

INAUGURAL - DISSERTATION  
zur  
Erlangung der Doktorwürde  
der  
Naturwissenschaftlich-Mathematischen Gesamtfakultät  
der  
Ruprecht - Karls - Universität  
Heidelberg

vorgelegt von  
Diplom-Mathematiker Andreas A. S. Schäfer  
aus Wiesbaden

Tag der mündlichen Prüfung: 21. Dezember 2004



**Effiziente reduzierte Newton-ähnliche Verfahren zur  
Behandlung hochdimensionaler strukturierter  
Optimierungsprobleme mit Anwendung bei  
biologischen und chemischen Prozessen**

Gutachter: Prof. Dr. Dr. h. c. Hans Georg Bock  
Prof. Dr. Rolf Rannacher



## Abstract

A new direct multiple shooting method for multistage constrained optimization problems with large-scale differential-algebraic process models is presented. It is advantageous in case of optimization problems with many state variables but few degrees of freedom, as they arise often in optimal control and parameter identification of spatially discretized partial differential-algebraic equation (PDAE) models. By a piecewise control parameterization and a multiple shooting state parameterization on the same grid we solve the resulting multipoint boundary value problem with specially tailored partially reduced Newton-type methods. The present work provides an extension of the direct multiple shooting approach for DAE models of index one (Bock and Plitt 1984, [20] and Leineweber 1999, [57]; Bock 1987, [16] and Schlöder 1988, [82]) to problems with many states but few degrees of freedom due to e. g. initial conditions on the states or a low number of control and global parameters.

Emphasis is put on reducing the number of directional derivatives in the Newton-type methods to speed up computation of the Newton iterates. It is shown that the number of directional derivatives is independent of the state dimension. This can be achieved by projection onto the reduced space of control and global parameters. Intertwining of algorithmic differentiation for the model equations and Internal Numerical Differentiation can be used to efficiently set up the reduced QPs of the Newton-type optimization algorithms. It is shown that the new methods are well suited for offline optimization and also for online purposes in the context of Nonlinear Model Predictive Control.

The performance of the new methods is demonstrated by multistage optimal control applications from the literature and applications from (bio-)chemical engineering:

1. Parameter estimation of the in-vitro drug release of a gum implant modelled by an instationary reaction-diffusion 1D-PDE.
2. Optimal setpoint control of a continuous distillation column modelled by a large-scale stiff DAE (maintaining purity requirements).
3. Parameter estimation and optimal control of capacity maximization of a catalytic tube reactor modelled by an instationary convection-diffusion 2D-PDE (cooperation with Bayer, Leverkusen).

The first application is scaled in the space discretization mesh. Different IND approaches, model implementations and reduced space approaches are compared. The second application serves as a benchmark problem in the offline and online context. A comparison to a state-of-the-art real-time optimization algorithm (Diehl 2002, [28]) is presented. The last application shows the suitability of the new methods for industrial large-scale PDAE constrained optimization. It is shown that the optimal capacity can be increased by 12 % together with a 50 % reduction of the amount of catalyst.

## Keywords

Large-Scale Optimal Control and Parameter Identification, Instationary Partial Differential-Algebraic Equations, Direct Multiple Shooting, Reduced Newton-type Methods, Directional Derivatives, Internal Numerical Differentiation, Algorithmic and Symbolic Differentiation, Non-linear Model Predictive Control, Drug Release, Distillation Column, Catalytic Tube Reactor.

## Zusammenfassung

Ein neues direktes Mehrfach-Schießverfahren für beschränkte Mehrstufen-Optimierungsprobleme mit hochdimensionalen differentiell-algebraischen Prozeß-Modellen wird präsentiert. Insbesondere für Optimierungsprobleme mit vielen Zustandsvariablen aber wenigen Freiheitsgraden ist dieses Verfahren geeignet. Diese kommen beispielsweise bei Optimal-Steuerungs- und Parameteridentifizierungsproblemen mit örtlich diskretisierten instationären partiellen differential-algebraischen Gleichungsmodellen (PDAEs) vor. Durch stückweise Steuerungsparameterisierung und Mehrziel-Zustandsparameterisierung auf demselben Gitter wird das resultierende Mehrpunkt-Randwertproblem mit speziell angepaßten partiell reduzierten Newton-ähnlichen Methoden gelöst. Die Methoden stellen eine Erweiterung der direkten Mehrfach-Schießverfahren für DAE-Modelle vom Index 1 dar (siehe Bock und Plitt 1984, [20] und Leineweber 1999, [57]; Bock 1987, [16] und Schlöder 1988, [82]).

Essentiell ist die Reduktion der Anzahl der Richtungsableitungen in den Newton-ähnlichen Methoden, um die Berechnung der Newton-Iterierten zu beschleunigen. Bei den hier betrachteten Problemen wird gezeigt, daß die Zahl der Richtungsableitungen unabhängig von der Zustandsdimension ist. Dies kann durch Projektion auf den Raum der Steuerungen und globalen Parametern erreicht werden. Dazu wird die algorithmische Differentiation für die Modell-Gleichungen und die Interne Numerische Differentiation miteinander kombiniert, um die reduzierten QPs der Newton-ähnlichen Optimierungsalgorithmen effizient aufzusetzen. Es wird gezeigt, daß die neuen Methoden sowohl für den Offline-Einsatz als auch für den Online-Einsatz im Rahmen der Nichtlinearen Modellprädiktiven Regelung besonders geeignet sind.

Anhand von Mehrstufen-Optimal-Steuerungsproblemen aus der Literatur und Anwendungen aus der (bio-)chemischen Verfahrenstechnik wird die Performance der neuen Methoden demonstriert:

1. Parameterschätzung einer in-vitro Wirkstoff-Freisetzung eines Zahnfleisch-Implantats modelliert durch eine instationäre eindimensionale Reaktions-Diffusions-PDE.
2. Optimal-Steuerung (Einhaltung von Reinheitsbedingungen) einer kontinuierlichen Destillationskolonne modelliert durch eine hochdimensionale steife DAE.
3. Parameterschätzung und Optimal-Steuerung (Durchsatzmaximierung) bei einem katalytischen Rohrreaktor modelliert durch eine instationäre zweidimensionale Konvektions-Diffusions-PDE (Kooperation mit Bayer, Leverkusen).

Die erste Anwendung ist in der Ortsdiskretisierung skaliert und dient zum Vergleich der IND-Ansätze, der Modell-Implementationen und der Reduktionsansätze. Die zweite Anwendung stellt ein Benchmark-Problem für die neuen Methoden im Offline- und Online-Zusammenhang dar. Ein Vergleich zu State-of-the-Art Echtzeit-Optimierungsmethoden (siehe Diehl 2002, [28]) wird angegeben. Die letzte Anwendung zeigt die Eignung der neuen Methoden für industrielle hochdimensionale PDAE-beschränkte Optimierungsprobleme. Es wird gezeigt, daß der Durchsatz um 12 % gesteigert werden kann, zusammen mit einer Reduktion des Katalysatorgehalts um 50 %.

## Stichworte

Hochdimensionale Optimal-Steuerung und Parameteridentifizierung, instationäre PDAE, direktes Mehrfach-Schießverfahren, reduzierte Newton-ähnliche Verfahren, Richtungsableitungen, Interne Numerische Differentiation, Algorithmische und Symbolische Differentiation, Nichtlineare Modellprädiktive Regelung, Wirkstoff-Freisetzung, Destillationskolonne, katalytischer Rohrreaktor.

# Inhaltsverzeichnis

<b>Einleitung</b>	<b>1</b>
<b>1 Problemformulierung</b>	<b>6</b>
1.1 Formulierung eines Mehrstufen-Optimal-Steuerungsproblems in DAEs . . .	6
1.1.1 Erweiterung auf parabolische PDEs . . . . .	7
1.2 Beispiele aus der chemischen Verfahrenstechnik . . . . .	8
1.3 Direkter Ansatz . . . . .	11
1.3.1 Stückweise Parameterisierung der Steuerung . . . . .	11
1.4 Simultaner Ansatz . . . . .	12
1.4.1 Ein Mehrfach-Schießverfahren für DAEs . . . . .	13
1.4.2 Diskretisierung der Steuerungs- und Pfadbeschränkungen . . . . .	13
1.5 Struktur des diskretisierten und parameterisierten Problems . . . . .	15
<b>2 Grundlegende Lösungsstrategien für Nichtlineare Programme</b>	<b>16</b>
2.1 Klassifikation der endlich-dimensionalen Probleme . . . . .	16
2.2 Optimalitätsbedingungen . . . . .	17
2.2.1 Bedingungen erster Ordnung . . . . .	18
2.2.2 Bedingungen zweiter Ordnung . . . . .	19
2.3 Iterative Methoden basierend auf linear-quadratischen Approximationen . .	20
2.3.1 Lokale Konvergenz . . . . .	24
2.3.2 Globalisierung der Konvergenz . . . . .	27
2.3.3 Adaptive Genauigkeitssteuerung . . . . .	31
<b>3 Ein speicherplatz- und rechenzeitsparender reduzierter SQP-Algorithmus</b>	<b>34</b>
3.1 Übersicht über partiell reduzierte SQP-Verfahren . . . . .	34
3.2 Ein reduzierter SQP-Algorithmus für Probleme mit wenigen Freiheitsgraden	37
3.2.1 Rekursive Berechnung der Koordinaten-Basis . . . . .	39
3.2.2 Hessematrix-Updates bei allgemeinen Optimierungsproblemen . . .	41
3.2.3 Gauß-Newton Hessematrix für Least-Squares Probleme . . . . .	42
3.2.4 Aufwandsvergleich je nach Reduktionsstufe . . . . .	43
3.3 Lokale Konvergenzuntersuchungen . . . . .	44
3.4 Globalisierungstechniken für reduzierte SQP-Algorithmen . . . . .	45
3.4.1 Liniensuchmethoden mit Straffunktionen . . . . .	46
3.4.2 Liniensuchmethoden mit Natürlichen Niveaufunktionen . . . . .	47

3.5	Algorithmische Zusammenfassung der reduzierten direkten Mehrfach-Schießverfahren . . . . .	47
<b>4</b>	<b>Berechnung von Ableitungen</b>	<b>50</b>
4.1	Interne Numerische Differentiation . . . . .	51
4.1.1	Direkte Berechnung einer Lösung der Richtungs-VDAE . . . . .	52
4.1.2	Iterative Berechnung einer Lösung der Richtungs-VDAE . . . . .	53
4.2	Berechnung der Modell-Jacobimatrizien . . . . .	53
4.2.1	C/Adol-C: automatische Differentiation . . . . .	53
4.2.2	gPROMS: symbolische Differentiation . . . . .	55
4.3	Anwendung auf Koordinaten-Basis . . . . .	55
<b>5</b>	<b>Parameterschätzung bei einem Zahnfleisch-Implantat</b>	<b>57</b>
5.1	Modell . . . . .	58
5.2	Aufsetzen der Parameterschätzung und Ergebnis . . . . .	59
5.3	Vergleich der IND-Ansätze zur Berechnung der Sensitivitäten . . . . .	61
5.4	Vergleich zum Vollraum-Ansatz . . . . .	68
5.5	Vergleich der Modell-Implementationen (gPROMS - C/Adol-C) . . . . .	70
<b>6</b>	<b>Optimierung einer kontinuierlichen Destillationskolonne</b>	<b>72</b>
6.1	Modell . . . . .	72
6.2	Aufsetzen eines Offline-Optimierungsproblems und Ergebnis . . . . .	76
6.3	Vergleich zu MUSCOD-II . . . . .	77
<b>7</b>	<b>Parameterschätzung und Optimierung eines katalytischen Rohrreaktors</b>	<b>81</b>
7.1	Modell . . . . .	82
7.2	Simulation . . . . .	87
7.3	Parameterschätzung . . . . .	88
7.4	Optimierungsszenarien . . . . .	89
<b>8</b>	<b>Zusammenfassung und Ausblick</b>	<b>92</b>
<b>A</b>	<b>Zusätzliche Informationen</b>	<b>95</b>
A.1	Zahnfleisch-Implantat . . . . .	95
A.2	Kontinuierliche Destillationskolonne . . . . .	97
A.3	Katalytischer Rohrreaktor . . . . .	99
	<b>Literaturverzeichnis</b>	<b>102</b>

## Danksagung

An dieser Stelle bedanke ich mich bei meinen Betreuern Prof. Dr. Dr. h. c. Hans Georg Bock und Dr. Johannes P. Schlöder, Interdisziplinäres Zentrum für Wissenschaftliches Rechnen (IWR), Universität Heidelberg, für die Unterstützung und für die Diskussionen im Umfeld der dynamischen Prozeß-Simulation und Optimierung, bei denen die neuen Methoden in dieser Arbeit entstanden sind. Insbesondere stellte die Entwicklung der Optimal-Steuerungssoftware MUSCOD-II die Grundlage für die Implementation der neuen Methoden dar.

Mehrere Kollegen haben ebenfalls zu dem Erfolg dieser Arbeit beigetragen. Dr. Moritz Diehl danke ich für seine wertvollen Tips und Diskussionen im Umfeld von MUSCOD-II und sein stetes Bemühen, neue Anwendungsprobleme zu erschließen, von denen auch ich profitieren konnte. Insbesondere danke ich ihm – als sein Zimmerkollege – für seine inspirierende Atmosphäre, die von ihm ausgeht. Weiterhin danke ich Ulrich Brandt-Pollmann, mit dem zusammen Teile der Ableitungsgenerierung entstanden sind und Dr. Irene Bauer für die Kooperation bei Fragen um den Integrator DAESOL.

Ein besonderer Dank geht an Dr. Daniel Leineweber, der mich an der Entwicklung von MUSCOD-II teilnehmen ließ und durch den ich sowohl softwaretechnische Aspekte als auch algorithmische Konzepte erlernen konnte. Durch seine spätere Tätigkeit bei Bayer, Leverkusen, konnten zahlreiche industrielle Anwendungsprojekte initiiert werden. Ihm und seinen Kollegen Dr. Peter Lehner und Dr. Thomas Turek danke ich auch für die Unterstützung im Rohrreaktor-Projekt, insbesondere für das Modell, das ich an realistische Meßdaten anpassen konnte, und das Aufsetzen der Optimierungsszenarien. Dr. Christian Schulz von Process Systems Enterprise, London, und Dr. Rainer Perne sowie Dr. Andreas Schuppert von Bayer, Leverkusen, danke ich für die Unterstützung bei der Kopplung der Modellierungsumgebung gPROMS mit der Optimal-Steuerungssoftware MUSCOD-II.

Den Direktoren des IWR danke ich für die exzellente Umgebung, die an der Universität Heidelberg für interdisziplinäre Forschung geschaffen wurde. Insbesondere gebührt unserer Sekretärin Margret Rothfuß großer Dank für ihren Beitrag zu einer freundlichen Atmosphäre in der Arbeitsgruppe. Des weiteren danke ich dem Sonderforschungsbereich 359 „Reaktive Strömungen, Diffusion und Transport“ für die finanzielle Unterstützung.

Mein persönlicher Dank gilt meinen Eltern Ute und Hans-Dieter Schäfer und meiner Freundin Michaela Schneider für ihre Liebe und fortwährende Unterstützung.

# Einleitung

Durch Fortschritte in den Numerischen Methoden und der Hardware-Umgebung nimmt die Bedeutung der Simulation von detaillierteren (instationären) Partiellen Differential-Algebraischen Gleichungen (PDAEs) in den Ingenieurwissenschaften stetig zu. Damit verbunden ist auch die Anpassung der Modelle an Meßdaten des zu beschreibenden Prozesses sowie Optimales Design und Optimale Fahrweise des Prozesses. Dieses wiederum ist der Ausgangspunkt für PDAE-beschränkte kontinuierliche Ausgleichs- und Optimal-Steuerungsprobleme. Um diese in der Praxis zu lösen, sind schnelle und robuste Algorithmen notwendig. Grundlegende Methoden sind der Variationsansatz, Ansatz über Dynamische Programmierung oder über Nichtlineare Programmierung (NLP). Der NLP-Ansatz wird auch als *direkter Ansatz* bezeichnet, weil durch Parameterisierung der Steuerungen das kontinuierliche (unendlich-dimensionale) Ausgangsproblem in ein NLP-Problem mit einer endlichen Zahl von Steuerungsparametern transformiert wird. Die Lösung dieses NLP-Problems stellt je nach Parameterisierung eine suboptimale Lösung des Ausgangsproblems dar, welche jedoch in der Regel gute Startwerte für den indirekten Ansatz (Pontryagin'sches Maximumprinzip) darstellen würde. Die direkten Ansätze unterscheiden sich in Parameterisierung der Steuerung und (gleichzeitiger) Diskretisierung bzw. Parameterisierung der Modellgleichungen. Dabei ist eine ähnliche Entwicklung wie bei DAE-beschränkten Optimierungsproblemen festzustellen. Diese reicht von den *sequentiellen Ansätzen*, die das Modell als Black-Box betrachten und den Optimierer um die Simulation herum konstruieren, hin zu den *simultanen Ansätzen*, welche Simulation und Optimierung miteinander verzahnen. Der sequentielle Ansatz ist nur geeignet für wenige Entscheidungsvariablen (Steuerungen und Parameter), da Gradienten für jede Entscheidungsvariable durch wiederholtes Lösen der PDAE gewonnen wird (z.B. durch finite Differenzen). Der simultane Ansatz wiederum diskretisiert die Modell-Gleichungen explizit und nimmt die diskretisierten Gleichungen in die NLP-Formulierung auf. Dadurch wird z. B. die iterative Lösung der linearisierten PDAE Teil des NLP-Algorithmus. Dieses NLP-Problem wird nicht nur in den Entscheidungsvariablen sondern auch in den Zustandsvariablen gelöst, so daß gewöhnlich Optimierung und Zulässigkeit (Lösung der Modellgleichungen) *simultan* erfolgen, wenn ein unzulässig iterierender NLP-Löser verwendet wird. Durch diese Verzahnung läßt sich der Gesamtaufwand erheblich reduzieren.

Ein typischer Vertreter von sequentiellen Strategien sind *direkte Einfach-Schießverfahren*, wobei zuerst die Steuerungen parameterisiert werden und die kontinuierlichen Zustandsvariablen durch numerische Integration berechnet *und* aus dem NLP-Problem eliminiert werden. Somit sind die Zustandsvariablen im Fall eines Anfangswertproblems während der NLP-Iterationen zulässig. Methoden für gewöhnliche Differentialgleichungen (ODEs)

sind in Hicks und Ray (1971, [48]), Sargent und Sullivan (1978, [80]) und Kraft (1985, [52]) zu finden und für DAE-Modelle in Morison und Sargent (1986, [64]) und Vassiliadis *et al.* (1993, [96] und [97]).

Demgegenüber stehen simultane Strategien wie das *direkte Mehrfach-Schießverfahren*, welches sowohl die Steuerung als auch die Zustandstrajektorie auf einem Mehrzielgitter parameterisiert. Das entstehende Mehrpunkttrandwertproblem wird in den parameterisierten Zustands- *und* Entscheidungsvariablen gelöst, wobei zusätzliche Bedingungen (Stetigkeits- bzw. Konsistenzbedingungen) die Zulässigkeit der NLP-Lösung garantieren. Dieser Ansatz wurde für ODE-Modelle zuerst von Plitt (1981, [73]) und Bock *et al.* (1984, [20]) eingeführt und später durch Bock *et al.* (1988, [18], 1999, [17]) sowie Leineweber *et al.* (1995, [56], 1999, [57], 2003, [58] und [59]) auf DAE-Modelle erweitert. Weitere Anwendungen dieses Ansatzes sind bei Tanarkit und Biegler (1995, [85], 1996, [86]), Petzold *et al.* (1997, [72] und 2000, [42]), Hinsberger *et al.* (1996, [50] und 1998, [49]) zu finden.

Eine weitere simultane Strategie stellt die vollständige Diskretisierung der Zustandstrajektorie durch *Kollokation* dar. Dadurch findet die Diskretisierung vollständig auf NLP-Ebene statt, jedoch führt diese für stark nicht-lineare (steife) Modellgleichungen auf sehr große Systeme. Daher erscheinen für diesen Diskretisierungstyp spezielle (rekursive) Zerlegungstechniken sowie eine adaptive Genauigkeitssteuerung notwendig, um effizient zu sein. Anwendung dieser Diskretisierungsmethode finden sich bei Tsang *et al.* (1975, [90]), Bär (1983, [4]), Biegler (1984, [11]), Cuthrell und Biegler (1989, [26]), Logsdon und Biegler (1992, [61]) und Schulz (1996, [83]).

In der Abbildung 1 sind Charakteristiken der genannten sequentiellen und simultanen Ansätze gezeigt. Während beim direkten Einfach- und Mehrfach-Schießverfahren Adaptivität in den Integrator verlagert werden kann, muß dies beim Kollokationsansatz direkt auf NLP-Ebene geschehen. Vom Aufwand (Anzahl der Variablen) scheint es, als hätten die simultanen Strategien einen Nachteil gegenüber den sequentiellen Strategien. Dies trifft jedoch nur zu, wenn inhärente Strukturen in der jeweiligen NLP-Formulierung nicht ausgenutzt werden. Die Vorteile liegen in der Wahl der Startdaten und der besseren Konvergenzeigenschaften (siehe Bock 1987, [16]).

Ziel dieser Arbeit ist, für den direkten Mehrfach-Schießansatz einen strukturausnutzenden und effizienten NLP-Löser zu entwickeln, der bei wenigen Freiheitsgraden nur den Aufwand eines direkten Einfach-Schießverfahrens und die Vorteile des Mehrfach-Schießverfahrens hat. Dieses wird durch neue partiell reduzierte Newton-ähnliche Methoden möglich, welche das Aufstellen der vollen (teuren) Sensitivitätsmatrizen der Zustandsvariablen vermeiden und das einfache Einbinden von Ungleichungen erlauben. Im Rahmen von Gauß-Newton-Methoden für große beschränkte Parameterschätzprobleme ist dies von Schlöder (1988, [82]) erfolgreich eingeführt worden. Hauptidee ist, die Anzahl der Richtungsableitungen im reduzierten QP einer Newton-Iteration erstens soweit zu verringern, daß sie unabhängig von der Anzahl der Zustandsvariablen ist und zweitens diese direkt zu berechnen, um damit das Aufstellen der vollen Sensitivitätsmatrizen zu vermeiden. Diese Anforderungen sind notwendig, wenn im Ort diskretisierte instationäre PDAEs betrachtet werden sollen. Algorithmisch geschieht dabei eine Projektion des großen strukturierten NLP-Problems auf den reduzierten Raum von Steuerungen und Parametern,

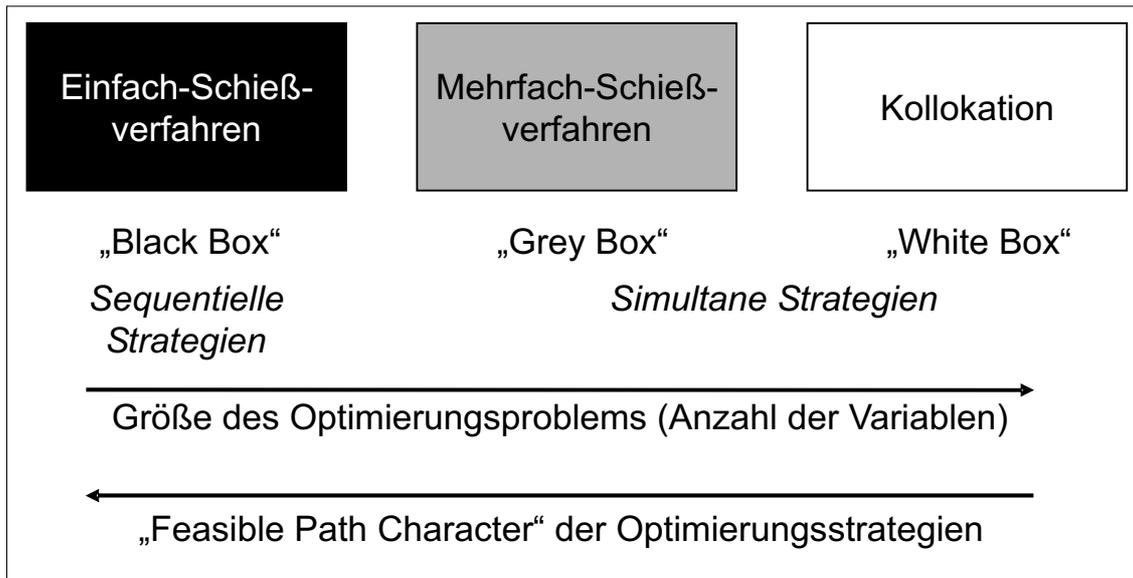


Abbildung 1: Eigenschaften der direkten NLP-Ansätze für DAE beschränkte Optimierungsprobleme

indem Konsistenz-, Anfangs- und Stetigkeitsbedingungen ausgenutzt werden. Durch Verbinden der Projektion und Gradientenauswertung können die Richtungsableitungen direkt berechnet werden. Abgestimmt auf die Anzahl der Richtungsableitungen in den Sensitivitäten wurden verschiedene Ansätze entwickelt, die alle das Prinzip der Internen Numerischen Differentiation (IND-Prinzip) erfüllen. Dieses von Bock (1981, [14]) eingeführte Prinzip ermöglicht eine effiziente und zugleich genaue Berechnung der Sensitivitäten und hat sich in der Praxis bewährt. Im Vergleich zu dem direkten Mehrfach-Schießverfahren von Petzold *et al.* (1997, [72] und 2000, [42]) welche eine ähnliche Strukturausnutzung des NLP-Problems vornehmen, müssen jedoch keine konsistenten Startwerte vorliegen, da in jeder Iteration ein *relaxiertes dynamisches System* gelöst wird (siehe Bock *et al.* 1988, [18]). Die Eigenschaft der Newton-ähnlichen Verfahren, unzulässige Iterierte zu erzeugen, wird hierdurch besser ausgenutzt, welches unter anderem in der Regel bessere Konvergenzeigenschaften zur Folge hat. Bei der Erweiterung der direkten Schießverfahren von DAEs auf parabolische PDAEs spielt Adaptivität eine besondere Rolle, jedoch beinhaltet bei Anwendung der Linienmethode eine Änderung des Ortsgitters auch eine Interpolation der Zustandsvariablen samt Integratorinformationen. Erfahrungen zeigen, daß das Ortsgitter nicht häufig adaptiert werden sollte, um Ineffizienzen des Integrators z. B. durch Restart und/oder Ordnungsreduktion zu vermeiden (siehe Nowak *et al.* 2003, [69] und Li *et al.* 2003, [60]). Im Zusammenhang mit Optimal-Steuerungsproblemen zeigt sich dieses Problem auch bei Quasi-Newton-Methoden im Update der Hessematrix. Auch hier ist bei erfolgter Ortsadaptierung ein Restart erforderlich. Bei den in dieser Arbeit vorgestellten direkten Mehrfach-Schießverfahren findet eine adaptive Genauigkeitssteuerung über die Zeitdiskretisierung des Integrators und über die Integratorgenauigkeit im NLP-Löser statt. Die Ortsdiskretisierung wird durch den Benutzer vorgenommen.

# Erläuterung der Kapitel dieser Arbeit

In Kapitel 1-4 werden die theoretischen Grundlagen der Newton-ähnlichen Methoden innerhalb des direkten Mehrzielansatzes vorgestellt. Die Beispiele aus dem Bereich der (bio-)chemischen Verfahrenstechnik erläutern die Eigenschaften der partiell reduzierten Newton-ähnlichen Methoden in Kapitel 5-7. Insbesondere die Eignung für Offline- und Online-Fragestellungen wird anhand eines Beispiels erläutert. In Kapitel 8 werden die (theoretischen wie praktischen) Ergebnisse zusammengefasst. Im Anhang A sind zusätzliche Informationen zu den beschriebenen Anwendungen aufgeführt. Für die einzelnen Kapitel heißt dies:

- Kapitel 1: Formulierung der Klasse von Mehrstufen-Optimierungsprobleme, die durch die neuen direkten Mehrfach-Schießverfahren gelöst werden können. Anhand von Anwendungsbeispielen werden die Vorteile der Mehrstufenformulierung aufgezeigt. Weiterhin werden die Schritte bei der Parameterisierung und Diskretisierung von DAEs und parabolischen PDEs beschrieben.
- Kapitel 2: Es werden die Optimalitätsbedingungen und Newton-ähnliche Methoden für Ausgleichs- und allgemeine NLP-Probleme beschrieben. Lokale und globale Konvergenz sowie Adaptivität hinsichtlich der Genauigkeit der Residuen und Jacobimatrizen werden diskutiert. Dies stellt die Grundlage für die partiell reduzierten Newton-ähnlichen Methoden aus Kapitel 3 dar.
- Kapitel 3: Es werden die partiell reduzierten Newton-ähnlichen Verfahren zur Lösung des strukturierten NLP-Problems aus Kapitel 1 beschrieben. Erläutert werden die Rekursionen zur Berechnung der Koordinaten-Basis und die lokalen Konvergenzeigenschaften der Newton-ähnlichen Verfahren. Spezielle Globalisierungsstrategien werden für den Parameterschätz- und Optimal-Steuerungsfall angegeben.
- Kapitel 4: Es werden IND-Ansätze zur Berechnung der Sensitivitäten sowie der Modellableitungen durch Automatische bzw. Symbolische Differentiation beschrieben. Speziell die Richtungen der Variationsdifferentialgleichung werden für die Koordinaten-Basis angegeben. Die effiziente Berechnung der Richtungsableitungen stellt eine wesentliche Grundlage für effiziente Newton-ähnliche Methoden aus Kapitel 3 dar.
- Kapitel 5: Anhand der Parameterschätzung zur Wirkstoff-Freisetzung in einem Zahnfleisch-Implantat werden verschiedene Modell-Implementationen, IND-Ansätze und (reduzierte) Gauß-Newton-Verfahren miteinander verglichen. Durch die Ortsdiskretisierung des instationären 1D-PDE-Modells fand eine Skalierung der Anwendung statt.
- Kapitel 6: Die Optimierung einer kontinuierlichen Destillationskolonne wiederum ermöglicht einen Vergleich von unterschiedlichen Reduktionsstrategien und IND-Strategien und zeigt Anwendungsfelder im Online- und Offline-Bereich auf.
- Kapitel 7: Anhand eines instationären 2D-PDE Modells eines katalytischen Rohrreaktors der Firma Bayer, Leverkusen, erfolgt eine Schätzung der kinetischen Parameter und eine Studie über den Einfluß der Katalysatoraktivität anhand von unter-

schiedlichen Optimierungsszenarien. Sowohl Parameterschätzung als auch Optimal-Steuerung konnten erfolgreich mit Hilfe der neu entwickelten direkten Mehrfach-Schießverfahren durchgeführt werden.

- Kapitel 8: Zusammenfassung der wesentlichen Merkmale des neuen direkten Mehrfach-Schießverfahrens und Ausblick auf mögliche Erweiterungen.

# Kapitel 1

## Problemformulierung

In diesem Kapitel stellen wir eine Klasse von Optimierungsproblemen vor, die mit dem neu entwickelten Algorithmus (siehe Kapitel 3) gelöst werden können. Insbesondere die Mehrstufenformulierung, wie sie auch schon in MUSCOD-II (siehe Leineweber 1999, [57]; 2003, [58] und [59]) implementiert wurde, ermöglicht eine sehr vielseitige Anwendbarkeit.

### 1.1 Formulierung eines Mehrstufen-Optimal-Steuerungsproblems in DAEs

Viele Probleme der dynamischen Prozeßoptimierung können als Mehrstufen-Optimierungsproblem formuliert werden. In dieser Arbeit wird der folgende Typ von  $M$ -Stufen Optimierungsproblemen betrachtet, bei dem der Zeithorizont  $[t_0, t_M]$  in  $M$  Modellstufen mit Zeitintervallen  $[t_i, t_{i+1}]$ ,  $i = 0, 1, \dots, M-1$ , unterteilt wird:

$$\min_{x_i, z_i, u_i, p, t_i} \sum_{i=0}^{M-1} \phi_i(x_i(t_{i+1}), z_i(t_{i+1}), p, t_{i+1}) \quad (1.1a)$$

unter Berücksichtigung der DAE-Modellstufen (vom Index 1)

$$\left. \begin{aligned} B_i(\cdot) \dot{x}_i(t) &= f_i(x_i(t), z_i(t), u_i(t), p, t) \\ 0 &= g_i(x_i(t), z_i(t), u_i(t), p, t) \end{aligned} \right\}, \quad t \in [t_i, t_{i+1}], \quad i = 0, 1, \dots, M-1, \quad (1.1b)$$

der Anfangsbedingungen an  $x_0$

$$d(x_0, z_0, p) = 0, \quad (1.1c)$$

der Kontroll- und Pfadbeschränkungen

$$h_i(x_i(t), z_i(t), u_i(t), p, t) \geq 0, \quad t \in [t_i, t_{i+1}], \quad i = 0, 1, \dots, M-1, \quad (1.1d)$$

der Stufenübergangsbedingungen

$$x_{i+1}(t_{i+1}) = c_i(x_i(t_{i+1}), z_i(t_{i+1}), p), \quad i = 0, 1, \dots, M-2, \quad (1.1e)$$

und der linear gekoppelten Mehrpunktrandbedingungen

$$\sum_{i=0}^{M-1} \left( r_i^s(x_i(t_i), z_i(t_i), p, t_i) + r_i^e(x_i(t_{i+1}), z_i(t_{i+1}), p, t_{i+1}) \right) \left\{ \begin{array}{l} = \\ \geq \end{array} \right\} 0. \quad (1.1f)$$

Die Zielfunktion (1.1a) vom verallgemeinerten Mayertyp wird bzgl. der differentiellen und algebraischen Zustandsvariablen  $x_i$  und  $z_i$ , der Steuerungsprofile  $u_i$ , des zeitunabhängigen globalen Parametervektors  $p$  und der Modellstufengitterpunkten  $t_i$  minimiert. Die Vektoren  $x_i(t)$ ,  $z_i(t)$ ,  $u_i(t)$  und  $p$  haben die Dimension  $n_i^x$ ,  $n_i^z$ ,  $n_i^u$  und  $n^p$ .

### 1.1.1 Erweiterung auf parabolische PDEs

Viele Probleme aus dem Bereich reaktive Strömungen, darunter auch der in Kapitel 7 vorgestellte instationäre katalytische Rohrreaktor, können durch folgenden Typ von parabolischen partiellen Differentialgleichungen modelliert werden:

$$\partial_t y - \nabla(D\nabla y) + q\nabla y = f_1(x, y, z, p, t), \quad (1.2a)$$

$$\partial_t x = f_2(x, y, z, p, t), \quad (1.2b)$$

$$0 = g(x, y, z, p, t) \quad (1.2c)$$

mit weiteren Rand- und Anfangsbedingungen. Dabei wird zwischen differentiellen Variablen unterschieden, die räumlich und zeitlich variabel sind (Bezeichnung  $y$  mit Dimension  $n^y$ ) und solche die nur zeitlich variabel sind (Bezeichnung  $x$ ). Im obigen Fall ist die Gleichung 1.2a vom Konvektions-Diffusions-Typ. Die Modellgleichungen des Zahnfleisch-Implantats aus Kapitel 5 sind vom Reaktions-Diffusions-Typ folgender Form:

$$\partial_t y - \nabla(D\nabla y) = f_1(x, y, p, t), \quad (1.3a)$$

$$\partial_t x = f_2(x, y, p, t) \quad (1.3b)$$

mit weiteren Rand- und Anfangsbedingungen. Die klassische Linienmethode führt nun zunächst eine Semidiskretisierung der instationären PDE durch: dabei werden die örtlichen Ableitungen durch ein Diskretisierungsverfahren (z.B. finite Differenzen, Kollokation oder finite Elemente) ersetzt und damit die partiellen Differentialgleichungen 1.2 bzw. 1.3 in nichtlineare, steife, blockstrukturierte DAEs vom Typ 1.1b überführt. Dabei führt eine schlechte Ortsdiskretisierung zu instabilen semidiskreten Systemen, welche sich im Zeitintegrationsprozeß bemerkbar machen können. Aus diesem Grund ist im Fall von bekannten „kritischen“ Bereichen (z.B. steile Gradienten) durch *a priori* Wahl des räumlichen Gitters der Ortsdiskretisierungsfehler klein zu halten, um den Gesamtdiskretisierungsfehler durch die Integrationsgenauigkeit kontrollieren zu können. Für Probleme wo dieses nicht a priori möglich ist, existieren Techniken, welche die Ortsdiskretisierung adaptiv gestaltet: beim *dynamischen Regridding* wird das Ortsdiskretisierungsgitter mit einer *festen* Anzahl von Knoten im Laufe der Zeitintegration mitbewegt. Beim *statischen Regridding* wird durch Einfügen und Entfernen von Knoten auch während der Zeitintegration ein meist lokales Fehlerkriterium minimiert, so daß der Gesamtdiskretisierungsfehler eine vorgegebene Schranke nicht überschreitet. Anwendungen dieser Techniken sind z. B. in Nowak (1993, [68]) für instationäre räumlich eindimensionale PDEs und Li *et al.* (2003, [60]) für instationäre räumlich zweidimensionale Reaktions-Diffusions-PDE beschrieben. Insbesondere wird deutlich, daß beim statischen Regridding die Zeitintegration neu gestartet werden muß, da sich das zugrundeliegende DAE-System ändert. Beiden Typen von Adaptivität ist gemeinsam, daß im Fall von Optimal-Steuerungsproblemen die Quasi-Newton-Methode nach Adaption der Ortsdiskretisierung neu gestartet werden muß. In dieser Arbeit haben wir

deshalb die Ortsdiskretisierung *nicht* adaptiv gewählt, sondern verfolgen stattdessen die adaptive Anpassung der Integrationsgenauigkeit im Laufe der Optimierungsiterationen - siehe auch Abschnitt 2.3.3.

## 1.2 Beispiele aus der chemischen Verfahrenstechnik

Mehrstufenprobleme treten in der Praxis zum Beispiel bei gekoppelten Prozessen auf. Sie sind dadurch definiert, daß dynamische Prozesse, gekoppelt durch Stufenübergangsbedingungen, einem gemeinsamen Optimierungskriterium unterliegen. Durch Mehrpunkttrandbedingungen können prozeßübergreifende Bedingungen formuliert werden. Im folgenden wird demonstriert, wie sich bei dem Modell eines extern gekühlten Batch-Reaktors mit Hilfe des Stufenkonzepts Umschaltunkte der Kühlmittel-Steuerung optimal lokalisieren lassen und wie mit Hilfe von Stufenübergangs- und Mehrpunkttrandbedingungen ein quasi-periodisches Recycling-Problem bei einer Destillationskolonne formuliert und effizient gelöst werden kann. Allgemein können somit auch Modelle mit Unstetigkeiten behandelt werden, bei denen die Schaltstruktur vorher bekannt und fest ist. Es sei betont, daß eine Mehrstufen-Formulierung nicht notwendigerweise zur Kopplung von Prozessen dienen müssen, sondern z. B. zum Aufsetzen einer Multiple-Setpoint-Optimierung dienen, bei der die Prozesse bezüglich der Zustände entkoppelt sind, aber gemeinsame Parameter und Steuerungen besitzen.

### Extern gekühlter Batch-Reaktor

Beim Beispiel eines extern gekühlten Batch-Reaktors (siehe Abbildung 1.1) geht es darum, eine exotherme Reaktion durch Kühlung so zu steuern, daß die Menge des gewünschten Produkts innerhalb eines festen Zeithorizonts (1000 sec) maximiert wird unter Berücksichtigung der Kosten für die Kühlung und Einhaltung einer maximalen Reaktortemperatur. Dieses Optimierungsproblem entstammt dem gPROMS Manual (2004, [34]). Dort

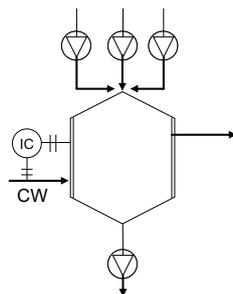


Abbildung 1.1: Schema eines extern gekühlten Batch-Reaktors

wurde zur Lösung des Optimal Steuerungsproblems eine stückweise konstante Steuerung auf äquidistantem Gitter verwendet - dessen optimale Lösung ein Bang-Singulär-Bang-Steuerungsprofil aufweist (siehe Abbildung 1.2). Die suboptimale Lösung läßt sich einerseits durch Verfeinerung der Steuerungsgitters oder aber durch Hinzunahme der Dauern

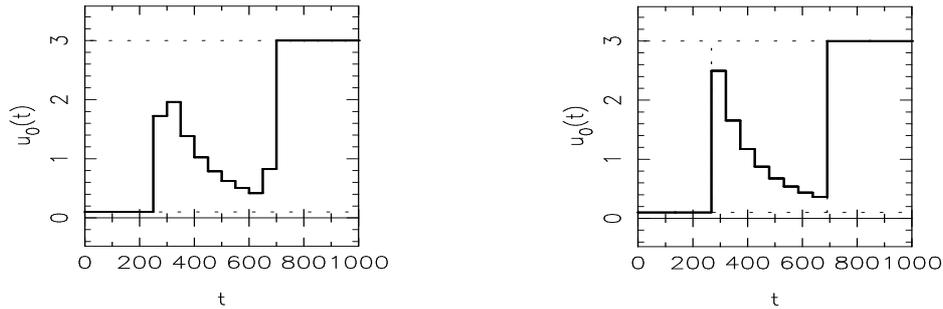


Abbildung 1.2: Steuerung des Kühlmittelzuflusses bei Einstufenformulierung mit 20 äquidistanten Steuerungsintervallen (links) und Mehrstufenformulierung mit 1/8/1 Steuerungsintervallen (rechts)

der Bang-Steuerung als Optimierungsparameter (siehe Abbildung 1.2) verbessern. Damit können beide Umschaltpunkte genau bestimmt werden ohne die Freiheitsgrade im System (Anzahl der Steuerungsintervalle) drastisch erhöhen zu müssen (siehe dazu die Zielfunktionswerte in Tabelle 1.1). Dabei ist die Anzahl der Freiheitsgrade bei der Mehrstufenformulierung (8 freie stückweise konstante Steuerungen plus 2 freie Stufendauern) und der Einstufenformulierung mit 10 Steuerungsintervallen identisch. Das Mehrstufen-

Ansatz	Zielfunktionswert	Umschaltzeitpunkte [sec]
10 Steuerungsintervalle	7.6453	200, 700
20 Steuerungsintervalle	7.6471	250, 700
40 Steuerungsintervalle	7.6478	250, 700
1/8/1 Steuerungsintervalle	7.6490	268, 691

Tabelle 1.1: Statistik der Stufen- und Steuerungsansätze beim Batch-Reaktor

problem bestand dabei aus 3 Instanzen desselben Modells direkt hintereinander gekoppelt (Endwerte der Zustandsvariablen des vorherigen Modells identisch mit Anfangswerten der Zustandsvariablen des nachfolgenden Modells) mit variablen Stufen-Zeithorizonten und fester Gesamtzeit.

## Recycling bei einer Mehrkomponenten-Destillation

Die Fraktionierung eines Mehrkomponentengemisches in ihre Einzelbestandteile mit vorgegebener Reinheit ist das Ziel des nun betrachteten Prozesses einer Batch-Destillationskolonne mit 5 Böden. Das Mehrkomponentengemisch besteht hierbei aus 3 Komponenten (mit Konzentrationen  $X_i$ ). Der Prozeß wird in 4 Phasen unterteilt: zuerst wird die leichteste Komponente in Phase  $P_1$  gewonnen, danach die restlichen Spuren in Phase  $S_1$  abgetrennt, dann die zweitleichteste Komponente in Phase  $P_2$  gewonnen und danach die restlichen Spuren in Phase  $S_2$  abgetrennt. Übrig bleibt die schwerste Komponente im Sumpf (siehe auch Abbildung 1.3). Der Prozess wird gesteuert durch das zeitlich variable Rücklaufverhältnis  $R$  des Kondensats. Im folgenden wird dargestellt, in wie weit es

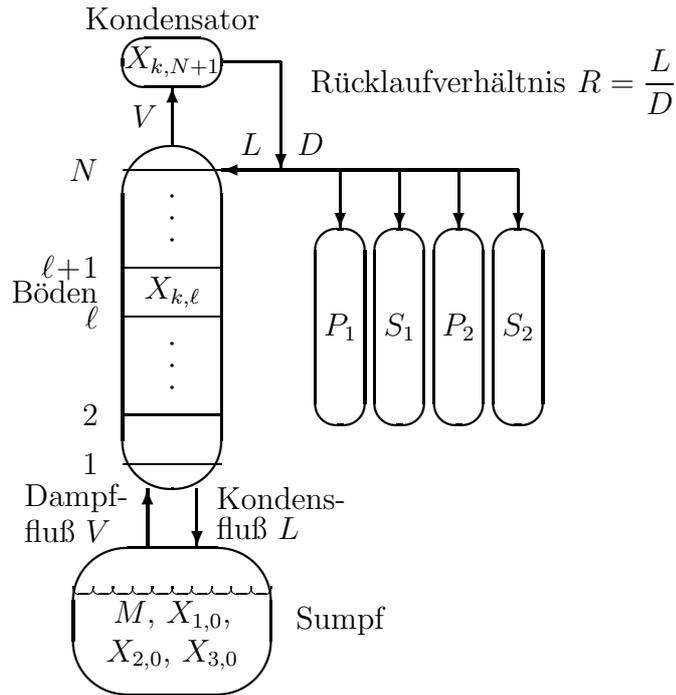


Abbildung 1.3: Schema einer Batch-Destillationskolonne mit Produktfraktionen  $P_1, P_2$  und Abfallfraktionen  $S_1, S_2$

sich lohnt, den Abfall in Phase  $S_1$  und  $S_2$  zu recyceln, anstatt ihn aus dem Prozeß herauszunehmen. Diese Fragestellung wurde von Diehl *et al.* (2002, [30]) untersucht, indem mit Hilfe von Stufenübergangs- und Mehrpunkttrandbedingungen ein quasi-periodisches Recycling-Problem formuliert und gelöst wurde. Um ein teilweises Recycling im Optimierungskontext zu ermöglichen, wurden Recyclingfaktoren  $R_1, R_2 \in [0, 1]$  eingeführt, so daß nur die Abfallmengen  $(1 - R_i) \cdot S_i$ ,  $i = 1, 2$ , aus dem Prozeß herausgenommen werden und die Abfallmengen  $R_i \cdot S_i$ ,  $i = 1, 2$ , zurückgeführt werden: dabei wird die erste Abfallfraktion am Anfang der ersten Produktphase, die zweite Abfallfraktion am Anfang der zweiten Produktphase zugeführt. Das Schema dazu ist in Abbildung 1.4 gezeigt. Die Stufenübergangsbedingungen (ST) dienen dabei zur Beschreibung der hinzugefügten

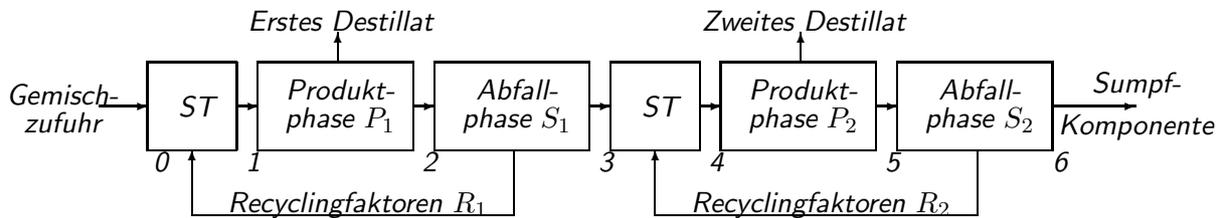


Abbildung 1.4: Schema der Mehrstufenformulierung einer Batch-Destillationskolonne

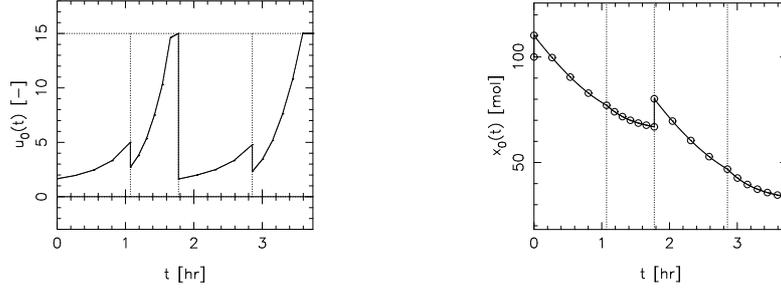


Abbildung 1.5: Optimale quasi-periodische Betriebsweise: Rücklaufverhältnis  $R$  (links) und Sumpf-Inhalt  $M$  (rechts)

Abfallfraktionen zum Mehrkomponenten-Ausgangsgemisch im Sumpf. So liegen folgende Sprünge bzgl. des Sumpf-Inhalts  $M$  vor:

$$\begin{aligned} M|_1 - M|_0 &= R_1 S_1 = R_1 (M|_2 - M|_3), \\ M|_4 - M|_3 &= R_2 S_2 = R_2 (M|_5 - M|_6). \end{aligned}$$

Ergebnis der Optimierung ist ein erhöhter Gewinn bei vollem Recycling ( $R_1 = R_2 = 1$ ) der Abfallfraktionen  $S_1$  und  $S_2$  für gegebene Energiekosten und Produktpreise. Die Steuerung des Rücklaufverhältnisses  $R$  sowie der Verlauf des Sumpf-Inhalts  $M$  sind für die berechnete optimale Betriebsweise in Abbildung 1.5 links bzw. rechts zu sehen. Zu beachten ist der Sprung des Sumpf-Inhalts  $M$  resultierend aus der Zuführung der Abfallfraktion  $S_2$ .

## 1.3 Direkter Ansatz

Um das Problem der Form (1.1) mit einem NLP-Löser zu behandeln, wird der unendlich-dimensionale Steuerungsraum mittels einer endlichen Zahl von Steuerungsparametern parameterisiert. Dabei kann sowohl eine globale als auch lokale Parameterisierung auf jedem Stufenhorizont vorgenommen werden. Wir betrachten hier nur eine *lokale Parameterisierung* mittels einer *stückweisen* Repräsentation der Steuerungsfunktionen, da sie flexibler ist und numerisch nicht zu einer zusätzlichen nichtlinearen Kopplung führt.

### 1.3.1 Stückweise Parameterisierung der Steuerung

Auf jeder Stufe  $i = 0, 1, \dots, M-1$  führen wir eine Zeittransformation durch

$$\theta_i(\tau, v) = t_i + \tau d_i, \quad t_i = t_0 + \sum_{k=0}^{i-1} d_k, \quad \tau \in [0, 1] \quad (1.4)$$

mit  $v := (t_0, d_0, d_1, \dots, d_{M-1})$  und einem festem, dimensionslosen Gitter

$$0 = \tau_{i,0} < \tau_{i,1} < \dots < \tau_{i,m_i} = 1, \quad (1.5)$$

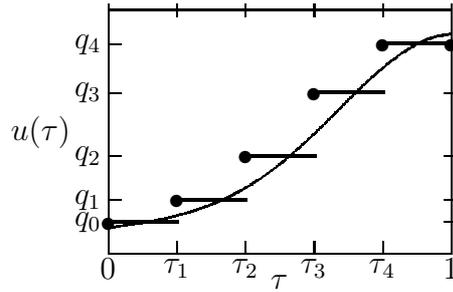


Abbildung 1.6: Stückweise konstante Repräsentation der Steuerungsfunktion

so daß  $\theta_i(\tau_{i,0}, v) = t_i$  und  $\theta_i(\tau_{i,m_i}, v) = t_{i+1}$ . Darauf wird eine *stückweise Repräsentation*  $\hat{u}_i$  der Steuerungsfunktion  $u_i$  durchgeführt:

$$\hat{u}_i(\tau) := \varphi_{ij}(\tau, q_{ij}), \quad \tau \in I_{ij} = [\tau_{ij}, \tau_{i,j+1}], \quad j = 0, 1, \dots, m_i - 1. \quad (1.6)$$

Dabei sind  $\varphi_{ij}$  *lokale Basisfunktionen* (z. B. konstante, lineare oder kubische Polynome). In der Abbildung 1.6 ist eine stückweise konstante Repräsentation angegeben. Stetigkeit der Steuerungsrepräsentation kann zusätzlich gefordert werden. Die Parameterisierung kann auf jeder Stufe individuell gewählt werden, ebenso wie die Anzahl und Ort der Stützpunkte.

In der Praxis erlaubt diese Repräsentation ein flexibles Umsetzen des ursprünglich unendlich-dimensionalen Problems. Ein anfänglich grobes Gitter (1.5) kann sukzessive verfeinert werden, um die Güte der Approximation zu verbessern. Diese Wahl der Steuerungsparameterisierung im Zusammenhang mit dem nachfolgenden Mehrfach-Schießverfahren ist entscheidend für die Eigenschaften der simultanen strukturausnutzenden Verfahren in dieser Arbeit und wurden von Bock und Plitt (1981, [73] sowie 1984, [20]) eingeführt.

## 1.4 Simultaner Ansatz

Nach der Parameterisierung liegt ein endlich-dimensionales NLP vor, bei dem die Zustandsvariablen z. B. mittels eines Integrators berechnet werden. Hierbei wird zwischen zwei Ansätzen unterschieden - dem *sequentiellen* und dem *simultanen* Ansatz. Der sequentielle Ansatz löst das dynamische System für gegebene Eingangsgrößen (Anfangswerte, Steuerungsparameter, globale Parameter) und berechnet neue Werte in einer NLP-Iteration. Dem Vorteil einer einfachen Implementierung stehen numerische Nachteile gegenüber, die später erläutert werden. Der simultane Ansatz dagegen diskretisiert das dynamische System innerhalb des Stufenhorizonts und führt somit mehr Informationen in das NLP-Problem ein. Im später beschriebenen Algorithmus kann gezeigt werden, daß diese Zusatzinformationen gegenüber dem sequentiellen Ansatz nicht zu einem höheren Aufwand führen. Der Vorteil liegt in der daraus resultierenden höheren numerischen Stabilität und Effizienz des Verfahrens.

### 1.4.1 Ein Mehrfach-Schießverfahren für DAEs

Neben der Kollokation ist das Mehrfach-Schießverfahren eine Möglichkeit, Zusatzinformationen in das Problem einzuführen. Im folgenden betrachten wir die Mehrfach-Schießvariante, wie sie von Bock *et al.* (1988, [18]) und verallgemeinert von Schulz *et al.* (1998, [84]) vorgeschlagen wurde. Dazu werden zusätzliche Optimierungsvariablen  $s_{ij}^x, s_{ij}^z$  in das NLP eingeführt und die folgenden *relaxierten* Anfangswertprobleme gelöst:

$$\left. \begin{aligned} B_i(\cdot) dx_i/d\tau &= f_i(x_i, z_i, \varphi_{ij}(\tau, q_{ij}), p, \theta_i(\tau, v)) d_i \\ 0 &= g_i(x_i, z_i, \varphi_{ij}(\tau, q_{ij}), p, \theta_i(\tau, v)) \\ &\quad - \alpha_{ij}(\tau) g_i(s_{ij}^x, s_{ij}^z, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) \end{aligned} \right\}, \quad \tau \in I_{ij}, \quad (1.7)$$

mit den Anfangswerten  $x_i(\tau_{ij}) = s_{ij}^x$  und  $z_i(\tau_{ij}) = s_{ij}^z$ . Um inkonsistente algebraische Anfangswerte benutzen zu können, wurde dazu die ursprüngliche DAE (1.1b) modifiziert (ähnlich einer Homotopie-Strategie). Der skalare Dämpfungsfaktor  $\alpha_{ij}$  ist dabei durch eine skalare Funktion gegeben, die nichtzunehmend und nichtnegativ auf  $I_{ij}$  ist und für die  $\alpha_{ij}(\tau_{ij}) = 1$  gilt. Implementiert ist die folgende Variante:

$$\alpha_{ij}(\tau) = \exp\left(-\bar{\alpha} \frac{\tau - \tau_{ij}}{\tau_{i,j+1} - \tau_{ij}}\right), \quad \bar{\alpha} \geq 0. \quad (1.8)$$

Um eine Lösung des Ausgangsproblems (1.1) zu bekommen, müssen auf jeder Stufe  $i = 0, 1, \dots, M-1$  die *Stetigkeitsbedingungen*

$$x_i(\tau_{i,j+1}; s_{ij}^x, s_{ij}^z, q_{ij}, p, v) - s_{i,j+1}^x = 0, \quad j = 0, 1, \dots, m_i - 1 \quad (1.9)$$

und die *Konsistenzbedingungen*

$$g_i(s_{ij}^x, s_{ij}^z, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) = 0, \quad j = 0, 1, \dots, m_i \quad (1.10)$$

in das NLP-Problem eingeführt werden. Ein Beispiel einer (unzulässigen) Anfangstrajektorie sowie einer (stetigen) Lösung ist in der Abbildung 1.7 angegeben. Die im Abschnitt 2.3 angegebenen NLP-Lösungsalgorithmen haben im allgemeinen unzulässige Iterierte, bei denen Simulation und Optimierung gleichzeitig durchgeführt werden. Um die Zustände  $x_i(\tau_{i,j+1}; s_{ij}^x, s_{ij}^z, q_{ij}, p, v)$  und deren Sensitivitäten zu berechnen, werden effiziente, adaptive DAE-Löser benutzt - siehe Kapitel 4.

### 1.4.2 Diskretisierung der Steuerungs- und Pfadbeschränkungen

Im folgenden wird der Ansatz der punktweisen Diskretisierung dieser Beschränkungen weiter verfolgt, da sich damit die Anwendungsprobleme in dieser Arbeit effizient und ausreichend genau behandeln lassen. Dazu werden an den Punkten  $\tau_{ij}$  des Gitters (1.5) folgende Bedingungen gestellt:

$$h_i(s_{ij}^x, s_{ij}^z, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) \geq 0, \quad j = 0, 1, \dots, m_i. \quad (1.11)$$

Damit können Schranken an die stückweise konstant oder linear parameterisierte Steuerung exakt eingehalten werden. Es sei betont, daß für die Diskretisierung der Pfadbeschränkungen auch ein feineres Diskretisierungsgitter verwendet werden kann. Weitere Möglichkeiten bestehen in der Formulierung von semi-infiniten Min-Max Problemen und deren Lösung über konsistente Approximationen (siehe Polak 1997, [74]).

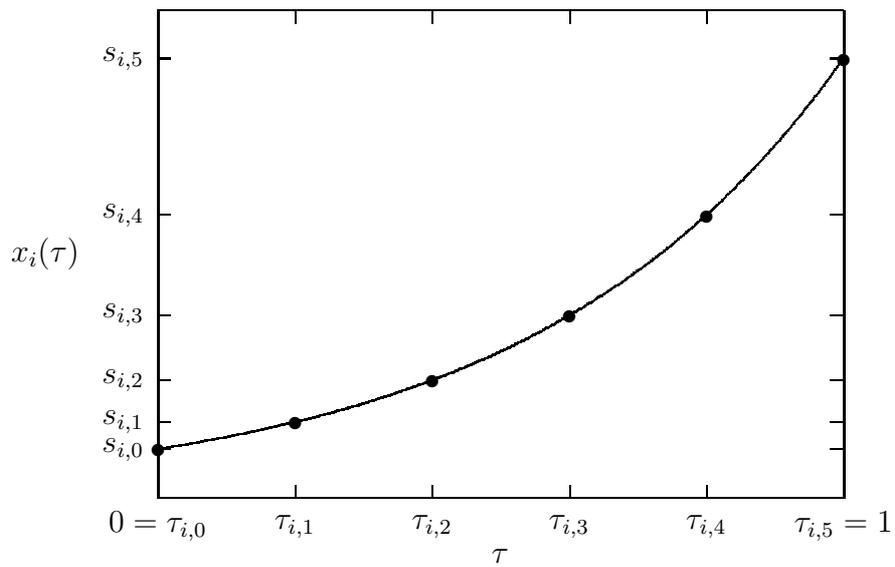
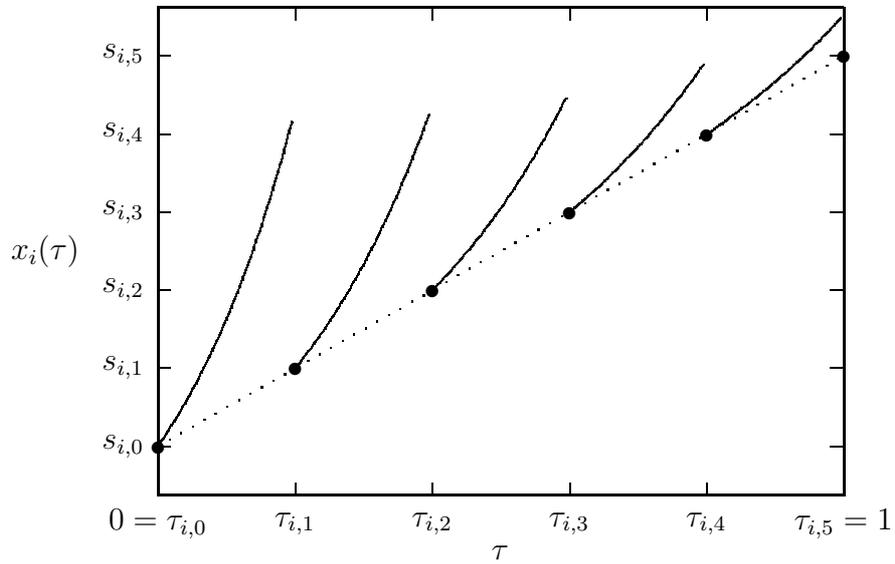


Abbildung 1.7: Direktes Mehrfach-Schießverfahren: Anfangstrajektorie (Startwerte interpoliert zwischen Randwerten) und Optimale Lösung

## 1.5 Struktur des diskretisierten und parameterisierten Problems

Die beschriebene Parametrisierung und Diskretisierung des  $M$ -stufigen Ausgangsproblems (1.1) führt zu folgendem strukturierten NLP, wobei  $v := (t_0, d_0, d_1, \dots, d_{M-1})$  ein weiterer Parametervektor ist, bestehend aus Anfangszeit und Stufendauern:

$$\min_{s_{ij}^x, s_{ij}^z, q_{ij}, p, v} \sum_{i=0}^{M-1} \phi_i(s_{i,m_i}^x, s_{i,m_i}^z, p, \theta_i(\tau_{i,m_i}, v)) \quad (1.12a)$$

unter den Stetigkeits- und Konsistenzbedingungen

$$\left. \begin{aligned} x_i(\tau_{i,j+1}; s_{ij}^x, s_{ij}^z, q_{ij}, p, v) - s_{i,j+1}^x &= 0, & j = 0, 1, \dots, m_i - 1 \\ g_i(s_{ij}^x, s_{ij}^z, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) &= 0, & j = 0, 1, \dots, m_i \end{aligned} \right\}, \quad i = 0, 1, \dots, M-1, \quad (1.12b)$$

den Anfangsbedingungen an  $s_{00}^x$

$$d(s_{00}^x, s_{00}^z, p) = 0, \quad (1.12c)$$

den diskretisierten Steuerungs- und Pfadbeschränkungen

$$h_i(s_{ij}^x, s_{ij}^z, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) \geq 0, \quad j = 0, 1, \dots, m_i, \quad i = 0, 1, \dots, M-1, \quad (1.12d)$$

den Stufenübergangsbedingungen

$$c_i(s_{i,m_i}^x, s_{i,m_i}^z, p) - s_{i+1,0}^x = 0, \quad i = 0, 1, \dots, M-2, \quad (1.12e)$$

und den linear gekoppelten Mehrpunkttrandbedingungen

$$\sum_{i=0}^{M-1} \left( r_i^s(s_{i,0}^x, s_{i,0}^z, p, \theta_i(\tau_{i,0}, v)) + r_i^e(s_{i,m_i}^x, s_{i,m_i}^z, p, \theta_i(\tau_{i,m_i}, v)) \right) \left\{ \begin{array}{l} = \\ \geq \end{array} \right\} 0. \quad (1.12f)$$

Zusätzlich können für die NLP-Variablen Schranken formuliert werden:

$$\left. \begin{aligned} \underline{s}_{ij}^x &\leq s_{ij}^x \leq \overline{s}_{ij}^x \\ \underline{s}_{ij}^z &\leq s_{ij}^z \leq \overline{s}_{ij}^z \\ \underline{q}_{ij} &\leq q_{ij} \leq \overline{q}_{ij} \end{aligned} \right\}, \quad j = 0, 1, \dots, m_i, \quad i = 0, 1, \dots, M-1, \quad (1.12g)$$

und

$$\left. \begin{aligned} \underline{p} &\leq p \leq \overline{p} \\ \underline{v} &\leq v \leq \overline{v}. \end{aligned} \right\} \quad (1.12h)$$

Die Zustandsvariablen  $x_i(\tau_{i,j+1}; s_{ij}^x, s_{ij}^z, q_{ij}, p, v)$  in (1.12b) stellen Lösungen der relaxierten Anfangswertprobleme (1.7) dar. Die festen, dimensionslosen Gitterpunkte  $\tau_{ij}$  sind in (1.5) und die Zeittransformationen  $\theta_i$  in (1.4) definiert. Die in diesem Kapitel dargestellte Parameterisierung und Diskretisierung des Optimierungsproblems und deren Lösung mit Hilfe der Methoden aus Kapitel 2, 3 und 4 hat sich für die Anwendungsprobleme aus dem Bereich der (bio-)chemischen Verfahrenstechnik als effizient und robust erwiesen.

# Kapitel 2

## Grundlegende Lösungsstrategien für Nichtlineare Programme

Dieses Kapitel beschreibt die Grundlagen für die Entwicklung von Verfahren, die zur Lösung von endlich-dimensionalen nichtlinearen Programmen (NLP) dienen. Im wesentlichen wird das Verfahren der Sequentiellen Quadratischen Programmierung (SQP) beschrieben. Ein Spezialfall eines NLP stellen wiederum solche Probleme dar, die eine Least-Squares-Zielfunktion haben. Für diese Klasse von Problemen werden in der Regel verallgemeinerte Gauß-Newton-Verfahren angewendet, welche hier als Spezialfall von SQP-Verfahren dargestellt werden.

### 2.1 Klassifikation der endlich-dimensionalen Probleme

Als *allgemeines Optimierungsproblem* wird das Problem bezeichnet, ein  $w^* \in S$  zu finden, das folgende Aufgabe löst:

$$F(w^*) = \min_{w \in S} F(w).$$

Dabei wird die Menge  $S$  auch als Menge aller *zulässigen Werte* bezeichnet und es sei  $F : \mathbb{R}^n \rightarrow \mathbb{R}$  eine Funktion.

Ein *gleichungs- und ungleichungsbeschränktes Optimierungsproblem* ist bestimmt durch Funktionen  $G_i : \mathbb{R}^n \rightarrow \mathbb{R}$  mit  $i \in \mathcal{E} = \{1, \dots, m_G\}$ ,  $H_j : \mathbb{R}^n \rightarrow \mathbb{R}$  mit  $j \in \mathcal{I} = \{1, \dots, m_H\}$  und die zulässige Menge  $S = \{x \in \mathbb{R}^n \mid G_i(x) = 0, i \in \mathcal{E}, H_j(x) \geq 0, j \in \mathcal{I}\}$  sowie einer Zielfunktion  $F : \mathbb{R}^n \rightarrow \mathbb{R}$ . In Kurzform schreiben wir dafür:

$$\begin{array}{ll} \text{[NLP]} & \min F(w) \end{array} \tag{2.1a}$$

$$\text{unter } G(w) = 0 \tag{2.1b}$$

$$H(w) \geq 0. \tag{2.1c}$$

Weiterhin von Interesse ist auch die Menge der *aktiven Ungleichungsbedingungen* im Punkt  $w : \mathcal{I}_0(w) = \{j \in \mathcal{I} \mid H_j(w) = 0\}$ .

Als *gleichungsbeschränktes Optimierungsproblem* [EQ] wird ein NLP bezeichnet, bei dem

$S$  nur durch Gleichungsbeschränkungen beschrieben wird:  $S = \{w \in \mathbb{R}^n \mid G_i(w) = 0, i \in \mathcal{E}\}$ .

Einen Spezialfall eines NLP stellen *Least-Squares-Probleme* dar, die folgenden Zielfunktionsstyp haben:  $F(w) = \|f(w)\|_2^2 = f(w)^T f(w)$  mit  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ , wobei häufig  $m \gg n$  ist. Dieses Problem wird mit [LSQ] bezeichnet.

Auf die Differenzierbarkeitseigenschaften der in den Problemen vorkommenden Funktionen  $F, G_i, i \in \mathcal{E}$  und  $H_j, j \in \mathcal{I}$  wird an geeigneter Stelle eingegangen.

## 2.2 Optimalitätsbedingungen

Eine lokale Lösung  $w^*$  eines NLP muß zunächst einmal zulässig sein und weiterhin in einer Umgebung der zulässigen Menge keinen niedrigeren Wert der Zielfunktion als  $F(w^*)$  haben. Dazu wird ein zulässiger Pfad definiert, dessen Ende  $w^*$  sei. Um zulässige Richtungen entlang dieses Pfades zu definieren, gibt es Bedingungen, welche die lokale Struktur der zulässigen Menge  $S$  beschreiben. Diese werden auch als *Regularitätsbedingungen* bezeichnet, weil sie die Existenz eines zulässigen Pfades sichern. Zwei davon dienen uns dabei zur Definition von Optimalitätsbedingungen: LICQ, MFCQ.

### Definition 1 (LICQ: linear independence constraint qualification)

Es gilt die Bedingung LICQ in  $w$ , wenn  $w \in S$  und die Vektoren  $\nabla G_i(w)$ ,  $i \in \mathcal{E}$  und  $\nabla H_j(w)$ ,  $j \in \mathcal{I}_0(w)$  linear unabhängig sind.

### Definition 2 (MFCQ: Mangasarian-Fromovitz constraint qualification)

Es gilt die Bedingung MFCQ in  $w$ , wenn  $w \in S$  und

- die Vektoren  $\nabla G_i(w)$ ,  $i \in \mathcal{E}$  linear unabhängig sind,
- ein Vektor  $\xi \in \mathbb{R}^n$  existiert mit

$$\begin{aligned} \nabla G_i(w)^T \xi &= 0, \quad i \in \mathcal{E}, \\ \nabla H_j(w)^T \xi &> 0, \quad j \in \mathcal{I}_0(w). \end{aligned}$$

### Bemerkung 1

Wenn  $w$  LICQ erfüllt, dann erfüllt  $w$  auch MFCQ.

In Abbildung 2.1 wird der Unterschied zwischen den beiden Regularitätsbedingungen anhand von Ungleichungsbedingungen dargestellt. Im linken Bild ist die Bedingung LICQ erfüllt, da die Gradienten der beiden im Punkt  $w$  aktiven Ungleichungen linear unabhängig sind. Im rechten Bild dagegen ist die gestrichelte Ungleichungsbedingung hinzugefügt worden. In diesem Fall gilt nur noch die Bedingung MFCQ, da in  $w$  der Gradient der gestrichelten Ungleichungsbedingung von den Gradienten der anderen beiden linear abhängig ist.

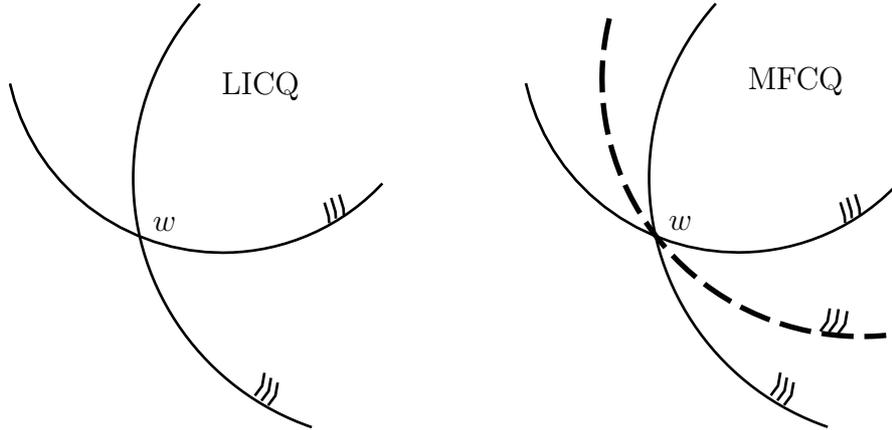


Abbildung 2.1: Unterschied zwischen LICQ und MFCQ

## 2.2.1 Bedingungen erster Ordnung

Für einen Punkt  $w^*$ , der MFCQ erfüllt, können wir nun *notwendige Bedingungen erster Ordnung* für ein lokales Minimum des NLP aufstellen, welche unter anderem geometrisch aussagen, daß  $\nabla F(w^*)$  durch die Gradienten  $\nabla G_i(w^*), i \in \mathcal{E}$  und  $\nabla H_j(w^*), j \in \mathcal{I}_0(w^*)$  aufgespannt wird. Dazu wird die Lagrange-Funktion  $L$  für das NLP eingeführt:

$$\begin{aligned} L(w, \lambda, \mu) &= F(w) - \lambda^T G(w) - \mu^T H(w) \\ &= F(w) - \sum_{i \in \mathcal{E}} \lambda_i G_i(w) - \sum_{j \in \mathcal{I}} \mu_j H_j(w). \end{aligned}$$

Dabei bezeichnen  $\lambda \in \mathbb{R}^{|\mathcal{E}|}$  und  $\mu \in \mathbb{R}^{|\mathcal{I}|}$  die *Lagrange-Multiplikatoren* für die Gleichungen und Ungleichungen.

### Satz 1 (notwendige Bedingungen erster Ordnung)

Seien  $F, G$  und  $H$  stetig differenzierbar. Es gelte MFCQ für  $w^*$ . Liegt in  $w^*$  ein lokales Minimum des NLP vor, so existieren Multiplikatoren  $\lambda^* \in \mathbb{R}^{|\mathcal{E}|}$  und  $\mu^* \in \mathbb{R}^{|\mathcal{I}|}$ , so daß die Bedingungen gelten:

$$\nabla_w L(w^*, \lambda^*, \mu^*) = 0 \quad (2.2a)$$

$$G(w^*) = 0 \quad (2.2b)$$

$$H(w^*) \geq 0 \quad (2.2c)$$

$$\mu^* \geq 0 \quad (2.2d)$$

$$\mu_j^* H_j(w^*) = 0, \quad j \in \mathcal{I}. \quad (2.2e)$$

Dabei wird (2.2e) als *Komplementaritätsbedingung* bezeichnet.

### Bemerkung 2 (Beschränktheit der Multiplikatoren)

Gauvin (1977, [41]) hat bewiesen, daß die Menge der Multiplikatoren  $(\lambda^*, \mu^*)$  an einem stationären Punkt  $w^*$  genau dann ein kompaktes konvexes Polyeder bilden, wenn die Regularitätsbedingung MFCQ gilt. Dabei ist ein in stationärer Punkt  $w^*$  dadurch definiert, daß die Bedingungen (2.2) gelten.

### Bemerkung 3 (KKT-Bedingungen)

Unter der stärkeren Voraussetzung LICQ an  $w^*$  sind die Multiplikatoren eindeutig. In dieser Form ist der Satz 1 von Karush (1939, [51]) und Kuhn, Tucker (1951, [53]) aufgestellt worden und man spricht beim Tripel  $(w^*, \lambda^*, \mu^*)$  auch von einem Karush-Kuhn-Tucker (KKT) Punkt. Die Jacobimatrix der Gleichungsbedingungen (2.2a), (2.2b) und aktiven Ungleichungsbedingungen (2.2c) in Satz 1 wird auch KKT-Matrix genannt.

Man bezeichnet die Multiplikatoren auch als sogenannte *Schattenpreise*, was damit zu tun hat, daß bei einer Variation der Nebenbedingung die Multiplikatoren angeben, um wieviel sich die Zielfunktion ändert. Diese Interpretation ist nützlich, um zum Beispiel eine *Active-Set-Methode* zur Lösung eines ungleichungsbeschränkten Quadratischen Programms zu konstruieren. Dieses quadratische Programm ist definiert durch eine quadratische Zielfunktion und lineare Nebenbedingungen. Bei einer primalen Active-Set-Methode wird zuerst ein zulässiger Punkt bezüglich der linearen Nebenbedingungen gesucht und anschließend ein sogenanntes Active-Set aus den Ungleichungsbedingungen gebildet, welche temporär als Gleichungsbedingungen angesehen werden. Beim Test, ob eine Ungleichung aus dem Active-Set genommen werden soll, wird eine der Ungleichungen genommen, welche einen negativen Multiplikator besitzt (siehe Fletcher 1987, [37]).

## 2.2.2 Bedingungen zweiter Ordnung

Als nächstes werden hinreichende Bedingungen 2. Ordnung angeben, da die notwendigen Bedingungen erster Ordnung bisher noch keine Unterscheidung zwischen lokalem Minimum, Maximum und Sattelpunkt gemacht haben:

### Satz 2 (hinreichende Bedingungen zweiter Ordnung)

Seien  $F$ ,  $G$  und  $H$  zweimal stetig differenzierbar. Eine hinreichende Bedingung für ein striktes lokales Minimum  $w^*$  des NLP ist erfüllt, wenn Multiplikatoren  $\lambda^* \in \mathbb{R}^{|\mathcal{E}|}$  und  $\mu^* \in \mathbb{R}^{|\mathcal{I}|}$  existieren, so daß  $(w^*, \lambda^*, \mu^*)$  die notwendigen Bedingungen erster Ordnung erfüllen und für jedes  $\xi \in \mathbb{R}^n, \xi \neq 0$  mit:

$$\xi^T \nabla G_i(w^*) = 0, \quad i \in \mathcal{E}, \quad (2.3a)$$

$$\xi^T \nabla H_j(w^*) = 0, \quad j \in \mathcal{I}_0(w^*) \text{ mit } \mu_j^* > 0, \quad (2.3b)$$

$$\xi^T \nabla H_j(w^*) \geq 0, \quad j \in \mathcal{I}_0(w^*) \text{ mit } \mu_j^* = 0 \quad (2.3c)$$

gilt:  $\xi^T \nabla_w^2 L(w^*, \lambda^*, \mu^*) \xi > 0$ .

### Bemerkung 4 (Positiv-Definitheit der reduzierten Lagrange-Hessematrix)

Insbesondere sei betont, daß hier nicht gefordert wird, daß die Hessematrix der Lagrange-funktion auf dem gesamten Raum  $\mathbb{R}^n$  positiv definit ist, sondern nur auf einem Teil des Kerns der aktiven nichtdegenerierten Nebenbedingungen.

### Bemerkung 5 (Stabilität gegen Störungen)

Die hinreichenden Bedingungen zweiter Ordnung sind hier in der schwachen Formulierung angegeben. Wenn man von dem Punkt  $(w^*, \lambda^*, \mu^*)$  zusätzlich fordert, daß er die Regularitätsbedingung LICQ erfüllt und die strikte Komplementarität ( $j \in \mathcal{I}_0(w^*) \Rightarrow \mu_j^* > 0$ ) gilt, dann ist  $(w^*, \lambda^*, \mu^*)$  stabil gegen Störungen und die Jacobimatrix der Gleichungsbedingungen aus Satz 1 am Optimum nichtsingulär.

## 2.3 Iterative Methoden basierend auf linear-quadratischen Approximationen

Die Basis der hier vorgestellten iterativen Methoden für die Lösung eines Nichtlinearen Programms (NLP) stellen die Optimalitätsbedingungen erster Ordnung dar. Sie charakterisieren die Lösung in Form von nichtlinearen Gleichungs- und Ungleichungsbedingungen. In einem Optimum des NLP spielen jedoch nur die aktiven Ungleichungsbeschränkungen eine wesentliche Rolle - in einer Umgebung des Optimums kann das Problem als rein gleichungsbeschränktes Problem formuliert werden. Wir beschränken uns daher auf die Behandlung von gleichungsbeschränkten Problemen (EQ).

Die Optimalitätsbedingungen erster Ordnung für EQ liefern ein nichtlineares Gleichungssystem, dessen Lösung  $(w^*, \lambda^*)$  Kandidaten für eine Lösung von EQ sind:

$$0 = \nabla_w L(w^*, \lambda^*) = \nabla F(w^*) - \sum_{i \in \mathcal{E}} \lambda_i^* \nabla G_i(w^*), \quad (2.4a)$$

$$0 = \nabla_\lambda L(w^*, \lambda^*) = G(w^*). \quad (2.4b)$$

Nun läßt sich mit Hilfe von Newton-ähnlichen Verfahren eine Iterationsvorschrift definieren, die ausgehend von einem Startwert  $(w_0, \lambda_0)$  eine Folge von Iterierten generiert, welche lokal gegen eine Lösung des nichtlinearen Gleichungssystem (2.4) konvergiert. Dazu muß die Newton-Iterationsmatrix genauer definiert werden, von der wiederum die Konvergenzrate abhängig ist. Für den Fall, daß die Iterationsmatrix exakt ist, d. h. sie der Hessematrix der Lagrange-Funktion entspricht, sieht die Definition der Newton-Iterierten  $(\Delta w_k, \Delta \lambda_k)$  wie folgt aus:

$$\nabla^2 L(w_k, \lambda_k) \begin{pmatrix} \Delta w_k \\ \Delta \lambda_k \end{pmatrix} = -\nabla L(w_k, \lambda_k). \quad (2.5)$$

Wir erhalten also Folgen  $w_{k+1} = w_k + \Delta w_k$  und  $\lambda_{k+1} = \lambda_k + \Delta \lambda_k$ . Diesen Ansatz nennen wir den *Lagrange-Newton-Ansatz*.

Die Multiplikatoren  $\lambda_{k+1}$  lassen sich auch direkt berechnen, wie man leicht an der ausformulierten Iterationsgleichung (2.5) sehen kann:

$$\begin{pmatrix} \nabla_w^2 L(w_k, \lambda_k) & \nabla G(w_k) \\ \nabla G(w_k)^T & \end{pmatrix} \begin{pmatrix} \Delta w_k \\ \Delta \lambda_k \end{pmatrix} = - \begin{pmatrix} \nabla F(w_k) + \nabla G(w_k) \lambda_k \\ G(w_k) \end{pmatrix}. \quad (2.6)$$

Dazu ist die rechte Seite von (2.6) zu modifizieren:

$$\begin{pmatrix} \nabla_w^2 L(w_k, \lambda_k) & \nabla G(w_k) \\ \nabla G(w_k)^T & \end{pmatrix} \begin{pmatrix} \Delta w_k \\ \lambda_{k+1} \end{pmatrix} = - \begin{pmatrix} \nabla F(w_k) \\ G(w_k) \end{pmatrix}. \quad (2.7)$$

Man sieht, daß die Multiplikatoren nur noch in der Hessematrix der Lagrange-Funktion vorkommen. Dies kann bei lokalen Konvergenzuntersuchungen ausgenutzt werden.

Eine wesentliche Feststellung ist, daß unter bestimmten Bedingungen an die Lagrange-Hessematrix die Lösung  $(\Delta w_k, \lambda_{k+1})$  der Lagrange-Newton-Gleichungen (2.7) auch eine

Lösung des folgenden *Quadratischen Programms* ist:

$$[\text{QP}] \quad \min \quad \nabla F(w_k)^T \Delta w_k + \frac{1}{2} \Delta w_k^T \nabla_w^2 L(w_k, \lambda_k) \Delta w_k \quad (2.8a)$$

$$\text{unter} \quad G(w_k) + \nabla G(w_k)^T \Delta w_k = 0. \quad (2.8b)$$

Ein sequentielles Verfahren, das auf der wiederholten Lösung von Quadratischen Programmen beruht, nennt man auch *SQP-Verfahren*, welches zuerst von Wilson (1963, [99]) aufgestellt und untersucht wurde.

Dabei stimmen die Lösungen von (2.7) und (2.8) überein, wenn die Lagrange-Hessematrix *positiv definit* auf dem Kern der strikt aktiven Restriktionen sind und die Lösung von (2.8) eindeutig ist. Die Bedingung an die Hessematrix ist dabei in einer Umgebung eines KKT-Punktes erfüllt, an dem die hinreichenden Bedingungen zweiter Ordnung gelten (siehe auch Bemerkung 5). In den anderen Fällen existiert mitunter keine Lösung des QP, da Richtungen negativer Krümmungen vorhanden sein können. Dies wiederum gibt auch den Hinweis auf Modifikationen des Problems: um eine endliche Lösung zu erhalten, müssen die entsprechenden Eigenwerte der Newton-Iterationsmatrix geschiftet oder der Schritt beschränkt werden. Ansätze dafür sind in der Literatur auch als Levenberg-Marquardt- oder Trust-Region-Modifikationen bekannt.

In vielen Fällen ist es jedoch zu teuer, die Lagrange-Hessematrix  $\nabla_w^2 L(w_k, \lambda_k)$  auszurechnen und daher wird versucht, sie durch Updates zu approximieren. Die Grundlage dafür ist die *Sekantenbedingung*. Um sie herzuleiten, wird das Problem einer unbeschränkten Minimierung einer nichtlinearen zweimal stetig differenzierbaren Funktion  $F : \mathbb{R}^n \rightarrow \mathbb{R}$  betrachtet. Durch Taylorentwicklung von  $\nabla F(w)$  um  $w_k$  erhält man folgende Bedingung:

$$\nabla^2 F(w_k)(w_k - w_{k-1}) = \nabla F(w_k) - \nabla F(w_{k-1}) + O(\|w_k - w_{k-1}\|^2).$$

Daraus läßt sich die besagte Sekantenbedingung für die Approximation  $B_k$  von  $\nabla^2 F(w_k)$  definieren:

$$B_k \delta_{k-1} = \gamma_{k-1}, \quad (2.9)$$

mit  $\delta_{k-1} = w_k - w_{k-1}$  und  $\gamma_{k-1} = \nabla F(w_k) - \nabla F(w_{k-1})$ . Da dies jedoch nur  $n$  Bedingungen für die Einträge der Hessematrix sind, sind weitere Bedingungen notwendig:

**Symmetrie-Erhaltung:**  $B_k$  symmetrisch  $\rightarrow B_{k+1}$  symmetrisch,

**Erhaltung der Positiv Definitheit:**  $B_k$  positiv definit  $\rightarrow B_{k+1}$  positiv definit,

**Least Change Bedingung:** Veränderung möglichst weniger Krümmungsinformationen:

$$B_{k+1} z = B_k z \quad \forall z^T (w_{k+1} - w_k) = 0.$$

Daraus lassen sich Update-Vorschriften für Rang-1- und Rang-2-Updates mit vollständigem beziehungsweise teilweise Merkmalen alter Krümmungsinformationen gewinnen:

$$B_{k+1} = B_k + U(B_k, \delta_k, \gamma_k)$$

beziehungsweise

$$B_{k+1} = U(B_0, \delta_l, \dots, \delta_k, \gamma_l, \dots, \gamma_k) \quad \text{für } l \leq k.$$

Das bekannteste Update ist das Rang-2-Update nach *Broyden, Fletcher, Goldfarb und Shanno*, welches die Positiv-Definitheit und die Symmetrie berücksichtigt:

$$U_{BFGS}(B_k, \delta_k, \gamma_k) = \frac{\gamma_k \gamma_k^T}{\gamma_k^T \delta_k} - \frac{B_k \delta_k \delta_k^T B_k}{\delta_k^T B_k \delta_k}. \quad (2.10)$$

Es existiert auch eine Version mit teilweiseem Merken alter Krümmungsinformationen, das sogenannte *Limited-Memory-Update* von Nocedal (1980, [67]):

$$U_{L-BFGS}(B_0, \delta_1, \dots, \delta_k, \gamma_1, \dots, \gamma_k) = B_0 + \sum_{i=1}^k \left( \frac{\gamma_i \gamma_i^T}{\gamma_i^T \delta_i} - \frac{B_i \delta_i (B_i \delta_i)^T}{\delta_i^T B_i \delta_i} \right).$$

Eingebettet in SQP-Verfahren wurden für beide Update-Vorschriften zahlreiche Untersuchungen durchgeführt. Dabei gibt es sowohl Resultate hinsichtlich der lokalen Konvergenz der Iterierten  $w_k$  als auch im Hinblick der Hessematrixapproximationen  $B_k$ . In Boggs und Tolle (1994, [21]) ist die Konvergenz der Hessematrixapproximationen bei Rang-2-Updates näher untersucht worden. Ihre Analyse wurde unabhängig von einem zugrunde liegenden Optimierungsverfahren durchgeführt und sie zeigten unter bestimmten Annahmen an  $\delta_k$  und  $\gamma_k$ , daß die Konvergenzrate der Hessematrixapproximationen im wesentlichen r-linear ist und damit in einem Algorithmus, bei dem die Iterierten  $w_k$  schneller konvergieren, keine Konvergenz der Hessematrixapproximationen vorliegt. Wie wir später sehen werden, ist die Konvergenz der Hessematrixapproximation gegen die exakte Hessematrix nicht notwendig.

Die Verallgemeinerung der BFGS-Update-Vorschriften in SQP-Verfahren für NLP ist dabei im wesentlichen durch Powell (1978, [76]) durchgeführt worden: dazu sei nun  $\gamma_{k-1} = \nabla_w L(w_k, \lambda_k) - \nabla_w L(w_{k-1}, \lambda_k)$ . Da unter Umständen die Positiv Definitheit von  $B_k$  (auch durch Verkürzung des Schritts) nicht gewährleistet werden kann, wird folgende Heuristik eingeführt. In der Update-Formel wird die Differenz der Gradienten  $\gamma_k$  durch den Vektor  $\eta_k$  ersetzt:

$$\eta_k := \Theta_k \gamma_k + (1 - \Theta_k) B_k \delta_k, \quad 0 < \Theta_k \leq 1.$$

Damit ist dann folgende Bedingung einzuhalten:

$$\delta_k^T \eta_k \geq \epsilon_\Theta \delta_k^T B_k \delta_k > 0 \quad \text{mit } \epsilon_\Theta \in [0.1, 0.2].$$

Für  $\Theta_k$  ergibt sich daraus folgende Bedingung:

$$\Theta_k = \begin{cases} 1 & \text{wenn } \delta_k^T \gamma_k \geq \epsilon_\Theta \delta_k^T B_k \delta_k \\ \frac{(1 - \epsilon_\Theta) \delta_k^T B_k \delta_k}{\delta_k^T B_k \delta_k - \delta_k^T \gamma_k} & \text{sonst.} \end{cases}$$

Zu beachten ist, daß bei Werten  $\Theta_k < 1$  die Sekantenbedingung verletzt wird und somit eventuell keine sinnvollen Krümmungsinformationen aufdatiert werden. Aus Gründen, die die lokale Konvergenz betreffen, sollte ein häufiger Einsatz hintereinander vermieden werden. In der Praxis haben sich als Alternative auch ein Aussetzen des Updates als sinnvoll erwiesen, beziehungsweise der Einsatz eines Limited-Memory-BFGS-Updates, bei dem dann eventuell größere Werte für  $\Theta_k$  beobachtet werden können.

Bisher wurden Verfahren beschrieben, die die Natur der Zielfunktion  $F(w)$  nicht ausnutzen. Bei Least-Squares-Funktionen jedoch sind sowohl statistische als auch numerische Interpretationen von Interesse. Dazu habe die Zielfunktion folgende Gestalt:

$$F(w) = \|f(w)\|_2^2 = \sum_{j=1}^m \frac{(\eta_j - h(t_j, w))^2}{\sigma_j^2}.$$

Hierbei seien  $\eta_j$  Messungen,  $h(t_j, w)$  die Modellantworten an den Zeitpunkten  $t_j$  mit Parametern  $w$  und  $\sigma_j^2$  die Varianzen der Messungen.  $m$  ist dabei die Anzahl der Daten. Wenn nun die Fehler der Messungen

$$\epsilon_j = \eta_j - h(t_j, w), \quad j = 1, \dots, m$$

normalverteilt, mit bekannter Varianz  $\sigma_j$ , Erwartungswert 0 sowie unabhängig sind, dann ist eine Lösung des Least-Squares-Problems (LSQ) eine Maximum-Likelihood-Schätzung, welche die Stichprobe mit größter Wahrscheinlichkeit reproduziert.

Die Ausformulierung der Hessematrix der Zielfunktion ergibt für diesen Fall:

$$\nabla^2 F(w) = 2\nabla f(w)\nabla f(w)^T + 2\sum_{j=1}^m f_j(w)\nabla^2 f_j(w).$$

Für Probleme, bei denen  $f_j(w_k), j = 1, \dots, m$  klein ist, ist  $B_k = 2\nabla f(w_k)\nabla f(w_k)^T$  eine gute Approximation der exakten Hessematrix. Diese Eigenschaft nutzen wir im sogenannten *verallgemeinerten Gauß-Newton-Verfahren* aus, bei dem im Quadratischen Programm (QP) anstatt  $\nabla_w^2 L(w_k, \lambda_k)$  nun  $B_k$  verwendet wird. Anders formuliert wird also zur Berechnung eines Schritts  $\Delta w_k$  folgendes Problem gelöst:

$$[\text{LSQ-QP}] \quad \min \quad \|f(w_k) + \nabla f(w_k)^T \Delta w_k\|_2^2 \quad (2.11a)$$

$$\text{unter} \quad G(w_k) + \nabla G(w_k)^T \Delta w_k = 0. \quad (2.11b)$$

### Satz 3 (Fixpunkteigenschaft des Gauß-Newton-Verfahrens)

Sei  $(w^*, \lambda^*)$  ein KKT-Punkt des Problems (LSQ), der die hinreichenden Bedingungen zweiter Ordnung erfüllt. Dann gilt:  $(w^*, \lambda^*)$  ist ein KKT-Punkt von (LSQ)  $\Leftrightarrow$   $(0, \lambda^*)$  ist ein KKT-Punkt von (LSQ-QP). Das heißt, daß ein KKT-Punkt von LSQ auch ein Fixpunkt des verallgemeinerten Gauß-Newton-Verfahrens ist.

### Bemerkung 6 (Genauigkeit der Lösung von LSQ-QP)

Im unbeschränkten Fall führt die Berechnung einer Lösung über die Normalgleichung zu einer Verschlechterung der Genauigkeit der Lösung (hier geht das Quadrat der Kondition der LSQ-Jacobimatrix ein). Im beschränkten Fall ist dies ebenso, wenn das LSQ-QP als allgemeines QP gelöst wird, welches nicht die spezielle Natur der Hessematrix berücksichtigt. Eine bessere Lösungsmethode ist, die (reduzierte) LSQ-Jacobimatrix orthogonal zu zerlegen und aufzulösen. Dabei geht die Kondition der (reduzierten) LSQ-Jacobimatrix nur linear in die Genauigkeit der Lösung ein.

Der nachfolgende Satz von Bock (1987, [16]) definiert eine verallgemeinerte Inverse, den Lösungsoperator des verallgemeinerten Gauß-Newton-Verfahrens. Damit ist es möglich, einen *Lokalen Kontraktionssatz* in einer einheitlichen Weise für die hier behandelten iterativen Verfahren zu verwenden.

**Satz 4 (Existenz einer verallgemeinerten Inverse)**

Seien LICQ und Positiv Definitheit der Gauß-Newton-Hessematrix auf dem Kern der Jacobimatrix der Gleichungsbedingungen erfüllt (auf einem Gebiet  $D \subset \mathbb{R}^n$ ) und  $J = (\nabla f^T, \nabla G^T)^T : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^{(m+m_G) \times n}$ , dann gilt:

1. Für jedes  $r = (f^T, G^T)^T \in \mathbb{R}^{m+m_G}$  existiert genau ein KKT-Punkt  $(\Delta w, \lambda)$  von LSQ-QP und  $\Delta w$  ist ein striktes Minimum.
2. Es existiert eine lineare Abbildung  $J^+ : \mathbb{R}^{m+m_G} \rightarrow \mathbb{R}^n$ , so daß  $\Delta w = J^+ r$  das LSQ-QP löst für jedes  $r \in \mathbb{R}^{m+m_G}$  mit

$$J^+ = (I \ 0) \begin{pmatrix} \nabla f \nabla f^T & \nabla G \\ \nabla G^T & \end{pmatrix}^{-1} \begin{pmatrix} \nabla f \\ I \end{pmatrix}.$$

3. Der Lösungsoperator  $J^+$  ist eine verallgemeinerte Inverse und erfüllt folgende Bedingung:  $J^+ = J^+ J J^+$ .

**2.3.1 Lokale Konvergenz**

Zentraler Punkt in diesem Abschnitt ist der lokale Kontraktionssatz von Bock (1987, [16]) für Newton-ähnliche Verfahren zur Lösung von nichtlinearen Gleichungen. Dabei wird die lokale Konvergenz von SQP-Verfahren mit und ohne exakte Hessematrizen und von Gauß-Newton-Verfahren erläutert. Dazu wird die Berechnung der Schritte auf die Lösung von Gleichungssystemen zurückgeführt (siehe Bemerkung 3 und Satz 4).

Sei dazu  $\Delta x_{k+1} = -M(x_k)r(x_k)$  eine Verfahrensvorschrift, welche die Iterierten zur Lösung eines nichtlinearen Gleichungssystems  $r(x) = 0$  mit einer Näherungsinversen  $M(x) \approx \nabla r(x)^{-T}$ ,  $M(x) \neq 0$  definiert.

**Satz 5 (Lokaler Kontraktionssatz)**

Sei  $D \subset \mathbb{R}^n$  ein Gebiet und  $r : D \rightarrow \mathbb{R}^m$  stetig differenzierbar,  $J(x) = \nabla r(x)^T$ . Für alle  $x, y \in D$  und alle  $t \in [0, 1]$  mit  $y - x := -M(x)r(x)$  gelte:

$$\exists \omega < \infty : \quad \|M(y)[J(x + t(y - x)) - J(x)](y - x)\| \leq \omega t \|y - x\|^2, \quad (2.12)$$

$$\exists \kappa(x) \leq \kappa < 1 : \quad \|M(y)R(x)\| \leq \kappa(x) \|y - x\| \leq \kappa \|y - x\| \quad (2.13)$$

mit  $R(x) = r(x) - J(x)M(x)r(x)$  (Residuum).

Sei  $x_0 \in D$  gegeben mit:

$$\delta_0 := \frac{\alpha_0 \omega}{2} + \kappa < 1, \quad \alpha_j := \|M(x_j)r(x_j)\|, \quad j = 0, 1, \dots, \quad (2.14)$$

$$D_0 := \left\{ x \mid \|x - x_0\| \leq \frac{\alpha_0}{1 - \delta_0} \right\} \subset D. \quad (2.15)$$

Dann gilt:

1. Die Iteration  $x_{j+1} = x_j + \Delta x_j$ ,  $\Delta x_j = -M(x_j)r(x_j)$  bleibt in  $D_0$  und ist wohl definiert.
2.  $x_j \rightarrow x^* \in D_0$  für  $j \rightarrow \infty$  mit  $M(x^*)r(x^*) = 0$ .
3. Konvergenzrate:  $\|\Delta x_{j+1}\| \leq (\frac{\alpha_j \omega}{2} + \kappa)\|\Delta x_j\| =: \delta_j \|\Delta x_j\|$ .
4. A priori Abschätzung:  $\|x_{j+1} - x^*\| \leq \delta_0^j \frac{\alpha_0}{1 - \delta_0}$ .

**Bemerkung 7 (Lipschitzkonstanten im lokalen Kontraktionsatz)**

1. Gauß-Newton-Verfahren:  $\omega$  ist ein Maß für die Nichtlinearität und wird groß, wenn die erste Ableitung von  $J$  groß wird.  $\kappa$  ist ein Maß für die Kompatibilität von den Meßdaten mit dem Modell (Beobachtung).  $\kappa < 1$  kann gezeigt werden, wenn Residuum  $R$  und erste Ableitung der verallgemeinerten Inverse  $M$  klein sind:

$$\|M(y)R(x)\| = \|(M(y) - M(x))R(x)\| \leq \kappa \|y - x\|.$$

2. SQP-Verfahren:  $\omega$  ist ebenfalls ein Maß für die Nichtlinearität,  $\kappa$  ist hier jedoch ein Maß für die Güte der Hessematrix-Approximation  $M^{-1}$  (siehe Bemerkung 9):

$$\|M(y)R(x)\| = \|M(y)[M(x)^{-1} - J(x)]M(x)r(x)\| \leq \kappa \|y - x\|.$$

**Bemerkung 8 (Eigenschaft der Näherungsinversen  $M$ )**

1. Gauß-Newton-Verfahren für unbeschränktes LSQ-Problem:  
 $M$  ist Moore-Penrose-Pseudo-Inverse.
2. Gauß-Newton-Verfahren für beschränktes LSQ-Problem:  
 $M$  ist die verallgemeinerte Inverse (siehe Satz 4).
3. Newton-ähnliches Verfahren für Gleichungssysteme:  
 $M$  ist eine Näherung der exakten Inverse.
4. SQP-Verfahren mit exakten Hessematrizen für gleichungsbeschränktes Minimierungsproblem:  $M$  ist die Inverse der KKT-Matrix.
5. SQP-Verfahren mit BFGS-Update-Hessematrizen für gleichungsbeschränktes Minimierungsproblem:  $M$  ist die Inverse der KKT-Matrix mit nichtexakter Lagrange-Hessematrix.

**Bemerkung 9 (Konvergenzraten bzgl. Quotientenkriterium)**

1. Gauß-Newton-Verfahren für unbeschränktes LSQ-Problem:  
Konvergenzrate linear in  $w$ .
2. Gauß-Newton-Verfahren für beschränktes LSQ-Problem:  
Konvergenzrate linear in  $w$ .
3. Newton-Verfahren für Gleichungssystem:  
Konvergenzrate quadratisch in  $w$ , da  $R(w) = 0$ .

4. SQP-Verfahren mit exakten Hessematrizen für gleichungsbeschränktes Minimierungsproblem: Konvergenzrate quadratisch in  $(w, \lambda)$ , da  $R(w, \lambda) = 0$ .
5. SQP-Verfahren mit Update-Hessematrizen für gleichungsbeschränktes Minimierungsproblem: im folgenden werden die Bedingungen für superlineare Konvergenz in den primalen Variablen anhand von Abschätzungen für die Lipschitzkonstante  $\kappa$  hergeleitet. Äquivalent zum Lösen der KKT-Bedingungen in den primalen und dualen Variablen ist das Lösen der reduzierten Gleichungen  $r \equiv [(S^N \nabla F)^T, G^T]^T$  in den primalen Variablen, wobei  $S^N$  Kern der Jacobimatrix von  $G$  ist. Konsistente duale Variablen (Lagrange-Multiplikatoren) können über die Stationaritätsbedingung der Lagrange-Funktion (2.2a) berechnet werden. Bedingungen an die Lipschitzkonstante  $\omega$ , welche ein Maß für die Nichtlinearität der exakten KKT-Matrix ist, können zur Globalisierung des Verfahrens genutzt werden (siehe RMT im Abschnitt 2.3.2).

Die Abschätzung der Lipschitzkonstanten  $\kappa$ :

$$\begin{aligned}
\Delta w &:= \begin{pmatrix} S_{k+1}^N{}^T B_{k+1} \\ \nabla G_{k+1}^T \end{pmatrix}^{-1} \left[ \begin{pmatrix} S_k^N{}^T B_k \\ \nabla G_k^T \end{pmatrix} - \begin{pmatrix} S_k^N{}^T \nabla_w^2 L(w_k, \lambda_k) \\ \nabla G_k^T \end{pmatrix} \right] \Delta w_k \\
&= \begin{pmatrix} S_{k+1}^N{}^T B_{k+1} \\ \nabla G_{k+1}^T \end{pmatrix}^{-1} \begin{pmatrix} S_k^N{}^T [B_k - \nabla_w^2 L(w_k, \lambda_k)] \Delta w_k \\ 0 \end{pmatrix} \\
&= \begin{pmatrix} S_{k+1}^N{}^T B_{k+1} \\ \nabla G_{k+1}^T \end{pmatrix}^{-1} \left[ \begin{pmatrix} S_k^N{}^T [B_k - \nabla_w^2 L(w^*, \lambda^*)] \Delta w_k \\ 0 \end{pmatrix} \right. \\
&\quad \left. + \begin{pmatrix} S_k^N{}^T [\nabla_w^2 L(w^*, \lambda^*) - \nabla_w^2 L(w_k, \lambda_k)] \Delta w_k \\ 0 \end{pmatrix} \right] \\
\|\Delta w\| &\leq \kappa \|\Delta w_k\|
\end{aligned}$$

Wenn nun die Inverse von  $[B_{k+1}^T S_{k+1}^N, \nabla G_{k+1}]^T$  beschränkt bleibt und

$$\frac{\|S_k^N{}^T [B_k - \nabla_w^2 L(w^*, \lambda^*)] \Delta w_k\|}{\|\Delta w_k\|} \rightarrow 0 \quad \text{für } k \rightarrow \infty, \quad (2.16)$$

dann liegt superlineare Konvergenz vor, da  $S_k^N{}^T [\nabla_w^2 L(w^*, \lambda^*) - \nabla_w^2 L(w_k, \lambda_k)] \rightarrow 0$  für  $k \rightarrow \infty$  und konsistenten Lagrange-Multiplikatoren. Dabei entspricht die Bedingung  $\kappa \rightarrow 0$  und damit Bedingung (2.16) der Charakterisierung der superlinearen Konvergenz von Dennis-Moré (1977, [27]) übertragen auf den restringierten Fall.

### Bemerkung 10 (Rolle der Lagrange-Multiplikatoren)

1. Gauß-Newton-Verfahren: Die hier vorgestellten Verfahren zur Lösung von LSQ sind rein primal. D. h. die Hessematrix im LSQ-QP ist sowohl von den Multiplikatoren als auch von vergangenen Iterierten unabhängig. Selbst die Globalisierung kann mit Hilfe von Natürlichen Niveaufunktionen (siehe Bock 1987, [16]) rein primal durchgeführt werden.
2. SQP-Verfahren: Die Multiplikatoren kommen bei den hier vorgestellten Verfahren nur in der Hessematrix des QPs vor. Fletcher (1987, [37]) hat in einem Satz (12.4.1, Seite 308) gezeigt, daß die Konvergenz der Multiplikatoren in einem SQP-Verfahren

mit exakten Hessematrizen schon dann erfolgt, wenn der Startwert des Multiplikators eine nichtsinguläre KKT-Matrix zur Folge hat und die Lösungen der QPs eindeutig sind. In der Regel werden Multiplikatoren auch zur Globalisierung mittels Straffunktionen benötigt (Update der Straffunktionsparameter).

### 2.3.2 Globalisierung der Konvergenz

In Satz 5 wird Konvergenz von einem verallgemeinerten Newton-Schema innerhalb einer Kugel um ein lokales Optimum garantiert. Globalisierungsstrategien sollen nun den Konvergenzradius dieses Schemas vergrößern, indem sie gegebenenfalls den Newton-Schritt verkleinern.

*Liniensuchmethoden* verkleinern dabei den berechneten Schritt *a posteriori* durch Multiplikation mit einem Schrittweitenparameter. Im Gegensatz dazu existieren auch Methoden, die den Schritt *a priori* während der Berechnung beschränken. Diese Methoden sind auch als *Restricted-Step-* oder *Trust-Region-*Methoden bekannt.

In dieser Einführung beschränken wir uns jedoch auf Liniensuchmethoden und betrachten dabei das Problem [NLP]. Für die Globalisierung der Konvergenz ist es notwendig, Optimalität und Zulässigkeit gleichzeitig zu erfüllen. *Niveaufunktionen* beziehungsweise *Straffunktionen* sind ein Mittel dazu. Sie dienen dazu, die Qualität eines Schrittes zu beurteilen und ihn gegebenenfalls zu modifizieren. Dazu muß die Lösung  $w^*$  des [NLP] ein Minimum der Niveaufunktion sein und jeder Schritt einen hinreichenden Abstieg in der Niveaufunktion bewirken. Eine Voraussetzung für den Abstieg ist die *Kompatibilität* zwischen Schritt und Niveaufunktion. Weiterhin sind leider die Niveaufunktionen meistens *nicht* parameterfrei, das heißt, daß Parameter (an die meist unbekanntesten Multiplikatoren am Lösungspunkt) angepaßt werden müssen.

Einen anderen Ansatz stellen *Filtermethoden* dar, die Methoden der Mehrfach-Zielfunktionsoptimierung verwenden (siehe Fletcher und Leyffer 2000, [38], sowie Wächter und Biegler 2001, [98]). Bei der Globalisierung werden zwei (unabhängige) Ziele berücksichtigt: Zulässigkeit und Minimierung der Zielfunktion. In jedem SQP-Schritt sollen nun entweder die Unzulässigkeit und/oder die Zielfunktion hinreichend reduziert werden. Der Filter wird nun aus Wertepaaren (Zielfunktionswert und Maß der Unzulässigkeit) bestimmter SQP-Iterierten gebildet und soll verhindern, daß die Iterierten zykeln. Ein neues Wertepaar einer Iterierten wird nun vom Filter akzeptiert, wenn entweder die Zielfunktion oder die Unzulässigkeit reduziert wird. Die Aufnahme des Wertepaares in den Filter kann dabei so gestaltet werden, daß die Akzeptanz einer neuen SQP-Iterierten bzgl. dem Filter weniger restringent ist als bei der Reduktion einer Niveaufunktion. In einer Globalisierung mittels *a posteriori* Anpassung des Schritts durch Liniensuchmethoden wird der Schrittweitenparameter solange reduziert, bis die neue SQP-Iterierte vom Filter akzeptiert wird. Im Gegensatz zu der Globalisierung mittels Liniensuche über die  $l_1$ -Niveaufunktion (2.17) mit Strafparameter-Updates (2.21) sind diese Globalisierungsansätze nicht skalierungsinvariant bzgl. Variablen und Funktionen. Dies liegt an den hinreichenden Kriterien für Abstieg, welche für den Filter verwendet werden. Deswegen wird im folgenden nur der Ansatz über Niveau- bzw. Straffunktionen weiter verfolgt.

Eine der am besten untersuchten Niveaufunktionen ist die  $l_1$ -Niveaufunktion:

$$\phi_{\sigma,\tau}(w) = F(w) + \sum_{i \in \mathcal{E}} \sigma_i |G_i(w)| + \sum_{j \in \mathcal{I}} \tau_j H_j(w)^-. \quad (2.17)$$

wobei  $H_j(w)^- = \max\{0, -H_j(w)\}$ . Diese Niveaufunktion wird auch *exakt* genannt, da sie für endliche Werte  $\sigma_i > |\lambda_i^*|$  und  $\tau_j > \mu_j^*$  von einem Optimum  $w^*$  von [NLP] lokal minimiert wird.

Die  $l_1$ -Niveaufunktion ist eine nicht überall differenzierbare Funktion, welches Konsequenzen für die lokale Konvergenz eines SQP-Verfahrens haben kann (Maratos-Effekt: obwohl der SQP-Schritt näher zum Optimum führt, liegt für den Vollschrift kein hinreichender Abstieg in der  $l_1$ -Niveaufunktion vor und der Schritt wird verkürzt). Abhilfe dafür sind die Watchdog-Technik (siehe Chamberlain *et al.* 1982, [24]) oder die Second-Order-Correction (siehe Mayne und Polak 1982, [63] und Fletcher 1982, [36]). Die erste Technik bewirkt eine Relaxierung der Liniensuchmethode, während die zweite Technik den SQP-Schritt Richtung Zulässigkeit verschiebt. Vom Maratos-Effekt nicht betroffen sind Ansätze mit differenzierbaren Funktionen wie die von Rockafellar (1974, [79]) eingeführte exakte Funktion:

$$\begin{aligned} \bar{\phi}_{\sigma,\tau}(w, \lambda, \mu) = F(w) &+ \sum_{i \in \mathcal{E}} (-\lambda_i G_i(w) + \frac{1}{2} \sigma_i G_i(w)^2) \\ &+ \sum_{j \in \mathcal{I}} \begin{cases} -\mu_j H_j(w) + \frac{1}{2} \tau_j H_j(w)^2 & \text{wenn } H_j(w) \leq \mu_j / \tau_j \\ -\frac{1}{2} \mu_j^2 / \tau_j & \text{sonst} \end{cases}. \end{aligned}$$

Für die Globalisierung von SQP-Verfahren zur Lösung eines [NLP] wird im folgenden eine Liniensuchmethode in Kombination mit der  $l_1$ -Niveaufunktion beschrieben. Hierbei wird der Schritt  $\Delta w_k$  sowie  $\lambda_{k+1}, \mu_{k+1}$  aus dem [QP] *a posteriori* korrigiert:

$$\begin{pmatrix} w_{k+1} \\ \lambda_{k+1} \\ \mu_{k+1} \end{pmatrix} := \begin{pmatrix} w_k \\ \lambda_k \\ \mu_k \end{pmatrix} + \alpha_k \begin{pmatrix} \Delta w_k \\ \lambda_{k+1} - \lambda_k \\ \mu_{k+1} - \mu_k \end{pmatrix}.$$

Der Schrittweitenparameter  $\alpha_k$  wird dabei durch Minimierung der Niveaufunktion entlang des Schritts  $\Delta w_k$  vorgegeben:

$$\alpha_k = \arg \min_{\alpha \in (0,1]} \phi_{\sigma,\tau}(x_k + \alpha \Delta x_k).$$

Um zu zeigen, unter welchen Bedingungen ein solches  $\alpha_k \in (0, 1]$  in jedem Schritt existiert, betrachten wir die Richtungsableitung der  $l_1$ -Niveaufunktion im Punkt  $w_k$  in Richtung  $\Delta w_k$ :

$$\begin{aligned} D_{\Delta w_k} \phi_{\sigma,\tau}(w_k) &= \nabla F(w_k)^T \Delta w_k \\ &+ \sum_{i \in \tilde{\mathcal{E}}} \sigma_i (-\nabla G_i(w_k))^T \Delta w_k \\ &+ \sum_{i \in \bar{\mathcal{E}}} \sigma_i |\nabla G_i(w_k)^T \Delta w_k| \\ &+ \sum_{i \in \hat{\mathcal{E}}} \sigma_i \nabla G_i(w_k)^T \Delta w_k \\ &+ \sum_{j \in \tilde{\mathcal{I}}} \tau_j (-\nabla H_j(w_k))^T \Delta w_k \\ &+ \sum_{j \in \bar{\mathcal{I}}} \tau_j (\nabla H_j(w_k))^T \Delta w_k^- \end{aligned} \quad (2.18)$$

mit den Indexmengen der Gleichungsbeschränkungen

$$\tilde{\mathcal{E}} := \{i \mid G_i(w_k) < 0\}, \quad \bar{\mathcal{E}} := \{i \mid G_i(w_k) = 0\}, \quad \hat{\mathcal{E}} := \{i \mid G_i(w_k) > 0\}$$

und denen der Ungleichungsbeschränkungen

$$\check{\mathcal{I}} := \{j \mid H_j(w_k) < 0\}, \quad \bar{\mathcal{I}} := \{j \mid H_j(w_k) = 0\}.$$

Der Schritt  $\Delta w_k$  sei nun Lösung des Problems [QP]. Aus den KKT-Bedingungen für das Problem [QP] gewinnt man:

$$\nabla F(w_k)^T \Delta w_k = -\Delta w_k^T B_k \Delta w_k + \sum_{i \in \check{\mathcal{E}}_k \cup \hat{\mathcal{E}}_k} \lambda_{k,i} \nabla G_i(w_k)^T \Delta w_k + \sum_{j \in \check{\mathcal{I}}_k \cup \hat{\mathcal{I}}_k} \mu_{k,j} \nabla H_j(w_k)^T \Delta w_k.$$

Eingesetzt in die Richtungsableitung von  $\phi_{\sigma_k, \tau_k}$  ergibt sich:

$$\begin{aligned} D_{\Delta w_k} \phi_{\sigma_k, \tau_k}(w_k) &= -\Delta w_k^T B_k \Delta w_k \\ &\quad - \sum_{i \in \check{\mathcal{E}}_k} (\sigma_{k,i} - \lambda_{k,i}) \nabla G_i(w_k)^T \Delta w_k \\ &\quad - \sum_{i \in \hat{\mathcal{E}}_k} (\sigma_{k,i} + \lambda_{k,i}) (-\nabla G_i(w_k)^T \Delta w_k) \\ &\quad - \sum_{j \in \check{\mathcal{I}}_k} (\tau_{k,j} - \mu_{k,j}) \nabla H_j(w_k)^T \Delta w_k \\ &\quad - \sum_{j \in \hat{\mathcal{I}}_k} \mu_{k,j} (-\nabla H_j(w_k)^T \Delta w_k). \end{aligned} \quad (2.19)$$

Für die *Kompatibilität* zwischen Schritt und Niveaufunktion und damit die Existenz eines positiven Schrittweitenparameters sind folgende Bedingungen an die Parameter einzuhalten:

$$\sigma_{k,i} \geq |\lambda_{k,i}|, \quad \tau_{k,j} \geq \mu_{k,j}. \quad (2.20)$$

und die Hessematrixapproximation muß positiv semidefinit sein. Damit ist die Richtungsableitung von  $\phi_{\sigma_k, \tau_k}$  negativ und Abstieg in jedem SQP-Schritt entlang der Lösung des [QP] möglich. In der Praxis verwendet man für die Parameter die von Powell (1978, [76]) eingeführte Updateprozedur:

$$\sigma_{k,i} = \max \left\{ |\lambda_{k,i}|, \frac{1}{2}(\sigma_{k-1,i} + |\lambda_{k,i}|) \right\}, \quad \tau_{k,j} = \max \left\{ \mu_{k,j}, \frac{1}{2}(\tau_{k-1,j} + \mu_{k,j}) \right\}, \quad (2.21)$$

wobei  $\sigma_{0,i} = |\lambda_{0,i}|$  und  $\tau_{0,j} = \mu_{0,j}$  initialisiert werden. Dies hat den Vorteil, daß die Parameter auch wieder kleiner werden können und somit die Zulässigkeit gegenüber der Optimalität nicht so stark gewichtet wird. Dies ist insbesondere weit weg vom Optimum von Interesse.

Anstatt eine exakte Liniensuche in jedem SQP-Schritt durchzuführen, wird in der Praxis nur die Einhaltung sogenannter relaxierter Bedingungen gefordert - wie sie Armijo (1966) und Goldstein (1965) für den unbeschränkten Fall aufgestellt haben:

$$F(w_k + \alpha_k \Delta w_k) \leq F(w_k) + \varepsilon_1 \alpha_k \nabla F(w_k)^T \Delta w_k, \quad \varepsilon_1 \in (0, 1/2) \quad (2.22)$$

und

$$\nabla F(w_k + \alpha_k \Delta w_k)^T \Delta w_k \geq \varepsilon_2 \nabla F(w_k)^T \Delta w_k, \quad \varepsilon_2 \in (\varepsilon_1, 1).$$

Die letzte Bedingung ist dabei automatisch erfüllt, wenn ein sogenanntes *Backtracking* durchgeführt wird. Das heißt beispielsweise, daß für ein  $\beta \in (0, 1)$  eine kleinste natürliche Zahl  $l$  gesucht wird, so daß die Bedingung (2.22) für  $\alpha_k = \beta^l$  erfüllt ist.

### Satz 6 (Globale Konvergenz von SQP-Verfahren mit Liniensuche)

Ein SQP-Verfahren mit Liniensuche bzgl. der  $l_1$ -Niveaufunktion konvergiert global gegen einen KKT-Punkt unter folgenden Voraussetzungen:

- die Funktionen  $F, G_i, i \in \mathcal{E}$  und  $H_j, j \in \mathcal{I}$  sind stetig differenzierbar,
- bei jeder Liniensuche seien die Armijo-Goldstein-Bedingungen (2.22) für die  $l_1$ -Niveaufunktion erfüllbar:  $\phi_{\sigma_k, \tau_k}(w_k + \alpha_k \Delta w_k) \leq \phi_{\sigma_k, \tau_k}(w_k) + \varepsilon_1 \alpha_k D_{\Delta w_k} \phi_{\sigma_k, \tau_k}(w_k)$ ,
- die Folge  $\{w_k\}$  bleibt in einer kompakten und konvexen Menge des  $\mathbb{R}^n$ ,
- die Exaktheitsbedingung (2.20) für die Niveaufunktion ist erfüllt,
- Bedingung an die Hessematrixapproximationen  $B_k$ : es existieren  $\beta_1, \beta_2 \in (0, \infty)$ , so daß für alle  $k$  und jedes  $w \in \mathbb{R}^n$  gilt:  $\beta_1 w^T w \leq w^T B_k w \leq \beta_2 w^T w$ .

Dieser Satz wurde von Powell (1980, [77]) für ein SQP-Verfahren mit modifizierten BFGS-Updates der Lagrange-Hessematrix bewiesen. Zentral ist dabei die Kompatibilität der Niveaufunktion mit dem Schritt und damit die Positiv-Definitheit der Hessematrix. Dies ist also für modifizierte Quasi-Newton- und Gauß-Newton-Verfahren der Fall.

Einfache Beispiele zeigen jedoch, daß die Liniensuche auch zur Verlangsamung der lokalen Konvergenz führen kann. Der Grund ist, daß bei schlecht konditionierten Problemen die Niveaufunktion ein Bananental aufweisen können, so daß der Newton-Schritt stark verkleinert werden muß, um Abstieg in der Niveaufunktion zu erzeugen.

Für Least-Squares-Probleme [LSQ] mit Gauß-Newton-Verfahren existiert dabei eine in der Praxis sehr erfolgreiche Globalisierung mit einer sich in jedem Schritt ändernden Niveaufunktion, welche auch bei schlecht konditionierten Problemen Vollschritt zulassen – im Gegensatz zur  $l_1$ -Niveaufunktion. Die weitere Darstellung folgt dabei den Ideen von Bock, Kostina und Schlöder (2000, [19]) bzgl. der Globalisierung von Gauß-Newton-Verfahren mit *Natürlichen Niveaufunktionen*.

Wir betrachten nun den Gauß-Newton-Schritt  $\Delta w_k$  für ein unbeschränktes Least-Squares-Problem (die Verallgemeinerung auf restringierte Probleme ist mit Satz 4 möglich). Für die Natürliche Niveaufunktion  $\phi_k$  ist dabei der Gauß-Newton-Schritt identisch mit dem steilsten Abstieg der Niveaufunktion  $\nabla \phi_k(w_k)$ :

$$\phi_k(w) := \frac{1}{2} \|\nabla f(w_k)^+ f(w)\|_2^2. \quad (2.23)$$

Damit ist die vorhin beschriebene Beobachtung insbesondere bei schlecht konditionierten Problemen behoben, das heißt, daß der Schritt im lokalen Konvergenzbereich des Verfahrens nicht durch die Globalisierung restringiert wird. Für diese Niveaufunktion kann nun folgende Abschätzung (*Quadratic Upper Bound*) gezeigt werden:

$$\begin{aligned} \text{[QUB]} \quad & \phi_k(w_k + \alpha \Delta w_k) \leq (1 - \alpha + \frac{\alpha^2}{2} \omega_1(\alpha) \|\Delta w_k\|) \phi_k(w_k) \\ & \text{mit } \omega_1(\alpha) := \sup_{0 < \alpha_1 \leq \alpha} \frac{\|\nabla f(w_k)^+ [\nabla f(w_k + \alpha_1 \Delta w_k) - \nabla f(w_k)]\|}{\alpha_1 \|\Delta w_k\|}. \end{aligned} \quad (2.24)$$

Die folgende Liniensuche nach Armijo-Goldstein mit der Natürlichen Niveaufunktion (2.23) wird auch als *restriktiver Monotonietest* bezeichnet. Dabei wähle  $\alpha \in (0, 1]$ , so daß gilt (beachte  $D_{\Delta w_k} \phi_k(w_k) = \phi_k(w_k)$ ):

$$\begin{aligned} \text{[RMT]} \quad & 1 - \alpha + \frac{\alpha^2}{2} \omega_1(\alpha) \|\Delta w_k\| \leq 1 - \varepsilon \alpha, \quad \varepsilon \in (0, 1), \\ & \alpha \omega_1(\alpha) \|\Delta w_k\| \leq 2(1 - \varepsilon) =: \eta, \quad \eta \in (0, 2). \end{aligned} \quad (2.25)$$

**Bemerkung 11 (Erläuterung des RMT)**

1. Der restringierte Schritt  $\alpha\Delta w_k$  bleibt in einer Kugel um  $w_k$  mit Radius  $\eta/\omega$ .
2. Praktische Berechnung und Schätzung für Krümmung  $\omega_1(\alpha)$ :

$$\omega_3(\alpha) := 2 \frac{\|\nabla f(w_k)^+ [f(w_k + \alpha\Delta w_k) - (1 - \alpha)f(w_k)]\|}{\alpha^2 \|\Delta w_k\|^2} \leq \omega_1(\alpha). \quad (2.26)$$

3. Anfangswahl des Schrittweitenparameters  $\alpha_k^s$  bei Verwendung der Krümmungsinformationen  $\omega_3(\alpha_{k-1})$  des letzten Schritts :

$$\alpha_k^s := \min \left\{ 1, \frac{1}{\omega_3(\alpha_{k-1}) \|\Delta w_k\|} \right\}.$$

4. Bestimmung des Schrittweitenparameters  $\alpha$  durch Nullstellensuche von:

$$\varphi(\alpha) := \alpha\omega_3(\alpha) \|\Delta w_k\| - \eta = 0.$$

**Bemerkung 12 (Verallgemeinerung auf Problem LSQ mit Ungleichungen)**

Für eine korrekte Auswertung von  $\omega_3$  muß das Active-Set des LSQ-QPs eingefroren, Funktionswerte der Nebenbedingungen und LSQ-Zielfunktion neu berechnet sowie deren Jacobimatrizen beibehalten werden. Damit kann dann  $J_k^+ r(w_k + \alpha\Delta w_k)$  oder auch die Differenz  $J_k^+ [r(w_k + \alpha\Delta w_k) - (1 - \alpha)r(w_k)]$  direkt berechnet werden, was numerisch stabiler ist (für die Definition von  $J^+$  und  $r$  siehe Satz 4 mit  $G \equiv (G, H_{act})$ ).

**2.3.3 Adaptive Genauigkeitssteuerung**

Im folgenden werden zwei Lösungsansätze beschrieben, um eine Lösung eines Nichtlinearen Programms [NLP] mit geforderter Genauigkeit zu erhalten unter der Voraussetzung, daß die Iterierten des NLP-Algorithmus exakt berechnet werden können und die Genauigkeiten der Eingangsgrößen des NLP (Funktions- und Ableitungswerte) steuerbar sind.

Beim *sequentiellen* Ansatz wird die Genauigkeit der Eingangsgrößen in einer äußeren Schleife um den NLP-Algorithmus gesteuert. Die resultierenden gestörten NLPs werden bis zur Konvergenz ausiteriert. Die Steuerung der Genauigkeit der Eingangsgrößen geschieht mit einem *a posteriori* Fehlerschätzer angewendet auf die jeweilige gestörte NLP-Lösung. Für eine Diskussion von *a posteriori* Fehlerschätzern bei Finiten-Elemente-Methoden - siehe Becker und Rannacher (1996, [9] und 2001, [10]) sowie Nowak *et al.* (2003, [69]) für eine adaptive Zeit- und Ortsdiskretisierung bei einer räumlichen Semidiscretisierung durch Differenzenverfahren und anschließender Zeitintegration.

Beim *simultanen* Ansatz ist die Steuerung der Genauigkeit der Eingangsgrößen mit den Iterationen des NLP-Algorithmus verzahnt. Die Genauigkeit der Eingangsgrößen wird dabei so gewählt, daß das zugrundeliegende gestörte Newton(-ähnliche) Verfahren in jedem Schritt kontrahiert ( $\|\Delta \tilde{w}_{k+1}\| \leq \delta_k \|\Delta \tilde{w}_k\|$  mit  $\delta_k < 1$ ) und den (globalen) Fehler  $e(w_k)$  der Iterierten  $\Delta \tilde{w}_k$  in der Größenordnung von  $\Delta \tilde{w}_{k+1}$  wählt - siehe Bock (1981 [14], 1983 [15], 1987 [16]) im Rahmen von Gauß-Newton-Verfahren. Zuverlässige Schätzungen der Konvergenzraten  $\delta_k$  über die Lipschitzkonstanten ( $\kappa$  und  $\omega$ ) des Kontraktionsatzes (Satz

5) sind dabei notwendig, um die Genauigkeit der Eingangsgrößen nicht zu konservativ zu wählen und damit ein ineffizientes Verfahren zu erhalten.

Während der sequentielle Ansatz einfacher zu implementieren ist, ist der simultane Ansatz effektiver, da nur ein NLP-Problem gelöst werden muß, bei dem sicherzustellen ist, daß Attraktoren des ungestörten NLP-Algorithmus auch Attraktoren des gestörten NLP-Algorithmus sind. Dies ist im Rahmen von Parameterschätzproblemen mit kleiner Inkompatibilitätskonstante  $\kappa$  bei einem (verallgemeinerten) Gauß-Newton-Algorithmus keine Einschränkung - siehe Bock (1987, [16]).

Im folgenden wird beschrieben, wie der *globale Fehler* der Newton-Inkremente eines SQP-Verfahrens mit Hilfe der Fehler der Eingangsgrößen (z.B. Diskretisierungsfehler und Fehler bzgl. Meßpunktauswertung) abgeschätzt werden kann. Am Beispiel eines Parameterschätzproblems wird dies dann konkretisiert.

Wir betrachten nun folgendes Quadratische Programm in dem gestörten Inkrement  $\Delta\tilde{w} := \Delta w + \delta w$  des gestörten gleichungsbeschränkten Optimierungsproblems [EQ]:

$$\begin{aligned} \min \quad & \overbrace{[\nabla F(w) + \nabla \delta F(w)]^T}_{=: \nabla \tilde{F}(w)} (\Delta w + \delta w) + \frac{1}{2} (\Delta w + \delta w)^T \overbrace{[B + \delta B]}_{=: \tilde{B}} (\Delta w + \delta w) \\ \text{unter} \quad & \underbrace{[G(w) + \delta G(w)]}_{=: \tilde{G}(w)} + \underbrace{[\nabla G(w) + \nabla \delta G(w)]^T}_{=: \nabla \tilde{G}(w)} (\Delta w + \delta w) = 0. \end{aligned}$$

Dabei ist  $\Delta w$  die Lösung von obigem QP mit ungestörten Eingangsgrößen:  $\delta F = 0$ ,  $\delta G = 0$  und  $\delta B = 0$ . Um den globalen Fehler  $\delta w$  abzuschätzen, wird das obige QP als QP in den primalen Variablen  $\delta w$  mit festem  $\Delta w$  aufgefasst:

$$\begin{aligned} \min \quad & [\nabla \tilde{F} + \tilde{B} \Delta w]^T \delta w + \frac{1}{2} \delta w^T \tilde{B} \delta w \\ \text{unter} \quad & [\tilde{G} + \nabla \tilde{G}^T \Delta w] + \nabla \tilde{G}^T \delta w = 0. \end{aligned}$$

Nach Satz 7 existiert unter den Regularitätsannahmen [LICQ] und [PD] folgende Lösungsdarstellung:

$$\delta w = -\tilde{S}^{\mathcal{R}} [\tilde{G} + \nabla \tilde{G}^T \Delta w] - \tilde{S}^{\mathcal{N}} \underbrace{[\tilde{S}^{\mathcal{N}T} \tilde{B} \tilde{S}^{\mathcal{N}}]^{-1}}_{=: \tilde{B}^{\mathcal{N}}} \tilde{S}^{\mathcal{N}T} [\nabla \tilde{F} + \tilde{B} \delta w - \tilde{B} \tilde{S}^{\mathcal{R}} (\tilde{G} + \nabla \tilde{G}^T \Delta w)]$$

mit den Koordinaten-Basen  $\tilde{S}^{\mathcal{R}T} := [\nabla_{w_1} \tilde{G}^{-1} : 0]$  und  $\tilde{S}^{\mathcal{N}T} = [-\nabla_{w_2} \tilde{G} \nabla_{w_1} \tilde{G}^{-1} : I]$ . Gruppieren der Terme nach den Störungen der Eingangsgrößen ergibt:

$$\begin{aligned} \delta w & := \delta w_1 + \delta w_2 + \delta w_3, \\ \delta w_1 & = -[\tilde{S}^{\mathcal{R}} + \tilde{S}^{\mathcal{N}} \tilde{B}^{\mathcal{N}-1} \tilde{S}^{\mathcal{N}T} \tilde{B} \tilde{S}^{\mathcal{R}}] \delta G - [\tilde{S}^{\mathcal{N}} \tilde{B}^{\mathcal{N}-1} \tilde{S}^{\mathcal{N}T}] \nabla \delta F, \\ \delta w_2 & = -\tilde{S}^{\mathcal{N}} \tilde{B}^{\mathcal{N}-1} \tilde{S}^{\mathcal{N}T} [\nabla F + B \Delta w], \\ \delta w_3 & = -\underbrace{[(\tilde{S}^{\mathcal{R}} + \tilde{S}^{\mathcal{N}} \tilde{B}^{\mathcal{N}-1} \tilde{S}^{\mathcal{N}T} \tilde{B} \tilde{S}^{\mathcal{R}}) \nabla \delta G + \tilde{S}^{\mathcal{N}} \tilde{B}^{\mathcal{N}-1} \tilde{S}^{\mathcal{N}T} \delta B]}_{=: A}^T \Delta w. \end{aligned}$$

In einem Fixpunkt des gestörten SQP-Verfahrens gilt:  $\Delta\tilde{w} = 0$ . Für  $\|A\| \ll 1$  kann nun der Fehlerterm  $\delta w_3$  wie folgt abgeschätzt werden:

$$\begin{aligned} \|\delta w_3\| & \leq \|A\| \cdot \|\Delta w\| = \|A\| \cdot \|\delta w\| \\ & \leq \frac{\|A\|}{1 - \|A\|} \cdot \|\delta w_1 + \delta w_2\|. \end{aligned}$$

Demnach beeinflußt dieser Term zwar die Konvergenzgeschwindigkeit nicht jedoch maßgeblich die Störung des Fixpunktes des SQP-Vefahrens. Wir konzentrieren uns nun auf die Fehlerterme  $\delta w_1$  und  $\delta w_2$ . In den ersten Term gehen die Fehler der Funktionswerte der Nebenbedingungen (z.B. Diskretisierungsfehler der Zustandsgleichungen) und Fehler des Zielfunktional (z.B. bei Meßfunktionsauswertung) ein. Diese sind bei einem dynamischen Optimierungsproblem über die Integrationsgenauigkeiten kontrollierbar. Im zweiten Term geht der korrigierte Zielfunktionsgradient ein:  $\nabla F + B\Delta w$ , der im allgemeinen  $-\nabla G\lambda$  entspricht und bei einem Parameterschätzproblem [LSQ] und einer Gauß-Newton-Hessematrix mit dem Term  $\nabla f^T r$  identisch ist, wobei  $r := f + \nabla f^T \Delta w$  das exakte linearisierte Residuum ist. Im Falle von Parameterschätzproblemen [LSQ] und Gauß-Newton-Hessematrizen kann damit der globale Fehler wie folgt abgeschätzt werden:

$$\|\delta w\| \leq K_G \|\delta G\| + K_f \|\delta f\| + K_f^2 \|\nabla \delta f^T r + \nabla \delta G^T \lambda\|. \quad (2.27)$$

Der letzte Term ist dabei aus  $\delta w_2$  hergeleitet - siehe Bock (1987, [16]). Die Konstanten  $K_G$  und  $K_f$  können mit Hilfe der Zerlegungen der Hessematrix und Basen abgeschätzt werden und stehen dabei ohne weitere Kosten zur Verfügung. Es sei betont, daß durch Ausnutzen der Stationaritätsbedingungen die Berechnung der Lagrange-Multiplikatoren  $\lambda$  vermieden werden kann.

# Kapitel 3

## Ein speicherplatz- und rechenzeitsparender reduzierter SQP-Algorithmus

Ausgangspunkt für dieses Kapitel ist das folgende Nichtlineare Programm:

$$\begin{aligned} & \min F(w) && (3.1a) \\ \text{[NLP]} \quad & \text{unter } G_1(w) = 0 && (3.1b) \\ & G_2(w) = 0 && (3.1c) \\ & H(w) \geq 0. && (3.1d) \end{aligned}$$

Wie in Kapitel 1 dargestellt, sind folgende Typen von Gleichungsrestriktionen in dem diskretisierten und parameterisierten Problem vorhanden:

- Stetigkeitsbedingungen
- Konsistenzbedingungen
- Stufenübergangsbedingungen
- entkoppelte und gekoppelte Mehrpunkt-Randbedingungen

Folgende Ungleichungsrestriktionen können vorkommen:

- Kontroll- und Pfadbeschränkungen
- entkoppelte und gekoppelte Mehrpunkt-Randbedingungen

Die Zuordnung speziell der Gleichungsrestriktionen auf  $G_1$  und  $G_2$  und deren unterschiedliche Behandlung im Rahmen von SQP-Verfahren sind Thema dieses Kapitels.

### 3.1 Übersicht über partiell reduzierte SQP-Verfahren

Wie in Abschnitt 2.3.1 beschrieben können SQP-Verfahren mit Update-Hessematrizen superlineare Konvergenz aufweisen. Dabei ist die Approximation der Lagrange-Hessematrix

durch positiv definite Updates in der Praxis und in der Theorie nicht immer zufriedenstellend - siehe die Powell-Modifikation des BFGS-Updates und die verallgemeinerte Dennis-Moré-Bedingung (2.16). Die hinreichenden Bedingungen zweiter Ordnung (Satz 2) besagen, daß die Lagrange-Hessematrix am Lösungspunkt nur auf dem Kern der aktiven Restriktionen positiv definit sein muß. Es erscheint daher sinnvoll, nur mit positiv definiten Updates dieser *reduzierten* Lagrange-Hessematrix zu arbeiten. Im folgenden wird kurz dargestellt, wie dieses im Rahmen von *reduzierten SQP-Verfahren* möglich ist - siehe auch Gabay (1982, [40]). Dazu betrachten wir ein gleichungsrestringiertes Minimierungsproblem [EQ]. Die Idee ist nun, die Lösung des vollen KKT-Systems zu vermeiden und zuerst diejenigen Variablen zu eliminieren, welche durch die linearisierten Gleichungsrestriktionen bestimmt sind. Anschließend muß das Problem korrespondierend zu den übrigen Variablen reduziert werden. Eine verallgemeinerte Elimination wird durch folgenden Satz beschrieben (siehe Fletcher, 1987 [37], Seite 234).

**Satz 7 (Verallgemeinerte Elimination im KKT-System von EQ)**

Im Problem [EQ] seien die Funktionen  $F : \mathbb{R}^n \rightarrow \mathbb{R}$  und  $G : \mathbb{R}^n \rightarrow \mathbb{R}^m$  zweimal stetig differenzierbar und es gelte LICQ am Punkt  $w_k$  sowie die Positiv-Definitheit-Bedingung an die Lagrange-Hessematrix (siehe Bemerkung 4). Sei  $V_k \in \mathbb{R}^{n \times (n-m)}$  eine Basisergänzung des  $\mathbb{R}^n$  zu  $\nabla G_k := \nabla G(w_k)$ , so daß  $[\nabla G_k : V_k]^T$  invertierbar ist mit der Inversen  $[S_k^R : S_k^N]$ , wobei  $S_k^R \in \mathbb{R}^{n \times m}$  und  $S_k^N \in \mathbb{R}^{n \times (n-m)}$ . Dann folgt  $\nabla G_k^T S_k^R = I$  und  $\nabla G_k^T S_k^N = 0$ . Sei  $\Delta w_k$  eine Lösung des quadratischen Programms von EQ mit Lagrange-Hessematrix  $\nabla_w^2 L_k$ , dann gilt bzgl. der Basistransformation mit  $[\nabla G_k : V_k]$ :

$$\begin{aligned} \Delta w_k &= S_k^R y_k^R + S_k^N y_k^N \\ \text{mit } y_k^N &= -(S_k^{NT} \nabla_w^2 L_k S_k^N)^{-1} S_k^{NT} (\nabla F_k - \nabla_w^2 L_k S_k^R G_k) \\ \text{und } y_k^R &= -G_k. \end{aligned}$$

Man nennt  $y_k^R$  auch die Bildraumkomponente und  $y_k^N$  die Nullraumkomponente. Die Matrix  $B_k^N := S_k^{NT} \nabla_w^2 L_k S_k^N$  wird auch *reduzierte Lagrange-Hessematrix* genannt. Der zweite Teil in  $y_k^N$  wird *Crossterm* genannt, weil die Lagrange-Hessematrix sowohl bzgl. der Nullraumbasis  $S_k^N$  als auch der Bildraumbasis  $S_k^R$  projiziert wird.

**Bemerkung 13 (Wahl der Basisergänzung in der Basistransformation)**

1. *Koordinaten-Basis:* Sei  $w = (w_1, w_2)$  mit  $\dim(w_1) = m$ ,  $\nabla_{w_1} G_k$  invertierbar und  $V_k^T = [0 : I]$ . Dann gilt:  $S_k^{RT} = [\nabla_{w_1} G_k^{-1} : 0]$  und  $S_k^{NT} = [-\nabla_{w_2} G_k \nabla_{w_1} G_k^{-1} : I]$ . Diese Basis hat den Vorteil, daß die dünnbesetzte Struktur von  $\nabla_{w_1} G_k$  bei der Faktorisierung ausgenutzt werden kann.
2. *orthogonale Basis:* Sei  $\nabla_w G_k = [Q_1 : Q_2][R^T : 0]^T$  mit  $Q := [Q_1 : Q_2]$  orthogonal,  $R$  obere Dreiecksmatrix und  $V_k = Q_2$ . Dann gilt:  $S_k^{RT} = [R^{-1} Q_1^T] = \nabla G_k^+$  und  $S_k^{NT} = [Q_2^T]$ . Diese Basis hat den Vorteil, zu numerisch stabilerer Elimination zu führen, jedoch ist die Stetigkeit der Basis ein Problem (Pivoting bei der QR-Faktorisierung).

Ein *reduziertes SQP-Verfahren* bzgl. einer Koordinaten-Basis und der reduzierten Lagrange-Hessematrix-Approximation  $B_k^N$  löst in der Iteration  $k$  das folgende reduzierte System:

$$\begin{pmatrix} 0 & 0 & \nabla_{w_1} G_k \\ 0 & B_k^N & \nabla_{w_2} G_k \\ \nabla_{w_1} G_k^T & \nabla_{w_2} G_k^T & \end{pmatrix} \begin{pmatrix} \Delta w_{1,k} \\ \Delta w_{2,k} \\ \lambda_{k+1} \end{pmatrix} = - \begin{pmatrix} \nabla_{w_1} F_k \\ \nabla_{w_2} F_k \\ G_k \end{pmatrix}, \quad (3.2)$$

um die Iterierten  $w_{k+1} = w_k + (\Delta w_{1,k}^T, \Delta w_{2,k}^T)^T$  und  $\lambda_{k+1}$  zu bestimmen. Das heißt, daß der Crossterm in der Berechnung der Nullraumkomponente nicht mitberücksichtigt wird. Dies hat den Vorteil, daß nur die reduzierte Lagrange-Hessematrix  $B_k^N$  gespeichert werden muß. Wie wir später sehen werden, führt dies jedoch zu einer 2-Schritt quadratischen bzw. superlinearen Konvergenz (bzgl. dem Quotientenkriterium bei Verwendung einer reduzierten exakten bzw. Quasi-Newton-Update Lagrange-Hessematrix). Eine Erklärung ist, daß in Iteration  $k$  der Schritt  $\Delta w_k$  hauptsächlich in Richtung Zulässigkeit geht und in der nächsten Iteration  $k+1$  dann  $G_{k+1} = G(w_{k+1}) \approx 0$  und damit der Crossterm sowie die Bildraumkomponente vernachlässigbar sind und damit der Schritt  $\Delta w_{k+1}$  hauptsächlich in Richtung Optimalität geht.

Um bei der Lösung des Systems (3.2) zu vermeiden, daß die gesamte Jacobimatrix  $\nabla G^T = (\nabla_{w_1} G^T, \nabla_{w_2} G^T)$  berechnet werden muß, wird eine Transformation im Variablen- und Funktionenraum durchgeführt:

$$\begin{pmatrix} 0 & 0 & I \\ 0 & B_k^N & \nabla_{w_2} G_k \nabla_{w_1} G_k^{-1} \\ I & \nabla_{w_1} G_k^{-T} \nabla_{w_2} G_k^T & \end{pmatrix} \begin{pmatrix} \Delta w_{1,k} \\ \Delta w_{2,k} \\ \nabla_{w_1} G_k \lambda_{k+1} \end{pmatrix} = - \begin{pmatrix} \nabla_{w_1} F_k \\ \nabla_{w_2} F_k \\ \nabla_{w_1} G_k^{-T} G_k \end{pmatrix}. \quad (3.3)$$

Beim Lösen dieses Systems werden anstatt der Lagrangeparameter  $\lambda_{k+1}$  für die Gleichungsbedingungen  $G(w)$  nun Lagrangeparameter  $\hat{\lambda}_{k+1} := \nabla_{w_1} G_{1,k} \cdot \lambda_{k+1}$  für die transformierten Gleichungsbedingungen  $\nabla_{w_1} G_1(w) \cdot G_1(w)$  berechnet. Dies muß in den Globalisierungsstrategien berücksichtigt werden (siehe auch Gill *et al.* 2000, [42]). In der Lösung eines reduzierten SQP-Verfahrens, welches den Schritt  $(\Delta w_{1,k}^T, \Delta w_{2,k}^T)^T$  durch (3.3) berechnet, liegt jedoch auch eine Lösung des reduzierten SQP-Verfahrens mit (3.2) vor, da für  $\nabla_{w_1} G(w_k)^{-1} G(w_k) \rightarrow 0$  gilt:  $G(w_k) \rightarrow 0$ . Dies liegt an der Annahme für die Koordinaten-Basis (siehe Bemerkung 13).

Die Verallgemeinerung des Reduktionsansatzes auf den Fall [NLP] mit Ungleichungen ist schwierig, da sich die Basen  $[S_k^R : S_k^N]$  bei Wechsel der Anzahl der aktiven Beschränkungen nicht stetig ändern. Ausgehend von der reduzierten Hessematrix  $B_k^N$  ist  $B_k := V_k B_k^N V_k^T$  die bestmögliche Approximation von  $\nabla_w^2 L_k$ . Damit läßt sich dann eine neue reduzierte Hessematrix berechnen:  $S_{k+1}^N{}^T B_k S_{k+1}^N$ .

*Partiell reduzierte SQP-Verfahren* versuchen die Vorteile von SQP- und RSQP-Verfahren in sich zu vereinen. Der Begriff partiell reduzierte SQP-Verfahren wurde geprägt von Schulz (1996, [83]) und hebt hervor, daß der Reduktionsansatz im SQP-Verfahren nur auf einen festen Teil  $G_1$  der Gleichungsrestriktionen  $G$  angewendet wird. Prinzipiell unterscheiden sich partiell reduzierte SQP-Verfahren in folgenden Punkten:

- Wahl der Gleichungsrestriktionen in  $G_1$  und  $G_2$ .
- Berücksichtigung von Informationen zweiter Ordnung (Crossterm) bei der Berechnung der Nullraumkomponente.

- Approximation der reduzierten Lagrange-Hessematrix bei der Berechnung der Nullraumkomponente.
- Wahl und Berechnen der Basis (Koordinaten-Basis oder orthogonale Basis)
- Berechnung der Bildraumkomponente und damit Berechnung der Lagrange-Multiplikatoren der Gleichungsrestriktionen im Reduktionsansatz.

## 3.2 Ein reduzierter SQP-Algorithmus für Probleme mit wenigen Freiheitsgraden

Ziel der im folgenden beschriebenen partiell reduzierten SQP-Algorithmen ist es, möglichst viele Konvergenz-Eigenschaften der nicht reduzierten SQP-Algorithmen zu erhalten. Im Rahmen von Optimal-Steuerungsproblemen, parametrisiert durch den Randwertproblem-Ansatz und der Mehrzielmethode, existieren charakteristische Strukturen im KKT-System.

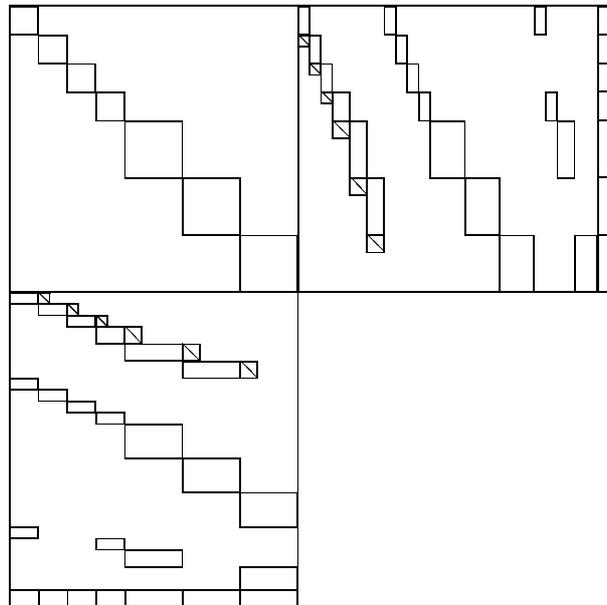


Abbildung 3.1: KKT-Matrix des Randwertproblems

Die Hessematrix des KKT-Systems ist aufgrund der *linearen* Kopplungen innerhalb der Stetigkeitsbedingungen und der Mehrpunktrandbedingungen blockdiagonal bezogen auf die Variablen zwischen den Mehrzielknoten. Dies wird in den direkten Mehrfach-Schießverfahren von Bock, Plitt (1981, [73] und 1984, [20]) und Leineweber *et al.* (2003, [58] und [59]) durch *High-Rank-Updates* ausgenutzt und führt zu einer Beschleunigung der Konvergenz. Leider läßt sich diese Eigenschaft nicht in einem partiell reduzierten Verfahren retten, welches die Stetigkeitsbedingungen zur Reduktion verwendet! Der Grund ist, daß dazu die Hessematrix-Updates im *vollem* Raum der Variablen durchgeführt werden

müßte und nicht im Kern der aktiven Restriktionen.

Bei den in dieser Arbeit vorgestellten Anwendungsproblemen, bei denen die Dynamik zum Teil als PDAE vorliegt, haben wir es mit einer großen Anzahl von algebraischen *und* differentiellen Zustandsvariablen zu tun. Meistens sind die Anfangswerte der Zustände meist durch (nichtlineare) Gleichungsbedingungen festgelegt.

Wie wir im nächsten Abschnitt zeigen werden, kann die Berechnung eines SQP-Schritts in einem partiell reduzierten SQP-Verfahren erheblich beschleunigt werden, wenn die Anfangsbedingungen an die Zustände, die Stetigkeitsbedingungen sowie die Konsistenzbedingungen in  $G_1$  gruppiert und dann lediglich  $G_1$  zur Berechnung einer Koordinaten-Basis im PRSQP-Verfahren eingesetzt wird. Die Berechnung der Koordinaten-Basis kann rekursiv erfolgen und die Anzahl der Richtungsableitungen auf jedem Mehrzielknoten wird soweit reduziert, daß sie unabhängig von der Modelldimension ist!

In jeder Iteration wird ein System gelöst, welches eine Verallgemeinerung des Systems (3.3) auf den PRSQP-Fall darstellt (hier ohne Ungleichungen  $H_k$ ):

$$\begin{aligned} & \begin{pmatrix} 0 & 0 & I & 0 \\ 0 & B_{1,k}^{\mathcal{N}} & \nabla_{w_2} G_{1,k} \nabla_{w_1} G_{1,k}^{-1} & S_{1,k}^{\mathcal{N}T} \nabla G_{2,k} \\ I & \nabla_{w_1} G_{1,k}^{-T} \nabla_{w_2} G_{1,k}^T & & \\ 0 & \nabla G_{2,k}^T S_{1,k}^{\mathcal{N}} & & \end{pmatrix} \begin{pmatrix} \Delta w_{1,k} \\ \Delta w_{2,k} \\ \hat{\lambda}_{1,k+1} \\ \hat{\lambda}_{2,k+1} \end{pmatrix} \\ & = - \begin{pmatrix} \nabla_{w_1} F_k \\ \nabla_{w_2} F_k \\ \nabla_{w_1} G_{1,k}^{-T} G_{1,k} \\ G_{2,k} - \nabla G_{2,k}^T S_{1,k}^{\mathcal{R}} G_{1,k} \end{pmatrix}. \end{aligned} \quad (3.4)$$

Dabei ist  $[S_{1,k}^{\mathcal{R}} : S_{1,k}^{\mathcal{N}}]$  die Koordinaten-Basis bzgl. der Jacobimatrix von  $G_{1,k}$ . Die Lagrange-Multiplikatoren  $\lambda_{k+1}^T := (\lambda_{1,k+1}^T, \lambda_{2,k+1}^T)^T$  des nicht modifizierten Systems (3.2) stehen in folgender Beziehung zu den Multiplikatoren  $\hat{\lambda}_{k+1} := (\hat{\lambda}_{1,k+1}^T, \hat{\lambda}_{2,k+1}^T)$  des modifizierten Systems (3.4):

$$\begin{pmatrix} \hat{\lambda}_{1,k+1} \\ \hat{\lambda}_{2,k+1} \end{pmatrix} = \begin{pmatrix} \nabla_{w_1} G_{1,k} & \nabla_{w_1} G_{2,k} \\ & I \end{pmatrix} \begin{pmatrix} \lambda_{1,k+1} \\ \lambda_{2,k+1} \end{pmatrix}.$$

Es sei darauf hingewiesen, daß  $\hat{\lambda}_{2,k+1} = \lambda_{2,k+1}$ . Es kann jedoch gezeigt werden, daß auf die Berechnung der Lagrange-Parameter  $\hat{\lambda}_{1,k+1}$  bzw.  $\lambda_{1,k+1}$  im Rahmen von Globalisierungsstrategien bei partiell reduzierten SQP-Verfahren verzichtet werden kann. Dies vereinfacht auch die Berechnung des Schritts

$$\Delta w := (\Delta w_{1,k}^T, \Delta w_{2,k}^T)^T := S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}} + S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}}$$

eines partiell reduzierten SQP-Verfahrens für das Problem [NLP]. Die Nullraumkomponente  $y_k^{\mathcal{N}}$  ist hier Lösung von folgendem reduzierten Quadratischen Programm

$$[\text{RQP}] \quad \min \quad \nabla F_k^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}} + \frac{1}{2} y_k^{\mathcal{N}T} B_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}} \quad (3.5a)$$

$$\text{unter} \quad G_{2,k} + \nabla G_{2,k}^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}} + \nabla G_{2,k}^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}} = 0 \quad (3.5b)$$

$$H_k + \nabla H_k^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}} + \nabla H_k^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}} \geq 0. \quad (3.5c)$$

Die Bildraumkomponente sei durch den ersten Teil der Gleichungsbedingungen bestimmt:  $y_k^{\mathcal{R}} := -G_{1,k}$ .

### 3.2.1 Rekursive Berechnung der Koordinaten-Basis

Die Koordinaten-Basis zeichnet sich dadurch aus, daß die Struktur in der Jacobimatrix  $\nabla_{w_1} G_{1,k}$  ausgenutzt werden kann. Im vorliegenden Fall sind in  $G_1$  die Anfangsbedingungen  $d_0(s_0^x, s_0^z, p)$  an die differentiellen Zustände  $s_0^x$  am ersten Mehrzielknoten, die Konsistenzbedingungen  $g_i(s_i^x, s_i^z, u_i, p)$  an die algebraischen Zustände  $s_i^z$  mit  $i := 0, \dots, m$  und die Stetigkeitsbedingungen  $c(s_{i+1}^x, s_i^x, s_i^z, u_i, p) := x_i(\tau_{i+1}, s_i^x, s_i^z, u_i, p) - s_{i+1}^x$  an die übrigen differentiellen Zustände  $s_i^x$  mit  $i := 0, \dots, m-1$  gruppiert. Dabei entsprechen  $u_i$  den parametrisierten Steuerungen an den Mehrzielknoten  $i := 0, \dots, m$  und  $p$  den globalen Parametern. Die NLP-Variablen  $w := (s_0^x, s_0^z, \dots, s_m^x, s_m^z, u_0, \dots, u_m, p)$  sind dabei wie folgt aufgespalten:

$$w_1 := (s_0^x, s_0^z, \dots, s_m^x, s_m^z), \quad w_2 := (u_0, u_1, \dots, u_m, p).$$

Folgende Abkürzungen für die Jacobimatrizen in  $G_1$  werden verwendet:

$$D_0^{x,z,p} := \frac{\partial d_0}{\partial (s_0^x, s_0^z, p)}, \quad X_i^{x,z,u,p} := \frac{\partial x_i}{\partial (s_i^x, s_i^z, u_i, p)}, \quad G_i^{x,z,u,p} := \frac{\partial g_i}{\partial (s_i^x, s_i^z, u_i, p)}.$$

Die spezielle Struktur der Jacobimatrizen  $\nabla_{w_1} G_{1,k}$  und  $\nabla_{w_2} G_{1,k}$  ermöglicht die rekursive Berechnung der Koordinaten-Basis  $[S_{1,k}^R : S_{1,k}^N]$ . Nachfolgend die Ableitung von  $G_1$  nach den abhängigen Variablen  $w_1$

$$\nabla_{w_1} G_{1,k}^T := \left( \begin{array}{c} \left[ \begin{array}{cc} D_0^x & D_0^z \\ G_0^x & G_0^z \\ X_0^x & X_0^z \end{array} \right] \quad \left[ \begin{array}{cc} -I & \\ G_1^x & G_1^z \end{array} \right] \\ \ddots & \ddots \\ \left[ \begin{array}{cc} X_{m-1}^x & X_{m-1}^z \end{array} \right] \quad \left[ \begin{array}{cc} -I & \\ G_m^x & G_m^z \end{array} \right] \end{array} \right)$$

sowie die Ableitung nach den freien Variablen  $w_2$

$$\nabla_{w_2} G_{1,k}^T := \left( \begin{array}{c} D_0^p \\ G_0^p \\ X_0^p \\ G_1^p \\ \vdots \\ X_{m-1}^p \\ G_m^p \end{array} \right).$$

Die Inverse der Ableitung nach den abhängigen Variablen hat folgende Dreiecksgestalt:

$$\nabla_{w_1} G_{1,k}^{T^{-1}} := \left( \begin{array}{cccc} A_0^{-1} & & & \\ \hat{X}_0 A_0^{-1} & A_1^{-1} & & \\ \hat{X}_1 \hat{X}_0 A_0^{-1} & \hat{X}_1 A_1^{-1} & A_2^{-1} & \\ \vdots & \vdots & & \ddots \\ (\prod_{j=m-1}^0 \hat{X}_j) A_0^{-1} & (\prod_{j=m-1}^1 \hat{X}_j) A_1^{-1} & \cdots & \hat{X}_{m-1} A_{m-1}^{-1} \quad A_m^{-1} \end{array} \right).$$

Dabei wurden folgende Abkürzungen benutzt:

$$A_0 := \begin{bmatrix} D_0^x & D_0^z \\ G_0^x & G_0^z \end{bmatrix}, \quad A_j := \begin{bmatrix} -I & 0 \\ G_j^x & G_j^z \end{bmatrix}, \quad A_j^{-1} = \begin{bmatrix} -I & 0 \\ G_j^{z^{-1}} G_j^x & G_j^{z^{-1}} \end{bmatrix},$$

$$\bar{X}_i := \begin{bmatrix} X_i^x & X_i^z \\ 0 & 0 \end{bmatrix}, \quad \hat{X}_i := -A_{i+1}^{-1} \bar{X}_i = \begin{bmatrix} X_i^x & X_i^z \\ -G_{i+1}^{z^{-1}} G_{i+1}^x X_i^x & -G_{i+1}^{z^{-1}} G_{i+1}^x X_i^z \end{bmatrix},$$

für  $j = 1, \dots, m$  und  $i = 0, \dots, m-1$ . Die Zeilen der Inverse können rekursiv berechnet werden, indem die  $i$ -te Zeile mit der verallgemeinerten Sensitivität  $\hat{X}_i$  multipliziert wird und ein neues Diagonalelement  $A_i^{-1}$  hinzukommt. Dieses ist leicht und billig zu berechnen, da die Jacobimatrizen der Konsistenzbedingungen bzgl. der algebraischen Zustände  $G_i^z$  in der Regel dünnbesetzt sind und mit entsprechenden Routinen (z. B. MA48, [78]) effizient zerlegt werden können.

Damit läßt sich die Bildraumbasis  $S_{1,k}^{\mathcal{R}}$  berechnen. Für die Nullraumbasis  $S_{1,k}^{\mathcal{N}}$  muß die Inverse noch mit der Jacobimatrix bzgl. der freien Variablen multipliziert werden. Nachfolgend ist der nichttriviale Teil der Nullraumbasis  $S_{1,k}^{\mathcal{N}T} = [-\nabla_{w_2} G_{1,k} \nabla_{w_1} G_{1,k}^{-1} : I]$  angegeben bzgl. der Steuerungen  $u_0$  am ersten Mehrzielknoten:

$$(\nabla_{w_1} G_{1,k}^{T-1} \nabla_{w_2} G_{1,k}^T)|^{u_0} := \begin{pmatrix} A_0^{-1} \begin{pmatrix} 0 \\ G_0^u \end{pmatrix} \\ \hat{X}_0 A_0^{-1} \begin{pmatrix} 0 \\ G_0^u \end{pmatrix} + A_1^{-1} \begin{pmatrix} X_0^u \\ 0 \end{pmatrix} \\ \hat{X}_1 \left[ \hat{X}_0 A_0^{-1} \begin{pmatrix} 0 \\ G_0^u \end{pmatrix} + A_1^{-1} \begin{pmatrix} X_0^u \\ 0 \end{pmatrix} \right] \\ \vdots \\ \hat{X}_{m-1} \left[ (\Pi_{j=m-2}^0 \hat{X}_j) A_0^{-1} \begin{pmatrix} 0 \\ G_0^u \end{pmatrix} + (\Pi_{j=m-2}^1 \hat{X}_j) A_1^{-1} \begin{pmatrix} X_0^u \\ 0 \end{pmatrix} \right] \end{pmatrix}$$

sowie bzgl. der Steuerungen  $u_i$  am  $i$ -ten Mehrzielknoten:

$$(\nabla_{w_1} G_{1,k}^{T-1} \nabla_{w_2} G_{1,k}^T)|^{u_i} := \begin{pmatrix} 0 \\ \vdots \\ 0 \\ A_i^{-1} \begin{pmatrix} 0 \\ G_i^u \end{pmatrix} \\ \hat{X}_i A_i^{-1} \begin{pmatrix} 0 \\ G_i^u \end{pmatrix} + A_{i+1}^{-1} \begin{pmatrix} X_i^u \\ 0 \end{pmatrix} \\ \vdots \\ \hat{X}_{m-1} \left[ (\Pi_{j=m-2}^i \hat{X}_j) A_i^{-1} \begin{pmatrix} 0 \\ G_i^u \end{pmatrix} + (\Pi_{j=m-2}^{i+1} \hat{X}_j) A_{i+1}^{-1} \begin{pmatrix} X_i^u \\ 0 \end{pmatrix} \right] \end{pmatrix}.$$

Zu sehen ist, daß bei der Basis bzgl.  $u_0$  eine Zeile durch Multiplikation der vorherigen Zeile mit der verallgemeinerten Sensitivität  $\hat{X}_i$  gewonnen wird. Bei der Basis bzgl.  $u_i$  werden die Zeilen ebenso durch Multiplikation mit der verallgemeinerten Sensitivität  $\hat{X}_i$  gewonnen

mit der Ausnahme in der  $(i + 1)$ -ten Zeile, bei der noch ein Term mit  $X_i^u$  addiert wird. Bei dem nichttrivialen Teil der Nullraumbasis bzgl. der Parameter  $p$ :

$$(\nabla_{w_1} G_{1,k}^{T-1} \nabla_{w_2} G_{1,k}^T)^p := \begin{pmatrix} A_0^{-1} \begin{pmatrix} D_0^p \\ G_0^p \end{pmatrix} \\ \hat{X}_0 A_0^{-1} \begin{pmatrix} D_0^p \\ G_0^p \end{pmatrix} + A_1^{-1} \begin{pmatrix} X_0^p \\ G_1^p \end{pmatrix} \\ \hat{X}_1 \left[ \hat{X}_0 A_0^{-1} \begin{pmatrix} D_0^p \\ G_0^p \end{pmatrix} + A_1^{-1} \begin{pmatrix} X_0^p \\ G_1^p \end{pmatrix} \right] + A_2^{-1} \begin{pmatrix} X_1^p \\ G_2^p \end{pmatrix} \\ \vdots \\ (\prod_{j=m-1}^0 \hat{X}_j) A_0^{-1} \begin{pmatrix} D_0^p \\ G_0^p \end{pmatrix} + \sum_{i=1}^m \left[ (\prod_{j=m-1}^i \hat{X}_j) A_i^{-1} \begin{pmatrix} X_{i-1}^p \\ G_i^p \end{pmatrix} \right] \end{pmatrix}$$

haben wir folgende Rekursion vorliegen: zuerst Multiplikation der  $i$ -ten Zeile mit der verallgemeinerten Sensitivität  $\hat{X}_i$  und anschließender Addition eines Terms, der  $X_i^p$  und  $G_{i+1}^p$  enthält.

Wichtig ist, daß die verallgemeinerten Sensitivitäten  $\hat{X}_i$  nicht vollständig berechnet werden müssen, sondern nur Produkte mit modifizierten Ableitungen nach Steuerungen und Parametern. Das heißt, daß die Anzahl der Richtungsableitungen in der Koordinaten-Basis nun unabhängig von der Modelldimension ist, wenn die Anfangsbedingungen die differentiellen Zustände vollständig bestimmen. Wie dies effizient im Rahmen von BDF-Integratoren mit hoher Genauigkeit (siehe IND-Prinzip) geschehen kann, wird im Kapitel 4 erläutert.

### 3.2.2 Hessematrix-Updates bei allgemeinen Optimierungsproblemen

Das partiell reduzierte SQP-Verfahren mit Berechnung des Schritts durch (3.5) besitzt im Fall des parameterisierten Optimal-Steuerungsproblem folgende partiell reduzierte KKT-Matrix:

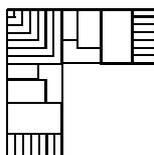


Abbildung 3.2: partiell reduzierte KKT-Matrix des Randwertproblems

Wie man sehen kann, ist die Hessematrix nun nicht mehr blockdiagonal und die entkoppelten Randbedingungen am  $i$ -ten Mehrzielknoten hängen nun nicht mehr nur von den Variablen  $(x_i, z_i, u_i, p)$  sondern von den Steuerungen und Parametern  $(u_0, \dots, u_i, p)$  ab. Dies liegt daran, daß über die Stetigkeitsbedingungen eine Kopplung mit den Variablen an den vorherigen Mehrzielknoten eingeführt wird. Deutlich wird dies an den Einträgen

der Basis  $S_{1,k}^{\mathcal{N}}$ , welche bzgl. den Steuerungen eine untere Dreiecksmatrix darstellt. Um den Aufwand für die Berechnung einer reduzierten Hessematrix-Approximation  $B_{1,k}^{\mathcal{N}}$  gering zu halten, werden BFGS-Updates verwendet. Dazu wird der reduzierte Lagrange-Gradient definiert, der unabhängig von den Lagrange-Multiplikatoren  $\lambda_1$  der Gleichungsbedingungen  $G_1$  im Problem [NLP] ist:

$$\begin{aligned}\nabla_{y^{\mathcal{N}}} L(w_k + S_{1,k}^{\mathcal{N}} y^{\mathcal{N}}, \lambda, \mu)|_{y^{\mathcal{N}}=0} &= S_{1,k}^{\mathcal{N}T} \nabla_w L(w_k, \lambda, \mu) \\ &= \nabla F_k^T S_k^{\mathcal{N}} - (\nabla G_{2,k}^T S_k^{\mathcal{N}})^T \lambda_2 - (\nabla H_k^T S_k^{\mathcal{N}})^T \mu\end{aligned}$$

Damit lassen sich verschiedene BFGS-Updates  $U_{BFGS}(B_{1,k}^{\mathcal{N}}, \delta_k^{\mathcal{N}}, \gamma_k^{\mathcal{N}})$ , siehe (2.10), der reduzierten Hessematrix definieren, welche sich in der Definition der Lagrange-Gradienten-Differenz  $\gamma_k^{\mathcal{N}}$  und sich damit an der Position der Auswertungen unterscheiden. Sei nun  $\delta_k^{\mathcal{N}} := y_k^{\mathcal{N}}$ . Bzgl. dieser Informationen im Variablenraum lassen sich entsprechende Informationen im Funktionenraum der Gradienten definieren:

$$\gamma_k^{\mathcal{N}} := S_{1,k+1}^{\mathcal{N}T} \nabla_w L(w_{k+1}, \lambda_{k+1}, \mu_{k+1}) - S_{1,k}^{\mathcal{N}T} \nabla_w L(w_k, \lambda_{k+1}, \mu_{k+1}) \quad (3.6a)$$

oder alternativ

$$\gamma_k^{\mathcal{N}} := S_{1,k}^{\mathcal{N}}(w_k^+)^T \nabla_w L(w_k^+, \lambda_{k+1}, \mu_{k+1}) - S_{1,k}^{\mathcal{N}T} \nabla_w L(w_k, \lambda_{k+1}, \mu_{k+1}) \quad (3.6b)$$

beziehungsweise

$$\gamma_k^{\mathcal{N}} := S_{1,k+1}^{\mathcal{N}T} \nabla_w L(w_{k+1}, \lambda_{k+1}, \mu_{k+1}) - S_{1,k}^{\mathcal{N}}(w_{k+1}^-)^T \nabla_w L(w_{k+1}^-, \lambda_{k+1}, \mu_{k+1}) \quad (3.6c)$$

wobei  $w_k^+ := w_k + S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}}$  und  $w_{k+1}^- := w_k + S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}}$ . Die erste Definition (3.6a) ist ohne Zwischenauswertungen durchführbar und damit deutlich billiger als die beiden letzten Definitionen (3.6b, 3.6c), welche jedoch zu numerisch stabileren Update-Matrizen führen (siehe auch Schulz 1996, [83]). Im Abschnitt 3.3 über lokale Konvergenz von partiell reduzierten SQP-Verfahren wird die Dennis-Moré-Bedingung (2.16) für superlineare Konvergenz erweitert, so daß die asymptotische Konvergenz nur bzgl. der vollreduzierten Hessematrix benötigt wird. Damit wird deutlich, daß positiv definite Updates der (voll) reduzierten Hessematrix auch im Hinblick auf superlineare Konvergenz sinnvoll erscheinen.

### 3.2.3 Gauß-Newton Hessematrix für Least-Squares Probleme

Wie im Abschnitt 2.3 erwähnt, ist es für LSQ-Probleme mit kleinem Residuum der LSQ-Zielfunktion vorteilhaft, Gauß-Newton Hessematrizen zu verwenden. Das heißt, daß wir die LSQ-Zielfunktion innerhalb der Norm linearisieren, um das Quadratische Problem [LSQ-QP] aufzustellen. Eine Partitionierung des Schritts, wie in Satz 7 durch Einführen einer neuen Basis  $[S_{1,k}^{\mathcal{R}} : S_{1,k}^{\mathcal{N}}]$  bzgl.  $\nabla G_{1,k}^T$ , führt zu folgendem reduzierten LSQ-Problem (vergleiche mit 3.5):

$$\begin{aligned}[\text{LSQ-RQP}] \quad \min \quad & \|f_k + \nabla f_k^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}} + \nabla f_k^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}}\|_2^2 \quad (3.7a)\end{aligned}$$

$$\text{unter } G_{2,k} + \nabla G_{2,k}^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}} + \nabla G_{2,k}^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}} = 0 \quad (3.7b)$$

$$H_k + \nabla H_k^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}} + \nabla H_k^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}} \geq 0. \quad (3.7c)$$

Dessen Lösung sei nun  $y_k^{\mathcal{N}}$  und die Bildraumkomponente wie bisher  $y_k^{\mathcal{R}} := -G_{1,k}$ . Um die Eigenschaften dieses Quadratischen Problems zu verdeutlichen, wird die Zielfunktion

ausformuliert:

$$\begin{aligned} \|f_k + \nabla f_k^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}} + \nabla f_k^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}}\|_2^2 &= (f_k + \nabla f_k^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}})^T (f_k + \nabla f_k^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}}) \\ &\quad + f_k^T \nabla f_k^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}} \\ &\quad + \underbrace{y_k^{\mathcal{R}T} S_{1,k}^{\mathcal{R}T} \nabla f_k \nabla f_k^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}}}_{\text{Crossterm}} \end{aligned} \quad (3.8a)$$

$$\begin{aligned} &\quad + y_k^{\mathcal{N}T} \underbrace{S_{1,k}^{\mathcal{N}T} \nabla f_k \nabla f_k^T S_{1,k}^{\mathcal{N}} y_k^{\mathcal{N}}}_{\text{red. GN-Hm}}. \end{aligned} \quad (3.8b)$$

Die beiden interessanten Terme sind der Crossterm, (3.8a), sowie die reduzierte Gauß-Newton-Hessematrix in (3.8b). Beide können im Gegensatz zur Zielfunktion in [RQP], siehe (3.5), leicht berechnet werden, da nicht die Produkte der reduzierten Jacobimatrizen  $S_{1,k}^{\mathcal{N},\mathcal{R}T} \nabla f_k \nabla f_k^T S_{1,k}^{\mathcal{N},\mathcal{R}}$  berechnet und gespeichert werden, sondern die reduzierten Jacobimatrizen  $\nabla f_k^T S_{1,k}^{\mathcal{N},\mathcal{R}}$  selbst. Die Verwendung beider Terme hat Konsequenzen bzgl. der lokalen Konvergenz. So ist die Konvergenzrate, ebenso wie beim Vollraum-Gauß-Newton-Verfahren (2.11), 1-Schritt linear! Was sich hierbei ändert, ist lediglich die lineare Algebra zur Berechnung des Schritts - mathematisch sind beide Verfahren äquivalent (siehe auch Schlöder 1988, [82]).

### 3.2.4 Aufwandsvergleich je nach Reduktionsstufe

Im Rahmen von großen dynamischen Optimierungsproblemen liegt der Aufwand pro SQP-Iteration hauptsächlich bei der Berechnung der Sensitivitäten. Jedoch müssen je nach Reduktionsstufe weniger Richtungsableitungen berechnet werden, was sich in kürzeren Laufzeiten widerspiegelt. Ein Vergleich der Anzahl der Richtungsableitungen (bzgl. der Lösung der Differentialgleichungen) pro SQP-Iteration gibt Aufschluß darüber, welche Laufzeiten zu erwarten sind:

$$\begin{aligned} \underbrace{n(n_x + n_z + n_u + n_p)}_{\text{MUSCOD-II, SQP}} &\rightarrow \underbrace{n(n_x + n_u + n_p + 1)}_{\text{MUSCOD-II, PRSQP}} \\ &\rightarrow \underbrace{\frac{1}{2}n(n+1)n_u + n n_p + n}_{\text{dieser Algorithmus}} \end{aligned}$$

mit folgenden Abkürzungen für die Dimensionen:  $n$  Anzahl der Mehrzielintervalle,  $n_x$  Anzahl der differentiellen Variablen,  $n_z$  Anzahl der algebraischen Variablen,  $n_u$  Anzahl der Steuerungen,  $n_p$  Anzahl der globalen Parameter. Dabei sind zwei Varianten von MUSCOD-II aufgeführt: die Vollraum-SQP Variante und die PRSQP-Variante, bei der nur die Konsistenzbedingungen in die Reduktion eingehen.

Insbesondere ist zu sehen, daß beim vorliegenden Algorithmus die Anzahl der Richtungsableitungen nun unabhängig von der Modelldimension ist, so daß bei Vergrößern der Zustandsdimension der Aufwand nun proportional mit der Integrationszeit ist!

### 3.3 Lokale Konvergenzuntersuchungen

Die hinreichenden Bedingungen zweiter Ordnung besagen, daß die reduzierte Lagrange-Hessematrix am Lösungspunkt positiv definit sein muß. Die verallgemeinerte Dennis-Moré Bedingung (2.16) charakterisiert die superlineare Konvergenz für den restringierten Fall. Jedoch wird eine Bedingung an die einseitig projizierten Lagrange-Hessematrix-Approximation gestellt. Ziel ist es, superlineare Konvergenz für SQP- und später auch RSQP- bzw. PRSQP-Verfahren zu zeigen, bei dem nur Bedingungen an die (partiell) reduzierte Lagrange-Hessematrix-Approximation gestellt werden. Das heißt, daß die Dennis-Moré Bedingung für die beidseitig projizierte Lagrange-Hessematrix-Approximation formuliert werden muß. Das Ergebnis ist eine 2-Schritt superlineare Konvergenz, wie sie von Powell (1978, [75]) gezeigt wurde. Wir betrachten dazu wieder das gleichungsbeschränkte Minimierungsproblem [EQ], welches wir mittels eines SQP-Verfahrens mit Hessematrix-Approximation  $B_k$  iterativ lösen (siehe 2.8). Satz 7 beschreibt die Berechnung des Schritts  $\Delta w_k$  bei einer Basistransformation mit  $[S_k^{\mathcal{R}} : S_k^{\mathcal{N}}]$ . Zusammen mit der Dennis-Moré Bedingung (2.16) folgt für  $P_k := I - S_k^{\mathcal{R}} \nabla G_k^T = S_k^{\mathcal{N}} V_k^T$ :

$$\begin{aligned} & \frac{\|S_k^{\mathcal{N}T}[B_k - \nabla_w^2 L(w^*, \lambda^*)]\Delta w_k\|}{\|\Delta w_k\|} \\ = & \frac{\|S_k^{\mathcal{N}T}[B_k - \nabla_w^2 L(w^*, \lambda^*)][P_k + (I - P_k)]\Delta w_k\|}{\|\Delta w_k\|} \\ \leq & \frac{\|S_k^{\mathcal{N}T}[B_k - \nabla_w^2 L(w^*, \lambda^*)]S_k^{\mathcal{N}}V_k^T\Delta w_k\|}{\|\Delta w_k\|} + \frac{\|S_k^{\mathcal{N}T}[B_k - \nabla_w^2 L(w^*, \lambda^*)]S_k^{\mathcal{R}}y_k^{\mathcal{R}}\|}{\|\Delta w_k\|}. \end{aligned} \quad (3.9)$$

Die Terme in (3.9) ergeben sich aus  $[I - P_k]S_k^{\mathcal{N}} := S_k^{\mathcal{R}}\nabla G_k^T S_k^{\mathcal{N}} = 0$  und  $[I - P_k]S_k^{\mathcal{R}} = S_k^{\mathcal{R}}$ , siehe Definition der Basistransformation. Wenn nun die Bedingung gilt:

$$\frac{\|S_k^{\mathcal{N}T}[B_k - \nabla_w^2 L(w^*, \lambda^*)]S_k^{\mathcal{N}}V_k^T\Delta w_k\|}{\|\Delta w_k\|} \rightarrow 0 \quad \text{für } k \rightarrow \infty, \quad (3.10)$$

dann liegt nach dem lokalen Kontraktionssatz (Satz 5) folgende Konvergenzrate vor:

$$\begin{aligned} \|w_{k+1} - w^*\| & \leq \left( \frac{\|\Delta w_k\|}{2} \omega + \kappa \right) \frac{\|\Delta w_k\|}{1 - \delta_k} \\ & \leq \frac{1}{1 - \delta_k} \left[ \frac{\omega}{2} \|\Delta w_k\|^2 \right. \\ & \quad + \left\| \left( \begin{array}{c} S_{k+1}^{\mathcal{N}T} B_{k+1} \\ \nabla G_{k+1}^T \end{array} \right)^{-1} \right\| \cdot \|S_k^{\mathcal{N}T}[\nabla_w^2 L(w^*, \lambda^*) - \nabla_w^2 L(w_k, \lambda_k)]\Delta w_k\| \\ & \quad + \left\| \left( \begin{array}{c} S_{k+1}^{\mathcal{N}T} B_{k+1} \\ \nabla G_{k+1}^T \end{array} \right)^{-1} \right\| \cdot \|S_k^{\mathcal{N}T}[B_k - \nabla_w^2 L(w^*, \lambda^*)]S_k^{\mathcal{N}}V_k^T\Delta w_k\| \\ & \quad \left. + \left\| \left( \begin{array}{c} S_{k+1}^{\mathcal{N}T} B_{k+1} \\ \nabla G_{k+1}^T \end{array} \right)^{-1} \right\| \cdot \|S_k^{\mathcal{N}T}[B_k - \nabla_w^2 L(w^*, \lambda^*)]S_k^{\mathcal{R}}y_k^{\mathcal{R}}\| \right] \\ & = O(\|\Delta w_k\|^2) + o(\|\Delta w_k\|) + o(\|\Delta w_k\|) + O(\|G_k\|) \\ & = O(\|\Delta w_k\|^2) + o(\|\Delta w_k\|) + o(\|\Delta w_k\|) + O(\|\Delta w_{k-1}\|^2) \\ & = o(\|\Delta w_{k-1}\|) = o(\|w_{k-1} - w^*\|) \end{aligned}$$

Diese 2-Schritt superlineare Konvergenz hat ihre Ursache in der schwächeren Bedingung (3.10) an die Hessematrix-Approximation gegenüber Bedingung (2.16).

Für RSQP-Verfahren bzgl. Problem [EQ] gilt dieses Ergebnis ebenfalls, wie man sich leicht überlegt: setze  $B_k := V_k B_k^N V_k^T$  und erhalte somit eine Äquivalenz des Systems (2.7) mit den Systemen (3.2) bzw. (3.3), da sie sich alle Systeme auf  $B_k^N y_k^N = -S_k^N \nabla F_k$  für die primalen Variablen zurückführen lassen. Der Crossterm verschwindet bei dieser Approximation der Lagrange-Hessematrix, da gilt:  $V_k^T S_k^R = 0$ . Um 1-Schritt superlineare Konvergenz bei RSQP-Verfahren zu erhalten und gleichzeitig nur die reduzierte Hessematrix zu approximieren, muß demnach für den Term mit der Bildraumkomponente  $y_k^R := -G_k$  gelten:

$$\left\| \left( \begin{array}{c} S_{k+1}^N{}^T B_{k+1} \\ \nabla G_{k+1}^T \end{array} \right)^{-1} \right\| \cdot \|S_k^{NT} [B_k - \nabla_w^2 L(w^*, \lambda^*)] S_k^R y_k^R\| = o(\|w_k - w^*\|).$$

Gabay (1982, [40]) zeigte, daß durch Hinzunahme eines vereinfachten Newton-Schritts auf die Mannigfaltigkeit der Gleichungsbedingungen, d. h.

$$\Delta w_k := \underbrace{S_k^N y_k^N + S_k^R y_k^R}_{\text{bisheriger Schritt}} - \underbrace{S_k^R G(w_k + S_k^N y_k^N + S_k^R y_k^R)}_{\text{vereinf. Newton-Schritt}}, \quad (3.11)$$

diese Bedingung erfüllt wird und damit 1-Schritt superlineare Konvergenz vorliegt.

Für RSQP-Verfahren mit BFGS-Updates und Gradientendifferenzen (3.6b) bzw. (3.6c) hat Kupfer (1992, [54]) zeigen können, daß die asymptotische Bedingung (3.10) an die reduzierte Lagrange-Hessematrix-Approximation  $B_k^N$  erfüllt ist. Dieses Ergebnis hat Schulz (1996, [83]) auf PRSQP-Verfahren übertragen, indem er BFGS-Updates der *teilweise* reduzierten Lagrange-Hessematrix-Updates

$$B_{1,k}^N \approx S_{1,k}^N{}^T \nabla_w^2 L(w_k, \lambda_k, \mu_k) S_{1,k}^N$$

mit den Gradientendifferenzen (3.6b) bzw. (3.6c) durchführt, wobei  $S_{1,k}^N$  eine Nullraum-Basis bzgl. der Jacobimatrix von  $G_1$  ist. Er konnte zeigen, daß  $B_k^N := S_{2,k}^N{}^T B_{1,k}^N S_{2,k}^N$  die asymptotische Bedingung (3.10) erfüllt, wobei das Update ausgelassen wird, wenn  $V_k \Delta w_k = 0$ . Damit konnte er 2-Schritt superlineare Konvergenz zeigen. Entsprechend lassen sich die Ergebnisse bzgl. der 1-Schritt superlinearen Konvergenz übertragen.

### 3.4 Globalisierungstechniken für reduzierte SQP-Algorithmen

In diesem Abschnitt werden die Ergebnisse für SQP-Verfahren aus Abschnitt 2.3.2 auf den PRSQP-Fall übertragen. Dabei spielt es eine Rolle, ob Lagrange-Multiplikatoren berechnet werden, da dann Informationen für ein Update der Strafparameter zur Verfügung stehen. Aus Gründen der Effizienz bei der Berechnung eines PRSQP-Schritts wird jedoch auf die Berechnung der Lagrange-Parameter  $\lambda_{1,k}$  für die Gleichungsbedingungen  $G_{1,k}$  verzichtet und die Straffunktion so modifiziert, daß nur ein Strafparameter  $\rho_k$  für die

Gleichungsbedingungen  $G_1$  angepaßt werden muß, vergleiche (2.17). Es wird gezeigt, daß Kompatibilität zwischen PRSQP-Schritt und Straffunktion gewährleistet werden kann. Diese Art von Globalisierung wurde zuerst von Biegler et al. (1995, [12] und 1996, [87]) bei RSQP-Verfahren eingeführt.

Natürliche Niveaufunktionen bei (verallgemeinerten) Gauß-Newton-Verfahren benötigen keine Informationen bzgl. der Lagrange-Parameter. Unter anderem konnte dadurch bisher keine globale Konvergenz bewiesen werden. In der Praxis sind jedoch bei LSQ-Problemen mit  $\omega < \infty$  und  $\kappa < 1$  und Globalisierung über Natürliche Niveaufunktionen keine Schwierigkeiten hinsichtlich der globalen Konvergenz aufgetreten. Im Unterschied zum nicht reduzierten Fall muß bei der Berechnung einer Schätzung der Krümmung  $\omega_3$ , siehe (2.26), zusätzlich zu den Funktionswerten eine Richtungsableitung berechnet werden.

### 3.4.1 Liniensuchmethoden mit Straffunktionen

Für die Globalisierung der Konvergenz betrachten wir die modifizierte  $l_1$ -Niveaufunktion:

$$\phi_{\rho,\sigma,\tau}(w) = F(w) + \rho \sum_{i \in \mathcal{E}_1} |G_{1,i}(w)| + \sum_{i \in \mathcal{E}_2} \sigma_i |G_{2,i}(w)| + \sum_{j \in \mathcal{I}} \tau_j H_j(w)^-, \quad (3.12)$$

bei der für die Gleichungsbedingungen  $G_1$  nur ein skalarer Strafparameter  $\rho$  verwendet wird. Dies ähnelt in gewisser Hinsicht der  $l_1$ -Niveaufunktion von Han (1977, [47]), die er im Rahmen von SQP-Verfahren eingesetzt hat. Wie wir im Satz 6 gesehen haben, ist die Kompatibilität zwischen Schritt und Straffunktion neben der Beschränktheit der Lagrange-Hessematrix die Voraussetzung für globale Konvergenz. Die Richtungsableitung der Straffunktion (3.12) am Punkt  $(w_k, \lambda_{1,k}, \lambda_{2,k}, \mu_k)$  in Richtung  $\Delta w_k$  hat folgende Gestalt:

$$\begin{aligned} D_{\Delta w_k} \phi_{\rho_k, \sigma_k, \tau_k}(w_k) = & -y_k^{\mathcal{N}T} B_k^{\mathcal{N}} y_k^{\mathcal{N}} - \rho_k \|G_{1,k}\|_1 - \lambda_{1,k}^T G_{1,k} \\ & - \sum_{i \in \check{\mathcal{E}}_k} (\sigma_{k,i} - \lambda_{k,2,i}) \nabla G_{2,i}(w_k)^T \Delta w_k \\ & - \sum_{i \in \hat{\mathcal{E}}_k} (\sigma_{k,i} + \lambda_{k,2,i}) (-\nabla G_{2,i}(w_k))^T \Delta w_k \\ & - \sum_{j \in \check{\mathcal{I}}_k} (\tau_{k,j} - \mu_{k,j}) \nabla H_j(w_k)^T \Delta w_k \\ & - \sum_{j \in \hat{\mathcal{I}}_k} \mu_{k,j} (-\nabla H_j(w_k))^T \Delta w_k. \end{aligned} \quad (3.13)$$

Abstieg in dieser Straffunktion kann nun gewährleistet werden, wenn folgende Bedingungen erfüllt sind:

$$\rho_k \|G_{1,k}\|_1 \geq |\lambda_{1,k}^T G_{1,k}|, \quad \sigma_{k,i} \geq |\lambda_{k,2,i}|, \quad \tau_{k,j} \geq \mu_{k,j} \quad (3.14)$$

für  $i \in \mathcal{E}_2$  und  $j \in \mathcal{I}$ . Die Bedingung für  $\rho_k$  ist schwächer als die Han-Bedingung  $\rho_k \geq \|\lambda_{1,k}\|_\infty$  und es müssen Vorkehrungen getroffen werden, damit die Exaktheitsbedingung  $\rho^* \geq \|\lambda_1^*\|$  an die Straffunktion am Lösungspunkt  $w^*$  erfüllt ist. Dazu wird der Verlauf der Werte  $\|G_{1,i}\|_1$  für  $i = k-1, \dots, k-t$  über die letzten  $t$  Schritte betrachtet. Der Strafparameter  $\rho_k$  wird nur dann vergrößert, wenn  $\|G_{1,k}\|_1 \geq \max_{i=k-1, \dots, k-t} \|G_{1,i}\|_1$ . Damit soll am Lösungspunkt  $w^*$  Vollschrirt gewährleistet werden, siehe auch Fletcher (1987, [37]). In der Praxis werden die Strafparameter  $\sigma_k$  und  $\tau_k$  über die Powell-Updatevorschrift (2.21) aktualisiert und  $\rho_k$  wie folgt bestimmt ( $\hat{\rho} > 0$ , fest):

$$\rho_k = \begin{cases} \rho_{k-1} & \text{if } \rho_{k-1} \|G_{1,k}\|_1 \geq |\tilde{\lambda}_{1,k}^T G_{1,k}| + \hat{\rho} \|G_{1,k}\|_1 \\ \frac{|\tilde{\lambda}_{1,k}^T G_{1,k}|}{\|G_{1,k}\|_1} + 2\hat{\rho} & \text{sonst} \end{cases}$$

Im PRSQP-Verfahren liegen die Lagrange-Multiplikatoren  $\lambda_{k,2}$  und  $\mu_k$  vor, da sie beim Lösen des [RQP] mitberechnet werden. Wichtig in diesem Zusammenhang ist, daß

$$\lambda_{1,k}^T G_{1,k} = -\nabla F_k S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}} + \lambda_{2,k}^T \nabla G_{2,k}^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}} + \mu_k^T \nabla H_k^T S_{1,k}^{\mathcal{R}} y_k^{\mathcal{R}}$$

billig berechnet werden kann. Die Formel ist motiviert durch die Stationaritätsbedingung der Lagrange-Funktion:  $\nabla_{w_1} L(w, \lambda_1, \lambda_2, \mu) = 0$ .

### 3.4.2 Liniensuchmethoden mit Natürlichen Niveaufunktionen

Um den restriktiven Monotonietest [RMT] durchzuführen, wird eine Schätzung der Krümmung  $\omega$  benötigt. Praktisch wird die Krümmung wie folgt berechnet, vergleiche (2.26):

$$\omega_3(\alpha) = 2 \frac{|\Delta \tilde{w}_k(\alpha) - (1 - \alpha) \Delta w_k|}{\alpha^2 |\Delta w_k|^2}$$

Dabei ist  $\Delta \tilde{w}_k(\alpha) := S_{1,k}^{\mathcal{R}} \tilde{y}_k^{\mathcal{R}}(\alpha) + S_{1,k}^{\mathcal{N}} \tilde{y}_k^{\mathcal{N}}(\alpha)$  mit der Bildraumkomponente  $\tilde{y}_k^{\mathcal{R}}(\alpha) := -G_1(w_k + \alpha \Delta w_k)$  und der Nullraumkomponente  $\tilde{y}_k^{\mathcal{N}}(\alpha)$  Lösung des folgenden LSQ-RQPs:

$$\begin{aligned} \min \quad & \|f(w_k + \alpha \Delta w_k) + \nabla f_k^T S_{1,k}^{\mathcal{R}} \tilde{y}_k^{\mathcal{R}} + \nabla f_k^T S_{1,k}^{\mathcal{N}} \tilde{y}_k^{\mathcal{N}}\|_2^2 \\ \text{unter} \quad & G_2(w_k + \alpha \Delta w_k) + \nabla G_{2,k}^T S_{1,k}^{\mathcal{R}} \tilde{y}_k^{\mathcal{R}} + \nabla G_{2,k}^T S_{1,k}^{\mathcal{N}} \tilde{y}_k^{\mathcal{N}} = 0 \\ & H(w_k + \alpha \Delta w_k) + \nabla H_k^T S_{1,k}^{\mathcal{R}} \tilde{y}_k^{\mathcal{R}} + \nabla H_k^T S_{1,k}^{\mathcal{N}} \tilde{y}_k^{\mathcal{N}} \geq 0. \end{aligned}$$

Zu beachten ist, daß bei diesem LSQ-RQP das gleiche Active-Set verwendet wird wie bei der Berechnung von  $y_k^{\mathcal{N}}$  in  $\Delta w_k$ .

Da  $S_{1,k}^{\mathcal{R}}$  nicht explizit berechnet wird, sondern nur das Produkt mit einem Richtungsvektor, muß also zusätzlich zur Funktionsauswertung eine Richtungsableitung berechnet werden. Das heißt im vorliegenden Fall eine zusätzliche Integration, da in  $G_1$  auch die Anfangsbedingungen und Stetigkeitsbedingungen enthalten sind. Bei dieser Integration kann jedoch die Informationen über die Nominaltrajektorie wiederverwendet werden. Es sei betont, daß die Differenz  $\Delta \tilde{w}_k(\alpha) - (1 - \alpha) \Delta w_k$  auch direkt berechnet werden kann, was Stabilitätsvorteile in der Berechnung von  $\omega_3$  bringt.

## 3.5 Algorithmische Zusammenfassung der reduzierten direkten Mehrfach-Schießverfahren

Im nachfolgenden Algorithmus werden die Teilschritte, welche zur Bestimmung einer neuen Iterierten notwendig sind, zusammengefaßt. Dabei wurden in  $G_1$  sowohl die Anfangs-, Stetigkeits- und Stufenübergangsbedingungen als auch die Konsistenzbedingungen gruppiert, in  $G_2$  die Gleichungsrestriktionen der Mehrpunktrandbedingungen und in  $H$  die diskretisierten Kontroll- und Pfadbedingungen sowie die Ungleichungsrestriktionen der Mehrpunktrandbedingungen zusammengefaßt. Eine Partitionierung der NLP-Variablen  $w = (w_1, w_2)$  wurde entsprechend der Bedingungen in  $G_1$  wie folgt vorgenommen:  $w_1 := (s_0^x, s_0^z, \dots, s_m^x, s_m^z)$ ,  $w_2 := (u_0, u_1, \dots, u_m, p)$ . Damit ist die Koordinaten-Basis

bzgl.  $G_1$  wohldefiniert. Im Falle eines Parameterschätzproblems mit Least-Squares-Termen im Mayer-Zielfunktional (Bezeichnung LSQ) entspricht der Algorithmus einem partiell reduziertem VGN-Verfahren und beim Optimal Steuerungsproblem mit allgemeinem Mayer-Zielfunktional (Bezeichnung OPT) einem partiell reduziertem SQP-Verfahren.

### Ein reduzierter direkter Mehrzielalgorithmus:

- (0) Initialisiere Mehrzielgitter (siehe Abschnitt 1.5), Parameterisierung der Steuerung (siehe Abschnitt 1.3.1), Startwerte für Variablen, setze  $k = 0$
- (1) Berechne Funktionswerte des diskretisierten und parameterisierten Problems (1.12):  $F_k$  (1.12a),  $G_{1,k}$  (1.12b,1.12c,1.12e),  $G_{2,k}$  (1.12f),  $H_k$  (1.12d,1.12f) an der Stelle  $w_k$
- (2) Berechne für Koordinaten-Basis  $[S_k^{\mathcal{R}}, S_k^{\mathcal{N}}] := [\nabla G_{1,k}, (0, I)^T]^{-T}$  den Schritt in der Bildraumkomponente  $\Delta w_k^{\mathcal{R}} := -S_k^{\mathcal{R}} G_{1,k}$  und die Nullraumkoordinaten-Basis  $S_k^{\mathcal{N}}$  (siehe Bemerkung 13, Abschnitt 3.2.1 und 4.3)
- (3) Berechne reduzierte Gradienten der Zielfunktion  $\nabla F_k^T S_k^{\mathcal{N}}$  und der Restriktionen  $\nabla G_{2,k}^T S_k^{\mathcal{N}}$ ,  $\nabla H_k^T S_k^{\mathcal{N}}$  sowie  $\nabla G_{2,k}^T \Delta w_k^{\mathcal{R}}$ ,  $\nabla H_k^T \Delta w_k^{\mathcal{R}}$
- (4) Berechne reduzierte Hessematrix  $B_k^{\mathcal{N}}$  (und Crossterm  $C_k$ ):

LSQ: Berechne  $B_k^{\mathcal{N}} := S_k^{\mathcal{N},T} \nabla f_k^T \nabla f_k S_k^{\mathcal{N}}$  und  $C_k := -G_{1,k}^T S_k^{\mathcal{R},T} \nabla f_k^T \nabla f_k S_k^{\mathcal{N}}$  (siehe Abschnitt 3.2.3)

OPT:  $k = 0$ : Initialisiere  $B_k^{\mathcal{N}}$  als skalierte Einheitsmatrix,  
 $k > 0$ : Berechne reduzierten Lagrangegradientendifferenz  $\gamma_{k-1}^{\mathcal{N}}$ ,  
 Schritt  $\delta_{k-1}^{\mathcal{N}} := \alpha_{k-1} y_{k-1}^{\mathcal{N}}$  und führe damit BFGS-Update durch:  
 $B_k^{\mathcal{N}} := U_{BFGS}(B_{k-1}^{\mathcal{N}}, \delta_{k-1}^{\mathcal{N}}, \gamma_{k-1}^{\mathcal{N}})$ , setze  $C_k = 0$  (siehe Abschnitt 3.2.2)

- (5) Löse das reduzierte Quadratische Programm in  $y_k^{\mathcal{N}}$  mit Schranken  $\Omega_k$  (siehe auch Bemerkung 6 für den LSQ-Fall):

$$\begin{aligned} \min_{y_k^{\mathcal{N}} \in \Omega_k} \quad & (\nabla F_k^T S_k^{\mathcal{N}} + C_k) y_k^{\mathcal{N}} + \frac{1}{2} y_k^{\mathcal{N},T} B_k^{\mathcal{N}} y_k^{\mathcal{N}} \\ \text{unter} \quad & G_{2,k} + \nabla G_{2,k}^T S_k^{\mathcal{R}} y_k^{\mathcal{R}} + \nabla G_{2,k}^T S_k^{\mathcal{N}} y_k^{\mathcal{N}} = 0 \\ & H_k + \nabla H_k^T S_k^{\mathcal{R}} y_k^{\mathcal{R}} + \nabla H_k^T S_k^{\mathcal{N}} y_k^{\mathcal{N}} \geq 0. \end{aligned}$$

- (6) Berechne Schritt  $\Delta w_k^{\mathcal{N}} := S_k^{\mathcal{N}} y_k^{\mathcal{N}}$  und evtl. reduzierten Lagrangegradienten  $S^{\mathcal{N},T} \nabla w L$  an der Stelle  $(w_k + \Delta w_k^{\mathcal{N}}, \lambda_{k+1}, \mu_{k+1})$  wie in Schritt (1-3) für Hessematrix-Update in Schritt (4)
- (7) Bestimme Schrittweitenparameter  $\alpha_k \in (0, 1]$  durch Liniensuche:
  - LSQ: mit Natürlichen Niveaufunktionen und Restriktivem Monotonietest (siehe Abschnitte 2.3.2 und 3.4.2)
  - OPT: mit der  $l_1$ -Niveaufunktion und partiell-multiplikatorfreien Update-Strategie (siehe Abschnitte 2.3.2 und 3.4.1)
- (8) Berechne neue Iteration  $w_{k+1} := w_k + \alpha_k \Delta w_k^{\mathcal{N}} + \alpha_k \Delta w_k^{\mathcal{R}}$
- (9)  $k = k + 1$ , gehe zu (1) bis Konvergenz erreicht

# Kapitel 4

## Berechnung von Ableitungen

In diesem Kapitel geht es um die effiziente Berechnung von Ableitungen, die im PRSQP-Algorithmus benötigt werden. Wir unterscheiden zwischen zwei Typen: einerseits Richtungsableitungen bzgl. den Gleichungsrestriktionen  $G_1$ , wie sie in der Berechnung der Koordinaten-Basis  $S_{1,k}^{\mathcal{N}}$  und der Bildraumkomponente  $S_{1,k}^{\mathcal{R}}y_k^{\mathcal{R}}$  vorkommen. Andererseits Richtungsableitungen bzgl. Zielfunktion  $F$  und übrigen Restriktionen  $G_2, H$  mit Richtung  $S_{1,k}^{\mathcal{R}}y_k^{\mathcal{R}}$ .

Wie wir in Abschnitt 3.2.1 gesehen haben, kann die Koordinaten-Basis rekursiv berechnet werden. Hierbei kommen Richtungsableitungen der Stetigkeits- und Konsistenzbedingungen vor. Diese Ableitungen werden bei der hier verwendeten Mehrziel-Methode durch Integration der Richtungs-Variationsdifferentialgleichung (Richtungs-VDAE) bestimmt. Dabei wird das Prinzip der *Internen Numerischen Differentiation (IND)* verwendet, siehe Bock (1981, [14]). Die Einhaltung dieses Prinzips gewährleistet, daß zur Lösung der Richtungs-VDAE die adaptiven Komponenten des Integrators eingefroren werden und damit das Diskretisierungsschema zur Erzeugung der Nominaltrajektorie selbst abgeleitet wird. Es sei betont, daß die adaptiven Komponenten nicht abgeleitet werden. Dieses Prinzip ermöglicht daher auch den Einsatz von adaptiven Schrittweiten- und Ordnungssteuerungen im Integrator, was Voraussetzung für effiziente Algorithmen ist. Diese Anforderungen sind in dem BDF-Integrator DAESOL umgesetzt, siehe Eich (1987, [33]), Bauer (1994, [6] und 1999, [7]). Verwendung findet hier der sog. iterative Ansatz, bei dem das Newton-Schema für die Berechnung einer neuen Iterierten der Nominaltrajektorie differenziert wird und der somit ein iteratives Schema für die Richtungs-VDAE ergibt. Dabei wurde der iterative Ansatz so gewählt, daß Richtungsableitungen der Modell-Funktionen bzgl. Zuständen, Steuerungen und Parametern kombiniert und als *eine* Richtungsableitung ausgewertet wird. Zusätzlich wird die Iterationsmatrix aus der Berechnung der Nominaltrajektorie in dem iterativen Schema für die Richtungs-VDAE weiter verwendet. Für die Aufstellung der Iterationsmatrix des BDF-Schemas ist die effiziente Auswertung von Modell-Jacobimatrizen der DAE wichtig. Diese Jacobimatrizen sind dünnbesetzt und werden effizient im Vorwärtsmodus der *Automatischen Differentiation* mit Adol-C (Griewank et al. 1996, [44]) und Seed-Matrix-Komprimierung nach Curtis, Powell, Reid (1974, [25]) berechnet. Im Falle von gPROMS-Modellen (Barton und Pantelides 1994, [5]) werden die dünnbesetzten Modell-Jacobimatrizen über *Symbolische Differentiation* berechnet und über ein ESO-Interface (siehe Braunschweig et al. 2000, [23]) angesprochen.

Um das reduzierte Quadratische Programm [RQP] aufzustellen, werden noch Richtungsableitungen bzgl. Zielfunktion  $F$  und übrigen Restriktionen  $G_2, H$  mit Richtungen  $[S_{1,k}^N : S_{1,k}^R, y_k^R]$  benötigt. Diese werden über finite Differenzen berechnet, da deren Genauigkeit bei den in dieser Arbeit vorgestellten Problemen ausreichend ist.

## 4.1 Interne Numerische Differentiation

Ausgangspunkt für die Berechnung der Richtungsableitungen der Stetigkeits- und Konsistenzbedingungen ist folgende quasilinear-implizite DAE:

$$\left. \begin{aligned} B(\tau, x, z, q_i) \frac{dx}{d\tau} &= f(\tau, x, z, q_i) \\ 0 &= g(\tau, x, z, q_i) - \alpha_i(\tau)g(\tau_i, s_i^x, s_i^z, q_i) \end{aligned} \right\}, \quad \tau \in [\tau_i, \tau_{i+1}] \quad (4.1)$$

mit Anfangswerten  $x(\tau_i) = s_i^x$  sowie  $z(\tau_i) = s_i^z$  und den Parametern  $q_i \equiv (u_i, p)$ . Im folgenden werden die Methoden des BDF-Integrators DAESOL dargestellt, die effizient die in den Rekursionen vorkommenden Matrix-Vektor-Produkte der verallgemeinerten Sensitivitäten berechnen (siehe 3.2.1).

Prinzipiell gibt es verschiedene Ansätze, wie Sensitivitäten, d. h. die Ableitungen der Zustände am Ende des Zeithorizonts nach Anfangswerten und Parametern, berechnet werden können. Eine Möglichkeit ist die Berechnung von Nominaltrajektorie und variierten Trajektorien mit gestörten Anfangswerten bzw. Parametern im Rahmen von finiten Differenzen mit Störungsinkrement  $\eta$ . Wenn dabei die adaptiven Komponenten des Integrator-Schemas *nicht* eingefroren werden, hängen die Endzustände nicht differenzierbar von den Anfangswerten und Parametern ab. Dieser Ansatz wird auch *Externe Numerische Differentiation (END)* genannt, da die Differentiation außerhalb der Diskretisierung der Nominaltrajektorie vorgenommen wird. Die Genauigkeit der Sensitivitäten ist auf  $O(\sqrt{TOL})$ ,  $TOL$  die Integrationsgenauigkeit, beschränkt für  $\eta^2 = TOL$ . Im Gegensatz dazu wird der Ansatz, durch die Diskretisierung adaptiv erzeugte Folge von Abbildungen direkt abzuleiten, *Interne Numerische Differentiation (IND)* genannt. Das bedeutet, daß alle adaptiv erzeugten Komponenten bei der Berechnung der variierten Trajektorien und damit der Sensitivitäten eingefroren werden. Die Genauigkeit der Sensitivitäten setzt sich zusammen aus dem Diskretisierungsfehler des Integrators  $O(TOL)$ ,  $TOL$  die Integrationsgenauigkeit, und dem Fehler durch die Approximation durch finite Differenzen  $O(\sqrt{EPS})$  für  $\eta^2 := EPS$  (Maschinengenauigkeit), siehe Bock (1981, [14]).

Im Rahmen von BDF-Integratoren konzentrieren wir uns auf die Berechnung der Sensitivitäten durch die *Variationsdifferentialgleichung (VDAE)*. Das IND-Prinzip ist jetzt erfüllt, wenn nach dem Satz über implizite Funktionen (SIF) die Ableitung der Diskretisierung der Nominaltrajektorie nach Anfangswerten und Parametern berechnet wird. Diese Ableitung ist identisch mit der BDF-Diskretisierung der VDAE (siehe Bauer 1999, [7]). Es gibt nun zwei unterschiedliche Implementierungen, als direkte bzw. iterative Variante bezeichnet, um Lösungen der VDAE zu erhalten. Wir definieren dazu die Richtungs-

Wronski-Matrizen  $\mathcal{W}$  bzgl. Richtungen  $\mathcal{D}$ :

$$\mathcal{W} := \begin{pmatrix} \mathcal{W}^x \\ \mathcal{W}^z \\ \mathcal{W}^q \end{pmatrix} := \begin{pmatrix} \frac{\partial x}{\partial s_i^x} & \frac{\partial x}{\partial s_i^z} & \frac{\partial x}{\partial q_i} \\ \frac{\partial z}{\partial s_i^x} & \frac{\partial z}{\partial s_i^z} & \frac{\partial z}{\partial q_i} \\ 0 & 0 & I \end{pmatrix} \mathcal{D}$$

und führen die Richtungs-VDAE ein:

$$\left. \begin{aligned} B(\cdot) \frac{d}{d\tau} \mathcal{W}^x &= \left( -\frac{\partial B}{\partial x} \dot{x} + \frac{\partial f}{\partial x} & -\frac{\partial B}{\partial z} \dot{x} + \frac{\partial f}{\partial z} & -\frac{\partial B}{\partial q} \dot{x} + \frac{\partial f}{\partial q} \right) \mathcal{W} \\ 0 &= \left( \frac{\partial g}{\partial x} & \frac{\partial g}{\partial z} & \frac{\partial g}{\partial q} \right) \mathcal{W} - \alpha_i(\tau) \begin{pmatrix} G_i^x & G_i^z & G_i^q \end{pmatrix} \mathcal{D} \end{aligned} \right\}, \quad \tau \in [\tau_i, \tau_{i+1}] \quad (4.2)$$

mit den Anfangswerten  $\mathcal{W}^x|_{\tau=\tau_i} = (I \ 0 \ 0) \mathcal{D}$ ,  $\mathcal{W}^z|_{\tau=\tau_i} = (0 \ I \ 0) \mathcal{D}$ . Die weitere Darstellung der o.g. Varianten bezieht sich auf eine BDF-Diskretisierung der DAE (4.1) mit  $k$  Vergangenheitswerten und aktueller Schrittweite  $h$ . Um eine neue Iterierte  $(x_{n+1}, z_{n+1})$  der Nominaltrajektorie zum Zeitpunkt  $t_{n+1} \in [\tau_i, \tau_{i+1}]$  mit Anfangswerten  $(s_i^x, s_i^z)$  und Parametern  $q_i$  zu erhalten, muß nun folgende nichtlineare Gleichung in  $(x_{n+1}, z_{n+1})$  gelöst werden:

$$\begin{aligned} &F(x_{n+1}, z_{n+1}, q_i) \\ &:= \begin{pmatrix} B(t_{n+1}, x_{n+1}, z_{n+1}, q_i) \sum_{i=0}^k \beta_i x_{n+1-i} + hf(t_{n+1}, x_{n+1}, z_{n+1}, q_i) \\ g(t_{n+1}, x_{n+1}, z_{n+1}, q_i) - \alpha_i(t_{n+1})g(\tau_i, s_i^x, s_i^z, q_i) \end{pmatrix} = 0 \end{aligned} \quad (4.3)$$

#### 4.1.1 Direkte Berechnung einer Lösung der Richtungs-VDAE

Wendet man die BDF-Diskretisierung der Nominaltrajektorie auf die Richtungs-VDAE (4.2) an, dann erhält man folgendes lineare System

$$\begin{aligned} &\underbrace{\begin{pmatrix} \beta_0 B + h \left( \frac{\partial f}{\partial x} - \frac{\partial B}{\partial x} \dot{x} \right) & h \left( \frac{\partial f}{\partial z} - \frac{\partial B}{\partial z} \dot{x} \right) \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial z} \end{pmatrix}}_{=:J} \begin{pmatrix} \mathcal{W}_{n+1}^x \\ \mathcal{W}_{n+1}^z \end{pmatrix} \\ &= - \underbrace{\begin{pmatrix} h \left( \frac{\partial f}{\partial q} + \frac{\partial B}{\partial q} \dot{x} \right) \\ \frac{\partial g}{\partial q} \end{pmatrix}}_{=:J^q} \mathcal{W}_{n+1}^q + \underbrace{\begin{pmatrix} h \sum_{i=1}^k \beta_i \mathcal{W}_{n+1-i}^x \\ \alpha_i(t_{n+1}) \begin{pmatrix} G_i^x & G_i^z & G_i^q \end{pmatrix} \mathcal{D} \end{pmatrix}}_{=:c} \end{aligned} \quad (4.4)$$

in den Richtungs-Wronski-Matrizen  $(\mathcal{W}_{n+1}^x, \mathcal{W}_{n+1}^z)^T$ . Nach dem Satz über implizite Funktionen ist eine Lösung  $\mathcal{W}_{n+1}$  des linearen Systems (4.4) ausgewertet an einer Lösung der Nominaltrajektorie (4.3) auch Lösung der Richtungs-VDAE (4.2) am Punkt  $t_{n+1}$ . Der zusätzliche Aufwand (gegenüber der Berechnung der Nominaltrajektorie) bei dieser Variante liegt in der einmaligen Auswertung der Matrizen  $J$  und  $J^q$  sowie der Zerlegung von  $J$  pro BDF-Schritt. Diese Variante empfiehlt sich, wenn die Matrix  $J$  billig zu berechnen ist!

### 4.1.2 Iterative Berechnung einer Lösung der Richtungs-VDAE

Zur Erläuterung dieser Variante wird das vereinfachte Newton-Verfahren zur Lösung der nichtlinearen Gleichung (4.3) dargestellt:

$$\begin{pmatrix} x_{n+1}^{m+1} \\ z_{n+1}^{m+1} \end{pmatrix} = \begin{pmatrix} x_{n+1}^m \\ z_{n+1}^m \end{pmatrix} - \tilde{J}^{-1} F(x_{n+1}^m, z_{n+1}^m, q_i) \quad (4.5)$$

mit einer Approximation  $\tilde{J}$  der exakten Iterationsmatrix  $\partial F/\partial(x_{n+1}, z_{n+1})$  und maximal 3 Iterationen ( $m \leq 3$ ). Nun können die Newton-Gleichungen nach Richtungen  $\mathcal{D}$  direkt abgeleitet werden, und wir erhalten mit Hilfe der Kettenregel:

$$\begin{pmatrix} \mathcal{W}_{n+1}^{x, m+1} \\ \mathcal{W}_{n+1}^{z, m+1} \end{pmatrix} = \begin{pmatrix} \mathcal{W}_{n+1}^{x, m} \\ \mathcal{W}_{n+1}^{z, m} \end{pmatrix} - \tilde{J}^{-1} \underbrace{((J \ J^q) \mathcal{W}_{n+1}^m - c)}_{\text{Richtungsabl.}} \quad (4.6)$$

eine Iterationsvorschrift für die Richtungs-Wronski-Matrizen  $\mathcal{W}_{n+1}^{x,z}$  mit Bezeichnungen  $J, J^q$  und  $c$  wie bei (4.4). Der zusätzliche Aufwand bei dieser Variante liegt in der Berechnung der Richtungsableitungen (Anzahl der Richtungen mal Anzahl von Newton-Iterationen pro BDF-Schritt), wobei die Iterationsmatrix  $\tilde{J}$  sowie deren Zerlegung weiterverwendet werden kann. Diese Variante empfiehlt sich also genau dann, wenn wenige Richtungen  $\mathcal{D}$  vorliegen.

## 4.2 Berechnung der Modell-Jacobimatrizen

Bei der Berechnung der Richtungs-VDAE ist es notwendig, die Iterationsmatrix  $J$  bzw.  $\tilde{J}$  effizient zu berechnen und zu zerlegen. In der Iterationsmatrix kommen die Ableitung der Modellfunktionen nach den differentiellen und algebraischen Variablen vor, welche meistens dünnbesetzt sind. Zur Zerlegung der Iterationsmatrix werden daher spezielle LA-Löser verwendet, wie z. B. MA48, [78]. Nachfolgend werden zwei in der Praxis verwendete Methoden zur Auswertung der Jacobimatrizen der Modellfunktionen vorgestellt. Die erste Variante operiert auf Software-Code der Modellfunktionen und wird auch *automatische Differentiation* bzw. *algorithmische Differentiation (AD)* genannt (siehe Griewank 2000, [43]). Als Stellvertreter sei hier Adol-C (siehe Griewank et al. 1996, [44]) genannt, der Ableitungen für C bzw. C++ Quellcode erzeugt. Die zweite Variante operiert auf Formeln und wird *Symbolische Differentiation (SD)* genannt. Sie findet Anwendung bei Modellen, die in der Modellierungsumgebung gPROMS implementiert sind (siehe Barton und Panteledes 1994, [5]). Beide Varianten beruhen auf systematischer Anwendung der Kettenregel, wobei diese auf aktuelle numerische Werte (AD) bzw. auf Formeln (SD) angewendet wird. In beiden Fällen werden die Ableitungen bis auf Maschinengenauigkeit genau berechnet!

### 4.2.1 C/Adol-C: automatische Differentiation

Die Idee bei der Automatischen Differentiation ist, Quellcode einer Funktion  $f$  zu analysieren und in eine Folge elementarer Operationen zu untergliedern (Erstellen eines Ausführungsgraphen), deren Auswertung Zwischenergebnisse erzeugen. Die Ableitung der

Ausgangsvariablen (Endergebnisse)  $y := F(x)$  nach Eingangsvariablen  $x$  kann auf zwei unterschiedliche Varianten erzeugt werden. Die erste Variante berechnet sukzessive die Ableitung der Zwischenergebnisse nach den Eingangsvariablen und wird auch *Vorwärtsmodus* (*VM*) genannt. Der Aufwand dazu ist unabhängig von der Anzahl der Ausgangsvariablen:

$$[\text{VM}] \quad \text{Aufwand}\{f'\} \leq (1 + 1.5n)\text{Aufwand}\{f\},$$

wobei  $n$  die Dimension der Eingangsvariablen ist. Der umgekehrte Weg, nämlich sukzessive die Ableitung der Ausgangsvariablen nach den Zwischenergebnisse zu berechnen, wird auch *Rückwärtsmodus* (*RM*) genannt und hat eine Komplexitätsschranke, die unabhängig von der Anzahl der Eingangsvariablen ist:

$$[\text{RM}] \quad \text{Aufwand}\{f'\} \leq (1.5 + 2.5m)\text{Aufwand}\{f\},$$

wobei  $m$  die Dimension der Ausgangsvariablen. Beiden gemeinsam ist, daß die Kettenregel auf die numerischen Zwischenergebnisse angewendet wird und nicht auf die Formeln! In Adol-C (siehe Griewank et al. 1996, [44]) werden dabei die Variablen im Quellcode überladen und ein Tape angelegt, das die numerischen Werte der partiellen Ableitungen bzgl. der Zwischenergebnisse aufnimmt. Dies erfordert nur geringen Änderungsaufwand im Quellcode und kein weiterer Code wird erzeugt. So wird bei der Berechnung von Ableitungsinformationen auch die Funktion ausgewertet.

Wenn die Jacobimatrizen dünnbesetzt sind, dann kann die Komplexitätsschranke für die Auswertung der Jacobimatrix noch reduziert werden, indem  $n$  bzw.  $m$  ersetzt werden durch die maximale Anzahl der Nichtnulleinträge in den Zeilen bzw. Spalten von  $f'$ . Vorausgesetzt wird, daß die Eigenschaft *Path Connectedness* gilt: Die Zwischenergebnisse hängen von mindestens einer Eingangsvariable ab und gehen in mindestens eine Ausgangsvariable ein. Im Vorwärtsmodus kann die Komplexitätsschranke dadurch motiviert werden, daß nur solche Zwischenergebnisse (neu) berechnet werden müssen, die von den Eingangsvariablen in einer nichttrivialen Art abhängen. Im Rückwärtsmodus müssen nur solche Ausgangsvariablen (neu) berechnet werden, die von Zwischenergebnissen in einer nichttrivialen Art abhängen.

Die Jacobimatrix-Kompression erlaubt das Ausnutzen der Dünnbesetztheit durch Berechnen des Produkts aus Jacobimatrix mit einer Seed-Matrix (Vorwärtsmodus: Multiplikation von rechts, Rückwärtsmodus: Multiplikation von links). Voraussetzung für den praktischen Einsatz ist, daß daraus die Jacobimatrix wieder rekonstruiert werden kann. Wesentlich dazu ist die Bestimmung strukturell orthogonaler Spalten bzw. Zeilen. Die optimale Berechnung dieser ist jedoch NP-schwer. Idee ist die Zusammenfassung strukturell orthogonaler Spalten bzw. Zeilen in Gruppen. Für jede Gruppe wird ein Richtungsvektor in der Seed-Matrix bestimmt. Curtis, Powell und Reid (1974, [25]) bestimmen die Richtungsvektoren in der Seed-Matrix so, daß ein Richtungsvektor bzgl. einer Gruppe in der  $j$ -ten Zeile bzw. Spalte den Eintrag 1 hat, wenn die  $j$ -te Spalte bzw. Zeile der Jacobimatrix in der Gruppe vorhanden ist, ansonsten den Eintrag 0 hat. Newsam und Ramsdell (1983, [66]) wählen die Seed-Matrix so, daß die komprimierte Jacobimatrix effizient und hinreichend genau berechnet wird. Vorgeschlagen wurden Vandermonde Matrizen, deren

Kondition jedoch exponentiell wächst mit der Anzahl der Spalten bzw. Zeilen der Gruppen. Eine Alternative sind orthogonale Polynome, bei denen jedoch die transponierten Systeme nicht billig ausgewertet werden können. Diese und weitere Techniken zur Bestimmung von Seed-Matrizen im Kontext von Optimal-Steuerungsalgorithmen sind in Brandt-Pollmann (2004, [22]) zu finden.

## 4.2.2 gPROMS: symbolische Differentiation

In der Modellierungsumgebung gPROMS (siehe Barton und Pantelides 1994, [5]) werden Modelle in einer eigenen Modellierungssprache implementiert. Die Modellfunktionen und deren dünnbesetzte Jacobimatrizen sind über eine *Equation-Set-Object*-Schnittstelle in anderen Programmiersprachen verfügbar (siehe Braunschweig et al. 2000, [23]). Intern beruht die Ableitungsgenerierung auf einer symbolischen Differentiation der Modellfunktionen (siehe Pantelides 1988, [70]). Der internen Darstellung der Modellfunktionen und Ableitungen liegt eine binäre Baumstruktur zugrunde, d. h. sie beruht auf einem azyklisch gerichteten Graphen, bei dem jeder Knoten entweder zwei oder keinen Nachkommen hat. Der Vorteil dieser Darstellung ist, daß die Operator-Reihenfolge explizit definiert ist durch die gerichteten Zweige im Baum, und Teilausdrücke im Modell durch Teilbäume repräsentiert werden, die auch bei mehrfachem Auftreten im Modell nur einmal berechnet und gespeichert werden müssen. Weiterhin birgt die binäre Baumstruktur ein hohes Maß an Rekursivität, welche eine effiziente Implementierung ermöglicht. Dies gilt ebenfalls für die Ableitungsberechnung, wobei gemeinsame Teilausdrücke von Modellfunktion und Ableitung nur einmal berechnet werden müssen. Aus Effizienzgründen können sowohl Gleichungen als auch Variablen, nach denen abgeleitet wird, gruppiert und deren Jacobimatrizen bzgl. gruppierter Variablen getrennt berechnet werden.

## 4.3 Anwendung auf Koordinaten-Basis

In Abschnitt 3.2.1 sind Rekursionen angegeben, um die Nullraum-Koordinaten-Basis zu berechnen. Im folgenden werden die Richtungen  $\mathcal{D}$  für die Richtungs-VDAE (4.2) im  $i$ -ten

Mehrzielintervall spezifiziert, woraus sich die Koordinaten-Basis ergibt:

$$\mathcal{D} := \begin{pmatrix} I & 0 & 0 \\ -G_i^{z^{-1}} G_i^x & -G_i^{z^{-1}} G_i^{u,p} & -G_i^{z^{-1}} g_i \\ 0 & I & 0 \end{pmatrix}$$

$$\cdot \begin{pmatrix} [(\prod_{k=i-1}^1 \tilde{X}_k^x)(\tilde{X}_0^u - \tilde{X}_0^x \tilde{D}_0^{x^{-1}} \tilde{D}_0^u)]^T & 0 & 0 \\ [(\prod_{k=i-1}^2 \tilde{X}_k^x) \tilde{X}_1^u]^T & 0 & 0 \\ \vdots & 0 & 0 \\ [\tilde{X}_{i-1}^u]^T & 0 & 0 \\ 0 & [I \ 0] & 0 \\ [(\prod_{k=i-1}^1 \tilde{X}_k^x)(\tilde{X}_0^p - \tilde{X}_0^x \tilde{D}_0^{x^{-1}} \tilde{D}_0^p) \\ + \sum_{j=1}^{i-1} (\prod_{k=i-1}^{j+1} \tilde{X}_k^x) \tilde{X}_j^p]^T & [0 \ I] & 0 \\ [(\prod_{k=i-1}^0 \tilde{X}_k^x) \tilde{D}_0^{x^{-1}} \tilde{d}_0 \\ + \sum_{j=1}^{i-1} (\prod_{k=i-1}^{j+1} \tilde{X}_k^x) \tilde{c}_j]^T & 0 & 1 \end{pmatrix}^T.$$

Dabei wurden folgende Abkürzungen verwendet (siehe auch Abschnitt 3.2.1):

$$\begin{aligned} \tilde{X}_i^{x,u,p} &:= \begin{pmatrix} X_i^x - X_i^z G_i^{z^{-1}} G_i^x & X_i^u - X_i^z G_i^{z^{-1}} G_i^u & X_i^p - X_i^z G_i^{z^{-1}} G_i^p \end{pmatrix}, \\ \tilde{D}_0^{x,u,p} &:= \begin{pmatrix} D_0^x - D_0^z G_0^{z^{-1}} G_0^x & -D_0^z G_0^{z^{-1}} G_0^u & D_0^p - D_0^z G_0^{z^{-1}} G_0^p \end{pmatrix}, \\ \tilde{d}_0 &:= d_0 - D_0^z G_0^{z^{-1}} g_0, \\ \tilde{c}_i &:= c_0 - X_i^z G_i^{z^{-1}} g_i. \end{aligned}$$

Die zweite Matrix in  $\mathcal{D}$  sind die eigentlichen akkumulierten Richtungen in der Reihenfolge:  $u_0, u_1, \dots, u_i, p$  sowie der Bildraumkomponente.

## Kapitel 5

# Parameterschätzung bei einem Zahnfleisch-Implantat

Die kontrollierte Freisetzung von Wirkstoffen in einem medizinischen Umfeld hat vielfältige Anwendungsgebiete. Zum Beispiel wird Parodontose durch Freisetzung von antimikrobiellen Wirkstoffen aus einem Zahnfleisch-Implantat behandelt (siehe Abbildung 5.1). Ein



Abbildung 5.1: Zahnfleisch-Implantat

solches Implantat stellt der PerioChip<sup>©</sup> der Firma Dexcel Pharma, Israel, dar, welches eine kleine hydrolysierte Gelatine-Membran ist, die mit 2,5 mg Chlorhexidin D-Gluconate vernetzt ist. In den ersten 24 Stunden wird 40% des Chlorhexidins in die entzündete Zahnfleischtasche abgegeben. Während den folgenden 7 Tagen folgt eine langsame und kontinuierliche Abgabe. Tzafriri *et al.* (2000, [93] und 1999, [95]) entwickelten ein Reaktions-Diffusions-Modell für die *in-vitro*-Freisetzung des Wirkstoffs aus dem PerioChip in einer gepufferten Enzym-Lösung. Dabei wurden drei wesentliche Mechanismen berücksichtigt: Diffusion, reversible elektrostatische Bindung des Wirkstoffs auf der Gelatine-Membran und Enzym-katalysierter Abbau der Membran. Eine Parameterschätzung wurde für die instationäre 1D-PDE-Variante (siehe Abschnitt 5.1) erstmalig durch Tzafriri und Diehl (2000, [94]) durchgeführt. Hier dient dieses Beispiel als Benchmark-Problem: durch Wahl der Anzahl von Diskretisierungsintervallen entsteht durch Ortsdiskretisierung mit finiten Differenzen (Linienmethode) eine skalierbare Anfangswertaufgabe bestehend aus einer gewöhnlichen Differentialgleichung und nichtlinearen Anfangsbedingungen (in den Parametern). Im Parameterschätzproblem (siehe Abschnitt 5.2) sind der Wirkstoff-Diffusions-

Koeffizient, ein Michaelis-Menten-Parameter sowie Wirkstoff-Gelatine-Bindungsparameter die einzigen Freiheitsgrade. Dieses kann durch reduzierte Gauß-Newton-Verfahren ausgenutzt werden - ein Vergleich zu einem Vollraum-Verfahren wird in Abschnitt 5.4 gegeben. Der iterative und direkte IND-Ansatz zur Berechnung der Sensitivitäten wird im Abschnitt 5.3 verglichen. Weiterhin findet ein Vergleich der Modellimplementationen auf Basis von gPROMS und C/Adol-C in Abschnitt 5.5 statt.

## 5.1 Modell

Für die Beschreibung der Prozesse im PerioChip wird im folgenden ein instationäres 1D-Modell dargestellt, mit dem unter anderen eine schnelle diffusionskontrollierte Freisetzung des Wirkstoffs, gefolgt von einer langsameren konstanten Abgabe durch Membranabbau, beschrieben werden kann. Weitere Szenarien bzgl. der Freisetzungsrates sind ebenfalls mit diesem Modell möglich (siehe Tzafriri *et al.* 2000, [94]). Es wird vorausgesetzt, daß die Membran in der Enzym-Lösung bereits aufgequollen ist, das heißt, daß sie sich im Gleichgewicht befindet. Daraus ergibt sich auch die Dicke  $L$  der Membran im Gleichgewicht (siehe auch Tzafriri 2000, [93]). Weiterhin werden für die anfänglichen Konzentrationen des Wirkstoffs und Gelatine Durchschnittswerte angenommen. Damit ergibt sich folgendes instationäres 1D-PDE-Modell (für eine Übersicht der verwendeten Konstanten siehe Tabelle A.1 im Anhang A.1):

- Konzentration  $C$  des freigesetzten Wirkstoffs bestimmt durch eine Diffusionsgleichung mit Quellterm bzgl. der Freisetzung des gebundenen Wirkstoffs durch Membranabbau:

$$\frac{\partial C}{\partial t} = D \frac{\partial^2 C}{\partial x^2} - \rho B_{max} \frac{\partial Q}{\partial t} - Q B_{max} \frac{\partial \rho}{\partial t}, \quad (5.1)$$

- Abbau des Membran-Polymers, beschrieben durch Michaelis-Menten-Gleichung (modelliert über Gelatine-Konzentration  $\rho$ ):

$$\frac{\partial \rho}{\partial t} = - \frac{k_{cat} E_0 \rho}{K_M + \rho (1 + \beta B_{max} Q)}, \quad (5.2)$$

- zeitlicher Verlauf des Verhältnisses  $Q$  von gebundenem Wirkstoff zu Gelatine-Konzentration, beschrieben durch eine Reaktionsgleichung:

$$\frac{\partial Q}{\partial t} = k_a (1 - Q) C - k_d Q, \quad (5.3)$$

- örtliche Randbedingung für die Konzentration:

$$\frac{\partial C}{\partial x} = 0, \quad x = 0, \quad (5.4a)$$

$$C = 0, \quad x = L. \quad (5.4b)$$

- Anfangsbedingungen zum Start der Freisetzung nach Aufquellphase ( $t = 0$ ) unter der Bedingung  $C_0 + \rho_0 B_{max} Q(0) = C_{load}$ :

$$\rho = \rho_0, \quad (5.5a)$$

$$Q = \frac{K C_0}{1 + K C_0}, \quad (5.5b)$$

$$C = C_0. \quad (5.5c)$$

Man beachte, daß die Anfangsbedingung für  $Q$  aus einer Stationaritätsbedingung für  $Q$  hergeleitet wird. Durch Approximation der örtlichen Ableitung in der Diffusionsgleichung (5.1) durch zentrale finite Differenzen der Ordnung 2 mit  $N$  Diskretisierungspunkten und in den Randbedingungen (5.4) durch zentrale Differenzen der Ordnung 1 wird das obige Anfangs-Randwertproblem in ein gewöhnliches Anfangswertproblem (ODE) überführt. Dabei hängen die Anfangsbedingungen ebenfalls von dem (zu schätzenden) Parameter  $B_{max}$  ab. Die differentiellen Variablen sind wie folgt zusammengefasst:  $x := (C_1, \dots, C_N, Q_1, \dots, Q_N, \rho_1, \dots, \rho_N)$  mit den Abkürzungen  $C_i := C|_{\frac{i-1}{N-1}L}$ ,  $Q_i := Q|_{\frac{i-1}{N-1}L}$ ,  $\rho_i := \rho|_{\frac{i-1}{N-1}L}$ .

## 5.2 Aufsetzen der Parameterschätzung und Ergebnis

Zur Anpassung des Modells aus Abschnitt 5.1 an die in-vitro-Freisetzung des PerioChip in einer gepufferten Enzym-Lösung aus bakterieller Collagenase (*Closteridium hystolicum*) unter kontrollierten Bedingungen  $pH = 7.5$  und  $37^\circ C$  wurde die Freisetzungsrates  $M$  gemessen, um die Parameter  $p := (D, B_{max}, k_{cat})$  zu schätzen:

$$M(t) = 1 - \frac{\int_0^L (C + \rho B_{max} Q) dx}{L C_{load}} \quad (5.6)$$

Im Parameterschätzproblem wird das Integral über die Dicke der Membran durch die Trapezregel mit  $N$  äquidistanten Stützstellen approximiert. Es lagen 11 Messwerte  $M_k$  zusammen mit Standardabweichungen  $\sigma_k$  zu den Zeitpunkten  $t_k$  im Zeithorizont  $[0, 192]$  Stunden vor. Diese waren aufgrund der zu erwartenden zweiphasigen Freisetzung am Anfang gehäuft und später in Abständen von 24 Stunden verteilt (siehe Tabelle A.2). Wir nehmen an, daß die Fehler der Meßwerte mit bekannter Varianz, Erwartungswert 0 und unabhängig sind. Der Zeithorizont wurde mit 11 Mehrzielintervallen diskretisiert, wobei die Mehrzielknoten auf die Zeitpunkte der Messwerte gelegt wurden. Die Minimierung der folgenden LSQ-Zielfunktion liefert dann eine Maximum-Likelihood-Schätzung:

$$\sum_{k=1}^K \frac{||M(t_k) - M_k||^2}{\sigma_k^2}. \quad (5.7)$$

Für  $N = 41$  Diskretisierungspunkte sind die optimalen Parameterwerte für die Anpassung an die Meßwerte in Tabelle 5.1 angegeben. Deutlich zu erkennen ist, daß die zugehörigen Standardabweichungen moderat sind im Vergleich zum Parameterwert. In Abbildung 5.2 ist die gefittete Freisetzungsrates mit den optimalen Parameterwerten gezeigt. Zu sehen

Parameter	Startwerte	Optimale Werte
$D [cm^2/s]$	2.36e-5	$1.65e-4 \pm 0.332e-4$
$B_{max} [mol/mol]$	2.05	$4.14 \pm 0.0931$
$k_{cat} [h^{-1}]$	1.65e+3	$1.79e+3 \pm 0.0734e+3$

Tabelle 5.1: Übersicht der Parameterwerte

ist die zweiphasige Freisetzung des Wirkstoffs: der erste Abschnitt (3 Stunden) ist bestimmt durch Diffusion des nichtgebundenen Wirkstoffs. Dann geht die Kurve in einen linearen Verlauf über, bestimmt durch die lineare Abbau-Kinetik (siehe auch Tzafriri *et al.* 2000, [93]). Ziel der nachfolgenden Vergleiche ist es, anhand dieses Parameterschätz-

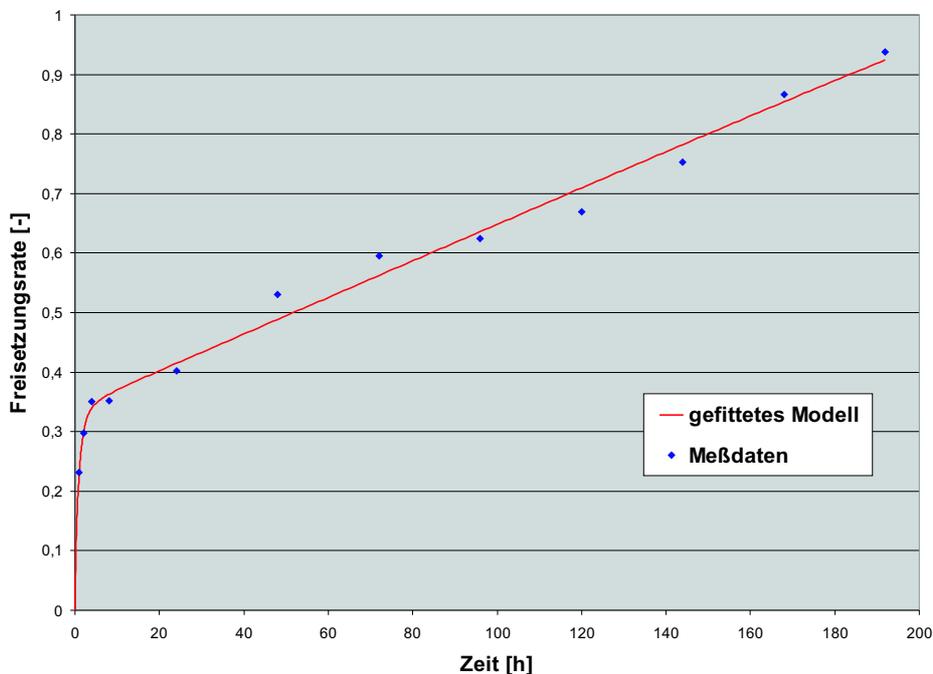


Abbildung 5.2: Gefittete Freisetzungsrates des PerioChip-Modells

problems aufzuzeigen, wo der zeitliche Aufwand in den verschiedenen reduzierten und Vollraum-Gauß-Newton-Verfahren steckt und welche Möglichkeiten existieren, diesen zu verringern. Dabei wurde das Modell mit der Anzahl  $N := \{41, 81, 161, 321, 641\}$  von Diskretisierungspunkten skaliert, so daß die Anzahl der Differentialgleichungen  $n_x := 3 \cdot N = \{123, 243, 483, 963, 1923\}$  variiert. Die entsprechenden Probleme wurden durch Vollschritt-Verfahren gelöst mit einer Integrationsgenauigkeit  $1.E-5$  und Abbruchgenauigkeit  $1.E-3$  ( $l_2$ -Norm des Schritts). Alle Rechnungen wurden auf einer Dual-Xeon-Maschine mit jeweils 2.8 GHz, 2 GB RAM sowie Linux als Betriebssystem mit Kernel 2.4.7 und gcc-Suite Version 2.96 (mit Optimierung `-O3`) durchgeführt. Die Compilerversion 2.96 (aus

RedHat 7.2) war notwendig, da die aktuelle gPROMS-Version damit kompiliert wurde und so inkompatibel zur aktuellen gcc-Suite ist. Für die Berechnung der Modellableitungen kamen gPROMS (Version 2.3.0) und Adol-C (Version 1.8.7) zum Einsatz.

### 5.3 Vergleich der IND-Ansätze zur Berechnung der Sensitivitäten

In diesem Abschnitt werden die in Kapitel 4 vorgestellten IND-Ansätze (*direkte* und *iterative* Variante in DAESOL) zur Berechnung von Sensitivitäten im vollreduzierten Gauß-Newton-Verfahren aus Kapitel 3 miteinander verglichen. Die Modellableitungen werden wahlweise mit finiten Differenzen oder automatischer Differentiation (Adol-C) berechnet, wobei im letzteren Fall wahlweise auch eine Seedmatrix-Komprimierung erfolgt. Richtungsableitungen im iterativen Ansatz werden ebenfalls mit finiten Differenzen oder mit automatischer Differentiation (Adol-C) berechnet. Die Behandlung der LA-Systeme mit den dünnbesetzten Modellableitungen erfolgte in DAESOL mittels dem LA-Löser MA48, siehe [78]. Folgende Varianten wurden anhand der obigen skalierten Parameterschätzprobleme miteinander verglichen:

- [RVGN + DIR + FD]: Standard-Variante mit vollreduziertem Gauß-Newton-Verfahren, direkte Berechnung der VDAE, Modellableitungen über finite Differenzen,
- [RVGN + DIR + AD]: vollreduziertes Gauß-Newton-Verfahren, direkte Berechnung der VDAE, Modellableitungen über automatische Differentiation (Adol-C),
- [RVGN + DIR + AD + CPR]: vollreduziertes Gauß-Newton-Verfahren, direkte Berechnung der VDAE, Modellableitungen über automatische Differentiation (Adol-C) mit Seedmatrix-Komprimierung nach Curtis-Powell-Reid,
- [RVGN + IT + R-AD + AD]: vollreduziertes Gauß-Newton-Verfahren, iterative Berechnung der VDAE, Richtungsableitungen und Modellableitungen über automatische Differentiation (Adol-C),
- [RVGN + IT + R-AD + AD + CPR]: vollreduziertes Gauß-Newton-Verfahren, iterative Berechnung der VDAE, Richtungsableitungen über automatische Differentiation (Adol-C), Modellableitungen über automatische Differentiation (Adol-C) mit Seedmatrix-Komprimierung nach Curtis-Powell-Reid,
- [RVGN + IT + R-FD + AD + CPR]: vollreduziertes Gauß-Newton-Verfahren, iterative Berechnung der VDAE, Richtungsableitungen über finite Differenzen, Modellableitungen über automatische Differentiation (Adol-C) mit Seedmatrix-Komprimierung nach Curtis-Powell-Reid.

Neben der CPU-Zeit zum Lösen der Parameterschätzprobleme sind die Anzahl der Aufrufe der Modellgleichungen  $f$  und die Anzahl der Modellableitungen  $f_x$  nach den differentiellen Variablen sowie die Anzahl der Richtungsableitungen  $Df \cdot w$  der Modellgleichungen nach den Richtungen  $w$  von Interesse.

Zunächst betrachten wir den direkten IND-Ansatz. In Tabellen 5.2 und 5.3 sind die Statistiken für die Fälle angegeben, bei denen die Modellableitungen über finite Differenzen bzw. über automatische Differentiation mit Adol-C berechnet wurden. Die große Anzahl

# Knoten	CPU-Zeit [sec]	# $f$ -Aufrufe	# Iter
641	1599.84	13450903	9
321	420.93	6715836	9
161	113.16	3382506	9
81	34.44	1732106	9
41	9.75	820801	8

Tabelle 5.2: [RVGN + DIR + FD]: Statistik für reduziertes Gauß-Newton-Verfahren, direkter Ansatz für IND und finite Differenzen für Modellableitungen

# Knoten	CPU-Zeit [sec]	# $f_x$ -Aufrufe	# $f$ -Aufrufe	# Iter
641	4094.62	6917	22019	9
321	1035.06	6814	21860	9
161	280.77	6678	21377	9
81	81.78	6501	20878	9
41	23.26	5604	17873	8

Tabelle 5.3: [RVGN + DIR + AD]: Statistik für reduziertes Gauß-Newton-Verfahren, direkter Ansatz für IND und automatische Differentiation für Modellableitungen

von Modellaufrufen im Finite-Differenzen-Fall kommt durch besagte Approximation der Modellableitungen nach den differentiellen Variablen und Parametern zustande. Anhand der Abbildung 5.3 ist deutlich zu sehen, daß der Gesamtaufwand für die Optimierung quadratisch mit der Anzahl der Diskretisierungspunkte wächst, wobei die Berechnung der Modellableitungen über finite Differenzen billiger ist. Bei diesen Varianten nimmt der Anteil zur Berechnung der Modellableitung  $f_x$  nach den differentiellen Variablen deutlich zu und dominiert schließlich die gesamte Rechenzeit, siehe Abbildung 5.4. Das heißt, daß Strategien zur Vermeidung und/oder Beschleunigung dieser Modellableitungen von Interesse sind. In unserem Fall entspricht die Anzahl der  $f_x$ -Aufrufe auch gleichzeitig der Anzahl der Integrationsschritte, da die Iterationsmatrix des BDF-Verfahrens in jedem Schritt für die Lösung der VDAE berechnet wird. Dies kann jedoch in der iterativen Variante zur Lösung der VDAE vermieden werden, wobei die Monitorstrategie in DAE-SOL benutzt wird. Der Tabelle 5.4 ist für diesen Fall zu entnehmen, daß die Anzahl der  $f_x$ -Modellableitungen deutlich reduziert wurde (sie entspricht jetzt der Anzahl der Auswertung der Iterationsmatrix für die reine Simulation). Neu hinzugekommen sind die Richtungsableitungen der Modellgleichungen  $Df \cdot w$ . Sie wurden im vorliegenden Fall über automatische Differentiation berechnet. Deutlich zu sehen ist die drastische Reduktion der Rechenzeit.

Der andere Weg, die Rechenzeit zu verringern, ist die Beschleunigung der Berechnung der

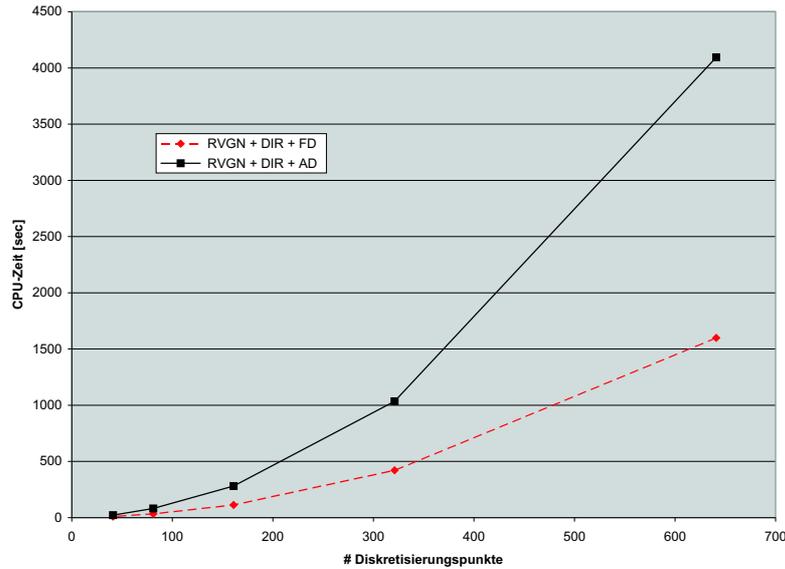


Abbildung 5.3: Rechenzeit des reduzierten Gauß-Newton-Verfahren mit direktem IND-Ansatz, Modellableitungen über finite Differenzen und automatische Differentiation

# Knoten	CPU-Zeit [sec]	# $f_x$ -Aufrufe	# $f$ -Aufrufe	# $Df \cdot w$ -Aufrufe	# Iter
641	795.95	546	77624	58508	9
321	203.87	506	75555	57036	9
161	80.23	522	73807	55648	9
81	37.2	505	71079	53576	9
41	15.73	414	60349	45712	8

Tabelle 5.4: [RVGN + IT + R-AD + AD]: Statistik für reduziertes Gauß-Newton-Verfahren, iterativer Ansatz für IND, automatische Differentiation für Modellableitungen und Richtungsableitungen

Modellableitungen. Wie im Abschnitt 4.2.1 beschrieben, stellt die Seed-Matrix-Komprimierung eine Möglichkeit dar, die automatische Differentiation bei dünnbesetzten Modellableitungen zu beschleunigen. Im vorliegenden Fall kam die Curtis-Powell-Reid-Technik zum Einsatz, so daß die Anzahl der Spalten der Seedmatrix soweit reduziert werden kann, daß diese unabhängig von der Anzahl der Diskretisierungspunkte des PDE-Modells ist. Für  $N = 41$  Diskretisierungspunkte wird dies anhand der Modellableitung  $f_x$  in Abbildung 5.5 verdeutlicht. In Tabelle 5.5 und 5.6 sind die Ergebnisse beim Einsatz dieser Technik zur Beschleunigung der Berechnung von  $f_x$  zusammengefaßt. In Abbildung 5.6 sind die Rechenzeiten der iterativen und direkten Variante mit und ohne Seeding dargestellt. Dabei wird eine Reduktion der Rechenzeit um den Faktor 1.6 bzw. 21 beim iterativen Ansatz und direktem Ansatz beobachtet. In Abbildung 5.7 ist der prozentuale Anteil der Berechnung von  $f_x$  an der jeweiligen gesamten Rechenzeit zu sehen. Der direkte Ansatz profitiert natürlich mehr von der Seedmatrix-Komprimierung, da hier die Anzahl

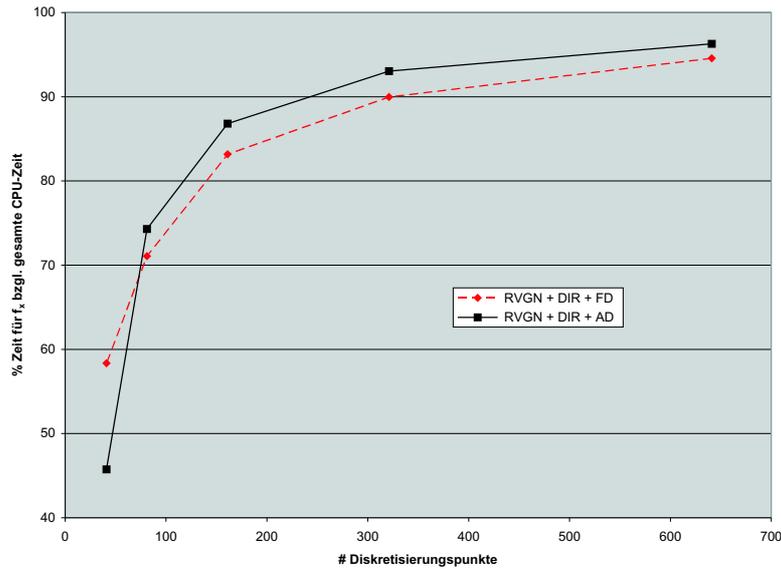


Abbildung 5.4: Prozentualer Aufwand für  $f_x$  bezogen auf gesamte Rechenzeit der reduzierten Gauß-Newton-Verfahren, Teil I

# Knoten	CPU-Zeit [sec]	QP-Zeit [sec]	# $f_x$ -Aufrufe	# $f$ -Aufrufe	# Iter
641	194.99	5.18	6917	22019	9
321	92.48	1.00	6814	21860	9
161	46.16	0.27	6678	21377	9
81	25.97	0.09	6501	20878	9
41	13.63	0.01	5604	17873	8

Tabelle 5.5: [RVGN + DIR + AD + CPR]: Statistik für reduziertes Gauß-Newton-Verfahren, direkter Ansatz für IND und automatische Differentiation für Modellableitungen mit CPR-Seeding

der  $f_x$ -Auswertungen höher ist. Ein Vergleich der iterativen mit der direkten Variante anhand der Anzahl der  $f_x$ -Aufrufe zeigt, daß die direkte Variante durchaus günstiger ist, da durch die Seedmatrix-Komprimierung die Anzahl der Richtungsableitungen in der direkten Variante kleiner ist als in der iterativen Variante. Dies liegt daran, daß für eine Auswertung der Modellableitung  $f_x$  nur 5 Richtungsableitungen in Adol-C notwendig sind. Zusammen mit der Anzahl der Integrationsschritte ergibt dies eine Zahl von 34585 Richtungsableitungen im Fall von 641 Diskretisierungspunkten. Bei der iterativen Variante liegen allein schon 58508 Richtungsableitungen  $Df \cdot w$  vor.

Um das Ergebnis beim direkten Ansatz mit finiten Differenzen und automatischer Differentiation wieder aufzugreifen, wurde die Berechnung der Richtungsableitungen  $Df \cdot w$  nun über finite Differenzen vorgenommen. Die Ergebnisse des iterativen Ansatzes mit Modellableitungen über automatische Differentiation mit CPR-Seeding sind in Tabelle 5.7

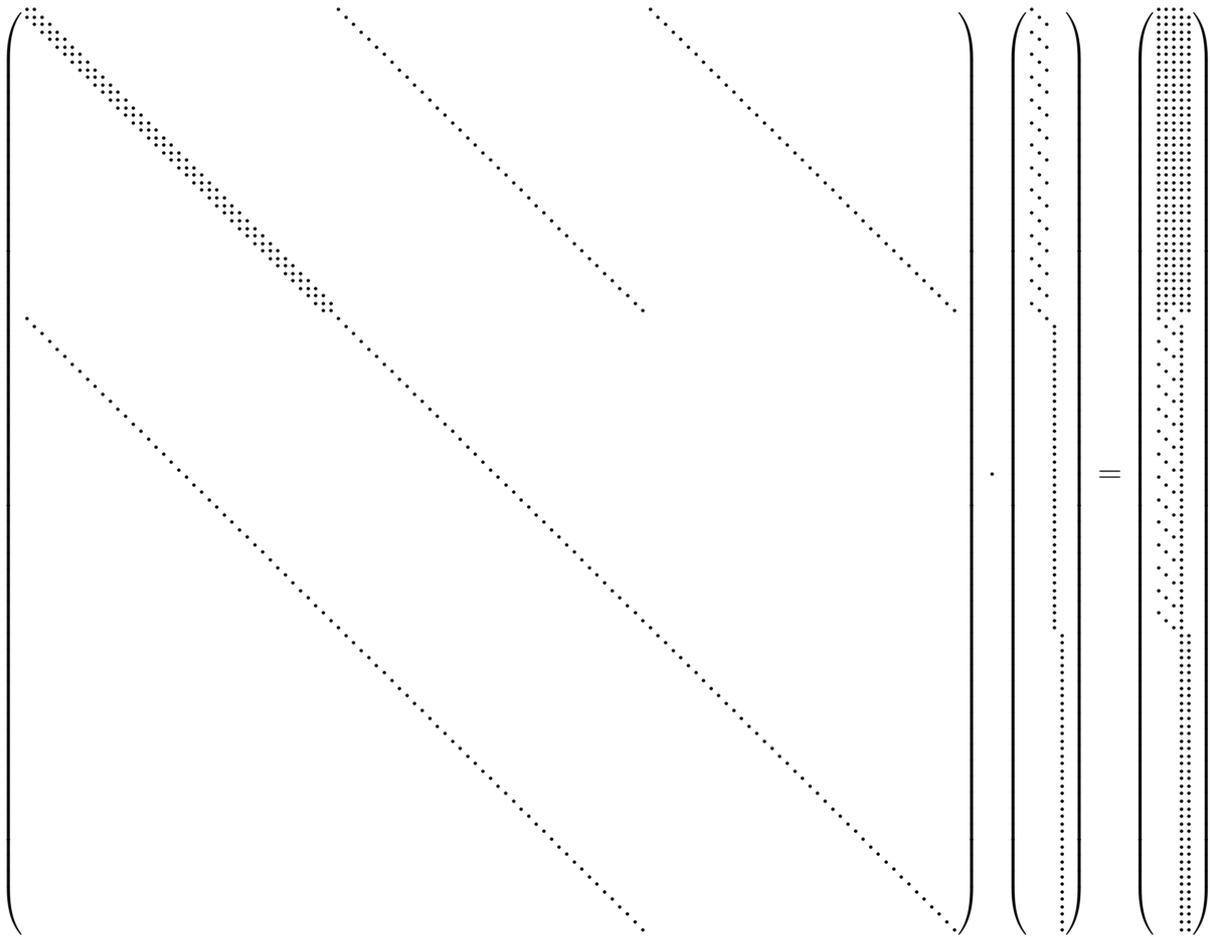


Abbildung 5.5: Struktur der Jacobimatrix  $f_x$  des PerioChip-Modells bzgl. differentieller Variablen: unkomprimierte Jacobimatrix  $\cdot$  CPR-Seedmatrix = komprimierte Jacobimatrix

dargestellt. Bei dieser Variante ist eine Reduktion der Rechenzeit um den Faktor 7 im Vergleich zur Berechnung der Richtungsableitungen über automatische Differentiation zu beobachten - siehe Abbildung 5.8. Damit ist das reduzierte Gauß-Newton-Verfahren mit iterativer Lösung der VDAE, Modellableitungen über automatische Differentiation und Richtungsableitungen über finite Differenzen die in diesem Abschnitt vorgestellte schnellste Variante! Es liegt eine Reduktion der Rechenzeit der skalierten Parameterschätzprobleme um den Faktor 23 vor, verglichen mit der Standard-Variante (direkte Berechnung der VDAE mit finiten Differenzen für die Modellableitungen).

# Knoten	CPU-Zeit [sec]	# $f_x$ -Aufrufe	# $f$ -Aufrufe	# $Df \cdot w$ -Aufrufe	# Iter
641	484.55	546	77624	58508	9
321	147.77	506	75555	57036	9
161	67.75	522	73807	55648	9
81	31.64	505	71079	53576	9
41	14.79	414	60349	45712	8

Tabelle 5.6: [RVGN + IT + R-AD + AD + CPR]: Statistik für reduziertes Gauß-Newton-Verfahren, iterativer Ansatz für IND, automatische Differentiation für Modellableitungen und Richtungsableitungen mit CPR-Seeding

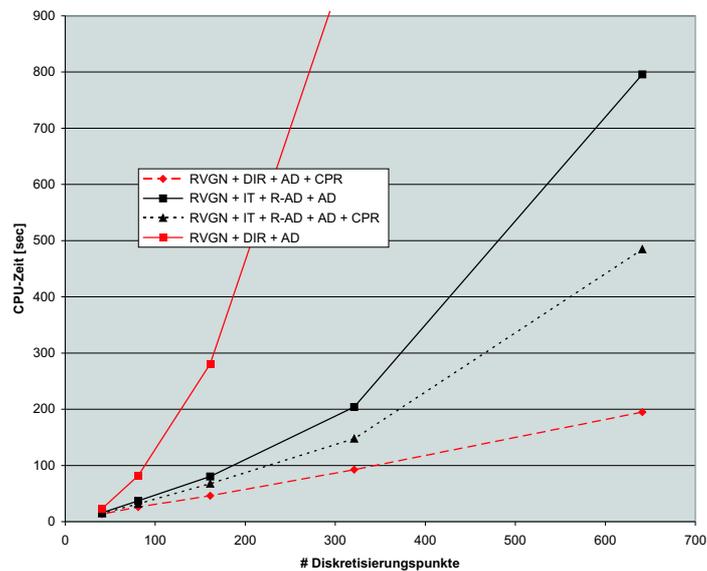


Abbildung 5.6: Rechenzeiten der reduzierten Gauß-Newton-Verfahren mit Reduktion des Aufwands für die Iterationsmatrix und/oder Reduktion der Anzahl der Iterationsmatrizen

# Knoten	CPU-Zeit [sec]	# $f_x$ -Aufrufe	# $f$ -Aufrufe	# $Df \cdot w$ -Aufrufe	# Iter
641	70.3	521	80551	60712	9
321	32.82	521	76148	57432	9
161	13.74	521	73651	55548	9
81	6.54	461	68904	52008	8
41	2.95	393	61891	46896	8

Tabelle 5.7: [RVGN + IT + R-FD + AD + CPR]: Statistik für reduziertes Gauß-Newton-Verfahren, iterativer Ansatz für IND, automatische Differentiation für Modellableitungen mit CPR-Seeding und Richtungsableitungen über finite Differenzen

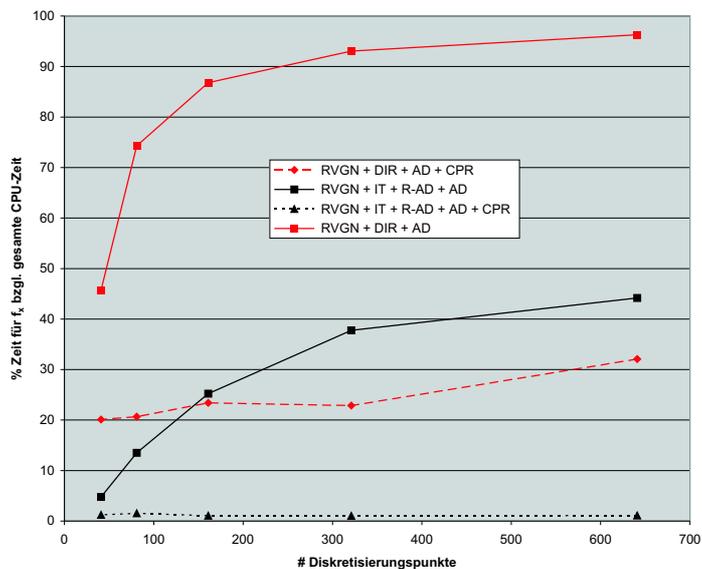


Abbildung 5.7: Prozentualer Aufwand für  $f_x$  bezogen auf gesamte Rechenzeit der reduzierten Gauß-Newton-Verfahren, Teil II

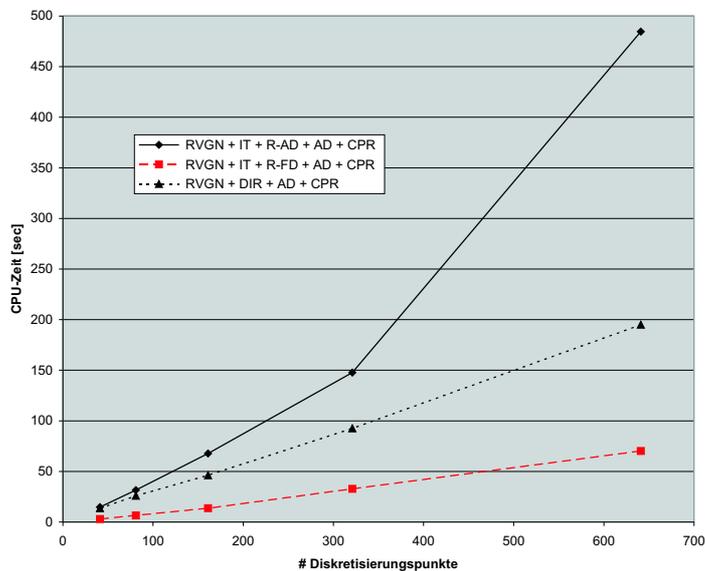


Abbildung 5.8: Rechenzeiten der reduzierten Gauß-Newton-Verfahren mit CPR-Seedling

## 5.4 Vergleich zum Vollraum-Ansatz

In diesem Abschnitt vergleichen wir den *Reduktionsansatz* aus Kapitel 3 mit einem *Vollraum-Ansatz* im Gauß-Newton-Verfahren, wobei die Diskretisierung der ODE identisch war (11 Mehrzielintervalle). Eine entscheidende Rolle spielt dabei die Anzahl der Richtungsableitungen und die Dimension und Art der LA-Systeme, die im Integrator und im Kondensieren beim Vollraum-Ansatzes zu lösen sind. Folgende Varianten wurden miteinander verglichen:

- [RVGN + DIR + AD + CPR]: vollreduziertes Gauß-Newton-Verfahren, direkte Berechnung der VDAE, Modellableitungen über automatische Differentiation (Adol-C) mit Seedmatrix-Komprimierung nach Curtis-Powell-Reid,
- [RVGN + IT + R-AD + AD + CPR]: vollreduziertes Gauß-Newton-Verfahren, iterative Berechnung der VDAE, Richtungsableitungen und Modellableitungen über automatische Differentiation (Adol-C) mit Seedmatrix-Komprimierung nach Curtis-Powell-Reid,
- [VGN + DIR + AD + CPR]: Gauß-Newton-Verfahren, direkte Berechnung der VDAE, Modellableitungen über automatische Differentiation (Adol-C) mit Seedmatrix-Komprimierung nach Curtis-Powell-Reid,
- [VGN + IT + R-AD + AD + CPR]: Gauß-Newton-Verfahren, iterative Berechnung der VDAE, Richtungsableitungen und Modellableitungen über automatische Differentiation (Adol-C) mit Seedmatrix-Komprimierung nach Curtis-Powell-Reid.

In Tabelle 5.8 und 5.9 sind die Statistiken für die Vollraum-Varianten aufgeführt. Dabei ist aufgrund der Anzahl von Richtungsableitungen der direkte Ansatz wesentlich schneller als der iterative Ansatz. Interessant ist, daß die Zeit für das Kondensieren und Lösen des QPs bis zu 50 % bei diesem Anwendungsproblem ausmacht. Dies liegt daran, daß die Sensitivitäten auf den einzelnen Mehrzielknoten dichtbesetzte Matrizen sind und dementsprechend viel Zeit beim Kondensieren benötigt wird. Die Matrizen in den Rekursionen passen schließlich auch nicht mehr in den Level-2-Cache (512 KB), wodurch diese aus dem relativ langsamen Hauptspeicher nachgeladen werden müssen. In der Abbildung 5.9 sind die Rechenzeiten der oben genannten Varianten gezeigt. Dabei dominieren bei der iterativen Variante die Auswertung der Richtungsableitungen  $Df \cdot w$ , deren Anzahl beim Vollraum-Verfahren mit der Ortsdiskretisierung  $N$  skaliert und beim vollreduzierten Verfahren konstant ist.

# Knoten	CPU-Zeit [sec]	Kondensier- und QP-Zeit [sec]	# $f_x$ Aufrufe	# $f$ Aufrufe	# Iter
321	4210.49	2129.22	6816	21882	9
161	809.62	275.21	6676	21363	9
81	179.39	18.4	6499	20864	9
41	50.93	1.86	5614	17929	8

Tabelle 5.8: [VGN + DIR + AD + CPR]: Statistik für Gauß-Newton-Verfahren, direkter Ansatz für IND und automatische Differentiation für Modellableitungen mit CPR-Seeding

# Knoten	CPU-Zeit [sec]	# $f_x$ -Aufrufe	# $f$ -Aufrufe	# $Df \cdot w$ -Aufrufe	# Iter
321	32823.7	508	13790789	13772262	9
161	8348.11	583	7330041	7310412	11
81	1741.73	513	3328680	3311160	9
41	411.11	412	1454568	1439928	8

Tabelle 5.9: [VGN + IT + R-AD + AD + CPR]: Statistik für Gauß-Newton-Verfahren, iterativer Ansatz für IND, automatische Differentiation für Modellableitungen und Richtungsableitungen mit CPR-Seeding

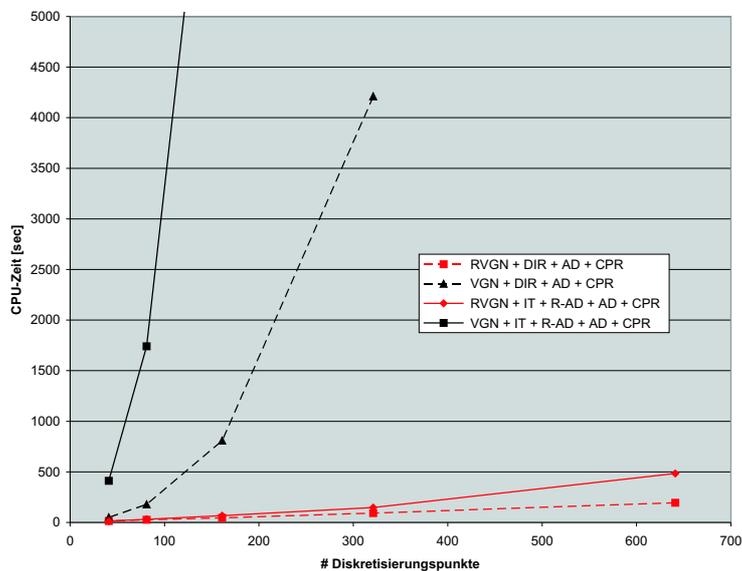


Abbildung 5.9: Rechenzeiten der vollreduzierten und nichtreduzierten Gauß-Newton-Verfahren

## 5.5 Vergleich der Modell-Implementationen (gPROMS - C/Adol-C)

Viele dynamische Optimierungsprobleme aus der chemischen Industrie werden mit gPROMS modelliert, da diese Modellierungsumgebung auf komfortable Weise ein interaktives Bearbeiten des Modells ermöglicht. Die Modelle sind über eine ESO-Schnittstelle anderen Optimierungsalgorithmen verfügbar. Dabei können sowohl Residuen als auch deren dünnbesetzte Jacobimatrizen ausgewertet werden. In diesem Zusammenhang kam es uns darauf an, die Effizienz der internen *symbolischen Differentiation* von gPROMS zu testen und mit *automatischer Differentiation* (Adol-C) zu vergleichen. Es fand dabei eine Indexierung der Modellableitungen nach differentiellen Variablen und Parametern statt (IndexSet-Erweiterung der ESO-Schnittstellen IGPDiffAlgESO sowie IGPAlgESO, siehe [35]). Folgende vollreduzierte Gauß-Newton-Varianten wurden miteinander verglichen:

- [RVGN + DIR + AD + CPR]: vollreduziertes Gauß-Newton-Verfahren, direkte Berechnung der VDAE, Modellableitungen über automatische Differentiation (Adol-C) mit Seedmatrix-Komprimierung nach Curtis-Powell-Reid,
- [RVGN + DIR + SD]: vollreduziertes Gauß-Newton-Verfahren, direkte Berechnung der VDAE, Modellableitungen über symbolische Differentiation (gPROMS),
- [RVGN + IT + R-FD + AD + CPR]: vollreduziertes Gauß-Newton-Verfahren, iterative Berechnung der VDAE, Richtungsableitungen über finite Differenzen, Modellableitungen über automatische Differentiation (Adol-C) mit Seedmatrix-Komprimierung nach Curtis-Powell-Reid,
- [RVGN + IT + R-FD + SD]: vollreduziertes Gauß-Newton-Verfahren, iterative Berechnung der VDAE, Richtungsableitungen über finite Differenzen, Modellableitungen über symbolische Differentiation (gPROMS).

In Tabelle 5.10 und 5.11 sind die Statistiken der Testläufe mit Berechnung der Modellableitungen durch symbolische Differentiation (gPROMS) angegeben. Die direkte Variante ist im Vergleich zu derjenigen mit Adol-C Ableitungen und CPR-Seeding um den Faktor 2 langsamer. Dies liegt daran, daß durch die ESO-Schnittstelle ein gewisser Overhead entsteht und bei der symbolischen Differentiation von gPROMS kein Seeding möglich ist. Am Beispiel der iterativen Variante ist deutlich zu erkennen, daß die Residuen-Auswertung des Modells über die ESO-Schnittstelle deutlich teurer ist als die Auswertung der C-Funktion. In Abbildung 5.10 sind die Rechenzeiten der oben genannten Varianten gezeigt. Dabei ist die schnellste Variante mit symbolischer Differentiation (gPROMS) um den Faktor 5 langsamer als die schnellste Variante mit automatischer Differentiation und Seedmatrix-Komprimierung.

# Knoten	CPU-Zeit [sec]	# $f_x$ -Aufrufe	# $f$ -Aufrufe	# Iter
641	389.63	6909	22035	9
321	175.45	6825	21892	9
161	82.72	6668	21360	9
81	38.71	6495	20861	9
41	18.39	5607	17852	8

Tabelle 5.10: [RVGN + DIR + SD]: Statistik für reduziertes Gauß-Newton-Verfahren, direkter Ansatz für IND und symbolische Differentiation für Modellableitungen

# Knoten	CPU-Zeit [sec]	# $f_x$ -Aufrufe	# $f$ -Aufrufe	# $Df \cdot w$ -Aufrufe	# Iter
641	540.36	508	79726	60124	9
321	234.89	513	76540	57704	9
161	105.97	542	76957	58048	10
81	25.91	456	68552	51756	8
41	6.35	418	60603	45864	8

Tabelle 5.11: [RVGN + IT + R-FD + SD]: Statistik für reduziertes Gauß-Newton-Verfahren, iterativer Ansatz für IND, symbolische Differentiation für Modellableitungen und Richtungsableitungen über finite Differenzen

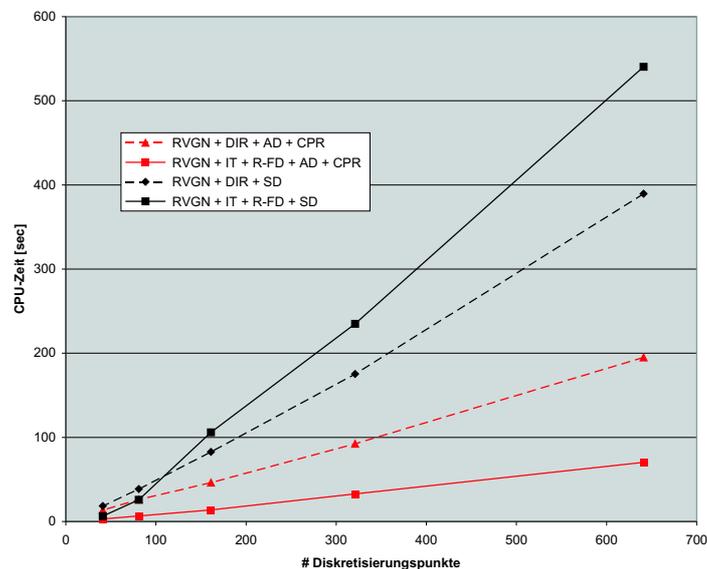


Abbildung 5.10: Rechenzeiten der vollreduzierten Gauß-Newton-Verfahren mit symbolischer und automatischer Differentiation

# Kapitel 6

## Optimierung einer kontinuierlichen Destillationskolonne

Destillationskolonnen werden in der Chemie zur Trennung von (nichtazeotropen) Gemischen eingesetzt. Dabei gibt es unterschiedliche Zielkriterien: Produktreinheit und Energiekosten. Bei der hier betrachteten kontinuierlichen Destillation eines binären Gemisches aus Methanol und n-Propanol soll der Prozeß so gefahren werden, daß möglichst schnell die gewünschte Reinheit der Produkte erreicht wird. Dies wird erreicht, indem die Temperatur und damit auch die Konzentrationen auf (sensitiven) Referenzböden kontrolliert wird. Die verwendete Problemformulierung hat einen regelungstechnischen Hintergrund: so soll die Reinheit der Produkte nach einer Störung (z.B. des Rückflusses aus dem Kondensator) möglichst schnell wiederhergestellt werden. In Abschnitt 6.1 gehen wir zunächst auf das Modell ein und beschreiben danach das oben genannte Optimierungsproblem in Abschnitt 6.2. In Abschnitt 6.3 erläutern wir anschließend die Vor- und Nachteile verschiedener reduzierter SQP-Ansätze im NMPC-Kontext.

### 6.1 Modell

Im folgenden wird ein Modell einer kontinuierlichen Destillationskolonne beschrieben, die als Pilot-Prozeß am Institut für Systemdynamik und Regelungstechnik (ISR) der Universität Stuttgart betrieben wird. Die Kolonne hat einen Durchmesser von 0.1 m und eine Höhe von 7 m; sie besteht aus 40 Böden. Der Dampf am Kopf der Kolonne wird vollständig durch einen Kondensator kondensiert. Der Sumpf wird elektrisch beheizt. Eine schematische Darstellung ist in Abbildung 6.1 gezeigt. Das vorgeheizte flüssige Gemisch wird kontinuierlich einem Zulaufboden in der Mitte der Kolonne zugeführt. Zur Steuerung stehen die Heizleistung des Sumpfs  $Q$  und das Rücklaufverhältnis  $L_{vol}/D_{vol}$  zur Verfügung. Die Reinheit der Produktströme  $D_{vol}$  und  $B_{vol}$  soll garantiert werden. Die Modellformulierung, welche hydrodynamische Effekte berücksichtigt, wird in Diehl *et al.* (2001, [29] und [31] sowie 2002, [28]) beschrieben. Wir geben sie daher nur zusammenfassend an. Seien dazu die  $N = 40$  Böden von unten nach oben nummeriert, wobei der Sumpf und der Kondensator die Boden-Nummer 0 bzw.  $N + 1$  haben. Die kontinuierliche Zufuhr des flüssigen Gemisches geschieht auf Boden  $N_F = 21$ . Es werden auf jedem Boden die mo-

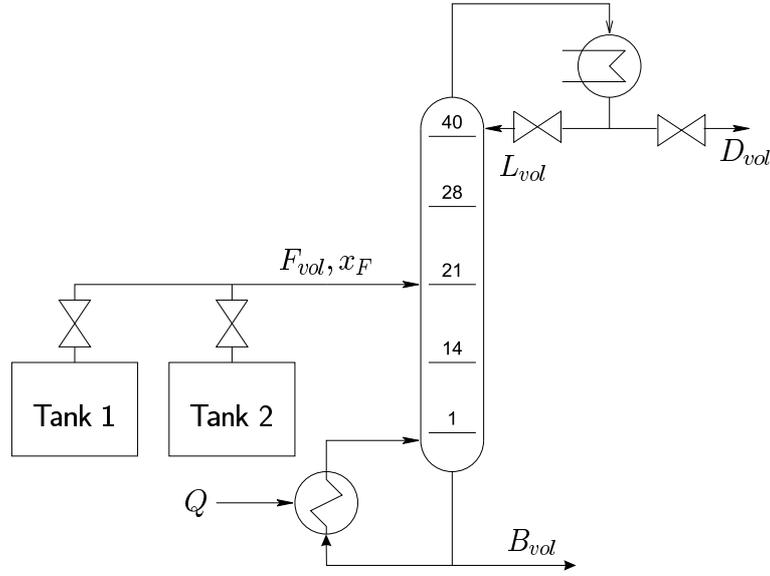


Abbildung 6.1: Schema der modellierten Destillationskolonne

laren Flüsse der Dampf- und Flüssigkeitsphasen betrachtet. Unter der Annahme, daß der Druck auf den Böden, dem Kondensator und dem Sumpf sowie die Volumen-Holdups im Kondensator und Sumpf konstant sind sowie die hydrodynamische Effekte mit einer Heuristik, basierend auf der bekannten Wehr-Formel von Francis, beschrieben werden können, ergibt sich die nachfolgend diskutierte steife DAE. Dabei werden folgende Bezeichnungen verwendet: Temperaturen  $T_l$ , molare Dampf- und Flüssigkeitsströme  $V_l$  und  $L_l$ , molare Konzentrationen von Methanol in Dampf- und Flüssig-Phase  $Y_l$  und  $X_l$ , molarer Sumpfproduktstrom  $B$ , molarer Destillatstrom  $D$ , Drücke  $P_l$  und molare Flüssigkeitsmengen  $n_l$  jeweils auf den Böden  $l = \{0, 1, \dots, N, N + 1\}$ :

- Gesamt-Massenbilanzen für die molaren Flüssigkeitsmengen auf Böden  $l = 1, \dots, N_{F-1}, N_{F+1}, \dots, N$  und Zulaufboden  $N_F$ :

$$\dot{n}_l = V_{l-1} - V_l + L_{l+1} - L_l, \quad (6.1a)$$

$$\dot{n}_{N_F} = V_{N_F-1} - V_{N_F} + L_{N_F+1} - L_{N_F} + F, \quad (6.1b)$$

- Bilanzen für die molaren Methanolvolumen in der Flüssigkeit von Sumpf, Böden  $l = 1, \dots, N_{F-1}, N_{F+1}, \dots, N$ , Zulaufboden  $N_F$  und Kondensator  $N + 1$ :

$$\dot{X}_0 n_0 + X_0 \dot{n}_0 = -V_0 Y_0 + L_1 X_1 - B X_0, \quad (6.1c)$$

$$\dot{X}_l n_l + X_l \dot{n}_l = V_{l-1} Y_{l-1} - V_l Y_l + L_{l+1} X_{l+1} - L_l X_l, \quad (6.1d)$$

$$\begin{aligned} \dot{X}_{N_F} n_{N_F} + X_{N_F} \dot{n}_{N_F} &= V_{N_F-1} Y_{N_F-1} - V_{N_F} Y_{N_F} \\ &+ L_{N_F+1} X_{N_F+1} - L_{N_F} X_{N_F} + F X_F, \end{aligned} \quad (6.1e)$$

$$\dot{X}_{N+1} n_{N+1} + X_{N+1} \dot{n}_{N+1} = V_N Y_N - D X_{N+1} - L_{N+1} X_{N+1}. \quad (6.1f)$$

- Enthalpie-Bilanzen für die Böden  $l = 1, \dots, N_{F-1}, N_{F+1}, \dots, N-1$ , den Zulaufboden  $N_F$  und den Boden  $N$  mit Dampf- und Flüssigkeitsenthalpien  $h^L$  und  $h^V$  und der Temperatur  $T_C$  des Kondensats:

$$\dot{n}_l h_l^L + n_l \left( \frac{\partial h_l^L}{\partial X_l} \dot{X}_l + \frac{\partial h_l^L}{\partial T_l} \dot{T}_l \right) = V_{l-1} h_{l-1}^V - V_l h_l^V + L_{l+1} h_{l+1}^L - L_l h_l^L, \quad (6.2a)$$

$$\begin{aligned} & \dot{n}_{N_F} h_{N_F}^L + n_{N_F} \left( \frac{\partial h_{N_F}^L}{\partial X_{N_F}} \dot{X}_{N_F} + \frac{\partial h_{N_F}^L}{\partial T_{N_F}} \dot{T}_{N_F} \right) \\ &= V_{N_F-1} h_{N_F-1}^V - V_{N_F} h_{N_F}^V + L_{N_F+1} h_{N_F+1}^L - L_{N_F} h_{N_F}^L \\ & \quad + F h^L(X_F, T_F, P_F), \end{aligned} \quad (6.2b)$$

$$\begin{aligned} & \dot{n}_N h_N^L + n_N \left( \frac{\partial h_N^L}{\partial X_N} \dot{X}_N + \frac{\partial h_N^L}{\partial T_N} \dot{T}_N \right) \\ &= V_{N-1} h_{N-1}^V - V_N h_N^V + L_{N+1} h^L(X_{N+1}, T_C, P_{N+1}) - L_N h_N^L, \end{aligned} \quad (6.2c)$$

- Hydrodynamik: molare Dampf-Flüsse werden durch heuristische Formel basierend auf der Wehr-Formel von Francis mit den Parametern  $n_l^{ref}$  und  $W_l$  für die Böden  $l = 1, \dots, N$  bestimmt:

$$L_l V^m(X_l, T_l) = W_l (n_l^v - n_l^{ref})^{\frac{3}{2}}, \quad (6.2d)$$

- Bestimmung der Temperaturen auf den Böden  $l = 0, 1, \dots, N + 1$  mit Hilfe der Dalton-Formel (Summe der Partialdrücke  $P_k^s$  gleich Gesamtdruck):

$$P_l - P_1^s(T_l) X_l - P_2^s(T_l) (1 - X_l) = 0, \quad (6.2e)$$

Die im DAE-Modell vorkommenden Hilfsgrößen  $Y_l$  und  $P_l$  ( $l = 0, \dots, N$ ),  $F$ ,  $n_0$ ,  $n_{N+1}$ ,  $L_{vol}$ ,  $P_k^s$ ,  $V_0$ ,  $h^L$  und  $h^V$  wurden direkt berechnet:

- molare Konzentrationen von n-Propanol:  $X_{n-Propol,l} := 1 - X_l$ ,  $l = 0, \dots, N + 1$ ,
- molare Konzentrationen der Flüsse  $Y_l$  der Dampf-Phase von Methanol auf den Böden  $l = 1, \dots, N$  werden mit einem Parameter  $\alpha_l$  modelliert, der Abweichungen vom realen Prozess beschreibt. Dabei ist  $Y_0 := \frac{P_1^s(T_0)}{P_0} X_0$  und

$$Y_l = \alpha_l \frac{P_1^s(T_l)}{P_l} X_l + (1 - \alpha_l) Y_{l-1}, \quad (6.3a)$$

- Druck im Kondensator sei gleich dem äußeren Druck:  $P_{N+1} := P_{top}$ , Drücke auf Böden  $l = N, \dots, 0$  seien definiert durch  $P_l := P_{l+1} + \Delta P_l$ ,
- molares Volumen des flüssigen Gemisches  $V^m(X_l, T_l) := X V_1^m(T) + (1 - X) V_2^m(T)$  setzt sich zusammen aus den molaren Volumen der unverdünnten Einzelkomponenten  $V_k^m(T)$ ,  $k = 1, 2$ , , welche von den molaren Koeffizienten  $a_k, b_k, c_k, d_k$  abhängen (siehe Tabelle A.3 im Anhang A.2):

$$V_k^m(T) := \frac{1}{a_k} \exp_{b_k} (1 + \exp_{(1-T/c_k)}(d_k)), \quad (6.3b)$$

- Zusammenhang zwischen molarem Zulauf-Strom  $F$  und Zulauf-Volumenstrom  $F_{vol}$ :

$$F_{vol} = V^m(X_F, T_F) F, \quad (6.3c)$$

- molare Flüssigkeitsmengen im Sumpf  $n_0$  und Kondensator  $n_{N+1}$  (unter der Bedingung, daß die Flüssigkeitsvolumina dort konstant sind):

$$-V^m(X_0, T_0)(-V_0 + L_1 - B) = \frac{\partial V^m}{\partial(X, T)}(\dot{X}_0, \dot{T}_0)^T n_0, \quad (6.3d)$$

$$-V^m(X_{N+1}, T_{N+1})(V_N - D - L_{N+1}) = \frac{\partial V^m}{\partial(X, T)}(\dot{X}_{N+1}, \dot{T}_{N+1})^T n_{N+1}, \quad (6.3e)$$

- Rücklauf-Volumenstrom  $L_{vol}$  aus Kondensator in die Kolonne:

$$L_{vol} = V^m(X_{N+1}, T_C)L_{N+1}, \quad (6.3f)$$

- Partialdrücke  $P_k^s(T) := \exp\left(A_k - \frac{B_k}{T+C_k}\right)$  der unverdünnten Komponenten  $k = 1, 2$  mit Koeffizienten  $A_k, B_k, C_k$  in Tabelle A.4).
- Dampf-Strom  $V_0$  im beheizten Sumpf mit Heizleistung  $Q$  und dem Parameter der Verlustleistung  $Q_{loss}$ :

$$\dot{n}_0 h_0^L + n_0 \left( \frac{\partial h_0^L}{\partial X_0} \dot{X}_0 + \frac{\partial h_0^L}{\partial T_0} \dot{T}_0 \right) = Q - Q_{loss} - V_0 h_0^V + L_1 h_1^L - B h_0^L, \quad (6.3g)$$

- molare Flüssigkeits- und Dampfenenthalpien  $h^L(X, T)$  und  $h^V(Y, T, P)$ :

$$h^L(X, T) := X h_1^L(T) + (1 - X) h_2^L(T), \quad (6.3h)$$

$$h^V(Y, T, P) := Y h_1^V(T, P) + (1 - Y) h_2^V(T, P), \quad (6.3i)$$

mit den zugehörigen Reinstoff-Enthalpien:

$$h_k^L(T) := C \left\{ h_{1,k}(T - T_0) + h_{2,k}(T - T_0)^2 + h_{3,k}(T - T_0)^3 \right\}, \quad (6.3j)$$

$$h_k^V(T, P) := h_k^L(T) + RT_k^c \sqrt{1 - \frac{P}{P_k^c} \left( \frac{T}{T_k^c} \right)^{-3}} \cdot \left\{ a - b \frac{T}{T_k^c} + c \left( \frac{T}{T_k^c} \right)^7 + \Omega_k \left( d - e \frac{T}{T_k^c} + f \left( \frac{T}{T_k^c} \right)^7 \right) \right\}, \quad (6.3k)$$

mit  $T_0 = 273.15$  K,  $C = 4.186$  J mol<sup>-1</sup>,  $R = 8.3147$  J mol<sup>-1</sup> K<sup>-1</sup>,  $a = 6.09648$ ,  $b = 1.28862$ ,  $c = 1.016$ ,  $d = 15.6875$ ,  $e = 13.4721$  und  $f = 2.615$  und den Enthalpie-Koeffizienten in Tabelle A.5,

- andere System-Parameter in Tabelle A.6.

Damit können jetzt die differentiellen und algebraischen Variablen definiert werden:  
 $x := (X_0, \dots, X_{N+1}, n_1, \dots, n_N)^T$  und  $z := (L_1, \dots, L_N, V_1, \dots, V_N, T_0, \dots, T_{N+1})^T$ .

## 6.2 Aufsetzen eines Offline-Optimierungsproblems und Ergebnis

Der Prozeß wird gesteuert über den Rücklauf in die Kolonne und die Heizleistung:  $u := (L_{vol}, Q)$ . Die schnelle Einhaltung der Reinheit der Komponenten  $B$  und  $D$  im Sumpf und Kondensator nach Störungen des stationären Zustands, z. B. durch Abfall des Rücklaufes  $L_{vol}$  in die Kolonne für wenige Minuten, dient hier als Offline-Optimierungsproblem. Das System soll also schnell wieder in einen Steady-State  $x_s$  überführt werden. Die zugehörige Steuerung wird mit  $u_s := (4.1833 \text{ l/h}, 2.4899 \text{ kW})$  bezeichnet. Wie schon erwähnt, werden nicht die Konzentrationen gesteuert, sondern die Temperaturen auf bestimmten Referenzböden:  $\tilde{T}z := (T_{28}, T_{14})^T$ . Dazu wird die Abweichung von Referenztemperaturen  $\tilde{T}_{ref} := (T_{28}^{ref}, T_{14}^{ref})^T := (70 \text{ °C}, 88 \text{ °C})^T$  über das Integral eines regularisierten Least-Squares-Terms  $L$  bestraft:

$$L(z, u, u_s) := \|\tilde{T}z - \tilde{T}_{ref}\|_2^2 + \|R(u - u_s)\|_2^2,$$

wobei  $R := \text{diag}\{1.0 \text{ °C h l}^{-1}, 0.05 \text{ °C kW}^{-1}\}$  die diagonale Gewichtungsmatrix darstellt. Damit kann das Offline-Optimierungsproblem auf einem Zeithorizont  $[t_0, t_0 + T_p]$  mit  $t_0 := 0 \text{ [sec]}$  und  $T_p := 3600 \text{ [sec]}$  wie folgt beschrieben werden:

$$\min_{u(\cdot), x(\cdot), z(\cdot)} \int_{t_0}^{t_0 + T_p} L(z(t), u(t), u_s) dt \quad (6.4a)$$

unter dem DAE-Modell (Umformulierung der differentiellen Gleichungen 6.1 in ein explizites System und Hinzunahme der algebraischen Gleichungen 6.2):

$$\left. \begin{aligned} \dot{x}(t) &= f(x(t), z(t), u(t), p) \\ 0 &= g(x(t), z(t), u(t), p) \end{aligned} \right\}, \quad t \in [t_0, t_0 + T_p]. \quad (6.4b)$$

mit den Startwerten:

$$x(t_0) = x_0 \quad (6.4c)$$

und problemtechnische Schranken an die Produktströme:

$$h(x(t), z(t), u(t), p) := (D, B)^T \geq 0, \quad t \in [t_0, t_0 + T_p]. \quad (6.4d)$$

Die Steuerung  $u$  wurde dabei auf dem Zeithorizont  $[t_0, t_0 + T_c]$  stückweise konstant mit 6 äquidistanten Intervallen und auf  $[t_0 + T_c, t_0 + T_p]$  konstant (identisch  $u_s$ ) gewählt, wobei  $T_c := 1800 \text{ [sec]}$ .

Das Lagrange-Zielfunktional wurde mit der Trapezregel approximiert, wobei in den Mehrzielintervallen der ersten Stufe jeweils 50 Stützstellen und im Mehrzielintervall der zweiten Stufe 500 Stützstellen gewählt wurden. Durch diese Umformung wurde das Lagrange-Zielfunktional in ein LSQ-Zielfunktional umgewandelt, welches mit einer Gauß-Newton-Variante aus Kapitel 3 gelöst wurde. Die Ergebnisse der Optimierung sind in Abbildung 6.2 zu sehen.

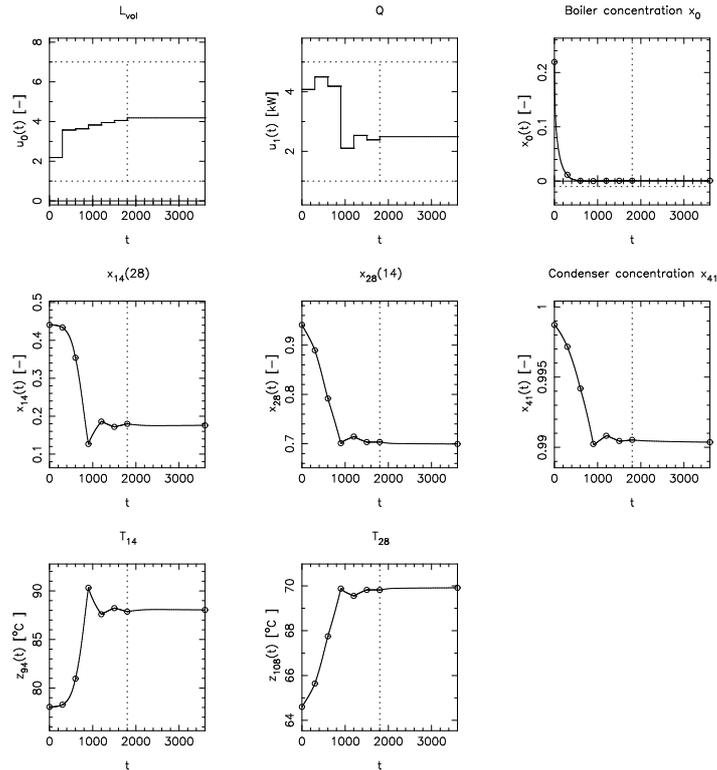


Abbildung 6.2: Optimale Betriebsführung der Destillationskolonne (nach Störung)

### 6.3 Vergleich zu MUSCOD-II

Zunächst betrachten wir den Offline-Fall, das heißt, es geht um eine schnelle Berechnung des obigen Optimierungsproblems aus Abschnitt 6.2. Alle Rechnungen wurden auf einer Dual-Xeon-Maschine mit jeweils 2.8 GHz, 2 GB RAM sowie Linux als Betriebssystem mit Kernel 2.4.19 und gcc-Suite Version 3.2 (mit Optimierung `-O3`) durchgeführt. Die Testläufe wurden durch folgende Vollschrift-Varianten mit Integrationsgenauigkeit  $1.E-6$  und Abbruchgenauigkeit  $1.E-3$  ( $l_2$ -Norm des Schritts) gelöst, wobei unterschiedliche Reduktionsstrategien und IND-Ansätze miteinander verglichen wurden:

- [RVGN(x,z) + IT]: reduziertes Gauß-Newton-Verfahren bzgl. differentieller *und* algebraischer Zustandsvariablen sowie iterativem IND-Ansatz mit Richtungsableitungen durch finite Differenzen und Modellableitungen durch automatische Differentiation (Adol-C) und CPR-Seeding,
- [RVGN(x,z) + DIR]: reduziertes Gauß-Newton-Verfahren bzgl. differentieller *und* algebraischer Zustandsvariablen sowie direktem IND-Ansatz mit Modellableitungen durch automatische Differentiation (Adol-C) und CPR-Seeding,
- [RVGN(z) + IT]: reduziertes Gauß-Newton-Verfahren bzgl. algebraischer Variablen (MUSCOD-II) sowie iterativem IND-Ansatz mit Richtungsableitungen durch fini-

te Differenzen und Modellableitungen durch automatische Differentiation (Adol-C) und CPR-Seeding,

- [RVGN(z) + DIR]: reduziertes Gauß-Newton-Verfahren bzgl. algebraischer Variablen (MUSCOD-II) sowie direktem IND-Ansatz mit Modellableitungen durch automatische Differentiation (Adol-C) und CPR-Seeding.

Die Statistiken der Optimierungsläufe sind in Tabelle 6.1 gezeigt. Hauptanteil des Rechenaufwands stellt die Berechnung der Ableitungsinformationen (VDAE) dar. Durch die festen Anfangswerte kann die Anzahl der Freiheitsgrade in dem reduzierten Gauß-Newton-Algorithmus RVGN(x,z) auf die Kontrollen beschränkt werden. Zwar nimmt die Zahl der Freiheitsgrade in jedem Mehrzielintervall von  $n_u = 2$  bis auf  $n \cdot n_u = 14$  konstant zu, bleibt jedoch immer deutlich kleiner als  $n_x + n_u = 84$ , was in einem Speedup in der Ableitungsberechnung resultiert (siehe Abbildung 6.3). Damit ist die Variante mit Reduktion bzgl. differentieller und algebraischer Variablen um den Faktor 5 schneller als die Variante mit Reduktion nur bzgl. algebraischer Variablen.

Im Online-Fall kommt es auf eine schnelle Antwort des Optimierers auf neue Anfangs-

Zeit [sec] / Anzahl	RVGN(x,z)		RVGN(z)	
	IT	DIR	IT	DIR
CPU	31.34	98.34	247.04	159.67
Integration	5.88	5.87	6.79	6.74
VDAE	24.68	91.63	237.75	150.41
Reduktion	0.76	0.77	1.61	1.66
Kondensieren	-	-	0.26	0.27
QP lösen	< 0.01	< 0.01	< 0.01	< 0.01
Richtungsableitungen	17.21	-	162.47	-
Iterationsmatrizen	1.76	54.66	2.07	59.73
# Richtungsableitungen	82083	-	814581	-
# Iterationsmatrizen	151	4049	183	4458
# Integrationssschritte	3691	3923	4150	4308
# Iterationen	6	6	7	7

Tabelle 6.1: Statistik der GN-Reduktionsstrategien bzgl. differentieller (x) und/oder algebraischer (z) Variablen sowie IND-Ansätze (direkt oder iterativ)

werte der Zustandsvariablen an. In diesem Zusammenhang wird nun eine Echtzeititeration für den reduzierten SQP-Algorithmus aus Kapitel 3 vorgestellt. Dieser ist an die Echtzeititeration von MUSCOD-II angelehnt (siehe Diehl 2002, [28]). Ausschlaggebend ist, daß in der Feedback-Phase nur die Inhomogenität bzw. Bildraumkomponente  $S^{\mathcal{R}}y^{\mathcal{R}}$  berechnet und das reduzierte QP gelöst werden muß, um eine Feedback-Steuerung zu erhalten. Dabei kann die Berechnung der Inhomogenität beschleunigt werden, indem die Integrator-Informationen zur Berechnung der Nominaltrajektorie gespeichert werden und somit nur die Newton-Iterationen für die Richtungsableitung bzgl. der Inhomogenität durchgeführt werden müssen. Der Aufwand hängt dabei im wesentlichen von der Anzahl

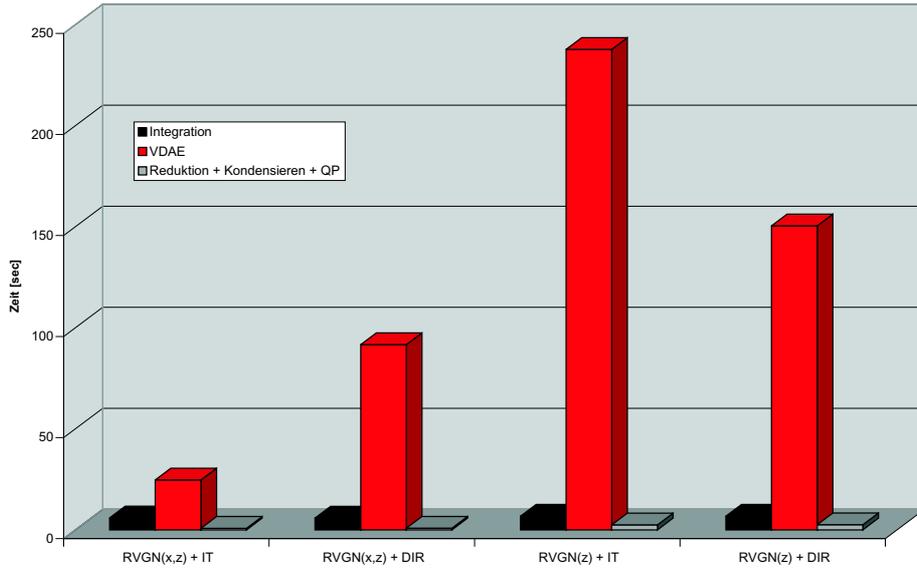


Abbildung 6.3: Rechenzeiten der GN-Reduktionsstrategien bzgl. differentieller (x) und/oder algebraischer (z) Variablen sowie IND-Ansätze (direkt oder iterativ)

der Integratorschritte und der durchschnittlichen Anzahl der Newton-Iterationen in jedem Integratorschritt ab. Betont sei, daß das Kondensieren bereits bei der Berechnung der Koordinaten-Basis stattfindet. Nachfolgend angegeben ist der Algorithmus einer Echtzeititeration mit Gauß-Newton Hessematrix und  $n$  Mehrzielintervallen:

1. Berechne die Koordinaten-Basis  $S^{\mathcal{R}}$  und  $S^{\mathcal{N}}$ . Integriere und löse simultan die reduzierte Richtungs-VDAE (siehe Kapitel 4) ohne die Inhomogenität  $S^{\mathcal{R}}y^{\mathcal{R}}$ . Berechne den Zielfunktionswert  $f$  und die Residuen der Beschränkungen  $G_2, H$ .
2. Berechne die reduzierten Jacobimatrizen der Gleichungs- und Ungleichungsrestriktionen  $\nabla G_2^T S^{\mathcal{N}}$  und  $\nabla H^T S^{\mathcal{N}}$  sowie die der LSQ-Zielfunktion  $\nabla f^T S^{\mathcal{N}}$  und berechne die reduzierte Gauß-Newton-Hessematrix.
3. Wenn der Anfangswert  $x_0$  bekannt ist, berechne die Inhomogenität bzw. Bildraumkomponente  $S^{\mathcal{R}}y^{\mathcal{R}}$  und die projizierten Funktionswerte der Restriktionen und Zielfunktion:  $G_2 + \nabla G_2^T S^{\mathcal{R}}y^{\mathcal{R}}$ ,  $H + \nabla H^T S^{\mathcal{R}}y^{\mathcal{R}}$  und  $f + \nabla f^T S^{\mathcal{R}}y^{\mathcal{R}}$ .
4. Löse das reduzierte LSQ-QP (3.7) und erhalte  $y^{\mathcal{N}} := (\Delta q_0, \dots, \Delta q_n)^T$  sowie die Feedback-Steuerung  $q_0 + \Delta q_0$ . Diese wird der Prozess-Steuerung übergeben.
5. Expandiere Lösung des reduzierten LSQ-QP. Berechne den Vollschrirt  $\Delta w := S^{\mathcal{R}}y^{\mathcal{R}} + S^{\mathcal{N}}y^{\mathcal{N}}$ . Gehe mit neuer Iterierten  $w + \Delta w$  zu Schritt 1.

Wir fassen jetzt Schritt 3 und 4 als *Feedback-Phase* und Schritt 5, 1 und 2 als *Vorbereitungsphase* zusammen. Im Vergleich zur Echtzeititeration von MUSCOD-II dauert die Feedback-Phase um die Berechnung der Inhomogenität länger, welche aber mit Festhalten

der Nominaltrajektorien-Informationen beim Destillationskolonnen-Modell immer noch im Zentelsekundenbereich liegt (siehe Tabelle 6.2). Dagegen ist die Vorbereitungsphase wie oben gezeigt um den Faktor 5 schneller.

Zeit	[sec]
Integration	1.21
VDAE	3.15
Inhomogenität	0.36
Reduktion	0.11
Lösung des RQP	< 0.01

Tabelle 6.2: Statistik einer Echtzeititeration für reduziertes Gauß-Newton-Verfahren bzgl. differentieller und algebraischer Variablen

# Kapitel 7

## Parameterschätzung und Optimierung eines katalytischen Rohrreaktors

Heterogen katalysierte, exotherme Gasphasenreaktionen werden in der chemischen und petrochemischen Industrie unter anderem in Rohrbündelreaktoren durchgeführt. Diese bestehen typischerweise aus mehreren tausend einzelnen Reaktorrohren innerhalb einer gemeinsamen Hülle (siehe Abbildung 7.1). Der umzusetzende Gasstrom (Reaktanden)

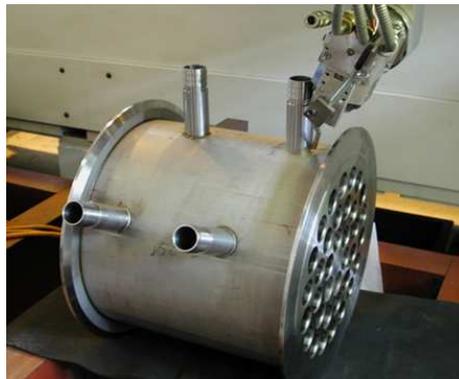


Abbildung 7.1: Rohrbündelreaktor

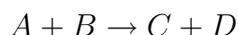
fließt innerhalb der Rohre durch ein Bett von porösen Katalysatorpartikeln, wo er zu dem gewünschten Produkt reagiert. Außerhalb der Rohre (im Mantelraum) fließt eine Flüssigkeit, die als Wärmeaustauscher fungiert und die durch die Reaktion entstandene Wärme abführen soll. Gewöhnlich findet die Reaktion nur innerhalb eines kleinen Abschnitts der Katalysatorschüttung statt. Diese Reaktionszone bewegt sich langsam von der Einlaß-Seite der Rohrschüttung zur Auslaß-Seite, da der Katalysator zum Beispiel durch Verkokung deaktiviert wird. Wenn die Reaktionszone sich dem Ende der Katalysatorschüttung nähert, das heißt, die Reaktanden nicht mehr umgesetzt werden (Durchbruchbedingung), muß der Reaktor heruntergefahren werden, damit der Katalysator regeneriert werden kann.

In der chemischen Reaktionstechnik haben sich vor allem Kontinuumsmodelle zur mathematischen Modellierung der Temperatur- und Konzentrationsfelder im durchströmten Festbettrohrreaktor durchgesetzt (siehe Adler 2000, [1] und [2]). In der Praxis werden heterogene und pseudo-homogene Kontinuumsmodelle verwendet. Sie unterscheiden sich darin, ob die fluide Phase (Gas) von der Katalysatorphase in der Modellierung unterschieden wird (heterogenes Modell) oder ob das disperse System quasihomogen betrachtet wird (pseudo-homogenes Modell). Weiterhin wird zwischen stationären und instationären Modellierung, bzgl. der räumlichen Ausdehnung des Lösungsgebiets (ein- beziehungsweise zweidimensional), bzgl. der Berücksichtigung der axialen und radialen Leitung und bzgl. der zugrundeliegenden (mehr oder weniger detaillierten) Chemie unterschieden. Weitere Punkte, die im industriellen Rahmen eine Rolle spielen, sind der Stoff- und Wärmetransport in Katalysatorpartikeln, die Katalysatordeaktivierung bei industriellen Prozessen sowie der Aspekt der Strömungsungleichverteilung im Mantelraum.

Im folgenden wird ein mathematisches Modell vorgestellt, welches von der Firma Bayer, Leverkusen, für Parameterschätz- und Optimierungsstudien zur Verfügung gestellt wurde. Dieses Modell dient bei Bayer zur Beschreibung eines Rohrreaktors im Pilotprozeß und wurde als gPROMS-Modell implementiert.

## 7.1 Modell

Um eine modellgestützte Optimierung des Durchsatzes vorzunehmen, betrachten wir ein einzelnes zylindrisches Reaktionsrohr. Als Modell-Grundlage dient ein instationäres, pseudo-homogenes 2D-Modell, welche das heterogene, zweiphasige System (zusammenhängende fluide Gasphase und disperse Katalysator-Feststoffphase) als Quasikontinuum betrachtet. Das heißt, daß keine Konzentrations- und Temperaturgradienten zwischen Gasphase und Katalysatoroberfläche berücksichtigt werden. Die Wärme- und Stofftransportvorgänge im Quasikontinuum werden durch die Ansätze von Fourier und Fick (siehe Bird *et al.* 2002, [13]) beschrieben. Für die Gasgeschwindigkeit wird Pfropfenströmung angenommen, das heißt, es wird auf eine Impulsbilanz bzgl. der Strömungsgeschwindigkeit im Festbett verzichtet, und es liegt kein radiales Porositätsprofil vor. Diese Annahmen scheinen derzeit gerechtfertigt, da bei detaillierteren Modellen unter Berücksichtigung der Porositäts- und Strömungsungleichverteilung zu große Unsicherheiten bei Modellparametern und Messungen vorliegen (siehe Bauer 2001, [8]). Die Wärme- und Stofftransportparameter setzen sich aus effektiven Transportkoeffizienten zusammen, die sich aus Beiträgen beider Phasen berechnen lassen (sowohl axiale und radiale Dispersion sowie axiale und radiale Wärmeleitfähigkeit werden berücksichtigt). Für die Katalysatorschüttung wird ein Kugelprofil der Katalysatorpartikel angenommen. Der Druckabfall im Festbett wird mittels der Ergun-Gleichung (siehe Bird *et al.* 2002, [13]) berücksichtigt, wobei die Porosität der Schüttung als konstant gewählt wurde. Bei der hier betrachteten exothermen Reaktion vom Typ



handelt es sich um einen instationären Prozess. Dies liegt an der experimentell beobachteten Deaktivierung des Katalysators (zum Beispiel Verkokung des Katalysators und damit

Hemmung der aktiven Zentren). Die Stoffdaten werden über den Querschnitt des Reaktors als konstant (Dichte der fluiden Gasphase) bzw. im gesamten Reaktor als konstant (Viskosität, Wärmeleitfähigkeit und Wärmekapazität) angenommen. Den Randbedingungen kommt in einem heterogen-gaskatalytischen Rohrreaktor eine besondere Bedeutung zu. Experimentell werden hier steile Temperaturgradienten beobachtet, wobei in der Newtonschen Theorie der Gradient in einen Temperatursprung übergeht, wenn ein endlicher Transportwiderstand an der Wand vorhanden ist. Daher wird die Randbedingung an der Wand mit dem klassischen Ansatz über einen Wärmedurchgangskoeffizienten modelliert. Modelle mit variablem radialen Wärmeleitfähigkeitsansatz und Temperaturidentität haben sich sowohl aus theoretischer als auch praktischer Sicht bisher nicht bewährt (siehe Bauer 2001, [8]). Auf die Modellierung des Wärmetransports in der Wand über eine instationäre Energiebilanzgleichung wurde verzichtet, statt dessen wurde der Wärmedurchgangskoeffizient für die Energiebilanz der Kühlmittels und für deren Randbedingung modifiziert. Dieser Wärmedurchgangskoeffizient leitet sich im stationären Fall über den Fluß eines Wärmestroms (Wärmeübergang disperses Gas/Katalysator System - Rohrwand, Wärmeleitung durch die Wand sowie Wärmeübergang Rohraußenwand - Kühlmittel) her (siehe Baehr *et al.* 2003, [3]). Weiterhin wurde angenommen, daß die Kühlmitteltemperatur entlang der Rohrwand auf der Wärmeträgerseite konstant ist.

Nachfolgend wird das Randwertproblem für die Parameterschätzung und Optimierung im Detail beschrieben. Dazu ist im Anhang A.3 eine Übersicht über die Bedeutung der im Modell vorkommenden Symbole (Tabellen A.7, A.8, A.9, A.10) angegeben. Für einen schematischen Aufbau des Rohrreaktors siehe Abbildung 7.2, in der auch der Wärmetransport durch die Wand verdeutlicht ist. Die Bilanzierungen des Quasikontinuums und des Mantelraums sowie die Randbedingungen am Rohranfang, in der Rohrmitte und an der Rohrwand ergeben unter den oben genannten Annahmen folgende instationäre 2D-PDEs:

- Stoffbilanz im Reaktionsrohr für jede an der Reaktion beteiligte Komponente  $i = 1, \dots, N$ :

$$\varepsilon \frac{\partial c_i}{\partial t} = -\frac{\partial(u_f c_i)}{\partial z} + D_{eff,z} \frac{\partial^2 c_i}{\partial z^2} + D_{eff,r} \left( \frac{\partial^2 c_i}{\partial r^2} + \frac{1}{r} \frac{\partial c_i}{\partial r} \right) + \nu_i r_0 \rho_{s,bed} a_{Kat}, \quad (7.1)$$

- Energiebilanz im Einzelrohr:

$$\begin{aligned} (\varepsilon \rho_f c_{p,f} + \rho_{s,bed} c_{p,s}) \frac{\partial T}{\partial t} = & -c_{p,f} \frac{\partial(u_f \rho_f T)}{\partial z} + \lambda_{eff,z} \frac{\partial^2 T}{\partial z^2} \\ & + \lambda_{eff,r} \left( \frac{\partial^2 T}{\partial r^2} + \frac{1}{r} \frac{\partial T}{\partial r} \right) \\ & - \Delta H_R \nu_3 r_0 \rho_{s,bed} a_{Kat}, \end{aligned} \quad (7.2)$$

- Energiebilanz im Mantelraum:

$$\begin{aligned} V_{MR} \frac{\partial(\rho_{KM} c_{p,KM} T_{KM})}{\partial t} = & \dot{m}_{KM} c_{p,KM} (T_{KM,ein} - T_{KM}) \\ & + k_\alpha A_{Rohr} \int_{z=0}^L (T(z, R) - T_{KM}) dz, \end{aligned} \quad (7.3)$$

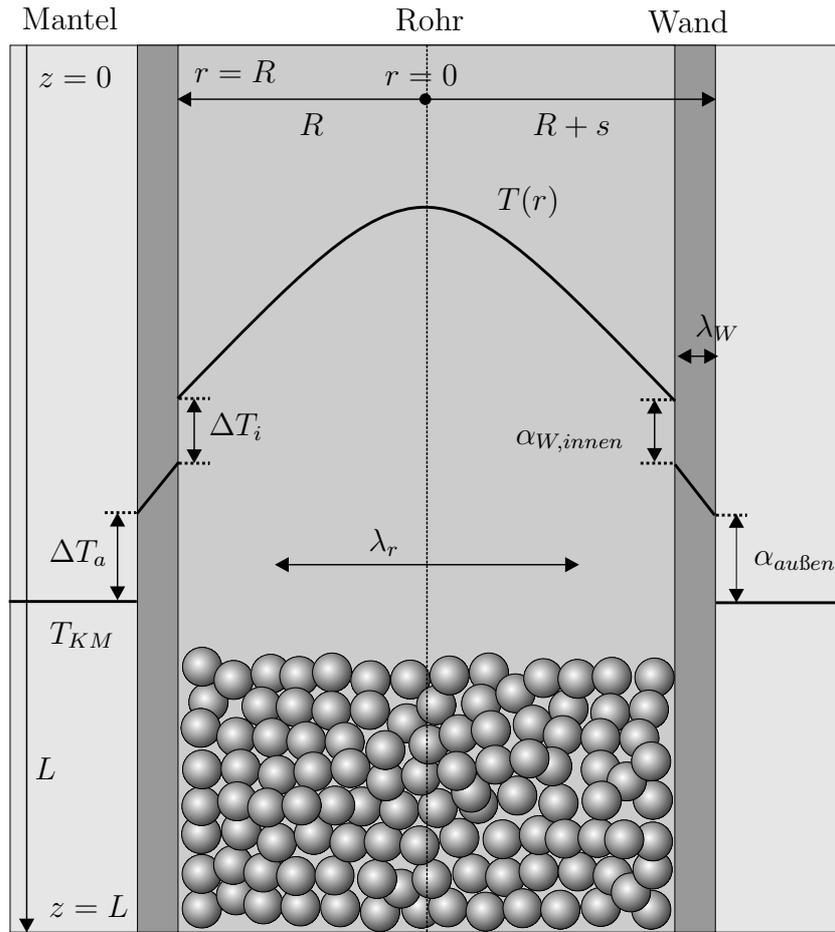


Abbildung 7.2: Schematischer Aufbau eines Rohrreaktors

- Randbedingungen am Rohranfang ( $z = 0$ ):

$$c_i(0, r) = c_{i, \text{ein}}, \quad (7.4a)$$

$$\lambda_{\text{eff}, z} \frac{\partial T(0, r)}{\partial z} = -u_f \rho_f c_{p, f} (T_{\text{ein}} - T(0, r)), \quad (7.4b)$$

- Randbedingungen in der Rohrmitte ( $r = 0$ ):

$$\frac{\partial c_i(z, 0)}{\partial r} = 0, \quad (7.5a)$$

$$\frac{\partial T(z, 0)}{\partial r} = 0, \quad (7.5b)$$

- Randbedingungen an der Rohrwand ( $r = R$ ):

$$\frac{\partial c_i(z, R)}{\partial r} = 0, \quad (7.6a)$$

$$\lambda_{\text{eff}, r} \frac{\partial T(z, R)}{\partial r} = k_\alpha (T_{KM} - T(z, R)). \quad (7.6b)$$

- Anfangsbedingungen zum Start der Reaktion ( $t = 0$ ):

$$T_{KM} = T_{KM,ein}, \quad (7.7a)$$

$$c_i = c_{i,ein}, \quad (7.7b)$$

$$T = T_{ein}. \quad (7.7c)$$

Nach Annahme der Pfropfenströmung für die Gasgeschwindigkeit  $u_f$  strömt das Gas gleichmäßig mit Leerrohrgeschwindigkeit (totale Massenflußrate geteilt durch Dichte und Rohrquerschnitt) durch den Reaktor:

$$u_f = \frac{\sum_{i=1}^N M_{i,ein}}{\rho_f \pi R^2}. \quad (7.8)$$

Wichtig für den praktischen Einsatz dieses Modells sind die effektiven Parameter, um die oben genannten Phänomene korrekt zu beschreiben. Für diese Parameter existieren Korrelationsbeziehungen über ihre Abhängigkeit von den Zustandsgrößen des Systems. Im folgenden werden diese dargestellt:

- Stofftransport: die effektiven Dispersionskoeffizienten  $D_{eff}$  setzen sich zusammen aus Beiträgen der molekularen Diffusion ( $\delta_{bed}$ ) und der Konvektion. Im vorliegenden Fall werden die effektiven Dispersionsparameter nur durch die Konvektion bestimmt.

- axiale Dispersion (siehe Edwards und Richardson 1968, [32]):

$$D_{eff,z} = \frac{d_p u_f}{\varepsilon Pe_z} \quad \text{mit} \quad \frac{1}{Pe_z} = \frac{0.73 \varepsilon}{Re_p Pr} + \frac{0.5}{1 + \frac{9.7 \varepsilon}{Re_p Pr}}. \quad (7.9)$$

- radiale Dispersion (siehe Gunn und Price 1969, [46]):

$$D_{eff,r} = \frac{d_p u_f}{\varepsilon Pe_r} \quad \text{mit} \quad \frac{1}{Pe_r} = 0.1 + \frac{2}{3} \frac{\varepsilon}{Re_p Pr}. \quad (7.10)$$

- Wärmetransport: hierbei wird die Wärmeleitfähigkeit  $\lambda_{bed}$  der statischen Schüttung berücksichtigt. Eine Beziehung, die sowohl die Porosität  $\varepsilon$  als auch die Partikelform berücksichtigt, ist im Wärmetatlas 1997, Abschnitt Dee [91] angegeben:

$$\lambda_{bed} = \lambda_f (1 - \sqrt{1 - \varepsilon} + \theta \sqrt{1 - \varepsilon}) \quad (7.11)$$

mit  $\theta = f(\varepsilon, \lambda_p, \lambda_f, \text{Partikelform})$ .

Die effektive Wärmeleitfähigkeit  $\lambda_{eff}$  ist durch die Gasströmung (Wärmeübertragung durch erzwungene Konvektion in den Hohlräumen der Schüttung) größer als die Wärmeleitfähigkeit der statischen Schüttung  $\lambda_{bed}$ .

- axiale Wärmeleitfähigkeit (siehe Edwards und Richardson 1968, [32]):

$$\lambda_{eff,z} = \lambda_{bed} + \frac{Re_p Pr \lambda_f}{K_z} \quad \text{mit} \quad \frac{1}{K_z} = \frac{0.73 \varepsilon}{Re_p Pr} + \frac{0.5}{1 + \frac{9.7 \varepsilon}{Re_p Pr}}. \quad (7.12)$$

– radiale Wärmeleitfähigkeit (siehe Wärmetlas 1997, Abschnitt Mh [92]):

$$\lambda_{eff,r} = \lambda_{bed} + \frac{Re_p Pr \lambda_f}{K_r}$$

mit  $K_r = K_\infty f\left(\frac{D}{d_p}\right) = K_\infty \left(2 - \left(1 - 2 \frac{d_p}{D}\right)^2\right)$ , (7.13)

wobei  $K_\infty = 7.4 \in [7.0, 11.0]$ .

- Wärmeübergang an der Rohrwand zwischen Festbett und Kühlmittel: Der Wärmedurchgangskoeffizient  $k_\alpha$  kann aus innerem und äußeren Wärmeübergangskoeffizienten sowie der Wärmeleitfähigkeit  $\lambda_W$  und der Dicke  $s$  der Rohrwand berechnet werden (siehe Baehr *et al.* 2003, [3]):

$$\frac{1}{k_\alpha R} = \frac{1}{R \alpha_{W,innen}} + \frac{\log \frac{R+s}{R}}{\lambda_W} + \frac{1}{(R+s) \alpha_{außen}}. \quad (7.14)$$

Dabei wird angenommen, daß der Wärmeübergangskoeffizient an der Außenseite ( $\alpha_{außen}$ ) größer als an der Innenseite ( $\alpha_{W,innen}$ ) ist ( $\rightarrow$  Bedingung an die Umströmungsbedingung der Rohre im Mantelraum).

– innerer Wärmeübergang (siehe Martin und Nilles 1993, [62]):

$$\alpha_{W,innen} = \alpha_{W,0} + 0.19 \frac{\lambda_f}{d_p} Re^{0.75} Pr^{0.33}$$

mit  $\alpha_{W,0} = \left(1.3 + 5.0 \frac{d_p}{D}\right) \frac{\lambda_{bed}}{d_p}$ , (7.15)

– äußerer Wärmeübergang: der äußere Wärmeübergangskoeffizient wird als über die Länge des Rohres konstanter Wert angesetzt,

$$\alpha_{außen} = 1000.0 [W/(m^2 K)], \quad (7.16)$$

in Abhängigkeit vom Volumenstrom des Kühlmittels und der Rohrumströmung (Längs- und Querströmung) und limitiert bei ausreichender Zirkulation des Kühlmittels den Wärmedurchgang nicht!

Der Druckverlust wird unter den oben genannten Annahmen (keine radiale Porositäts- und Strömungungleichverteilung im Festbett) durch die Ergun-Gleichung berechnet:

$$-\frac{\partial p}{\partial z} = 150.0 \frac{(1-\varepsilon)^2}{\varepsilon^3 d_p^2} \eta_f u_f + 1.75 \frac{1-\varepsilon}{\varepsilon^3 d_p} \rho_f u_f^2. \quad (7.17)$$

Die Porosität der Schüttung wurde unter Annahme für Kugelschüttung als konstant angesetzt:

$$\varepsilon = 0.4 \in [0.37, 0.42]. \quad (7.18)$$

Die Berechnung der Stoffwerte berücksichtigt die Zusammensetzung des Gasgemisches. Die Gewichtung der einzelnen Komponenten wurde mit Massenanteilen ( $w_i$ ) bzw. Molanteilen ( $y_i$ ) vorgenommen. Die in der Gleichung 7.17 vorkommende Dichte des Gasgemisches wurde nach dem idealen Gasgesetz berechnet:

$$\rho_f = \frac{p_{abs} \sum_{i=1}^n (M_i y_i)}{R_g T}.$$

Für die Kinetik der betrachteten exothermen Reaktion ( $A + B \rightarrow C + D$ ) wurde ein Potenzansatz gewählt, bei dem die Deaktivierung (und damit die Instationarität) durch einen Aktivitätsfaktor  $a_{Kat}$  berücksichtigt wird:

$$r_0 = k c_A^n a_{Kat} \quad \text{mit} \quad k = A_0 \exp\left(-\frac{E_a}{R_g T}\right), \quad n = 1. \quad (7.19)$$

Eine Übersicht über die möglichen Ansätze zur Beschreibung der Deaktivierung gibt Forzatti und Lietti (1999, [39]). In dieser Anwendung wird für die Deaktivierung ein formal-kinetischer Potenzansatz gewählt,

$$\begin{aligned} \frac{\partial a_{Kat}}{\partial t} &= -r_{0,Des} a_{Kat}^m \quad \text{mit} \quad a_{Kat}|_{t=0} = 1, \\ r_{0,Des} &= k_{Des} c_A^{n_{Des}}, \quad k_{Des} = k_{0,Des} + A_{0,Des} \exp\left(-\frac{E_{a,Des}}{R_g T}\right), \\ n_{Des} &= 1, \quad m = 2, \end{aligned} \quad (7.20)$$

deren Parameter ( $k_{0,Des}$ ,  $A_{0,Des}$ ,  $E_{a,Des}$ ) sich zusammen mit den Parametern der Hauptreaktion ( $A_0$ ,  $E_a$ ) über Messungen des Temperaturprofils in der Rohrmitte schätzen lassen (siehe Abschnitt 7.3).

## 7.2 Simulation

Das parabolische Anfangs-Randwertproblem (7.1, 7.2, 7.3) mit örtlichen Randbedingungen (7.4, 7.5, 7.6) sowie zeitlichen Anfangsbedingungen (7.7) wird nun mit Hilfe der (vertikalen) Linienmethode (siehe Großmann *et al.* 1992, [45]) in eine semi-explizite DAE vom Typ (1.1b) mit Anfangsbedingungen (1.1c) überführt. Dazu wird der Ortsbereich wie folgt diskretisiert: der axiale Bereich wird in 45 Diskretisierungsintervalle unterteilt, wobei die korrespondierenden Ableitungen durch Rückwärts-Finite-Differenzen der Ordnung 1 approximiert werden. Der radiale Bereich wird in 5 Finite Elemente unterteilt, auf denen jeweils ein orthogonales Polynom in den radialen Ortskoordinaten der Ordnung 2 die differentiellen und algebraischen Variablen (für feste axiale Komponente) approximiert. Für die Definition dieser Diskretisierungsmethoden siehe Pantelides (2001, [71]). Diese Wahl der Diskretisierungsschemata ist bestimmt durch die notwendige Auswertung der Temperatur in der Rohrmitte und Stoffzusammensetzung am Ende des Reaktors und der damit geforderten Genauigkeit. Wenn insbesondere nur die Auswertung in der Rohrmitte und an der Rohrwand gefragt ist, liefert die orthogonale Kollokation auf den finiten Elementen gegenüber finiten Differenzen eine höhere Genauigkeit (siehe Nagel *et al.* 1971, [65]). Weiterhin konnte die Instationarität bei der hier betrachteten Reaktion auf die Temperatur im Quasikontinuum  $T$ , die Temperatur des Kühlmittels  $T_{KM}$  sowie die Aktivität  $a_{Kat}$  beschränkt werden. Auch konnte die Berechnung der Stoffbilanz auf die erste Komponente  $A$  beschränkt werden, da die Reaktionsrate  $r_0$  nur von dieser abhängt. Daraus resultieren die folgenden Dimensionen der DAE: 548 differentielle und 1074 algebraische Variablen.

### 7.3 Parameterschätzung

Ausgehend von dem Anfangswertproblem, wie in Abschnitt 7.2 beschrieben, sollen nun die kinetischen Parameter  $p = (A_0, E_a, k_{0,Des}, A_{0,Des}, E_{a,Des})^T$  des Modells durch Anpassung der Temperaturprofile an Temperatur-Meßwerte  $T_{mess}$  in der Rohrmitte (siehe Tabelle A.11) geschätzt werden. Diese Meßwerte wurden dabei bei konstanter Zufuhr der Reaktanden ermittelt. Aus numerischen Gründen wird zunächst die Reaktionsrate (7.20) in dem formalkinetischen Ansatz für die Aktivität modifiziert, so daß um eine Referenztemperatur  $T_{ref}$  „zentriert“ wird:

$$k_{Des} = k_{0,Des} + A_{0,Des} \exp\left(-\frac{E_{a,Des}}{R_g} \left(\frac{1}{T} - \frac{1}{T_{ref}}\right)\right). \quad (7.21)$$

Folgende regularisierte Least-Squares-Funktion (7.22) dient im Parameterschätzproblem als Zielfunktion. Dabei liegen jeweils axial 16 Meßwerte an 5 Zeitpunkten vor.

$$\sum_{i=1}^5 \sum_{j=0}^{15} \left[ \frac{T(t, z, r) - T_{mess}(t, z, r)}{T_{mess}(t, z, r)} \right]^2 (t, z, r) + \underbrace{\frac{1}{w} \sum_i \left[ \frac{p_i - p_{reg,i}}{p_{reg,i}} \right]^2}_{\text{Regularisierungsterm}}. \quad (7.22)$$

$= (\frac{i}{5} t_c, \frac{j}{15} L, 0)$

Für die Gewichtung des Regularisierungsterms wurde  $w = 100$  gewählt. Die Regularisierungswerte  $p_{reg}$  sind in Tabelle 7.1 aufgeführt. Die Parameterschätzung wurde mit

Parameter	Startwerte	Optimale Werte	Reg.werte
$A_0$ [mol/(kg s)]	8.00e+2	$9.25e+2 \pm 3.92e+2$	8.00e+2
$E_a$ [KJ/mol]	5.00e+1	$4.86e+1 \pm 0.22e+1$	5.00e+1
$k_{0,Des}$ [mol/(kg s)]	0.62e-4	$2.18e-4 \pm 0.39e-4$	1.32e-4
$A_{0,Des}$ [mol/(kg s)]	0.94e-3	$3.98e-3 \pm 1.43e-3$	5.66e-3
$E_{a,Des}$ [KJ/mol]	1.04e-1	$4.75e-1 \pm 1.61e-1$	7.58e-1

Tabelle 7.1: Übersicht der Kinetik-Parameterwerte

der in Kapitel 3 vorgestellten reduzierten Gauß-Newton-Variante des reduzierten SQP-Algorithmus gelöst. Die optimalen Parameterwerte aus der Parameterschätzung bewirken einen deutlichen besseren Verlauf der Temperaturprofile - siehe Abbildung 7.3. Der Temperatur-Peak wurde sowohl in der Höhe als auch in der axialen Lage wesentlich besser angepaßt, was auch für die Flanken des Temperatur-Peaks eingeschränkt gilt - nur die Restaktivität vor dem Peak ist noch zu hoch und damit auch die Temperatur. Untersuchungen mit anderen Modellierungsansätzen für die Aktivität, die das beschriebene Phänomen der erhöhten Restaktivität beheben, sind geplant. Insgesamt wurde die LSQ-Zielfunktion deutlich reduziert - von 0.800 mit Parameter-Startwerten auf 0.089 mit optimalen Parameterwerten aus der Schätzung (jeweils ohne Regularisierungsterm). Die Startwerte und optimalen Werte der Kinetik-Parameter zusammen mit Standardabweichungen sind in Tabelle 7.1 angegeben.

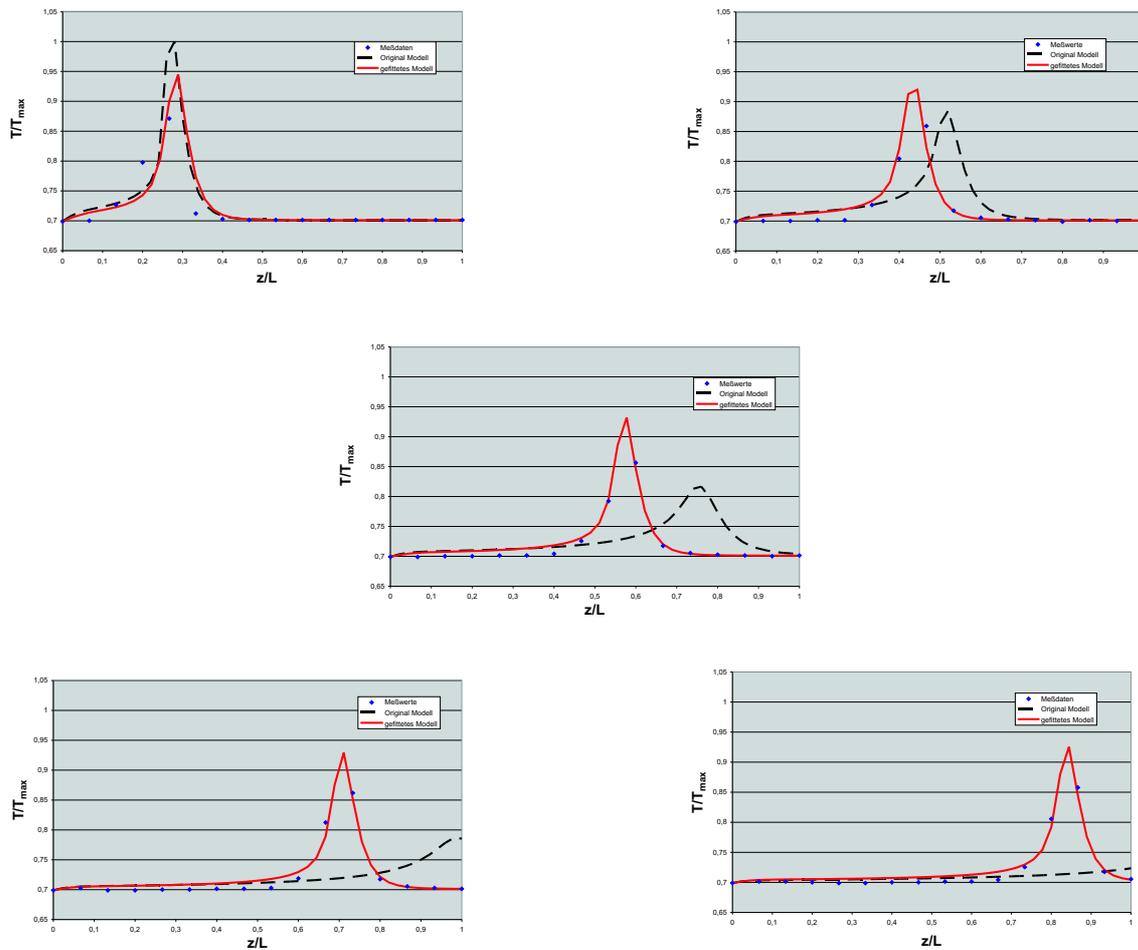


Abbildung 7.3: Temperaturprofile in der Rohrmitte zu den Zeitpunkten  $t = \{0.2, 0.4, 0.6, 0.8, 1.0\} t_c$

## 7.4 Optimierungsszenarien

Durch optimale Steuerung des Durchsatzes sowie durch optimale Wahl der Dauer der Reaktionszyklen im Verhältnis zur festen Regenerationszeit soll die Gesamtproduktivität des Reaktors maximiert werden. Dabei muß eine vorgegebene obere Temperaturgrenze  $T_{lim}$  im Festbett sicher eingehalten werden, um eine irreversible Schädigung des Katalysators zu vermeiden. Zur Absenkung der Hot-Spot-Temperatur kann als weiterer Optimierungsparameter die „Verdünnung“ des Katalysatormaterials betrachtet werden. Die Reaktion im Reaktor wird abgebrochen, wenn eine Durchbruchbedingung ( $c_A(t_c, L, r) < 0,01 \cdot c_{A, ein}$ ) verletzt wird, das heißt, daß die Reaktanden im Rohr nicht mehr vollständig umgesetzt werden und somit das Produkt am Reaktorausgang verunreinigen.

Ausgehend von dieser Zielvorgabe wurden zwei Optimierungsszenarien entworfen: das erste mit einer festen „Verdünnung“ des Katalysators von 100 % (Vergleichsszenario) und das zweite mit variabler globaler „Verdünnung“ des Katalysators (optimale Betriebswei-

se). Als Steuerung dient hierbei der zeitvariable Durchsatz  $c_{A, \text{ein}}$ , der stückweise linear parameterisiert wurde. Die Maximierung der im Jahr durchgesetzten Gesamtmenge (Zielfunktion) dient als Kriterium für die Gesamtproduktivität des Reaktors. Dabei wechseln die Reaktionsphasen (freie Dauer  $t_c$ ) mit den Regenerationsphasen des Katalysators (mit fester Dauer  $t_{\text{reg}}$ ) zyklisch ab, wobei insgesamt eine feste Produktionszeit  $t_{\text{prod}}$  im Jahr zur Verfügung steht. Damit wird die Zielfunktion wie folgt formuliert:

$$\frac{t_{\text{prod}}}{t_c + t_{\text{reg}}} \int_{t=0}^{t_c} c_{A, \text{ein}} dt \quad (7.23)$$

Die Berechnungen beider Szenarien wurde dabei mit Hilfe der in Kapitel 3 vorgestellten BFGS-Variante des reduzierten SQP-Algorithmus gelöst. Dabei wurde ein Mehrstufenansatz verwendet: die erste Stufe (Anfahrvorgang) war von fester kurzer Dauer (mit 5 Mehrzielintervallen) während die zweite Stufe (Produktionsphase) von variabler Dauer (mit 10 Mehrzielintervallen) war. Zusammen mit der festen Anfangssteuerung verbessert dies die numerische Stabilität des Optimierungsalgorithmus erheblich!

In der Abbildung 7.4 sind die optimierten normierten Durchsätze beider Szenarien über

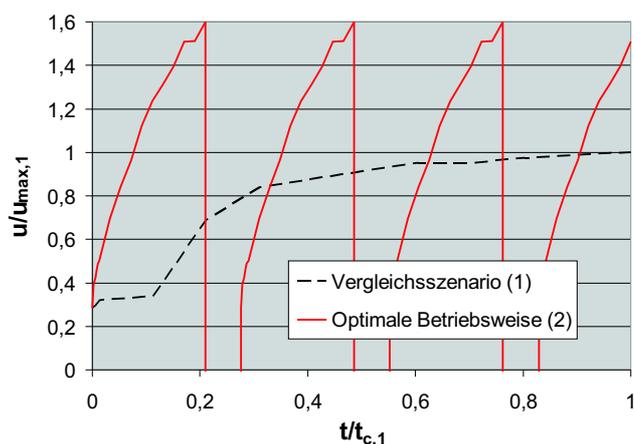


Abbildung 7.4: Optimaler Durchsatz für Vergleichsszenario und optimale Betriebsweise

die normierte Zeit geplottet. Die Normierung wurde dabei bzgl. dem maximalen Wert des Durchsatzes bzw. der Dauer der Reaktionsphase des Vergleichsszenarios durchgeführt. In Abbildung 7.5 sind die optimalen Temperaturprofile in der Rohrmitte bzgl. der optimalen Betriebsweise des Reaktors gezeigt. Ein oberes Temperaturlimit  $T_{\text{lim}}$  wird nahezu überall eingehalten. Auch ist die aktive Durchbruchbedingung am Ende des Reaktors und der Reaktionsphase am beginnenden Anstieg der Temperatur am Rohrende (exotherme Reaktion) zu erkennen. Die optimale Betriebsweise des Rohrreaktors (gegenüber dem Vergleichsszenario mit fester „Verdünnung“ des Katalysators) wird durch eine deutliche Reduktion der „Verdünnung“ des Katalysators erreicht. Die optimale Verdünnung beträgt  $\sim 50\%$ . Die Gesamtkapazität konnte gegenüber dem Vergleichsszenario von  $4.601e+3$  kg um  $\sim 12\%$  auf  $5.132e+3$  kg bei der optimalen Betriebsweise erhöht werden.

Dieses Ergebnis ist in zwei Punkten bemerkenswert: die Ausbeute wird gesteigert (Gesamtkapazität) und die Kosten gesenkt (Katalysatorgehalt). Insbesondere letzterer Punkt bringt bei vielen Reaktionen eine enorme Profitsteigerung.

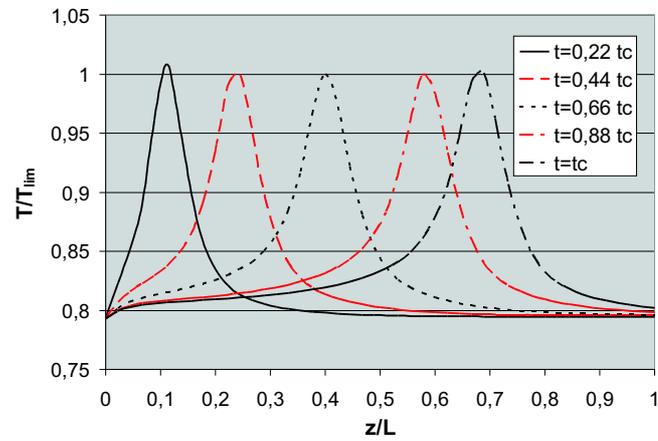


Abbildung 7.5: Optimale Temperaturprofile in der Rohrmitte

# Kapitel 8

## Zusammenfassung und Ausblick

In dieser Arbeit wurden reduzierte Gauß-Newton- und Quasi-Newton-Ansätze im Rahmen des direkten Mehrzielansatzes vorgestellt. Im Hinblick auf die diskutierten Anwendungsbeispiele werden die entwickelten Methoden nachfolgend zusammengefaßt und mögliche Weiterentwicklungen aufgezeigt.

### Zusammenfassung

Der direkte Mehrzielansatz angewendet auf dynamische Optimierungsprobleme mit großen nichtlinearen DAE-Prozeßmodellen vom Index 1 ergibt große strukturierte nichtlineare Probleme (siehe Kapitel 1). Als Methoden zur iterativen Lösung dieser NLPs wurden für Parameterschätzprobleme ein Gauß-Newton-Verfahren und für Optimal-Steuerungsprobleme ein Quasi-Newton-Verfahren vorgestellt (siehe Kapitel 2). Diese Newton-ähnliche Verfahren wurden durch einen Reduktionsansatz (bzgl. differentieller und algebraischer Variablen) beschleunigt, indem Richtungsableitungen direkt berechnet wurden, was sich insbesondere auf die Berechnung der Koordinaten-Basis auswirkt (siehe Kapitel 3 und 4). Dabei fand eine Verknüpfung der Linearisierung der Nebenbedingungen und deren Projektion statt, so daß die Anzahl der Richtungsableitungen der Variationsdifferentialgleichung unabhängig von der Dimension der Zustandsvariablen wird (siehe Kapitel 4). Dieses erst ermöglicht die Berechnung von Lösungen von dynamischen Optimierungsproblemen mit mehrdimensionalen instationären PDEs. Diese wurden durch Anwendung der Linienmethode in DAE-Systeme vom Index 1 überführt. Sowohl die Zeitdiskretisierung im Integrator als auch die Integratorgenauigkeit wurden während der Optimierungsiterationen adaptiv gewählt. Bei der Berechnung der Richtungsableitungen wurde das Prinzip der Internen Numerischen Differentiation eingehalten und mit Methoden der Automatischen (bzw. Symbolischen) Differentiation verbunden (siehe Kapitel 4). Durch Seed-Matrix-Kompression konnte gezeigt werden, daß der Aufwand für die Iterationsmatrizen des Integrators nur linear in der Feinheit der Ortsdiskretisierung bei 1D-PDEs ist (siehe Anwendung im Kapitel 5 für zentrale Differenzenquotienten der Ortsableitungen auf äquidistantem Gitter). Weiterhin existieren effiziente Schnittstellen zu Residuen- und Jacobimatrixauswertung des Modells: über die ESO-Schnittstelle bei gPROMS-Modellen, über Adol-C und Seed-Matrix-Komprimierung nach Curtis-Powell-Reid bei C- und C++-Modellen und über Finite Differenzen und CPR-Seedmatrix-Komprimierung allgemein.

Zur Globalisierung der Newton-ähnlichen Verfahren wurden für den Gauß-Newton-Fall der Restriktive Monotonietest und für den Quasi-Newton-Fall eine partiell multiplikatorfreie Strategie angewendet (siehe Kapitel 3).

Anhand der Parameterschätzung bei der Wirkstoff-Freisetzung eines Zahnfleisch-Implantats mit instationärem 1D-PDE Modell (siehe Kapitel 5) zeigt sich, daß der Gesamtaufwand der neuen reduzierten Variante nicht mehr quadratisch sondern nahezu linear von der Feinheit der Ortsdiskretisierung abhängt. Ein Vergleich der Modell-Implementierungen (gPROMS und C/Adol-C) bestätigt, daß die Variante mit Automatischer Differentiation (C/Adol-C) bis um den Faktor 5 schneller ist als die Variante mit Symbolischer Differentiation (gPROMS).

Anhand eines Optimierungsproblems bei einer Destillationskolonne wird gezeigt, daß die neue reduzierte Variante sowohl im Offline- als auch im Online-Fall effizient ist (siehe Kapitel 6). Im Vergleich zur partiell reduzierten Variante in MUSCOD-II ist die Gesamt-rechenzeit der neuen reduzierten Variante um den Faktor 5 geringer. Insbesondere liegt die Zeit für die Berechnung einer Feedback-Steuerung im Zentelsekundenbereich und die Vorbereitungsphase für eine neue Optimierungs-Iteration dauert unter 5 Sekunden, wobei das Problem 2 Steuerungen und 206 Zustände besitzt.

Die optimale Betriebsweise eines katalytischen Rohrreaktors mit instationärem 2D-PDE-Modell (siehe Kapitel 7) konnte mit Hilfe des in dieser Arbeit vorgestellten Reduktionsansatzes berechnet werden, wobei zuvor eine Parameterschätzung der kinetischen Parameter (der Hauptreaktion und der Katalysatordeaktivierung) durchgeführt wurde. Damit konnte gezeigt werden, daß durch Optimierung der Katalysator-Aktivität sowohl die Kosten für den Katalysator gesenkt (50 %) als auch die Kapazität des Reaktors erhöht (12 %) werden konnte. Hierbei zeigte sich insbesondere, daß die Mehrziel diskretisierung ein wirksames Mittel ist, um die Stabilität des Verfahrens zu erhöhen und eine Optimierung überhaupt möglich zu machen.

## Ausblick

Bei der Untersuchung von Optimal-Steuerungsproblemen hat sich gezeigt, daß durch den Einsatz von guten Approximationen der Hessematrizen in SQP-Verfahren sowohl die Konvergenz beschleunigt als auch das zunehmende Verschlechtern der Kondition der Hessematrix durch Updates vermieden werden kann (siehe Schäfer 1999, [81]). Dort wurden die Hessematrizen durch finite Differenzen approximiert, wobei deren Genauigkeit durch Einhaltung des Prinzips der Internen Numerischen Differentiation auf 2-3 Stellen genau war. Für große Modelle, wie in dieser Arbeit vorgestellt, ist diese Vorgehensweise jedoch nicht effizient genug. Durch Einsatz eines zum Vorwärtsmodus passenden Rückwärtsmodus des Integrators können jedoch sowohl die Multiplikatoren als auch die einseitig reduzierte Hessematrix bis auf Integrationsgenauigkeit genau berechnet werden. Damit kann auch der Crossterm berechnet werden, der das reduzierte SQP-Verfahren „kompatibel“ zum nichtreduzierten SQP-Verfahren macht (1-Schritt quadratische Konvergenz).

Passend zu dem SQP-Verfahren mit exakten Hessematrizen (Lagrange-Newton-Verfahren) ist eine Globalisierung durch Trust Regions. In Verbindung mit Natürlichen Niveaufunk-

tionen in den primalen *und* dualen Variablen und den Restriktiven Monotonietest läßt sich ein robustes und effizientes Verfahren konstruieren (siehe Bock *et al.* 2000, [19]).

Durch das Einbeziehen von Unstetigkeiten in das Modell können neue Problemklassen dynamischer Prozesse (zum Beispiel Wehrüberläufe bei Destillationskolonnen) behandelt werden. Die korrekte Berechnung der Sensitivitäten spielt dabei eine zentrale Rolle und ist im Rahmen des Parameterschätzverfahrens PARFIT (siehe Bock 1987, [16]) und des Optimal-Steuerungsverfahrens MUSCOD-II (siehe Brandt-Pollmann 2004, [22]) implementiert worden. Eine Kombination der Unstetigkeitsbehandlung mit den Methoden dieser Arbeit erweitert die Klasse effizient behandelbarer Optimierungsprobleme auf solche mit wenigen Freiheitsgraden.

Weitere Anwendungsfelder für die in dieser Arbeit vorgestellten Methoden sind z. B.:

Die Online-Optimierung eines Simulated-Moving-Bed (SMB) Prozesses zur kontinuierlichen chromatographischen Trennung von chiralen Wirkstoffen im industriellen Maßstab mit dem in dieser Arbeit (Kapitel 6) vorgestellten Echtzeit-Iterationsschema. Der SMB-Prozeß weist durch periodisches Umschalten der Zu- und Ableitungen meistens kein stationäres Konzentrationsprofil der Komponenten auf, sondern ist von periodischer Natur. Die Mehrstufenformulierung erlaubt die direkte Umsetzung dieser Problemklasse und deren effiziente Lösung (siehe Toumi *et al.* 2004a, [89] und 2004b, [88]). Die hochdimensionalen Modelle (mehrere hundert bis tausend differentielle Zustände und wenige Parameter) sind stark nichtlinear, so daß sich die Mehrzielparameterisierung vorteilhaft nutzen läßt.

Die Optimierung der ortszeitgebundenen Musterbildung und Wellenausbreitung beim Vorgang der bakteriellen Chemotaxis durch optimale Steuerung des Randflusses der chemotaktisch aktive Substanz stellt ein hochdimensionales Optimierungs-Randwertproblem mit mehreren hundert bis tausend differentiellen Zuständen und einer Steuerung dar (siehe Lebedz *et al.* 2003, [55]). Aufgrund fester Anfangswerte liegen in diesem Problem nur wenige Freiheitsgrade vor und somit lassen sich mit den in dieser Arbeit vorgestellten Methoden sowohl bei der Offline- als auch Online-Optimierung erhebliche Performance-Gewinne gegenüber MUSCOD-II erzielen.

# Anhang A

## Zusätzliche Informationen

### A.1 Zahnfleisch-Implantat

Symbol	Beschreibung	Wert
$L$	Dicke der Membran-Scheibe	0.02 [cm]
$D$	Diffusionsparameter	2.364e-5 [cm <sup>2</sup> /s]
$B_{max}$	Bindungskapazität Wirkstoff-Gelatine	2.054 [mol/mol]
$k_{cat}$	Michaelis-Menten-Parameter	1.654e+3 [h <sup>-1</sup> ]
$E_0$	Enzym-Konzentration	1.000e-5 [M]
$K_M$	Gleichgewichtskonstante des Enzym-Substrat-Komplex	4.000e-2 [M]
$K_i$	Gleichgewichtskonstante des Enzym-Inhibitor-Komplex	1.967e-3 [M]
$\beta$	Quotient $K_M/K_i$	2.032e+1 [-]
$k_a$	Wirkstoff-Gelatine Assoziationskoeffizient	1.045e+4 [M <sup>-1</sup> h <sup>-1</sup> ]
$k_d$	Abbaurrate des Gelatine gebundenen Wirkstoffs	7.816e-2 [h <sup>-1</sup> ]
$K$	Quotient $k_a/k_d$	1.337e+5 [M <sup>-1</sup> ]
$C_{load}$	anfängliche Wirkstoff-Konzentration	2.800e-1 [M]
$\rho_0$	anfängliche Gelatine-Konzentration	0.045 [M]

Tabelle A.1: Konstanten im PerioChip-Modell

Zeitpunkt [h]	gemessene Freisetzungsrage [-]	Standardabweichung [-]
1.0	0.232	0.042
2.0	0.298	0.010
4.0	0.351	0.027
8.0	0.352	0.025
24.0	0.402	0.028
48.0	0.531	0.035
72.0	0.595	0.026
96.0	0.624	0.018
120.0	0.669	0.030
144.0	0.753	0.053
168.0	0.867	0.042
192.0	0.938	0.023

Tabelle A.2: Meßwerte für Parameterschätzung beim PerioChip-Modell

## A.2 Kontinuierliche Destillationskolonne

$k$	$a_k$ [kmol l <sup>-1</sup> ]	$b_k$	$c_k$ [K]	$d_k$
1 (Methanol)	2.288	0.2685	512.4	0.2453
2 (n-Propanol)	1.235	0.27136	536.4	0.2400

Tabelle A.3: Molare Volumen-Koeffizienten im Destillationskolonnenmodell

$k$	$A_k$	$B_k$ [K]	$C_k$ [K]
1 (Methanol)	23.48	3626.6	-34.29
2 (n-Propanol)	22.437	3166.4	-80.15

Tabelle A.4: Antoine-Koeffizienten für partielle Drücke

$k$	$h_{1,k}$ [K <sup>-1</sup> ]	$h_{2,k}$ [K <sup>-2</sup> ]	$h_{3,k}$ [K <sup>-3</sup> ]	$T_k^c$ [K]	$P_k^c$ [Pa]	$\Omega_k$
1	18.31	$1.713 \cdot 10^{-2}$	$6.399 \cdot 10^{-5}$	512.6	$8.096 \cdot 10^6$	0.557
2	31.92	$4.49 \cdot 10^{-2}$	$9.663 \cdot 10^{-5}$	536.7	$5.166 \cdot 10^6$	0.612

Tabelle A.5: Enthalpie-Koeffizienten

Symbol	Wert	Symbol	Wert
$n_0^v$	8.5 [l]	$P_{\text{top}}$	939 [h Pa]
$n_{N+1}^v$	0.17 [l]	$\Delta P_{0,\dots,N_F-1}$	2.5 h [Pa]
$n_{\text{tray}}^{\text{ref}}$	0.155 [l]	$\Delta P_{N_F,\dots,N}$	1.9 [h Pa]
$\alpha_{1,\dots,N_F}$	62 %	$T_F$	71 [°C]
$\alpha_{N_F+1,\dots,N}$	35 %	$T_C$	47.2 [°C]
$W_{1,\dots,N}$	0.166 [l <sup>-1/2</sup> s <sup>-1</sup> ]	$F_{\text{vol}}$	14.0 [l h <sup>-1</sup> ]
$Q_{\text{loss}}$	0.51 [kW]	$X_F$	0.32 [-]

Tabelle A.6: System-Parameter

## A.3 Katalytischer Rohrreaktor

Symbol	Beschreibung	Maßeinheit
$a_{Kat}$	Aktivität des Katalysators	$[-]$
$A_{Rohr}$	Mantelfläche des Rohres/Rohrbündels	$[m^2]$
$c$	Konzentration	$[kg/mol]$
$c_p$	isobare Wärmekapazität	$[J/(kg \cdot K)]$
$d_p$	Partikeldurchmesser des Katalysators	$[m]$
$D$	Rohrdurchmesser	$[m]$
$D_{eff}$	effektiver Diffusionskoeffizient	$[m/s^2]$
$E_a$	Aktivierungsenergie	$[J]$
$\Delta H_R$	Bildungsenthalpie	$[J/mol]$
$k$	Reaktionsgeschwindigkeitskonstante	$[mol/(kg \cdot s)]$
$k_0$	Reaktionsgeschwindigkeitskonstante	$[mol/(kg \cdot s)]$
$k_\alpha$	Wärmedurchgangskoeffizient	$[W/(m^2 \cdot K)]$
$K$	charakteristische Größe für die Dispersion	$[-]$
$K_\infty$	char. Größe K bei unendlicher Ausdehnung der Schüttung	$[-]$
$M$	Molare Masse	$[mol]$
$\dot{m}$	Massenstrom	$[kg/h]$
$p$	Druck	$[Pa]$
$r$	radiale Koordinate	$[m]$
$R$	Rohrradius	$[m]$
$r_0$	Reaktionsgeschwindigkeit	$[mol/(kg \cdot s)]$
$s$	Wandstärke	$[m]$
$t$	Zeit	$[s]$
$T$	Temperatur	$[K]$
$u$	Leerrohrgeschwindigkeit	$[m/s]$
$\dot{V}$	Volumenstrom	$[m^3/h]$
$y$	Molanteil	$[-]$
$z$	axiale Koordinate	$[m]$

Tabelle A.7: Lateinische Symbole im Rohrreaktor-Modell

Symbol	Beschreibung	Maßeinheit
$\varepsilon$	Porosität	$[-]$
$\rho_{s,bed}$	Schüttdichte Feststoff	$[kg/m^3]$
$\rho$	Dichte	$[kg/m^3]$
$\nu$	kinematische Viskosität	$[m^2/s]$
$\lambda_{eff}$	effektive Wärmeleitfähigkeit	$[W/(m \cdot K)]$
$\lambda$	Wärmeleitfähigkeit	$[W/(m \cdot K)]$
$\theta$	Parameter zur Berücksichtigung sekundärer Effekte bei der Wärmeleitung	$[-]$
$\alpha_W$	Wärmeübergangskoeffizient an der Wand	$[W/(m^2 \cdot K)]$
$\alpha$	Wärmeübergangskoeffizient	$[W/(m^2 \cdot K)]$
$\eta$	dynamische Viskosität	$[Pa \cdot s]$

Tabelle A.8: Griechische Symbole im Rohrreaktor-Modell

Indices	Beschreibung
<i>außen</i>	Außenseite des Rohres
<i>bed</i>	Festbett/Schüttung
<i>Des</i>	Desaktivierung
<i>ein</i>	am Rohreintritt
<i>f</i>	Gas
<i>i</i>	Laufvariable für Komponenten
<i>innen</i>	Innenseite des Rohres
<i>Kat</i>	Katalysator
<i>KM</i>	Kühlmittel
<i>MR</i>	Mantelraum
<i>r</i>	in radialer Richtung
<i>s</i>	Feststoff
<i>z</i>	in axialer Richtung

Tabelle A.9: Indices im Rohrreaktor-Modell

Kennzahl	Beschreibung
$Pe = \frac{u_f \cdot d_p \cdot \rho_f \cdot c_{p,f}}{\lambda_f}$	Peclet-Zahl
$Pr = \frac{c_{p,f} \cdot \eta_f}{\lambda_f}$	Prandtl-Zahl
$Re_p = \frac{u_f \cdot d_p \cdot \rho_f}{\eta_f}$	Reynolds-Zahl

Tabelle A.10: Dimensionslose Kennzahlen im Rohrreaktor-Modell

Axiale Position $z/L$	Reaktionsdauer				
	$0.2 t_c$	$0.4 t_c$	$0.6 t_c$	$0.8 t_c$	$t_c$
0.000	0.699	0.699	0.699	0.699	0.699
0.067	0.700	0.700	0.699	0.703	0.701
0.133	0.727	0.700	0.700	0.699	0.701
0.200	0.798	0.701	0.700	0.699	0.700
0.267	0.871	0.701	0.701	0.700	0.699
0.333	0.712	0.727	0.701	0.700	0.699
0.400	0.703	0.804	0.704	0.701	0.700
0.467	0.701	0.859	0.726	0.701	0.700
0.533	0.701	0.718	0.792	0.703	0.701
0.600	0.701	0.705	0.856	0.719	0.701
0.667	0.701	0.703	0.718	0.812	0.704
0.733	0.701	0.701	0.705	0.862	0.726
0.800	0.701	0.699	0.703	0.718	0.806
0.867	0.701	0.701	0.701	0.705	0.858
0.933	0.701	0.700	0.700	0.703	0.718
1.000	0.701	0.701	0.701	0.701	0.705

Tabelle A.11: Normierte Temperatur-Meßwerte  $T/T_{max}$  in der Rohrmitte eines Rohrreaktors

# Literaturverzeichnis

- [1] R. Adler. Stand der Simulation von heterogen-gaskatalytischen Reaktionsabläufen in Festbettrohrreaktoren – Teil 1. *Chemie Ingenieur Technik*, 6:555–564, 2000.
- [2] R. Adler. Stand der Simulation von heterogen-gaskatalytischen Reaktionsabläufen in Festbettrohrreaktoren – Teil 2. *Chemie Ingenieur Technik*, 7:688–699, 2000.
- [3] H. D. Baehr and E. h. K. Stephan. *Wärme- und Stoffübertragung*. Springer-Verlag Berlin Heidelberg, 2003.
- [4] V. Bär. Ein Kollokationsverfahren zur numerischen Lösung allgemeiner Mehrpunkt-randwertaufgaben mit Schalt- und Sprungbedingungen mit Anwendung in der Optimalen Steuerungen und der Parameteridentifizierung. Master’s thesis, Universität Bonn, 1983.
- [5] P. I. Barton and C. C. Pantelides. Modelling of combined discrete/continuous processes. *AIChE J.*, 40:966–979, 1994.
- [6] I. Bauer. Numerische Behandlung Differentiell-Algebraischer Gleichungen mit Anwendungen in der Chemie. Master’s thesis, Universität Augsburg, 1994.
- [7] I. Bauer. *Numerische Verfahren zur Lösung von Anfangswertaufgaben und zur Generierung von ersten und zweiten Ableitungen mit Anwendungen bei Optimierungsaufgaben in Chemie und Verfahrenstechnik*. PhD thesis, University of Heidelberg, 1999.
- [8] M. Bauer. *Theoretische und experimentelle Untersuchungen zum Wärmetransport in gasdurchströmten Festbettrohrreaktoren*. PhD thesis, Martin-Luther-Universität Halle-Wittenberg, 2001.
- [9] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: basic analysis and examples. *East-West Journal Numer. Math.*, 4:237–264, 1996.
- [10] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. In A. Iserles, editor, *Acta Numerica 2001*. Cambridge University Press, Cambridge, 2001.
- [11] L. T. Biegler. Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. *Comput. Chem. Engng.*, 8:243–248, 1984.

- [12] L. T. Biegler, C. Schmid, and D. J. Ternet. A multiplier-free, reduced hessian method for process optimization. Technical report, Engineering Design Research Center, Canegie Mellon University, Pittsburgh, 1995.
- [13] R. B. Bird, W. E. Stewart, and E. N. Lightfoot. *Transport Phenomena*. John Wiley & Sons, Inc., 2002.
- [14] H. G. Bock. Numerical treatment of inverse problems in chemical reaction kinetics. In K. H. Ebert, P. Deuffhard, and W. Jäger, editors, *Modelling of Chemical Reaction Systems (Springer Series in Chemical Physics 18)*. Springer, Heidelberg, 1981.
- [15] H. G. Bock. Recent advances in parameter identification techniques for ODE. In P. Deuffhard and E. Hairer, editors, *Numerical Treatment of Inverse Problems in Differential and Integral Equations*. Birkhäuser, Boston, 1983.
- [16] H. G. Bock. Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen. In *Bonner Mathematische Schriften 183*. Universität Bonn, 1987.
- [17] H. G. Bock, I. Bauer, D. B. Leineweber, and J. P. Schlöder. Direct multiple shooting methods for control and optimization of dae in chemical engineering. In F. Keil, W. Mackens, H. Voß, and J. Werther, editors, *Scientific Computing in Chemical Engineering II*, volume 2, pages 2–18, Springer, Berlin, 1999.
- [18] H. G. Bock, E. Eich, and J. P. Schlöder. Numerical solution of constrained least squares boundary value problems in differential-algebraic equations. In K. Strehmel, editor, *Numerical Treatment of Differential Equations*. Teubner, Leipzig, 1988.
- [19] H. G. Bock, E. A. Kostina, and J. P. Schlöder. On the role of natural level functions to achieve global convergence for damped newton methods. In M. Powell et al., editor, *System Modelling and Optimization. Methods, Theory and Applications*. Kluwer, 2000.
- [20] H. G. Bock and K.-J. Plitt. A multiple shooting algorithm for direct solution of optimal control problems. In *Proceedings of the 9th IFAC World Congress, Budapest*. Pergamon Press, 1984.
- [21] P. T. Boggs and J. W. Tolle. Convergence properties of a class of rank-two updates. *SIAM Journal on Optimization*, 4(2):262–287, 1994.
- [22] U. Brandt-Pollmann. *Numerical solution of optimal control problems with implicitly defined discontinuities - Applications to engineering*. PhD thesis, Ruprecht-Karls-Universität Heidelberg, 2004.
- [23] B. L. Braunschweig, C. C. Pantelides, H. I. Britt, and S. Sama. Process modeling: the promise of open software architectures. *Chemical Engineering Progress*, 96(9):65–76, 2000.

- [24] R. M. Chamberlain, M. J. D. Powell, C. Lemarechal, and H. C. Pedersen. The watchdog technique for forcing convergence in algorithms for constrained optimization. *Mathematical Programming*, 16:1–17, 1982.
- [25] A. R. Curtis, M. J. D. Powell, and J. K. Reid. On the estimation of sparse Jacobian matrices. *J. Inst. Math. Appl.*, 13:117–119, 1974.
- [26] J. E. Cuthrell and L. T. Biegler. Simultaneous optimization and solution methods for batch reactor profiles. *Computers chem. Engng*, 13(1/2):49–62, 1989.
- [27] J. E. Dennis Jr. and J. J. Moré. Quasi-Newton methods, motivation and theory. *SIAM Review*, 19(1):46–89, 1977.
- [28] M. Diehl. *Real-Time Optimization for Large Scale Nonlinear Processes*, volume 920 of *Fortschr.-Ber. VDI Reihe 8, Meß, Steuerungs- und Regelungstechnik*. VDI Verlag, Düsseldorf, 2002.
- [29] M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer. Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *Journal on Process Control*, 12(4):577–585, 2002.
- [30] M. Diehl, D. B. Leineweber, A. Schäfer, H. G. Bock, and J. P. Schlöder. Optimization of multiple-fraction batch distillation with recycled waste cuts. *AIChE Journal*, 48(12):2869–2874, 2002.
- [31] M. Diehl, I. Uslu, R. Findeisen, S. Schwarzkopf, F. Allgöwer, H. G. Bock, T. Bürner, E. D. Gilles, A. Kienle, J. P. Schlöder, and E. Stein. Real-time optimization for large scale processes: Nonlinear model predictive control of a high purity distillation column. In M. Grötschel, S. O. Krumke, and J. Rambau, editors, *Online Optimization of Large Scale Systems: State of the Art*, pages 363–384. Springer, 2001.
- [32] M. F. Edwards and J. F. Richardson. Gas Dispersion in Packed Beds. *Chem. Eng. Science*, 23:109–123, 1968.
- [33] E. Eich. Numerische Behandlung semi-expliziter differentiell-algebraischer Gleichungssysteme vom Index 1 mit BDF-verfahren. Master’s thesis, Universität Bonn, 1987.
- [34] Process Systems Enterprise. gPROMS Advanced User Guide, 2004.
- [35] Process Systems Enterprise. gPROMS System Programmer Guide, 2004.
- [36] R. Fletcher. Second order corrections for nondifferentiable optimization. In A. Watson, editor, *Numerical Analysis, Dundee 1981*. Springer, Berlin, 1982.
- [37] R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, 1987.
- [38] R. Fletcher and S. Leyffer. Nonlinear programming without a penalty function. *Mathematical Programming*, 91(2):239–269, 2002.

- [39] P. Forzatti and L. Lietti. Catalyst Deactivation. *Catalysis Today*, 52:165–181, 1999.
- [40] D. Gabay. Reduced Quasi-Newton methods with feasibility improvement for nonlinear constrained optimization. *Mathematical Programming Study*, 16:18–44, 1982.
- [41] J. Gauvin. A necessary and sufficient regularity condition to have bounded multipliers in nonconvex programming. *Mathematical Programming*, 27:136–138, 1977.
- [42] P. E. Gill, L. O. Jay, M. W. Leonard, L. R. Petzold, and V. Sharma. An SQP method for the optimal control of large-scale dynamical systems. *Journal Comp. Appl. Math.*, 120:197–213, 2000.
- [43] A. Griewank. *Evaluating Derivatives. Principles and Techniques of Algorithmic Differentiation*. SIAM, Philadelphia, 2000.
- [44] A. Griewank, D. Juedes, H. Mitev, J. Utke, O. Vogel, and A. Walther. ADOL-C: A package for the automatic differentiation of algorithms written in C/C++. *ACM TOMS*, 22(2):131–167, 1996.
- [45] Ch. Großmann and H.-G. Roos. *Numerik partieller Differentialgleichungen*. Teubner, Stuttgart, 1992.
- [46] D. J. Gunn and C. Pryce. Dispersion in Packed Beds. *Trans. Inst. Chem. Eng.*, 47:341–350, 1969.
- [47] S. P. Han. A globally convergent method for nonlinear programming. *Journal of Optimization Theory and Applications*, 22(3):297–309, 1977.
- [48] G. A. Hicks and W. H. Ray. Approximation methods for optimal control systems. *Can. J. Chem. Engng*, 49:522–528, 1971.
- [49] H. Hinsberger. *Ein direktes Mehrzielverfahren zur Lösung von Optimalsteuerungsproblemen mit großen, differential-algebraischen Gleichungssystemen und Anwendungen aus der Verfahrenstechnik*. PhD thesis, Technical University of Clausthal, 1998.
- [50] H. Hinsberger, S. Miesbach, and H. J. Pesch. Optimal temperature control of semibatch polymerization reactors. In F. Keil, W. Mackens, H. Voß and J. Werther, editors, *Scientific Computing in Chemical Engineering*, Springer, Heidelberg, 1996.
- [51] W. Karush. Minima of functions of several variables with inequalities as side conditions. Master’s thesis, University of Chicago, Chicago IL, 1939.
- [52] D. Kraft. On converting optimal control problems into nonlinear programming problems. In K. Schittkowski, editor, *Computational Mathematical Programming*, volume F15 of *NATO ASI*, pages 261–280. Springer, 1985.
- [53] H. W. Kuhn and A. W. Tucker. Nonlinear programming. In *Proc. Second Berkeley Symp. Mathematics, Statistics, and Probability*. University of California Press, 1951.

- [54] F.-S. Kupfer. *Reduced successive quadratic programming in Hilbert space with applications to optimal control*. PhD thesis, Universität Trier, 1992.
- [55] D. Lebiedz and U. Brandt-Pollmann. Manipulation of Self-Aggregation Pattern and Waves in a Reaction-Diffusion System by Optimal Boundary Control Strategies. *Physical Review Letters*, 91(20), 2003.
- [56] D. B. Leineweber. Analyse und Restrukturierung eines Verfahrens zur direkten Lösung von Optimal-Steuerungsproblemen (The theory of MUSCOD in a nutshell). Master's thesis, Ruprecht-Karls-Universität Heidelberg, 1995.
- [57] D. B. Leineweber. *Efficient reduced SQP methods for the optimization of chemical processes described by large sparse DAE models*, volume 613 of *Fortschr.-Ber. VDI Reihe 3, Verfahrenstechnik*. VDI Verlag, Düsseldorf, 1999.
- [58] D. B. Leineweber, I. Bauer, H. G. Bock, and J.P. Schlöder. An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part I: Theoretical aspects. *Computer & Chemical Engineering*, 27:157–166, 2003.
- [59] D. B. Leineweber, A. Schäfer, H. G. Bock, and J. P. Schlöder. An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part II: Software aspects and applications. *Computer & Chemical Engineering*, 27:167–174, 2003.
- [60] S. Li, L. R. Petzold, and J. M. Hyman. Solution Adapted Mesh Refinement and Sensitivity Analysis for Parabolic Partial Differential Equation Systems. In L. Biegler, O. Ghattas, M. Heinkenschloss, and B. van Bloemen Waanders, editors, *Large-Scale PDE-Constraint Optimization*, pages 117–132, 2003.
- [61] J. S. Logsdon and L. T. Biegler. Decomposition strategies for large-scale dynamic optimization problems. *Chemical Engineering Science*, 47(4):851–864, 1992.
- [62] H. Martin and N. Nilles. Radiale Wärmeleitung in durchströmten Schüttungsrohren. *Chemie Ingenieur Technik*, 65(12):1468–1477, 1993.
- [63] D. Q. Mayne and E. Polak. A superlinearly convergent algorithm for constrained optimization problems. *Mathematical Programming*, 16:45–61, 1982.
- [64] K. R. Morison and R. W. H. Sargent. Optimization of multistage processes described by differential-algebraic equations. In J. P. Hennart, editor, *Numerical Analysis, Lecture Notes in Mathematics 1230*, Springer, Berlin, 1986.
- [65] G. Nagel and R. Adler. Anwendung der Kollokationsmethode auf die Berechnung polytroper Rohrreaktoren. *Chemie Technik*, 23(6):335–341, 1971.
- [66] G. N. Newsam and J. D. Ramsdell. Estimation of sparse Jacobian matrices. *SIAM J. Algebraic Discrete Methods*, 4:404–417, 1983.
- [67] J. Nocedal. Updating Quasi-Newton matrices with limited storage. *Mathematics of Computation*, 35:773–782, 1980.

- [68] U. Nowak. Adaptive Linienmethoden für nichtlineare parabolische Systeme in einer Raumdimension. Technical Report TR 93-14, Konrad-Zuse-Zentrum für Informationstechnik, Berlin, 1993.
- [69] U. Nowak, A. Grah, and M. Schreier. Parameter Estimation and Accuracy Matching Strategies for 2-D Reactor Models. Technical Report TR 03-52, Konrad-Zuse-Zentrum für Informationstechnik, Berlin, 2003.
- [70] C. C. Pantelides. *Symbolic and numerical techniques for the solution of large systems of nonlinear algebraic equations*. PhD thesis, Imperial College of Science and Technology, London, 1988.
- [71] C. C. Pantelides. Numerical Methods for the Solution of PDAE systems. Internal Report, 2001.
- [72] L. Petzold, J. B. Rosen, P. E. Gill, L. O. Jay, and K. Park. Numerical optimal control of parabolic PDEs using DASOPT. In Biegler, Coleman, Conn, and Santosa, editors, *Large Scale Optimization with Applications, Part II*. Springer, 1997.
- [73] K. J. Plitt. Ein superlinear konvergentes Mehrzielverfahren zur direkten Berechnung beschränkter optimaler Steuerungen. Master's thesis, University of Bonn, 1981.
- [74] E. Polak. *Optimization - Algorithms and Consistent Approximations*. Springer, New York, 1997.
- [75] M. J. D. Powell. The convergence of variable metric methods for nonlinearly constrained optimization calculations. In O. L. Mangasarian, R. R. Meyer, and S. M. Robinson, editors, *Nonlinear Programming 3*, pages 27–63, Academic Press, New York, 1978.
- [76] M. J. D. Powell. A fast algorithm for nonlinearly constrained optimization calculations. In G. A. Watson, editor, *Numerical Analysis*, number 630 in Lect. Not. in Math., pages 144–157, 1978.
- [77] M. J. D. Powell. Variable metric methods for constrained optimization. In *Nonlinear Optimization: Theory and Algorithms*. Birkhäuser, Boston, 1980.
- [78] J. K. Reid and I. S. Duff. MA48, A Fortran code for direct solution of sparse unsymmetric linear systems of equations. Technical Report RAL 93 072, Rutherford Appleton Laboratory, 1993.
- [79] R. T. Rockafellar. Augmented Lagrange multiplier functions and duality in nonconvex programming. *SIAM Journal on Control*, 12:268–285, 1974.
- [80] R. W. H. Sargent and G. R. Sullivan. The development of an efficient optimal control package. In J. Stoer, editor, *Proceedings of the 8th IFIP Conference on Optimization Techniques (1977), Part 2*, Springer, Heidelberg, 1978.

- [81] A. Schäfer. Numerische Verfahren für große nichtlineare beschränkte Optimierungsprobleme und ihr Einsatz bei Optimal-Steuerungsproblemen. Master's thesis, Ruprecht-Karls-Universität Heidelberg, 1999.
- [82] J. P. Schlöder. *Numerische Methoden zur Behandlung hochdimensionaler Aufgaben der Parameteridentifizierung*, volume 187 of *Bonner Mathematische Schriften*. University of Bonn, Bonn, 1988.
- [83] V. H. Schulz. *Reduced SQP Methods for Large-Scale Optimal Control Problems in DAE with Application to Path Planning Problems for Satellite Mounted Robots*. PhD thesis, Ruprecht-Karls-Universität Heidelberg, 1996.
- [84] V. H. Schulz, H. G. Bock, and M. C. Steinbach. Exploiting invariants in the numerical solution of multipoint boundary value problems for DAEs. *SIAM J. Sci. Comp.*, 19:440–467, 1998.
- [85] P. Tanartkit and L. T. Biegler. Stable decomposition for dynamic optimization. *Ind. Eng. Chem. Res.*, 34:1253–1266, 1995.
- [86] P. Tanartkit and L. T. Biegler. A nested, simultaneous approach for dynamic optimization problems - I. *Comput. Chem. Engng*, 20:735–741, 1996.
- [87] D. J. Ternet and L. T. Biegler. Recent improvements to a multiplier-free reduced hessian SQP algorithm. Technical report, Department of Chemical Engineering, Canegie Mellon University, Pittsburgh, PA 15213, USA, 1996.
- [88] A. Toumi. *Optimaler Betrieb und Regelung von Simulated Moving Bed Prozessen*. PhD thesis, Fachbereich Bio und Chemieingenieurwesen, University of Dortmund, 2004.
- [89] A. Toumi, S. Engell, M. Diehl, H. G. Bock, and J. Schlöder. Efficient Optimization of Simulated Moving Bed Prozesses. *to be submitted*, 2004.
- [90] T. H. Tsang, D. M. Himmelblau, and T. F. Edgar. Optimal control via collocation and non-linear programming. *Int. J. Control*, 1975.
- [91] E. Tsotsas. Abschnitt Dee: Wärmeleitfähigkeit in Schüttschichten. In GVC, editor, *VDI-Wärmeatlas 8. Auflage*. Springer-Verlag, Düsseldorf, 1997.
- [92] E. Tsotsas. Abschnitt Mh: Wärmeleitung und Dispersion in durchströmten Schüttungsrohren. In GVC, editor, *VDI-Wärmeatlas 8. Auflage*. Springer-Verlag, Düsseldorf, 1997.
- [93] A. R. Tzafriri. Mathematical modeling of diffusion-mediated release from bulk degrading matrices. *Journal of Controlled Release*, 63:69–79, 2000.
- [94] A. R. Tzafriri and M. Diehl. Parameter fitting of a non-linear reaction diffusion model of drug release using a multiple shooting optimal control algorithm. Technical Report CS 2000-29, Leibniz-Center for Research in Computer Science, 2000.

- [95] A. R. Tzafriri and H. Parnas. Mathematical modeling of the in vitro drug release of the periochip implant. In *Drug Delivery for the third millennium, Pisa, Italy*, 1999.
- [96] V. S. Vassiliadis, R. W. H. Sargent, and C. C. Pantelides. Solution of a class of multistage dynamic optimization problems. Part I - problems without path constraints. Technical report, Centre for Process Systems Engineering, Imperial College of Science, Technology and Medicine, London SW7 2BY, United Kingdom, 1993.
- [97] V. S. Vassiliadis, R. W. H. Sargent, and C. C. Pantelides. Solution of a class of multistage dynamic optimization problems. Part II - problems with path constraints. Technical report, Centre for Process Systems Engineering, Imperial College of Science, Technology and Medicine, London SW7 2BY, United Kingdom, 1993.
- [98] A. Wächter and L. T. Biegler. Global and local convergence of line search filter methods for nonlinear programming. Technical report, Department of Chemical Engineering, Canegie Mellon University, Pittsburgh, 2002.
- [99] R. B. Wilson. *A simplicial Algorithm for Concave Programming*. PhD thesis, Harvard University, Cambridge, 1963.