bestehen hier somit nur aus $\|y\|_{L^2(\Omega)}^2$, die gerade bei 1 liegen (vergleiche hierzu [24], S. 22f).

Bemerkung 7.3

Setzen wir in unserer zugrundeliegenden PDE

$$y_t = -y_x + \nu y_{xx} + \mu y(y+1)(1-y)$$

für den Wert 1 im nichtlinearen Term zum Beispiel den Wert 2, so erhalten wir entsprechend eine Stabilisierung in die 2. ◦

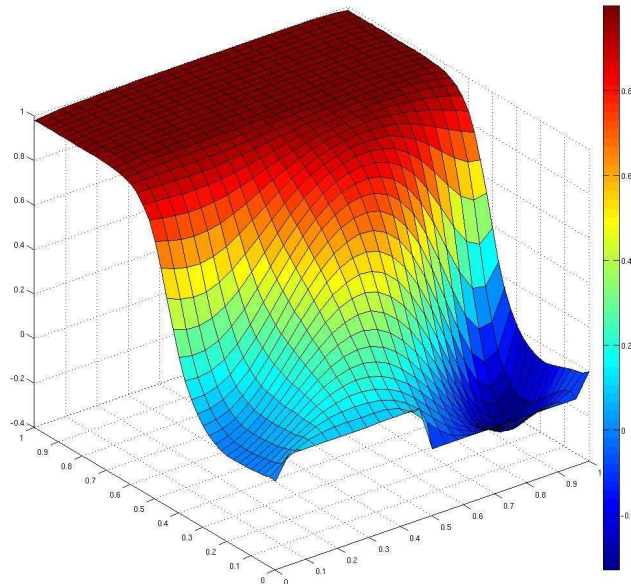


Abbildung 7.8: Dirichlet-RB, BC, H^1 -Norm, $N = 5$

Wählt man jedoch für den Stabilisierungshorizont einen Wert $N \geq 17$, so erhält man bereits für die Endzeit $t_f = 1.25$ die gewünschte Stabilisierung in die Null (vergleiche Abbildung 7.9).

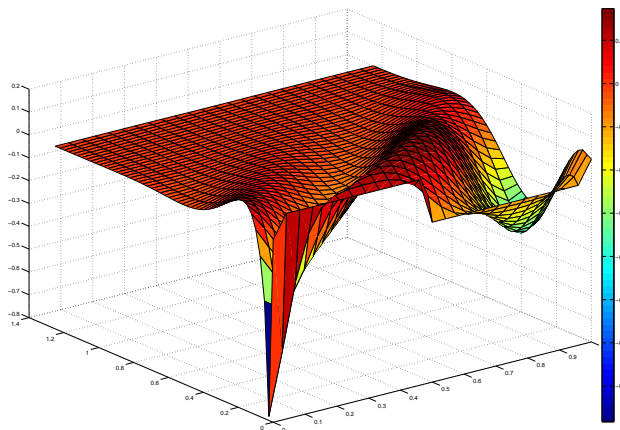


Abbildung 7.9: Dirichlet-RB, BC, H^1 -Norm, $N = 17$ und Endzeit $t_f = 1.25$

Stabilisierung durch Modifikation des Systems

Da eine Stabilisierung in die Null des Ausgangssystems nur für großen Horizont (hier: $N \geq 17$) gelingt, wollen wir hier nun zwei Varianten von Modifikationen untersuchen, die auf der Veränderung zweier das System definierender Parameter basieren.

Variante 1: Änderung von ν

Durch Heraufsetzen des *Diffusionsparameters* ν von ursprünglich 0.1 auf einen Wert ≥ 0.35 erreicht man ab einem Horizont $N \geq 8$ die gewünschte Stabilisierung (vergleiche Abbildung 7.10). Da es sich beim Diffusionsparameter ν um einen Wert für die gleichmäßige Verteilung von Teilchen in einem Stoff handelt, bewirkt eine Veränderung dieses Parameters auch eine substantielle Änderung des zu Grunde liegenden Systems, so dass hier ein einfacher Vergleich des Lösungsverlaufs des Systems mit dem ursprünglichen Lösungsverlauf nicht mehr möglich ist.

Variante 2: Hinzunahme einer Gewichtung für y_x im Kostenfunktional

Wir verändern die Zielfunktion dahingehend, dass wir die Ableitung y_x ähnlich der Steuerung nun mit einer Gewichtung $\omega = 0.01$ versehen:

$$l(y(\cdot, nT), u(\cdot, nT)) = \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \omega \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Gamma)}^2$$

Auch in diesem Fall erhält man nun bereits für die Endzeit $t_f = 1.5$ die gewünschte Stabilisierung, allerdings für einen Horizont von $N = 20$. Dieser kann aber durch weiteres Verringern der Gewichtung ω nach unten korrigiert werden. Dabei sei aber

7 Auswertung und numerische Ergebnisse

zu beachten, dass für ein gegen Null gehendes ω ($\omega \rightarrow 0$) die Norm der Ableitung y_x in der Zielfunktion praktisch keine “Kosten” mehr verursacht und wir damit den bereits behandelten Fall der Dirichlet-Randsteuerung mit L^2 -Norm erhalten. Hier liegt das gleiche Prinzip zu Grunde wie in Variante 1: Eine Erhöhung von ν bewirkt einen “flacheren Wellenverlauf”, was wiederum zu einer kleineren örtlichen Veränderung von y und damit zu einem kleineren y_x führt. Selbiges erreicht man durch die Gewichtung mit ω (siehe dazu Abbildung 7.11).

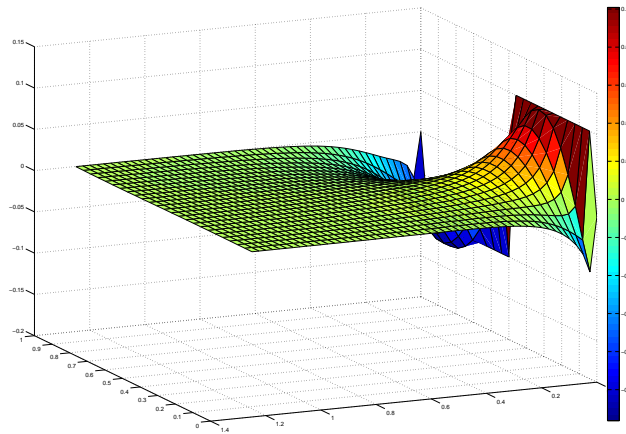


Abbildung 7.10: Dirichlet-RB, BC, H^1 -Norm, $N = 10$ und $\nu = 0.35$

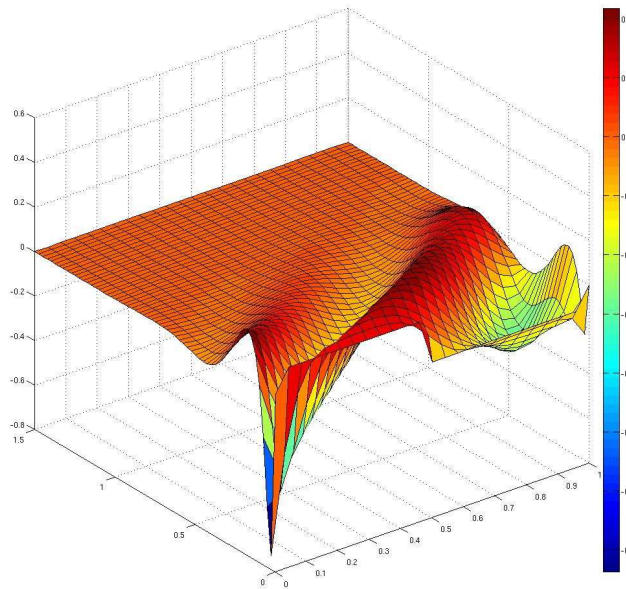


Abbildung 7.11: Dirichlet-RB, BC, H^1 -Norm, $N = 20$ und $\omega = 0.01$

7.2.2.5 Änderung der Anfangswerte: Zickzack-Profil

Wir erhalten eine weitere Variation unseres Systems durch Veränderung des Ausgangsprofils, indem wir anstelle des Z-Profiles nun das Zickzack-Profil wählen.

Für diese Auswertung widmen wir uns der Randsteuerung, da man für eine verteilte Steuerung ausschließlich eine Verbesserung der Stabilisierung erhält. Dies begründen wir damit, dass beim Zickzack-Profil im Vergleich zum vorher verwendeten Z-Profil die Hälfte der Werte bereits auf Null gesetzt ist. In der Zielfunktion findet die L^2 -Norm Verwendung, da sich für den Fall der H^1 -Norm keine Stabilisierung einstellt.

Beobachtungen

Für $N = 10$ wird für eine Endzeit $t_f = 2.5$ noch keine Stabilisierung erreicht, auch wenn sich aus dem Verlauf der Lösungen anhand der bereits gemachten Beobachtungen vermuten lässt, dass diese durch Veränderung der Parameter erreicht werden kann. Dies begründen wir damit, dass ein Abfallen der Welle erkennbar ist, gegen Ende hin aber deutlich schwächer wird. Erst ab $N = 12$ stellt sich die Stabilisierung in die Null ein, und das bereits für eine Endzeit $t_f = 2$ (siehe Abbildung 7.12).

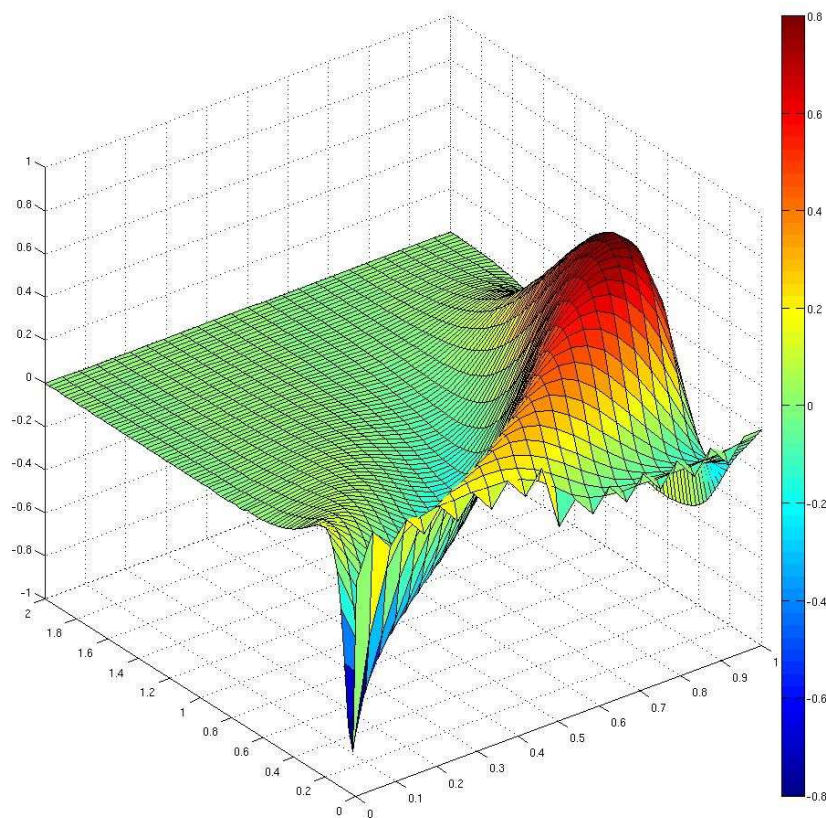


Abbildung 7.12: Dirichlet-RB, BC, L^2 -Norm, $N = 12$, "Zickzack-Profil", Endzeit $t_f = 2$

7.2.2.6 Erhöhung der Anzahl der Diskretisierungspunkte

Für eine Verdoppelung der Anzahl der Diskretisierungspunkte von 25 auf 50 sind leichte Verbesserungen bezüglich der Stabilisierung erkennbar, jedoch nicht so entscheidend wie erwartet, obwohl man nun doppelt so viele Punkte an einem Zeitschritt auswertet und sozusagen “doppelt so viel Informationen” zur Berechnung der nächsten Steuerfolge zur Verfügung hat. Dahingehend wird diese Variation für dieses System nicht weiter verfolgt. Einer genaueren Betrachtung der Auswirkung einer Änderung der Anzahl der Diskretisierungspunkte werden wir uns bei der Wellengleichung widmen.

7.2.3 Neumann-Randbedingung

Analog zu der Dirichlet-Randbedingung werden die Systeme für den Fall einer einseitigen Neumann-Randbedingung (siehe Abschnitt 2.5.2.2) ausgewertet. Auf einem Gebiet $\Omega = (0, 1)$ schreiben wir diese Bedingung am linken Rand $\Gamma := \partial\Omega$ folgendermaßen:

$$\frac{\partial y}{\partial \nu}(0, t) = \sigma_3 \quad \text{mit } \sigma_3 \in \mathbb{R}$$

Am rechten Rand verwenden wir weiterhin die Dirichlet-Randbedingung.

Bemerkung 7.4

Für die Diskretisierung unseres Systems unter Neumann-Randbedingung bei verteilter Steuerung vergleiche Abschnitt 6.3, Diskretisierung (6.16) und bei Randsteuerung Abschnitt 6.3, Diskretisierung (6.23). ◦

7.2.3.1 Verteilte Steuerung mit L^2 -Norm im Zielfunktional

Das optimale Steuerungsproblem ist für Neumann-Randbedingung mit verteilter Steuerung und L^2 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ &\text{mit } l(y(\cdot, nT), u(\cdot, nT)) = \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2 \\ \text{s.t.} \\ u &= y_t + y_x - \nu y_{xx} - \mu y(y+1)(1-y) \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y_\nu(0, t) &= 0,5 \\ y(1, t) &= 0 \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \\ \text{mit} \\ \Omega &= (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}. \end{aligned}$$

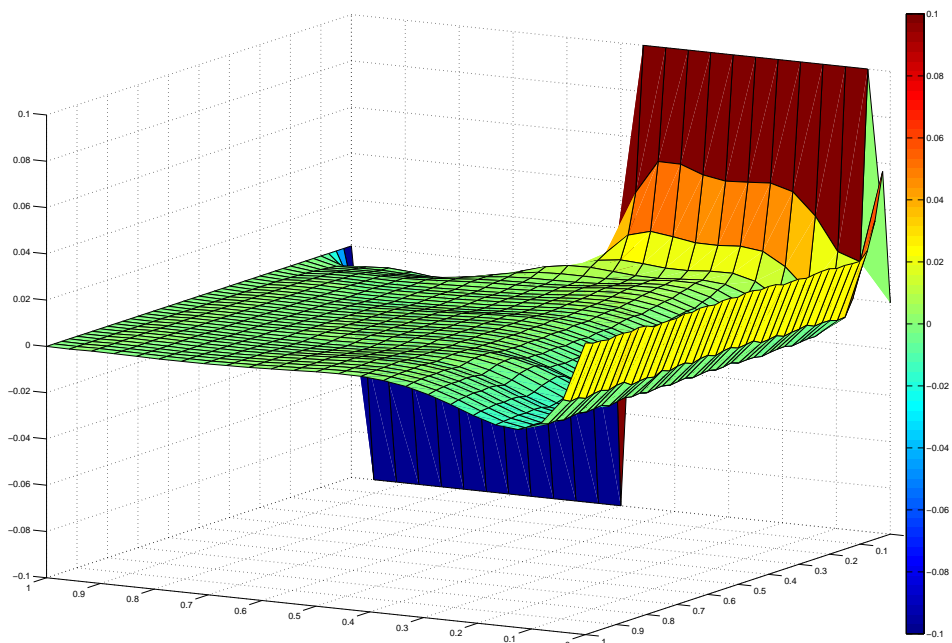
Bemerkung 7.5

Wir weisen darauf hin, dass wir für die Neumann-Randbedingung am Rand $x_1 = 0$ den Wert der Ableitung von y in Richtung der äußeren Normalen auf konstant $\sigma_3 = 0.5$ setzen, da man sonst für eine Wertebelegung gleich Null den bereits behandelten Fall der Dirichlet-Randbedingung erhält. Dabei ist der Wert willkürlich gewählt und soll lediglich den Einfluss einer Neumann-Randbedingung auf das Stabilisierungsverhalten verdeutlichen. \circ

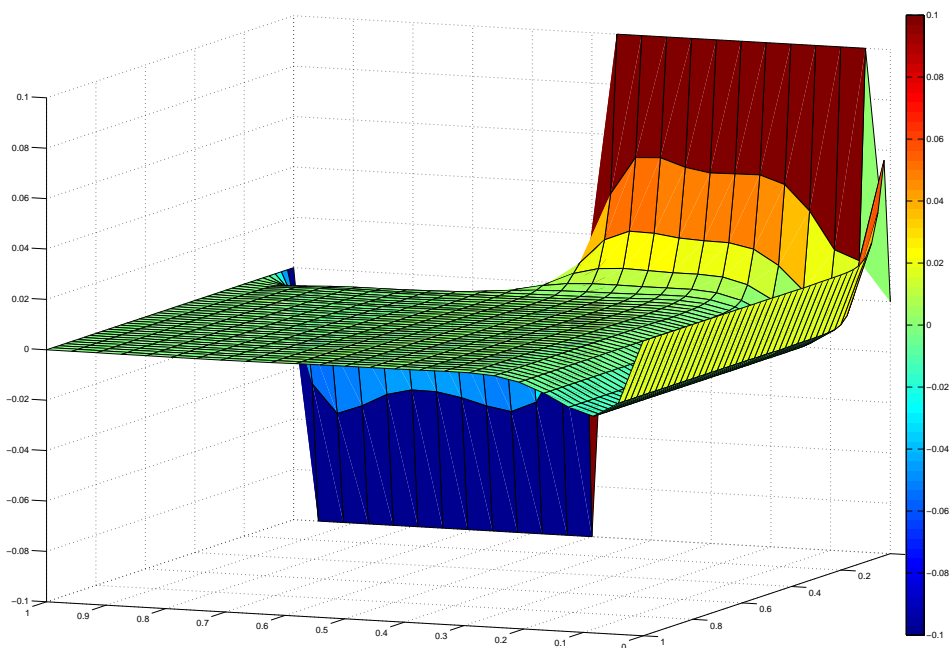
Beobachtungen

Für $N = 5$ erhält man bereits eine gute Stabilisierung in die Null, wobei auch hier wie im Fall der Dirichlet-Randbedingung eine Abweichung in Form von kleinen Wellenverläufen entlang der Zeit auftritt. Diese verschwinden aber durch Erhöhung des Horizonts.

Aufgrund der Neumann-Randbedingung und deren Auswirkung auf benachbarte Diskretisierungspunkte entsteht in der Nähe des Randes eine "unstabilisierbare Rinne", deren Größe mit zunehmendem Horizont abnimmt und nur für einen Ableitungswert gleich Null gänzlich verschwindet (vergleiche dazu Abbildung 7.13). Da aufgrund dieser Tatsache eine exakte Stabilisierung in das Null-Profil, d.h. also auch am Rand, nicht erreichbar ist, sprechen wir hier von einer "(teilweisen) Stabilisierung bis auf den linken Rand".



(a) $N = 5$



(b) $N = 15$

Abbildung 7.13: Neumann-RB, DC, L^2 -Norm

7.2.3.2 Verteilte Steuerung mit H^1 -Norm im Zielfunktional

Das optimale Steuerungsproblem ist für Neumann-Randbedingung mit verteilter Steuerung und H^1 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ \text{mit } l(y(\cdot, nT), u(\cdot, nT)) &= \\ &= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2 \\ \text{s.t.} \\ u &= y_t + y_x - \nu y_{xx} - \mu y(y+1)(1-y) \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y_\nu(0, t) &= 0,5 \\ y(1, t) &= 0 \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \\ \text{mit} \\ \Omega &= (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}. \end{aligned}$$

Beobachtungen

Im Gegensatz zur L^2 -Norm erhalten wir hier für einen Horizont von $N = 3$ ein anderes Stabilisierungsverhalten: Vielmehr zeigt sich nun eine Änderung unserer "Rinne" am linken Rand in einen vom rechten Rand auf Höhe 0 beginnenden und auf -0.02 abfallenden **ebenen** Verlauf (vergleiche Abbildung 7.14). Dies begründen wir durch die Hinzunahme der Ableitung in die Zielfunktion und den dadurch entstehenden höheren Kosten für das Erfüllen der Neumann-Randbedingung. Im H^1 -Sinne jedoch ist das sich hier einstellende Profil billiger als das bezüglich der L^2 -Norm erhaltenen Profil.

Aufgrund des im zeitlichen sehr ebenen, im örtlichen Profil aber abfallenden Verlaufs, lässt sich vermuten, dass ein Setzen der Neumann-Randbedingung $y_\nu(0, t)$ auf Null die gewünschte Stabilisierung wie im Dirichlet-Fall erzielt.

Auch für größere Horizonte $N > 3$ stellt sich keine Stabilisierung in die Null ein.

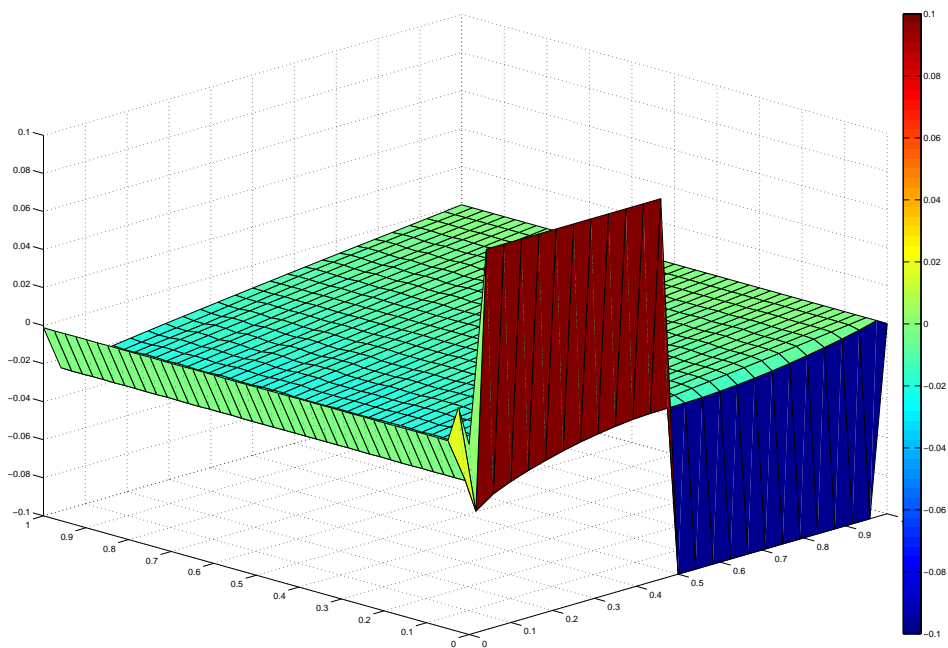


Abbildung 7.14: Neumann-RB, DC, H^1 -Norm, $N = 10$

7.2.3.3 Randsteuerung mit L^2 -Norm im Zielfunktional

Hinweis: Es wird an beiden Rändern die Neumann-Randsteuerung verwendet.

Das optimale Steuerungsproblem ist für Neumann-Randbedingung mit Randsteuerung und L^2 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\min J_N(y_0(\cdot), u) = \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT))$$

$$\text{mit } l(y(\cdot, nT), u(\cdot, nT)) = \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Gamma)}^2$$

s.t.

$$0 = y_t + y_x - \nu y_{xx} - \mu y(y+1)(1-y)$$

$$y(x, 0) = y_0(x) \quad \text{für } x \in \Omega$$

$$y_\nu(0, t) = u(0, t)$$

$$y_\nu(1, t) = u(1, t)$$

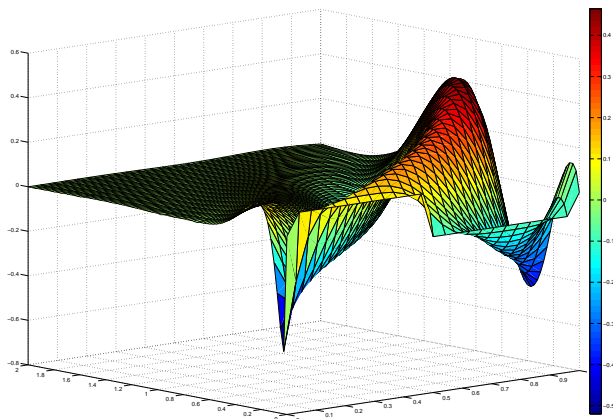
$$u(x, t) \in U \quad \text{für } x \in \Omega$$

$$u(\cdot, t) = u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T)$$

mit

$$\Omega = (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}.$$

Beobachtungen



Analog zum Fall der Dirichlet-Randbedingung stellt sich hier für einen Horizont $N \geq 11$ die gewünschte Stabilisierung ein (siehe Abbildung 7.15).

Abbildung 7.15: Neumann-RB, BC, L^2 -Norm, $N = 11$

7.2.3.4 Randsteuerung mit H^1 -Norm im Zielfunktional

Hinweis: Es wird an beiden Rändern die Neumann-Randsteuerung verwendet.

Das optimale Steuerungsproblem ist für Neumann-Randbedingung mit Randsteuerung und H^1 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\min J_N(y_0(\cdot), u) = \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT))$$

mit $l(y(\cdot, nT), u(\cdot, nT)) =$

$$= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Gamma)}^2$$

s.t.

$$0 = y_t + y_x - \nu y_{xx} - \mu y(y+1)(1-y)$$

$$y(x, 0) = y_0(x) \quad \text{für } x \in \Omega$$

$$y_\nu(0, t) = u(0, t)$$

$$y_\nu(1, t) = u(1, t)$$

$$u(x, t) \in U \quad \text{für } x \in \Omega$$

$$u(\cdot, t) = u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T)$$

mit

$$\Omega = (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}.$$

Beobachtungen

Bei Hinzunahme der Ableitung in die Zielfunktion ist keine Stabilisierung mit Hilfe des Programms unter Verwendung der diesbezüglich uns zur Verfügung stehenden Modifikationsmöglichkeiten erzielbar (vgl. Abbildung 7.16). Bei Beibehaltung der Steuerbeschränkungen (mit Obergrenze = 10^{20} und Untergrenze = 10^{-20}) gibt das Programm ab einem Horizont von $N = 20$ eine Fehlermeldung aus, die darauf hinweist, dass die maximale verfügbare Genauigkeit des Rechners erreicht bzw. überschritten wurde. Dies kann dahingehend vermieden werden, dass die Genauigkeiten der verwendeten Verfahren erhöht bzw. die Steuerbeschränkungen gesenkt werden. Auch dies wurde in mehreren Auswertungen untersucht, allerdings ist auch in diesen Fällen keine Tendenz zur Stabilisierung in die Null erkennbar.

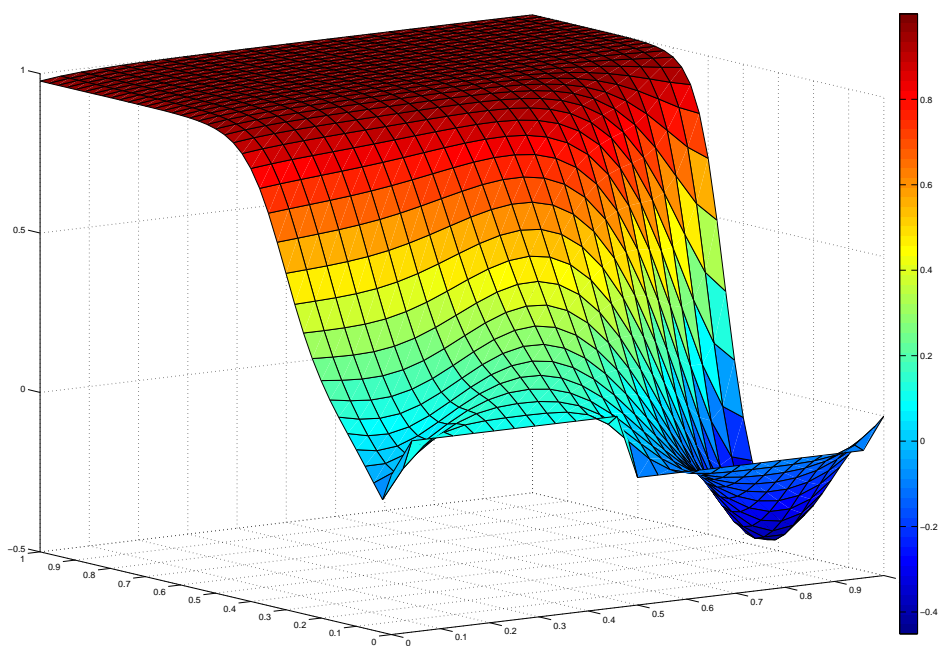


Abbildung 7.16: Neumann-RB, BC, H^1 -Norm, $N = 10$

7.2.4 Zusammenfassung der Ergebnisse

Vergleichen wir die Ergebnisse der Auswertung bezüglich der verwendeten Randbedingungen (in unserem Fall der Dirichlet- und der Neumann-Randbedingung), so lassen sich Unterschiede bei den minimalen stabilisierenden Horizonten nur aufgrund der Tatsache feststellen, dass wir im Neumann-Fall einen (beliebigen) Wert am Rand gesetzt haben. Setzen wir den Rand ebenfalls Null, so wären beide Auswertungen nahezu gleich. Wir erhalten also:

Bei Neumann-Randbedingung bewirkt eine Wertebelegung ungleich Null am Rand im Vergleich zu Dirichlet-Randbedingung eine Verschlechterung des minimalen stabilisierenden Horizonts. Des Weiteren ist eine exakte Stabilisierung nicht mehr möglich.

Betrachten wir die Unterscheidung in verteilte Steuerung und Randsteuerung, so stellen sich hier gravierende Unterschiede bezüglich des minimalen stabilisierenden Horizonts ein: Erhält man für L^2 - und H^1 -Norm bei verteilter Steuerung einen Stabilisierungshorizont von $N = 3$, so liegt dieser bei Randsteuerung mit L^2 -Norm bei $N = 11$, mit H^1 -Norm sogar bei $N = 17$. In beiden Fällen kann dieser nur durch Systemmodifikationen (z.B. Änderung des Diffusionsparameters ν) oder durch Verringerung der Gewichtung λ der Steuerung in der Zielfunktion verkleinert werden. Dies lässt sich damit erklären, dass die Einflussnahme auf das System durch Verwendung einer verteilten Steuerung auf das Gebiet "verteilt" geschieht, so dass nahezu fast jedes beliebige Verhalten dem System aufgezwungen werden kann. Dies gelingt mit einer ausschließlich an den Rändern wirkenden Steuerung nicht. Wir halten fest:

Mittels verteilter Steuerung kann (im Gegensatz zur Randsteuerung) bereits bei (viel) kleinerem Horizont eine Stabilisierung erzielt werden.

Auch die Verwendung unterschiedlicher Normen in der Zielfunktion wirkt sich entscheidend auf den minimalen Stabilisierungshorizont aus: Erhält man bei Verwendung von Dirichlet-Randbedingung und der verteilten Steuerung nur eine Unterscheidung bezüglich der Zeit, bis wann der Horizont von $N = 3$ stabilisiert (L^2 -Norm: gute Stabilisierung bei $t \approx 1.0$; H^1 -Norm: sehr gute Stabilisierung bereits bei $t \approx 0.1$), ergibt sich bei der Randsteuerung und Verwendung der L^2 -Norm ein Stabilisierungshorizont von $N = 11$, bei H^1 -Norm ein Stabilisierungshorizont von $N = 17$. Dieses Verhalten lässt sich auch auf den Fall der Neumann-Randbedingung übertragen. Zusammenfassend kann man sagen:

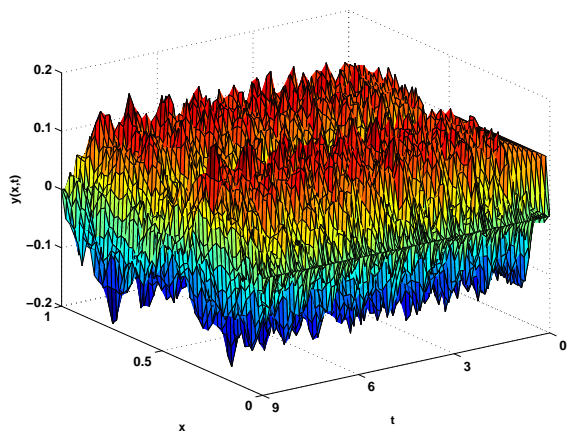
Die Verwendung der H^1 -Norm im Zielfunktional kann die Stabilisierung bei verteilter Steuerung in die Null "beschleunigen". Im Falle einer Randsteuerung allerdings verschlechtert sich der minimale Stabilisierungshorizont im Vergleich zur L^2 -Norm.

7.3 Numerische Auswertung der Wellengleichung

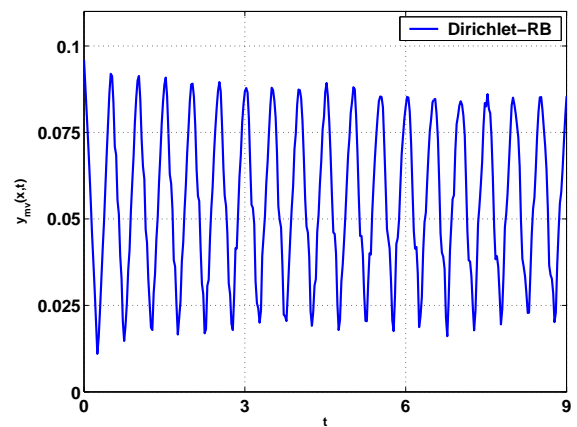
In den nachfolgenden Abschnitten wurde die *Wellengleichung*

$$y_{tt} = c y_{xx}$$

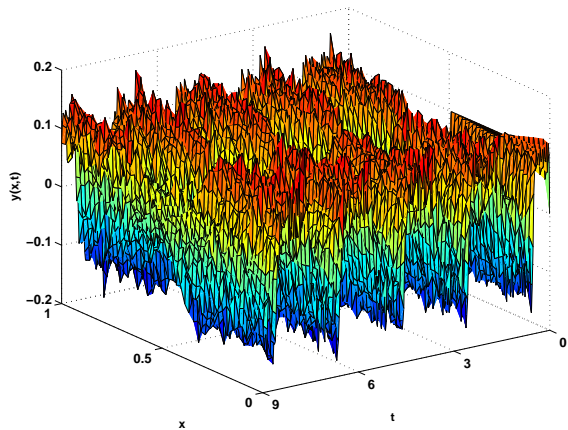
mit Wellengeschwindigkeit c unter Dirichlet- und Neumann-Randbedingungen ausgewertet und ihr Stabilisierungsverhalten sowohl für verteilte Steuerung als auch für Randsteuerung untersucht. Um deren Auswirkungen auf den Verlauf der Lösungen des Systems zu verdeutlichen, zeigt Abbildung 7.17 eine Simulation des Lösungsverhaltens im **ungesteuerten** Fall für Dirichlet- und Neumann-Randbedingung:



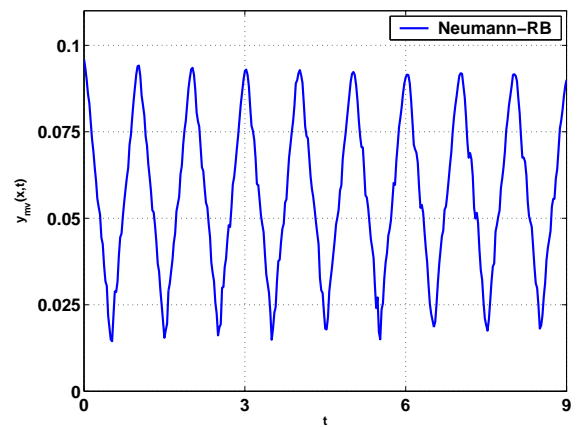
(a) Dirichlet-Randbedingung



(b) DIR: Betragsmäßig gemittelte Abweichung von Null



(c) Neumann-Randbedingung



(d) NEUM: Betragsmäßig gemittelte Abweichung von Null

Abbildung 7.17: Simulation des Lösungsverhaltens der Wellengleichung im ungesteuerten Fall

Die Unterschiede im Wellenverlauf rühren dabei von den verwendeten Randbedingungen her: Im Dirichlet-Fall wird hier gefordert, dass die Ränder stets gleich Null sind, während im Neumann-Fall am Rand eine Bedingung an die Ableitung in Richtung der äußeren

Normalen gestellt wird. Dementsprechend verhält sich auch die Wellenausbreitung unterschiedlich.

Wichtiger Hinweis:



Sämtliche zu den folgenden Auswertungen berechneten Ergebnisse können im Einzelnen auf der beigefügten Daten-CD-ROM eingesehen und mit den darauf enthaltenen MATLAB-Codes geplottet werden (siehe Anhang). Das jeweilige Verzeichnis wird unter Verwendung des CD-ROM-Symbols zu Beginn jeder Auswertung angegeben.

7.3.1 Ausgangssystem und -parameter

Zur Auswertung der Wellengleichung wurden die folgenden Belegungen der Parameterwerte verwendet:

Ausgangsparameter

| | |
|--|------------------------|
| Startprofil zum Zeitpunkt t_0 | Z-Profil |
| Startzeitpunkt t_0 | 0 |
| Angestrebtes Endprofil zum Zeitpunkt t_f | Null-Profil |
| Endzeitpunkt t_f | variiert |
| λ (Gewichtung der Steuerung u in der Zielfunktion) | 10^{-3} |
| Steuerbeschränkungen | $] - \infty, +\infty[$ |
| Zustandsbeschränkungen | $] - \infty, +\infty[$ |

Tabelle 7.3: Ausgangsparameter für die Auswertung der Wellengleichung

Wir unterscheiden im Weiteren die folgenden Varianten unter Beibehalt der obigen Ausgangsparameter:

Variante 1: "Standardbelegung"

| | |
|-----------------------------------|-------|
| Anzahl Diskretisierungspunkte M | 50 |
| Abtastperiode T | 0.025 |
| Endzeit t_f | 3.0 |
| Anzahl Zeitintervalle I | 120 |

Tabelle 7.4: Parameterbelegung in Variante 1

Variante 2: Veränderung der Abtastperiode T

| | |
|-----------------------------------|------|
| Anzahl Diskretisierungspunkte M | 50 |
| Abtastperiode T | 0.05 |
| Endzeit t_f | 3.0 |
| Anzahl Zeitintervalle I | 60 |

Tabelle 7.5: Parameterbelegung in Variante 2

Variante 3: Erhöhung der Anzahl der Diskretisierungspunkte M

| | |
|-----------------------------------|-------|
| Anzahl Diskretisierungspunkte M | 100 |
| Abtastperiode T | 0.025 |
| Endzeit t_f | 3.0 |
| Anzahl Zeitintervalle I | 120 |

Tabelle 7.6: Parameterbelegung in Variante 3

Variante 4: Verringerung der Anzahl der Diskretisierungspunkte M

| | |
|-----------------------------------|-------|
| Anzahl Diskretisierungspunkte M | 25 |
| Abtastperiode T | 0.025 |
| Endzeit t_f | 3.0 |
| Anzahl Zeitintervalle I | 120 |

Tabelle 7.7: Parameterbelegung in Variante 4

Bemerkung 7.6

Für Variante 3, die eine Verdoppelung der Anzahl der Diskretisierungspunkte des Standardsystems (Variante 1) beinhaltet, hat sich zunächst nicht die vermutete Verbesserung bezüglich der Stabilisierung in die Null ergeben (vgl. Auswertung Dirichlet-Randbedingung bei verteilter Steuerung). Da wir für diese stets die schlechtesten Ergebnisse in den Auswertungen erhalten haben, wurde in Betracht gezogen, das Lösungsverhalten des Systems für eine Halbierung der Anzahl der Diskretisierungspunkte aus Variante 1 zu untersuchen (entspricht Variante 4).

Variante 3 findet daher nur bei den Auswertungen zu Dirichlet-Randbedingung bei verteilter Steuerung Verwendung. \circ

Auf Abweichungen bzw. Veränderungen der Parameter wird gegebenenfalls in den einzelnen Auswertungen hingewiesen.

Implementation der Normen

Die in der Zielfunktion auftretenden Normen wurden folgendermaßen implementiert:

- Normen im Inneren:

$$\|y(i, \cdot)\|_{L^2(\Omega)}^2 := \frac{1}{M-2} \sum_{j=2}^{M-1} y(i, j)^2 \quad (7.6)$$

$$\|y_x(i, \cdot)\|_{L^2(\Omega)}^2 := \frac{1}{M-2} \sum_{j=2}^{M-1} \left(\frac{y(i+1, j) - y(i, j)}{h} \right)^2 \quad (7.7)$$

$$\|u(i, \cdot)\|_{L^2(\Omega)}^2 := \frac{1}{M-2} \sum_{j=2}^{M-1} u(i, j)^2 \quad (7.8)$$

- Norm auf dem Rand:

$$\|u(i, \cdot)\|_{L^2(\Gamma)}^2 := \frac{1}{2} \sum_{j=1}^2 u(i, j)^2 \quad (7.9)$$

7.3.2 Vergleich der SQP-Verfahren

Die in den Auswertungen verwendeten und zu Beginn bereits erwähnten SQP-Verfahren zur Lösung der Optimalsteuerungsproblematik werden von uns bezüglich ihrer Performanz und Leistungsfähigkeit untersucht und verglichen. Als Hauptkriterien werden dabei die Auswirkungen auf

- den zeitlichen (graphischen) Verlauf der Systemlösungen,
- die entstandenen Kosten entlang der MPC-closed-loop Lösung für eine Auswertung und
- die für eine Auswertung benötigte Rechenzeit

betrachtet.

Als Grundlage für den Vergleich wählen wir die Dirichlet-Randbedingung bei verteilter Steuerung mit L^2 - als auch H^1 -Norm im Zielfunktional und führen diesen auf Basis gleicher Horizonte N durch.

7.3.2.1 Vergleich anhand des zeitlichen Verlaufs

Wir vergleichen als erstes die beiden Verfahren bezüglich ihrer Auswirkung auf den zeitlichen Verlauf der Lösungen des Systems. Dies wird mittels Vergleichs der durch die beiden Verfahren erhaltenen Plotdaten für verschiedene Horizonte N durchgeführt, indem wir zum einen den jeweiligen graphischen Verlauf, zum anderen den betragsmäßig gemittelten Wert $y_{mv}(x, t)$ der Abweichung von der Null

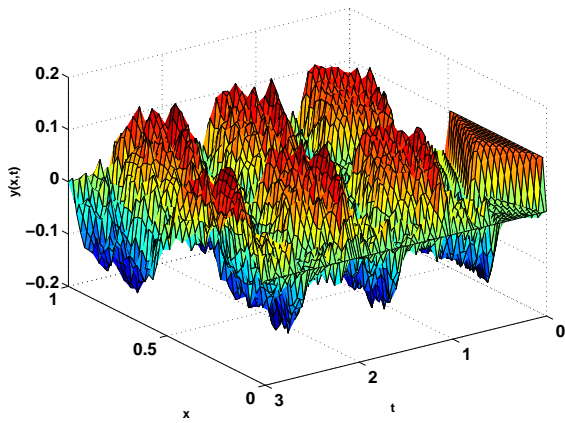
$$y_{mv}(x, \cdot) := \frac{1}{M} \sum_{i=1}^M |y(x_i, \cdot)|$$

zu Abtastzeitpunkten t gegenüberstellen.

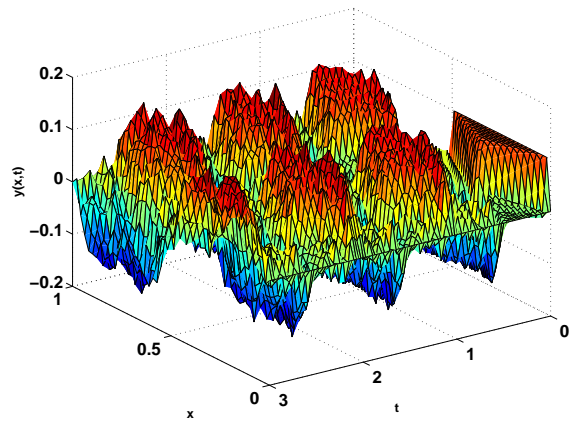
Beobachtungen

Wir erhalten beim Vergleich von NAGC mit SQPF für sämtliche Auswertungen nur minimale Unterschiede in Bezug auf die Abweichung von der Null. Dies wird besonders deutlich bei Gegenüberstellung der gemittelten Werte der Abweichung von Null: Die Graphen verlaufen zu Beginn nahezu identisch, wobei sich für SQPF im fortgeschrittenen zeitlichen Verlauf kleinere Abweichungen von der Null erkennen lassen (siehe Abbildungen 7.18, 7.19, 7.20 und 7.21).

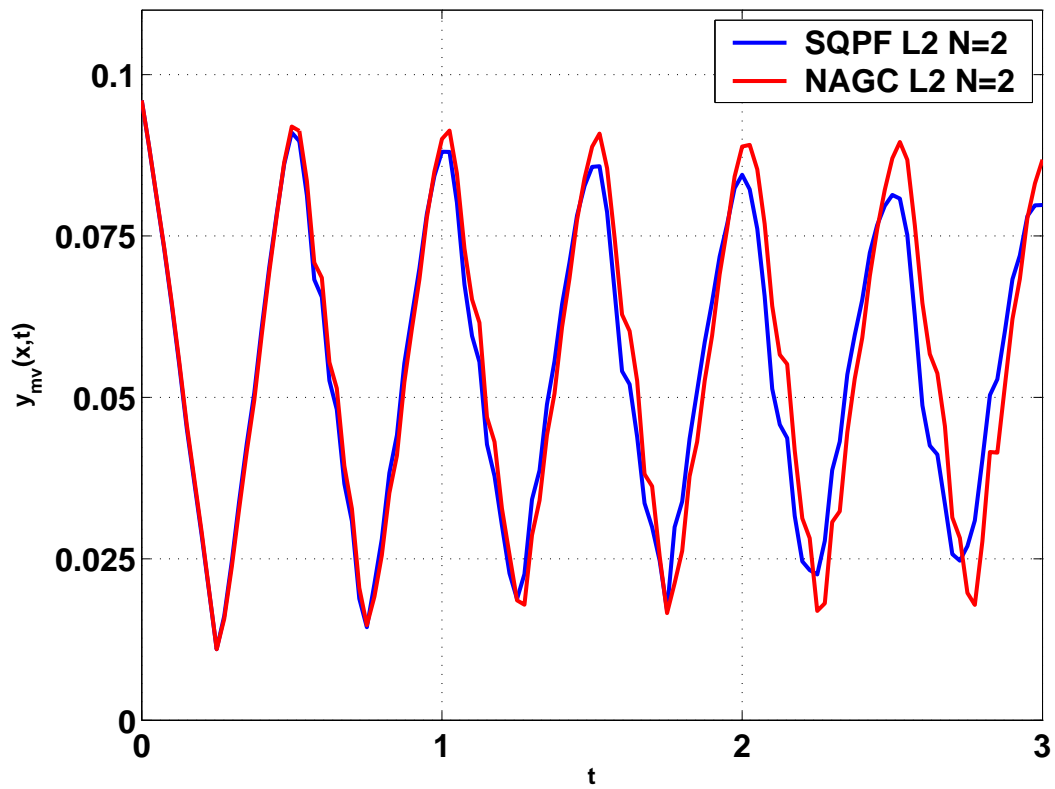
Stellt sich im Falle der H^1 -Norm im Zielfunktional für einen Horizont N eine Stabilisierung ein, so sind keine Unterschiede mehr erkennbar (siehe Abbildungen 7.22 und 7.23).



(a) SQPF

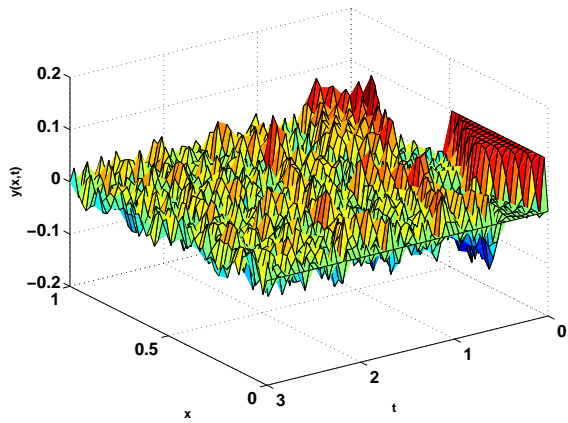


(b) NAGC

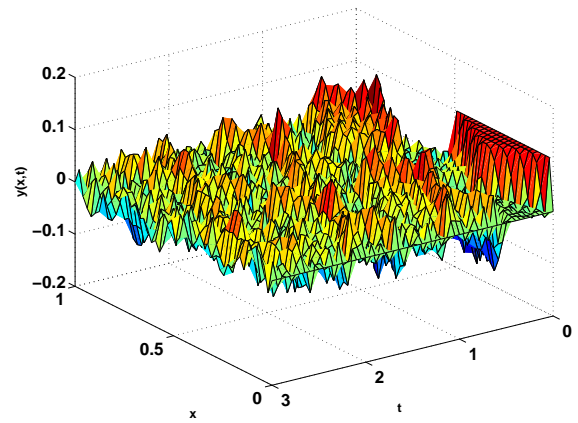


(c) Vergleich SQPF und NAGC

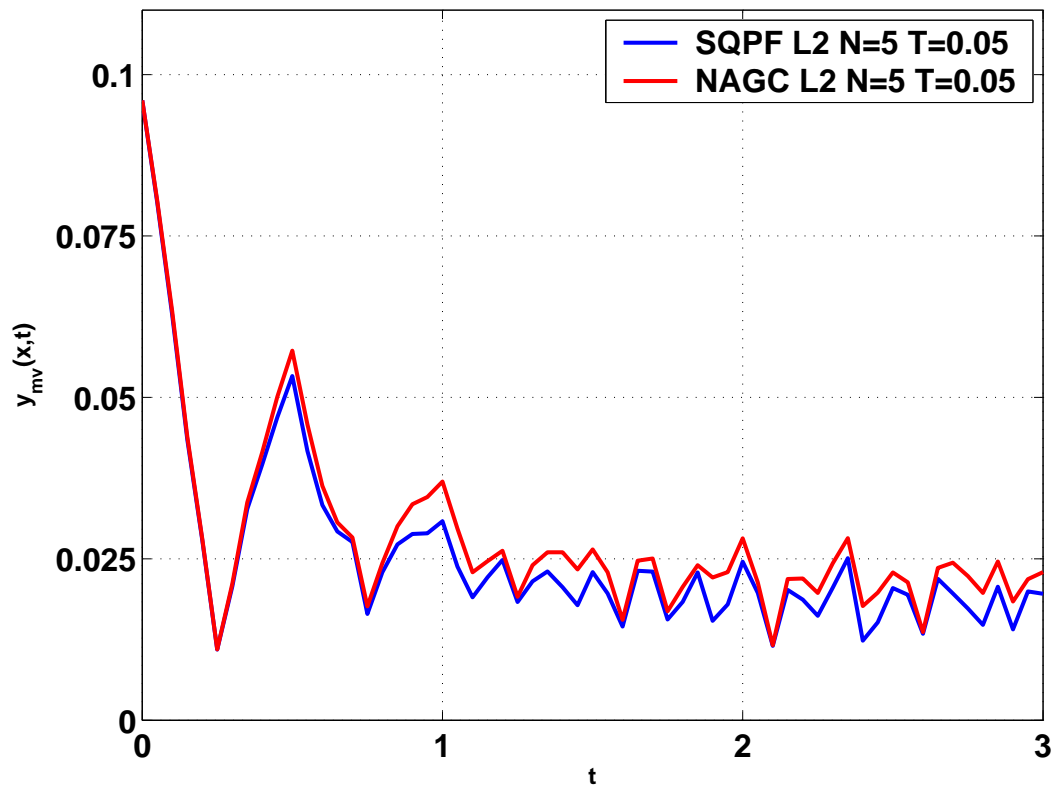
Abbildung 7.18: Dirichlet-RB, DC, L^2 -Norm, Variante 1, $N = 2$



(a) SQPF



(b) NAGC



(c) Vergleich SQPF und NAGC

Abbildung 7.19: Dirichlet-RB, DC, L^2 -Norm, Variante 2, $N = 5$

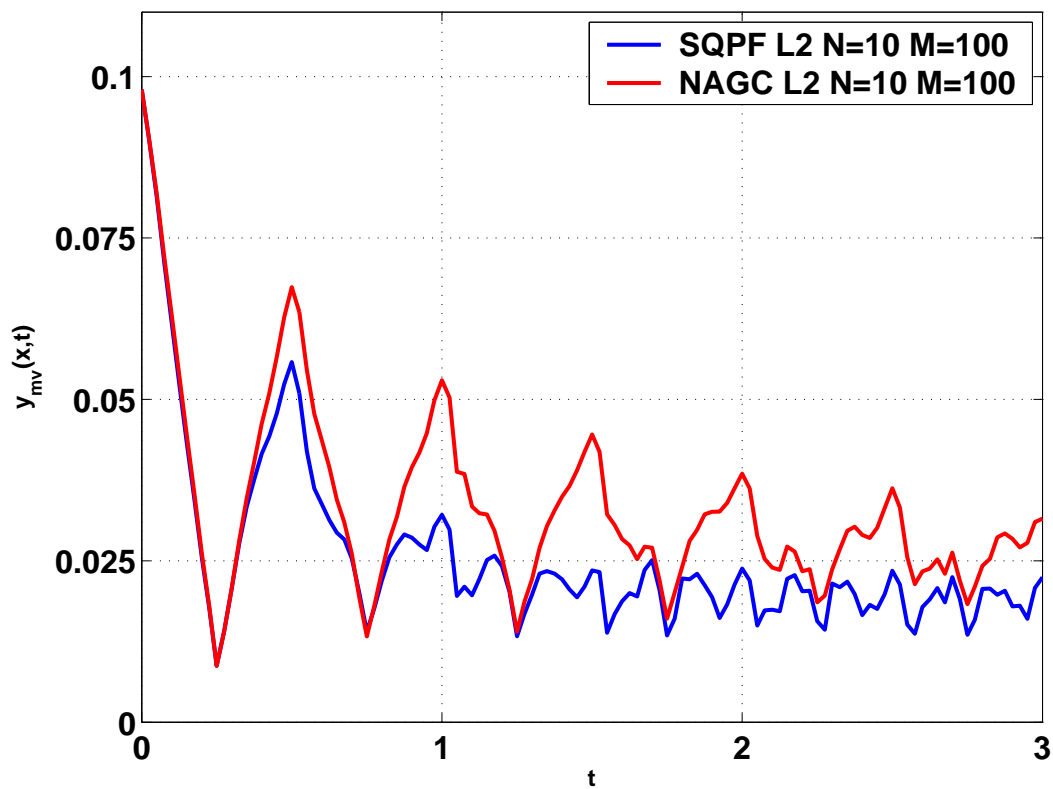
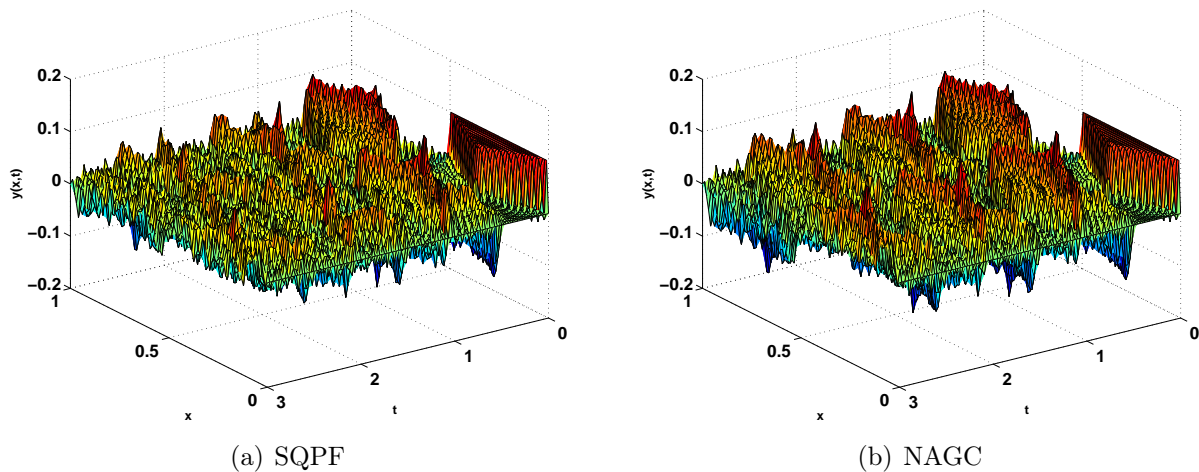
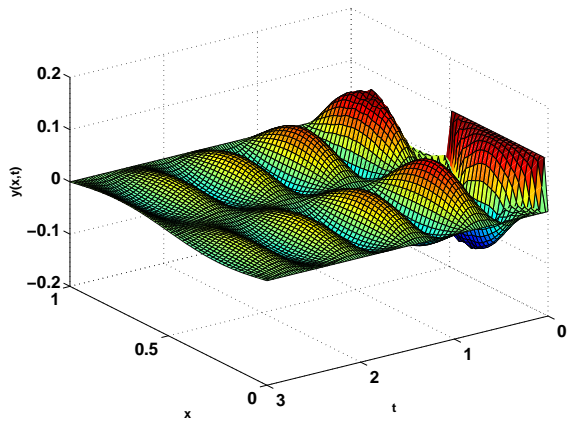
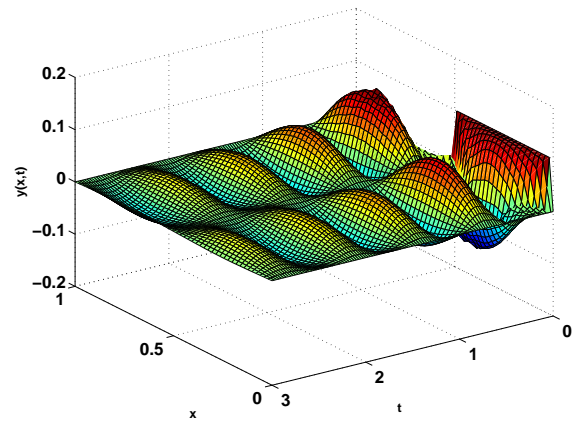


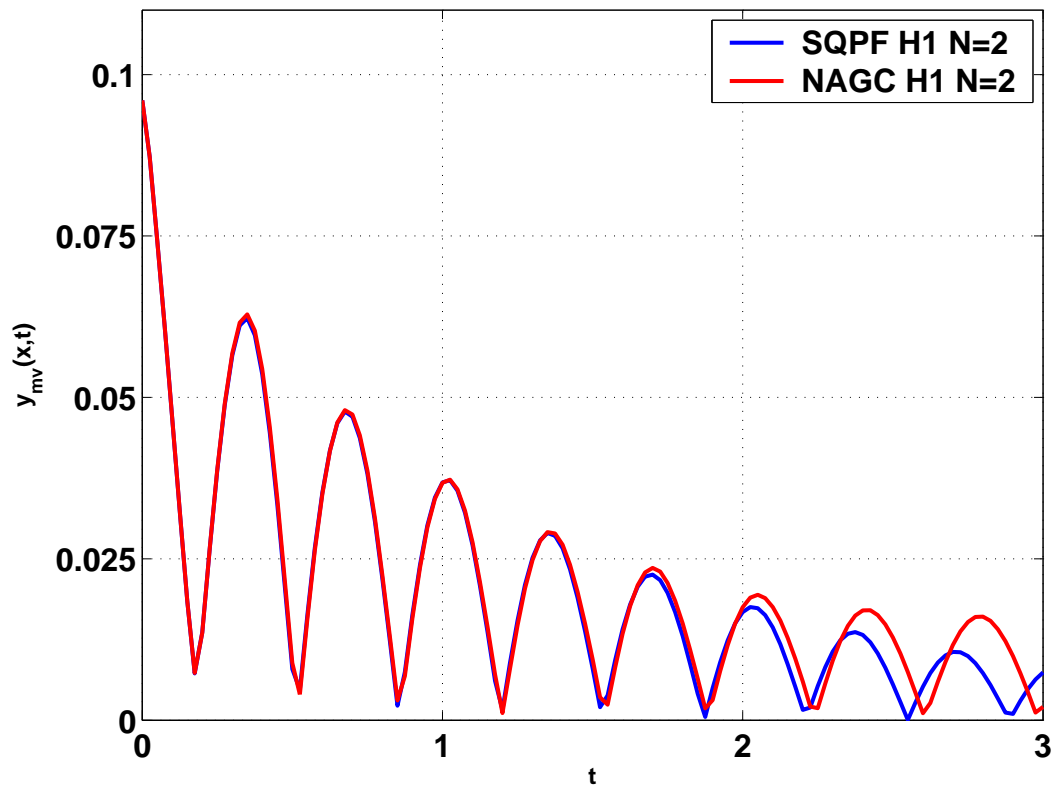
Abbildung 7.20: Dirichlet-RB, DC, L^2 -Norm, Variante 3, $N = 10$



(a) SQPF

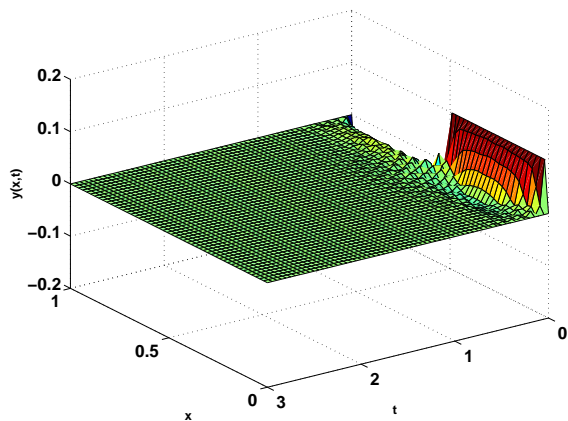


(b) NAGC

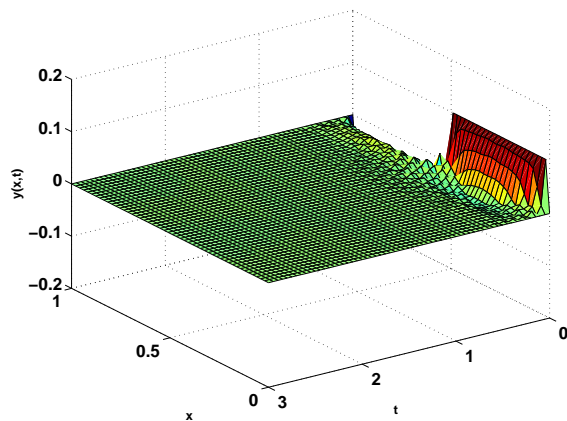


(c) Vergleich SQPF und NAGC

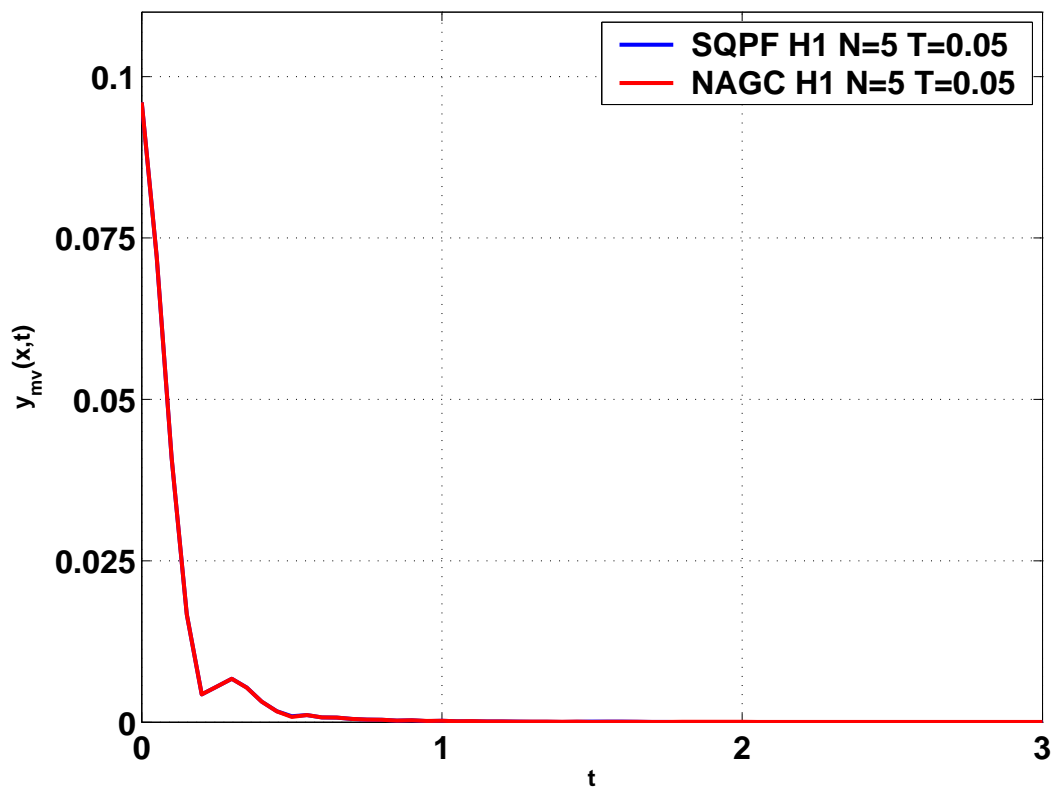
Abbildung 7.21: Dirichlet-RB, DC, H^1 -Norm, Variante 1, $N = 2$



(a) SQPF

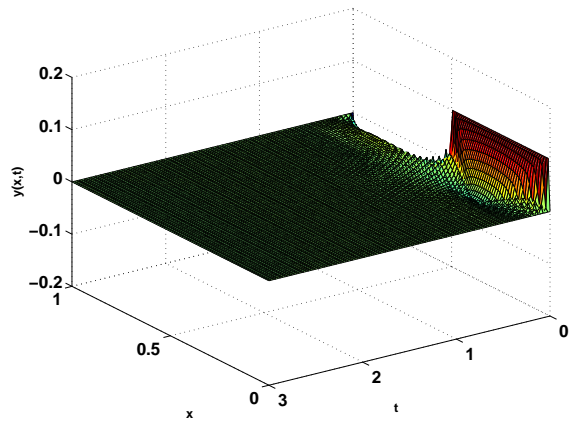


(b) NAGC

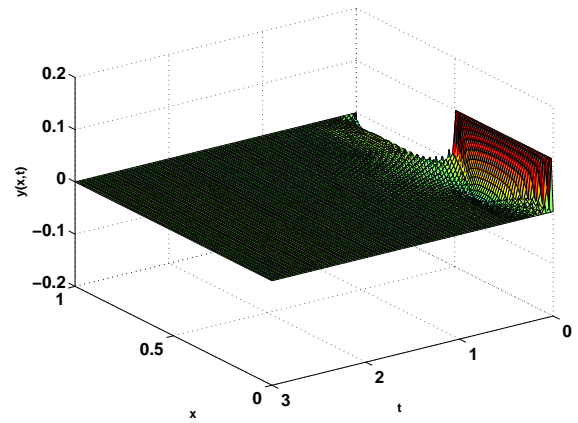


(c) Vergleich SQPF und NAGC

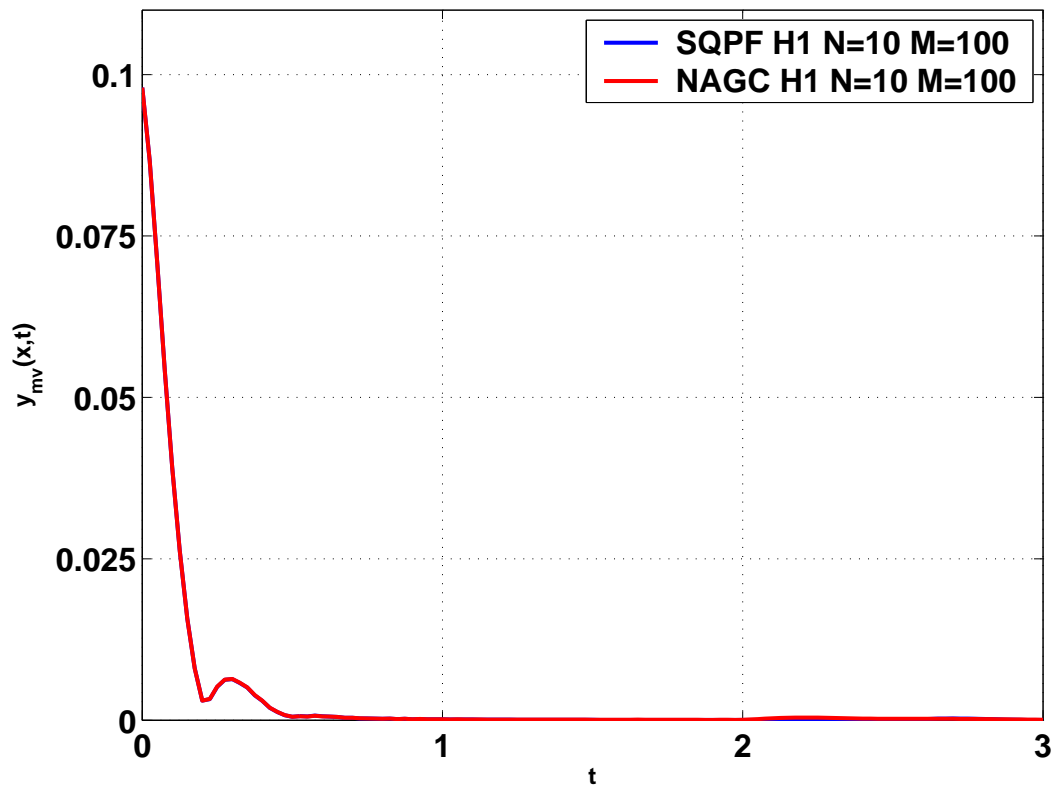
Abbildung 7.22: Dirichlet-RB, DC, H^1 -Norm, Variante 2, $N = 5$



(a) SQPF



(b) NAGC



(c) Vergleich SQPF und NAGC

Abbildung 7.23: Dirichlet-RB, DC, H^1 -Norm, Variante 3, $N = 10$

7.3.2.2 Vergleich anhand der Kosten

Wir vergleichen nun die Kosten K entlang der MPC-closed-loop Lösung, verursacht durch die Abweichung y von der Null, die Ableitung y_x und die Steuerung u , aufsummiert über

$$I := \frac{t_f}{T}, \quad (7.10)$$

der Anzahl der Zeitintervalle bis zum Endzeitpunkt t_f . Diese berechnen sich für die L^2 -Norm folgendermaßen:

$$K = \sum_{i=1}^I (\|y(\cdot, iT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, iT)\|_{L^2(\Omega)}^2) \quad (7.11)$$

Für die H^1 -Norm gilt entsprechend:

$$K = \sum_{i=1}^I (\|y(\cdot, iT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, iT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, iT)\|_{L^2(\Omega)}^2) \quad (7.12)$$

Wir vergleichen die beiden Verfahren anhand der *durchschnittlichen Kosten pro Zeitintervall* T , kurz K_T , für die gilt:

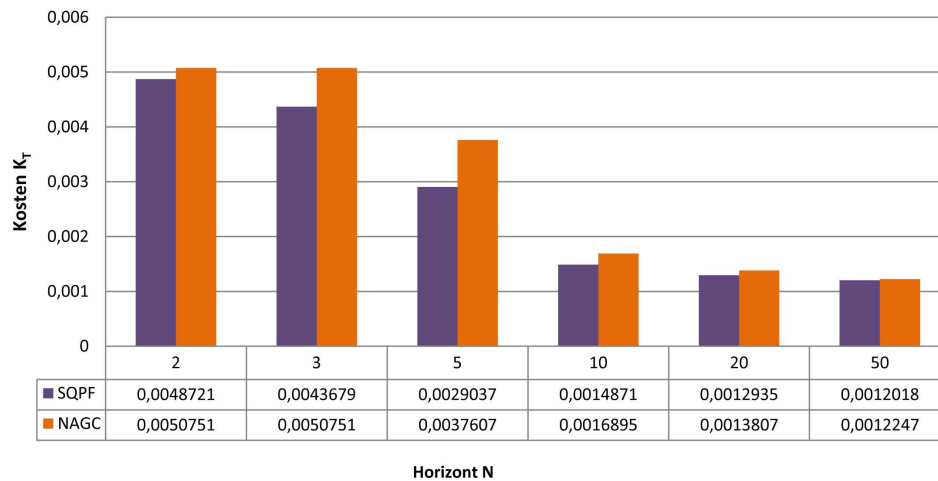
$$K_T := \frac{K + K_{\text{Startprofil}}}{I} \quad (7.13)$$

Dabei bezeichnet $K_{\text{Startprofil}}$ die *Kosten für das Startprofil*, die sich nur aus der Abweichung y von der Null gemäß dem Startprofil und dem daraus resultierenden Wert der Ableitung y_x zusammensetzen.

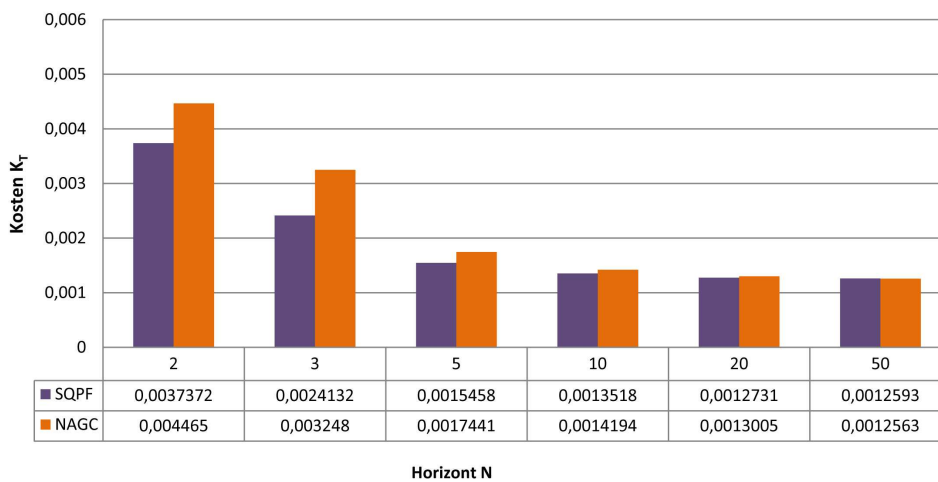
Beobachtungen

Die durchschnittlichen Kosten pro Zeitintervall weisen für die L^2 -Norm kaum Unterschiede im Vergleich von SQPF und NAGC auf: Diese machen sich für kleine Horizonte ($N < 10$) in der dritten Nachkommastelle bemerkbar, nehmen aber mit größerem Horizont ($N \geq 10$) weiter ab (siehe Abbildung 7.24).

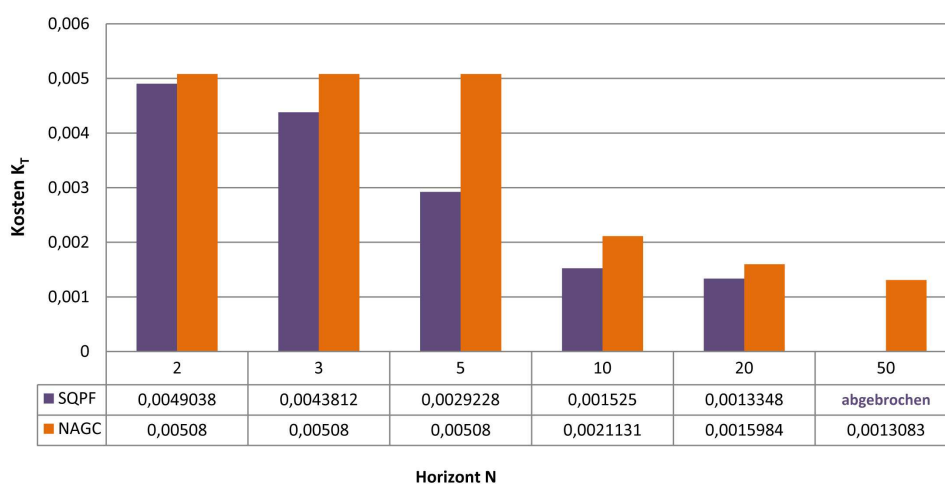
Für den Fall der H^1 -Norm erhalten wir nahezu identische Kosten (siehe Abbildung 7.25).



(a) Variante 1

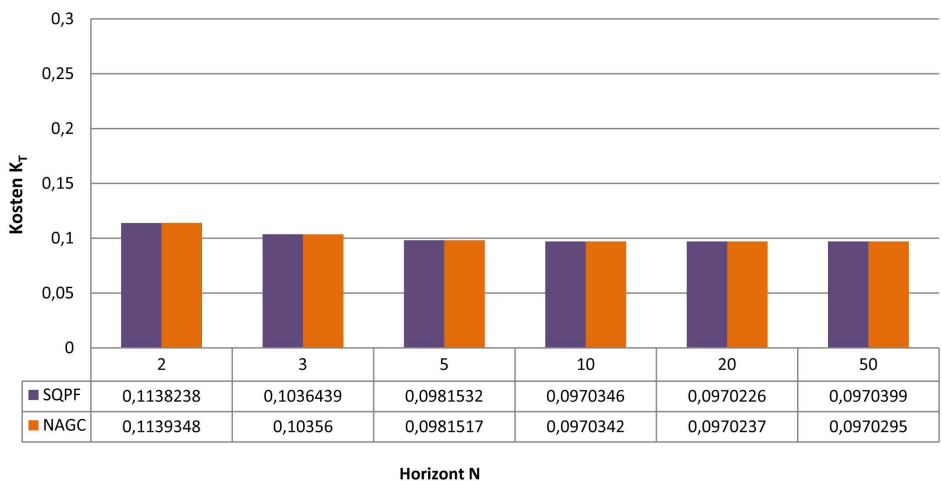


(b) Variante 2

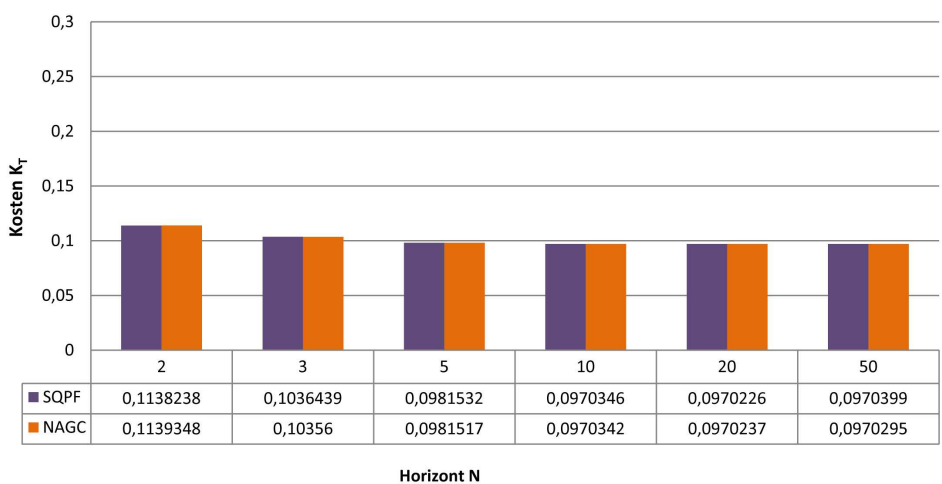


(c) Variante 3

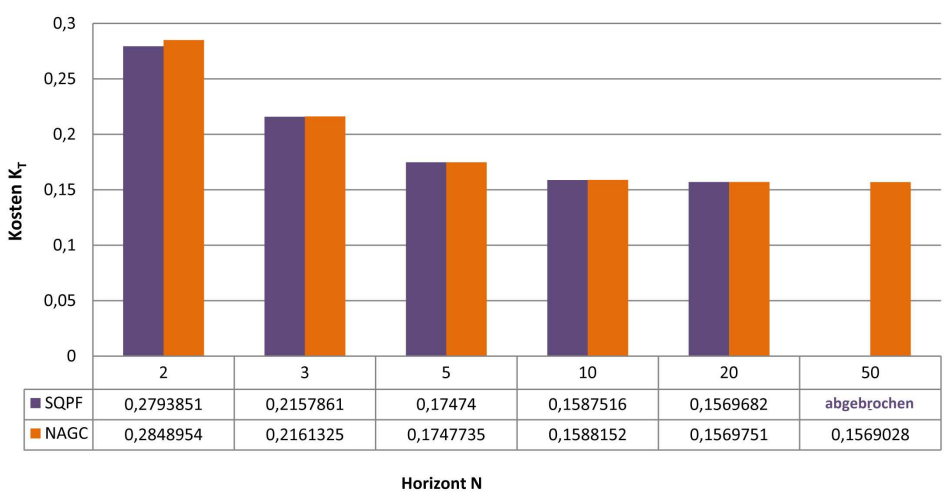
Abbildung 7.24: Kostenvergleich bei Dirichlet-RB, DC, L^2 -Norm



(a) Variante 1



(b) Variante 2



(c) Variante 3

Abbildung 7.25: Kostenvergleich bei Dirichlet-RB, DC, H^1 -Norm

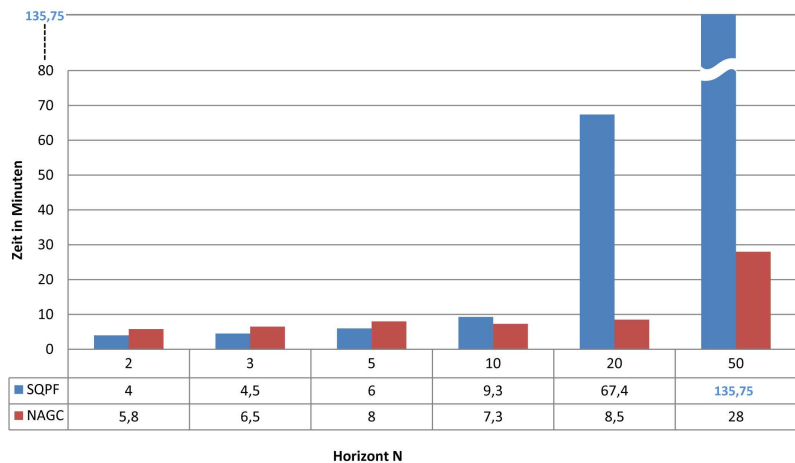
7.3.2.3 Vergleich anhand der Rechenzeit

Abschließend vergleichen wir noch die beiden Verfahren bezüglich ihrer für eine Auswertung aufgewendeten *Rechenzeiten*. Dabei sei darauf hingewiesen, dass es sich hierbei nicht um die eigentliche CPU-Zeit handelt, die der Prozessor ausschließlich zur Berechnung der für die Auswertung verwendeten Routinen und Befehle benötigt. Es handelt sich vielmehr um die Systemzeit, die aufgrund systemabhängiger Hintergrundprozesse variiert. Dies wurde aber toleriert, zumal die verglichenen Auswertungen zeitgleich und unter gleichen Voraussetzungen (d.h. Vermeidung der Beanspruchung des Doppelkern-Prozessors durch weitere rechenintensive Programme) gestartet wurden. Man erhält sozusagen eine zeitliche **Tendenz**, die für den Vergleich der beiden Verfahren für unsere Zwecke ausreichend ist.

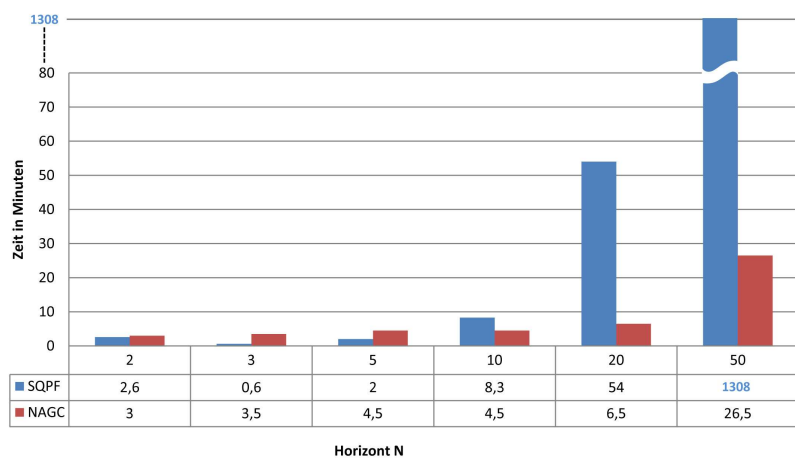
Beobachtungen

Für kleine Horizonte $N \leq 5$ ist SQPF das schnellere Verfahren, wo hingegen sich ab einem Horizont $N > 5$ das NAGC-Verfahren als das Schnellere herausstellt. Der zeitliche Unterschied nimmt dabei drastisch zu und geht sogar so weit, dass wir nur mit dem NAGC-Verfahren für alle von uns gewählten Horizonte eine Lösung erhalten (Abbruch bei SQPF aufgrund einer Rechendauer von über 7 Tagen), insbesondere für Auswertungen mit hoher Anzahl an örtlichen Diskretisierungspunkten (siehe dazu die Abbildungen 7.26 und 7.27).

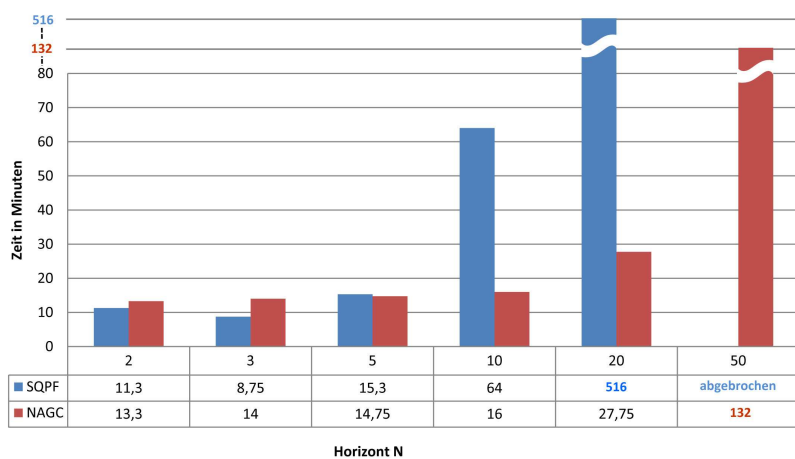
7 Auswertung und numerische Ergebnisse



(a) Variante 1

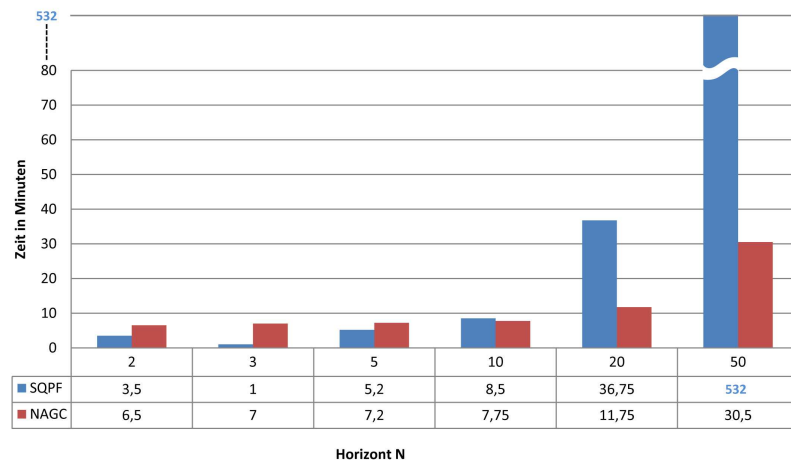


(b) Variante 2

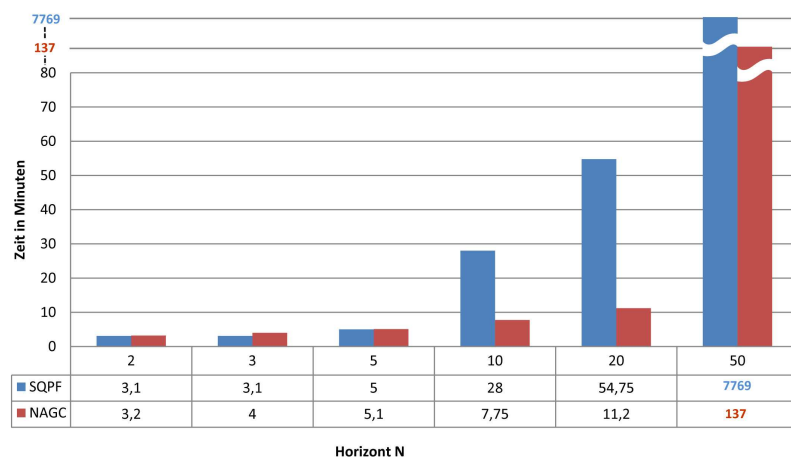


(c) Variante 3

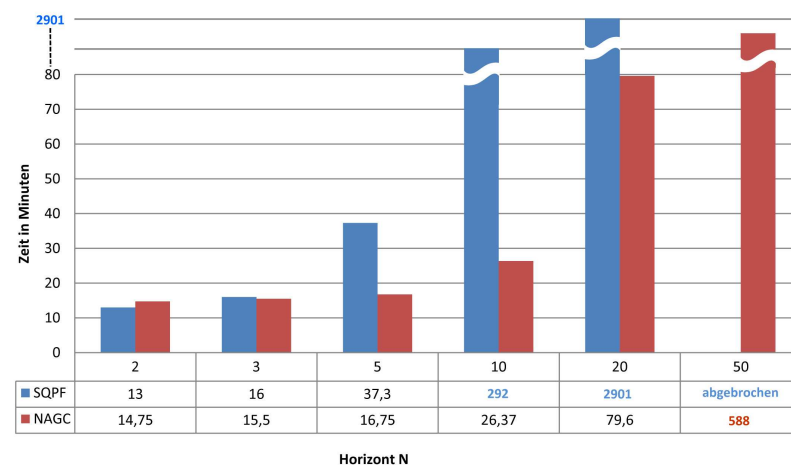
Abbildung 7.26: Rechenzeitvergleich bei Dirichlet-RB, DC, L^2 -Norm



(a) Variante 1



(b) Variante 2



(c) Variante 3

Abbildung 7.27: Rechenzeitvergleich bei Dirichlet-RB, DC, H^1 -Norm

7.3.2.4 Zusammenfassung der Ergebnisse

Die Wahl eines geeigneten Verfahrens gestaltet sich sehr subjektiv bzw. von gegebenen Anforderungen oder Beschränkungen abhängig.

Will man den Verlauf der Systemlösungen für Horizonte bis $N \leq 10$ unter dem Aspekt möglichst kleiner Kosten und “annehmbarer” Rechenzeit (in unserem Fall ≤ 1 Tag) untersuchen, so stellt sich SQPF als sinnvolles Verfahren heraus. NAGC eignet sich dabei vornehmlich für eine Untersuchung von großen Horizonten $N > 20$ und hoher Anzahl an örtlichen Diskretisierungspunkten. Wir halten fest:

Da für uns der Erhalt eines minimalen stabilisierenden Horizonts unter möglichst geringen Kosten und unter einer festgesetzten oberen Grenze der Anzahl der räumlichen Diskretisierungspunkte ($M = 50$, ausgenommen Variante 3 mit $M = 100$) an erster Stelle steht, ist eine Entscheidung für das SQPF-Verfahren sinnvoll.

Bemerkung 7.7

Aufgrund der gemachten Feststellung weisen wir darauf hin, dass **die nachfolgenden Auswertungen ausschließlich unter Verwendung des SQPF-Verfahrens** durchgeführt werden. ○

7.3.3 Vergleich der beiden Normen H^1 und L^2

In diesem Abschnitt widmen wir uns dem Vergleich der beiden in unserem Zielfunktional verwendeten Normen H^1 und L^2 bezüglich ihres Stabilisierungsverhaltens für jeweils Dirichlet- und Neumann-Randbedingung und unterscheiden hierbei in verteilte Steuerung und Randsteuerung. Als Hauptkriterien wurden hierzu

- die Gegenüberstellung des jeweiligen zeitlichen Verlaufs der Lösungen des Systems,
- die betragsmäßig gemittelte örtliche Abweichung $y_{mv}(x, t)$ von der Null

$$y_{mv}(x, \cdot) := \frac{1}{M} \sum_{i=1}^M |y(x_i, \cdot)| \quad (7.14)$$

an jedem Abtastzeitpunkt t und

- der Vergleich der an den Diskretisierungspunkten anliegenden Steuerungen
 - anhand der betragsmäßig gemittelten verteilten Steuerung

$$u_{mv}(x, \cdot) := \frac{1}{M-2} \sum_{i=2}^{M-1} |u(x_i, \cdot)| \quad (7.15)$$

an jedem Abtastzeitpunkt t ,

beziehungsweise

- der graphischen Gegenüberstellung der beiden jeweiligen Randsteuerungen

betrachtet.

Bemerkung 7.8 (Hinweis zur gemittelten Steuerung)

Wegen der bei uns verwendeten großen Anzahl an verteilten Steuerungen ($M \geq 23$) haben wir uns aufgrund fehlender Übersichtlichkeit dazu entschlossen, nicht den zeitlichen Verlauf jeder Steuerung im Einzelnen darzustellen, sondern das betragsmäßige Mittel darüber zu bilden und dieses zu plotten. Zur graphischen Darstellung aller Steuerungen verweisen wir auf folgenden MATLAB-Code auf der beiliegenden Daten-CD-ROM:



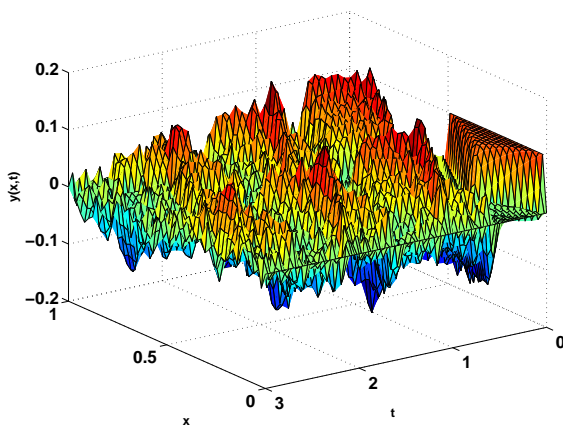
MATLAB-Codes/WEQ_U_1DPlot.m

7.3.3.1 Dirichlet-Randbedingung mit verteilter Steuerung

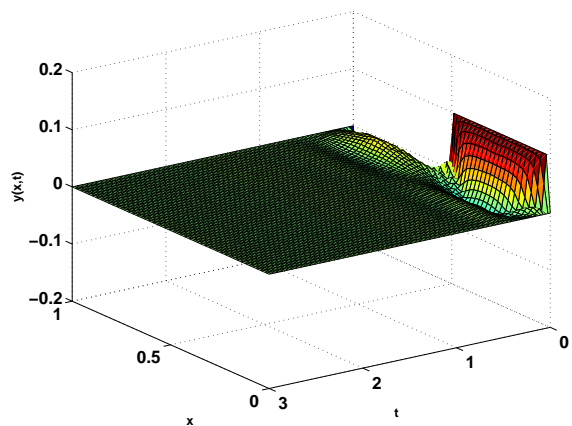
Beobachtungen

Bei Dirichlet-Randbedingung mit verteilter Steuerung erhalten wir bei Verwendung der H^1 -Norm im Zielfunktional für alle von uns ausgewerteten Horizonte $N = \{2, 3, 5, 10, 20, 50\}$ im Mittel spätestens ab Zeitpunkt $t = 0.75$ stets kleinere Abweichungen von der Null als bei der L^2 -Norm. Dieses Verhalten können wir bei den Varianten 1, 2 und 3 beobachten.

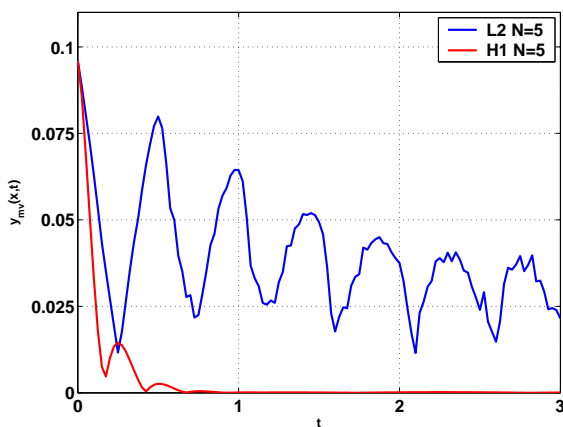
Außerdem stabilisiert die H^1 -Norm teilweise bereits für solche Horizonte N , bei denen die L^2 -Norm maximal eine Tendenz zur Stabilisierung in die Null erkennen lässt, wie wir auch an dem Vergleich der gemittelten Steuerungen sehen können (vergleiche hierzu Abbildung 7.28 für $N = 5$).



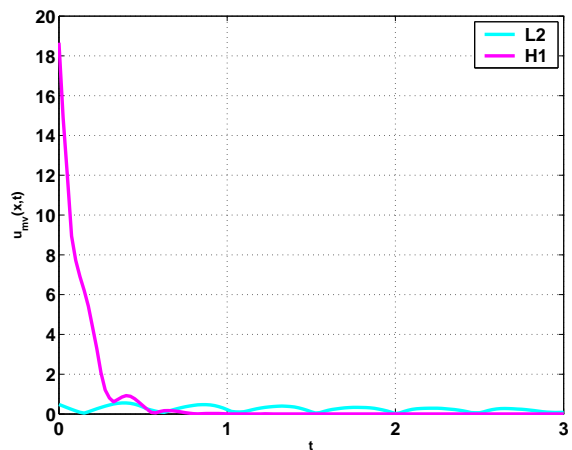
(a) L^2 -Norm



(b) H^1 -Norm



(c) Vergleich der betragsmäßig gemittelten örtlichen Abweichung von der 0



(d) Vergleich der betragsmäßig gemittelten Steuerungen

Abbildung 7.28: Normenvergleich: Dirichlet-RB, DC, $N = 5$, Variante 1

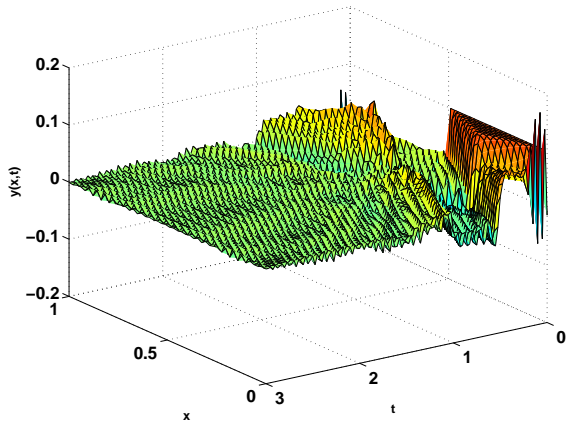
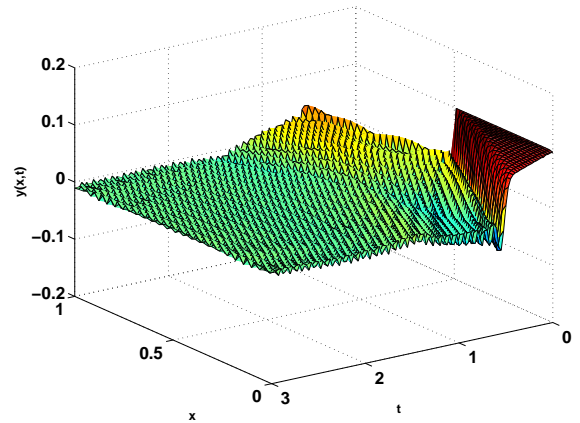
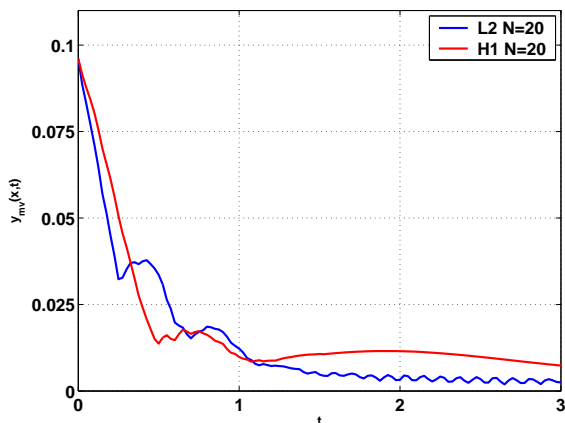
7.3.3.2 Dirichlet-Randbedingung mit Randsteuerung

Beobachtungen

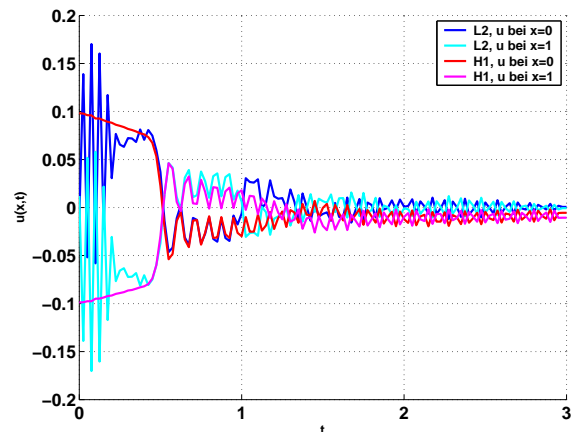
Bei Dirichlet-Randbedingung mit Randsteuerung erhalten wir bei Verwendung sowohl der L^2 -Norm als auch der H^1 -Norm im Zielfunktional für die von uns ausgewerteten Horizonte $N \geq 10$ bei den Varianten 1, 2 und 4 einen Verlauf der Lösungen in eine Umgebung $\epsilon = \pm 0.02$ der Null. Variante 3 wurde nicht ausgewertet (vgl. Bemerkung 7.6). Des Weiteren ergibt sich bei beiden Normen zu Beginn noch unterschiedliches und gegen Ende hin fast identisches Steuerverhalten (vergleiche Abbildung 7.29(d)).

Im Gegensatz zur verteilten Steuerung erhalten wir bei Randsteuerung für die L^2 -Norm bei den Horizonten $N \geq 10$ gegen Endzeit $t_f = 3.0$ im Mittel kleinere Abweichungen von der Null im Vergleich zur H^1 -Norm (vergleiche Abbildung 7.29(c)).

Für kleine Horizonte ist der Verlauf der Systemlösungen durch Wellen mit hoher Wellenfrequenz und großen Ausschlägen gekennzeichnet, der nur bedingt eine Tendenz zur Stabilisierung erkennen lässt.

(a) L^2 -Norm(b) H^1 -Norm

(c) Vergleich der betragsmäßig gemittelten örtlichen Abweichung von der 0



(d) Vergleich der anliegenden Steuerungen

Abbildung 7.29: Normenvergleich: Dirichlet-RB, BC, $N = 20$, Variante 1

7.3.3.3 Neumann-Randbedingung mit verteilter Steuerung

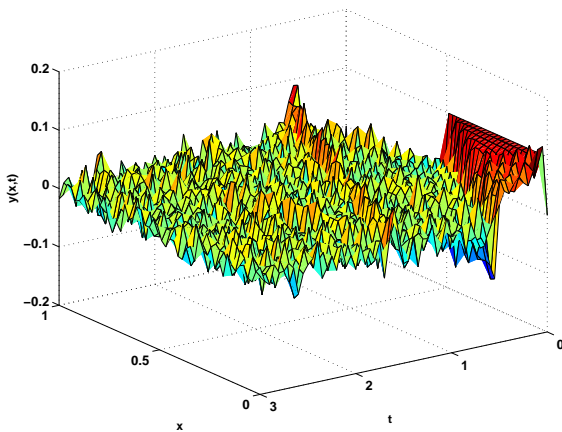
Bemerkung 7.9

Die folgenden Beobachtungen gelten für die von uns untersuchten Fälle mit Neumann-Randbedingung an beiden Rändern sowie Neumann-Bedingung am linken Rand $x_1 = 0$ und Dirichlet-Bedingung am rechten Rand $x_M = 1$. ◦

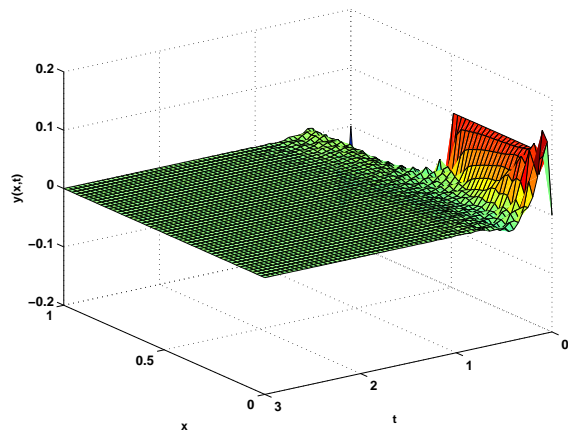
Beobachtungen

Bei Neumann-Randbedingung mit verteilter Steuerung erhalten wir analog zur Dirichlet-Randbedingung im Mittel stets kleinere Abweichungen von der Null als im Vergleich zur L^2 -Norm. Dieses Verhalten können wir bei den Varianten 1, 2 und 4 beobachten.

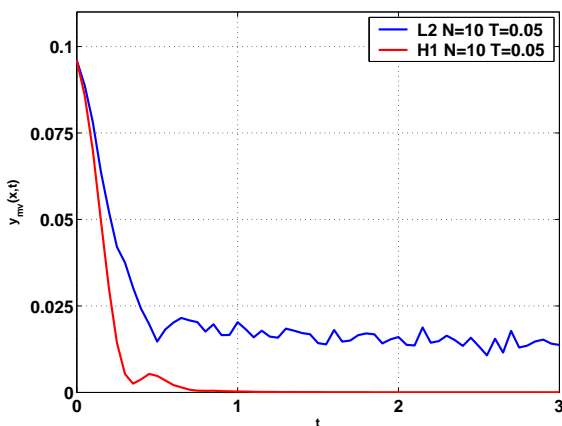
Für weitere Beobachtungen verweisen wir auf den Fall der Dirichlet-Randbedingung.



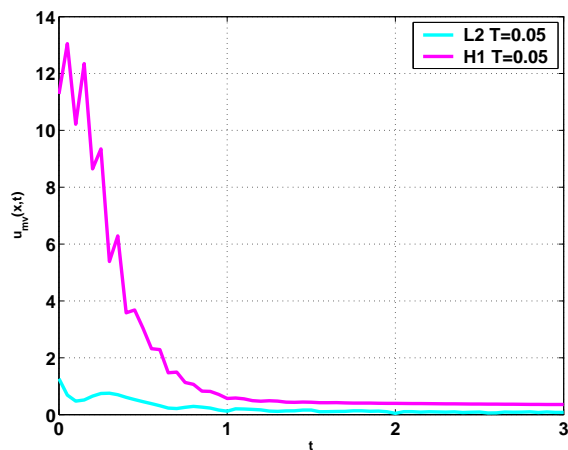
(a) L^2 -Norm



(b) H^1 -Norm



(c) Vergleich der betragsmäßig gemittelten örtlichen Abweichung von der 0



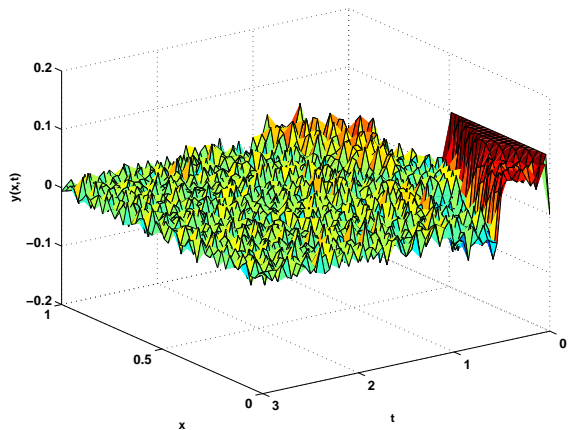
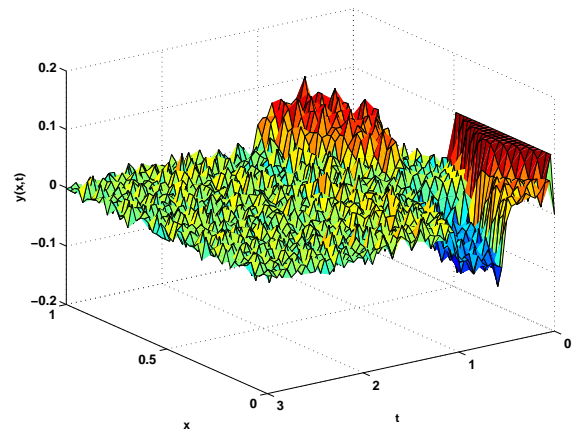
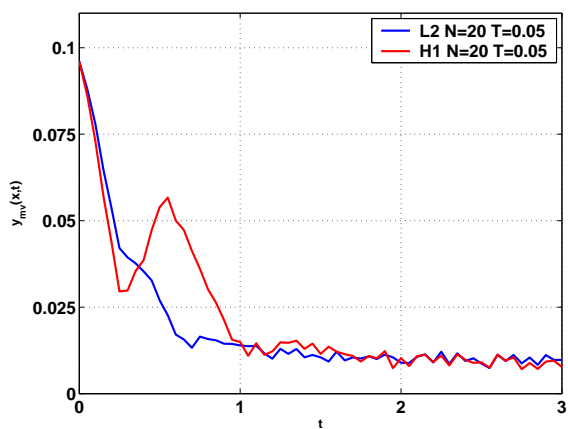
(d) Vergleich der betragsmäßig gemittelten Steuerungen

Abbildung 7.30: Normenvergleich: Neumann-RB, DC, $N = 10$, Variante 2

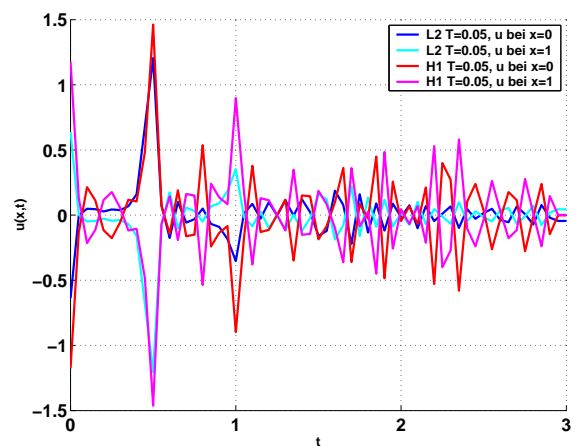
7.3.3.4 Neumann-Randbedingung mit Randsteuerung

Beobachtungen

Die Auswertung für Neumann-Randbedingung mit Randsteuerung verhält sich in ihren Resultaten analog zu der für Dirichlet-Randbedingung, allerdings erst ab einem Horizont von $N = 20$. Auch hier zeigt sich bei beiden Normen gegen Ende fast identisches Steuerungsverhalten (vergleiche Abbildung 7.31(d)), wobei sich wiederum die L^2 -Norm im Vergleich zur H^1 -Norm als die mit "besserem" Stabilisierungsverhalten herausstellt.

(a) L^2 -Norm(b) H^1 -Norm

(c) Vergleich der betragsmäßig gemittelten örtlichen Abweichung von der 0



(d) Vergleich der anliegenden Steuerungen

Abbildung 7.31: Normenvergleich: Neumann-RB, BC, $N = 20$, Variante 2

7.3.3.5 Zusammenfassung der Ergebnisse

Unterscheidet man nur nach Dirichlet und Neumann-Randbedingung, so lässt sich keine eindeutige Aussage darüber treffen, welche der beiden Normen L^2 oder H^1 für eine Stabilisierung geeigneter ist. Vielmehr muss diesbezüglich zwischen einer verteilten und einer Randsteuerung unterschieden werden.

Somit wollen wir Folgendes festhalten:

Sowohl für Dirichlet- als auch Neumann-Randbedingung erzielt die L^2 -Norm bei Verwendung der Randsteuerung ein besseres Stabilisierungsverhalten, wohingegen die H^1 -Norm im Falle der verteilten Steuerung dieses Verhalten leistet.

Bemerkung 7.10

Trotz dieser Feststellung werden die folgenden Auswertungen sowohl für die L^2 - als auch die H^1 -Norm bei verteilter Steuerung und Randsteuerung durchgeführt. ◦

7.3.4 Angabe minimaler stabilisierender Horizonte

Ziel dieser Auswertung ist die **Angabe minimaler Horizonte N** , für die eine **Stabilisierung der Wellengleichung in das Null-Profil** gelingt.

Dabei unterscheiden wir wieder bezüglich der Randbedingung (Dirichlet oder Neumann), der verwendeten Steuerung (verteilte Steuerung oder Randsteuerung) und der verwendeten Norm im Zielfunktional (L^2 - oder H^1 -Norm) und betrachten dazu standardmäßig vornehmlich die folgenden Horizonte $N = 2, 3, 5, 10, 20$ und 50 .

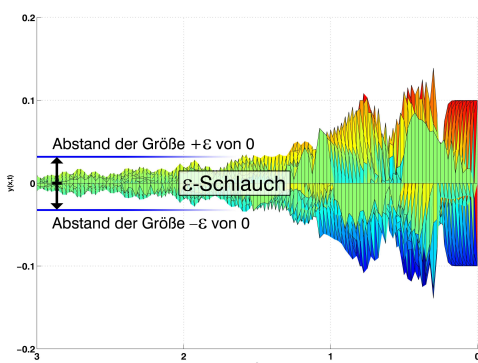
Des Weiteren differenzieren wir auch bezüglich der **Qualität** der erreichten Stabilisierung:

(numerisch exakte) Stabilisierung in das Null-Profil: Erhalten wir in den Auswertungen einen Lösungsverlauf des Systems, für den die durch das von uns verwendete Programm berechnete Abweichung vom Null-Profil einen Wert gleich Null (d.h. bis zu einer Genauigkeit von ungefähr 10^{-6}) hat, so nennen wir das eine (numerische exakte) Stabilisierung in das Null-Profil bzw. in die Null.

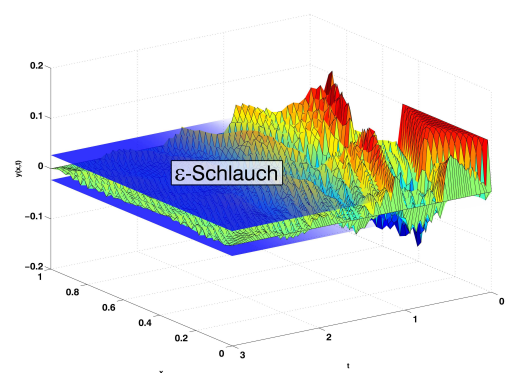
praktische Stabilisierung: Gelingt eine (numerisch exakte) Stabilisierung in die Null nicht, sondern hält sich der Verlauf der Lösungen des Systems ab einem Zeitpunkt t in einer beliebig kleinen (bis auf Rechenungenauigkeiten) Umgebung der Null auf, so sprechen wir von praktischer Stabilität in Anlehnung an Definition 3.26 der semiglobal praktisch asymptotischen Stabilität.

Die bei der praktischen Stabilität verwendeten Umgebungen von Null stellen dabei **ϵ -Umgebungen** (oder besser einen “ ϵ -Schlauch”) in folgendem Sinne dar:

Der Parameter ϵ gibt dabei den maximalen Wert an, um den die berechnete Lösung ab einem bestimmten Zeitpunkt t in jedem Diskretisierungspunkt von Null abweicht (siehe hierzu Abbildung 7.32).



(a) Ab einem bestimmten Zeitpunkt t beträgt die maximale Abweichung von der Null an jedem Diskretisierungspunkt höchstens $\pm\epsilon$. Im zeitlichen Verlauf betrachtet ist die Beschreibung dieser Umgebungen als “Schlauch” treffender.



(b) 3-dimensionale Ansicht des ϵ -Schlauchs.

Abbildung 7.32: Veranschaulichung des ϵ -Schlauchs.

Für die sich ergebenden unterschiedlichen Fälle nehmen wir die Auswertung folgendermaßen vor:

- Wir suchen den minimalen (numerisch exakt) stabilisierenden Horizont für die Parameterbelegung gemäß Variante 1 (Standardbelegung) auf dem Zeitintervall $[0, t_f]$ mit Endzeit $t_f = 3.0$.
- Kann ein solcher gefunden werden, geben wir diesen an, überprüfen die restlichen Varianten, ob diese diesbezüglich einen kleineren (numerisch exakt) stabilisierenden Horizont liefern und geben diesen gegebenenfalls an.
- Kann kein (numerisch exakt) stabilisierender Horizont gefunden werden, betrachten wir den ersten Horizont, für den innerhalb des Zeitintervalls $[0, t_f]$ mit $t_f = 3.0$ ein Verlauf der Lösungen gegen Null erkennbar ist. Diesen untersuchen wir nun auf praktische Stabilität, indem wir dessen Auswirkung auf das Langzeitverhalten (z.B. für $t_f = 120.0$) der Lösungen betrachten und geben dazu anschließend die für uns sinnvollen ϵ -Umgebungen an, in denen der weitere Verlauf der Lösungen zu erwarten ist. Für die restlichen Varianten gehen wir analog vor.
- Lassen sich im Vergleich zu Variante 1 in den übrigen Varianten keine Verbesserungen feststellen, werden diese nicht weiter betrachtet.

Zur Veranschaulichung und Belegung der von uns gemachten Beobachtungen und erhaltenen Resultate verwenden wir u.a. die zum Vergleich der Normen L^2 und H^1 in Auswertung 7.3.3 eingeführte betragsmäßig gemittelte örtliche Abweichung $y_{mv}(x, t)$ von der Null (definiert in (7.14)) und, im Falle einer verteilten Steuerung, die betragsmäßig gemittelte Steuerung $u_{mv}(x, t)$ (definiert in (7.15)).

Wir untersuchen die Wellengleichung sowohl unter der Dirichlet-Randbedingung (siehe Abschnitt 2.5.2.1)

$$y(0, t) = \sigma_1 = 0, \quad y(1, t) = \sigma_2 = 0$$

als auch unter der Neumann-Randbedingung (siehe Abschnitt 2.5.2.2)

$$\frac{\partial y}{\partial n}(0, t) = \sigma_3 = 0.0 \quad \frac{\partial y}{\partial n}(1, t) = \sigma_4 = 0.0$$

auf dem Gebiet $\Omega = (0, 1)$ mit Rand $\Gamma := \partial\Omega$.

Für die Diskretisierung unseres Systems verweisen wir auf Abschnitt 6.3:

- Für Dirichlet-RB mit verteilter Steuerung: Diskretisierung (6.11)
- Für Dirichlet-RB mit Randsteuerung: Diskretisierung (6.14)
- Für einseitige Neumann-RB mit verteilter Steuerung: Diskretisierung (6.17)
- Für zweiseitige Neumann-RB mit verteilter Steuerung: Diskretisierung (6.19)
- Für Neumann-RB mit Randsteuerung: Diskretisierung (6.24)

7.3.4.1 Dirichlet-Randbedingung, verteilte Steuerung, L^2 -Norm

Daten/welle_DIR_DC_SQPF/welle_DIR_DC_SQPF_L2

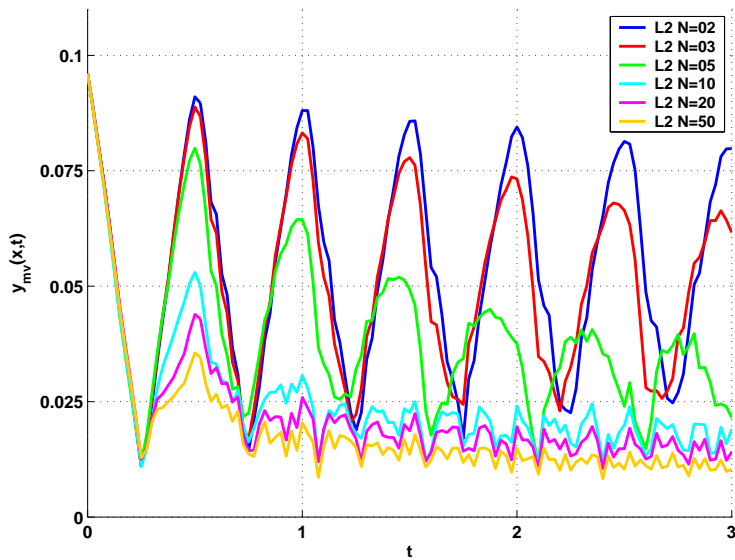
Das optimale Steuerungsproblem ist für Dirichlet-Randbedingung mit verteilter Steuerung und L^2 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ &\text{mit } l(y(\cdot, nT), u(\cdot, nT)) = \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2 \\ \text{s.t.} \\ y_{tt} - y_{xx} &= u \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y(0, t) &= 0 \\ y(1, t) &= 0 \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \\ \text{mit} \\ \Omega &= (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}. \end{aligned}$$

Beobachtungen

Für die von uns in den Auswertungen standardmäßig verwendeten Horizonte $N = 2, 3, 5, 10, 20$ und 50 kann bei einer Parameterbelegung entsprechend Variante 1 in keinem der Fälle eine numerisch exakte Stabilisierung in das Null-Profil erreicht werden, vielmehr gelangt man mit steigendem Horizont in immer nähere Umgebungen von der Null (vergleiche für Endzeit $t_f = 3.0$ die Abbildung 7.33).

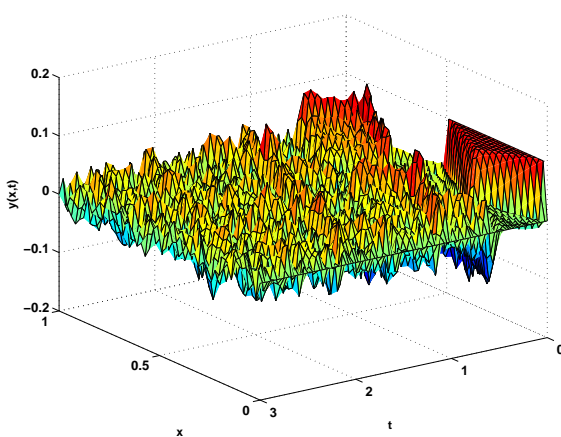
Dabei nehmen aber die Abstände zur Null mit steigendem Horizont nur minimal weiter ab. Diese minimale Abnahme legt die Vermutung nahe, dass wir auch durch eine weitere Erhöhung unseres Horizonts $N > 50$ nur eine minimale bis gar keine weitere Verbesserung mehr erwarten können. Da aber gegen Endzeit $t_f = 3.0$ noch immer eine Monotonie in der Abnahme der Abstände zur Null für Horizonte $N = 10, 20$ und 50 erkennbar ist, stellt sich uns die Frage, ob diese Eigenschaft auch für das Langzeitverhalten gilt, das im folgenden untersucht wird.



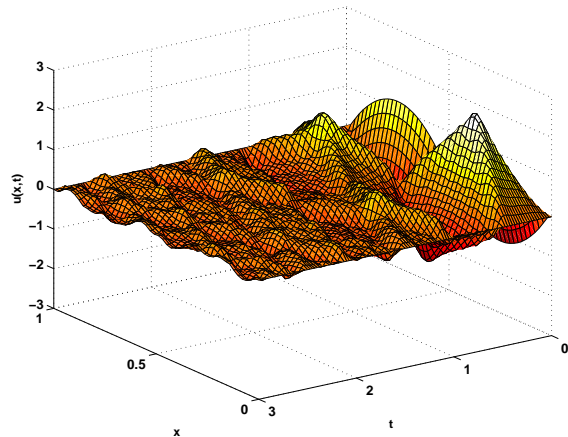
Erläuterung:

Man sieht deutlich, dass man sich für Horizonte $N \geq 10$ immer besser an die Null annähert. Werden allerdings die Horizonte untereinander verglichen, so stellt man fest, dass diese Verbesserung betragsmäßig immer stärker abnimmt.

Abbildung 7.33: Dirichlet-RB, DC, L^2 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 1



(a) Verlauf der Lösungen für $t_f = 3.0$

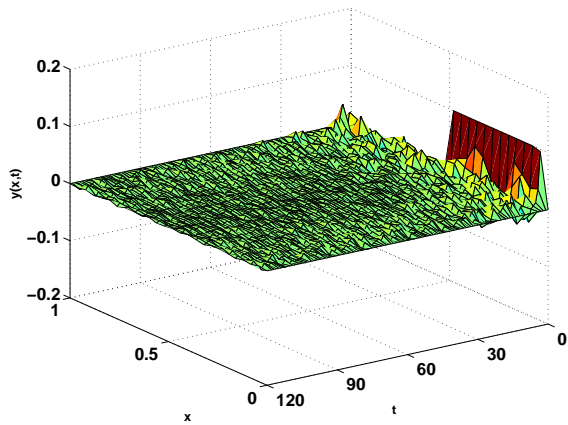


(b) Zugehöriger Steuerungsverlauf

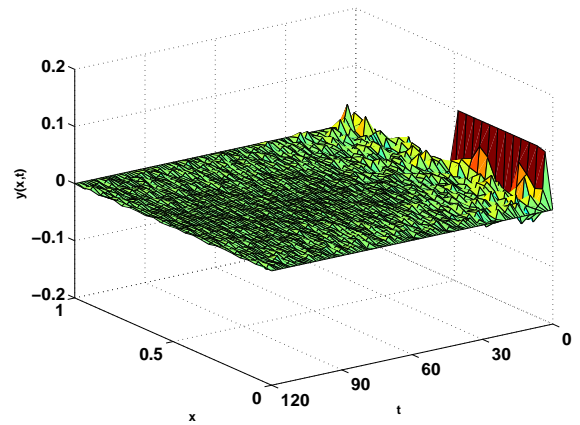
Abbildung 7.34: Dirichlet-RB, DC, L^2 -Norm, $N = 10$, Variante 1

Beobachtungen bezüglich des Langzeitverhaltens

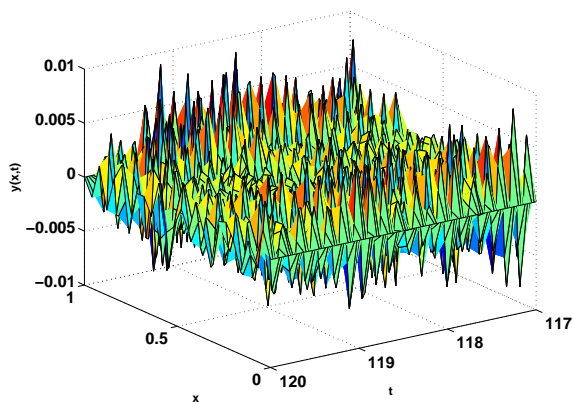
Um das Langzeitverhalten zu untersuchen haben wir die Auswertung von $N = 10$ (siehe Abbildung 7.34) für die neue Endzeit $t_f = 120.0$ wiederholt (siehe Abbildung 7.35(a)). Hier wird gegen Endzeit $t_f = 120.0$ eine kleinere Abweichung von der Null als im Vergleich zur Endzeit $t_f = 3.0$ festgestellt: Der Verlauf der Lösungen bewegt sich nun innerhalb eines ϵ -Schlauches um die Null mit $\epsilon = \pm 0.01$; vergleiche Abbildung 7.35(c). Wir erhalten also für längere Laufzeit und damit größere Endzeit t_f eine Verbesserung unseres Stabilisierungsverhaltens dahingehend, dass die Abweichungen von der Null weiterhin abnehmen. Allerdings bringt auch eine weitere Verdoppelung der Endzeit auf nun $t_f = 240.0$ nur noch eine minimale Verbesserung.



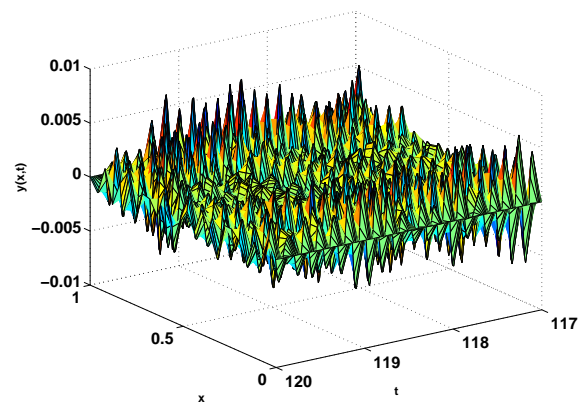
(a) Lösungsverlauf für $N = 10$ und $T = 0.025$



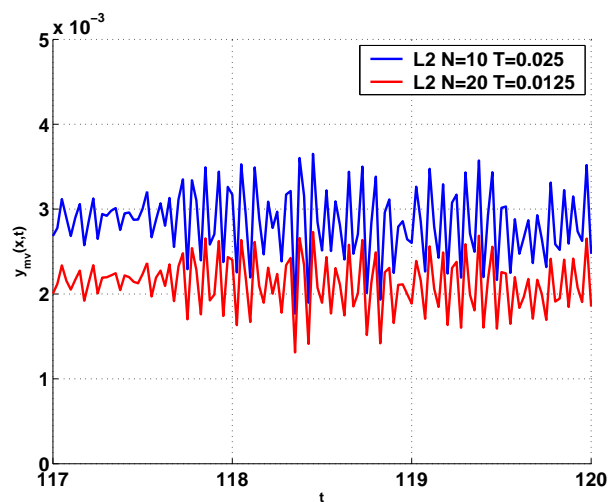
(b) Lösungsverlauf für $N = 20$ und $T = 0.0125$



(c) Lösungsverlauf für $N = 10$ ab $t = 117$



(d) Lösungsverlauf für $N = 20$ ab $t = 117$



(e) Vergleich der betragsmäßig gemittelten Abweichungen von der 0 an Zeitpunkten $t \geq 117$

Abbildung 7.35: Überprüfung auf praktische Stabilität, Dirichlet-RB, DC, L^2 -Norm, Variante 1 mit vergrößerter Endzeit $t_f = 120$

Viel eher scheint es, dass wir eine gewisse Genauigkeit unseres Verfahrens für diese Auswertung mit Dirichlet-Randbedingung, verteilter Steuerung und L^2 -Norm in Variante 1 erreichen und dass hierfür “nur” eine **praktische Stabilität** im Sinne von Definition 3.26 zu erwarten ist: Dazu muss sich auf dem betrachteten Zeitabschnitt für eine Verkürzung der Abtastperiode T und damit einhergehend einer größeren Anzahl von Abtastzeitpunkten t eine Verbesserung des Stabilisierungsverhaltens einstellen.

Dies wird dadurch überprüft, dass wir die Auswertung mit halbiertem Abtastzeit $T = 0.0125$ und doppeltem Horizont $N = 20$ für Endzeit $t_f = 120.0$ wiederholen.

Beobachtungen bezüglich praktischer Stabilität

Durch diese Parameterbelegung erreichen wir, dass beide Auswertungen bezüglich des betrachteten Zeitabschnitts $T_{\text{span}} \hat{=} 10 \cdot 0.025 \hat{=} 20 \cdot 0.0125$ übereinstimmen. Auch in diesem Fall tritt eine weitere (wenn auch minimale) Verbesserung ein, wie man an den Abbildungen 7.35(c), (d) und (e) erkennen kann.

Fazit aus unseren Beobachtungen

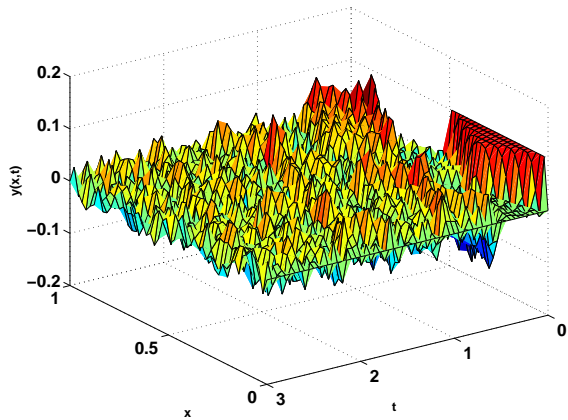
Eine numerisch exakte Stabilisierung in das Null-Profil wird für Variante 1 zwar nicht erreicht, dennoch kann im Falle $N = 10$ durch eine Überprüfung des Langzeitverhaltens auf eine praktische Stabilität geschlossen werden. Da die Angabe eines minimalen stabilisierenden Horizonts somit nicht möglich ist, wollen wir für $N = 10, 20$ und 50 und ab Zeitpunkt $t \geq 3.0$ die *Bereiche um die Null* (ϵ -Schlauch) angeben, in denen der weitere Lösungsverlauf liegt und somit zumindest praktische Stabilität erhalten werden kann:

| N | ϵ -Schlauch für $t \geq 3.0$ |
|-----------|---------------------------------------|
| ≥ 10 | $\epsilon = \pm 0.0663$ |
| ≥ 20 | $\epsilon = \pm 0.0530$ |
| ≥ 50 | $\epsilon = \pm 0.0400$ |

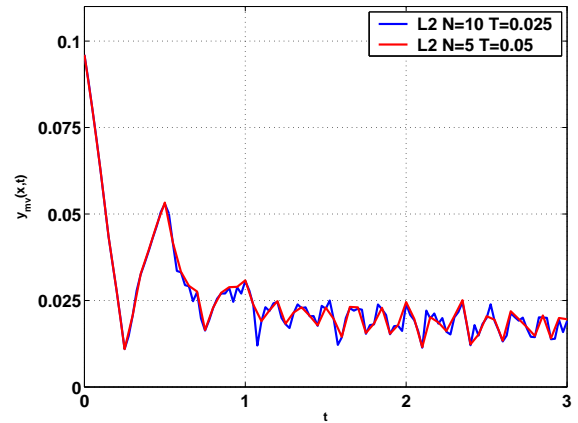
Tabelle 7.8: Bereiche der praktischen Stabilität: Dirichlet-RB, DC, L^2 -Norm, Variante 1

Auch bei der Auswertung von Variante 2 lässt sich kein (minimaler) Horizont für eine numerisch exakte Stabilisierung in das Null-Profil angeben. Vielmehr erhalten wir analoge Resultate wie für Variante 1, diesmal aber bereits für den Fall $N = 5$ (vergleiche Abbildung 7.36(a)), der im betragsmäßig gemittelten Lösungsverlauf fast identisch mit Variante 1 für $N = 10$ ist, wie Abbildung 7.36(b) zeigt. Auch in Bezug auf die Steuerungen erhalten wir kaum Unterschiede (vergleiche Abbildung 7.36(c)).

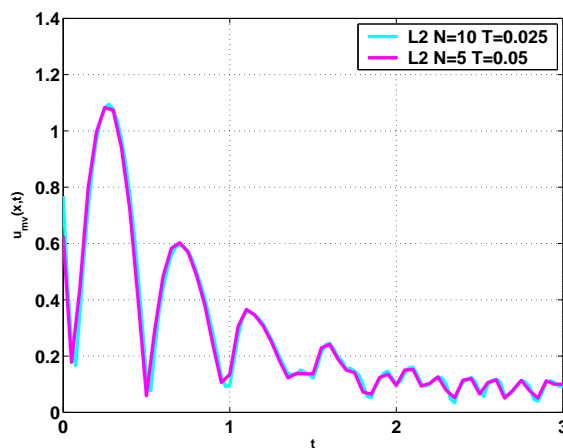
Dies ist allerdings nicht weiter verwunderlich, da Variante 2 nur bezüglich einer Verdopplung der Abtastzeit auf $T = 0.05$ variiert. Wir betrachten sozusagen also wieder in beiden Fällen die gleiche Zeitspanne $T_{\text{span}} \hat{=} 10 \cdot 0.025 \hat{=} 5 \cdot 0.05$ auf der der Lösungsverlauf präzisiert wird (d.h. auch hier bietet sich eine Überprüfung von praktischer Stabilität für größere Endzeit t_f an).



(a) Verlauf der Lösungen $N = 5$, Variante 2



(b) Vergleich Variante 1 und Variante 2 bzgl. der betragsmäßig gemittelten örtlichen Abweichung von 0



(c) Vergleich Variante 1 und Variante 2 bzgl. der betragsmäßig gemittelten Steuerungen

Abbildung 7.36: Dirichlet-RB, DC, L^2 -Norm

Bemerkung 7.11

Variante 3 wurde ebenfalls ausgewertet, liefert aber vergleichsweise die schlechtesten Ergebnisse bezüglich des Stabilisierungsverhaltens, der Kosten und der Zeit, so dass wir hier nicht näher darauf eingehen wollen. Sämtliche Daten dazu können aber auf der beigegeführten Daten-CD-ROM eingesehen werden:



Daten/welle_DIR_DC_SQPF/welle_DIR_DC_SQPF_L2/*_X200U98

7.3.4.2 Dirichlet-Randbedingung, verteilte Steuerung, H^1 -Norm



Daten/welle_DIR_DC_SQPF/welle_DIR_DC_SQPF_H1

Das optimale Steuerungsproblem ist für Dirichlet-Randbedingung mit verteilter Steuerung und H^1 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\min J_N(y_0(\cdot), u) = \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT))$$

mit $l(y(\cdot, nT), u(\cdot, nT)) =$
 $= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2$

s.t.

$$\begin{aligned} y_{tt} - y_{xx} &= u \\ y(x, 0) &= y_0(x) && \text{für } x \in \Omega \\ y(0, t) &= 0 \\ y(1, t) &= 0 \\ u(x, t) &\in U && \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) && \text{für } t \in [nT, (n+1)T) \end{aligned}$$

mit

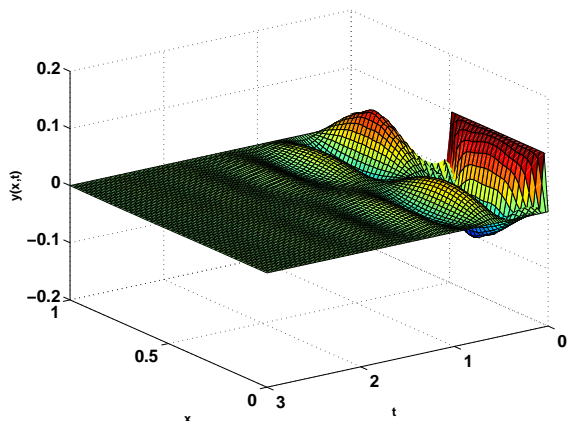
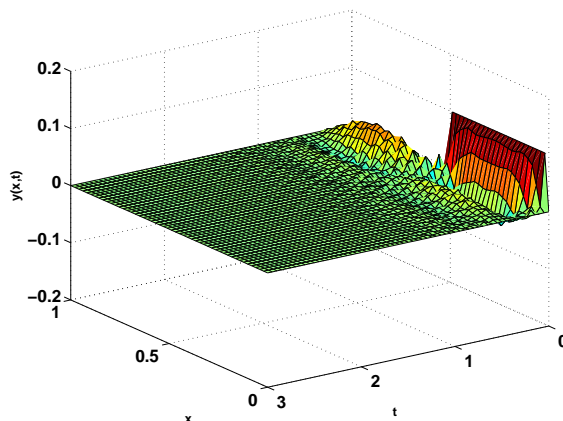
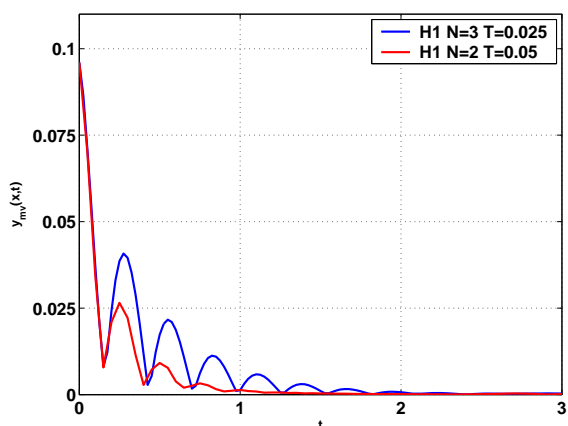
$$\Omega = (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}.$$

Beobachtungen

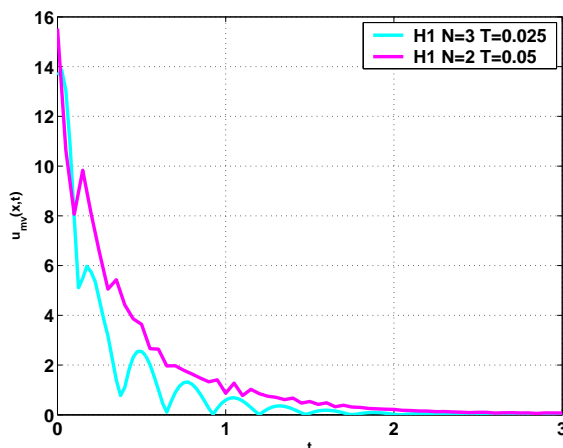
Bereits mit $N = 2$ in Variante 1 erhalten wir hier für eine Endzeit von $t_f = 3.0$ eine klar erkennbare Tendenz zur Stabilisierung in die Null, wobei aber bereits durch eine Erhöhung des Horizonts um 1 auf $N = 3$ schon ab einem Zeitpunkt $t > 2.0$ die numerisch exakte Stabilisierung in die Null eintritt (siehe Abbildung 7.37(a)). Somit erhalten wir in diesem Fall als minimalen stabilisierenden Horizont $N = 3$.

Variante 2 liefert uns für Dirichlet-Randbedingung mit verteilter Steuerung und H^1 -Norm im Zielfunktional sogar den minimalen stabilisierenden Horizont $N = 2$ mit numerisch exakter Stabilisierung in das Null-Profil bereits ab Zeitpunkt $t > 1.5$ (siehe Abbildung 7.37(b)).

Dass diese Auswertung des Weiteren eine Verbesserung der Auswertung für Variante 1 mit $N = 3$ darstellt, verdeutlichen die Gegenüberstellungen der jeweiligen betragsmäßig gemittelten Kosten und Steuerungen in den Abbildungen 7.38(a) und (b).

(a) Verlauf der Lösungen $N = 3$, Variante 1(b) Verlauf der Lösungen $N = 2$, Variante 2Abbildung 7.37: Dirichlet-RB, DC, H^1 -Norm

(a) Vergleich der betragsmäßig gemittelten Abweichungen von der 0



(b) Vergleich der betragsmäßig gemittelten Steuerungen

Abbildung 7.38: Vergleich der Variante 1 ($N = 3$) mit Variante 2 ($N = 2$): Dirichlet-RB, DC, H^1 -Norm**Bemerkung 7.12**

Auch hier wird auf eine Angabe der erhaltenen vergleichsweise schlechteren Ergebnisse aus der Auswertung für Variante 3 verzichtet. Wir verweisen stattdessen auf die beigefügte Daten-CD-ROM:



Daten/welle_DIR_DC_SQPF/welle_DIR_DC_SQPF_H1/*_X200U98

7.3.4.3 Dirichlet-Randbedingung, Randsteuerung, L^2 -Norm



Daten/welle_DIR_BC_SQPF/welle_DIR_BC_SQPF_L2

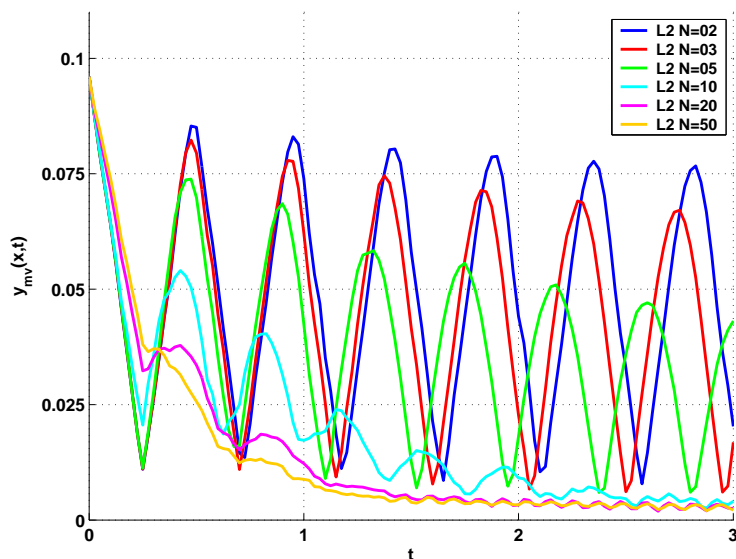
Das optimale Steuerungsproblem ist für Dirichlet-Randbedingung mit Randsteuerung und L^2 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ &\text{mit } l(y(\cdot, nT), u(\cdot, nT)) = \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Gamma)}^2 \\ \text{s.t.} \\ y_{tt} - y_{xx} &= 0 \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y(0, t) &= u(0, t) \\ y(1, t) &= u(1, t) \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \\ \text{mit} \\ \Omega &= (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}. \end{aligned}$$

Beobachtungen

Ähnlich der Auswertung mit verteilter Steuerung unter Dirichlet-Randbedingung und L^2 -Norm erhalten wir hier in Variante 1 für keinen der von uns ausgewerteten Horizonte $N = 2, 3, 5, 10, 20, 50$ bis zu einer Endzeit von $t_f = 3.0$ eine numerisch exakte Stabilisierung in das Null-Profil. Vielmehr ergibt sich erneut nur eine “Annäherung an die Null”, die für steigende Horizonte N weiter gegen Null hin verläuft (siehe dazu die betragsmäßig gemittelten Abweichungen vom Null-Profil in Abbildung 7.39). Auch hier erhalten wir einen für die weitere Auswertung charakteristischen Verlauf erst ab einem Horizont von $N = 10$ (vergleiche Abbildung 7.41(a) mit zugehöriger Randsteuerung 7.41(b)).

Selbiges lässt sich analog auch für die Varianten 2 und 4 beobachten, die für $N = 10$ gegen Endzeit $t_f = 3.0$ im Vergleich zu Variante 1 ein fast identisches Stabilisierungsverhalten aufweisen (vergleiche Abbildung 7.41(c) und (e)).

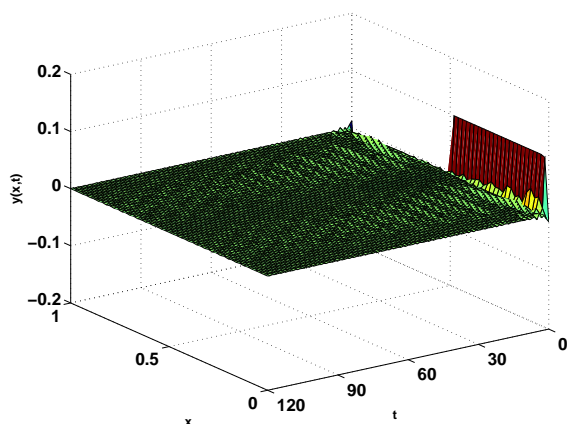
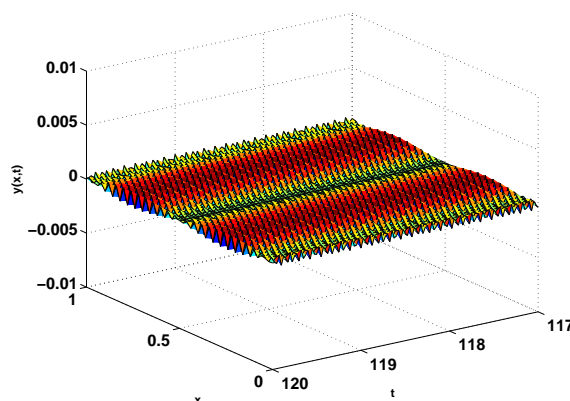
**Erläuterung:**

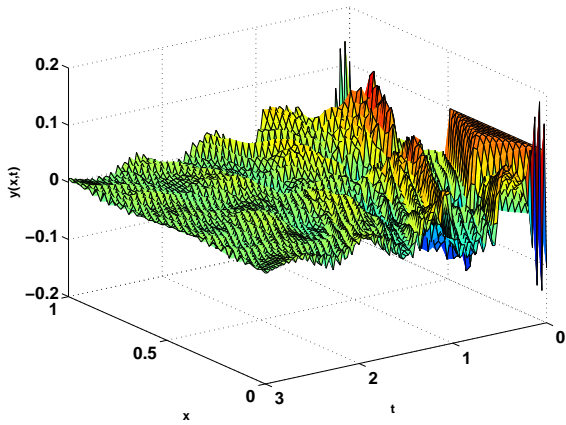
Für steigenden Horizont nimmt die betragsmäßig gemittelte Abweichung vom Null-Profil immer weiter ab, erreicht aber bis zur Endzeit $t_f = 3.0$ in Variante 1 keine numerisch exakte Stabilisierung in die Null.

Abbildung 7.39: Dirichlet-RB, BC, L^2 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 1

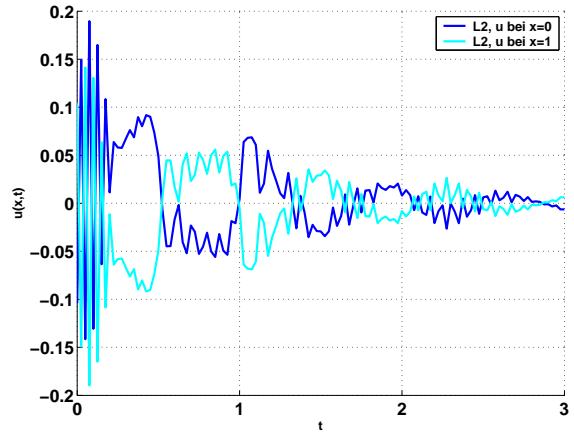
Dabei stellt sich Variante 1 letztendlich als die mit dem diesbezüglich besseren Verhalten heraus (vergleiche Abbildung 7.42). Dahingehend ist also auch durch Modifikation der Ausgangsparameter keine Verbesserung in Bezug auf den Erhalt eines minimalen stabilisierenden Horizonts zu erwarten.

Lediglich die Untersuchung des Langzeitverhaltens (hier in Variante 1 mit $N = 20$) für eine Endzeit $t_f = 120.0$ ergibt analog zur Auswertung mit verteilter Steuerung unter Dirichlet-Randbedingung und L^2 -Norm, dass für längere Laufzeiten eine praktische Stabilisierung erhalten werden kann (vergleiche hierzu Abbildung 7.40(a) und (b)).

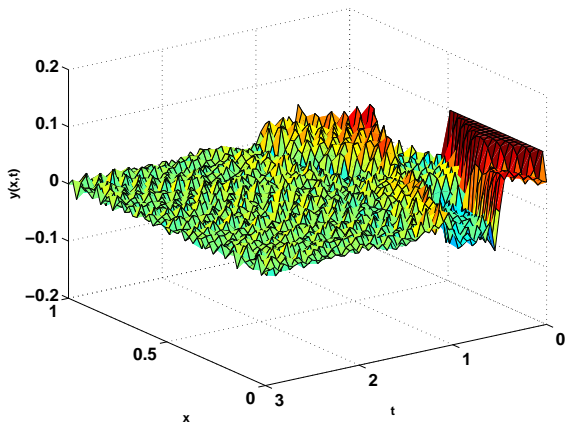
(a) Verlauf der Lösungen Variante 1 für $N = 20$ (b) Verlauf der Lösungen Variante 1 für $N = 20$ ab $t = 117$ Abbildung 7.40: Dirichlet-RB, BC, L^2 -Norm, $t_f = 120$



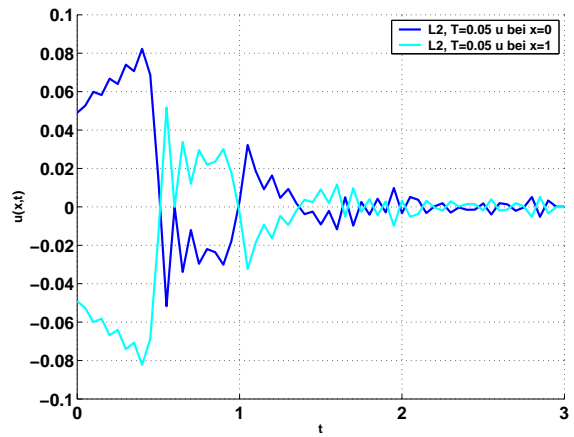
(a) Lösungsverlauf für $N = 10$, Variante 1



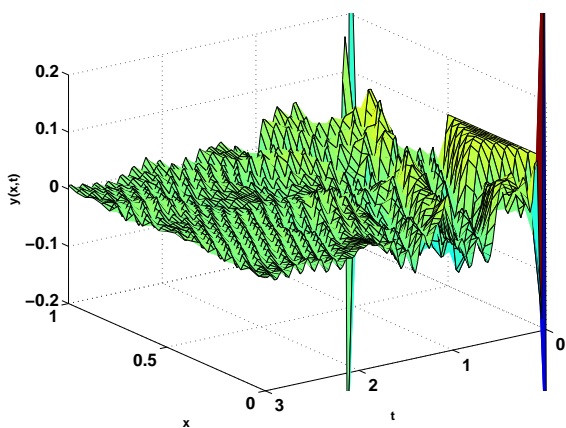
(b) Steuerung $N = 10$, Variante 1



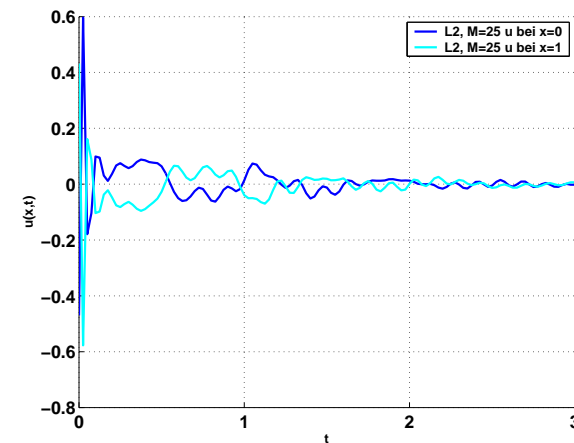
(c) Lösungsverlauf für $N = 10$, Variante 2



(d) Steuerung $N = 10$, Variante 2



(e) Lösungsverlauf für $N = 10$, Variante 3



(f) Steuerung $N = 10$, Variante 3

Abbildung 7.41: Dirichlet-RB, BC, L^2 -Norm

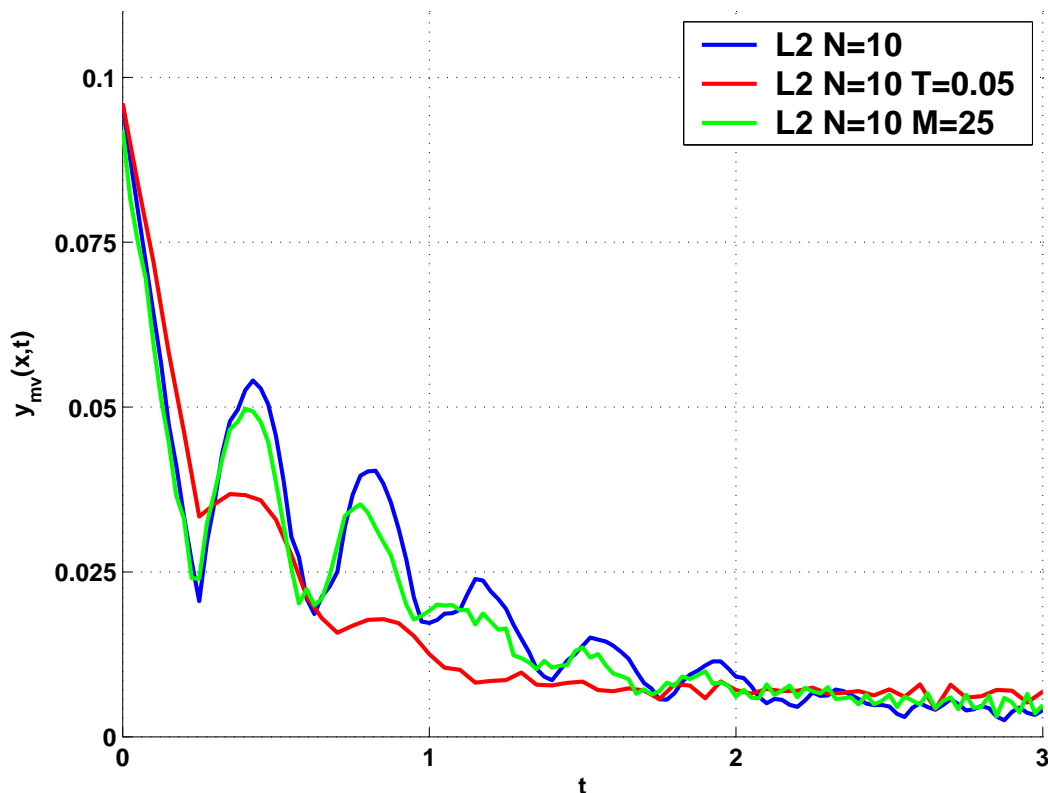


Abbildung 7.42: Dirichlet-RB, BC, L^2 -Norm: Vergleich der betragsmäßig gemittelten Abweichungen von der 0.

Wir wollen auch hier wieder für Horizonte N den jeweiligen ϵ -Schlauch angeben, in dem der weitere Lösungsverlauf ab Zeitpunkt $t \geq 3.0$ liegt:

| N | ϵ -Schlauch für $t \geq 3.0$ |
|-----------|---------------------------------------|
| ≥ 10 | $\epsilon = \pm 0.0172$ |
| ≥ 20 | $\epsilon = \pm 0.0079$ |
| ≥ 50 | $\epsilon = \pm 0.0066$ |

Tabelle 7.9: Bereiche der praktischen Stabilität: Dirichlet-RB, BC, L^2 -Norm, Variante 1

7.3.4.4 Dirichlet-Randbedingung, Randsteuerung, H^1 -Norm



Daten/welle_DIR_BC_SQPF/welle_DIR_BC_SQPF_H1

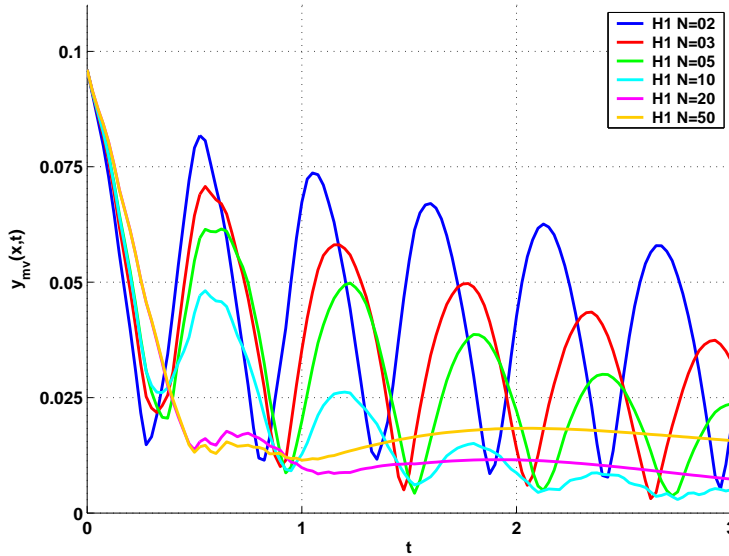
Das optimale Steuerungsproblem ist für Dirichlet-Randbedingung mit Randsteuerung und H^1 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ \text{mit } l(y(\cdot, nT), u(\cdot, nT)) &= \\ &= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Gamma)}^2 \\ \text{s.t.} \\ y_{tt} - y_{xx} &= 0 \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y(0, t) &= u(0, t) \\ y(1, t) &= u(1, t) \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \\ \text{mit} \\ \Omega &= (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}. \end{aligned}$$

Beobachtungen

Im Gegensatz zur H^1 -Norm mit verteilter Steuerung erhalten wir hier für keinen der von uns für die Auswertungen standardmäßig verwendeten Horizonte $N = 2, 3, 5, 10, 20, 50$ eine numerisch exakte Stabilisierung in das Null-Profil, unabhängig von der gewählten Variante 1, 2 oder 4. Bis zu einem Horizont von $N = 10$ gelangen wir zwar in immer kleinere Umgebungen der Null, ab $N = 20$ lassen sich jetzt aber größer werdende Abweichungen von der Null beobachten (vergleiche besonders für Horizonte $N = 10, 20$ und 50 gegen Endzeit $t_f = 3.0$ die Abbildung 7.43).

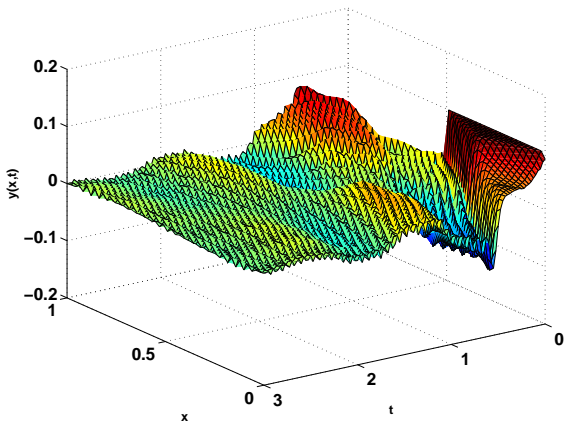
Da unser Hauptaugenmerk auf dem Stabilisierungsverhalten gegen Endzeit $t_f = 3.0$ liegt, ist trotz eines besseren Anfangsverhaltens des Lösungsverlaufs bei Übergang von $N = 10$ auf einen größeren Horizont von $N = 20$, die Tatsache einer Verschlechterung der Ergebnisse gegen Endzeit $t_f = 3.0$ für uns entscheidender. Deshalb macht hier eine nochmalige Erhöhung von N , um einen minimalen stabilisierenden Horizont zu finden, sicherlich keinen Sinn. Dies wollen wir durch Abbildung 7.44 anhand des einzelnen Verlaufs der Lösungen, der betragsmäßig gemittelten Abweichungen von der Null und den Steuerungen der jeweiligen Horizonte nochmals verdeutlichen: Für Horizont $N = 20$ ist zu erkennen, dass der Wellenverlauf sogar unter das angestrebte Null-Profil sinkt und bis zur Endzeit von $t_f = 3$ sich diesem wieder langsam annähert.



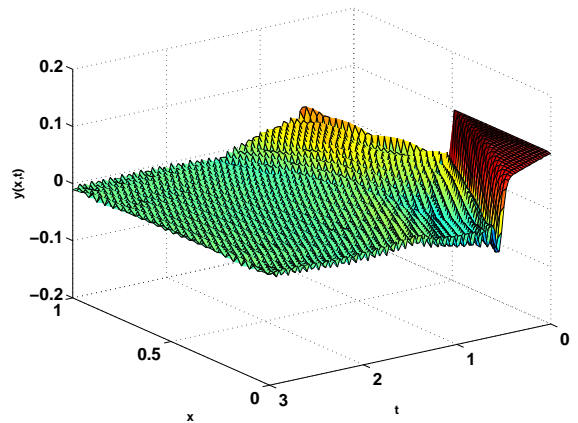
Erläuterung:

Bei diesem Vergleich ist auffällig, dass ab einem Horizont von $N = 20$ die Abweichungen von der Null wieder steigen und nicht wie erwartet mit steigendem Horizont immer weiter fallen. Die Auswertung für $N = 10$ scheint hier die sinnvollste zu sein.

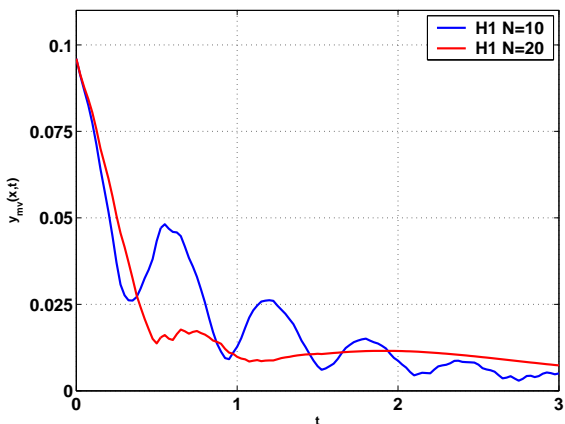
Abbildung 7.43: Dirichlet-RB, BC, H^1 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 1



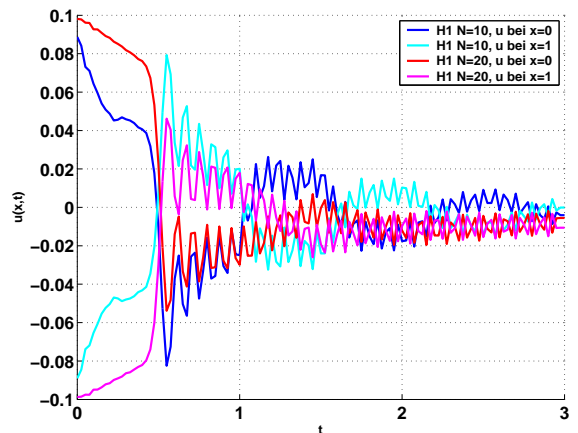
(a) Verlauf der Lösungen für $N = 10$



(b) Verlauf der Lösungen für $N = 20$



(c) Vergleich der betragsmäßig gemittelten Abweichung von der 0

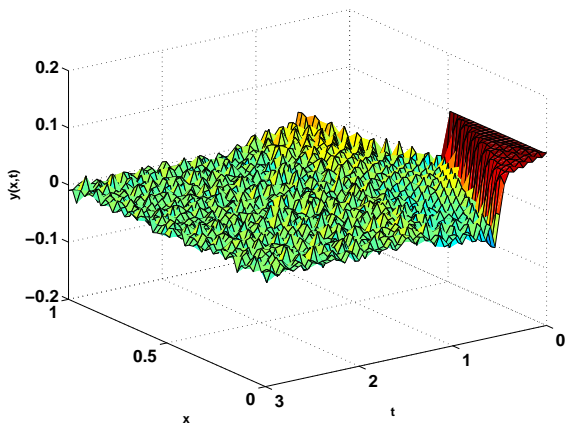


(d) Vergleich der anliegenden Steuerungen für $N = 10$ und $N = 20$

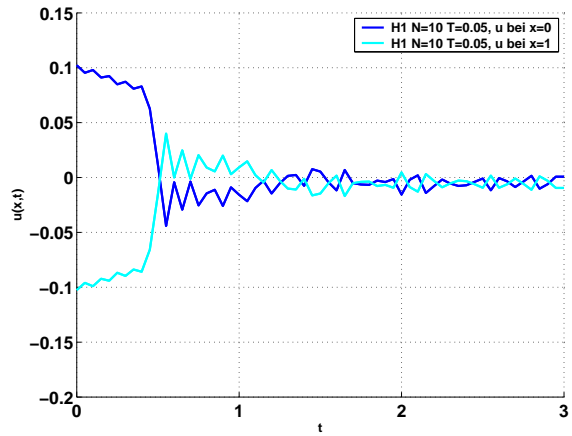
Abbildung 7.44: Dirichlet-RB, BC, H^1 -Norm, Variante 1

7 Auswertung und numerische Ergebnisse

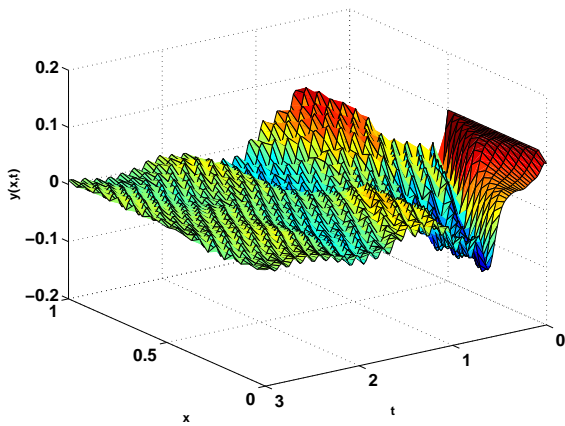
Da für uns die Auswertung mit Horizont $N = 10$ der Null am nächsten kommt, wird nun Variante 1 für $N = 10$ mit den Varianten 2 und 4 verglichen, ob sich diesbezüglich ein kleinerer Horizont mit besserem oder zumindest gleichem Stabilisierungsverhalten finden lässt (man vergleiche hierzu Abbildung 7.45).



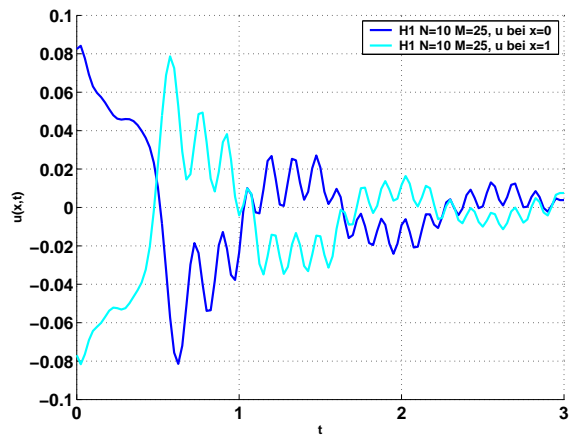
(a) Verlauf der Lösungen, Variante 2



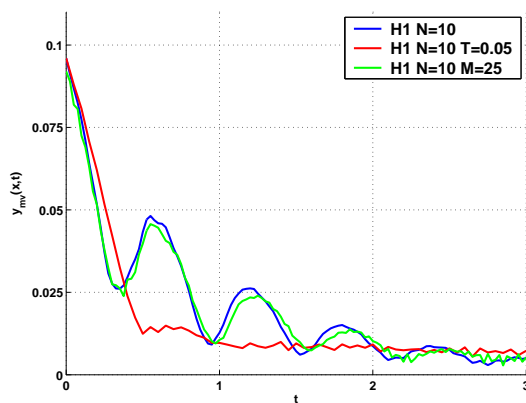
(b) zugehörige Steuerung, Variante 2



(c) Verlauf der Lösungen, Variante 3



(d) zugehörige Steuerung, Variante 3



(e) Vergleich Variante 1, 2 und 3 bzgl. der betragsmäßig gemittelten Abweichung von der 0

Abbildung 7.45: Dirichlet-RB, BC, H^1 -Norm, $N = 10$

Bei dem Vergleich der Abweichung von der Null ist zu beobachten, dass die Variante 2 für $N = 10$ zwar zu Beginn schnell in die Nähe der Null gelangt, bei fortschreitender Zeit jedoch von den anderen beiden Varianten “eingeholt” wird und somit nicht weiter in Betracht kommt.

Für Variante 4 hingegen stellt sich im Vergleich zu Variante 1 der gesamte Verlauf der Systemlösungen als sehr ähnlich, gegen Endzeit $t_f = 3$ sogar als fast identisch heraus. Es stellt sich nun die Frage, ob für Variante 4 ein stabilisierender Horizont $N > 10$ gefunden werden kann oder ob sich, wie in Variante 1, eine Verschlechterung der Ergebnisse ab einem Horizont von $N = 20$ einstellt. Dazu betrachten wir für Variante 4 den Vergleich der Abweichungen von der Null für alle Horizonte in Abbildung 7.46.

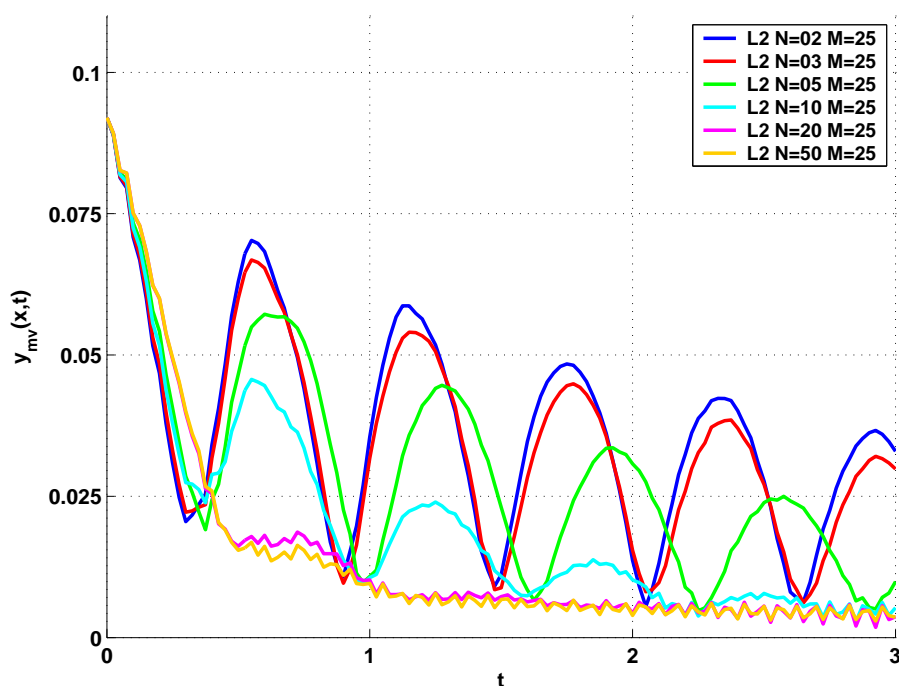


Abbildung 7.46: Dirichlet-RB, BC, H^1 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 4

Anders als bei dem Vergleich mit der Parameterbelegung entsprechend der Variante 1 können wir nun bei steigendem Horizont immer weiter abnehmende Abstände von der Null beobachten, auch wenn bei einem Horizont von $N = 20$ und $N = 50$ kaum mehr Veränderungen erkennbar sind, auf keinen Fall aber Verschlechterungen. Einen stabilisierenden Horizont finden wir jedoch auch hier nicht.

Abschließend überprüfen wir noch für Variante 4 und Horizont $N = 20$ das Langzeitverhalten der Lösungen in Bezug auf Stabilisierung für eine Endzeit von $t_f = 120.0$ (siehe 7.47).

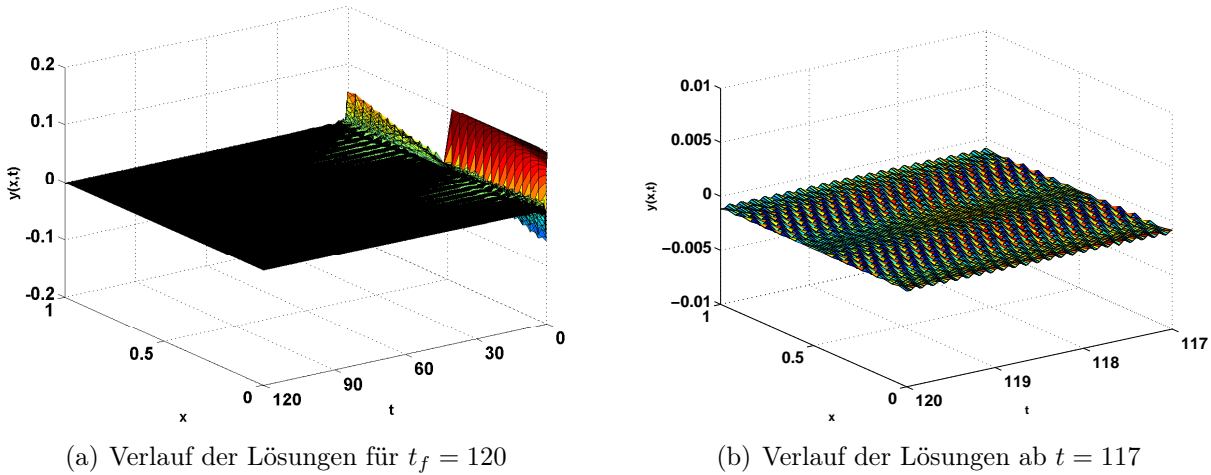


Abbildung 7.47: Dirichlet-RB, BC, H^1 -Norm, $N = 20$, Variante 4

Auch wenn es sich auf dem kompletten Zeitabschnitt $[0, 120]$ dem Anschein nach um eine numerisch exakte Stabilisierung in das Null-Profil handelt (vergleiche Abbildung 7.47(a)), so macht eine genauere Betrachtung des zeitlichen Endabschnitt $[117, 120]$ deutlich, dass der Verlauf der Lösungen zwar fast ebenmäßig, aber **unterhalb** der Null verläuft und dort auch bleibt (siehe Abbildung 7.47(b)). Auch hier müssen wir von der Aussage einer numerisch exakten Stabilisierung absehen, können aber zumindest eine praktische Stabilisierung in Betracht ziehen.

Wir wollen hier wieder für Horizonte N den jeweiligen ϵ -Schlauch angeben, in dem der weitere Verlauf der Lösungen ab Zeitpunkt $t \geq 3.0$ liegt, allerdings für Variante 4:

| N | ϵ -Schlauch für $t \geq 3.0$ |
|-----------|---------------------------------------|
| ≥ 10 | $\epsilon = \pm 0.0221$ |
| ≥ 20 | $\epsilon = \pm 0.0100$ |
| ≥ 50 | $\epsilon = \pm 0.0100$ |

Tabelle 7.10: Bereiche der praktischen Stabilität: Dirichlet-RB, BC, H^1 -Norm, Variante 4

7.3.4.5 Einseitige Neumann-Randbedingung, verteilte Steuerung



Daten/welle_NEUM1_DC_SQPF

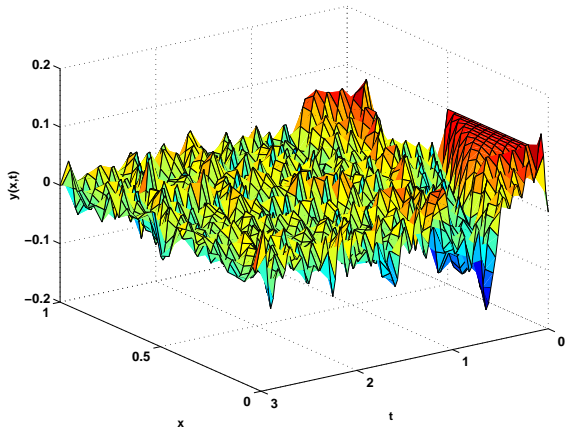
Das optimale Steuerungsproblem ist für einseitige Neumann-Randbedingung am linken Rand und Dirichlet-Randbedingung am rechten Rand mit verteilter Steuerung und L^2 -Norm bzw. H^1 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ &\text{für } L^2\text{-Norm mit } l(y(\cdot, nT), u(\cdot, nT)) = \\ &= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2 \\ &\text{für } H^1\text{-Norm mit } l(y(\cdot, nT), u(\cdot, nT)) = \\ &= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2 \\ \\ \text{s.t.} \\ y_{tt} - y_{xx} &= u \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y_\nu(0, t) &= 0 \\ y(1, t) &= 0 \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \\ \\ \text{mit} \\ \Omega &= (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}. \end{aligned}$$

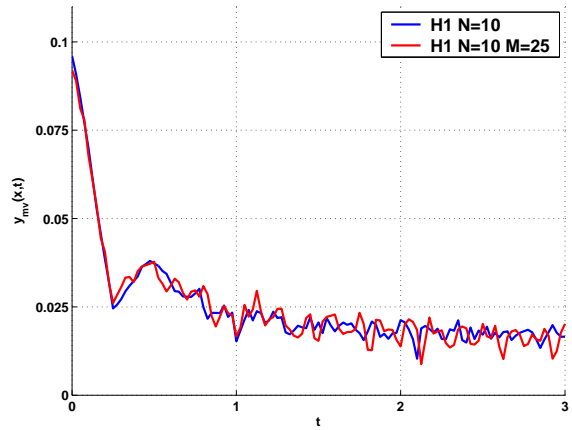
Beobachtungen im Falle der L^2 -Norm

Wir erhalten im Falle der Varianten 1 und 2 analoge Resultate entsprechend der Auswertung 7.3.4.1 unter Dirichlet-Randbedingung mit verteilter Steuerung und L^2 -Norm: Für Variante 1 erhalten wir auch hier ab einem Horizont von $N \geq 10$ praktische Stabilität, für Variante 2 ab einem Horizont von $N \geq 5$.

Für Variante 4 erhalten wir im Vergleich zu Variante 1 nur minimale Veränderungen des Stabilisierungsverhaltens (vgl. Abbildung 7.48 für $N = 10$).



(a) Verlauf der Lösungen Variante 4



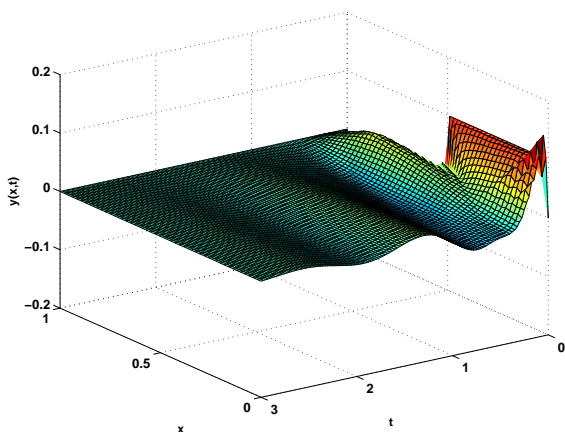
(b) Vergleich Variante 1 und 4 bzgl. der betragsmäßig gemittelten Abweichungen von 0

Abbildung 7.48: Neumann-RB, DC, $N = 10$, L^2 -Norm

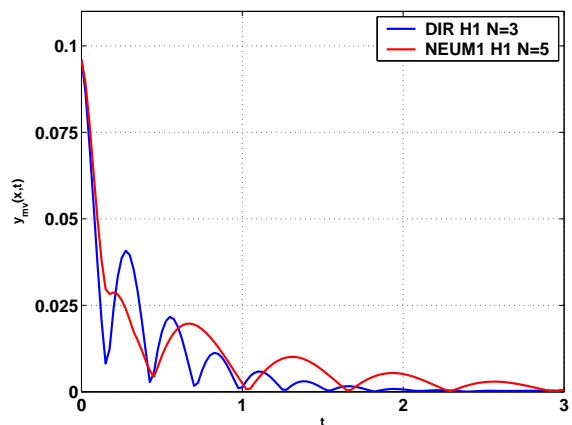
Beobachtungen im Falle der H^1 -Norm

Wir erhalten auch hier analoge Resultate entsprechend der Auswertung 7.3.4.2 unter Dirichlet-Randbedingung mit verteilter Steuerung und H^1 -Norm im Falle der Varianten 1 und 2, allerdings für größere Horizonte N : Im Fall der Variante 1 deutet sich die Tendenz zur Stabilisierung ab einem Horizont von $N = 5$ (vgl. 7.49) an und tritt spätestens ab $N = 10$ ein (im Dirichlet-Fall bereits ab $N = 3$), im Fall der Variante 2 die Tendenz ab einem Horizont $N = 3$ (vgl. 7.50) und die Stabilisierung ab $N = 5$ (im Dirichlet-Fall bereits ab $N = 2$).

Variante 4 verhält sich bezüglich der Stabilisierung in das Null-Profil wie Variante 1.

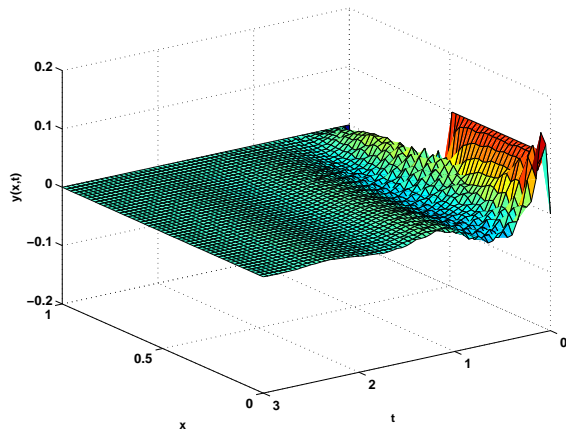


(a) Verlauf der Lösungen für $N = 5$

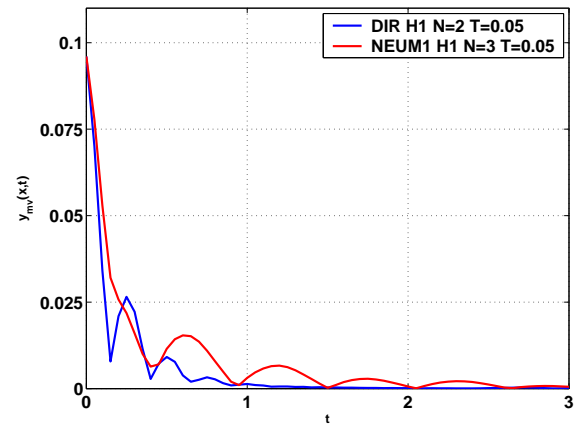


(b) Vergleich Dirichlet- und Neumann-Randbedingung bzgl. der betragsmäßig gemittelten Abweichungen von 0

Abbildung 7.49: Neumann-RB einseitig, DC, H^1 -Norm, Variante 1



(a) Verlauf der Lösungen für $N = 3$



(b) Vergleich Dirichlet- und Neumann-Randbedingung bzgl. der betragsmäßig gemittelten Abweichungen von 0

Abbildung 7.50: Neumann-RB einseitig, DC, H^1 -Norm, Variante 2

7.3.4.6 Zweiseitige Neumann-Randbedingung, verteilte Steuerung



Daten/welle_NEUM2_DC_SQPF

Das optimale Steuerungsproblem ist für zweiseitige Neumann-Randbedingung an beiden Rändern mit verteilter Steuerung und L^2 -Norm bzw. H^1 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\min J_N(y_0(\cdot), u) = \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT))$$

für L^2 -Norm mit $l(y(\cdot, nT), u(\cdot, nT)) =$
 $= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2$

für H^1 -Norm mit $l(y(\cdot, nT), u(\cdot, nT)) =$
 $= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2$

s.t.

$$y_{tt} - y_{xx} = u$$

$$y(x, 0) = y_0(x) \quad \text{für } x \in \Omega$$

$$y_\nu(0, t) = 0$$

$$y_\nu(1, t) = 0$$

$$u(x, t) \in U \quad \text{für } x \in \Omega$$

$$u(\cdot, t) = u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T)$$

mit

$$\Omega = (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}.$$

Beobachtungen im Falle der L^2 -Norm

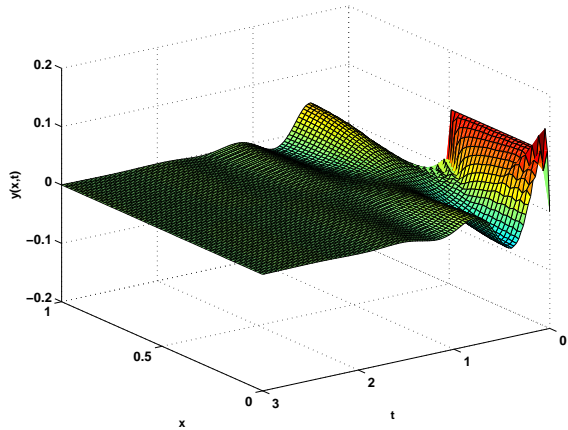
Mit Neumann-Randbedingung an beiden Rändern erhalten wir analoge Resultate zur Auswertung mit einseitiger Neumann-Randbedingung und L^2 -Norm.

Beobachtungen im Falle der H^1 -Norm

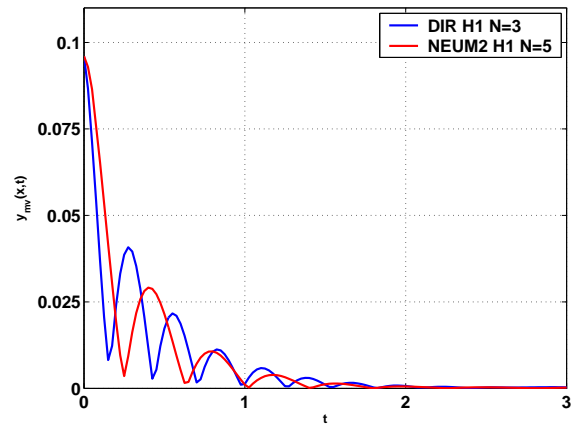
Wie auch bei einseitiger Neumann-Randbedingung erhalten wir hier analoge Resultate zur Auswertung 7.3.4.2 unter Dirichlet-Randbedingung mit verteilter Steuerung und H^1 -Norm. Allerdings bekommen wir im Fall der Varianten 1 und 2 jetzt die numerisch exakte

Stabilisierung bereits für $N = 5$ (Variante 1, vgl. Abbildung 7.51) und $N = 3$ (Variante 2, vgl. Abbildung 7.52), finden aber dennoch keine kleineren stabilisierenden Horizonte als im Dirichlet-Fall.

Die Variante 4 verhält sich auch bei zweiseitiger Neumann-Randbedingung bezüglich der numerisch exakten Stabilisierung in das Null-Profil wie Variante 1.

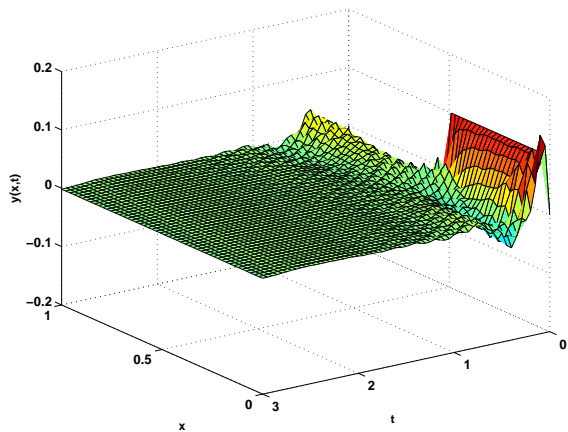


(a) Verlauf der Lösungen für $N = 5$

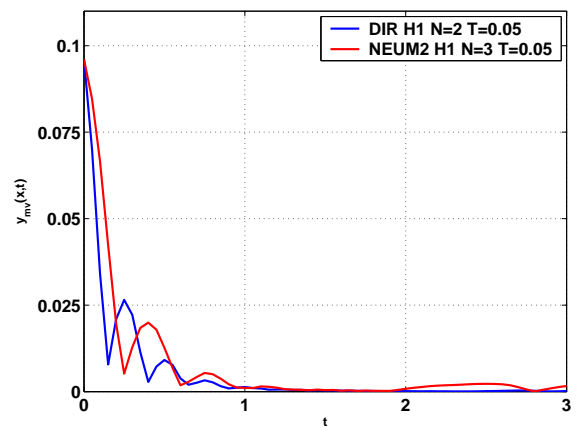


(b) Vergleich Dirichlet- und Neumann-Randbedingung bzgl. der betragsmäßig gemittelten Abweichungen von 0

Abbildung 7.51: Neumann-RB zweiseitig, DC, H^1 -Norm, Variante 1



(a) Verlauf der Lösungen für $N = 3$



(b) Vergleich Dirichlet- und Neumann-Randbedingung bzgl. der betragsmäßig gemittelten Abweichungen von 0

Abbildung 7.52: Neumann-RB zweiseitig, DC, H^1 -Norm, Variante 2

7.3.4.7 Neumann-Randbedingung, Randsteuerung



Daten/welle_NEUM_BC_SQPF

Das optimale Steuerungsproblem ist für Neumann-Randbedingung an beiden Rändern mit Randsteuerung und L^2 -Norm bzw. H^1 -Norm im Zielfunktional auf diskretem Zeitgitter gegeben durch:

$$\begin{aligned} \min J_N(y_0(\cdot), u) &= \sum_{n=0}^{N-1} l(y(\cdot, nT), u(\cdot, nT)) \\ &\text{für } L^2\text{-Norm mit } l(y(\cdot, nT), u(\cdot, nT)) = \\ &= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2 \\ &\text{für } H^1\text{-Norm mit } l(y(\cdot, nT), u(\cdot, nT)) = \\ &= \|y(\cdot, nT)\|_{L^2(\Omega)}^2 + \|y_x(\cdot, nT)\|_{L^2(\Omega)}^2 + \lambda \|u(\cdot, nT)\|_{L^2(\Omega)}^2 \end{aligned}$$

s.t.

$$\begin{aligned} y_{tt} - y_{xx} &= 0 \\ y(x, 0) &= y_0(x) \quad \text{für } x \in \Omega \\ y_\nu(0, t) &= u(0, t) \\ y_\nu(1, t) &= u(1, t) \\ u(x, t) &\in U \quad \text{für } x \in \Omega \\ u(\cdot, t) &= u(\cdot, nT) \quad \text{für } t \in [nT, (n+1)T) \end{aligned}$$

mit

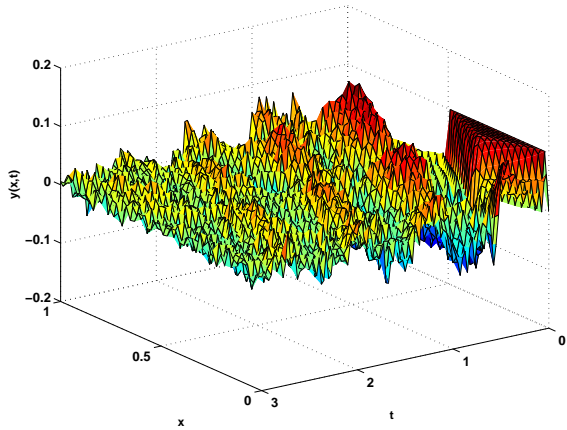
$$\Omega = (0, 1) \quad \text{und} \quad \Gamma = \{0, 1\}.$$

Beobachtungen im Falle der L^2 -Norm

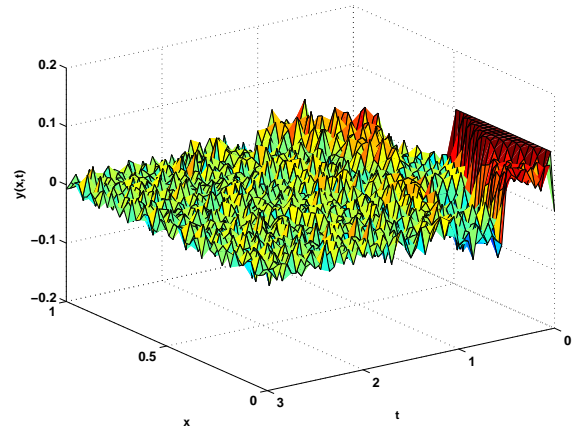
Bei einem Vergleich der betragsmäßigen Abweichungen von der Null für alle Horizonte $N = 2, 3, 5, 10, 20, 50$ ergeben sich analoge Resultate entsprechend der Auswertung 7.3.4.3 unter Dirichlet- Randbedingung mit Randsteuerung und L^2 -Norm: Für Variante 1 erhalten wir auch hier einen für die weitere Auswertung charakteristischen Verlauf der Lösungen ab einem Horizont von $N = 10$.

Bei einem Vergleich der Variante 4 mit der Standardvariante 1 und einem Horizont von $N = 10$ lässt sich keine Veränderung bezüglich des Stabilisierungsverhaltens feststellen. Ein Vergleich der Variante 2 mit Standardvariante 1 ergibt hingegen zu Beginn des zeitlichen Verlaufs eine Verbesserung des Stabilisierungsverhaltens, das heißt eine schnellere Abnahme der Abweichungen vom Nullprofil. Zur Endzeit $t_f = 3$ hin werden die Unterschiede jedoch verschwindend klein.

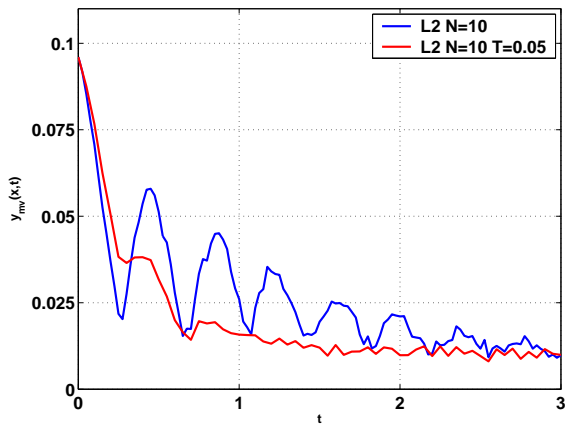
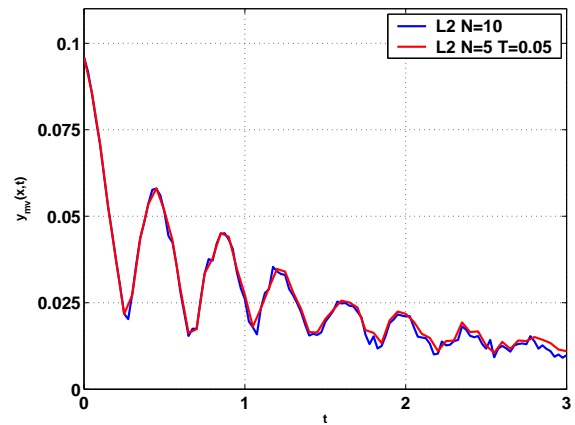
Es stellt sich die Frage, ob es einen kleineren Horizont für Variante 2 gibt, der vergleichbare Ergebnisse wie Variante 1 und $N = 10$ liefert. Bei Verringerung des Horizonts auf $N = 5$ in Variante 2 erreichen wir verglichen mit Variante 1 und $N = 10$ fast identischen Verlauf der Abweichungen vom Null-Profil und erhalten somit durch Variation der Parameterbelegung einen kleineren Horizont für die praktische Stabilität (vgl. Abbildung 7.53).



(a) Verlauf der Lösungen Variante 1



(b) Verlauf der Lösungen Variante 2

(c) Vergleich Variante 1 und 2 bzgl. der betragsmäßig gemittelten Abweichungen von 0 für $N = 10$ (d) Vergleich Variante 1 und 2 bzgl. der betragsmäßig gemittelten Abweichungen von 0 für $N = 10$ bzw. $N = 5$ Abbildung 7.53: Neumann-RB, BC, L^2 -Norm

Beobachtungen im Falle der H^1 -Norm

Auch in dieser Auswertung finden wir zwar keinen minimalen stabilisierenden Horizont, aber anders als in der Auswertung 7.3.4.4 unter Dirichlet-Randbedingung mit Randsteuerung und H^1 -Norm erhalten wir bei einem Vergleich der betragsmäßigen Abweichungen von der Null bei steigendem Horizont immer weitere Verbesserungen, also kleiner werdende Abweichungen. Der erste "sinnvolle" Horizont, der bei einer Endzeit von $t_f = 3.0$ einen charakteristischen Verlauf für die weiteren Auswertungen zeigt, ist $N = 20$ (vgl. Abbildung 7.54).

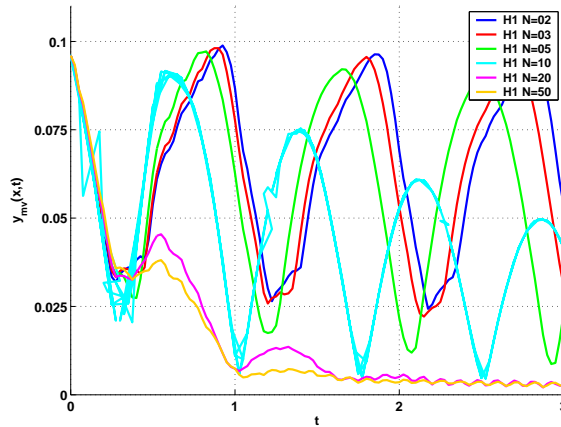
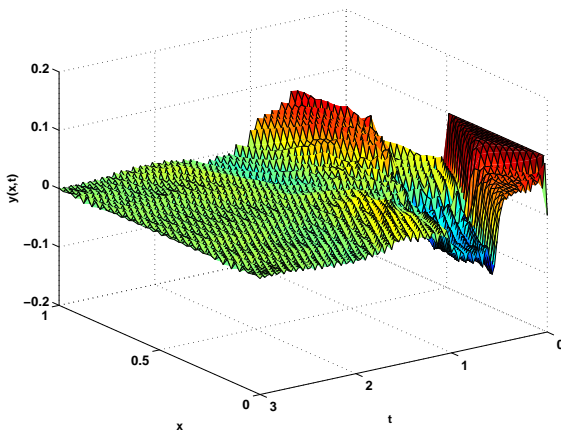
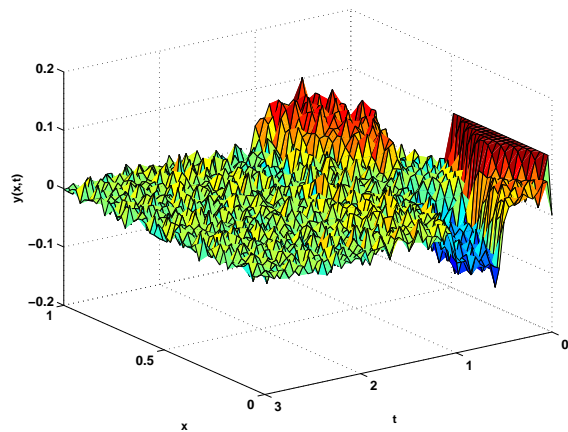


Abbildung 7.54: Neumann-RB, BC, H^1 -Norm, $N = 2, 3, 5, 10, 20, 50$, Variante 1

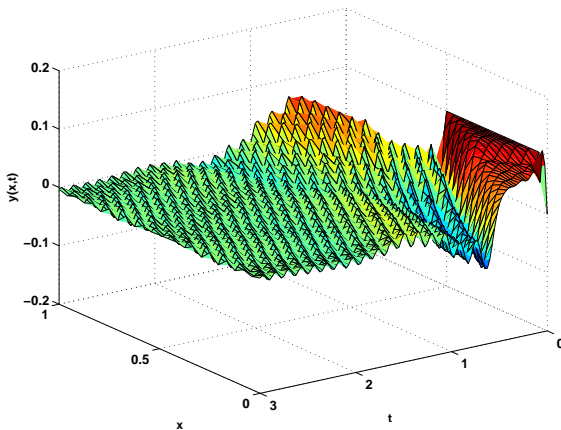
Sowohl ein Vergleich der Variante 2 als auch der Variante 4 mit der Standardvariante 1 und einem Horizont von $N = 20$ bringt uns nur kleine Verschlechterungen bezüglich des Stabilisierungsverhaltens in das Nullprofil (vgl. Abbildung 7.55).



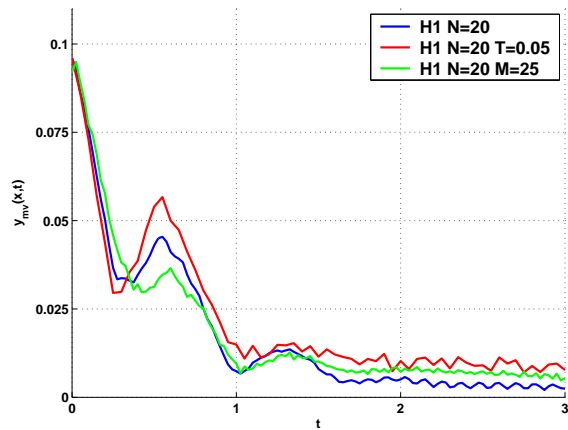
(a) Verlauf der Lösungen Variante 1



(b) Verlauf der Lösungen Variante 2



(c) Verlauf der Lösungen Variante 4



(d) Vergleich Variante 1, 2 und 4 bzgl. der betragsmäßig gemittelten Abweichungen von 0

Abbildung 7.55: Neumann-RB, BC, H^1 -Norm, $N = 20$

7.3.4.8 Zusammenfassung der Ergebnisse

Wir erhalten zusammenfassend folgende Horizonte N bezüglich numerisch exakter und praktischer Stabilität unter Berücksichtigung der Variante, in der für diesen Horizont in das Null-Profil bzw. in eine Umgebung davon stabilisiert wurde:

| Randbed. | Steuerung | Norm | Variante | Horizont N | Art der Stabilisierung |
|----------|-----------|-------|----------|--------------|------------------------|
| DIR | DC | L^2 | 2 | 5 | praktisch |
| DIR | DC | H^1 | 2 | 2 | numerisch exakt |
| DIR | BC | L^2 | 1 | 10 | praktisch |
| DIR | BC | H^1 | 1 | 10 | praktisch |
| NEUM1 | DC | L^2 | 2 | 5 | praktisch |
| NEUM1 | DC | H^1 | 2 | 5 | numerisch exakt |
| NEUM2 | DC | L^2 | 2 | 5 | praktisch |
| NEUM2 | DC | H^1 | 2 | 3 | numerisch exakt |
| NEUM | BC | L^2 | 2 | 5 | praktisch |
| NEUM | BC | H^1 | 1 | 20 | praktisch |

Tabelle 7.11: Übersicht der Horizonte bezüglich Stabilisierung

Aufgrund der in dieser Auswertung gemachten Beobachtungen und erhaltenen Resultate lässt sich Folgendes festhalten:

- Eine numerisch exakte Stabilisierung erhalten wir ausschließlich im Falle verteilter Steuerung mit H^1 -Norm, dies sogar ab einem sehr kleinen Horizont (siehe Tabelle 7.11).
- Im Falle numerisch exakter Stabilisierung in das Null-Profil erhalten wir bei Dirichlet-Randbedingung diesbezüglich kleinere Horizonte als im Vergleich zu Neumann-Randbedingungen.
- Eine Erhöhung der Abtastzeit T entsprechend Variante 2 liefert bei verteilter Steuerung stets eine Verbesserung des Stabilisierungsverhaltens in das Null-Profil im Vergleich zu den Varianten 1 und 3 bzw. 1 und 4.
- Bei verteilter Steuerung mit L^2 -Norm und Randsteuerung im Allgemeinen finden wir keinen minimalen numerisch exakt stabilisierenden Horizont. Lediglich können jeweils Horizonte angegeben werden, für die wir praktische Stabilität erhalten.
- Für eine Verdoppelung oder Halbierung der Anzahl der örtlichen Diskretisierungspunkte M entsprechend Variante 3 oder Variante 4 sind nur minimale Veränderungen im Vergleich zu Variante 1 zu beobachten.

7 *Auswertung und numerische Ergebnisse*

- Eine Unterscheidung der Verwendung der Randbedingungen bei verteilter Steuerung bewirkt keine Veränderung in der Art der Stabilisierung: sowohl numerisch exakte als auch praktische Stabilität bleibt erhalten.
- Beim Übergang von verteilter Steuerung auf Randsteuerung geht im Falle der H^1 -Norm die numerisch exakte Stabilisierung in das Null-Profil verloren und wir erhalten nur noch praktische Stabilität.

A Anhang

A.1 Dokumentation des Quelltextes für die Wellengleichung

Wir wollen nun den von uns in den Auswertungen zur Wellengleichung verwendeten Quelltext `wellengleichung.cpp` dokumentieren, um sowohl die Verifizierung der erhaltenen Resultate als auch die Verwendung für weitere Auswertungen zu ermöglichen. Dabei soll nur auf die für unsere Auswertungen relevanten Stellen im Programm eingegangen werden.

Wir werden dies im Folgenden am Beispiel der Auswertung unter **Dirichlet-Randbedingung** mit **Randsteuerung** veranschaulichen.

❶ Auswahl der Randbedingungen und der Art der Steuerung

Dies geschieht entsprechend folgender Tabelle über die Belegung des Parameters `ipar[2]`:

| <code>ipar[2]</code> | Randbedingung und Art der Steuerung |
|----------------------|--|
| 1 | Dirichlet-Randbedingung, verteilte Steuerung |
| 2 | Dirichlet-Randbedingung, Randsteuerung |
| 3 | Neumann-Randbedingung, verteilte Steuerung |
| 4 | Neumann-Randbedingung, Randsteuerung |

```
414 ipar[2] = 2; //Festlegen der Randbedingung und der Art der Steuerung
```

❷ Festlegen der Parameter

In Zeile 40 bis 44 werden der Reihe nach

- die Anzahl der Diskretisierungspunkte `DIMENSION_X`
- die Anzahl der Steuerungen `DIMENSION_U`
- der Horizont `HORIZONT` und
- die Abtastperiode `H_NEW`

einggegeben:

```

40 #define DIMENSION_X 100
41 #define DIMENSION_U 2
42 #define HORIZONT 20
43 #define H_NEW 0.025

```

Dabei ist zu beachten, dass als Anzahl der Diskretisierungspunkte `DIMENSION_X` stets die **doppelte Anzahl** angegeben werden muss: in diesem Fall beträgt die Anzahl der Diskretisierungspunkte eigentlich $M = 50$, dementsprechend wird `DIMENSION_X = 50 · 2 = 100` gesetzt. Dies ist implementierungstechnisch bedingt und hat keinen weiteren Einfluss auf die Berechnungen.

③ Festlegen der Randbedingungen am rechten Rand

Am rechten Rand kann entweder eine Dirichlet- (Zeilen 126,127) oder eine Neumann-Randbedingung (Zeilen 122,123) durch entsprechendes Ein- bzw. Auskommentieren gesetzt werden:

```

121 //Entkommentieren fuer: NEUMANN-Randbedingung am rechten Rand
122 dx[dimension_x - 2] = x[dimension_x - 1];
123 dx[dimension_x - 1] = 2.0*pow(c/gitter,2.0)*(sigma_neum_re*gitter - x[
    dimension_x - 2]+x[dimension_x - 4]);
124
125 //Entkommentieren fuer: DIRICHLET-Randbedingung am rechten Rand
126 //dx[dimension_x - 2] = x[dimension_x - 1];
127 //dx[dimension_x - 1] = 0.0;

```

④ Wertebelegungen am Rand entsprechend der Randbedingungen

Für die Dirichlet-Randbedingung entsprechend Abschnitt 2.5.2.1 werden die folgenden Variablen belegt:

```

175 double sigma_li = 0.0;
176 double sigma_re = 0.0;

```

Für die Neumann-Randbedingung entsprechend Abschnitt 2.5.2.2 müssen folgende Variablen belegt werden:

```

64 double sigma_neum_li = 0.0;
65 double sigma_neum_re = 0.0;

```

⑤ **Auswahl des SQP-Verfahrens** Die entsprechende Zeile mit dem gewünschten Verfahren muss einkommentiert werden:

```

419 SqpFortran * sqp = new SqpFortran();
420 //SqpNagC * sqp = new SqpNagC();

```

Je nach Wahl des Verfahrens müssen zusätzlich die folgenden Zeilen ein- bzw. auskommentiert werden. In diesem Fall fiel die Entscheidung auf SQPF, so dass die Zeilen 443–448 einkommentiert und die Zeilen 450–460 auskommentiert wurden:

```

442 // SQPF configuration example
443 sqp->setAccuracy ( 1E-10 );
444 sqp->setMaxIterations ( 15000 );

```



```

445 sqp->setMaxFun ( 50 );
446 sqp->setLineSearchTol ( 0.1 );
447 sqp->setMinimalStepsize ( 1E-6 );
448 string num_lib = "SQPF"; // dient nur zur Benennung der Auswertung in
    der Ausgabe
449
450 //NAGC configuration example
451 //NAGC_Major_Iterations_Limit = 50;
452 //MIL_factor = 1;
453 //NAGC_Memory_Factor = 1.5;
454 //MF_factor = 1.2;
455 //NagError fail;
456 //INIT_FAIL(fail);
457 //nag_opt_nlp_option_set_double("Major Optimality Tolerance", 1E-5, sqp
    ->nagState(), &fail);
458 //nag_opt_nlp_option_set_integer("Major Iterations Limit",
    NAGC_Major_Iterations_Limit, sqp->nagState(), &fail);
459 //sqp->setNAGMemoryFactor(NAGC_Memory_Factor);
460 //string num_lib = "NAGC"; // dient nur zur Benennung der Auswertung in
    der Ausgabe

```

⑥ Angabe weiterer Parameter

Der Parameter `laenge` entspricht dabei dem betrachteten eindimensionalen Gebiet (hier: $[0, 1]$), der Parameter `hoehe` den Abweichungen von Null im Z-Profil und der Parameter `lambda` der Gewichtung des Steueraufwands in der Zielfunktion.

```

159 laenge = 1.0;
160 hoehe = 0.1;
161 lambda = 1.0e-3;

```

⑦ Änderung des Startprofils

Eine Änderung des Startprofils (hier: Z-Profil) bezüglich der Abweichungen von der Null in jedem Diskretisierungspunkt wird durch eine Änderung des Parameters `hoehe` bewirkt:

```

181 for ( i = 2; i < dimension_x-2; i=i+2 )
182 {
183     x[i] = -hoehe;
184     x[i+1] = 0.0;
185 }
186 for ( i = 2; i < dimension_x/2; i=i+2 )
187 {
188     x[i] = x[i] + 2.0 * hoehe;
189     x[i+1] = 0.0;
190 }

```

Hier wurden zuerst alle Werte auf -0.1 gesetzt und anschließend die erste Hälfte auf $+0.1$ erhöht, so dass man ein Startprofil entsprechend des in den Auswertungen verwendeten Z-Profiles mit Abweichungen ± 0.1 von Null erhält.

⑧ Angabe des Zielprofils

In unserem Fall wurde auf eine Stabilisierung in das Null-Profil abgezielt. Dieses kann

durch Setzen des Vektors `rpar` der Länge entsprechend der Anzahl an Diskretisierungspunkten geändert werden.

```
199 for ( i = 0; i < dimension_x; i++ )
200 {
201     rpar[10 + i] = 0.0 ;
202 }
```

⑨ Festlegen der Anzahl der Programmdurchläufe

Für eine Auswertung zum Beispiel mit Endzeit $t_f = 3.0$ und Abtastperiode $T = 0.025$ muss die Variable `j` auf $\frac{3.0}{0.025} = 120$ gesetzt werden.

```
504 for ( int j = 0; j < 120; j++ )
```

Bemerkung A.1

Das Programm birgt eine Reihe weiterer Modifikations- und Einstellmöglichkeiten, wie z.B. Änderung der Zustands- und Steuerbeschränkungen, auf die hier aber nicht näher eingegangen werden soll. Diesbezüglich sei auf die Kommentare an entsprechenden Stellen im Quelltext verwiesen. ○

A.2 MATLAB-Codes zur graphischen Darstellung der Resultate

Zur graphischen Darstellung der erhaltenen Ergebnisse und Resultate aus den Auswertungen zur Wellengleichung werden die folgenden MATLAB-Codes als Quellcode zur Verfügung gestellt. Diese sollen in der folgenden Tabelle kurz vorgestellt werden. Für weitere Informationen und Erläuterungen sei auf die Kommentare in den jeweiligen Files verwiesen.

| Name: | Erklärungen und Hinweise: |
|-----------------------|---|
| WEQ_Y_2DPlot.m | 2-dimensionale Darstellung des Lösungsverlaufs entlang der Zeit. Das MATLAB-File wird in den jeweiligen Auswertungs-Ordner kopiert und dort ausgeführt. Hinweis: Ränder werden im Falle einer Dirichlet-Randsteuerung gleich Null dargestellt! Man verwende stattdessen hierfür WEQ_Y_2DPlot_BC.m |
| WEQ_Y_2DPlot_BC.m | 2-dimensionale Darstellung des Lösungsverlaufs entlang der Zeit bei Verwendung der Dirichlet-Randsteuerung. Wird in den jeweiligen Auswertungsordner kopiert und dort ausgeführt. |
| WEQ_Y_2DPlot_Grid.m | 2-dimensionale Darstellung des Lösungsverlaufs entlang der Zeit auf einem beliebigen Zeitabschnitt. Eignet sich vor allem für Auswertungen mit großen Endzeiten, z.B. $t_f \geq 120$. Dazu müssen im MATLAB-File die Parameter entsprechend belegt werden. Zur Ausführung wird es in den jeweiligen Auswertungs-Ordner kopiert. Hinweis: Ränder werden im Falle einer Dirichlet-Randsteuerung gleich Null dargestellt! |
| WEQ_Y_MVcomp_1DPlot.m | Vergleicht die durchschnittlichen betragsmäßig gemittelten Abweichungen von Null entlang der Zeit von zwei Auswertungen. Diese müssen dazu durch Angabe des jeweiligen Ordners im MATLAB-File bekannt gemacht und gegebenenfalls anhand von Parametern angepasst werden. Das File kann unter Berücksichtigung der richtigen Pfadangabe der Ordner an eine beliebige Stelle kopiert und von dort ausgeführt werden. Hinweis: Es können nur Auswertungen mit <i>gleicher Samplingperiode T</i> verglichen werden! |
| WEQ_U_1DPlot.m | 1-dimensionale Darstellung der Steuerungen entlang der Zeit. Wird in den jeweiligen Auswertungsordner kopiert und dort ausgeführt. |

| | |
|-------------------------|--|
| WEQ_U_compBC_1DPlot.m | 1-dimensionaler Vergleich der Randsteuerungen zweier Auswertungen entlang der Zeit. Diese müssen dazu durch Angabe des jeweiligen Ordners im MATLAB-File bekannt gemacht und gegebenenfalls anhand von Parametern angepasst werden. Das File kann unter Berücksichtigung der richtigen Pfadangabe der Ordner an eine beliebige Stelle kopiert und von dort ausgeführt werden. |
| WEQ_U_MVcompDC_1DPlot.m | 1-dimensionaler Vergleich der durchschnittlichen betragsmäßig gemittelten verteilten Steuerungen zweier Auswertungen entlang der Zeit. Diese müssen dazu durch Angabe des jeweiligen Ordners im MATLAB-File bekannt gemacht und gegebenenfalls anhand von Parametern angepasst werden. Das File kann unter Berücksichtigung der richtigen Pfadangabe der Ordner an eine beliebige Stelle kopiert und von dort ausgeführt werden. |

Tabelle A.1: Übersicht der bereitgestellten MATLAB-Codes.

Bemerkung A.2

Die ersten vier Files können ebenfalls durch eine interne Bekanntgabe des Ordnerpfades der Auswertung an jede beliebige Stelle kopiert und von dort ausgeführt werden. Aufgrund der Übersichtlichkeit und der einfacheren Handhabung empfiehlt sich aber eine Kopie des jeweiligen Files in jeden Auswertungsordner zu speichern, um sie sozusagen “vor Ort” auszuführen. ○

A.3 Inhalt der Daten-CD-ROM

Die dieser Arbeit beiliegende Daten-CD-ROM ist in vier Verzeichnisse gegliedert, die die vollständige Diplomarbeit als PDF, den für die Auswertungen verwendeten Quelltext, die bei der Auswertung der Wellengleichung erhaltenen Daten und die für die graphische Darstellung dieser Daten zur Verfügung gestellten MATLAB-Codes beinhalten. Im Folgenden wollen wir diese in einer Tabelle genau auflisten:

| Verzeichnis | Inhalt | Erklärungen |
|--------------|---|--|
| Diplomarbeit | diplomarbeit.pdf | Vollständige Ausarbeitung der Diplomarbeit im PDF-Format |
| Quelltext | wellengleichung.cpp | Enthält den in den Auswertungen zur Wellengleichung verwendeten Quelltext in C++. |
| Daten | welle_DIR_DC_SQPF welle_DIR_DC_NAGC welle_DIR_BC_SQPF welle_NEUM1_DC_SQPF welle_NEUM2_DC_SQPF welle_NEUM_BC_SQPF | Die Ordnerstruktur für die bei der Auswertung der Wellengleichung erhaltenen Daten wurde wie nachstehend beschrieben festgelegt. |
| MATLAB-Codes | WEQ_Y_2DPlot.m WEQ_Y_2DPlot_BC.m WEQ_Y_2DPlot_Grid.m WEQ_Y_MVcomp_1DPlot.m WEQ_U_1DPlot.m WEQ_U_compBC_1DPlot.m WEQ_U_MVcompDC_1DPlot.m | Hier sei auf den Anhang A.2 verwiesen, in dem bereits alle MATLAB-Codes für eine graphische Darstellung der erhaltenen Daten beschrieben wurden. |

Tabelle A.2: Inhalt der beiliegenden Daten-CD-ROM.

Anmerkung zur Ordnerstruktur:

- `welle` gibt die jeweilige PDE (hier: die Wellengleichung) an, die der Auswertung zugrunde liegt.
- Mit `DIR`, `NEUM1`, `NEUM2`, `NEUM` werden die verwendeten Randbedingungen angegeben: `DIR` bezeichnet dabei die Dirichlet- und `NEUM` die Neumann-Randbedingung. Bei der Neumann-Randbedingung wird noch bezüglich einseitiger (`NEUM1`) und zweiseitiger (`NEUM2` bzw. `NEUM`) Bedingung unterschieden.
- Die Art der verwendeten Steuerung wird durch `DC` für die verteilte und durch `BC` für die Randsteuerung angegeben.

- Mit SQPF bzw. NAGC wird das zur Anwendung kommende SQP-Verfahren unterschieden.
- Innerhalb dieser Ordner erweitern wir die Struktur um die verwendete Norm L2 und H1 in der Zielfunktion, dem betrachteten Horizont N und die Varianten aus unserer Auswertung: T005 steht dabei für die Abtastperiode $T = 0.05$ in Variante 2, die Bezeichnungen X200U98, X50U23, X50U02 für die Änderung der Anzahl der Diskretisierungspunkte (und der anliegenden Steuerungen) entsprechend Variante 3 und 4 (z.B. entspricht X200U98 dabei einer Anzahl von 200 Diskretisierungspunkten und 98 anliegenden verteilten Steuerungen).
- Wurde die Endzeit ausgehend von $t_f = 3.0$ variiert, haben wir dies mit Tf120 für $t_f = 120.0$ bzw. Tf240 für $t_f = 240.0$ angegeben.
- Eine Verkleinerung der Abtastperiode auf $T = 0.0125$ wurde durch den Zusatz T00125 kenntlich gemacht. Die Änderung eines Parameters, z.B. λ , wurde durch dessen namentliche Angabe und dem veränderten Wert angegeben: lambda10-3 bedeutet somit eine Belegung entsprechend $\lambda = 10^{-3}$ in dieser Auswertung.

Literaturverzeichnis

- [1] AESCHBACH-HERTIG, W. ; LEVIN, I. ; WAGENBACH, D.: *Umweltphysik für Alle: 6. Dynamik und Transport*. Universität Heidelberg, Präsentation, 2007
- [2] ALT, H. W.: *Lineare Funktionalanalysis*. Springer-Verlag Berlin, 1999
- [3] BAKER, G. L. ; BLACKBURN, J. A.: *The pendulum*. Oxford University Press, 2005. – A case study in physics
- [4] BANDLE, C.: Die Mathematik als moderne Weltsprache: Am Beispiel der Differenzialgleichungen. In: *Uni Nova, Wissenschaftsmagazin der Universität Basel* , Nr. 87, S. 10–13, 2000
- [5] BÄRWOLFF, G.: *Numerik für Ingenieure, Physiker und Informatiker*. Spektrum Akademischer Verlag München, 2007
- [6] BRACEWELL, R.: *The Fourier Transform and Its Applications*. McGraw-Hill New York, 2000. – 61–65 S. – Heaviside’s Unit Step Function $H(x)$
- [7] BRANDT, S.: *Die Heisenbergsche Unbestimmtheitsbeziehung - Entdeckung und Anwendungen*. – Universität Stuttgart, Vorlesung im Fachbereich Physik, 2003. – Veröffentlicht unter <http://alephwww.physik.uni-siegen.de/~brandt/sv02/sld001.html>
- [8] BÜSKENS, C.: *Linienmethoden*. Universität Bayreuth, Skript zur Vorlesung, 2003
- [9] BÜSKENS, C.: *Optimierung dynamischer Systeme (Teil 1: Zeitdiskrete Systeme)*. Universität Bremen, Skript zur Vorlesung, 2005
- [10] CAMACHO, E. F. ; BORDONS, C.: *Model Predictive Control*. Springer-Verlag London Ltd., 1999
- [11] CHARTON, J. M.: *Eine Linienmethode zur approximativen Lösung inverser Probleme für elliptische Differentialgleichungen*. Universität Siegen, Dissertation, 2004
- [12] CLARKE, D. W. ; MOHTADI, C. ; TUFFS, P. S.: Generalised predictive control - Part I. The basic algorithm. In: *Automatica* 23, Nr. 2, S. 137, 1987
- [13] CUTLER, C. R. ; RAMAKER, B. C.: Dynamic Matrix Control - A Computer Control Algorithm. In: *Proc. Joint Automatic Control Conference, San Francisco (CA), Paper WP5-B* , 1980
- [14] EVANS, D. J.: *Group Explicit Methods For The Numerical Solution Of Partial Differential Equations*. Gordon And Breach Science Publishers Amsterdam, 1997

- [15] EVANS, L. C.: *Partial Differential Equations*. American Mathematical Society, 1998
- [16] FINDEISEN, R.: *Nonlinear Model Predictive Control: A Sampled-Data Feedback Perspective*. Universität Stuttgart, Dissertation, 2004
- [17] FORSTER, O.: *Analysis II*. Vieweg Wiesbaden, 1999
- [18] FORSTER, O.: *Analysis I*. Vieweg Wiesbaden, 2001
- [19] FRAASS, M.: *Meß- und Regelungstechnik II*. Technische Fachhochschule Berlin, Diplomarbeit, 2003
- [20] FURLAN, P.: *Das gelbe Rechenbuch, Band 3: Gewöhnliche Differentialgleichungen, Funktionentheorie, Integraltransformationen, Partielle Differentialgleichungen*. Verlag Martina Furlan Dortmund, 1995
- [21] GARCIA, C. E. ; PRETT, D. M. ; MORARI, M.: Model Predictive Control: Theory and Practice - a Survey. In: *Automatica* 25, Nr. 3, S. 335–348, 1989
- [22] GEBHARDT, A.: *Regelungstechnik*. Fachhochschule Aachen, Skript zur Vorlesung, 2005
- [23] GROSSMANN, C. ; ROSS, H.-G.: *Numerik partieller Differentialgleichungen*. Teubner Stuttgart, 1994
- [24] GRÜNE, L.: *Analysis and design of unconstrained nonlinear MPC schemes for finite and infinite dimensional systems*. Universität Bayreuth, Preprint, 11/2007
- [25] GRÜNE, L.: *Mathematische Kontrolltheorie I: Lineare Systeme*. Universität Bayreuth, Skript zur Vorlesung, 2006
- [26] GRÜNE, L.: *Mathematische Kontrolltheorie II: Nichtlineare Systeme*. Universität Bayreuth, Skript zur Vorlesung, 2006
- [27] GUSTAFSON, K.: *Introduction to partial differential equations and Hilbert space methods*. Wiley New York, 1980
- [28] HANKE-BOURGEOIS, M.: *Grundlagen der Numerischen Mathematik und des Wissenschaftlichen Rechnens*. Teubner Stuttgart, 2002
- [29] HÖRMANN, G. ; SCHERZER, J. ; SUCKOW, F. ; ET. AL.: *Wasserhaushalt von Wald-ökosystemen: Methodenleitfaden zur Bestimmung der Wasserhaushaltskomponenten auf Level II-Flächen*. <http://www.odata.de/download/Methodenband.pdf>, 2003
- [30] ISIDORI, A.: *Nonlinear control systems II*. Springer-Verlag London Ltd., 1999 (Communications and Control Engineering Series)
- [31] KAHLBACHER, M. ; HOXHA, L. ; KANITSAR, M. ; PAK, D.: *Die Wellengleichung*. http://www.ing.unisannio.it/borzi/proj_wellen.pdf, Juni 2004
- [32] KLEISSLER, R.: *Schulung für Speicherprogrammierbare Steuerung: Grundlagen der Steuertechnik*. <http://www.kleissler-online.de/PDF/Steuerungstechnik.pdf>, 2006

-
- [33] KLOKER, M.: *FD-Zeitschrittverfahren höherer Ordnungen für die 1-D-Advektions-Diffusions-Modellgleichung bei $u(x,t)=f(t)\exp(ikx)$* . Universität Bremen, Diagrammkatalogreihe: Lösungseigenschaften von Finite-Differenzen-Verfahren, 2006
- [34] LANGTANGEN, H. P.: *Computational Partial Differential Equations*. Springer-Verlag Berlin, 1999
- [35] LEVEQUE, R. J.: *Finite Difference Methods for Ordinary and Partial Differential Equations*. SIAM Philadelphia, 2007
- [36] LI, R. ; CHEN, Z. ; WU, W.: *Generalized Difference Methods For Differential Equations*. Marcel Dekker New York, 2000
- [37] MEHRMANN, V.: *Kontrolltheorie*. Technische Universität Berlin, Skript zur Vorlesung, 2004
- [38] MORARI, M. ; LEE, J. H.: Model predictive control: past, present and future. In: *Computers and Chemical Engineering* 23, S. 667–682, 1999
- [39] MUNZ, C.-D. ; WESTERMANN, T.: *Numerische Behandlung gewöhnlicher und partieller Differenzialgleichungen*. Springer-Verlag Berlin, 2006
- [40] NAUMANN, J.: *Höhere Analysis II: Sobolev-Räume*. Humboldt-Universität Berlin, Skript zur Vorlesung, 2005
- [41] NOCEDAL, J. ; WRIGHT, J. S.: *Numerical optimization*. Second. Springer-Verlag New York, 2006 (Springer Series in Operations Research and Financial Engineering)
- [42] RICHALET, J. ; RAULT, A. ; TESTUD, J.L. ; PAPON, J.: Model predictive heuristic control: applications to industrial processes. In: *Automatica* 14, Nr. 5, S. 413, 1978
- [43] ROSSITER, J. A.: *Model-based predictive control : A Practical Approach*. CRC Press CLL Boca Raton, 2003
- [44] SCHITTKOWSKI, K.: NLPQL: A Fortran subroutine for solving constrained nonlinear programming problems. In: *Annals of Operations Research* 5, S. 485–500, 1986
- [45] SCHITTKOWSKI, K. ; DAI, Y.: A sequential quadratic programming algorithm with non-monotone line search. In: *Pacific Journal of Optimization* 4, S. 335–351, 2008
- [46] SIMADER, C.: *Lineare Funktionalanalysis I*. Universität Bayreuth, Skript zur Vorlesung, 1995
- [47] SONNEBORN, L. ; VLECK, F. V.: The Bang-Bang Principle for Linear Control Systems. In: *SIAM Journal on Control and Optimization* 2, S. 151–159, 1965
- [48] THEISSEN, K.: *Optimale Steuerprozesse unter partiellen Differentialgleichungs-Restriktionen mit linear eingehender Steuerfunktion*. Wilhelms-Universität Münster, Dissertation, 2006
- [49] THOMAS, J. W.: *Numerical Partial Differential Equations: Finite Difference Methods*. Springer-Verlag Berlin, 1995

- [50] TRACHT, R.: *Regelungstechnik*. Universität Essen, Skript zur Vorlesung, 2002
- [51] TRÖLTZSCH, F.: *Optimale Steuerung partieller Differentialgleichungen*. Vieweg Wiesbaden, 2005
- [52] TRÖLTZSCH, Fredi: *Partielle Differentialgleichungen*. Technische Universität Berlin, Skript zur Vorlesung, 2002
- [53] VINTER, R.: *Optimal Control*. Birkhäuser Boston, 2000
- [54] VOIT, H.: *MPC Schemata mit variablem Kontrollhorizont*. Universität Bayreuth, Diplomarbeit, 2008
- [55] WERNER, D.: *Funktionalanalysis*. Springer-Verlag Berlin, 2004
- [56] ZERZ, E.: *Kontrolltheorie I*. Universität Aachen, Skript zur Vorlesung, 2007

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen.

Neuhaus, den 29. September 2008

.....
Stefan Trenz