Allgemeine lineare Verfahren für Differential-Algebraische Gleichungen mit Index 2

INAUGURAL-DISSERTATION

zur

Erlangung des Doktorgrades

der Mathematisch-Naturwissenschaftlichen Fakultät

der Universität zu Köln

vorgelegt von

Daniel Weiß

aus Siegen

Hundt Druck GmbH, Köln

2007

Berichterstatter: Prof. Dr. Johannes Schropp
Prof. Dr. Tassilo Küpper

Tag der mündlichen Prüfung: 28.02.2007

Kurzzusammenfassung

Viele Prozesse der Physik, Chemie und der Ingenieurwissenschaften können durch gewöhnliche Differentialgleichungen beschrieben werden, wobei die mathematische Beschreibung dieser Prozesse oft auch algebraische Gleichungen liefert. Diese Gleichungen sind zum Beispiel durch die Kirchhoffschen Gesetze, bestimmte Erhaltungsgrößen wie Volumen oder Energie und geometrische und kinetische Nebenbedingungen gegeben. Dies führt zu den so genannten Differential-Algebraischen Gleichungen (DAEs). Insbesondere durch die verstärkte Simulation dieser Prozesse entstand großes Interesse an DAEs und deren numerischer Behandlung. Im Vordergrund stehen dabei die Simulationen von so genannten Mehrkörpersystemen, wie sie zum Beispiel in der Fahrzeugtechnik und der Robotik auftreten, und von elektrischen Schaltkreisen bei der Chipentwicklung.

Allgemeine lineare Verfahren wurden bereits Mitte der 60ziger Jahre des letzten Jahrhunderts als Verallgemeinerung der klassischen Verfahren, insbesondere der Runge-Kutta Verfahren und der linearen Mehrschrittverfahren, eingeführt. Dieses Konzept ermöglicht zum einen eine einheitliche theoretische Untersuchung der klassischen Verfahren und zum anderen die Herleitung neuer numerischer Methoden. Bis heute werden allgemeine lineare Verfahren auch für steife Differentialgleichungen entwickelt, die auch auf Differential-Algebraische Gleichungen angewendet werden können.

In der vorliegenden Arbeit werden zunächst gewisse grundlegende Aussagen über DAEs und allgemeine lineare Verfahren wiederholt. Anschließend wird die formale Anwendung allgemeiner linearer Verfahren auf DAEs diskutiert. Dabei stellt sich heraus, dass der Störungsindex einer impliziten DAE als Maß der numerischen Schwierigkeiten, welche bei der Berechnung einer Näherungslösung auftreten, irreführend sein kann. Tatsächlich sollte der Störungsindex eines entsprechend augmentierten Systems als dieses Maß betrachtet werden. Zudem werden allgemeine lineare Verfahren für Index-2 DAEs in Hessenberg Form ausführlich analysiert. Dabei liegt der Schwerpunkt bei der Beantwortung der klassischen Fragen der Numerischen Analysis an die Stabilität, Konsistenz und Konvergenz des Verfahrens. Des Weiteren wird eine Übertragung der Konvergenzresultate auf spezielle semi-explizite Index-2 DAEs durchgeführt. Abschließend sind einige Aspekte der Implementierung solcher Verfahren beschrieben und numerische Berechnungen an Beispielen präsentiert.

Abstract

The dynamic behaviour of various applications of physics, chemistry and science of engineering could be modelled by differential equations, while the mathematical description of these dynamics often includes algebraic equations. These equations are due to Kirchhoff's laws, certain conservation laws, i.e., conservation of volume or energy, and geometrical and kinematic constraints. This leads to Differential-Algebraic Equations (DAEs). Especially the simulation of these dynamics rose greater interest on DAEs and their numerical treatment. These are mainly simulations of multibody systems, for example of vehicle systems and robotics, and simulations of electrical circuits.

General Linear Methods (GLMs) were introduced as a generalization of the classical methods in particular of Runge-Kutta and linear multi-step methods in the sixties of the last century. They provide a unifying framework of classical methods and offer the possibility of developing new methods. Until now general linear methods also for stiff differential equations are derived, which could be applied to differential-algebraic equations.

In this thesis certain basic statements on DAEs and general linear methods are repeated at first. Then the formal application of general linear methods to DAEs is discussed. It turns out, that the perturbation index is sometimes not the right measure for the difficulties, which occur by the computation of an approximation. Actually the perturbation index of a certain augmented system should be interpreted as this measure. In addition general linear methods for Index-2 DAEs in Hessenberg form are analyzed in detail. The classical questions of the Numerical Analysis of stability, consistency and convergence are answered. Moreover the results of convergence are formulated for certain semi-explicit Index-2 DAEs. Finally some aspects of the implementation of these Methods are described and numerical computations are presented.

Vorwort

Ich möchte mich an dieser Stelle ganz herzlich bei Herrn Prof. Dr. J. Schropp für seine “doktorväterliche“ Art und für die tollen Zeiten in Konstanz bedanken. Zudem bedanke ich mich bei Herrn Prof. Dr. T. Küpper dafür, dass er mir in der Endphase dieser Arbeit in einer Art und Weise den Rücken frei gehalten hat, wie es nicht selbstverständlich war. Frau Prof. Dr. C. Tischendorf gilt mein Dank für die Unterstützung bei der Auseinandersetzung mit Differential-Algebraischen Gleichungen der Schaltkreissimulation.
Diese Arbeit widme ich meiner Tochter Julie, meinem Sohn Moritz und vor allem meiner Frau Sabine, die mir Kraft gegeben haben diese Arbeit zu vollenden.

Köln, 30. April 2007 Daniel Weiß

Inhaltsverzeichnis

Einleitung **1**

1 Das Problem: Differential-Algebraische Gleichungen **5**
1.1 "Der" Index einer DAE 6
1.2 Index-2 DAEs in Hessenberg Form 11

2 Das numerische Verfahren: Allgemeine lineare Verfahren **21**
2.1 Definition, Beispiele und lineare Stabilität 29
2.2 Stabilität, Konsistenz und Konvergenz 37

3 Allgemeine lineare Verfahren für DAEs **41**
3.1 Die Augmentierung impliziter DAEs 43
3.2 Index-2 DAEs in Hessenberg Form 50
3.2.1 Das Gleichungssystem der Stufenwerte 53
3.2.2 Stabilität 69
3.2.3 Konvergenz und Konsistenz 90
3.3 Semi-explizite Index-2 DAEs 108

4 Implementierung **119**
4.1 Startprozedur 120
4.2 Vereinfachtes Newton-Verfahren 131
4.3 Beispiele 137
4.3.1 Das Pendel 139
4.3.2 Eine lineare DAE mit properem Hauptterm 142
4.3.3 Der "Andrews' squeezer Mechanismus" 144

A DAEs mit properem Hauptterm **149**

B Programme **153**

Literaturverzeichnis **170**

Einleitung

Viele Prozesse der Physik, Chemie und der Ingenieurwissenschaften können durch gewöhnliche Differentialgleichungen beschrieben werden. Dabei liefert die mathematische Beschreibung dieser Prozesse oft auch algebraische Gleichungen bedingt durch zum Beispiel die Kirchhoffschen Gesetze, bestimmte Erhaltungsgrößen wie Volumen oder Energie und geometrische und kinetische Nebenbedingungen. Dies führt zu den so genannten **D**ifferential-**A**lgebraischen Gleichungen (**E**quations, kurz: DAEs), welche, wie der Name schon sagt, neben gewöhnlichen Differential- auch algebraische Gleichungen umfassen.
In der klassischen mathematischen Modellierung wurden diese algebraischen Gleichungen genutzt, um eine Differentialgleichung für die so genannten Zustandskoordinaten zu ermitteln. Diese Gleichung heißt Zustandsform. Die Zustandskoordinaten sind dabei sorgfältig gewählte Koordinaten minimaler Anzahl. Die Herleitung einer Zustandsform bedarf dabei analytischer Arbeit, die in der neueren, computergestützten Modellbildung nicht erwünscht oder nicht möglich ist. Zudem werden durch die Reduktion auf die Zustandskoordinaten gewisse Strukturen der Gleichungen zerstört. Dies ist zum Beispiel in der Modellierung elektrischer Schaltkreise die dünne Besetztheit von Matrizen.
Insbesondere durch die verstärkte Simulation gewisser Prozesse, bei der auch das mathematische Modell rechnergestützt aufgestellt wird, entstand großes Interesse an Differential-Algebraischen Gleichungen und deren numerischer Behandlung. Im Vordergrund stehen dabei die Simulationen von so genannten Mehrkörpersystemen, wie sie zum Beispiel in der Fahrzeugtechnik und der Robotik auftreten, und von elektrischen Schaltkreisen. Mitte der 80ziger Jahre des letzten Jahrhunderts wurden daher viele numerische Verfahren für DAEs entwickelt und theoretisch untersucht (vgl. zum Beispiel [HLR89, BCP, GM]). Dabei waren numerische Verfahren für steife Differentialgleichungen die ersten Verfahren, die erfolgreich auf DAEs angewendet wurden. Die wohl bekanntesten Codes zum Lösen Differential-Algebraischer Gleichungen sind DASSL von L. Petzold und RADAU5 (vgl. [HW] Appendix. Fortran Codes). Während DASSL auf BDF-Methoden basiert, welche von Curtiss und Hirschfelder Mitte des 20. Jahrhunderts für steife Differentialgleichungen entwickelt wurden, liegt dem RADAU5 Code ein Runge-Kutta Verfahren der RadauIIA-Klasse zugrunde. Beide Codes werden erfolgreich auf gewisse DAEs angewendet.

Die Klasse der allgemeinen linearen Verfahren (**G**eneral **L**inear **M**ethods, kurz: GLMs) stellt eine Verallgemeinerung der klassischen Verfahren dar. Sie wurden be-

reits 1966 von J. Butcher formuliert. Zwei Ziele standen dabei im Vordergrund. Zum einen die Bereitstellung eines *unifying framework* der linearen Mehrschritt- und der Runge-Kutta Verfahren und zum anderen die Entwicklung neuer numerischer Methoden, welche die Vorteile der klassischen Verfahren in sich vereinen, deren Nachteile jedoch nicht aufweisen. Bis heute werden in dieser sehr umfassenden Klasse neue Verfahren für gewöhnliche aber auch steife Differentialgleichungen entwickelt (vgl. [BJ04a, BJ04b, BP06, BR05, BW03]), welche theoretisch auch auf Differential-Algebraische Gleichungen angewendet werden könnten. Dieser Entwicklungsprozess ist noch nicht abgeschlossen, so dass auch in Zukunft weitere neue Verfahren entstehen werden.

Während lineare Mehrschrittverfahren und Runge-Kutta Verfahren für Differential-Algebraische Gleichungen bereits untersucht wurden (vgl. zu den linearen Mehrschrittverfahren [BCP, G71, GM, HW, LP86] und zu den Runge-Kutta Verfahren [HLR89, HW, GM, P86, BP89]) sind allgemeine lineare Verfahren angewendet auf DAEs noch nicht ausführlich analysiert. Stefan Schneider betrachtete bereits allgemeine lineare Verfahren für Index-3 DAEs in Hessenberg Form (vgl. [Sch97]) und Steffen Voigtmann entwickelte kürzlich allgemeine lineare Verfahren für eine Klasse Differential-Algebraischer Gleichungen, wie sie bei der Beschreibung elektrischer Schaltkreise vorkommen (vgl. [Voigtmann06]). Eine ausführliche Untersuchung von allgemeinen linearen Verfahren für Index-2 DAEs insbesondere der DAEs in Hessenberg Form scheint daher nötig zu sein.

Die Simulation von Prozessen in den Natur- und Ingenieurwissenschaften wird aufgrund der Kostenreduzierung weiter zunehmen. Insbesondere müssen dabei vermehrt auch Differential-Algebraische Gleichungen möglichst effizient gelöst werden. Zudem spielen verschiedene Aspekte in der Anwendung eine Rolle. Solche Aspekte sind zum Beispiel die Echtzeitintergration oder die Integration, welche geometrische Strukturen erhält. Es geht nicht nur darum eine DAE zu lösen, sondern sie "richtig" zu lösen. Wie bei gewöhnlichen Differentialgleichungen sollten daher verschiedene numerische Verfahren mit ihren jeweiligen Vorteilen zum Lösen von Differential-Algebraischen Gleichungen bereit stehen. Es ist daher vernünftig auch die Klasse der allgemeinen linearen Verfahren angewendet auf DAEs ausführlicher zu untersuchen.
Ein weiterer Grund sich mit allgemeinen linearen Verfahren für DAEs zu beschäftigen liegt in dem *unifying framework*. Aussagen die über GLMs angewendet auf Differential-Algebraische Gleichungen getroffen werden können, gelten für einen großteil der numerischen Verfahren, die heutzutage zum Lösen von DAEs eingesetzt werden. Die formale Anwendung allgemeiner linearer Verfahren auf DAEs kann somit zum besseren Verständnis der allgemeinen Numerik Differential-Algebraischer Gleichungen beitragen (vgl. zum Beispiel Unterkapitel 3.1).

Die Arbeit gliedert sich wie folgt: In den Kapiteln 1 und 2 sind grundlegende Aussagen über Differential-Algebraische Gleichungen und allgemeine lineare Verfahren formuliert. Dabei sind nur die Aspekte beschrieben, welche für die späteren Kapitel

bedeutsam sind. Kapitel 3 bildet den Hauptteil dieser Arbeit. In diesem Kapitel wird zunächst in Unterkapitel 3.1 die formale Anwendung von allgemeinen linearen Verfahren für Differential-Algebraische Gleichungen diskutiert. Dabei wird sich herausstellen, dass der Störungsindex einer impliziten DAE als Maß der numerischen Schwierigkeiten, welche bei der Berechnung einer Näherungslösung auftreten, irreführend sein kann. Tatsächlich sollte der Störungsindex eines entsprechend augmentierten Systems als dieses Maß betrachtet werden. Anschließend werden in Unterkapitel 3.2 allgemeine lineare Verfahren für Index-2 DAEs in Hessenberg Form ausführlich untersucht. Dabei bilden die klassischen Fragen der Numerischen Analysis an ein numerisches Verfahren den roten Faden: Es wird die Durchführbarkeit, die Stabilität, die Konsistenz und die Konvergenz der allgemeinen linearen Verfahren untersucht. Abschließend werden in Unterkapitel 3.3 die Konvergenzresultate auf weitere semi-explizite DAEs mit Störungsindex 2 übertragen. In Kapitel 4 sind Hinweise zur Implementierung allgemeiner linearer Verfahren für Index-2 DAEs und numerische Rechnungen zu Beispielen gegeben.

1 Das Problem: Differential-Algebraische Gleichungen

Wir geben in diesem Kapitel nötige Grundlagen Differential-Algebraischer Gleichungen an. Dabei gehen wir kurz auf die wesentlichen Unterschiede von gewöhnlichen und Differential-Algebraischen Gleichungen ein. Zudem definieren wir den Index einer DAE und gehen ausführlicher auf Index-2 DAEs in Hessenberg Form ein. Dabei bilden die mechanischen Mehrkörpersysteme in der GGL-Formulierung eine sehr wichtige Klasse solcher DAEs.

Die allgemeine Form Differential-Algebraischer Gleichungen ist die von impliziten Differentialgleichungen

$$0 = F(x, y', y), \qquad y(x_0) = y_0. \tag{1.1}$$

Dabei gehen wir davon aus, dass das Bild von F und y dieselbe Dimension besitzen und F stetig und bezüglich y' stetig differenzierbar ist. Zudem existiere eine eindeutige Lösung $y(x)$ auf dem Intervall $[x_0, x_e]$.

Ist die partielle Ableitung $\frac{\partial F}{\partial y'}$ in einer Umgebung der Lösung invertierbar, so ist die Gleichung nach dem Satz über implizite Funktionen lokal nach der Ableitung $y'(x)$ auflösbar. Es handelt sich also bei Gleichung (1.1) um ein System gewöhnlicher Differentialgleichungen in impliziter Form. Wir interessieren uns für Gleichungen, bei denen diese partielle Ableitung im gesamten betrachteten Gebiet singulär ist.

Die mathematische Beschreibung mechanischer Mehrkörpersysteme mit Nebenbedingungen, wie zum Beispiel in der Fahrzeugtechnik und der Robotik, führen auf semi-explizite DAEs der Form

$$\begin{aligned} M(y)y' &= f(t, y, \lambda), \\ 0 &= g(t, y, \lambda), \end{aligned}$$

wobei $M(y)$ eine reguläre Matrix ist (vgl. [EF]).

Die Simulation elektrischer Schaltkreise führt auf so genannte quasilineare Gleichungen:

$$C(U)U' = F(t, U).$$

Dabei ist $C(U)$ die Kapazitätsmatrix und $U(t)$ der Knotenspannungsvektor. Ist die Kapazitätsmatrix invertierbar, können wir diese Gleichung von links mit $C^{-1}(U)$

multiplizieren und erhalten eine gewöhnliche Differentialgleichung. Für eine singuläre Matrix $C(U)$ handelt es sich um eine DAE.

Ein wesentlicher Unterschied zwischen DAEs und gewöhnlichen Differentialgleichungen liegt darin, dass die Lösung bzw. Komponenten der Lösung noch nicht einmal differenzierbar sein müssen:

Beispiel 1.1 *Sei f eine stetige Funktion. Dann gilt für die Lösung der DAE*

$$\begin{aligned} y' &= z, \\ 0 &= z - f \end{aligned}$$

offenbar

$$z(x) = f(x), \qquad y(x) = y(x_0) + \int_{x_0}^{x} f(s)ds.$$

Ein weiterer wichtiger Unterschied zwischen DAEs und gewöhnlichen Differentialgleichungen liegt in der eingeschränkten Wahl der Anfangswerte: Es wird in dem Beispiel deutlich, dass der Anfangswert $z(x_0)$ nicht frei wählbar sondern durch die Funktion f vorgegeben ist. In dem folgenden Beispiel sind sogar alle Anfangswerte festgelegt:

Beispiel 1.2 *Sei f eine differenzierbare Funktion. Dann gilt für die Lösung der DAE*

$$\begin{aligned} y' &= z, \\ 0 &= y - f \end{aligned}$$

offenbar

$$y(x) = f(x), \qquad z(x) = f'(x).$$

Die Differential-Algebraischen Gleichungen der beiden Beispiele sind trotz des ähnlichen Aussehens strukturell verschieden. Im zweiten Beispiel müssen sogar Eingangsdaten differenziert werden, um eine Lösung zu erhalten. Die Differentiation ist jedoch ein instabiler Vorgang, da der Differentialoperator zumindest bezüglich der Supremumsnorm unbeschränkt ist. Gemäß dem Motto: Die Ableitung "kleiner" Störungen kann beliebig "groß" sein. Konkret gehen im zweiten Beispiel die Ableitungen von Störungen der algebraischen Gleichung in die Lösungskomponente z ein. Es sind viele Kriterien entwickelt worden, DAEs nach den eben angedeuteten Eigenschaften zu klassifizieren. Zwei dieser Kriterien stellen wir nun vor.

1.1 "Der" Index einer DAE

Eine Einteilung Differential-Algebraischer Gleichungen geschieht im Allgemeinen durch deren Index. Doch ist dies höchst problematisch und verwirrend, da es sehr

viele verschiedene Indexdefinitionen gibt. Es entstand so manches Missverständnis, weil jede "Schule" ihr eigenes Indexkonzept entwickelt hat und mit einer gewissen Sturheit und Ignoranz ausschließlich vertritt. Einige Gleichungen werden in der Literatur etwas salopp als "Index-k DAEs" (k=1 oder k=2) bezeichnet, obwohl der Index entsprechend der unterschiedlichen Definitionen durchaus verschieden sein kann.

Wir definieren in dieser Arbeit zum einen den Differentiationsindex als das bekannteste und zum anderen den Störungsindex als das in Hinblick auf die Numerik wichtigste Indexkonzept. Sind der Differentiations- und der Störungsindex einer DAE identisch, so sprechen wir vom Index der DAE, andernfalls verwenden wir die entsprechenden Begriffe.
Gear führte 1988 den so genannten Differentiationsindex für DAEs der Form (1.1) ein (vgl. [G88]), wobei die Idee auf früheren Arbeiten zusammen mit Petzold basiert. Die folgende Version ist dem Buch [HW] entnommen (vgl. [HW] S. 455).

Definition 1.3 *Für Differential-Algebraische Gleichungen* (1.1) *ist der Differentiationsindex entlang der Lösung $y(x)$ die minimale Anzahl k von Differentiationen*

$$\begin{aligned} F(x, y', y) &= 0, \\ \frac{\partial F}{\partial x}(x, y', y) &= 0, \\ &\vdots \\ \frac{\partial^k F}{\partial x^k}(x, y', y) &= 0, \end{aligned}$$

die nötig sind, um diese Gleichungen durch algebraische Umformungen in eine explizite gewöhnliche Differentialgleichung

$$y' = \varphi(t, y)$$

zu überführen. Diese Gleichung wird häufig die der DAE (1.1) *zugrunde liegende Differentialgleichung genannt.*

Bemerkungen:

(i) *Die Definition kann auch sinnvoll auf Gleichungen* (1.1) *mit invertierbarer partieller Ableitung $\frac{\partial F}{\partial y'}$ angewandt werden, indem wir in diesem Fall den Differentiationsindex gleich 0 setzen.*

(ii) *Für die oben schon angesprochenen semi-expliziten DAEs braucht nur die algebraische Gleichung differenziert werden.*

Semi-explizite Index-1 DAEs

Die folgende autonome DAE liegt in so genannter semi-expliziter Form vor:

$$\begin{aligned} y' &= f(y, z), & y(0) &= y_0, \\ 0 &= g(y, z), & z(0) &= z_0. \end{aligned}$$

Wir setzen die Invertierbarkeit von $g_z(y,z)$ in einer Umgebung der Lösung voraus. Für die Anfangswerte soll $g(y_0,z_0)=0$ gelten. Man nennt solche Anfangswerte konsistent. Die Invertierbarkeit von $g_z(y,z)$ garantiert, dass die semi-explizite DAE den Differentiationsindex 1 besitzt: In diesem Fall ist das Auflösen der zweiten Gleichung nach z möglich. Eine Differentiation liefert dann die zugrunde liegende Differentialgleichung.

Semi-explizite Index-2 DAEs in Hessenberg Form

Eine sehr wichtige Klasse von DAEs sind Index-2 DAEs in Hessenberg Form (vgl. für eine ausführliche Beschreibung Unterkapitel 1.2). In Kapitel 3 untersuchen wir sehr ausführlich allgemeine lineare Verfahren für solche DAEs. Sie bilden für diese Arbeit die grundlegende Problemklasse.
Diese DAEs sind semi-explizit, jedoch hängt im Unterschied zu den semi-expliziten Index-1 DAEs oben die algebraische Gleichung nicht von z ab:

$$\begin{aligned} y' &= f(y,z), & y(0)&=y_0,\\ 0 &= g(y), & z(0)&=z_0. \end{aligned}$$

Es wird die Invertierbarkeit von $Dg(y)f_z(y,z)$ in einer Umgebung der Lösung vorausgesetzt. Diese Invertierbarkeit garantiert, dass die semi-explizite DAE den Differentiationsindex 2 besitzt, wie leicht einzusehen ist: Das Differenzieren der algebraischen offensichtlichen Nebenbedingung $0=g(y)$ entlang der Lösungskomponente $y(x)$ liefert die so genannte versteckte Nebenbedingung

$$0 = g_y(y)f(y,z).$$

Diese ist nun nach z auflösbar und eine zweite Differentiation liefert die zugrunde liegende Differentialgleichung. Konsistente Anfangswerte erfüllen neben der offensichtlichen auch die versteckte Nebenbedingung.

Bemerkung: *Die Beispiele 1.1 und 1.2 sind semi-explizite DAEs vom Index 1 bzw. 2. Die zusätzliche versteckte Nebenbedingung $z=f'$ im zweiten Beispiel erklärt, warum in diesem 2-dimensionalen System die Anfangswerte eindeutig bestimmt sind.*

Oben wurde vereinbart, nur von Index-k Problemen zu sprechen, wenn der Störungs- und der Differentiationsindex identisch sind. Daher definieren wir zunächst den Störungsindex, wie er von Hairer et al. eingeführt wurde (vgl. [HLR89] S.1):

Definition 1.4 *Für Differential-Algebraische Gleichungen* (1.1) *ist der Störungsindex entlang der Lösung $y(x)$ die minimale ganze Zahl k, so dass für alle Funktionen $\hat{y}(x)$ mit*

$$\delta(x) = F(x,\hat{y}',\hat{y}), \qquad \hat{y}(x_0) = y_0+\delta_0$$

auf $[x_0, x_e]$ *die folgende Abschätzung gilt*

$$\|y(x) - \hat{y}(x)\| \leq C \left(\|\delta_0\| + \max_{\xi \in [x_0, x_e]} \|\delta(\xi)\| + \ldots + \max_{\xi \in [x_0, x_e]} \|\delta^{(k-1)}(\xi)\| \right),$$

wann immer der Ausdruck auf der rechten Seite hinreichend klein ist. Die Konstante C *hängt dabei nur von* F *und der Länge des Intervalls ab.*

Tatsächlich sind beide Indexkonzepte für die semi-expliziten DAEs oben identisch, womit die Bezeichnungen Index-1 bzw. Index-2 gerechtfertigt sind (vgl. [HLR89] S.2-4).

Der Störungsindex ist als Maß für die sensitive Abhängigkeit der Lösung von Störungen in der Gleichung eingeführt worden. Da die Lösung nicht nur von der Störung δ sondern auch von deren Ableitungen bis zur Ordnung $k-1$ abhängt, kann im Fall $k > 1$ die Konvergenz von $\hat{y}(x)$ gegen $y(x)$ für $\delta \to 0$ nicht garantiert werden. Tatsächlich können kleine Störungen beliebig große Änderungen der Lösung bewirken. In diesem Sinne gehören DAEs mit Störungsindex $k > 1$ zu der Klasse der schlecht konditionierten Probleme. Oder alternativ formuliert: Die Lösung einer solchen DAE hängt nicht stetig von Störungen der Gleichung ab.
Für die numerische Lösung spiegelt sich die höchste Ableitung der Störung δ in einer Division eines zusätzlichen Fehlerterms durch h^{k-1} wider. Dieser Fehlerterm entsteht zum Beispiel durch Rundungsfehler oder Fehler beim iterativen Lösen von nichtlinearen Gleichungssystemen. Für $h \to 0$ wird dieser Term unendlich groß, was zum Scheitern des numerischen Verfahrens führen kann (vgl. [A95]). Die numerische Behandlung von Differential-Algebraischen Gleichungen mit einem Störungsindex $k > 1$ ist daher problematisch. Der Störungsindex wird als Maß der numerischen Schwierigkeiten, mit denen man beim Lösen von (1.1) durch ein numerisches Verfahren rechnen muss, angesehen. Wir zeigen in Kapitel 3 mit Hilfe der Anwendung von allgemeinen linearen Verfahren auf DAEs der Form (1.1), dass dies nicht ganz präzise ist. Dort arbeiten wir zudem heraus, auf was geachtet werden muss, um den Störungsindex weiterhin als dieses Maß verstehen zu können.

Bemerkung: *Wie bei dem Differentiationsindex können gewöhnliche Differentialgleichungen* $y' = f(x, y)$ *mit in die Definition einbezogen werden, indem* $\delta^{(-1)}$ *mit dem Integral über* δ *identifiziert wird. Eine Anwendung des Lemmas von Gronwall liefert nämlich in diesem Fall*

$$\|y(x) - \hat{y}(x)\| \leq C \left(\|\delta_0\| + \max_{\xi \in [x_0, x_e]} \| \int_{x_0}^{\xi} \delta(s) ds \| \right).$$

Der Störungsindex ist demnach 0.

Lange Zeit bestand die Meinung, dass der Störungsindex einer DAE gleich oder maximal um eins größer sei als der Differentiationsindex dieser Gleichung. Gear, der

eine entsprechende Aussage "bewies", lieferte sechs Jahre später selbst ein Gegenbeispiel, bei welchem die Indizes tatsächlich mit der Dimension des Systems beliebig voneinander abweichen können (vgl. [CG95]):

Beispiel: *Wir betrachten die m-dimensionale DAE*

$$F(y', y) = y_m \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & 1 \\ 0 & \dots & \dots & \dots & 0 \end{pmatrix} y' + y = 0.$$

Für die einzelnen Komponenten gilt offenbar

$$\begin{aligned} y_m y_2' + y_1 &= 0, \\ y_m y_3' + y_2 &= 0, \\ &\vdots \\ y_m y_m' + y_{m-1} &= 0, \\ y_m &= 0. \end{aligned}$$

Somit besitzt die Gleichung den Differentiationsindex 1. Die Lösung ist $y \equiv 0$.
Wir betrachten nun die gestörte Gleichung $F(\hat{y}', \hat{y}) = \delta(x)$. Da nun im Allgemeinen die Komponente $\hat{y}_m = \delta_m$ nicht länger 0 ist, folgt für die anderen Komponenten

$$\begin{aligned} \hat{y}_{m-1} &= \delta_{m-1} - \delta_m \delta_m', \\ \hat{y}_{m-2} &= \delta_{m-2} - \delta_m [\delta_{m-1}' - \delta_m' \delta_m' - \delta_m \delta_m''], \\ &\vdots \\ \hat{y}_1 &= \delta_1 - \delta_m [\qquad], \end{aligned}$$

wobei die Klammer von $\delta_m^{(m-1)}$ abhängt. Der Störungsindex ist somit m.

Bemerkung: *Ein weiterer Indexbegriff im Zusammenhang von DAEs der elektrischen Schaltkreissimulation ist der so genannte Traktabilitätsindex, der von Griepentrog und März eingeführt wurde (vgl. [GM]). Dieser Index kann für DAEs formuliert und bestimmt werden, die Glattheitsdefizite aufweisen. Solche Gleichungen treten in der Schaltkreissimulation auf. Zudem bestimmt die Topologie der Schaltkreise in vielen Fällen den Traktabilitätsindex der diesen Schaltkreis beschreibenden DAE. Typischerweise ist dieser Index nicht größer als 2 und stimmt in den oben erwähnten semi-expliziten Gleichungen mit den beiden anderen definierten Indizes überein.*

1.2 Index-2 DAEs in Hessenberg Form

In diesem Unterkapitel betrachten wir Differential-Algebraische Gleichungen in semi-expliziter Form

$$\begin{array}{rcll} y' & = & f(y,z), & y(0) = y_0 \in \mathbb{R}^N, \\ 0 & = & g(y), & z(0) = z_0 \in \mathbb{R}^l, \end{array} \tag{1.2}$$

wobei wir die Invertierbarkeit von $Dg(y)f_z(y,z)$ in einer Umgebung der Lösung voraussetzen. Diese Voraussetzung heißt Index-2 Bedingung:

Die Matrix $\boldsymbol{Dg(y)f_z(y,z)}$ ist invertierbar. (DAE1)

Sie garantiert, dass diese DAEs den Index 2 besitzen. Diese Gleichungen heißen Index-2 DAEs in Hessenberg Form. Wir gehen im Folgenden von der Existenz einer eindeutigen Lösung $y(x), z(x)$ auf dem Intervall $[0, x_e]$ aus.

Diese Gleichungen bilden eine sehr wichtige Problemklasse unter den DAEs. Sie sind typische Gleichungen der Mechanik (vgl. weiter unten den Abschnitt über mechanische Mehrkörpersysteme). Zudem können andere Gleichungen in diese Form transformiert werden: Dies sind zum Beispiel gewöhnliche Differentialgleichungen mit Invarianten, quasilineare Systeme der Form

$$B(y)y' = f(y),$$

wie sie bei der Schaltungssimulation und der chemischen Reaktionskinetik vorliegen (vgl. weiter unten den Abschnitt "Weitere Beispiele"). Alle diese Probleme haben formuliert als Index-2 DAE in Hessenberg Form gemeinsam, dass sie von der algebraischen Komponente nur linear abhängen, d.h. es gilt:

$$f(y,z) = f_0(y) + f_z(y)z.$$

Diese Struktur hat Auswirkungen auf die Numerik solcher Gleichungen. Wir gehen darauf in Kapitel 3 näher ein.
Allgemeine Index-2 DAEs in Hessenberg Form findet man auch in der *optimal control theory* und bei *trajectory prescribed path control problems* (vgl. [BCP]).

Die y-Komponente der Lösungen dieser DAE liegt offenbar in der Mannigfaltigkeit

$$\mathcal{M} = \{y \in \mathbb{R}^N |\, g(y) = 0\}.$$

Die Index-2 Bedingung, d.h. die Invertierbarkeit von Dgf_z, garantiert lokal die Existenz einer auflösenden Funktion $z = \psi(y)$, so dass wir für die y-Komponente eine gewöhnliche Differentialgleichung erhalten:

$$y' = f(y, \psi(y)), \qquad y(0) = y_0.$$

Tatsächlich lässt sich also die Dynamik dieser DAE zumindest lokal durch eine gewöhnliche Differentialgleichung auf der Mannigfaltigkeit $\mathcal{M}$ beschreiben.
Da z durch $\psi(y)$ eindeutig bestimmt ist und die Mannigfaltigkeit $\mathcal{M}$ die Dimension $N - l$ besitzt, erhalten wir insgesamt $N - l$ Freiheitsgrade für die DAE, d.h. es können nur $N - l$ Anfangswerte frei vorgegeben werden.

Differenzieren wir die algebraische offensichtliche Nebenbedingung entlang einer Lösung, so erhalten wir die versteckte Nebenbedingung

$$Dg(y)f(y,z) = 0.$$

Lösungen liegen also in der "versteckten" Mannigfaltigkeit

$$\mathcal{M}_1 = \{(y,z) \in \mathbb{R}^N \times \mathbb{R}^l |\, g(y) = 0,\ Dg(y)f(y,z) = 0\}.$$

Die versteckte Nebenbedingung bzw. die "Kontrollvariable" z garantieren gerade, dass die rechte Seite f, das heißt der Geschwindigkeitsvektor $y' = f(y,z)$, tangential zur Mannigfaltigkeit $\mathcal{M}$ liegt. Somit ist $\mathcal{M}_1$ positiv invariant: eine Lösung mit Startwert in $\mathcal{M}_1$ bleibt für alle Zeiten, für die sie existiert, in $\mathcal{M}_1$.
Differenzieren wir die versteckte Nebenbedingung ein weiteres mal entlang der Lösung, so finden wir mit der Index-2 Bedingung lokal eine Differentialgleichung für die algebraische Variable z:

$$z' = \varphi(y,z).$$

Für konsistente Anfangswerte, d.h. für Werte in $\mathcal{M}_1$, sind die folgenden Formulierungen nach dem Hauptsatz der Differential- und Integralrechnung äquivalent:

$$\begin{array}{rcll} y' &=& f(y,z), & y(0) = y_0, \\ z' &=& \varphi(y,z), & z(0) = z_0. \end{array} \qquad \text{(Index-0)}$$

$$\begin{array}{rcll} y' &=& f(y,z), & y(0) = y_0, \\ 0 &=& Dg(y)f(y,z), & z(0) = z_0. \end{array} \qquad \text{(Index-1)}$$

$$\begin{array}{rcll} y' &=& f(y,z), & y(0) = y_0, \\ 0 &=& g(y), & z(0) = z_0. \end{array} \qquad \text{(Index-2)}$$

Tatsächlich liefern numerische Verfahren für die unterschiedlichen Formulierungen verschiedene Resultate. Der Störungsindex macht zudem deutlich: Formulierungen mit höherem Index sind schwieriger zu lösen, da Störungen der entsprechenden Gleichung größere Auswirkung auf die Lösung haben können, oder anders formuliert: Schon die analytische Lösung ist empfindlicher gegenüber Störungen der Gleichungen einer Formulierung mit höherem Index als gegenüber Störung einer Formulierung mit niedrigerem Index. Insbesondere wird dies auch für die entsprechenden numerischen Lösungen der Fall sein. Grundsätzlich ist also aus numerischen Gründen eine

Formulierung niedrigeren Indexes einer mit höherem Index vorzuziehen.
Jedoch ist ein numerisches Verfahren angewendet auf eine Index-k Formulierung "blind" für die algebraischen Gleichungen der Formulierungen mit Index größer k: Die numerische Lösung liegt nicht mehr auf $\mathcal{M}$ oder $\mathcal{M}_1$! Dieses Phänomen wird als *drift-off* bezeichnet. Eine solche Destabilisierung kann jedoch durch geeignete Projektionen ausgeglichen werden (vgl. [HW] S.470).
Wir wollen Index-2 DAEs in Hessenberg Form jedoch in ihrer ursprünglichen Form lösen. Zudem sind Index-2 Systeme der Mehrkörpermechanik bereits indexreduzierte Formulierungen gewisser Index-3 Gleichungen (vgl. weiter unten den Abschnitt über mechanische Mehrkörpersysteme).

Wichtig im Zusammenhang einer genauen Analyse von Störungen der Index-2 DAE (1.2) sind die beiden entlang der Lösung definierten Projektionen

$$\begin{array}{rcl} Q(x) & := & (f_z(Dgf_z)^{-1}Dg)(y(x),z(x)), \\ P(x) & := & I_N - Q(x). \end{array} \tag{1.3}$$

Bezeichne $T_{y(x)}\mathcal{M}$ den Tangentialraum der Mannigfaltikeit $\mathcal{M}$ an der Stelle $y(x)$ und sei der Unterraum $T_{f_z(x)}\mathcal{V}$ des $\mathbb{R}^N$ das Bild der partiellen Ableitung $f_z(y(x),z(x))$. Die Index-2 Bedingung garantiert, dass diese beiden Räume transversal sind. Zudem gilt:

$$T_{y(x)}\mathcal{M} = \mathrm{ker}Dg(y(x)).$$

Aus der Definition von Q und P folgt unmittelbar, dass $P(x)$ auf den Tangentialraum $T_{y(x)}\mathcal{M}$ parallel zu $T_{f_z(x)}\mathcal{V}$ projiziert und $Q(x)$ entsprechend auf $T_{f_z(x)}\mathcal{V}$ parallel zu $T_{y(x)}\mathcal{M}$ abbildet (vgl. Abbildung 1.1).

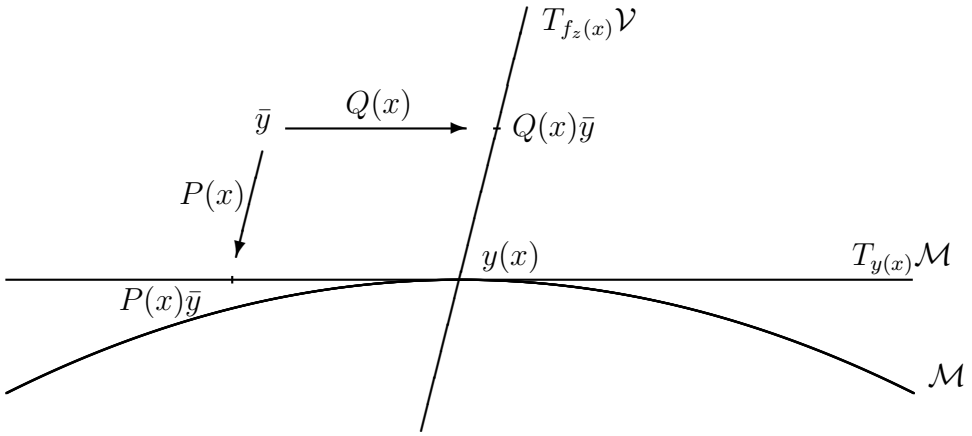

Abbildung 1.1: Die Projektion $P(x)$ bildet auf den Tangentialraum $T_{y(x)}\mathcal{M}$ parallel zu $T_{f_z(x)}\mathcal{V}$ ab. $Q(x) = I - P(x)$ projiziert entsprechend auf $T_{f_z(x)}\mathcal{V}$ parallel zum Tangentialraum.

Wir betrachten nun das gestörte System

$$\begin{aligned} \hat{y}' &= f(\hat{y}, \hat{z}) + \delta(x), & \hat{y}(0) &= y_0 + \delta_0, \\ \theta(x) &= g(\hat{y}), & \hat{z}(0) &= z_0 + \theta_0. \end{aligned}$$

Unter der Annahme, dass δ stetig und θ stetig differenzierbar sind, erhalten wir für hinreichend kleine Störungen und $\|\theta'\|$ folgende Abschätzung auf dem Intervall $[0, x_e]$ (vgl. [ASW95] Theorem 1a)):

$$\|y(x) - \hat{y}(x)\| \leq C \left(\|\delta_0\| + \int P\delta + \max_{\xi \in [0,x_e]} \|\theta(\xi)\|(1 + D(x)) + \mu D(x)^2 \right). \quad (1.4)$$

Dabei ist μ eine obere Schranke von $f_{zz}(\eta, \xi)$ in der Nähe der Lösung und

$$\begin{aligned} \int P\delta &:= \max_{\xi \in [0,x_e]} \| \int_0^{\xi} P(s)\delta(s)ds \|, \\ D(x) &:= \max_{\xi \in [0,x_e]} \|\delta(\xi)\| + \max_{\xi \in [0,x_e]} \|\theta'(\xi)\|. \end{aligned}$$

Diese Darstellung liefert eine genauere Abhängigkeit der Lösung von Störungen der Gleichung, als dies der Störungsindex in seiner Allgemeinheit leisten kann: Besitzt zum Beispiel die rechte Seite der Differentialgleichung die besondere Form

$$f(y, z) = f_0(y) + f_z(y)z,$$

so kann in (1.4) die Konstante μ als obere Schranke der partiellen Ableitung f_{zz} gleich 0 gesetzt werden. Somit fällt ein vergleichsweise großer Störungsterm in der Abschätzung (1.4) weg. Ist die Matrix $f_z(y) = f_z \in \mathbb{R}^{N,l}$ konstant, so gilt $D(x) = 0$ und wir erhalten die Abschätzung

$$\|y(x) - \hat{y}(x)\| \leq C \left(\|\delta_0\| + \int P\delta + \max_{\xi \in [0,x_e]} \|\theta(\xi)\| \right)$$

(vgl. [ASW95] Theorem 1b)). In dieser Abschätzung tritt keine Ableitung einer Störung auf. Es handelt sich also bei der y-Komponente tatsächlich um eine Index-1 Variable. Die z-Variable hängt dagegen von θ' ab, wie Arnold et al. sogar an einem linearen System zeigen(vgl. ebd. Example 1). Diese genaueren Abschätzungen im Fall der linearen Abhängigkeit der Funktion f von der algebraischen Variablen z finden auch in der Numerik der Index-2 DAEs in Hessenberg Form ihren Ausdruck. Wir gehen darauf an entsprechender Stelle in Unterkapitel 3.2 ein.

DAEs in Hessenberg Form lassen sich für beliebig hohen Index definieren (vgl. [KM] S.172). Dabei gibt die Hessenbergstruktur der Jacobi-Matrix der rechten Seite dieser Klasse von Problemen ihren Namen. Störungs- und Differentiationsindex sind für Hessenberg-DAEs identisch (vgl. [HLR89] S.13). Für uns sind neben den Index-2 Problemen auch die Hessenberg DAEs mit Index-3 interessant. Eine sehr wichtige Problemklasse dieser DAEs sind mechanische Mehrkörpersysteme mit Nebenbedingungen. Sie lassen sich in der weiter unten beschriebenen GGL-Formulierung als Index-2 DAEs formulieren.

Mechanische Mehrkörperprobleme

Die Bewegungsgleichungen eines mechanischen Mehrkörpersystems mit Nebenbedingungen sind gegeben durch (vgl. [EF]):

$$\begin{aligned} M(p)\ddot{p} &= f(t,p,\dot{p}) - f_c(p,\lambda), \\ 0 &= g(p). \end{aligned}$$

Dabei ist $g : \mathbb{R}^N \to \mathbb{R}^l$ eine die l Nebenbedingungen beschreibende Funktion, welche hier nur von der Position p des Systems abhängt. Die Funktion $f_c : \mathbb{R}^N \times \mathbb{R}^l \to \mathbb{R}^N$ beschreibt zusätzliche Kräfte zum Kräftevektor $f(t,p,\dot{p})$, die das System zur Erfüllung der Nebenbedingungen zwingen. Nach dem Prinzip von d'Alembert, Zwangskräfte verrichten keine Arbeit, sind diese Kräfte gegeben durch:

$$f_c(p,\lambda) = G(p)^T\lambda$$

mit $G(p) := Dg(p)$, die $\lambda \in \mathbb{R}^l$ heißen Lagrange-Multiplikatoren. Diese Zwangskräfte sind also orthogonal zur Mannigfaltigkeit

$$\mathcal{M} := \{y \in \mathbb{R}^N |\, g(y) = 0\}.$$

Tatsächlich setzen wir voraus, dass $G(p)$ maximalen Rang l, $l \leq N$ besitzt, d.h. die einzelnen Nebenbedingungen sind linear unabhängig. Die Massematrix $M(p)$ ist in der Regel eine symmetrische positiv definite Matrix, so dass wir im Folgenden von

$$GM^{-1}G \text{ invertierbar} \tag{1.5}$$

ausgehen werden.

Wie im Fall der Index-2 DAEs in Hessenberg Form erhalten wir durch Differenzieren der algebraischen Gleichung $0 = g(p)$ versteckte Nebenbedingungen

$$0 = G(p)\dot{p}.$$

Differenzieren wir diese Gleichung wiederum entlang einer Lösung, finden wir im Unterschied zu Index-2 DAEs weitere versteckte Nebenbedingungen:

$$\begin{aligned} 0 &= G(p)\ddot{p} + g_{pp}(p)(\dot{p},\dot{p}) \\ &= G(p)M(p)^{-1}f(t,p,\dot{p}) - (GM^{-1}G^T)(p)\lambda + g_{pp}(p)(\dot{p},\dot{p}) \\ &=: \psi(t,p,\dot{p}). \end{aligned}$$

Eine weitere Differentiation liefert mit der Invertierbarkeit von $GM^{-1}G^T$ und geeigneter Wahl von φ eine Differentialgleichung für die Lagrange-Multiplikatoren:

$$\dot{\lambda} = \varphi(t,p,\dot{p}).$$

Diese mechanischen Mehrkörpersysteme sind aufgrund der Bedingung (1.5) Differential-Algebraische Gleichungen vom Index 3.

Ein sehr einfaches "Mehr"-körpersystem ist das Pendel:

Beispiel 1.5 *Wir betrachten ein Pendel der Länge l und der Masse m, welches im Ursprung eines kartesischen Koordinatensystems (x_1, x_2) durch eine Stange aufgehängt ist. Die Masse der Stange sei vernachlässigt. In diesen redundanten Koordinaten lauten die Bewegungsgleichungen*

$$\begin{aligned} m\ddot{x}_1 &= -2x_1\lambda, \\ m\ddot{x}_2 &= -mg - 2x_2\lambda, \\ 0 &= x_1^2 + x_2^2 - l^2. \end{aligned}$$

Hier sind g die Gravitationskonstante und λ die Lagrange-Multiplikatoren. Die algebraische Gleichung definiert eine Mannigfaltigkeit: den Kreis mit Radius l.
Durch Einführung von Polarkoordinaten lässt sich die Dynamik auch nur durch den Winkel φ zwischen der Senkrechten und der Stange des Pendels beschreiben:

$$l\ddot{\varphi} = -g\sin\varphi.$$

Diese gewöhnliche Differentialgleichung ist eine Zustandsform und der Winkel eine Zustandskoordinate.
Die erste versteckte Nebenbedinung lautet:

$$0 = x_1\dot{x}_1 + x_2\dot{x}_2.$$

Diese Gleichung spiegelt wider, dass der Geschwindigkeitsvektor tangential zur Laufbahn der Masse steht, welche durch die algebraische Gleichung definiert ist. Differenzieren wir ein weiteres Mal, so erhalten wir die zweite versteckte Nebenbedingung

$$0 = m(\dot{x}_1^2 + \dot{x}_2^2) - 2l^2\lambda - mgx_2.$$

Hier haben wir zusätzlich die algebraische Gleichung investiert. Diese Gleichung lässt sich nach λ auflösen. Es handelt sich also um ein Index 3 Problem.

Für konsistente Anfangswerte (p_0, v_0, λ_0) - dies sind Werte, welche alle Nebenbedingungen erfüllen - sind die folgenden Formulierungen äquivalent:

$$\begin{aligned} \dot{p} &= v, & p(t_0) &= p_0, \\ M(p)\dot{v} &= f(t,p,v) - G(p)^T\lambda, & v(t_0) &= v_0 \end{aligned}$$

und eine der Gleichungen:

$$\dot{\lambda} = \varphi(t,p,v), \qquad \lambda(t_0) = \lambda_0. \tag{Index-0}$$

Beschleunigungslevel:

$$0 = \psi(t,p,v), \qquad \lambda(t_0) = \lambda_0. \tag{Index-1}$$

Geschwindigkeitslevel:

$$0 \;=\; G(p)v, \qquad\qquad \lambda(t_0) = \lambda_0. \qquad\qquad \text{(Index-2)}$$

Positionslevel:

$$0 \;=\; g(p), \qquad\qquad \lambda(t_0) = \lambda_0. \qquad\qquad \text{(Index-3)}$$

Die numerische Behandlung dieser Systeme ist jedoch sehr verschieden. Wie wir bereits oben erwähnten, sind Formulierungen mit niedrigerem Index denen mit höherem Index aus numerischen Gründen vorzuziehen, wobei jedoch numerische Verfahren angewendet auf eine Index-k Formulierung "blind" für die algebraischen Nebenbedingungen der Formulierung mit Index größer als k sind. Die Approximationen der Position $p(t)$ zum Beispiel liegen nicht in $\mathcal{M}$ bzw. driften sogar während der Integration immer weiter von $\mathcal{M}$ ab, obwohl der Anfangswert konsistent war. Bei jeder Indexreduktion durch Differentiation wird zwar das erhaltene System "leichter" numerisch lösbar, jedoch entsteht eine invariante Größe für die analytische Lösung des Systems. Diese Invariante stellt keine Restriktion der analytischen Lösung dar, sondern enthält zusätzliche Informationen über Eigenschaften der Lösung, die automatisch erfüllt sind. Es sind daher verschiedene Indexreduktionen zusammen mit Stabilisierungstechniken entwickelt worden, um diesen drift-off zu vermeiden (vgl. [HW] Kapitel VII.2).

GGL-Stabilisierung:
Eine wirklich brilliante Idee wurde von Gear, Gupta und Leimkuhler 1985 entwickelt (vgl. [GGL]): Sie führten zusätzliche Lagrange-Multiplikatoren $\mu \in \mathbb{R}^l$ in der Index-3 Formulierung ein und fügten die Index-2 Gleichung an. Das resultierende System ist zwar von höherer Dimension, besitzt aber tatsächlich den Index-2 und erhält keine zusätzliche Invariante:

$$\begin{aligned} \dot{p} &= v - G(p)^T\mu, & p(t_0) &= p_0, \\ M(p)\dot{v} &= f(t,p,v) - G(p)^T\lambda, & v(t_0) &= v_0, \\ 0 &= g(p), & \lambda(t_0) &= \lambda_0, \\ 0 &= G(p)v, & \mu(t_0) &= 0. \end{aligned}$$

Zudem ist diese Indexreduktion leicht zu realisieren, da die Funktion $G(p)$ bekannt ist. Multiplizieren wir die zweite Gleichung von links mit $M(p)^{-1}$, so erhalten wir in der Tat eine Index-2 DAE in Hessenberg Form (1.2) mit $y = (p, v)$ und $z = (\lambda, \mu)$, wobei die algebraischen Variablen λ und μ linear in die Differentialgleichungen eingehen. Die Index-2 Bedingung ist offenbar mit der Invertierbarkeit von $M(p)$ und dem vollen Rang der Matrix $G(p)$ garantiert. Die zusätzlichen Lagrange-Multiplikatoren verschwinden in der analytischen Lösung:

$$0 = \frac{d}{dt}g(p) = G(p)v - G(p)G(p)^T\mu = -G(p)G(p)^T\mu.$$

Aus Gründen der Symmetrie wird oft die erste Differentialgleichung vor der Kopplung mit den Lagrange-Multiplikatoren μ von links mit $M(p)$ multipliziert (vgl. [HW] S.465).

Beispiel 1.6 *Das Pendel aus Beispiel 1.5 lautet in der GGL-Formulierung:*

$$\begin{aligned} \dot{x}_1 &= y_1 - 2x_1\mu, \\ \dot{x}_2 &= y_2 - 2x_2\mu, \\ m\dot{y}_1 &= -2x_1\lambda, \\ m\dot{y}_2 &= -mg - 2x_2\lambda, \\ 0 &= x_1^2 + x_2^2 - l^2, \\ 0 &= 2x_1y_1 + 2x_2y_2. \end{aligned}$$

Weitere Beispiele von Index-2 DAEs in Hessenberg Form

Der Zusammenhang zwischen Differential-Algebraischen Gleichungen und Invarianten wurde bereits weiter oben angesprochen: Die Indexreduktion durch Differentiation führt auf DAEs niedrigeren Indexes, wobei die algebraischen Nebenbedingungen zu Invarianten des Systems werden. Am Ende dieses Reduktionsprozesses steht eine gewöhnliche Differentialgleichung mit Invarianten:

$$\begin{aligned} y' &= f(y), & y(0) = y_0, \\ 0 &= h(y). \end{aligned}$$

Umgekehrt liefert eine solche gewöhnliche Differentialgleichung mit Invarianten eine Index-2 DAE in Hessenberg Form, indem man eine neue Variable einführt:

$$\begin{aligned} y' &= f(y) - H(y)z, & y(0) = y_0, \\ 0 &= h(y), & z(0) = 0. \end{aligned}$$

Dabei ist $H(y)$ eine beliebige matrixwertige Funktion, so dass $Dh(y)H(y)$ invertierbar ist. Die analytische Lösung von z ist identisch 0.
Gear hat diese Stabilisation der Invarianten mit $H = Dh^T$ in Anlehnung an die GGL-Stabilisierung oben vorgeschlagen (vgl. [G86]).
Natürlich sind auch andere Techniken zum Lösen des überbestimmten gewöhnlichen Differentialgleichungssystems oben entwickelt worden, die eine DAE Formulierung vermeiden (vgl. [AP] S.249ff und [EFLR90]).

Quasilineare Differential-Algebraische Gleichungen der Form

$$B(y)y' = f(y)$$

treten zum Beispiel bei der Simulation elektrischer Schaltkreise und der chemischen Reaktionskinetik auf (vgl. [HLR89] und [L89] und die Referenzen dort).

Eine solche Gleichung durch ein allgemeines lineares Verfahren zu lösen, ist in einem gewissen Sinne äquivalent dazu, dieses Verfahren auf das augmentierte System

$$\begin{aligned} y' &= w, \\ 0 &= f(y) - B(y)w \end{aligned}$$

anzuwenden (vgl. dazu Unterkapitel 3.1). Die Näherungen der y-Komponente sind bei beiden Diskretisierungen völlig identisch.
Besitzt nun die Matrix $B(y)$ konstanten Rang, so existieren invertierbare Matrizen S und T mit

$$B(y) = S(y) \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} T(y).$$

Multiplizieren wir die algebraische Gleichung des augmentierten Systems von links mit $S^{-1}(y)$, so erhalten wir bei entsprechender Definition der Komponenten und Funktionen das äquivalente System

$$\begin{aligned} y_1' &= w_1, \\ y_2' &= w_2, \\ 0 &= h(y) - T_{11}(y)w_1 - T_{12}(y)w_2, \\ 0 &= g(y), \end{aligned}$$

wobei T_{11} ohne Einschränkung invertierbar ist (vgl. [HLR89] S.5 für eine ausführlichere Darstellung).
Diese Umformung ist nicht nur für die analytische Lösung äquivalent, sondern auch für die Diskretisierung durch ein allgemeines lineares Verfahren (vgl. dazu Unterkapitel 3.3). Ebenso äquivalent ist die Ersetzung von w_1 in der ersten Differentialgleichung durch die nach dieser Variablen aufgelösten dritten Gleichung. Diese so erhaltene DAE ist von Hessenberg Form. Die Index-2 Bedingung lautet:

$$Dg(y) \begin{pmatrix} -T_{11}^{-1}(y)T_{12}(y) \\ I \end{pmatrix} \text{ invertierbar.}$$

Wir werden auf diese Umformungen in Kapitel 3 in allgemeinerer Form nochmals eingehen.

Weitere Differential-Algebraische Gleichungen mit Index 2 in Hessenberg Form treten bei gewissen Diskretisierungen im Raum partieller Differentialgleichungen auf (vgl. [AP, LP86]).

2 Das numerische Verfahren: Allgemeine lineare Verfahren

Das wohl bekannteste numerische Verfahren zur Berechnung von Näherungslösungen einer gewöhnlichen Differentialgleichung ist das Euler-Cauchy Verfahren. Es basiert auf einem sehr einfachen Prinzip: Angenommen die Bewegung eines Teilchens kann durch eine gewöhnliche Differentialgleichung

$$\dot{y} = f(t, y)$$

beschrieben werden; zudem habe das Teilchen zur Zeit t_0 die Position y_0 und die Geschwindigkeit $v_0 = f(t_0, y_0)$. Wir gehen davon aus, dass die Geschwindigkeit für eine "kurze" Zeitspanne Δt mehr oder weniger konstant bleibt. Dann ist die neue Position zum Zeitpunkt $t_0 + \Delta t$ ungefähr

$$y_1 = y_0 + \Delta t v_0.$$

Ausgehend von y_0 werden auf diese Weise weitere Näherungen berechnet

$$y_{n+1} = y_n + \Delta t f(t_n, y_n)$$

mit $t_n = t_0 + n\Delta t$.
Diese Idee geht auf Euler und somit in das 18. Jahrhundert zurück. Erst Cauchy bewies in der ersten Hälfte des 19. Jahrhunderts die Konvergenz dieses Verfahrens. Eine einfache Variante dieser Idee besteht darin, die "konstante" Geschwindigkeit am Ende des Zeitintervalls $[t_0, t_0 + \Delta t]$ zu berechnen. Dies führt auf das so genannte implizite Euler-Cauchy Verfahren:

$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1}).$$

Hier ist ein implizites Gleichungssystem zur Berechnung der Näherung y_{n+1} zu lösen. Generell haben implizite Verfahren gegenüber expliziten Verfahren große Vorteile beim Lösen von steifen und Differential-Algebraischen Gleichungen.

Es gibt nun mindestens zwei Wege, die Güte der Näherungen zu verbessern (vgl. Abbildung 2.1):

- Mehr Auswertungen der Funktion f pro Schritt (*multistage*).
- Näherungen von früheren Zeitpunkten in die Rechnung mit einbeziehen (*multivalue*).

Bei einem Runge-Kutta Verfahren werden ausgehend von $y^{[0]} = y_0$ weitere Approximationen wie folgt berechnet:

$$y^{[n+1]} = y^{[n]} + \Delta t \sum_{i=1}^{s} b_i f(t_n + c_i \Delta t, Y_i).$$

Es werden also mehrere f-Auswertungen investiert und durch die b_i gewichtet:

$$\sum_{i=1}^{s} b_i = 1.$$

Die Knoten c_i liegen in der Regel zwischen 0 und 1. Die Größen Y_i approximieren die analytische Lösung an inneren Gitterpunkten $t_n + c_i \Delta t$ und werden durch ein im Allgemeinen nichtlineares Gleichungssystem berechnet:

$$Y_i = y^{[n]} + \Delta t \sum_{j=1}^{s} a_{ij} f(t_n + c_j \Delta t, Y_j), \qquad i = 1, \ldots, s.$$

Sie werden daher innere Stufenwerte genannt.
Die Idee, das Euler-Cauchy Verfahren in dieser Art zu verallgemeinern, geht auf Runge, Kutta und Heun, und somit in die Zeit der Jahrhundertwende des 19. und 20. Jahrhunderts zurück.

Bei den linearen Mehrschrittverfahren der Adamsklasse werden statt zusätzlicher Funktionsauswertungen an inneren Gitterpunkten die Funktionswerte früherer Näherungen verwendet:

$$y^{[n+1]} = y^{[n]} + \Delta t \sum_{i=0}^{r} \beta_i f(t_{n+1-i}, y^{[n+1-i]}).$$

Für $\beta_0 = 0$ erhalten wir explizite Verfahren der Adams-Bashforth Klasse, für $\beta_0 \neq 0$ sind die Verfahren implizit und werden Adams-Moulton Verfahren genannt. Diese Verfahren wurden ebenfalls am Ende des 19. Jahrhunderts entwickelt.
Für steife und Differential-Algebraische Gleichungen wurden insbesondere so genannte *Backward Differentiation Formulas*, kurz BDFs, verwendet. Sie wurden von Curtiss und Hirschfelder Mitte des 20. Jahrhunderts eingeführt und haben die Form

$$y^{[n+1]} = \sum_{j=1}^{r} \alpha_j y^{[n+1-j]} + \Delta t \beta_0 f(t_{n+1}, y^{[n+1]}).$$

Hier gehen also die Näherungen selbst und nicht deren Funktionswerte in die Rechnung ein.

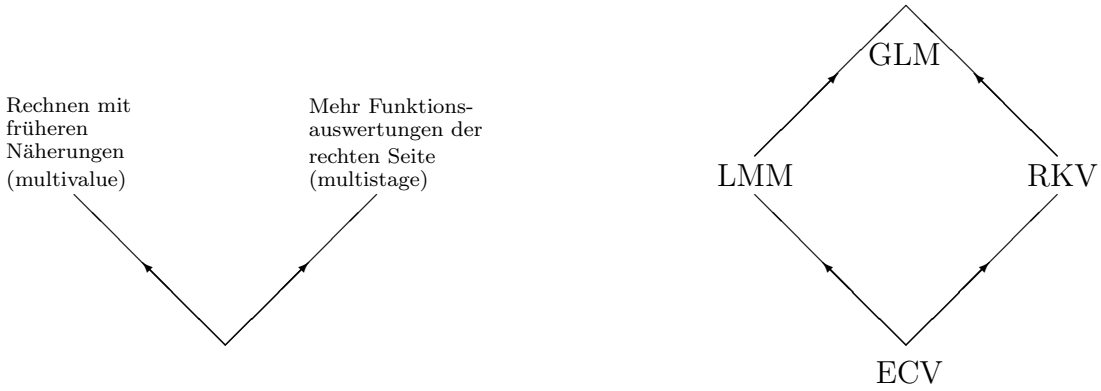

Abbildung 2.1: Verallgemeinerungen des Euler-Cauchy Verfahrens (ECV): Runge-Kutta Verfahren (RKV) investieren mehr f-Auswertungen pro Schritt. Lineare Mehrschrittverfahren (LMM) greifen auf frühere Approximationen zurück. Allgemeine lineare Verfahren (GLM) verwenden beide Ansätze.

Es entstanden viele Varianten dieser Verfahren. Trotzdem lassen sich die meisten Verfahren für gewöhnliche Differentialgleichungen und vielleicht sogar die Wissenschaftler, die sich mit der Numerik dieser Gleichungen beschäftigen, grob nach *multistage* und *multivalue* einteilen. Allgemeine lineare Verfahren, kurz GLMs (**G**eneral **L**inear **M**ethods), stellen eine Verallgemeinerung beider klassischen Verfahren dar. Die Idee, die Konzepte von Runge-Kutta und linearen Mehrschrittverfahren zu verbinden, geht auf Mitte der 60ziger Jahre des letzten Jahrhunderts zurück. Das Hauptziel bestand in der Entwicklung neuer Verfahren, welche die Vorteile der klassischen Verfahren besitzen, die Nachteile jedoch nicht mehr aufweisen.
Dabei liegen die Vorteile von Runge-Kutta Verfahren vor allem in den guten Stabilitätseigenschaften auch von Verfahren hoher Ordnung und in einer leicht zu realisierenden Schrittweitensteuerung. Ein großer Nachteil sind die hohen Kosten bei der Implementierung, insbesondere bei impliziten Verfahren, bei denen in jedem Iterationsschritt ein im Allgemeinen nichtlineares Gleichungssystem zu lösen ist. Insgesamt sind die Kosten der Implementierung von linearen Mehrschrittverfahren gegenüber denen von Runge-Kutta Verfahren geringer. Jedoch besitzen die linearen Mehrschrittverfahren bekanntermaßen Stabilitätsbarrieren: Es existieren keine A-stabilen Methoden der Ordnung größer 3 und BDF Verfahren sind sogar ab der Ordnung 7 nicht mehr nullstabil. Auch die Schrittweitensteuerung ist sehr aufwendig.
Es entstanden in der Zeit von 1964-1966 die so genannten *Pseudo Runge-Kutta Verfahren*, welche auch die Stufenwerte vorangegangener Schritte zur Berechnung der neuen Approximation benutzen, und die *Hybrid Methoden*, die von Butcher *Modifizierte Mehrschrittverfahren* genannt wurden (vgl. [HNW] S.430 und [B] S.115 für einen Überblick und die entsprechenden Referenzen). Die Hybrid Methoden sind ei-

ne Verallgemeinerung der Prädiktor-Korrektor Verfahren und berechnen zusätzliche Prädiktoren, die typischerweise nicht auf dem Gitter liegen (vgl. [B] S.117).
Eine bisher alle genannten Methoden umfassende Klasse ist die der allgemeinen linearen Verfahren. Butcher führte diese Verfahren 1966 ein, um eine einheitliche theoretische Untersuchung der in dieser Klasse enthaltenden Verfahren zu ermöglichen (vgl. [B66]). Zudem sollten durch dieses Konzept auch neue Verfahren entwickelt werden. Ein Vorteil dieser Klasse ist in ihrer Allgemeinheit zu sehen. Sie umfasst neben den oben erwähnten Methoden auch Nordsieck-, klassische Prädiktor-Korrektor- und zyklische Mehrschrittverfahren (vgl. [HNW] III. 6 und S.431-434):

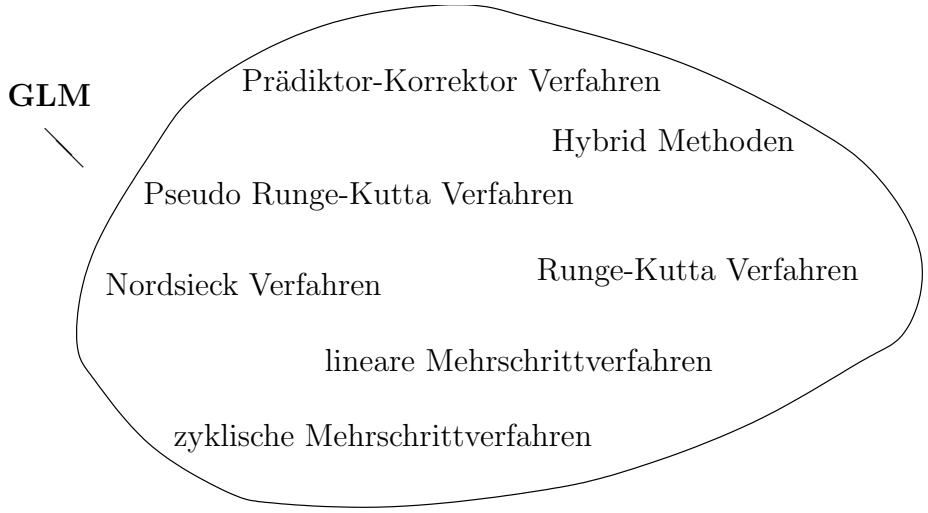

Abbildung 2.2: Die Klasse der allgemeinen linearen Verfahren

Andererseits liegt tatsächlich in dieser Allgemeinheit auch ein Nachteil der Klasse. Während Runge-Kutta und lineare Mehrschrittverfahren jeweils als Verallgemeinerung des Euler-Cauchy Verfahrens hergeleitet und motiviert werden können, sind die allgemeinen linearen Verfahren nicht anschaulich zu motivieren und wirken daher sehr abstrakt. Daran liegt es unter anderem, dass sich diese Verfahren nicht wirklich gut durchsetzten. Zudem stellt die allgemeine Formulierung der GLMs eine Fülle an Parametern bereit, welche die Entwicklung von praktisch brauchbaren Methoden zunächst unmöglich scheinen ließ. Es ist wahrscheinlich zuletzt der Hartnäckigkeit von Butcher zu verdanken, dass die allgemeinen linearen Verfahren nicht nur ein theoretisches Konzept sind, sondern auch praktikable Verfahren entwickelt worden sind und noch immer werden (vgl. [BJ04a, BJ04b, BP06, BR05, BW03]). Dabei geben immer wieder gewisse Eigenschaften, die man an das Verfahren stellt, die Richtung vor, in welche man die Verfahren konstruiert. Hier sind zum einen die Einführung der Diagonalstrukur, welche zu den bekannten DIMSIMs (**D**iagonally **I**mplicit **M**ultistage **I**ntegration **M**ethods) führte (vgl. [B93]), und zum anderen die Forderung nach Runge-Kutta Stabilität (vgl. [B01]) als wichtige Meilensteine zu

nennen. Die jüngsten Erfolge in der Entwicklung neuerer Verfahren für steife Differentialgleichungen motivieren zum einen, diese Klasse weiter zu untersuchen und zum anderen, die Verfahren auch für Differential-Algebraische Gleichungen zugänglich zu machen. Ziel dieser Arbeit ist eine ausführliche Analyse allgemeiner linearer Verfahren für Index-2 DAEs in Hessenberg Form und verwandter Gleichungen.

Doch bevor wir nun allgemeine lineare Verfahren definieren und grundlegende Eigenschaften angeben, wollen wir hilfreiche Schreibweisen anhand der bekannten Runge-Kutta Verfahren einführen.

Runge-Kutta Verfahren

Sei das folgende Anfangswertproblem gegeben:

$$y' = f(x, y), \qquad y(x_0) = y_0 \in \mathbb{R}^N.$$

Bei einem s-stufigen Runge-Kutta Verfahren für diese Gleichung werden ausgehend von $y^{[0]} = y_0$ weitere Approximationen wie folgt berechnet:

$$y^{[n+1]} = y^{[n]} + h(b^T \otimes I_N) f(x_n + ch, Y) \in \mathbb{R}^N.$$

Die Stufenwerte lösen dabei das im Allgemeinen nichtlineare Gleichungssystem

$$Y = \mathbb{1}_s \otimes y^{[n]} + h(A \otimes I_N) f(x_n + ch, Y) \in \mathbb{R}^{sN}.$$

Die Gitterpunkte sind bei einem äquidistanten Gitter definiert durch $x_n = x_0 + nh$. Wir haben in dieser Darstellung das Kronecker Produkt, die Supervektoren

$$Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_s \end{bmatrix}, \qquad f(x_n + ch, Y) = \begin{bmatrix} f(x_n + c_1 h, Y_1) \\ \vdots \\ f(x_n + c_s h, Y_s) \end{bmatrix}$$

und den Vektor $\mathbb{1} = (1, ..., 1)^T$ mit allen Komponenten identisch 1 benutzt. Das Kronecker Produkt zweier Matrizen $B \in \mathbb{R}^{M,N}$ und $C \in \mathbb{R}^{k,l}$ ist definiert durch

$$B \otimes C := \begin{pmatrix} b_{11}C & b_{12}C & \cdots & b_{1N}C \\ \vdots & \vdots & & \vdots \\ b_{M1}C & b_{M2}C & \cdots & b_{MN}C \end{pmatrix} \in \mathbb{R}^{kM,lN}.$$

Eine einfache, aber wichtige Folge dieser Definition ist:

Lemma 2.1
(i) *Für Matrizen $B, \tilde{B}, C$ und $\tilde{C}$ beliebiger Größe gilt*

$$(B \otimes C) \cdot (\tilde{B} \otimes \tilde{C}) = B\tilde{B} \otimes C\tilde{C},$$

solange die auftretenden Matrixmultiplikationen wohldefiniert sind.
(ii) *Für invertierbare Matrizen B und C ist auch das Kronecker Produkt $B \otimes C$ invertierbar, und es gilt*

$$(B \otimes C)^{-1} = B^{-1} \otimes C^{-1}.$$

Die Kronecker Schreibweise ermöglicht eine übersichtliche Handhabung, ohne sich dabei in überflüssigen Indizes zu verlieren.

Die Koeffizienten $A = (a_{ij})$, $b^T = (b_1, ..., b_s)$ und $c^T = (c_1, ..., c_s)$ können übersichtlich in einem Butcher-Tableau angeordnet werden:

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array},$$

wobei c als Knotenvektor, A als Verfahrensmatrix und b als Wichtungsvektor bezeichnet werden.

Stufenordnung: Eine sehr wichtige Größe im Zusammenhang mit steifen und Differential-Algebraischen Gleichungen ist die Stufenordnung eines Runge-Kutta Verfahrens:

Definition 2.2 *Ein Runge-Kutta Vefahren besitzt die Stufenordnung q, wenn für die Lösung $y(x)$ der Anfangswertaufgabe eingesetzt in das Gleichungssystem der Stufenwerte gilt:*

$$y(x_n + ch) = \mathbb{1}_s \otimes y(x_n) + h(A \otimes I_N)y'(x_n + ch) + \mathcal{O}(h^{q+1}). \tag{2.1}$$

Dabei ist der Supervektor $y(x_n + ch)$ definiert durch

$$y(x_n + ch) = \begin{bmatrix} y(x_n + c_1 h) \\ \vdots \\ y(x_n + c_s h) \end{bmatrix}.$$

Die Ableitung $y'(x_n + ch)$ ist entsprechend definiert.

An dieser Stelle nehmen wir uns die Zeit, die Bedeutung der Landau-Symbole genau zu klären: Für eine Funktion $f : J \to \mathbb{R}^n$ definieren wir

$$f = \mathcal{O}(h^p) \quad \Leftrightarrow \quad \lim_{h \to 0} \frac{\|f(h)\|}{h^p} = konst.,$$

wobei 0 im Intervall J liegt und $\|.\|$ eine beliebige Norm auf $\mathbb{R}^n$ ist. Dabei wird $\mathcal{O}$ das *große Landau-Symbol* genannt. Wir verwenden das *kleine Landau-Symbol*, wenn die Konstante gleich Null ist:

$$f = o(h^p) \quad \Leftrightarrow \quad \lim_{h \to 0} \frac{\|f(h)\|}{h^p} = 0.$$

Wir werden im Folgenden $o(1)$ auch unabhängig von h als beliebig kleine Konstante verwenden.

Eine sehr leichte Folgerung aus Definition 2.1 ist das folgende Lemma (vgl. [HNW] S.210):

Lemma 2.3 *Besitzt ein Runge-Kutta Verfahren die Stufenordnung q, so gilt für die Lösung*
$Y(h, y(x_n))$ von

$$Y = \mathbb{1}_s \otimes y(x_n) + h(A \otimes I_N) f(x_n + ch, Y) \tag{2.2}$$

die Darstellung

$$Y(h, y(x_n)) - y(x_n + ch) = \mathcal{O}(h^{q+1}).$$

Beweis: Zunächst ersetzen wir in (2.1) nach dem Willen der Differentialgleichung die Ableitung $y'(x_n + ch)$ durch $f(x_n + c_i h, y(x_n + c_i h))$. Subtraktion der erhaltenen Gleichung von (2.2) und Lipschitz-Stetigkeit der Funktion f mit Lipschitz-Konstante L_f liefern:

$$\|Y - y(x_n + ch)\| \leq hL_f \|A\| \|Y - y(x_n + ch)\| + \mathcal{O}(h^{q+1}).$$

Für hinreichend kleine h folgt die Behauptung.

□

Eine wichtige Konsequenz der Definition der Stufenordnung ist, dass sie unabhängig von der rechten Seite f und somit unabhängig von der Steifheit des Systems nur über die Lösung selbst gewährleistet werden kann (vgl. zum Beispiel die Lösung von (2.3), welche völlig unabhängig von der Steifheit des Systems ist). Im Gegensatz dazu basiert der Beweis des Lemmas auf der Lipschitz-Stetigkeit der rechten Seite f. Was hat diese Tatsache für Konsequenzen?

Um diese Frage zu beantworten, beschreiben wir kurz das Phänomen der Ordnungsreduktion. Prothero und Robinson beobachteten in der Praxis eine Ordnungsreduktion von Runge-Kutta Verfahren angewendet auf steife Differentialgleichungen (vgl. [PR74]). Der Grund dafür liegt in der Abhängigkeit gewisser $\mathcal{O}$-Terme des lokalen Fehlers von der Steifheit des Systems. Theoretisch ist für konstante Steifheit mit $h \to 0$ die klassische Ordnung garantiert. In der Praxis sind beliebig kleine Schrittweiten zu vermeiden, wodurch die Steifheit die Ordnung reduziert. Wir geben dazu

ein Beispiel: Prothero und Robinson betrachten gewöhnliche Differentialgleichungen der Form

$$y' = \lambda(y - \varphi(x)) + \varphi'(x), \qquad y(x_0) = \varphi(x_0) \tag{2.3}$$

mit $\mathrm{Re}(\lambda) \leq 0$. Die Lösung dieser Gleichung ist offenbar die Funktion φ. Wir wenden nun die implizite Mittelpunktsregel auf diese Gleichung an. Sie lautet für allgemeine gewöhnliche Differentialgleichungen

$$\begin{aligned} y^{[n+1]} &= y^{[n]} + hf(x_n + \frac{1}{2}h, Y), \\ Y &= y^{[n]} + h\frac{1}{2}f(x_n + \frac{1}{2}h, Y). \end{aligned}$$

Der lokale Fehler der Mittelpunktsregel angewendet auf die Differentialgleichung (2.3) ist gegeben durch

$$y^{[1]} - \varphi(x_0 + h) = -\frac{2h\lambda}{2 - h\lambda}\mathcal{O}(h^2) + \mathcal{O}(h^3)$$

(vgl. [HW] S. 225f für allgemeine Runge-Kutta Verfahren). Im Grenzfall $h \to 0$ erhalten wir die klassische Ordnung 2. Lassen wir aber gleichzeitig die Steifheit, hier gemessen in $|\lambda|$, gegen Unendlich laufen und zwar mit

$$h\lambda \to \infty,$$

so erhalten wir lediglich die Ordnung 1! Für viele Runge-Kutta Verfahren ist die reduzierte Ordnung von der Größe der Stufenordnung (vgl. [HW] S. 226 Tabelle 15.1).

Wie wir oben gesehen haben, ist die Stufenordnung nicht von diesem Phänomen betroffen, da sie unabhängig von der Steifheit der rechten Seite definiert ist. Die Darstellung von Lemma 2.3 ist jedoch unter dem Grenzprozess $h\lambda \to \infty$ so nicht mehr möglich. Man sollte also im Zusammenhang mit steifen Differentialgleichungen sehr sorgsam mit der Anwendung dieser Darstellung umgehen.

1964 führte Butcher die so genannten *vereinfachenden Annahmen* für Runge-Kutta Verfahren ein (vgl. [B64]):

$$\begin{aligned} B(p): &\quad \textstyle\sum_{i=1}^{s} b_i c_i^{\eta-1} = \frac{1}{\eta} && \eta = 1, ..., p, \\ C(q): &\quad \textstyle\sum_{j=1}^{s} a_{ij} c_j^{\eta-1} = \frac{c_i^{\eta}}{\eta} && i = 1, ..., s, \quad \eta = 1, ..., q, \\ D(\zeta): &\quad \textstyle\sum_{i=1}^{s} b_i c_i^{\eta-1} a_{ij} = \frac{b_j}{\eta}(1 - c_j^{\eta}) && j = 1, ..., s, \quad \eta = 1, ..., \zeta. \end{aligned}$$

Mit Hilfe dieser Bedingungen ist es möglich, implizite Runge-Kutta Verfahren hoher Ordnung und Stufenordnung herzuleiten. Für vorgegebenen Knotenvektor c ist dazu

nur ein lineares Gleichungssystem zu lösen. Die Bedingung $C(q)$ garantiert, dass die Quadraturformel

$$\sum_{j=1}^{s} a_{ij} g(c_j) \approx \int_0^{c_i} g(t)dt$$

die Ordnung q besitzt, für $i = 1, \ldots, s$. Aus ihr folgt daher die Darstellung (2.1) und somit die Stufenordnung q des Runge-Kutta Verfahrens. Zur Bestimmung der Ordnung ist das folgende berühmte Theorem von Butcher hilfreich (vgl. [B64]):

Theorem 2.4 *Gilt für ein Runge-Kutta Verfahren $B(p)$, $C(q)$ und $D(\zeta)$ mit $p \leq 2q + 2$ und $p \leq \zeta + q + 1$, so besitzt das Verfahren die Ordnung p.*

Tatsächlich lassen sich auch implizite Runge-Kutta Verfahren mit hoher Ordnung und guten Stabilitätseigenschaften herleiten. Diese Verfahren sind jedoch so genannte *fully implicit* Runge-Kutta Verfahren. Dies hat hohe Kosten bei der Implementierung zur Folge, da in jedem Integrationsschritt ein im Allgemeinen nichtlineares sN-dimensionales Gleichungssystem mit mehr oder weniger vollbesetzter Matrix A gelöst werden muss. Eine Idee besteht darin, zusätzlich für die Matrix A ein einpunktiges Spektrum zu fordern. Solche Verfahren heißen *singly implicit* Runge-Kutta Verfahren. Durch diese zusätzliche Forderung werden die Kosten der Implementierung durch geeignete Transformationen stark reduziert. Solche Verfahren können mit einer hohen Stufenordnung konstruiert werden, es ergeben sich jedoch daraus einige Nachteile: große Fehlerkonstanten sowie Knoten, die nicht zwischen 0 und 1 liegen (vgl. [B01]). Für kleine Probleme wirkt sich zudem die Transformation ungünstig auf die Gesamtkosten aus. Für solche Probleme bieten sich so genannte *singly diagonally implicit* Runge-Kutta Verfahren an, wobei zusätzlich die Matrix untere Dreiecksstruktur besitzt:

$$A = \begin{bmatrix} \lambda & 0 & \ldots & 0 \\ a_{21} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ a_{s,1} & \ldots & a_{s,s-1} & \lambda \end{bmatrix}.$$

Diese Verfahren haben aber nur Stufenordnung 1 und sind daher für Differential-Algebraische Gleichungen nicht geeignet, bei denen die Stufenordnung eine wichtige Rolle spielt (vgl. Kapitel 3 Abschnitt 3.2.3).

2.1 Definition, Beispiele und lineare Stabilität

Ein allgemeines lineares Verfahren für die Anfangswertaufgabe

$$y' = f(x, y), \qquad y(x_0) = y_0 \in \mathbb{R}^N$$

lautet in der von Burrage und Butcher in [BB80] eingeführten Darstellung

$$\begin{aligned} y^{[n+1]} &= (V \otimes I_N) y^{[n]} + h(B \otimes I_N) f(x_n + ch, Y) \in \mathbb{R}^{rN}, \\ Y &= (U \otimes I_N) y^{[n]} + h(A \otimes I_N) f(x_n + ch, Y) \in \mathbb{R}^{sN}. \end{aligned} \tag{2.4}$$

Wie im Fall von Runge-Kutta Verfahren heißen die s Subvektoren von Y bzw. der Supervektor selbst (innere) Stufenwerte und approximieren die Lösung an den inneren Gitterpunkten:

$$Y_i \approx y(x_n + c_i h).$$

Es ist möglich, die Verfahrensmatrizen $V \in \mathbb{R}^{r,r}$, $B \in \mathbb{R}^{r,s}$, $U \in \mathbb{R}^{s,r}$ und $A \in R^{s,s}$ kompakt in einer $(s+r) \times (s+r)$ Blockmatrix darzustellen:

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right].$$

Es ist aufgrund der besseren Lesbarkeit üblich, jeweils auf das Kronecker Produkt mit der Identität zu verzichten:

$$\begin{aligned} y^{[n+1]} &= V y^{[n]} + hBf(x_n + ch, Y) \in \mathbb{R}^{rN}, \\ Y &= U y^{[n]} + hAf(x_n + ch, Y) \in \mathbb{R}^{sN}. \end{aligned}$$

Wir werden von dieser Schreibweise wenn möglich Gebrauch machen.

Ein wesentlicher Unterschied von GLMs zu Runge-Kutta Verfahren liegt darin, dass nicht nur ein N-dimensionaler Vektor von einem Schritt zum nächsten übergeben wird, sondern r N-dimensionale Vektoren

$$y^{[n]} = \begin{bmatrix} y_1^{[n]} \\ \vdots \\ y_r^{[n]} \end{bmatrix} \in \mathbb{R}^{rN},$$

die so genannten äußeren Stufenwerte. Dies erklärt auch die Größen der Matrizen V, B, U und A. Jedoch wirft der Fall $r > 1$ unmittelbar die Fragen auf:

- Was approximieren die Supervektoren $y^{[n]}$ eigentlich?
- Wie erhält man den Startwert $y^{[0]}$?

Eine weitere Freiheit allgemeiner linearer Verfahren liegt in der Festlegung, was die Iterationen $y^{[n]} \in \mathbb{R}^{rN}$ approximieren sollen. Wir fordern zunächst sehr abstrakt die Existenz einer so genannten *correct value* Funktion

$$y_c : [x_0, x_e] \times [0, h] \to \mathbb{R}^{rN},$$

die an den einzelnen Gitterpunkten $x_n = x_0 + nh$ approximiert wird:

$$y^{[n]} \approx y_c(x_n, h).$$

Die *correct value* Funktion ist dabei nicht mit der Lösung $y(x)$ der Anfangswertaufgabe zu verwechseln, von deren Existenz wir ebenfalls auf dem Intervall $[x_0, x_e]$ ausgehen. Um deutlich zu machen, was sich hinter einer *correct value* Funktion verbirgt, hier zwei typische Formen:

$$y_c(x,h) = \begin{bmatrix} y(x) \\ y(x-h) \\ \vdots \\ y(x-(r-1)h) \end{bmatrix}, \qquad y_c(x,h) = \begin{bmatrix} y(x) \\ hy'(x) \\ \vdots \\ h^{r-1}y^{(r-1)}(x) \end{bmatrix}.$$

Die erste Darstellung steht in einem engen Zusammenhang mit linearen Mehrschrittverfahren, während die zweite von so genannter Nordsieck Form ist.
Nachdem wir nun festgelegt haben, was die Supervektoren $y^{[n]}$ approximieren, ergeben sich die folgenden Definitionen:
Der lokale Fehler ist gegeben durch

$$\delta_{n+1} := Vy_c(x_n,h) + hBf(x_n + ch, Y(h, y_c(x_n,h))) - y_c(x_{n+1},h),$$

wobei wir mit $Y(h,y)$ die Lösung des Gleichungssystems der Stufenwerte bezeichnen, von deren Existenz wir in dieser Definition ausgehen. Der globale Fehler ist definiert durch

$$\Delta_n := y^{[n]} - y_c(x_n,h).$$

In der Schreibweise des globalen Fehlers ist implizit die Abhängigkeit von einer Startprozedur gegeben.

Bei Runge-Kutta Verfahren ist der Startwert der Iteration $y^{[0]}$ der Anfangswert y_0. Bei GLMs benötigen wir dagegen eine so genannte Startprozedur, die sowohl von dem Anfangswert y_0 als auch von der Schrittweite h abhängt:

$$y^{[0]} = \varphi(h, y_0).$$

Die Notwendigkeit einer Startprozedur ist von linearen Mehrschrittverfahren bekannt. Für verschiedene Startprozeduren werden offenbar auch verschiedene Approximationen $y^{[n]}$ generiert, womit der globale Fehler bezüglich einer Startprozedur definiert ist. Wir werden im Folgenden davon ausgehen, dass eine Startprozedur mit

$$\varphi(h, y_0) - y_c(x_0, h) = \mathcal{O}(h^p),$$

p hinreichend groß, existiert. Für die theoretische Untersuchung ist dies ohne Einschränkung möglich (vgl. die Definition 2.10 des lokalen Fehlers). Bei einer Implementierung ohne variable Ordnung ist die Festlegung einer geeigneten Startprozedur jedoch nötig (vgl. Kapitel 4).
Natürlich muss es umgekehrt auch möglich sein, aus den Approximationen $y^{[n]}$ Näherungen einer gewissen Güte für die exakte Lösung $y(x_n)$ zu gewinnen. Das heißt, es muss eine Art Endprozedur geben (vgl. [B] s. 387-388). Wir gehen auch hier davon

aus, dass eine solche Prozedur existiert. In der Praxis stellt dies zumeist kein Problem dar, weil $y(x)$ eine Komponente der *correct value* Funktion ist (vgl. die beiden typischen Formen oben).

Die Stufenordnung ist wie bei den Runge-Kutta Verfahren definiert:

$$y(x_n + ch) = (U \otimes I_N) y_c(x_n, h) + h(A \otimes I_N) y'(x_n + ch) + \mathcal{O}(h^{q+1}).$$

Da die äußeren Stufen Näherungen der *correct value* Funktion darstellen, ist hier das Produkt mit der Lösung durch das Produkt mit der *correct value* Funktion ersetzt worden.
Alle oben getroffenen Aussagen über die Stufenordnung und das Phänomen der Ordnungsreduktion gelten auch für allgemeine lineare Verfahren. Ebenso gilt eine entsprechende Formulierung von Lemma 2.3, wobei wiederum bei steifen Differentialgleichungen auf die Anwendung des Lemmas verzichtet werden sollte.
Eine wichtige Konsequenz aus dem Phänomen der Ordnungsreduktion ist die Entwicklung allgemeiner linearer Verfahren hoher Stufenordnung. Typischerweise gilt $p = q$ und $p = q + 1$. Diese Verfahren sind also geeignet für steife Differentialgleichungen und scheinen somit auch für Differential-Algebraische Gleichungen eine gute Wahl zu sein. Dies wird durch die Ergebnisse von Kapitel 3 bestätigt. Im Kapitel über die Implementierung werden wir jedoch sehen, dass bestehende Programme nicht so einfach auf DAEs angewendet werden können. Dies liegt daran, dass einzelne Bestandteile dieser Codes zum Beispiel die Startprozeduren oder die Fehlerabschätzer nicht eins zu eins auf DAEs übertragbar sind. Sie sind von dem Phänomen der Ordnungsreduktion betroffen! Dies unterstreicht die These, dass numerische Verfahren und auch Computerprogramme für Differential-Algebraische Gleichungen entwickelt werden sollten. Sind diese für DAEs erfolgreich, so können sie auch für steife Differentialgleichungen eingesetzt werden.

Insgesamt ist ein allgemeines lineares Verfahren durch die Verfahrensmatrizen V, B, U und A, durch die *correct value* Funktion und durch die folgenden vier Größen charakterisiert:

- s: Die Anzahl der inneren Stufen.
- r: Die Anzahl der äußeren Stufen.
- p: Die Konvergenzordnung des Verfahrens.
- q: Die Stufenordnung des Verfahrens.

Bemerkung: *Hairer et al. betrachten in Kapitel III.8 des Buches [HNW] noch allgemeinere Integrationsverfahren, die auch multi-step multi-stage multi-derivative Methoden umfassen. Um aber Aussagen über die Konvergenz der Verfahren machen*

zu können, wird für die Verfahrensfunktion dort eine gewisse Lipschitz-Stetigkeit vorausgesetzt (vgl. [HNW] S. 438). Wir werden im nächsten Kapitel sehen, dass gerade hier ein großer Unterschied zwischen gewöhnlichen Differential- und Differential-algebraischen Gleichungen liegt: Während die Verfahrensfunktionen aller vernünftigen Verfahren für gewöhnliche Differentialgleichungen diese Lipschitz-Stetigkeit erfüllen, haben wir bei allgemeinen linearen Verfahren für Index-2 DAEs in Hessenberg Form eine solche Lipschitz-Stetigkeit nicht. Um trotzdem Stabilitäts- und Konvergenzaussagen machen zu können, werden wir die Struktur der Verfahrensfunktion wesentlich ausnutzen. Es ist daher nicht sinnvoll und auch nicht möglich die Analysis des nächsten Kapitels auf diese allgemeinen Integrationsverfahren anzuwenden.

Doch kommen wir nun konkret zu Beispielen allgemeiner linearer Verfahren.

Runge-Kutta Verfahren

Wir betrachten ein (s-stufiges) Runge-Kutta Verfahren mit dem Butcher-Tableau wie oben. Wie wir bereits erwähnten, wird bei einem Runge-Kutta Verfahren nur ein N-dimensionaler Vektor in jedem Schritt berechnet, d.h. es gilt $r = 1$, und die *correct value* Funktion ist gegeben durch $y_c(x, h) = y(x)$. Insgesamt folgt mit der Darstellung eines Runge-Kutta Verfahrens von oben:

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{c|c} A & \mathbb{1}_s \\ \hline b^T & 1 \end{array}\right].$$

Tatsächlich können bestimmte Runge-Kutta Verfahren auch in verschiedener Art und Weise als GLM interpretiert werden (vgl. [B] S. 360f).

Lineare Mehrschrittverfahren

Im Zusammenhang mit steifen Differentialgleichungen und auch mit Differential-Algebraischen Gleichungen spielen innerhalb der linearen Mehrschrittverfahren die BDF Methoden eine herausragende Rolle. Sie sind zum Beispiel gegeben durch:

$$y_{n+1} = \sum_{j=1}^{r} \alpha_j y_{n+1-j} + \beta_0 h f(x_n + h, y_{n+1}).$$

Dies entspricht bei GLM im Grunde dem Gleichungssystem der Stufenwerte; der hier einzige Stufenwert ist dabei identisch mit der neuen Approximation (c=1), d.h. es gilt insbesondere $p = q$. Die früheren r Approximationen werden bei dieser Rechnung

investiert. Interpretiert als GLM ergibt dies:

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{c|cccccc} \beta_0 & \alpha_1 & \alpha_2 & \cdots & \alpha_{r-1} & \alpha_r \\ \hline \beta_0 & \alpha_1 & \alpha_2 & \cdots & \alpha_{r-1} & \alpha_r \\ 0 & 1 & 0 & \cdots & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 & 0 \end{array}\right].$$

Die *correct value* Funktion entspricht dabei

$$y_c(x,h) = \begin{bmatrix} y(x) \\ y(x-h) \\ \vdots \\ y(x-(r-1)h) \end{bmatrix}.$$

Andere lineare Mehrschritt- und auch Prädiktor-Korrektor Verfahren sind in [Wright02] als GLM formuliert.

Mehrschritt Runge-Kutta Verfahren

Die möglicherweise intuitivste Form die klassischen Verfahren miteinander zu verbinden ist durch die Mehrschritt Runge-Kutta Verfahren realisiert. Informationen früherer Schritte, genauer Approximationen der exakten Lösung an früheren Gitterpunkten, werden verwendet, um die Stufenwerte und die neue Approximation zu berechnen:

$$\begin{aligned} y^{[n+1]} &= \sum_{j=1}^{r} v_j y^{[n+1-j]} + h \sum_{j=1}^{s} b_j f(x_n + c_j h, Y_j) \in \mathbb{R}^N, \\ Y_i &= \sum_{j=1}^{r} u_{ij} y^{[n+1-j]} + h \sum_{j=1}^{s} a_{ij} f(x_n + c_j h, Y_j),\ i = 1, ..., s. \end{aligned}$$

Als GLM lautet dieses Verfahren:

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{ccc|ccccc} \multicolumn{3}{c|}{A} & \multicolumn{5}{c}{U} \\ \hline b_1 & \cdots & b_s & v_1 & v_2 & \cdots & \cdots & v_r \\ 0 & \cdots & 0 & 1 & 0 & \cdots & \cdots & 0 \\ \vdots & & \vdots & 0 & \ddots & \ddots & & \vdots \\ \vdots & & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 & 1 & 0 \end{array}\right].$$

Die *correct value* Funktion ist von der Form wie bei den BDF Methoden.

Sonstige Verfahren

William Wright hat in seiner Doktorarbeit zudem angegeben, wie Hybrid Methoden im Allgemeinen als GLMs formuliert werden können, und Butcher gibt in seinem Buch konkrete Beispiele von zyklischen Mehrschrittverfahren, Pseudo Runge-Kutta Verfahren und Hybrid Methoden an (vgl. [Wright02] S. 48f und [B] S.365f).
Eine wichtige Klasse von GLMs, die nicht unmittelbar mit den klassischen Verfahren im Zusammenhang stehen, sind die bereits erwähnten *Diagonally Implicit Multistage Integration Methods*, kurz DIMSIMs.

Beispiel 2.5 *DIMSIM der Ordnung $p = q = 2$ mit $c = [0, 1/2, 1]^T$ (vgl. [Huang05] S.60):*

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{ccc|ccc} \frac{1}{4} & 0 & 0 & 1 & -\frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 1 & 0 & \frac{1}{8} \\ \hline \frac{1}{2} & -\frac{1}{8} & \frac{1}{2} & 1 & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{2} & -\frac{1}{2} & 1 & 0 & 0 & \frac{1}{4} \\ 0 & -2 & 2 & 0 & 0 & 0 \end{array}\right].$$

Die correct value Funktion ist von Nordsieck-Form.

Lineare Stabilität

Numerische Verfahren für steife Differentialgleichungen müssen sich zumindest an der Dahlquistschen Testgleichung

$$y' = \lambda y$$

mit komplexem λ, $\mathrm{Re}(\lambda) < 0$ messen lassen. Die Stabilität eines Verfahrens angewandt auf diesen Typ von Gleichungen wird als lineare Stabilität bezeichnet. Es besteht die Meinung, dass Verfahren mit guten linearen Stabilitätseigenschaften auch beliebige steife Differentialgleichungen zufrieden stellend lösen können. Zumindest sind sie eine Minimalanforderung. Tatsächlich wurden auch für allgemeinere nichtlineare Typen von Gleichungen Stabilitätsbegriffe entwickelt, die uns jedoch in dieser Arbeit nicht weiter interessieren.
Wenden wir auf die Dahlquistsche Testgleichung ein allgemeines lineares Verfahren an, so erhält man nach leichten Umformungen:

$$y^{[n+1]} = (V + h\lambda B(I - h\lambda A)^{-1}U)y^{[n]}.$$

Wir definieren dann die matrixwertige Stabilitätsfunktion durch

$$M(z) = V + zB(I - zA)^{-1}U.$$

Diese Funktion entspricht der bekannten Stabilitätsfunktion von Runge-Kutta Verfahren $R(z) = 1 + zb^T(I - zA)^{-1}\mathbb{1}$.
Lineare Stabilität ist nun mit der Stabilität der Matrix $M(h\lambda)$ verbunden. Wir bezeichnen dabei eine Matrix als stabil, wenn alle Eigenwerte der Matrix innerhalb des Einheitskreises liegen und die Eigenwerte auf dem Rand des Einheitskreises halbeinfach sind. Ein Eigenwert wiederum heißt halb-einfach, wenn die geometrische gleich der algebraischen Vielfachheit ist, oder alternativ, wenn in der Nebendiagonalen der Jordanschen Normalform an der entsprechenden Stelle des Eigenwertes nur Nullen stehen. Eine einfache Charakterisierung der Stabilität einer Matrix S besagt, dass ihre Potenzen beschränkt bleiben, d.h. es existiert eine positive Konstante K mit

$$\|S^n\| \leq K \text{ für alle } n > 0.$$

Um zu verhindern, dass nicht die Stabilität sondern die vorausgesetzte Genauigkeit in einem Code mit variabler Schrittweite letztlich die Wahl der Schrittweite bestimmt, möchte man möglichst unabhängig von h die Stabilität der Matrix $M(h\lambda)$ garantieren. Dies führt zu der von Runge-Kutta bekannten Definition:

Definition 2.6 *Ein allgemeines lineares Verfahren ist A-stabil, wenn für alle $z \in \mathbb{C}^-$ die Matrix $I - zA$ invertierbar und $M(z)$ stabil ist.*

Auch die L-Stabilität lässt sich sinnvoll für GLMs definieren (vgl. [B] Definition 520F).

Definition 2.7 *Ein allgemeines lineares Verfahren ist L-stabil, wenn sie A-stabil ist und zusätzlich $\rho(M(\infty)) = 0$ gilt. Dabei bezeichnet $\rho(S)$ den Spektralradius der Matrix S.*

Wünschenswert wären allgemeine lineare Verfahren mit Stabilitätseigenschaften von Runge-Kutta Verfahren. Wir definieren daher (vgl. [B] S.398):

Definition 2.8 *Ein allgemeines lineares Verfahren besitzt "Runge-Kutta Stabilität" wenn das charakteristische Polynom der Stabilitätsmatrix $\varphi(w, z) = det(wI - M(z))$ die folgende Form besitzt:*

$$\varphi(w, z) = w^{r-1}(w - R(z)).$$

Für ein Verfahren mit Runge-Kutta Stabilität wird die rationale Funktion $R(z)$ Stabilitätsfunktion genannt.

Man setzt also $r - 1$ Eigenwerte gleich 0. Die Betrachtung der linearen Stabilität wird auf die Stabilitätsfunktion $R(z)$ wie im Fall von Runge-Kutta Verfahren zurückgeführt.
Die Herausforderung der letzten zehn Jahre, bestand in der einfache Konstruktion praktikabler allgemeiner linearer Verfahren mit dieser Eigenschaft. Für bestimmte DIMSIMs ist dies gelungen (vgl. [Wright02]).

2.2 Stabilität, Konsistenz und Konvergenz

Die Ausführungen in diesem Abschnitt beziehen sich stark auf Kapitel III.8 des Buches [HNW]. Sie liefern grundlegende Ideen für die Konvergenzanalyse allgemeiner linearer Verfahren für Index-2 DAEs in Hessenberg Form, die wir im nächsten Kapitel durchführen. Wir nehmen uns daher hier die Zeit, die Dinge ausführlicher zu betrachten.

Wir definieren zunächst analog zu linearen Mehrschrittverfahren:

Definition 2.9 *Ein allgemeines lineares Verfahren heißt nullstabil, wenn die Verfahrensmatrix V stabil ist.*

Warum nennen wir in diesem Fall das Verfahren nullstabil? Dies wird sofort ersichtlich, wenn wir die triviale Differentialgleichung $y' = 0$ betrachten und ein allgemeines lineares Verfahren darauf anwenden:

$$y^{[n+1]} = (V \otimes I) y^{[n]}.$$

Stabilität der Matrix V garantiert die Stabilität dieser Iteration. Wie aus der Nullstabiliät des Verfahrens die Stabilität folgt, werden wir im nächsten Kapitel in Abschnitt 3.2.2 über die Stabilität von GLMs für Index-2 DAEs sehen. Wir gehen im Folgenden von der Nullstabilität des Verfahrens aus.

Wir formulieren wie oben bereits geschehen:

Definition 2.10 *Bezeichne $y_c(x,h)$ die correct value Funktion. Dann definieren wir den lokalen Fehler durch*

$$\begin{aligned} \delta_0 &= \varphi(h, y_0) - y_c(x_0, h), \\ \delta_{n+1} &= V y_c(x_n, h) + hBf(x_n + ch, Y(h, y_c(x_n, h))) - y_c(x_{n+1}, h), \end{aligned}$$

wobei wir hier die Lösbarkeit des Gleichungssytems der Stufenwerte voraussetzen und die Lösung mit $Y(h, y_c(x_n, h))$ bezeichnen.

Setzen wir voraus, dass mit h auch der lokale Fehler gegen 0 konvergiert, so folgt für festes $x = x_0 + nh$ die Gleichheit

$$y_c(x, 0) = (V \otimes I) y_c(x, 0).$$

Insbesondere ist also 1 Eigenwert der Matrix V, d.h. es gilt $1 \in \sigma(V)$. Diese Beobachtung wird bei der Definition der Konsistenz eine Rolle spielen, wo wir eine Projektion auf den zugehörigen Eigenraum definieren.

Die Konvergenz und Konvergenzordnung ist wie gewöhnlich über den globalen Fehler definiert:

Definition 2.11 *Ein allgemeines lineares Verfahren ist konvergent, falls für den globalen Fehler* $\Delta_n := y^{[n]} - y_c(x_n, h)$ *gilt*

$$\Delta_n = o(1)$$

mit $x = x_n = x_0 + nh \leq x_e$. *Ein allgemeines lineares Verfahren ist konvergent von der Ordnung* p, *falls für den globalen Fehler sogar gilt:*

$$\Delta_n = \mathcal{O}(h^p).$$

Auch hier haben wir die Lösbarkeit des Gleichungssystems der Stufenwerte und somit die Existenz der Approximationen $y^{[n]}$ vorausgesetzt. Diese Existenz ist insbesondere für hinreichend glatte rechte Seite f der Differentialgleichung wie im Fall von Runge-Kutta Verfahren gegeben. Daher definieren zum Beispiel Butcher und Hairer et al. Konvergenz nur für Differentialgleichungen mit Lipschitz-stetiger rechter Seite f (vgl. [B] S. 372 und [HNW] S.431 bzw. S.391). Wir wollen uns ebenfalls über die Lösbarkeit des Gleichungssystems keine weiteren Gedanken machen und gehen daher von global Lipschitz-stetigem f aus.

Wie für die klassischen Verfahren gilt:

Satz 2.12 *Ein nullstabiles allgemeines lineares Verfahren ist konvergent von der Ordnung* p, *falls der lokale Fehler die Ordnung* p *besitzt, d.h. wenn*

$$\delta_n = \mathcal{O}(h^{p+1})$$

für alle $0 \leq nh \leq konst.$ *gilt.*

□

Der Beweis des Satzes kann völlig analog zum Beweis für lineare Mehrschrittverfahren geführt werden (vgl. [HNW] S.395f).
Für nullstabile Verfahren ist somit Ordnung p des lokalen Fehlers hinreichend für Konvergenz der Ordnung p. Sie ist aber nicht notwendig, wie Skeel an einige Beispielen deutlich macht (vgl. [S76]).

Wir definieren die Konsistenz für allgemeine lineare Verfahren in Anlehnung an [HNW] (vgl. dort S.437):

Definition 2.13 *Ein allgemeines lineares Verfahren ist konsistent der Ordnung* p, *falls gilt:*

$$\begin{aligned} \delta_n &= \mathcal{O}(h^p), \\ (E \otimes I)(\delta_1 + \ldots + \delta_{n+1}) &= \mathcal{O}(h^p) \end{aligned}$$

für $0 \leq nh \leq konst.$

Bemerkung: *Skeel nennt diese Eigenschaft quasi-Konsistenz (vgl. [S76]).*

Die Matrix E ist dabei eine Projektion auf den Eigenraum zum Eigenwert 1 der Matrix V (vgl. Abbildung 2.3). Genauer gilt

$$E := T\mathrm{diag}\{I, 0, \ldots, 0\}T^{-1}, \tag{2.5}$$

wobei $T^{-1}VT$ folgende Blockstruktur besitzt:

$$T^{-1}VT = \mathrm{diag}\{1, \ldots, 1, \zeta_2, \ldots, \zeta_l, J_s\}$$

mit $\zeta_i \neq 1$ und $|\zeta_i| = 1$. Das Spektrum der Matrix J_s liegt dabei innerhalb des Einheitskreises. Eine solche Blockmatrix, zum Beispiel die Jordansche Normalform, und eine entsprechende Transformationsmatrix existieren, da V stabil ist. Die Matrix E bildet also auf den Eigenraum zum Eigenwert 1 ab und zwar parallel zu der direkten Summe der übrigen verallgemeinerten Eigenräume. Dabei ist der verallgemeinerte Eigenraum eines Eigenwertes der Raum maximaler Dimension, der durch zugehörige Eigenvektoren und Hauptvektoren aufgespannt werden kann.

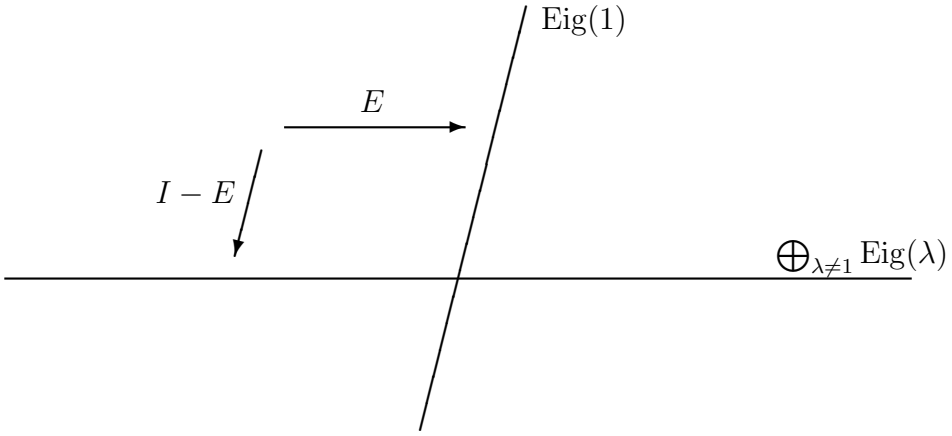

Abbildung 2.3: Die Projektion E bildet auf den Eigenraum zum Eigenwert 1 von V ab. Diese Projektion ist parallel zu der direkten Summe der übrigen verallgemeinerten Eigenräume.

Was rechtfertigt nun die Definition der Konsistenz oben? Ist diese Eigenschaft des lokalen Fehlers weiterhin hinreichend für die Konvergenz der Ordnung p eines nullstabilen Verfahrens? Ist sie sogar notwendig?
Wir gehen davon aus, dass die rechte Seite f der Differentialgleichung und die *correct value* Funktion p-mal stetig differenzierbar sind. Wir können in diesem Fall den lokalen Fehler in eine Taylorreihe in h entwickeln:

$$\delta_{n+1} = \delta^0(x_n) + \delta^1(x_n)h + \ldots + \delta^p(x_n)h^p + \mathcal{O}(h^{p+1}). \tag{2.6}$$

Die Funktionen $\delta^j(x)$ sind dabei $(p - j + 1)$-mal stetig differenzierbar. Unter dieser Voraussetzung sind folgende Aussagen äquivalent (vgl. [HNW] S.437):

- Das allgemeine lineare Verfahren ist konsistent von der Ordnung p.
- Für den lokalen Fehler gilt:

$$\delta_n = \mathcal{O}(h^p) \text{ für } 0 \leq nh \leq konst., \qquad (E \otimes I)\delta^p(x) = 0.$$

- Für den lokalen Fehler gilt:

$$\delta_n = \mathcal{O}(h^p), \qquad (E \otimes I)\delta_{n+1} = \mathcal{O}(h^{p+1}) \text{ für } 0 \leq nh \leq konst. \tag{2.7}$$

Gemäß nach dem Motto

Konsistenz (der Ordnung p) und Stabilität
$\Longleftrightarrow$ Konvergenz (der Ordnung p)

zeigte Hairer et al. unter der Voraussetzung, dass für den lokalen Fehler die Darstellung (2.6) gilt, folgendes Konvergenzresultat (vgl. [HNW] S.429 Theorem 8.13):

Satz 2.14 *Ein nullstabiles allgemeines lineares Verfahren ist genau dann konvergent von der Ordnung p, wenn es konsistent von der Ordnung p ist.*

Um die Dahlquistsche Äquivalenz wirklich vollständig zu begründen, sei darauf hingewiesen, dass auch die Stabilität notwendig für die Konvergenz ist (vgl. [B] Theorem 513A bzw. den Beweis des Theorems).
Mit diesem Satz ist die Definition der Konsistenz oben offenbar gerechtfertigt.

Wir folgern nun die Existenz eines verfahrensrelevanten Vektors über minimale Anforderungen an das allgemeine lineare Verfahren. Dieser Vektor kann daher als gegeben betrachtet werden und spielt eine wichtige Rolle in Kapitel 3.
Wir gehen dazu von der folgenden Form der *correct value* Funktion aus:

$$y_c(x,h) = u \otimes y(x) + v \otimes hy'(x) + \mathcal{O}(h^2). \tag{2.8}$$

Dabei sind u und v Vektoren aus $\mathbb{R}^k$. Die Forderung, dass die Iterationen $y^{[n]}$ eine solche rechte Seite approximieren, bezeichnet Butcher als minimale Forderung (vgl. [B] S.369).

Mit $h \to 0$ erhält man mit der Darstellung (2.8) eingesetzt in das allgemeine lineare Verfahren (2.4) die Präkonsistenz Bedingungen

$$\begin{aligned} V \cdot u &= u, \\ U \cdot u &= \mathbb{1}_s. \end{aligned} \tag{2.9}$$

Der Vektor u wird dabei *Präkonsistenzvektor* genannt.

3 Allgemeine lineare Verfahren für Differential-Algebraische Gleichungen

In diesem Kapitel untersuchen wir zum einen die formale Anwendung allgemeiner linearer Verfahren auf Differential-Algebraische Gleichungen der Form

$$0 = F(x, y', y). \tag{3.1}$$

Wir treffen grundsätzliche Aussagen, die zum Verständnis der Numerik Differential-Algebraischer Gleichungen wesentlich beitragen. Zum anderen beweisen wir konkret und ausführlich die Konvergenz allgemeiner linearer Verfahren angewandt auf Index-2 DAEs in Hessenberg Form

$$\begin{aligned} y' &= f(y, z), \\ 0 &= g(y). \end{aligned} \tag{3.2}$$

Diese Klasse von Gleichungen sind typisch für die Mechanik von Mehrkörpersystemen. Aber auch als theoretische Aussgangsgleichungen sind sie höchst interessant. Tatsächlich sind nämlich die Konvergenzaussagen dieser Probleme auf verwandte Differential-Algebraische Gleichungen übertragbar. Dabei helfen uns die über Gleichungen der Form (3.1) getroffenen Aussagen, um eine möglichst umfassende Klasse von DAEs als "verwandt" zu identifizieren.

Zunächst stellt sich jedoch die Frage, wie allgemeine lineare Verfahren formal auf Differential-Algebraische Gleichungen der allgemeinen Form (3.1) angewendet werden können.
Für eine gewöhnliche Differentialgleichung

$$y' = f(x, y)$$

lautet ein allgemeines lineares Verfahren nach dem vorangegangenen Kapitel (vgl. Gleichung (2.4)) wie folgt:

$$\begin{aligned} y^{[n+1]} &= Vy^{[n]} + hBY', \\ Y &= Uy^{[n]} + hAY', \\ Y' &= f(x_n + ch, Y), \end{aligned}$$

wobei wir hier auf das Kronecker-Produkt mit der Identität verzichtet haben. Zudem wurde die zusätzliche Variable Y' eingeführt, da in dieser Schreibweise nur die untere

Gleichung direkt von dem Problem, der Differentialgleichung, abhängt. Es ist nun leicht einzusehen, wie ein allgemeines lineares Verfahren auch auf die DAE (3.1) angewendet werden kann:

$$\begin{aligned} y^{[n+1]} &= Vy^{[n]} + hBY', \\ Y &= Uy^{[n]} + hAY', \\ 0 &= F(x_n + ch, Y', Y). \end{aligned} \tag{3.3}$$

Diese Form der Anwendung entspricht der "klassischen" Art, wie sie von Hairer et al. für Runge-Kutta Verfahren vorgeschlagen wurde (vgl. [HLR89] S.14f). Aus ihr ergeben sich unmittelbar die folgenden Fragen:

- Besitzt das System (3.3) eine eindeutige Lösung?
- Ist das allgemeine lineare Verfahren für DAEs stabil?
- Welche Approximationsgüte besitzt dieses Verfahren?
- Wie kann ein solches Verfahren implementiert werden?

Wir beantworten diese Fragen im Folgenden ausführlich für allgemeine lineare Verfahren angewendet auf Index-2 DAEs in Hessenberg Form (3.2): Die eindeutige Lösbarkeit von (3.3) ist im Wesentlichen auf die eindeutige Lösbarkeit des im Allgemeinen nichtlinearen Gleichungssystems der Stufenwerte zurückzuführen, welche wir im Abschnitt 3.2.1 nachweisen. Die Stabilität allgemeiner linearer Verfahren für Index-2 DAEs in Hessenberg Form analysieren wir in Abschnitt 3.2.2. Die Konvergenz und Konsistenz dieser Verfahren für solche DAEs untersuchen wir in Abschnitt 3.2.3. Hinweise zur Implementierung sind in Kapitel 4 gegeben.
Am Ende des Kapitels übertragen wir Konvergenzaussagen auf andere DAEs mit Störungsindex 2.

Doch bevor wir uns an die Arbeit machen, die klassischen Fragen der Numerischen Analysis an ein numerisches Verfahren, hier konkret für allgemeine lineare Verfahren angewendet auf Index-2 DAEs, zu beantworten, wollen wir die Konsequenzen der Einführung von Y' diskutieren.

3.1 Die Augmentierung impliziter DAEs

Die Variable $Y' = W$ wird zusätzlich eingeführt, um formal allgemeine lineare Verfahren auf DAEs der Form (3.1) anwenden zu können. Wie lautet ein solches Verfahren, wenn wir diese Variable bereits mit in die Formulierung der DAE einbeziehen? Wir betrachten also das augmentierte System

$$\begin{aligned} y' &= w, \\ 0 &= F(x, w, y). \end{aligned}$$

Ein allgemeines lineares Verfahren für dieses System lautet

$$\begin{aligned} y^{[n+1]} &= Vy^{[n]} + hBY', \\ w^{[n+1]} &= Vw^{[n]} + hBW', \\ Y &= Uy^{[n]} + hAY', \\ 0 &= F(x_n + ch, Y', Y), \\ Y' &= Uw^{[n]} + hAW', \end{aligned}$$

wobei wir die triviale Gleichung $Y' = W$ bereits eingesetzt haben. Wir schränken unsere Betrachtungen nun auf allgemeine lineare Verfahren mit invertierbarer Matrix A ein. Dies ist im Zusammenhang mit DAEs üblich (vgl. [HLR89] S.15). Ist nämlich die Funktion F in (3.1) unabhängig von gewissen Komponenten der Ableitung y', so müssen die entsprechenden Komponenten von Y' allein durch die zweite Gleichung in (3.3) bestimmt werden. Dies ist nur für invertierbare Matrix A möglich. Mit der Invertierbarkeit von A lässt sich nun W' aus der unteren Gleichung ermitteln und in die Iteration für w einsetzen. Wir erhalten demnach

$$\begin{aligned} y^{[n+1]} &= Vy^{[n]} + hBY', \\ w^{[n+1]} &= M(\infty)w^{[n]} + BA^{-1}Y', \\ Y &= Uy^{[n]} + hAY', \\ 0 &= F(x_n + ch, Y', Y). \end{aligned} \tag{3.4}$$

Hier ist $M(\infty) = V - BA^{-1}U$, wobei $M(z) = zB + (I - zA)^{-1}U$ die matrixwertige Stabilitätsfunktion des allgemeinen linearen Verfahrens ist.

Eine wirklich interessante Beobachtung ist, dass die Gleichungssysteme der Stufenwerte für ein allgemeines lineares Verfahren angewendet auf eine implizite DAE der Form (3.1) und angewendet auf das augmentierte System völlig identisch sind:

$$\begin{aligned} Y &= Uy^{[n]} + hAY', \\ 0 &= F(x_n + ch, Y', Y) \end{aligned}$$

(vgl. jeweils die unteren beiden Gleichungen in (3.3) und (3.4)). Damit sind aber auch die Iterationen der y-Komponente identisch:

$$y^{[n+1]} = Vy^{[n]} + hBY'$$

(vgl. jeweils die erste Gleichung in (3.3) und (3.4)). Wir sprechen daher von der Äquivalenz eines allgemeinen linearen Verfahrens angewendet auf eine allgemeine DAE der Form (3.1) und auf das entsprechende augmentierte System (vgl. Abbildung 3.1).

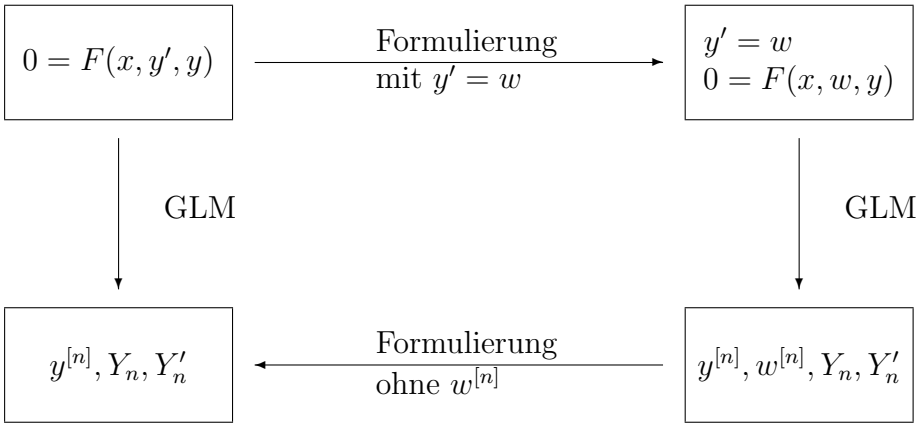

Abbildung 3.1: Äquivalenz allgemeiner linearer Verfahren mit invertierbarer Verfahrensmatrix A angewendet auf eine allgemeine DAE und auf das augmentierte System. Dabei bezeichnen Y_n, Y'_n die Stufenwerte des n-ten Iterationsschritts.

Bemerkung: *Es gibt jedoch andere numerische Verfahren, für welche die oben beschriebene Äquivalenz nicht gilt (vgl. [L89]).*

Bis auf die Approximationen für $y' = w$ lösen also allgemeine lineare Verfahren eine implizite DAE und das entsprechend augmentierte System identisch. Bringt aber die zusätzliche Berechnung der $w^{[n]}$ keine zusätzlichen Schwierigkeiten, so ist das Lösen dieser beiden DAEs gleichermaßen schwierig (vgl. Abbildung 3.2). Diese Konsequenz steht in einem gewissen Widerspruch zu der folgenden Aussage, welche allerdings für allgemeine numerische Verfahren formuliert ist (vgl. [BCP] S.39): "A numerical method may be applied to either the original DAE or to the enlarged system, but it is important to note that the resulting convergence and stability properties of the schemes may be quite different because of the change in the index." Tatsächlich sind minimale Voraussetzungen für Konsistenz, Stabilität und Konvergenz der y-Komponente allgemeiner linearer Verfahren angewendet auf das augmentierte System entsprechende minimale Voraussetzungen des Verfahrens angewendet auf (3.3), solange unter diesen Voraussetzung eine stabile Berechnung der w-Approximationen möglich ist.

Eine weitere wichtige Feststellung ist, dass auch bei der Anwendung der GLM auf (3.1) in der Regel eine Approximation der Ableitung y' nötig ist: Tatsächlich ist das Gleichungssystem der Stufenwerte, wie wir später konkret bei den Index-2 DAEs

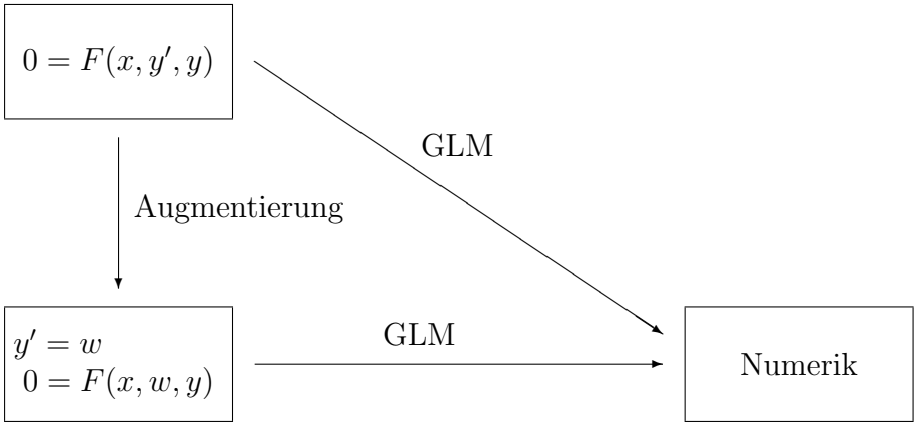

Abbildung 3.2: Bei formaler Anwendung allgemeiner linearer Verfahren mit invertierbarer Verfahrensmatrix A auf eine allgemeine DAE und auf das entsprechende augmentierte System muss mit den gleichen numerischen Schwierigkeiten gerechnet werden.

sehen werden, oft nur in der Nähe der Lösung, d.h. hier für

$$Y - \mathbb{1} \otimes y(x_n) = o(1), \qquad Y' - \mathbb{1} \otimes y'(x_n) = o(1),$$

$o(1)$ hinreichend klein, eindeutig lösbar. Um also zum Beispiel ein vereinfachtes Newton Verfahren zur Berechnung der Lösung des Gleichungssystems zu initialisieren, benötigen wir Näherungen für $y(x_n)$ und $y'(x_n)$. Dies bedeutet, dass eine Strategie zur Berechnung von $w^{[n]}$ angegeben werden muss (vgl. [Voigtmann06] Abschnitt 11.1 Newton Iteration).

Fazit: Die numerischen Schwierigkeiten, mit denen man beim Lösen einer Gleichung der Form (3.1) rechnen muss, sind formal die gleichen Schwierigkeiten, die beim Lösen des augmentierten Systems vorliegen. Diese Beobachtung ist umso wichtiger, da sich bei einer Augmentierung der Index tatsächlich erhöhen kann. Eine häufig vertretene Ansicht lautet (vgl. das Zitat oben [BCP] S.39): Vorsicht, eine Augmentierung der DAE kann den Index und somit die numerischen Schwierigkeiten erhöhen! Mit den Überlegungen oben sollte man besser formulieren: Vorsicht, das implizite System kann zu denselben numerischen Schwierigkeiten führen wie das augmentierte System!

Wie groß diese Schwierigkeiten konkret sind, lässt sich nicht mehr formal begründen. Wir zeigen dies im Folgenden an einigen Beispielen auf. Die formale Augmentierung war nämlich in dem Sinne naiv, dass wir alle Ableitungen durch Einführung neuer Variablen ersetzt haben. Bei konkreten DAEs sollte nur eine sorgsam ausgewählte Ersetzung erfolgen, um den Index nicht unnötig zu erhöhen.

Beispiel: *Die Überlegungen über die formale Anwendung allgemeiner linearer Verfahren auf Gleichungen der Form* (3.1) *sind nicht auf Differential-Algebraische Gleichungen beschränkt. Die Simulation elektrischer Schaltkreise zum Beispiel führt auf quasilineare Differentialgleichungen*

$$C(U)U' = F(t, U),$$

bei denen $C(U)$ invertierbar sein kann. Wir können in diesem Fall die Gleichung mit $C(U)^{-1}$ multiplizieren und erhalten eine gewöhnliche Differentialgleichung, welche mit bekannten numerischen Verfahren gelöst werden kann. Dabei muss $C(U)^{-1}$ nicht einmal explizit berechnet werden, stattdessen wird nach jeder Auswertung von F und C ein lineares Gleichungssystem gelöst. Insbesondere ist die sensitive Abhängigkeit der Ableitung von Störungen in der Gleichung nicht relevant. Jedoch werden dabei gewisse Strukturen zerstört, weshalb eine direkte Anwendung auf die implizite Form ratsam ist (vgl. [GP84] 1. Introduction, [HW] VI.6). Der Grund liegt hauptsächlich darin, dass sich die Ableitung von $C(U)^{-1}$ ungünstig auf das Newton-Verfahren zum Lösen des Gleichungssystems der Stufenwerte auswirken kann. Die Gleichung "implizit" zu behandeln, bedeutet für allgemeine lineare Verfahren nach den Überlegungen oben aber, dass wir im Grunde das augmentierte System

$$\begin{aligned} U' &= W, \\ 0 &= F(t, U) - C(U)W \end{aligned}$$

lösen. Dies ist jedoch eine semi-explizite DAE vom Index 1. Die Definition des Störungsindexes lässt vermuten, dass dies trotzdem unproblematisch ist.

Beispiel: *Wir betrachten die folgende implizite DAE:*

$$\begin{aligned} y' &= f_z u', & y(0) &= y_0, \\ 0 &= u - g(y), & u(0) &= u_0. \end{aligned}$$

Hier ist f_z eine konstante Matrix entsprechender Größe. Zudem sei $I - Dgf_z$ in einer Umgebung der Lösung invertierbar. Eine Augmentierung ist hier nur durch die Einführung einer Hilfsvariablen für u' vorzunehmen:

$$\begin{aligned} y' &= f_z z, & y(0) &= y_0, \\ u' &= z, & u(0) &= u_0, \\ 0 &= u - g(y), & z(0) &= u'(0). \end{aligned}$$

Dies ist eine Index-2 DAE in Hessenberg Form, wobei die Invertierbarkeit von $I - Dgf_z$ gerade der Index-2 Bedingung entspricht. Die implizite Formulierung besitzt dagegen den Index 1 (vgl. [ASW95] Theorem 1b)). Wir sehen im nächsten Unterkapitel, dass die Berechnung der zusätzlichen Approximationen $z^{[n]}$ für allgemeine lineare Verfahren mit $\rho(M(\infty)) < 1$ keine weiteren numerischen Schwierigkeiten

bringt. Die implizite DAE besitzt also für solche GLMs eigentlich den Störungsindex 2!

Das folgende sehr bekannte Beispiel von Gear und Petzold (vgl. [GP84]) dient in der Literatur als Problem, bei dem numerische Verfahren scheitern (vgl. [GP83, AP, HLR89, Voigtmann06]).

Beispiel 3.1 *Wir betrachten die lineare DAE*

$$\begin{pmatrix} 0 & 0 \\ 1 & \eta x \end{pmatrix} \begin{pmatrix} y' \\ z' \end{pmatrix} + \begin{pmatrix} 1 & \eta x \\ 0 & 1+\eta \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} = \begin{pmatrix} f(x) \\ g(x) \end{pmatrix}.$$

Dies ist ein implizites Index-2 System, wie leicht einzusehen ist:
Differenzieren wir die erste Komponente

$$y' + \eta z + \eta x z' = f'(x)$$

und setzen dies in die zweite Komponente ein, so erhalten wir

$$z = g(x) - f'(x).$$

Mit $y = f(x) - \eta x z$ ergibt sich unabhängig von $\eta \neq 0$, dass sowohl y als auch z Index-2 Variablen im Sinne des Differentiations- und Störungsindexes sind.
Wir ersetzen nun die Ableitungen im System durch neue Variablen und betrachten das augmentierte System

$$\begin{aligned} y' &= u, \\ z' &= w, \\ 0 &= \begin{pmatrix} 0 & 0 \\ 1 & \eta x \end{pmatrix} \begin{pmatrix} u \\ w \end{pmatrix} + \begin{pmatrix} 1 & \eta x \\ 0 & 1+\eta \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} - \begin{pmatrix} f(x) \\ g(x) \end{pmatrix}. \end{aligned}$$

Wie oben finden wir durch Differentiation

$$0 = y' + \eta z + \eta x z' - f'(x) = u + \eta z + \eta x w - f'(x)$$

und $z = g - f'$. Differenzieren wir dies ein weiteres Mal, so erhalten wir

$$w = z' = g'(x) - f''(x).$$

Somit handelt es sich für $\eta \neq 0$ bei w und $u = f'(x) - \eta z - \eta x w$ um Index-3 Variablen und bei z und $y = f(x) - \eta x z$ weiterhin um Index-2 Variablen. Dies gilt wieder sowohl für den Differentiations- als auch für den Störungsindex.
Tatsächlich wäre die Einführung von $z' = w$ als zusätzliche Gleichung ausreichend gewesen, um ein allgemeines lineares Verfahren anwenden zu können. Die Ableitung von y ist nämlich explizit gegeben durch

$$y' = g - \eta x z' - (1+\eta) z.$$

Diese reduzierte Augmentierung führt aber in diesem Fall ebenfalls zu einem expliziten Index-3 System.
Hairer et al. zeigen für Runge-Kutta Verfahren, im Besonderen für das implizite Euler-Cauchy Verfahren, dass es für $\eta < -1/2$ zu numerischen Schwierigkeiten kommt (vgl. [HLR89] S.22). Ihre Argumentation basiert auf den Stufenwerten Z'. Dies sind genau die zusätzlichen Variablen, um ein allgemeines lineares Verfahren anwenden zu können. Es ist also eine Index-3 Variable, die das numerische Verfahren zum Scheitern bringt!

Die drei vorangegangenen Beispiele besitzen in ihrer impliziten Form in der oben aufgeführten Reihenfolge die Indizes 0, 1 und 2. Eine Augmentierung, welche die Anwendbarkeit eines allgemeinen linearen Verfahrens ermöglicht, erhöht den Index jeweils um 1. Dies führt im ersten Beispiel auf eine Index 1 DAE, die für allgemeine lineare Verfahren mit invertierbarer Matrix A und $\rho(M(\infty)) < 1$ ohne Probleme gelöst werden kann (vgl. für Runge-Kutta Verfahren [HW] Theorem 1.1 S.380, welches sich entsprechend auch für allgemeine lineare Verfahren formulieren lässt). Das zweite Beispiel ist in der augmentierten Form eine Index-2 DAE in Hessenberg Form. Für diese Differential-Algebraischen Gleichungen werden wir im nächsten Unterkapitel die Durchführbarkeit, die Stabilität, die Konsistenz und Konvergenz allgemeiner linearer Verfahren ausführlich untersuchen. Es wird sich herausstellen, dass insbesondere allgemeine lineare Verfahren mit invertierbarer Matrix A und $\rho(M(\infty)) < 1$ zur Lösung solcher DAEs geeignet sind. Das dritte und letzte Beispiel oben bestätigt, dass erst das entsprechend augmentierte System über den Index und somit über die numerischen Schwierigkeiten entscheidet. Wir betrachten dieses Beispiel nochmal etwas genauer. Dabei wird deutlich, dass bei einer äquivalenten Formulierung die entsprechende Augmentierung nicht zur Indexerhöhung führt:

Beispiel 3.2 *Die DAE aus Beispiel 3.1 ist offenbar äquivalent zu:*

$$\begin{pmatrix} 0 \\ 1 \end{pmatrix} \left[\begin{pmatrix} 1 & \eta x \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} \right]' + \begin{pmatrix} 1 & \eta x \\ 0 & 1 \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} = \begin{pmatrix} f(x) \\ g(x) \end{pmatrix}.$$

Wir führen eine neue Variable u ein und erhalten

$$\begin{aligned} \begin{pmatrix} 0 \\ 1 \end{pmatrix} u' + \begin{pmatrix} 1 & \eta x \\ 0 & 1 \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} &= \begin{pmatrix} f(x) \\ g(x) \end{pmatrix}, \\ u &= \begin{pmatrix} 1 & \eta x \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix}. \end{aligned}$$

Wie in Beispiel 3.1 lässt sich einfach nachweisen, dass y und z für $\eta \neq 0$ Index-2 Variablen sind. Das Interessante an dieser Formulierung ist, dass $u = f(x)$ eine Index-1 Variable ist. Somit bleibt das System auch bei Einführung der zusätzlichen Variablen $w = u'$ ein Index-2 System. Eine unnötige Augmentierung durch Hinzunahme weiterer Hilfsvariablen für y' und z' würde den Index erhöhen.

Dieses Beispiel zeigt, dass eine Augmentierung des Systems den Störungsindex insgesamt nicht verändern muss. Dabei sollten so wenige Ableitungen wie möglich bzw. nur sorgfältig ausgewählte Ableitungen durch neue Variablen ersetzt werden, um den Index nicht unnötig zu erhöhen. Andererseits muss die formale Anwendbarkeit der GLM garantiert sein.
So trivial dieses Beispiel auch scheint, so lässt sich daran eine großartige Idee verdeutlichen, die Idee des *properly stated leading terms* (vgl. [M01, M02b]). Wir gehen darauf in Unterkapitel 3.3 und Anhang A.1 genauer ein.

Zusammenfassung: Die Augmentierung einer impliziten DAE kann den Index erhöhen. Dies könnte zum einen dazu verleiten die augmentierten Systeme lieber nicht zu betrachten und zum anderen nur implizite DAEs mit niedrigerem Index zu integrieren. Tatsächlich lösen allgemeine lineare Verfahren beide Systeme äquivalent. Daher muss beim Lösen der impliziten DAE mit den gleichen numerischen Schwierigkeiten wie bei dem augmentierten System gerechnet werden. Das Vorgehen ist daher Folgendes: Formuliere die Differential-Algebraische Gleichung in der Weise, dass eine entsprechende Augmentierung den Index nicht erhöht. Dabei ist eine entsprechende Augmentierung die Ersetzung möglichst weniger bzw. sorgfältig ausgewählter impliziter Komponenten der Ableitung durch Hilfsvariablen, so dass ein allgemeines lineares Verfahren formal anwendbar ist.

3.2 Index-2 DAEs in Hessenberg Form

Wir betrachten in diesem Unterkapitel autonome Index-2 DAEs in Hessenberg Form

$$\begin{array}{rcll} y' & = & f(y,z), & y(0) = y_0 \in \mathbb{R}^N, \\ 0 & = & g(y), & z(0) = z_0 \in \mathbb{R}^l, \end{array} \tag{3.5}$$

wobei wir die Invertierbarkeit der Matrix $Dg(y)f_z(y,z)$ in einer Umgebung der Lösung voraussetzen. Diese Invertierbarkeit entspricht der so genannten Index-2 Bedingung. Des Weiteren seien die Funktionen f und g einmal bzw. zweimal stetig differenzierbar. Wir gehen zudem von der Existenz der Lösung $(y(x), z(x))$ auf dem Intervall $[0, x_e]$ aus. Zusammenfassend setzen wir also voraus:

Die Matrix $\boldsymbol{Dg(y)f_z(y,z)}$ ist invertierbar. (DAE1)

Die Funktion $\boldsymbol{f}$ ist einmal stetig differenzierbar und $\boldsymbol{g}$ zweimal. (DAE2)

Es existiert eine eindeutige Lösung $\boldsymbol{(y(x), z(x))}$ auf $\boldsymbol{[0, x_e]}$. (DAE3)

Alle Aussagen dieses Unterkapitels lassen sich ohne großen Aufwand auch für nichtautonome Index-2 DAEs in Hessenberg Form formulieren. Wir betrachten jedoch aufgrund einer übersichtlicheren Darstellung den autonomen Fall.

Wie wir in Kapitel 1 gesehen haben, sind Index-2 DAEs in Hessenberg Form, insbesondere die GGL-Formulierung mechanischer Mehrkörpersysteme, typische Gleichungen der Mechanik. Zudem können, wie wir dort bereits erwähnten, einige Gleichungen in diese Form transformiert werden: Gewöhnliche Differentialgleichungen mit Invarianten, quasilineare Systeme der Form

$$B(y)y' = f(y),$$

wie sie bei der Schaltungssimulation und der chemischen Reaktionskinetik vorliegen. Diese Probleme haben als Index-2 DAE in Hessenberg Form formuliert gemeinsam, dass sie von der algebraischen Komponente nur linear abhängen, das heißt es gilt:

$$f(y,z) = f_0(y) + f_z(y)z.$$

Für Index-2 Gleichungen in Hessenberg Form mit einer solchen Funktion f ließ Abschätzung (1.4) mit $\mu = 0$ vermuten, dass die numerische Behandlung einfacher ist als im allgemeinen Fall. Wir werden in Abschnitt 3.2.1 sehen, wo genau diese Struktur Vorteile bringt. Ist die Matrix $f_z(y) = f_z$ sogar konstant, so handelt es sich bei y um eine Index-1 Variable, wie wir in Kapitel 1 bereits feststellten. Auf diesen Spezialfall gehen wir ausführlich in Abschnitt 3.2.2 ein.
Weitere Index-2 DAEs in Hessenberg Form treten in der *optimal control theory* und bei *trajectory prescribed path control problems* auf (vgl. [BCP]). Aber auch als

"typische" Index-2 DAE sind Gleichungen der Hessenberg Form sehr interessant: Die in diesem Unterkapitel erzielten Resultate lassen sich auf andere Index-2 DAEs übertragen (vgl. Unterkapitel 3.3).

Ein allgemeines lineares Verfahren für die DAE (3.5) lautet:

$$\begin{aligned} y^{[n+1]} &= (V \otimes I_N) y^{[n]} + h(B \otimes I_N) f(Y, Z), \\ z^{[n+1]} &= (V \otimes I_l) z^{[n]} + h(B \otimes I_l) Z', \\ Y &= (U \otimes I_N) y^{[n]} + h(A \otimes I_N) f(Y, Z), \\ Z &= (U \otimes I_l) z^{[n]} + h(A \otimes I_l) Z', \\ 0 &= g(Y). \end{aligned} \tag{3.6}$$

Dabei sind die Startwerte der Iteration durch eine Startprozedur φ gegeben:

$$\begin{pmatrix} y^{[0]} \\ z^{[0]} \end{pmatrix} = \begin{pmatrix} \varphi^y(h, y_0, z_0) \\ \varphi^z(h, y_0, z_0) \end{pmatrix} = \varphi(h, y_0, z_0).$$

Die unteren drei Gleichungen in (3.6) bilden das Gleichungssystem der Stufenwerte. Die Voraussetzungen an das allgemeine lineare Verfahren, welche wir ausführlich in den nächsten Abschnitten motivieren, lauten:

Die Matrix A ist regulär. (GLM1)

Die Matrix V ist stabil. (GLM2)

Die Matrix $M(\infty)$ ist stabil. (GLM3)

Dabei ist $M(\infty) = V - BA^{-1}U$ und $M(z) = V + zB(I - zA)^{-1}U$ die matrixwertige Stabilitätsfunktion des Verfahrens. Unter der Invertierbarkeitsvoraussetzung (GLM1) lässt sich die vorletzte Gleichung in (3.6) nach Z' auflösen, und wir erhalten für die z-Komponente die Iteration

$$z^{[n+1]} = (M(\infty) \otimes I_l) z^{[n]} + (BA^{-1} \otimes I_l) Z.$$

Zusätzlich gehen wir von folgender Darstellung der *correct value* Funktionen aus:

$$\begin{aligned} \boldsymbol{y_c(x, h)} &= \boldsymbol{u \otimes y(x) + v \otimes h y'(x) + \mathcal{O}(h^2)}, \\ \boldsymbol{z_c(x, h)} &= \boldsymbol{u \otimes z(x) + \mathcal{O}(h)}. \end{aligned} \tag{GLM4}$$

Dabei gilt für den Präkonsistenzvektor u die Gleichung

$$Uu = \mathbb{1}$$

(vgl. (2.8) und (2.9)).

Wir werden in den folgenden Abschnitten die klassischen Fragen der Numerischen Analysis an ein numerisches Verfahren konkret für allgemeine lineare Verfahren angewandt auf Index-2 DAEs in Hessenberg Form beantworten.

3.2.1 Das Gleichungssystem der Stufenwerte

In diesem Abschnitt weisen wir nach, dass die Stufenwerte Y, Z unter vernünftigen Voraussetzungen auf stabile Art und Weise, das heißt durch ein stabiles numerisches Modell, berechnet werden können. Dabei nennen wir ein numerisches Modell $T_h(y) = 0$ stabil, wenn "kleine" Fehler bei der Berechnung einer Näherungslösung keine "großen" Auswirkungen auf die Güte dieser Näherung haben, oder genauer, falls für den Operator T_h eine Stabilitätsungleichung der Form

$$\|y - \bar{y}\| \leq C\|T_h y - T_h \bar{y}\|$$

gleichmäßig für $0 \leq h \leq h_0$ gilt. Die Existenz eines solchen stabilen Modells ist notwendig für eine Konvergenzanalyse allgemeiner linearer Verfahren, bei welcher insbesondere der Grenzfall $h \to 0$ betrachtet wird.
Wir zeigen aber zudem, dass Störungen der DAE, genauer der algebraischen Gleichung, multipliziert mit dem Faktor $1/h$ in diese Rechnung eingehen. Trotz der stabilen Berechnung der Stufenwerte kann dieses Modell also für zu kleine Werte von h in der Praxis keine zufrieden stellende Näherungslösung liefern. Daher werden die Stufenwerte zum Beispiel in einer Implementierung tatsächlich über das ursprüngliche Gleichungssystem der Stufenwerte bestimmt, obwohl dieses Gleichungssystem für $h \to 0$ beliebig schlecht konditioniert ist, wie wir am Ende des Abschnitts sehen werden.

Das Gleichungssystem der Stufenwerte besitzt die folgende Struktur:

$$\begin{aligned} 0 &= Y - \eta - hAf(Y, Z), \\ 0 &= Z - \xi - hAZ', \\ 0 &= g(Y). \end{aligned}$$

Dabei haben wir zugunsten einer besseren Lesbarkeit auf das Kronecker Produkt mit der Identität verzichtet (vgl. die drei unteren Gleichungen in (3.6)). Wir gehen von der Glattheitsvorausetzung (DAE2) aus.

Bemerkung: *Im Fall von Runge-Kutta Verfahren angewendet auf Index-2 DAEs in Hessenberg Form gilt (vgl. [HLR89] S. 30):*

$$\eta = \mathbb{1} \otimes y^{[n]}, \qquad \xi = \mathbb{1} \otimes z^{[n]}.$$

Für eine singuläre Matrix A sind zusätzliche Informationen notwendig, um Y, Z und Z' so zu bestimmen, dass die nächsten Approximationen $y^{[n+1]}$ und $z^{[n+1]}$ berechnet werden können (vgl. [HLR89] S. 46 für Lobatto IIIA Methoden). Tatsächlich ist es in diesem Fall unmöglich, Z' eindeutig zu bestimmen. Wir gehen daher im Folgenden von der auch im Fall von Runge-Kutta Verfahren üblichen Voraussetzung (GLM1)

aus. Ist also A regulär, so lässt sich die zweite Gleichung nach Z' auflösen und wir erhalten das reduzierte System

$$\begin{aligned} 0 &= Y - \eta - hAf(Y,Z), \\ 0 &= g(Y). \end{aligned} \tag{3.7}$$

Beispiel 3.3 *Gegeben sei das lineare Index-2 System*

$$\begin{aligned} y' &= By + Cz, \\ 0 &= Dy. \end{aligned}$$

Die Index-2 Bedingung lautet hier: DC invertierbar. Wenden wir darauf das implizite Euler-Cauchy Verfahren ($A = 1$) an, so lautet das Gleichungssystem der Stufenwerte:

$$\begin{pmatrix} \eta \\ 0 \end{pmatrix} = \begin{pmatrix} I - hB & -hC \\ D & 0 \end{pmatrix} \begin{pmatrix} Y \\ Z \end{pmatrix}.$$

Für hinreichend kleine h ist die Matrix invertierbar und wir finden mit Satz 3.4 für die Stufenwerte den Ausdruck

$$\begin{pmatrix} Y \\ Z \end{pmatrix} = \begin{pmatrix} I - C(DC)^{-1}D + \mathcal{O}(h) & C(DC)^{-1} + \mathcal{O}(h) \\ -\frac{1}{h}(DC)^{-1}D + \mathcal{O}(1) & \frac{1}{h}(DC)^{-1} + \mathcal{O}(1) \end{pmatrix} \begin{pmatrix} \eta \\ 0 \end{pmatrix}.$$

Zum einen wird an diesem einfachen Beispiel die Problematik deutlich: Mit $h \to 0$ wird das Gleichungssystem beliebig schlecht konditioniert, da die Inverse Komponenten mit dem Faktor $1/h$ besitzt. Dadurch wird aber das Gleichungssystem (3.7) für theoretische Untersuchungen, die insbesondere den Fall $h \to 0$ umfassen, unbrauchbar. Dieser Grenzprozess wird zum Beispiel bei einer Konvergenzanalyse durchgeführt. Wir werden uns daher nach einem alternativen numerischen Modell zur Berechnung der Stufenwerte umschauen müssen.
Zum anderen besteht Hoffnung, dass alternativ (Y, hZ) durch das Gleichungssystem (3.7) auf stabile Art und Weise zumindest für $0 < h \leq h_0$ berechnet werden kann. In diesem Fall ist nämlich bei dem linearen Problem aus Beispiel 3.3 die Matrix des Gleichungssystems nur noch unproblematisch von h abhängig bzw. für $B = 0$ sogar unabhängig. In der Praxis könnte also das Gleichungssystem trotzdem zur Berechnung der Stufenwerte genutzt werden.
Wir werden diese beiden Aspekte im Folgenden genauer untersuchen, stellen aber zunächst einige Hilfsmittel bereit.

Um die Invertierbarkeit einer Blockmatrix zu garantieren und dann die Form der Inversen bestimmen zu können, ist der Satz über das Schurkomplement ein nützliches Hilfsmittel. Er spielt in der Numerischen Analysis eine wichtige Rolle und kann dennoch durch leichtes Nachrechnen bewiesen werden.

Satz (über das Schurkomplement) 3.4
Seien J und A quadratische Matrizen und C, B und D Matrizen entsprechender Größe mit

$$J = \begin{pmatrix} A & B \\ C & D \end{pmatrix}.$$

Ist die Matrix A invertierbar, so sind die beiden folgenden Aussagen äquivalent:

- *Die Matrix J ist invertierbar.*
- *Das Schurkomplement $S = D - CA^{-1}B$ ist invertierbar.*

In beiden Fällen ist die Inverse gegeben durch

$$J^{-1} = \begin{pmatrix} A^{-1}(I + BS^{-1}CA^{-1}) & -A^{-1}BS^{-1} \\ -S^{-1}CA^{-1} & S^{-1} \end{pmatrix}.$$

□

Ein weiteres nützliches Hilfsmittel ist das Konzept der verallgemeinerten Normen. Es ermöglicht hier die getrennte Behandlung der Y- und Z-Komponente. Zudem existiert für diese Normen eine lokale Version des Banachschen Fixpunktsatzes auf einer Kugel, welche sowohl die lokale Existenz der Lösung als auch die Konvergenz des vereinfachten Newton-Verfahrens garantiert. Es lohnt sich also das Konzept hier einzuführen:

Definition 3.5 *Eine Abbildung $|.| : V \to \mathbb{R}^k$ auf einem Vektorraum V heißt verallgemeinerte Norm, falls gilt*

$$\begin{array}{ll} |v| \geq 0,\ |v| = 0 \Leftrightarrow v = 0 & \text{(vektorwertige pos. Def.)}, \\ |v + w| \leq |v| + |w| & \text{(vektorwertige } \Delta\text{-Ungl.)}, \\ |\alpha v| = |\alpha|_{\mathbb{R}}|v| & \text{(vektorwertige Homogenität)} \end{array}$$

mit der natürlichen Ordnung "$\leq$" auf $\mathbb{R}^k$. Der Absolutbetrag auf $\mathbb{R}$ wird hier mit $|.|_{\mathbb{R}}$ bezeichnet.

Bemerkung: *Jede Norm $\|.\|_*$ auf $\mathbb{R}^k$ definiert durch*

$$\|v\| := \| \, |v| \, \|_*$$

eine Norm auf V. Alle auf diese Weise definierten Normen sind äquivalent. Wenn wir im Folgenden von $(V, |.|)$ als Banachraum sprechen, so meinen wir eigentlich den Raum V versehen mit einer wie oben definierten Norm $\|.\|$.

Die Version des Banachschen Fixpunktsatzes lautet nun (vgl. [BS00]):

Lemma 3.6
Sei $(V, |.|)$ ein Banachraum mit verallgemeinerter Norm $|.|$ und sei B die abgeschlossene Kugel um $\bar{v}$ vom "Radius" $r > 0 \in \mathbb{R}^k$, das heißt es gilt:

$$B := \{v \in V | \, |v - \bar{v}| \leq r\}.$$

Sei weiter $F : B \to V$ eine stetig differenzierbare Funktion mit invertierbarer Jacobi-Matrix $DF(\bar{v})$. Schließlich seien $P, K \in \mathbb{R}^{k,k}$ nicht negative Matrizen mit

$$\begin{aligned} |DF(\bar{v})^{-1} z| &\leq P|z|, \quad z \in V, \\ |(DF(\bar{v}) - DF(v))z| &\leq K|z|, \quad z \in V, \; v \in B, \\ P|F(\bar{v})| &< (I_k - PK)r. \end{aligned} \tag{3.8}$$

Dann besitzt die Gleichung $F(v) = 0$ eine eindeutige Lösung v^* in B. Zudem ist die Matrix $I_k - PK$ regulär und es gilt die Stabilitätsungleichung

$$|v - w| \leq (I_k - PK)^{-1} P|F(v) - F(w)| \text{ für alle } v, w \in B.$$

Zusätzlich ist das vereinfachte Newton-Verfahren

$$v_{n+1} = T(v_n), \qquad T(v) = v - DF(\bar{v})^{-1} F(v)$$

für $v_0 \in B$ durchführbar und die dadurch definierte Folge konvergiert gegen die eindeutige Lösung. Für diese Folge gilt die Abschätzung

$$|v_n - v^*| \leq (I_k - PK)^{-1} (PK)^n |v_1 - v_0|. \tag{3.9}$$

□

Dieses Lemma kann durch die Anwendung von Theorem B1 in [Beyn94] auf den vereinfachten Newton-Operator T bewiesen werden.

Wie wir an der linearen DAE in Beispiel 3.3 gesehen haben, spielt die Inverse der partiellen Ableitung nach (Y, Z) der rechten Seite von (3.7) eine entscheidende Rolle. Die partielle Ableitung lautet im Allgemeinen:

$$\begin{pmatrix} I - hA\frac{\partial f}{\partial Y}(Y, Z) & -hA\frac{\partial f}{\partial Z}(Y, Z) \\ Dg(Y) & 0 \end{pmatrix}. \tag{3.10}$$

Für hinreichend kleine h ist der linke obere Block invertierbar, wobei die Inverse die Form $I + \mathcal{O}(h)$ besitzt (vgl. zum Beispiel [Alt] Satz über die Neumannreihe 3.6). Daher ist nach dem Satz über das Schurkomplement 3.4 die Blockmatrix für solche h genau dann invertierbar, wenn das Schurkomplement

$$S_h = h(DgA\frac{\partial f}{\partial Z})(Y, Z) + \mathcal{O}(h^2)$$

invertierbar ist. Für gewisse Y und Z garantieren die Index-2 Bedingung und (GLM1) die Existenz von S_h^{-1}. Wir handeln uns jedoch in dieser Inversen den Faktor $1/h$ ein. Das Problem liegt zum einen in der Nullmatrix des unteren rechten Blocks und zum anderen im Faktor h im oberen rechten Block der Matrix in (3.10).

Es gibt nun zwei Möglichkeiten dieses Problem zu lösen:

- nach einer äquivalenten Formulierung des Gleichungssystems (3.7) zu suchen, bei der im rechten unteren Block statt der Nullmatrix eine von h unabhängige invertierbare Matrix steht (vgl. "In der Theorie").
- die Variablen Y, hZ zu betrachten, wodurch der Faktor h im oberen rechten Block verschwindet (vgl. "In der Praxis"). Dies macht natürlich nur für $h > 0$ Sinn.

In der Theorie:

Wie wir an dem Gleichungssystem der linearen Index-2 DAE in Beispiel 3.3 gesehen haben, ist das Gleichungssystem in der Form (3.7) für theoretische Untersuchungen, insbesondere für eine Konvergenzanalyse des Verfahrens, nicht brauchbar. Die Idee von Hairer und Wanner besteht darin, im rechten unteren Block der partiellen Ableitung eine von h unabhängige invertierbare Matrix zu erzeugen (vgl. [HW] Kapitel VII 3 und VII 4).

Eine äquivalente Umformung der zweiten Gleichung des Systems (3.7) ist gegeben durch:

$$0 = g(\eta) + \int_0^1 Dg(\eta + s(Y - \eta))ds \cdot (Y - \eta).$$

Das Einsetzen der ersten Gleichung von (3.7) ergibt

$$0 = g(\eta) + \int_0^1 Dg(\eta + s(Y - \eta))ds \cdot hAf(Y, Z).$$

Wir dividieren diese Gleichung nun durch h und erreichen damit, dass die partielle Ableitung der rechten Seite nach Z unabhängig von h wird. Wir untersuchen im Folgenden das Gleichungssystem

$$0 = F(h, \eta, Y, Z), \tag{3.11}$$

wobei wir den Operator F wie folgt definieren:

$$F(h, \eta, Y, Z) = \begin{pmatrix} Y - \eta - hAf(Y, Z) \\ g(\eta)/h + \int_0^1 Dg(\eta + s(Y - \eta))dsAf(Y, Z) \end{pmatrix}. \tag{3.12}$$

Die beiden Systeme (3.7) und (3.11) sind äquivalent, das heißt die Lösungen beider Systeme sind identisch.

Die zweite Gleichung hängt nun von Y und Z ab, so dass der rechte untere Block der partiellen Ableitung nach (Y, Z) nicht länger eine Nullmatrix ist. Die zusätzliche Division durch h wird bewirken, dass die Inverse von $\frac{\partial F}{\partial (Y,Z)}$ und somit die Stabilitätsmatrix unabhängig von h ist bzw. genauer, dass sie auf unproblematische Weise von h abhängen (vgl. Satz 3.7). Somit scheinen alle Probleme gelöst zu sein. Aber die äquivalente Umformung oben hat zur Folge, dass Störungen in der algebraischen Gleichung $g(y) = 0$ nun mit dem Faktor $1/h$ multipliziert werden. Wir werden darauf später noch genauer eingehen (vgl. Korollar 3.9).

Tatsächlich ist aber unter vernünftigen Voraussetzungen die Gleichung (3.11) ein stabiles numerisches Modell, wie wir im folgenden wichtigen Satz beweisen werden. Um den Satz formulieren zu können, wählen wir beliebige Normen auf dem Y- und Z-Raum und definieren die verallgemeinerte Norm

$$|(Y, Z)| = (\|Y\|, \|Z\|) \in \mathbb{R}^2.$$

Satz (über die lokale Lösbarkeit des Gleichungssystems) 3.7
Wir setzen (DAE1)-(DAE3) *und* (GLM1) *voraus. Angenommen* $\eta = \eta(h)$ *und* ξ *genügen*

$$\begin{aligned} \eta - \mathbb{1} \otimes y(x) &= o(1), \\ \xi - \mathbb{1} \otimes z(x) &= o(1), \\ g(\eta)/h &= o(1), \end{aligned} \tag{3.13}$$

wobei die $o(1)$*-Terme hinreichend klein sind und nicht notwendigerweise von* h *abhängen.*
Dann existieren $h_0 > 0$ *und ein "Radius"* $r > 0 \in \mathbb{R}^2$ *unabhängig von* h_0 *und den* $o(1)$*-Termen, so dass das Gleichungssystem*

$$\begin{aligned} 0 &= Y - \eta - hAf(Y, Z), \\ 0 &= g(Y) \end{aligned}$$

für $h \leq h_0$ *eine eindeutige Lösung*

$$(Y(h, \eta, \xi), Z(h, \eta, \xi)) \in B_r := \{V \in \mathbb{R}^{s(N+l)} |\, |V - (\eta, \xi)| \leq r\}$$

besitzt.
Weiterhin existieren nicht negative Matrizen $P, K \in \mathbb{R}^{2,2}$ *für solche* h *mit* $I - PK$ *regulär und*

$$((I - PK)^{-1}P)_{12} = \mathcal{O}(h),$$

so dass die folgende Stabilitätsungleichung in B_r *gilt:*

$$|(Y, Z) - (\bar{Y}, \bar{Z})| \leq (I - PK)^{-1}P|F(h, \eta, Y, Z) - F(h, \eta, \bar{Y}, \bar{Z})|. \tag{3.14}$$

Dabei ist der Operator F wie in (3.12) *definiert.*
Insbesondere gilt:

$$Y(h,\eta,\xi)-\eta=\mathcal{O}(h), \qquad Z(h,\eta,\xi)-\xi=o(1). \tag{3.15}$$

Zusatz: Die beiden $\mathcal{O}$-Terme und die Matrizen P und K sind unabhängig von den Konstanten, die implizit durch die o-Terme in (3.13) gegeben sind, wohingegen der o-Term in (3.15) linear von diesen Konstanten abhängt.

Bemerkung 3.8

(i) *Nehmen wir sogar*

$$\eta-\mathbb{1}\otimes y(x)=\mathcal{O}(h)$$

an, so ist die dritte Gleichung in (3.13) äquivalent zu

$$(I\otimes Q(x))(\eta-\mathbb{1}\otimes y(x))/h=o(1).$$

Dabei ist $Q(x)$ die Projektion aus (1.3)*, welche auf das Bild der partiellen Ableitung $f_z(y(x),z(x))$ parallel zum Tangentialraum $T_{y(x)}\mathcal{M}$ projiziert (vgl. Abbildung 1.1).*
Diese Aussage ist leicht einzusehen: Sei $P(x)=I-Q(x)$ die Projektion auf den Tangentialraum $T_{y(x)}\mathcal{M}=\ker Dg(y(x))$ parallel zum Bild der partiellen Ableitung $f_z(y(x),z(x))$ (vgl. (1.3)*). Dann gilt:*

$$Dg(y(x))P(x)=0.$$

Eine Taylorentwicklung liefert daher:

$$\begin{aligned} g(\eta) &= g(\eta)-g(\mathbb{1}\otimes y(x)) \\ &= Dg(\mathbb{1}\otimes y(x))(\eta-\mathbb{1}\otimes y(x))+\mathcal{O}(h^2) \\ &= (I\otimes Dg(y(x)))(I\otimes Q(x))(\eta-\mathbb{1}\otimes y(x))+\mathcal{O}(h^2). \end{aligned}$$

Die Aussage folgt nun aus der Unabhängigkeit der Ableitung $Dg(y(x))$ von h und der Invertierbarkeit dieser Ableitung auf $\mathrm{im}Q(x)$*, welche durch die Index-2 Bedingung garantiert ist.*

(ii) *Seien $y_c(x,h)$ und $z_c(x,h)$ die correct value Funktionen des allgemeinen linearen Verfahrens, für welche die Darstellungen in* (GLM4) *gelten. Dann erfüllen*

$$\eta=(U\otimes I)y_c(x,h), \qquad \xi=(U\otimes I)z_c(x,h)$$

die Voraussetzungen des Satzes mit $o(1)$ ersetzt durch $\mathcal{O}(h)$. Zudem gelten für die Lösung mit (3.15) *die Gleichungen*

$$\begin{aligned} Y(h,y_c(x,h),z_c(x,h))-\mathbb{1}\otimes y(x)&=\mathcal{O}(h),\\ Z(h,y_c(x,h),z_c(x,h))-\mathbb{1}\otimes z(x)&=\mathcal{O}(h). \end{aligned}$$

Nach Bemerkung (i) und den Darstellungen der correct value Funktionen (GLM4) *haben wir zum Beweis dieser Aussagen lediglich*

$$(I \otimes Q(x))((U \otimes I)y_c(x,h) - \mathbb{1} \otimes y(x))/h = o(1)$$

nachzuweisen. Dies wiederum ist eine leichte Folgerung aus der Darstellung der correct value Funktionen und der Präkonsistenzbedingung $Uu = \mathbb{1}$. *Dabei ist zu beachten, dass der Geschwindigkeitsvektor* $y'(x)$ *tangential zur Mannigfaltigkeit* $T_{y(x)}\mathcal{M}$ *liegt, also*

$$Q(x)y'(x) = 0 \tag{3.16}$$

gilt.

(iii) *Seien* $y^{[n]}$ *und* $z^{[n]}$ *Approximationen mit*

$$\begin{aligned} y^{[n]} - y_c(x_n,h) &= \mathcal{O}(h), \\ z^{[n]} - z_c(x_n,h) &= o(1), \\ (U \otimes Q(x_n))(y^{[n]} - y_c(x_n,h))/h &= o(1), \end{aligned} \tag{3.17}$$

wobei die correct value Funktion wieder der Darstellung in (GLM4) *genüge. Dann erfüllen*

$$\eta = (U \otimes I)y^{[n]}, \qquad \xi = (U \otimes I)z^{[n]}$$

aufgrund von Bemerkung (i) und Gleichung (3.16) *die Voraussetzungen des Satzes.*
Die Gleichungen (3.17) *lassen vermuten, dass stabile allgemeine lineare Verfahren (vgl. zur Stabilität Abschnitt 3.2.2) mit Konvergenzordnung 2 in der y-Komponente und 1 in z existieren, wohingegen der Existenzbeweis von stabilen Verfahren mit Ordnung 1 in beiden Komponenten mehr Aufwand erfordert.*

(iv) *Die Annahmen in (3.13) sind vernünftig, da sie ausdrücken, dass* η *und* ξ, *bzw. genauer deren s Subvektoren, nahe bei der Lösung* $y(x)$ *bzw.* $z(x)$ *liegen. Dies sollte für die Iterationen eines stabilen allgemeinen linearen Verfahrens der Fall sein (vgl. (iii)).*

(v) *Ersetzen wir die* $o(1)$*-Terme in (3.13) durch* $\mathcal{O}(h)$, *so erhalten wir auch* $\mathcal{O}(h)$ *in (3.15). Dabei hängt der Term weiterhin linear von den Konstanten ab, die dann implizit durch die* $\mathcal{O}$*-Terme in (3.13) gegeben sind.*

(vi) *Im Runge-Kutta Fall*

$$\eta = \mathbb{1} \otimes \bar{\eta}, \qquad \xi = \mathbb{1} \otimes \bar{\xi}$$

reduzieren sich die Annahmen in (3.13) zu

$$\begin{aligned} \bar{\eta} - y(x) &= o(1), \\ \bar{\xi} - z(x) &= o(1), \\ g(\bar{\eta})/h &= o(1), \end{aligned}$$

welche nicht ganz so restriktiv sind wie die Annahmen in [HLR89] (vgl. [HLR89] Theorem 4.1).

Beweis des Satzes: Der Beweis des Satzes beruht hauptsächlich in der Anwendung der Version des Banachschen Fixpunktsatzes Lemma 3.6. Daher wird der größte Teil des Beweises daraus bestehen, geeignete Matrizen P und K zu konstruieren, welche die Bedingungen (3.8) von Lemma 3.6 für hinreichend kleine h erfüllen. Es sei darauf hingewiesen, dass F stetig differenzierbar ist, da f als einmal und g als zweimal stetig differenzierbar in (DAE2) vorausgesetzt wurden.

Zunächst stellen wir fest

$$\begin{aligned} F(h,\eta,\eta,\xi) &= \begin{pmatrix} -hAf(\eta,\xi) \\ g(\eta)/h + Dg(\eta)Af(\eta,\xi) \end{pmatrix} \\ &= \begin{pmatrix} \mathcal{O}(h) \\ o(1) \end{pmatrix}, \end{aligned}$$

wobei der $\mathcal{O}$-Term unabhängig ist von den Konstanten, die implizit durch die o-Terme in (3.13) gegeben sind, wohingegen der o-Term dies nicht ist. Doch gehen die Konstanten nur in linearer Art und Weise ein. Wir formulieren ausführlich mit dem Kronecker Produkt, um diese Rechnung zu verdeutlichen:

$$\begin{aligned} Dg(\eta)(A\otimes I)f(\eta,\xi) &= Dg(\mathbb{1}\otimes y(x))(A\otimes I)f(\mathbb{1}\otimes y(x),\mathbb{1}\otimes z(x)) + o(1) \\ &= (I\otimes Dg(y(x)))(A\otimes I)(\mathbb{1}\otimes f(y(x),z(x))) + o(1) \\ &= A\mathbb{1}\otimes \underbrace{Dg(y(x))f(y(x),z(x))}_{=0} + o(1). \end{aligned} \tag{3.18}$$

Aus der Darstellung von $F(h,\eta,\eta,\xi)$ folgt unmittelbar, dass dieser Term für $o(1), h \to 0$ ebenfalls gegen Null konvergiert.

Die partielle Ableitung von F nach (Y,Z) ist von der Form

$$\frac{\partial F}{\partial(Y,Z)}(h,\eta,Y,Z) = \begin{pmatrix} I - hA\frac{\partial f}{\partial Y}(Y,Z) & -hA\frac{\partial f}{\partial Z}(Y,Z) \\ G(Y,Z) & H(Y,Z) \end{pmatrix},$$

wobei G und H stetige Funktionen sind. Für H gilt genauer:

$$H(Y,Z) = \int_0^1 Dg(\eta + s(Y-\eta))ds \cdot A\frac{\partial f}{\partial Z}(Y,Z).$$

Für $Y = \eta$ und $Z = \xi$ erhalten wir

$$\begin{aligned} J_h &:= \frac{\partial F}{\partial(Y,Z)}(h,\eta,\eta,\xi) \\ &= \begin{pmatrix} I + \mathcal{O}(h) & \mathcal{O}(h) \\ \mathcal{O}(1) & A \otimes (Dgf_z)(y(x),z(x)) + o(1) \end{pmatrix}, \end{aligned}$$

wobei die Herleitung des rechten unteren Blocks der Rechnung in (3.18) gleicht und nur dessen o-Term von den Konstanten abhängt, welche implizit durch die o-Terme in (3.13) gegeben sind. In diesem Block bringt das Kronecker Produkt mehr Einsicht in die Struktur.
Für hinreichend kleine h ist der linke obere Block invertierbar, wobei die Inverse ebenfalls die Form $I + \mathcal{O}(h)$ besitzt (vgl. [Alt] Satz 3.6). Wiederum mit dem Satz über die Neumannreihe lässt sich mit der Index-2 Bedingung (DAE1) und der Regularität von A für kleine $o(1)$ die Invertierbarkeit des rechten unteren Blocks von J_h garantieren. Die Inverse ist gegeben durch

$$A^{-1} \otimes (Dgf_z)^{-1}(y(x),z(x)) + o(1).$$

Daher existiert ein $h_0 > 0$, so dass mit dem Satz über das Schurkomplement 3.4 die Invertierbarkeit von J_h für hinreichend kleine $o(1)$ und $h \leq h_0$ gefolgert werden kann. Die Inverse besitzt dabei genau dieselbe Struktur wie J_h selber:

$$J_h^{-1} = \begin{pmatrix} I + \mathcal{O}(h) & \mathcal{O}(h) \\ \mathcal{O}(1) & A^{-1} \otimes (Dgf_z)^{-1}(y(x),z(x)) + o(1) \end{pmatrix}.$$

Für solche h und $o(1)$-Terme finden wir eine positive Konstante M, welche unabhängig von den o-Termen in (3.13) ist mit

$$|J_h^{-1}V| \leq M \begin{pmatrix} 1 & h \\ 1 & 1 \end{pmatrix} |V|$$

für alle $V \in \mathbb{R}^{s(N+l)}$. Diese Abschätzung entspricht bei geeigneter Definition von P der ersten Ungleichung in (3.8) von Lemma 3.6.

Des Weiteren erhalten wir

$$\begin{aligned} J_h - \frac{\partial F}{\partial(Y,Z)}(h,\eta,Y,Z) \;=\; & \\ & \begin{pmatrix} \mathcal{O}(h)(\frac{\partial f}{\partial Y}(\eta,\xi) - \frac{\partial f}{\partial Y}(Y,Z)) & \mathcal{O}(h)(\frac{\partial f}{\partial Z}(\eta,\xi) - \frac{\partial f}{\partial Z}(Y,Z)) \\ G(\eta,\xi) - G(Y,Z) & H(\eta,\xi) - H(Y,Z) \end{pmatrix}. \end{aligned}$$

Es existiert aufgrund der Stetigkeit der auftretenden Funktionen ein Radius $r = r(\epsilon) > 0 \in \mathbb{R}^2$ mit

$$|(J_h - \frac{\partial F}{\partial(Y,Z)}(h,\eta,Y,Z))V| \leq M\epsilon \begin{pmatrix} h & h \\ 1 & 1 \end{pmatrix} |V|$$

für alle $(Y, Z) \in B_r$ und $V \in \mathbb{R}^{s(N+l)}$. Diese Abschätzung wiederum entspricht bei geeigneter Definition von K der zweiten Ungleichung in (3.8). Ohne Einschränkung ist M dieselbe Konstante wie oben.
Tatsächlich hängt der Mittelpunkt von B_r, der Punkt (η, ξ), von h ab. Dieser Punkt bleibt jedoch für $h \leq h_0$ in einer (kompakten) Umgebung von $\mathbb{1} \otimes (y(x), z(x))$, so dass r unabhängig von h gewählt werden kann.

Wir definieren also

$$P := M \begin{pmatrix} 1 & h \\ 1 & 1 \end{pmatrix}, \qquad K := M\epsilon \begin{pmatrix} h & h \\ 1 & 1 \end{pmatrix}.$$

Für hinreichend kleine ϵ und h ist $I - PK$ eine M-Matrix. Da wir mit $o(1)$ und h auch $F(h, \eta, \eta, \xi)$ beliebig klein machen können, gehen wir ohne Einschränkung auch von der Gültigkeit der dritten Ungleichung in (3.8) aus.

Durch die Anwendung von Lemma 3.6 erhalten wir beinahe alle Aussagen des Satzes. Es bleibt noch zu beweisen, dass die rechte obere Komponente der Matrix $(I - PK)^{-1}P$ von der Größe $\mathcal{O}(h)$ ist. In diesem Fall wären nämlich die Darstellungen in (3.15) eine leichte Folge der Stabilitätsungleichung und der Darstellung von $F(h, \eta, \eta, \xi)$ am Anfang des Beweises.
Es gilt

$$I - PK = \begin{pmatrix} 1 & 0 \\ \mathcal{O}(\epsilon) & 1 + \mathcal{O}(\epsilon) \end{pmatrix} + \mathcal{O}(h).$$

In genau der gleichen Art und Weise wie oben bei der Matrix J_h erhalten wir, dass die Inverse der Matrix $I - PK$ dieselbe Struktur besitzt wie die Matrix selbst. Die Existenz der Inversen hatten wir für hinreichend kleine ϵ und h bereits weiter oben begründet. Eine leichte Rechnung zeigt nun, dass

$$((I - PK)^{-1}P)_{12} = \mathcal{O}(h)$$

unabhängig von den o-Termen in (3.13) gilt, womit der Satz vollständig bewiesen ist.

□

Der Operator F ist unabhängig von ξ. Die Kugel B_r natürlich nicht, ihr Mittelpunkt ist der Punkt (η, ξ). Dennoch ist der Radius sowohl von h als auch den $o(1)$-Termen in (3.13) unabhängig. Wir erhalten daher mit (3.15) für $\bar{\xi}$ mit $\xi - \bar{\xi} = o(1)$ die lokale Unabhängigkeit der Lösung von ξ, das heißt es gilt:

$$Y(h, \eta, \bar{\xi}) = Y(h, \eta, \xi), \qquad Z(h, \eta, \bar{\xi}) = Z(h, \eta, \xi),$$

falls h und die $o(1)$ hinreichend klein sind (vgl. Abbildung 3.3).

Aufgrund der Index-2 Bedingung ist $z(x)$ über die versteckte Nebenbedingung eindeutig durch $y(x)$ definiert. Wir werden daher im Folgenden die Lösung für η, ξ wie

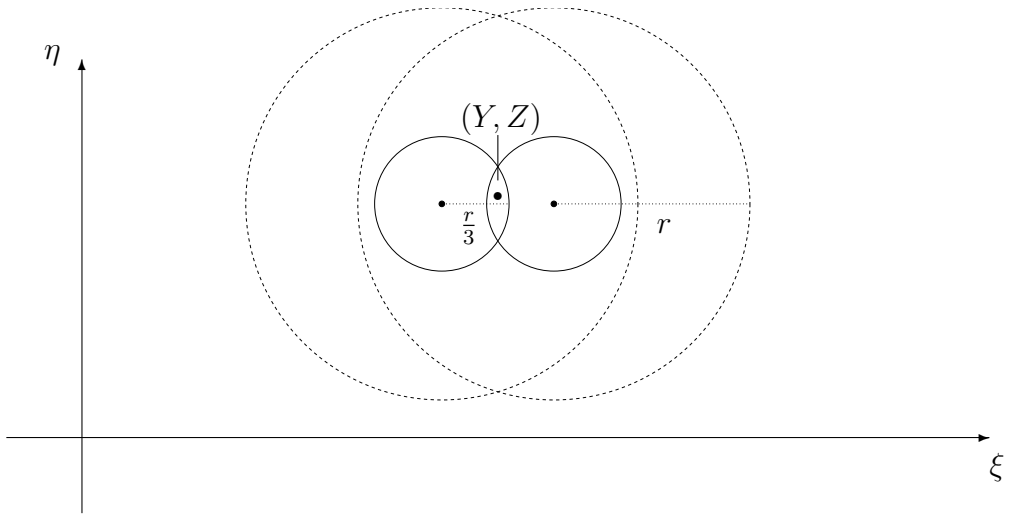

Abbildung 3.3: Lokale Unabhängigkeit der Lösung $(Y, Z) := (Y(h, \eta, \xi), Z(h, \eta, \xi))$ von der Komponente ξ.

in Satz 3.7 nur mit $Y(h, \eta),\ Z(h, \eta)$ bezeichnen. Für Iterationen mit einer Darstellung wie in Bemerkung 3.8 (iii) schreiben wir $Y(h, y^{[n]}),\ Z(h, y^{[n]})$.
Die Iteration für die y-Komponente ist dann gegeben durch

$$y^{[n+1]} = Vy^{[n]} + hBf(Y(h, y^{[n]}), Z(h, y^{[n]})).$$

Diese Darstellung bzw. genauer diese lokale Unabhängigkeit der Lösung lassen vermuten, dass sich die Stabilitäts-, Konvergenz- und Konsistenzanalyse der y Variable weitgehend unabhängig von der z Variablen durchführen lässt. Wir werden in den nächsten Abschnitten sehen, dass dies tatsächlich möglich ist.

Wie wir gesehen haben, garantiert uns die Division der zweiten Gleichung des Systems der Stufenwerte durch h und deren äquivalente Umformung die Lösbarkeit des Gleichungssystems (3.7) durch das stabile Modell

$$F(h, \eta, Y, Z) = 0.$$

Tatsächlich gehen aber Störungen der algebraischen Gleichung $g(y) = 0$ mit dem Faktor $1/h$ in diese Rechnung mit ein. Wir betrachten dazu die gestörte DAE

$$\begin{aligned} y' &= f(y, z) + \delta(x), \\ \theta(x) &= g(y). \end{aligned}$$

Wir finden dann bei entsprechender Definition von δ_0 und θ_0 für das Gleichungssystem der Stufenwerte

$$\begin{aligned} 0 &= Y - \eta - hAf(Y, Z) - h\delta_0, \\ \theta_0 &= g(Y). \end{aligned} \tag{3.19}$$

Eine Anwendung von Satz 3.7, insbesondere der Stabilitätsungleichung (3.14), liefert:

Korollar 3.9
Angenommen η und ξ erfüllen die Voraussetzungen von Satz 3.7. Sei Y, Z die entsprechende Lösung von (3.7), deren Existenz durch diesen Satz für hinreichend kleine h und $o(1)$-Terme garantiert werden kann. Dabei gehen wir ebenfalls von der Gültigkeit von (DAE1)-(DAE3) *und* (GLM1) *aus. Genüge $\tilde{Y}, \tilde{Z}$ dem System (3.19) und gelte*

$$Y - \tilde{Y} = o(1), \qquad Z - \tilde{Z} = o(1).$$

Dann existiert für hinreichend kleine Terme $o(1)$ eine positive Konstante K mit

$$\begin{aligned} \|Y - \tilde{Y}\| &\leq K(h\|\delta_0\| + \|\theta_0\|), \\ \|Z - \tilde{Z}\| &\leq \frac{K}{h}(h\|\delta_0\| + \|\theta_0\|). \end{aligned}$$

□

Bemerkung: *Die Aussage des Korollars ist in dem Buch [HW] in Theorem 4.2 Kapitel VII.4 enthalten. Dort wird für die Störungen etwas merkwürdig vorausgesetzt, dass sie sich wie $\mathcal{O}(h)$ bzw. die Störung in der algebraischen Gleichung wie $\mathcal{O}(h^2)$ verhalten.*

Durch das Korollar wird deutlich, dass zum Beispiel Rundungsfehler, die nicht mit $h \to 0$ verschwinden, eine zufrieden stellende Berechnung der Stufenwerte durch den Operator F für zu kleine h unmöglich machen.

In der Praxis:

Die Stufenwerte lassen sich also durch das stabile numerische Modell (3.11) berechnen. Trotzdem ist das Modell eher für die theoretische Analyse geeignet. Praktisch will man die Auswertung der Ableitung von g und die Berechnung des Integrals vermeiden. Zudem gehen Fehler in der Auswertung von $g(\eta)$ durch die Division mit h verstärkt in die Rechnung ein (vgl. Korollar 3.9). Somit ist die praktische Brauchbarkeit von F zur Berechnung der Stufenwerte relativiert. Tatsächlich wird der Operator F in der Implementierung eines Codes nicht verwendet, um die Stufenwerte zu berechnen.

Wir hatten bei dem linearen Problem von Beispiel 3.3 vermutet, dass statt der Stufenwerte Y, Z vielleicht die Variablen Y, hZ auf stabile Art und Weise berechnet werden können. Dies ist äquivalent dazu mit einem anderen Maß zu messen: Wir definieren die von h abhängige verallgemeinerte Norm

$$|(Y, Z)|_h := (\|Y\|, h\|Z\|) \in \mathbb{R}^2.$$

Zudem definieren wir den Operator $H(h, \eta, Y, Z)$ durch die rechte Seite von (3.7) und untersuchen die Gleichung

$$H(h, \eta, Y, Z) = 0. \tag{3.20}$$

Die Frage, die sich jetzt stellt, ist, ob Lemma 3.6 für diese verallgemeinerte Norm und für diesen Operator H anwendbar ist.
Das Problem wäre aber mit dieser Anwendbarkeit nicht unbedingt gelöst. Die Berechnung von Z aus der Variable $\tilde{Z} = hZ$ ist für $h \to 0$ beliebig schlecht konditioniert. Schon wieder werden Störungen, zum Beispiel Rundungsfehler, mit dem Faktor $1/h$ multipliziert.

Bemerkung: *Für Gleichungen, deren Funktion f linear von z abhängt,*

$$f(y,z) = f_0(y) + f_z(y)z, \tag{3.21}$$

ist die schlecht konditionierte Berechnung von Z für die y-Komponente nicht nötig. Die Variable Z tritt hier nur in der Form hZ auf:

$$y^{[n+1]} = Vy^{[n]} + hAf_0(Y) + Af_z(Y)hZ.$$

(Dabei ist die Schreibweise $f_z(Y)hZ$ streng genommen nicht ganz korrekt, aber in Bezug auf den Gedanken hier ausreichend.) Diese lineare Abhängigkeit weisen viele Index-2 Probleme der Anwendung auf, zum Beispiel die GGL-Formulierung mechanischer Mehrkörpersysteme mit Nebenbedingungen oder auch die Index-2 Formulierung der Probleme $B(y)y' = f(y)$ (vgl. Kapitel 1).

Wir wollen nun mit Hilfe von Lemma 3.6 nachweisen, dass das vereinfachte Newton-Verfahren

$$\begin{pmatrix} Y_{j+1} \\ Z_{j+1} \end{pmatrix} = \begin{pmatrix} Y_j \\ Z_j \end{pmatrix} - \left(\frac{\partial H}{\partial (Y,Z)}(h,\eta,\eta,\xi) \right)^{-1} H(h,\eta,Y_j,Z_j)$$

gegen eine lokal eindeutige Lösung der Gleichung (3.20) konvergiert. Wir gehen dabei wie im Beweis von Satz 3.7 vor:
Gelte also wieder (DAE1)-(DAE3) und (GLM1). Zunächst gilt mit $g(\eta) = o(1)$ offenbar

$$|H(h,\eta,\eta,\xi)|_h = \begin{pmatrix} \mathcal{O}(h) \\ ho(1) \end{pmatrix}.$$

Die partielle Ableitung von H nach (Y,Z) hatten wir bereits oben berechnet:

$$\frac{\partial H}{\partial (Y,Z)}(h,\eta,Y,Z) = \begin{pmatrix} I - hA\frac{\partial f}{\partial Y}(Y,Z) & -hA\frac{\partial f}{\partial Z}(Y,Z) \\ Dg(Y) & 0 \end{pmatrix}.$$

Mit den Voraussetzungen

$$\eta - \mathbb{1} \otimes y(x) = o(1), \qquad \xi - \mathbb{1} \otimes z(x) = o(1) \tag{3.22}$$

erhalten wir

$$\begin{aligned} J_h &:= \frac{\partial H}{\partial (Y,Z)}(h,\eta,\eta,\xi) \\ &= \begin{pmatrix} I + \mathcal{O}(h) & -h(A \otimes f_z(y(x),z(x))) + ho(1) \\ (I \otimes Dg(y(x))) + o(1) & 0 \end{pmatrix}. \end{aligned}$$

Auch hier verwenden wir das Kronecker Produkt, um die Struktur der jeweiligen Blöcke herauszustellen. Es sei noch bemerkt, dass wir im Fall der linearen Abhängigkeit der Funktion f von der algebraischen Variable z, wie in (3.21), die Forderung an ξ in (3.22) nicht benötigen.
Mit dem Satz über die Neumannreihe (vgl. [Alt] Satz 3.6) und dem Satz über das Schurkomplement finden wir für hinreichend kleine h und $o(1)$ die Invertierbarkeit von J_h, wobei die Inverse die folgende Gestalt besitzt:

$$J_h^{-1} = \begin{pmatrix} I \otimes P(x) + o(1) & I \otimes f_z(Dgf_z)^{-1} + o(1) \\ -\frac{1}{h}(A^{-1} \otimes (Dgf_z)^{-1}Dg + o(1)) & \frac{1}{h}(A^{-1} \otimes (Dgf_z)^{-1} + o(1)) \end{pmatrix}.$$

Dabei sind alle partiellen Ableitungen in $y(x)$ bzw. $(y(x), z(x))$ ausgewertet. Es existiert somit eine positive Konstante M, so dass für alle $V \in \mathbb{R}^{s(N+l)}$ gilt:

$$|J_h^{-1}V|_h \leq M \begin{pmatrix} 1 & 1/h \\ 1 & 1/h \end{pmatrix} |V|_h.$$

Des Weiteren erhalten wir

$$J_h - \frac{\partial H}{\partial(Y,Z)}(h,\eta,Y,Z) = \begin{pmatrix} \mathcal{O}(h) & \mathcal{O}(h)(\frac{\partial f}{\partial Z}(\eta,\xi) - \frac{\partial f}{\partial Z}(Y,Z)) \\ Dg(\eta) - Dg(Y) & 0 \end{pmatrix}.$$

Aufgrund der Stetigkeit der partiellen Ableitungen erhalten wir für vorgegebenes $\epsilon > 0$ einen Radius $r = r(\epsilon) > 0 \in \mathbb{R}$ mit

$$|(J_h - \frac{\partial H}{\partial(Y,Z)}(h,\eta,Y,Z))V|_h \leq M \begin{pmatrix} h & \epsilon \\ \epsilon h & 0 \end{pmatrix} |V|_h$$

für alle $V \in \mathbb{R}^{s(N+l)}$ und $(Y,Z) \in \mathbb{R}^{s(N+l)}$ mit

$$\|Y - \eta\| < r, \qquad \|Z - \xi\| < r.$$

Ohne Einschränkung ist M dieselbe Konstante wie in der Abschätzung oben.
Wir definieren

$$P := M \begin{pmatrix} 1 & 1/h \\ 1 & 1/h \end{pmatrix}, \qquad K := M \begin{pmatrix} h & \epsilon \\ h\epsilon & 0 \end{pmatrix}.$$

Wir haben darauf verzichtet, auch in der linken oberen Komponente der Matrix K den Faktor ϵ herzuleiten. Der Grund dafür ist Folgender: Für Funktionen f der Form (3.21), welche linear von z abhängen, kann somit weiterhin auf die Bedingung an ξ in (3.22) verzichtet werden.
Es gilt:

$$PK = M^2 \begin{pmatrix} h+\epsilon & \epsilon \\ h+\epsilon & \epsilon \end{pmatrix}.$$

Wie im Beweis von Satz 3.7 ist für hinreichend kleine h, ϵ und $o(1)$ das Lemma 3.6 anwendbar. Wir erhalten also die lokale Lösbarkeit und die Konvergenz des vereinfachten Newton-Verfahrens. Die Kugel ist dabei gegeben durch

$$B_r = \{V \in \mathbb{R}^{s(N+l)} |\, |V - (\eta,\xi)| \leq (r,r)^T\},$$

wobei die verallgemeinerte Norm, hier unabhängig von h, wie weiter oben definiert ist:

$$|(Y, Z)| = (\|Y\|, \|Z\|) \in \mathbb{R}^2.$$

Ist die Funktion f linear in z, so kann in der Definition von B_r die zweite Komponente des Radius $(r, r)^T$ durch ein beliebig großes $R > 0$ ersetzt werden, wie wir oben bereits begründet haben: Die Bedingung an ξ ist überflüssig.
Wählen wir $\epsilon = h$, so gewinnen wir aus der Darstellung von PK mit Gleichung (3.9) bei jeder Iteration des vereinfachten Newton-Verfahrens eine h Potenz. Somit empfiehlt es sich, das Iterationsverfahren mindestens so oft auszuüben, wie die Ordnung des allgemeinen linearen Verfahrens angibt. Aufgrund der Z-Komponente ist gegebenenfalls eine zusätzliche Iteration nötig. Hinweise zur Implementierung des vereinfachten Newton-Verfahrens sind in Kapitel 4 gegeben.

Zusammenfassung: Zum einen haben wir in diesem Abschnitt festgestellt, dass sich die Stufenwerte durch ein stabiles numerisches Verfahren berechnen lassen. Eine solche Berechnung ist insbesondere für die theoretische Konvergenzanalyse unabdingbar. Andererseits gehen Störungen der algebraischen Gleichung multipliziert mit dem Faktor $1/h$ in diese Rechnung ein. Das stabile numerische Verfahren besitzt daher nur relativen praktischen Nutzen. Zum anderen haben wir nachgewiesen, dass sich Y und hZ durch das vereinfachte Newton-Verfahren berechnen lassen, welches durch das ursprüngliche Gleichungssystem der Stufenwerte definiert ist. Diese Berechnung lässt sich in der Praxis gut realisieren. Die anschließende Bestimmung von Z ist dagegen für kleine Schrittweiten h schlecht konditioniert. Wieder werden Fehlerterme mit $1/h$ multipliziert. Bei Problemen, deren Funktion f linear von der algebraischen Variablen z abhängt, ist diese schlecht konditionierte Berechnung in der y-Komponente nicht nötig. Solche Probleme lassen sich in diesem Sinne numerisch einfacher lösen. Allgemein wird man beim Lösen von Index-2 DAEs in Hessenberg Form auf die Vermeidung zu kleiner Schrittweiten achten müssen.

3.2.2 Stabilität

In der Numerik gewöhnlicher Differentialgleichungen werden viele verschiedene Stabilitätsbegriffe verwendet. Im Zusammenhang mit der Konvergenz numerischer Verfahren ist die Stabilität genau die Eigenschaft, die mit der Konsistenz die Konvergenz des Verfahrens garantiert. Allgemein lässt sich die Stabilität eines numerischen Verfahrens wie folgt beschreiben: "Kleine" Störungen des Verfahrens haben keinen "großen" Einfluss auf die Güte der numerischen Lösung. Butcher schreibt in seinem Buch über iterative Verfahren: „Stability has the effect of guaranteeing that errors introduced in any step of a computation do not have disastrous effects on later steps" (vgl. [B] S. 372). Tatsächlich wird die Stabilität eines Verfahrens oft nicht explizit erwähnt. Hairer et al. zum Beispiel analysieren direkt den Einfluss von Störungen, um Konvergenz zu zeigen (vgl. [HNW] S. 438, [HW] S. 218, 484, 493). Natürlich wird die Stabilität implizit ausgenutzt, jedoch wird auf den Begriff der Stabilität verzichtet. Im Zusammenhang mit Index-2 DAEs in Hessenberg Form zeigt die genauere Untersuchung der Stabilität, wo die Singularität der DAE eingeht.

In diesem Abschnitt untersuchen wir die Stabilität von allgemeinen linearen Verfahren angewendet auf Index-2 DAEs in Hessenberg Form, indem wir die numerische Lösung als Nullstelle eines geeigneten Operators T_h interpretieren und dessen Stabilität beweisen:

$$\|u - \bar{u}\| \leq C\|T_h u - T_h \bar{u}\|.$$

Dabei bilden die Lemmata 3.12 und 3.13 den Kern des Abschnitts, während Satz 3.18 die Ergebnisse über die Stabilität zusammenfasst.

Bemerkung: *Der Zusammenhang zwischen Störungen und der Stabilität des Operators T_h ist Folgender: Bezeichne u^* die Lösung des numerischen Verfahrens, das heißt es gilt $T_h u^* = 0$, und sei u die Lösung der gestörten Gleichung*

$$T_h u = \varepsilon(u), \qquad \|\varepsilon(u)\| \leq \varepsilon.$$

Mit der Stabilitätsungleichung oben erhalten wir

$$\|u - u^*\| \leq C\varepsilon.$$

Somit liegt für kleine Werte von ε die Lösung u der gestörten Gleichung in der "Nähe" von u^*.

Für allgemeine lineare Verfahren ist es tatsächlich möglich, die y-Komponente weitgehend unabhängig von der z Variablen zu behandeln. Dies wird auf eine zweidimensionale Stabilitätsungleichung mit $C \in \mathbb{R}^{2,2}$ führen, wie wir sie auch im letzten Abschnitt hergeleitet haben (vgl. (3.14)).

Einführend betrachten wir Einschrittverfahren und allgemeine lineare Verfahren für gewöhnliche autonome Differentialgleichungen. Wir verwenden dabei Notationen in

einer nicht streng formalen Art und Weise.

Stabilität allgemeiner linearer Verfahren für ODEs

Es ist aus der Numerik gewöhnlicher Differentialgleichungen bekannt, dass die Stabilität eines Runge-Kutta Verfahrens angewandt auf die autonome Gleichung

$$y' = f(y), \qquad y(0) = y_0 \in \mathbb{R}^N$$

implizit durch die Glattheit der rechten Seite gegeben ist: Angenommen f genüge einer Lipschitz- Bedingung

$$\|f(u) - f(\bar{u})\| \leq L_f \|u - \bar{u}\|.$$

Dann ist die Verfahrensfunktion

$$\Phi(h, u) := (b^T \otimes I) f(Y(h, u))$$

für hinreichend kleine h wohldefiniert (vgl. [HNW] Theorem 7.2 S.206). Zudem ist sie Lipschitz-stetig im zweiten Argument und zwar für hinreichend kleines h_0 gleichmäßig für $h \leq h_0$.
Allgemein erhalten wir die Stabilität für Einschrittverfahren

$$\begin{aligned} y^{[0]} &= y_0, \\ y^{[n+1]} &= y^{[n]} + h\Phi(h, y^{[n]}), \end{aligned}$$

deren Verfahrensfunktionen in dieser Weise Lipschitz-stetig sind: Wir definieren dazu abkürzend den Iterationsoperator

$$Nu = u + h\Phi(h, u).$$

Zudem sei ein äquidistantes Gitter Ω_h der Gitterbreite h auf dem Integrationsintervall gegeben. Die numerische Lösung ist nun als Nullstelle des Operators

$$T_h : (\mathbb{R}^N)^{\Omega_h} \to (\mathbb{R}^N)^{\Omega_h}$$

gegeben, welcher definiert ist durch

$$T_h u(x_j) = \begin{cases} u(0) - y_0 \text{ für } j = 0, \\ h^{-1}(u(x_j) - Nu(x_{j-1})) \text{ für } j > 0. \end{cases}$$

Dabei bezeichnet $(\mathbb{R}^N)^{\Omega_h}$ den Raum der Funktionen, welche auf dem Gitter Ω_h definiert sind und nach $\mathbb{R}^N$ abbilden.
Wir leiten nun eine Stabilitätsungleichung für das numerische Modell

$$T_h y = 0$$

her: Es seien $\hat{y}, \bar{y}$ zwei Elemente aus $(\mathbb{R}^N)^{\Omega_h}$. Mit den abkürzenden aber nahe liegenden Schreibweisen

$$\begin{aligned} \Delta y_j &:= \hat{y}(x_j) - \bar{y}(x_j), \\ \Delta\Phi_j &:= \Phi(h, \hat{y}(x_j)) - \Phi(h, \bar{y}(x_j)), \\ \Delta T_j &:= T_h\hat{y}(x_j) - T_h\bar{y}(x_j), \\ \Delta T &:= T_h\hat{y} - T_h\bar{y} \end{aligned}$$

erhalten wir aufgrund der Definition von T_h die Gleichung

$$\Delta y_{j+1} = \Delta y_j + h\Delta\Phi_j + h\Delta T_{j+1}.$$

Mit der Lipschitz-Stetigkeit der Verfahrensfunktion finden wir die bekannten Ungleichungen

$$\begin{aligned} \|\Delta y_{j+1}\| &\leq (1 + hL_\Phi)\|\Delta y_j\| + h\|\Delta T\|_\infty \\ &\leq (1 + hL_\Phi)^{j+1} \underbrace{\|\Delta y_0\|}_{\leq \|\Delta T\|_\infty} + h\sum_{i=0}^{j}(1 + hL_\Phi)^i\|\Delta T\|_\infty. \end{aligned}$$

Da der Faktor $(1+hL_\Phi)^j$ für $jh \leq konst.$ durch eine positive Konstante abgeschätzt werden kann, die unabhängig von j ist, erhalten wir die Stabilität:

$$\|\hat{y} - \bar{y}\|_\infty \leq C\|T_h\hat{y} - T_h\bar{y}\|_\infty.$$

Es ist erwähnenswert, dass wir den Faktor h vor der Summe oben investieren müssen, um die Beschränktheit des zweiten Terms und somit die Stabilität zu erhalten.

Ein allgemeines lineares Verfahren für gewöhnliche Differentialgleichungen kann als Einschrittverfahren in einem höher dimensionalen Raum interpretiert werden (zur Definition von allgemeinen linearen Verfahren für gewöhnliche Differentialgleichungen vgl. Abschnitt 2.1):

$$\begin{aligned} y^{[0]} &= \varphi(h, y_0) \in \mathbb{R}^{rN}, \\ y^{[n+1]} &= (V \otimes I)y^{[n]} + h\Phi(h, y^{[n]}) \in \mathbb{R}^{rN}. \end{aligned}$$

Dabei ist φ eine Startprozedur und die Verfahrensfunktion Φ ist gegeben durch

$$\Phi(h, u) = (B \otimes I)f(Y(h, u)).$$

Die Stufenwerte $Y(h, u)$ erfüllen das Gleichungsystem

$$Y = (U \otimes I)u + h(A \otimes I)f(Y).$$

In genau derselben Art und Weise wie bei Runge-Kutta Verfahren erhalten wir für Lipschitz-stetiges f die Wohldefiniertheit der Verfahrensfunktion für hinreichend

kleine h und die in h gleichmäßige Lipschitz-Stetigkeit im zweiten Argument (vgl. [HNW] Theorem 7.2 S.206 für Runge-Kutta Verfahren). Der einzig neue Aspekt ist das Auftreten der Matrix V in der Iteration für $y^{[n+1]}$. Tatsächlich besitzt diese Matrix Einfluss auf die Stabilität, wie wir unmittelbar sehen werden.
Wir definieren wieder den Iterationsoperator

$$Nu = (V \otimes I)u + h\Phi(h, u)$$

und interpretieren die numerische Lösung als Nullstelle des Operators

$$T_h : (\mathbb{R}^{rN})^{\Omega_h} \to (\mathbb{R}^{rN})^{\Omega_h}$$

definiert durch

$$T_h u(x_j) = \begin{cases} u(0) - \varphi(h, y_0) \text{ für } j = 0, \\ h^{-1}(u(x_j) - Nu(x_{j-1})) \text{ für } j > 0. \end{cases}$$

Analog zu den Einschrittverfahren oben gilt für $\hat{y}, \bar{y} \in (\mathbb{R}^{rN})^{\Omega_h}$ mit denselben Schreibweisen

$$\Delta y_{j+1} = (V \otimes I)\Delta y_j + h\Delta\Phi_j + h\Delta T_{j+1}$$

und somit auch

$$\|\Delta y_{j+1}\| \leq (\|V\| + hL_\Phi)^{j+1}\|\Delta T\|_\infty + h\sum_{i=0}^{j}(\|V\| + hL_\Phi)^i\|\Delta T\|_\infty.$$

Ist die Matrix V stabil, so existiert eine Norm $\|.\|$ mit

$$\|V\| \leq \rho(V) \leq 1$$

(vgl. [SB] Satz 6.9.2). In diesem Fall folgt die Stabilität wieder aus der Beschränktheit des Faktors $(\|V\| + hL_\Phi)^j$ für $jh \leq konst.$ In Kapitel 2 bezeichneten wir ein allgemeines lineares Verfahren als nullstabil, wenn die Verfahrensmatrix V stabil ist (vgl. Definition 2.9).

Zusammenfassend lässt sich sagen, dass die Stabilität von allgemeinen linearen Verfahren angewendet auf gewöhnliche Differentialgleichungen durch die Nullstabilität und die Glattheit der rechten Seite f gegeben ist.

Stabilität allgemeiner linearer Verfahren für Index-2 DAEs

Wir untersuchen nun, welche zusätzlichen Forderungen wir an ein allgemeines lineares Verfahren stellen müssen, um die Stabilität solcher Verfahren angewandt auf Index-2 DAEs in Hessenberg Form gewährleisten zu können. Die Glattheit der rechten Seite bei gewöhnlichen Differentialgleichungen lieferte eine Lipschitz-Stetigkeit der Verfahrensfunktion, welche die Stabilität in diesem Fall für nullstabile Verfahren

garantierte. Wie sieht dies bei Index-2 DAEs aus?

Wir betrachten zunächst die y-Komponente eines allgemeinen linearen Verfahrens angewandt auf eine Index-2 DAE in Hessenbergform (vgl. (3.6)):

$$\begin{aligned} y^{[0]} &= \varphi(h, y_0), \\ y^{[n+1]} &= (V \otimes I) y^{[n]} + h\Phi(h, y^{[n]}). \end{aligned}$$

Tatsächlich gleicht diese Darstellung einem allgemeinen linearen Verfahren für gewöhnliche Differentialgleichungen. Es ist jedoch zu beachten, dass die Verfahrensfunktion Φ in diesem Fall auch von den Stufenwerten Z abhängt:

$$\Phi(h, u) = (B \otimes I) f(Y(h, u), Z(h, u)).$$

Um nun diese Verfahrensfunktion auf Lipschitz-Stetigkeit hin zu untersuchen, betrachten wir die Argumente dieser Funktion, die Stufenwerte, noch einmal genauer. Wir greifen dazu auf Ergebnisse des vorherigen Abschnitts zurück. Daher gehen wir auch hier von der Glattheitsvoraussetzung (DAE3) aus. Eine Anwendung von Satz 3.7 liefert:

Korollar 3.10
Es seien Vektoren $y, \bar{y} \in \mathbb{R}^{rN}$ und $z, \bar{z} \in \mathbb{R}^{rl}$ gegeben. Zudem setzen wir (DAE1)-(DAE3) *und* (GLM1) *voraus. Angenommen*

$$\begin{aligned} \eta &= (U \otimes I) y, & \xi &= (U \otimes I) z, \\ \bar{\eta} &= (U \otimes I) \bar{y}, & \bar{\xi} &= (U \otimes I) \bar{z} \end{aligned}$$

erfüllen die Voraussetzungen von Satz 3.7, so dass die Lösungen $Y(h, y), Z(h, y)$ und $Y(h, \bar{y}), Z(h, \bar{y})$ der entsprechenden Gleichungssysteme definiert sind. Zudem gelte

$$y - \bar{y} = \mathcal{O}(h), \qquad z - \bar{z} = o(1). \tag{3.23}$$

Dann existieren für einen hinreichend kleinen o-Term positive Konstanten K_1, K_2 und $h_0 > 0$, so dass für $h \leq h_0$ die folgenden Abschätzungen gelten:

$$\begin{aligned} \|Y(h, y) - Y(h, \bar{y})\| &\leq K_1 \|y - \bar{y}\|, \\ \|Z(h, y) - Z(h, \bar{y})\| &\leq \frac{K_1}{h} \|Dg(\eta)(y - \bar{y})\| + K_2 \|y - \bar{y}\|. \end{aligned} \tag{3.24}$$

Gilt für η und $\bar{\eta}$ sogar $g(\eta) = 0$ und $g(\bar{\eta}) = 0$, so ist in der z-Komponente die folgende strengere Abschätzung möglich:

$$\|Z(h, y) - Z(h, \bar{y})\| \leq K_2 \|y - \bar{y}\|.$$

Zusatz: Die Konstante K_1 hängt dabei nur von Schranken gewisser Ableitungen von f und g ab, nicht jedoch von den Konstanten, die implizit durch die $\mathcal{O}$- und o-Terme in (3.13) und (3.23) gegeben sind. K_2 hingegen hängt in linearer Weise von der Konstanten aus dem $\mathcal{O}$-Term in (3.23) ab.

Bemerkungen:

(i) *Der Beweis des Korollars wird deutlich machen, dass die Annahmen in (3.23) im Allgemeinen wirklich nötig sind, um brauchbare Abschätzungen zu erhalten.*

(ii) *Unter den gegebenen Voraussetzungen sind die rechten Seiten in den Abschätzungen klein:*

$$\begin{aligned} o(1) &= (g(\bar{\eta}) - g(\eta))/h \\ &= Dg(\eta)(\bar{\eta} - \eta)/h + \mathcal{O}(1)(\eta - \bar{\eta}). \end{aligned}$$

(iii) *Aufgrund der strengeren Abschätzung in der Z-Komponente ist die Konvergenzanalyse der Variablen y für gewisse allgemeine lineare Verfahren mehr oder weniger so einfach wie im Fall gewöhnlicher Differentialgleichungen (vgl. [HW] VII Beweis zu Theorem 4.5 für Runge-Kutta Verfahren). Dies sind insbesondere Runge-Kutta Vefahren, deren y-Iterationen in der Mannigfaltigkeit liegen, die durch $g = 0$ definiert ist. Für Stabilitätsuntersuchungen jedoch ist diese Abschätzung nicht länger brauchbar, da Störungen betrachtet werden, welche die Voraussetzung $g(y) = 0$ nicht länger erfüllen.*

Beweis des Korollars: Der Beweis besteht hauptsächlich in der Anwendung der Stabilitätsungleichung (3.14) von Satz 3.7.

Wir beginnen mit der Beobachtung, dass dieser Satz die Existenz von $h_0 > 0$ und $r > 0 \in \mathbb{R}^2$ garantiert, so dass $Y(h, y), Z(h, y)$ und $Y(h, \bar{y}), Z(h, \bar{y})$ wohldefiniert sind als Lösungen von

$$F(h, \eta, Y, Z) = 0 \qquad \text{bzw.} \qquad F(h, \bar{\eta}, Y, Z) = 0.$$

Dabei ist die Funktion F wie in (3.12) definiert.
Weiterhin können wir ohne Einschränkung wegen (3.15) und (3.23) davon ausgehen, dass gilt:

$$(Y(h, \bar{y}), Z(h, \bar{y})) \in B_r = \{V \in \mathbb{R}^{s(N+l)} | \, |V - (\eta, \xi)| \leq r\}.$$

Insbesondere ist die Stabilitätsungleichung des Satzes anwendbar.
Der Einfachheit halber definieren wir

$$\begin{aligned} Y &:= Y(h, y), & Z &:= Z(h, y), \\ \bar{Y} &:= Y(h, \bar{y}), & \bar{Z} &:= Z(h, \bar{y}). \end{aligned}$$

Die Stabilitätsungleichung garantiert nun

$$|(Y, Z) - (\bar{Y}, \bar{Z})| \leq (I - PK)^{-1} P |F(h, \eta, \bar{Y}, \bar{Z})|.$$

Zudem liefert $F(h, \bar{\eta}, \bar{Y}, \bar{Z}) = 0$ die Darstellung

$$F(h, \eta, \bar{Y}, \bar{Z}) = \begin{pmatrix} \bar{\eta} - \eta \\ (g(\eta) + \int_0^1 Dg(\eta + s(\bar{Y} - \eta))ds \cdot (\bar{Y} - \bar{\eta}))/h \end{pmatrix}.$$

Mit $\bar{Y} - \bar{\eta} = \bar{Y} - \eta + \eta - \bar{\eta}$ und $g(\bar{Y}) = 0$ folgern wir

$$F(h, \eta, \bar{Y}, \bar{Z}) = \begin{pmatrix} \bar{\eta} - \eta \\ \int_0^1 Dg(\eta + s(\bar{Y} - \eta))ds \cdot (\eta - \bar{\eta})/h \end{pmatrix}.$$

Wir betrachten nun die zweite Komponente etwas genauer. Mit der Darstellung $\bar{Y} - \bar{\eta} = \mathcal{O}(h)$ aus (3.15) erhalten wir durch eine Taylorentwicklung

$$\begin{aligned} \int_0^1 Dg(\eta + s(\bar{Y} - \eta))ds \cdot (\eta - \bar{\eta}) &= \int_0^1 Dg(\eta + s(\bar{\eta} - \eta))ds \cdot (\eta - \bar{\eta}) + \mathcal{O}(h)(\eta - \bar{\eta}) \\ &= g(\eta) - g(\bar{\eta}) + \mathcal{O}(h)(\eta - \bar{\eta}), \end{aligned}$$

wobei alle $\mathcal{O}$-Terme nur von Schranken gewisser Ableitungen von f und g abhängen, jedoch nicht von den Konstanten, die implizit durch die $\mathcal{O}$- und o-Terme in (3.13) und (3.23) gegeben sind.
Wiederum liefert eine Taylorentwicklung

$$g(\bar{\eta}) - g(\eta) = Dg(\eta)(\bar{\eta} - \eta) + \mathcal{O}(h)(\eta - \bar{\eta}),$$

wo nun $\mathcal{O}(h)$ in linearer Weise von der Konstante abhängt, die durch den $\mathcal{O}$-Term in (3.23) implizit gegeben ist.
Zusammenfassend ergibt sich

$$|(Y, Z) - (\bar{Y}, \bar{Z})| \leq (I - PK)^{-1}P| \begin{pmatrix} \bar{\eta} - \eta \\ Dg(\eta)(\eta - \bar{\eta})/h + \mathcal{O}(1)(\eta - \bar{\eta}) \end{pmatrix}|.$$

Die Existenz von K_1 und K_2 mit den geforderten Eigenschaften und somit die Gültigkeit der ersten beiden Abschätzungen folgen nun leicht aus

$$((I - PK)^{-1}P)_{12} = \mathcal{O}(h).$$

Dabei ist zu beachten, dass dieser $\mathcal{O}$-Term und die Matrizen P und K unabhängig von den Konstanten sind, die implizit durch die $\mathcal{O}$- und o-Terme in (3.13) und (3.23) gegeben sind.
Die strengere Abschätzung der Z-Komponente folgt leicht aus

$$0 = g(\bar{\eta}) - g(\eta) = Dg(\eta)(\bar{\eta} - \eta) + \mathcal{O}(\|\bar{\eta} - \eta\|^2).$$

□

Es ist aufgrund der lokalen Unabhängigkeit der Stufenwerte von ξ bzw. z nicht überraschend, dass die Abschätzungen des Korollars nicht von der Differenz der z-Komponenten abhängen. Problematisch ist aber die Abschätzung der Z-Komponente, bei welcher der Faktor $1/h$ eine Lipschitz-Stetigkeit der Verfahrensfunktion wie bei gewöhnlichen Differentialgleichungen unmöglich macht. Dies bedeutet insbesondere, dass die Glattheit der rechten Seite bei Index-2 DAEs nicht ausreichen wird, die

Stabilität nullstabiler Verfahren zu garantieren.

Wir suchen nun im Folgenden nach weiteren für die Stabilität notwendigen Eigenschaften. Doch bevor wir damit anfangen, definieren wir den Operator T_h. Dazu sei wieder ein äquidistantes Gitter zur Schrittweite $h > 0$ gegeben:

$$\Omega_h := \{x_j = jh |\ j = 0, ..., \sigma_h\}.$$

Die Abbildung

$$\sigma : [0, h_0[\to \mathbb{N}, \qquad h \mapsto \sigma_h$$

garantiert, dass das komplette Integrationsintervall $[0, x_e]$ von Ω_h erfasst wird, das heißt es gilt:

$$(\sigma_h - 1)h < x_e \leq \sigma_h h.$$

Die Einschränkungen der *correct value* Funktionen $y_c(x, h)$ und $z_c(x, h)$ auf das Gitter Ω_h bezeichnen wir mit y_h bzw. z_h. Wir definieren nun eine Umgebung

$$U_\delta := U^y \times U^z$$

von (y_h, z_h) in $(\mathbb{R}^{r(N+l)})^{\Omega_h}$, welche den Definitionsbereich des Operators T_h bilden wird. Die genaue Definition dieser Umgebung ist durch Bemerkung 3.8 (iii) und die Voraussetzungen von Korollar 3.10 motiviert:

$$\begin{aligned} U^y &:= \{u \in (\mathbb{R}^{rN})^{\Omega_h} |\ \|u - y_h\|_\infty \leq C_0 h, \|(U \otimes Q)(u - y_h)\|_\infty \leq \delta_1 h\}, \\ U^z &:= \{v \in (\mathbb{R}^{rl})^{\Omega_h} |\ \|v - z_h\|_\infty \leq \delta_2\}. \end{aligned}$$

Dabei ist die Supremumsnorm für eine beliebige Norm $\|.\|$ auf $\mathbb{R}^M$ für $u \in (\mathbb{R}^M)^{\Omega_h}$ in natürlicher Weise definiert durch:

$$\|u\|_\infty := \max\{\|u(x_j)\| |\ j = 0, ..., \sigma_h\}.$$

Tatsächlich ist U_δ eine "maximale" Umgebung, um für zwei Funktionenpaare $(\hat{y}, \hat{z})$ und $(\bar{y}, \bar{z})$ aus U_δ sowohl die Anwendbarkeit von Satz 3.7 als auch die des Korollars 3.10 für hinreichend kleine h und δ_i in jedem Punkt x_j garantieren zu können. Wir gehen im Folgenden davon aus, dass h und die δ_i entsprechend klein gewählt sind.

Bemerkung: *Es ist nicht erstaunlich aber bemerkenswert, dass die Umgebung U_δ, auf welcher wir die Stabilität des Operators T_h nachweisen wollen, von h abhängt und mit $h \to 0$ "zusammen schrumpft".*

Den Operator

$$T_h : U_\delta \to (\mathbb{R}^{r(N+l)})^{\Omega_h}, \qquad T_h(u, v) = (T_h^y u, T_h^z(u, v))$$

definieren wir, indem wir die Operatoren der beiden Komponenten in (3.25) bzw. (3.38) konkretisieren.

Wir untersuchen nun die Stabilität zunächst in der y- und dann in der z-Komponente.

Stabilität der y-Komponente
Wir definieren wieder den Iterationsoperator

$$N^y u = (V \otimes I)u + h\Phi(h, u)$$

und interpretieren die y-Komponente der numerischen Lösung als Nullstelle des Operators

$$T_h^y : U^y \to (\mathbb{R}^{rN})^{\Omega_h}$$

definiert durch

$$T_h^y u(x_j) = \begin{cases} u(0) - \varphi^y(h, y_0, z_0) \text{ für } j = 0, \\ h^{-1}(u(x_j) - N^y u(x_{j-1})) \text{ für } j > 0. \end{cases} \tag{3.25}$$

Wir leiten nun eine Stabilitätsungleichung für das numerische Modell

$$T_h^y y = 0$$

her: Es seien dazu $(\hat{y}, \hat{z})$ und $(\bar{y}, \bar{z})$ zwei Elemente aus U_δ. Aufgrund der Definition von T_h^y gilt für die Differenzen $\Delta y_j = \hat{y}(x_j) - \bar{y}(x_j)$ wie bei den gewöhnlichen Differentialgleichungen

$$\Delta y_{j+1} = (V \otimes I)\Delta y_j + h\Delta\Phi_j + h\Delta T_{j+1}^y. \tag{3.26}$$

Wir haben dabei die gleichen Schreibweisen wie oben benutzt. Bei den gewöhnlichen Differentialgleichungen lieferte die Glattheit der rechten Seite f die Lipschitz-Stetigkeit der Verfahrensfunktion und somit Stabilität, wenn V stabil ist. Aufgrund der zweiten Abschätzung (3.24) in Korollar 3.10 wird diese Folgerung trotz Glattheit der rechten Seite bei Index-2 DAEs in Hessenberg Form nicht möglich sein. Wir werden also $\Delta\Phi_j$ genauer untersuchen müssen und die Anwendung der Abschätzung (3.24) wenn möglich vermeiden. Tatsächlich werden wir die Differenz zweier Stufenwerte der z-Komponente lieber ersetzen, statt sie durch Korollar 3.10 abzuschätzen. Dies ermöglicht uns die genaue Struktur dieses Summanden herauszuarbeiten.
Wir führen dazu weitere abkürzende Schreibweisen ein:

$$\begin{aligned} \hat{Y}_j &:= Y(h, \hat{y}(x_j)), \\ \bar{Y}_j &:= Y(h, \bar{y}(x_j)), \\ \Delta Y_j &:= \hat{Y}_j - \bar{Y}_j. \end{aligned}$$

Für die z Variable sind alle Schreibweisen entsprechend.

Wir betrachten zunächst den einfachen Fall:

$$f(y, z) = f_0(y) + f_z z$$

mit konstanter Matrix f_z. Hier wird das weitere Vorgehen sehr schön deutlich. Wir finden in diesem Fall

$$h\Delta\Phi_j = h(B \otimes I)(f_0(\hat{Y}_j) - f_0(\bar{Y}_j)) + h(B \otimes f_z)\Delta Z_j. \tag{3.27}$$

Der erste Summand lässt sich aufgrund von Korollar 3.10 unproblematisch durch $O(h\|\Delta y_j\|)$ abschätzen. Den zweiten sollten wir lieber ersetzen. Wir benutzen dazu das Gleichungssystem der Stufenwerte. Die erste Gleichung dieses Systems liefert:

$$\Delta Y_j = (U \otimes I)\Delta y_j + h(A \otimes I)(f_0(\hat{Y}_j) - f_0(\bar{Y}_j)) + h(A \otimes f_z)\Delta Z_j. \tag{3.28}$$

Die Linearisierung der zweiten Gleichung ist gegeben durch

$$\begin{aligned} 0 &= g(\hat{Y}_j) - g(\bar{Y}_j) \\ &= Dg(\bar{Y}_j)\Delta Y_j + \mathcal{O}(\|\Delta Y_j\|^2) \\ &= (I \otimes Dg^j)\Delta Y_j + \mathcal{O}(h\|\Delta y_j\|). \end{aligned} \tag{3.29}$$

Der Index j an der partiellen Ableitung von g und später auch an den partiellen Ableitungen von f kennzeichnet eine Auswertung der Funktion entlang der Lösung in den Punkten x_j.
Für die Herleitung von (3.29) sei nochmals an (3.15), die Definition von U^y und die Darstellung der *correct value* Funktion (GLM4) erinnert, woraus folgt:

$$\begin{aligned} \bar{Y}_j &= (U \otimes I)\bar{y}(x_j) + \mathcal{O}(h) \\ &= (U \otimes I)y_c(x_j, h) + \mathcal{O}(h) \;=\; \mathbb{1} \otimes y(x_j) + \mathcal{O}(h). \end{aligned}$$

Infolgedessen ist der untere $\mathcal{O}$-Term in (3.29) abhängig von C_0 aus der Definition der Umgebung U^y, denn es gilt:

$$\|\bar{y}(x_j) - y_c(x_j, h)\| \leq C_0 h.$$

Zusätzlich haben wir in (3.29) die Abschätzung für ΔY_j aus Korollar 3.10 benutzt.

Bemerkung 3.11 *Im Fall*

$$\bar{y}(x_j) = y_c(x_j, h)$$

sind die $\mathcal{O}$-Terme in (3.29) *unabhängig von C_0.*

Setzen wir nun (3.28) in (3.29) ein, so können wir aufgrund der Index-2 Bedingung und der Invertierbarkeit von A nach $h\Delta Z_j$ auflösen:

$$h\Delta Z_j = (-A^{-1}U \otimes (Dg^j f_z)^{-1} Dg^j)\Delta y_j + \mathcal{O}(h\|\Delta y_j\|),$$

wobei wir wieder die Abschätzung für ΔY_j aus Korollar 3.10 benutzt haben. Setzen wir dies in die Darstellung von $h\Delta\Phi_j$ (3.27) ein, so finden wir wiederum mit dieser Abschätzung

$$h\Delta\Phi_j = -(BA^{-1}U \otimes f_z(Dg^j f_z)^{-1} Dg^j)\Delta y_j + \mathcal{O}(h\|\Delta y_j\|).$$

Dies ergibt mit den bereits oben definierten Projektionen

$$Q_j := Q(x_j) = f_z(Dg^j f_z)^{-1} Dg^j, \qquad P_j = I - Q_j$$

für die Ausgangsiteration (3.26) die Darstellung

$$\begin{aligned} \Delta y_{j+1} &= (V \otimes I)\Delta y_j - (BA^{-1}U \otimes Q_j)\Delta y_j + \mathcal{O}(h\|\Delta y_j\|) + h\Delta T^y_{j+1} \\ &= (V \otimes P_j)\Delta y_j + (M(\infty) \otimes Q_j)\Delta y_j + \mathcal{O}(h\|\Delta y_j\|) + h\Delta T^y_{j+1}. \end{aligned}$$

Dabei ist $M(\infty) = V - BA^{-1}U$ und $M(z) = V + zB(I - zA)^{-1}U$ die matrixwertige Stabilitätsfunktion des Verfahrens.
Eine sehr interessante Beobachtung ist, dass wir in der Herleitung dieser Iteration die "Index-2 Abschätzung" (3.24) aus Korollar 3.10 nicht verwendet haben. Dies passt hervorragend zu Theorem 1b) aus [ASW95]. Das Theorem besagt nämlich, dass unter der Annahme

$$f(y, z) = f_0(y) + f_z z, \qquad f_z \text{ konstant},$$

die y Variable eine Index-1 Variable ist. Es bleibt die Frage zu klären, ob eine ähnliche Iteration auch für beliebige f möglich ist. Eine Antwort gibt das folgende Lemma.

Lemma 3.12
Angenommen für die Index-2 DAE (3.5) *gelte* (DAE1)-(DAE3). *Für das allgemeine lineare Verfahren setzen wir* (GLM1) *und* (GLM4) *voraus. Seien* $(\hat{y}, \hat{z})$ *und* $(\bar{y}, \bar{z})$ *aus der Umgebung* U_δ. *Dann gilt für die Differenzen*

$$\Delta y_j = \hat{y}(x_j) - \bar{y}(x_j)$$

die folgende iterative Darstellung (ohne das Kronecker Produkt):

$$\Delta y_{j+1} = VP_j\Delta y_j + M(\infty)Q_j\Delta y_j + \mathcal{O}(h\|\Delta y_j\|) + \mathcal{O}(\delta\|Q_j\Delta y_j\|) + h\Delta T^y_{j+1}.$$

Dabei ist $\delta = \max \delta_i$.
Im Fall

$$\bar{y}(x_j) = y_c(x_j, h)$$

sind die $\mathcal{O}$*-Terme unabhängig von* C_0 *aus der Definition von* U_δ.

Die Stabilität eines allgemeinen linearen Verfahrens in der y-Komponente ist somit durch die Stabilität der Iteration aus Lemma 3.12 gegeben, welche wiederum durch die Matrizen V und $M(\infty)$ entschieden wird. Das folgenden Lemma, welches eine Verallgemeinerung von Lemma 4.5 aus [HLR89] darstellt, gibt genauere Auskunft darüber.

Stabilitätslemma 3.13
Gegeben sei eine Iteration in $\mathbb{R}^m$ der Form

$$\xi_{j+1} = VP_j\xi_j + MQ_j\xi_j + \mathcal{O}(\delta\|Q_j\xi_j\|) + \mathcal{O}(h\|\xi_j\|) + hr_{j+1} \tag{3.30}$$

für $j \geq 0$. Zudem gelte:

- *Die Matrizen M und V sind stabil; es gilt sogar $\rho(M) < 1$ und $1 \in \sigma(V)$.*
- *Die Projektionen P_j, $Q_j := I_m - P_j$ auf $\mathbb{R}^m$ kommutieren mit V und M und es gilt*
$$P_{j+1} = P_j + \mathcal{O}(h).$$
- *Für $j > 0$ existieren nicht negative Konstanten D_1 und D_2 mit*
$$\|P_jr_j\| \leq D_1, \qquad \|Q_jr_j\| \leq D_2.$$

Dann existieren $\delta_0 > 0$, $h_0 > 0$ und positive Konstanten C und $\rho < 1$, so dass die folgenden Abschätzungen

$$\|\xi_n\| \leq C\Big(\|\xi_0\| + D_1 + (h+\delta)D_2\Big), \tag{3.31}$$

$$\|Q_n\xi_n\| \leq C\Big(h\|P_0\xi_0\| + (\rho^n + h)\|Q_0\xi_0\| + hD_1 + hD_2\Big) \tag{3.32}$$

für alle $\delta \leq \delta_0$, $h \leq h_0$ und $nh \leq$ konst. gelten.

Zusatz: Sei $\delta = \mathcal{O}(h)$.
(i) *Ist M nilpotent vom Index k_0, so gilt:*

$$\|P_n\xi_n\| \leq C\Big(\|P_0\xi_0\| + h\|Q_0\xi_0\| + D_1 + hD_2\Big) \tag{3.33}$$

für alle $h \leq h_0$ und $nh \leq$ konst. Zudem erhalten wir die Abschätzung

$$\|Q_{k_0}\xi_{k_0}\| \leq C\Big(h\|\xi_0\| + hD_1 + hD_2\Big). \tag{3.34}$$

(ii) *Im Fall $\rho(M) = 1$ gilt die folgende Abschätzung:*

$$\|\xi_n\| \leq C\Big(\|\xi_0\| + D_1 + D_2\Big).$$

Für die projizierten Iterationen $Q_j\xi_j$ ist in diesem Fall keine Abschätzung wie in (3.32) *mehr möglich.*

Bemerkung: *Die Abschätzungen* (3.33) *und* (3.34) *verdeutlichen, dass im Fall der Nilpotenz von $M(\infty)$ eine h-Potenz unter der Projektion Q_j gewonnen wird.*

Die beiden Lemmata erklären die Annahmen (GLM2) und (GLM3) und motivieren folgende Definition:

Definition 3.14 *Wir nennen ein allgemeines lineares Verfahren stabil im Unendlichen, wenn die Matrix $M(\infty)$ stabil ist. Dabei ist $M(z) = V - zB(I - zA)^{-1}U$ die matrixwertige Stabilitätsfunktion des Verfahrens.*

Für die Projektionen P_j und Q_j aus der Darstellung in Lemma 3.12 gilt

$$P_j = I - Q_j, \qquad P_{j+1} = P_j + \mathcal{O}(h).$$

Letzteres ergibt sich aus einer Taylorentwicklung von $P_{j+1} = P(x_j + h)$ um x_j. Wir setzen

$$r_j := \Delta T_j^y \qquad D_i := \|\Delta T^y\|_\infty.$$

Wir erinnern an die triviale Gleichung $\Delta y_0 = \Delta T_0^y$. Eine Anwendung des Stabilitätslemmas 3.13 auf die Iteration der Δy_j liefert nun für eine positive Konstante C^y die Stabilitätsungleichung

$$\|\hat{y} - \bar{y}\|_\infty \leq C^y \|T_h^y \hat{y} - T_h^y \bar{y}\|_\infty.$$

Wir halten fest: Die zusätzliche Eigenschaft, die wir für die Stabilität des Verfahrens in der y-Komponente fordern müssen, ist die Stabilität im Unendlichen. Wir definieren daher:

Definition 3.15 *Wir nennen ein allgemeines lineares Verfahren für Index-2 DAEs in Hessenberg Form stabil in y, wenn es nullstabil und stabil im Unendlichen ist.*

Wir haben noch die Beweise der beiden Lemmata oben zu führen:

Beweis von Lemma 3.12: Das Vorgehen orientiert sich an dem oben behandelten Spezialfall.
Zunächst liefert eine Taylorentwicklung

$$\begin{aligned} h\Delta\Phi_j &= h(B \otimes I)\Big(\frac{\partial f}{\partial Y}(\bar{Y}_j, \bar{Z}_j)\Delta Y_j + \frac{\partial f}{\partial Z}(\bar{Y}_j, \bar{Z}_j)\Delta Z_j\Big) \\ &\qquad + \mathcal{O}(h\|\Delta Y_j\|^2) + \mathcal{O}(h\|\Delta Z_j\|^2). \end{aligned}$$

Die ΔY_j-Terme sind unproblematisch und können nach Korollar 3.10 durch $\mathcal{O}(\|\Delta y_j\|)$ abgeschätzt werden. Weiter folgt aus diesem Korollar

$$h\|\Delta Z_j\|^2 \leq K_1^2/h\|(U \otimes Q_j)\Delta y_j\|^2 + 2K_1K_2\|(U \otimes Q_j)\Delta y_j\|\|\Delta y_j\| + \mathcal{O}(h\|\Delta y_j\|).$$

Ohne Einschränkung besitzen die K_i die Eigenschaften der Konstanten aus Korollar 3.10. Dabei ist mit $C_0^i h < \delta$, $i = 1, 2$, der $\mathcal{O}$-Term unabhängig von C_0. Man beachte dabei, dass C_0 in die Konstante K_2 in linearer Weise eingeht. Im Unterschied zu dem Spezialfall haben wir hier die Abschätzung (3.24) verwendet.
Insgesamt folgt

$$h\Delta\Phi_j = h(B \otimes I)\frac{\partial f}{\partial Z}(\bar{Y}_j, \bar{Z}_j)\Delta Z_j + \mathcal{O}(\delta\|(I \otimes Q_j)\Delta y_j\|) + \mathcal{O}(h\|\Delta y_j\|).$$

Wir können die Argumente in der partiellen Ableitung aufgrund von (3.15) und der Definition von U_δ durch die exakte Lösung ersetzen:

$$h\Delta\Phi_j = h(B \otimes f_z^j)\Delta Z_j + \mathcal{O}(\delta\|(I \otimes Q_j)\Delta y_j\|) + \mathcal{O}(h\|\Delta y_j\|).$$

Wieder kann mit $C_0^i h < \delta$, $i = 1, 2$, die Unabhängigkeit der $\mathcal{O}$-Terme von C_0 garantiert werden.
In genau derselben Art und Weise erhalten wir

$$\Delta Y_j = (U \otimes I)\Delta y_j + h(A \otimes f_z^j)\Delta Z_j + \mathcal{O}(\delta\|(I \otimes Q_j)\Delta y_j\|) + \mathcal{O}(h\|\Delta y_j\|).$$

Setzen wir dies wieder in die Linearisierung (3.29) ein und lösen nach $h\Delta Z_j$ auf, so folgt

$$h\Delta\Phi_j = -(BA^{-1}U \otimes Q_j)\Delta y_j + \mathcal{O}(\delta\|(I \otimes Q_j)\Delta y_j\|) + \mathcal{O}(h\|\Delta y_j\|).$$

Durch die Linearisierung (3.29) handeln wir uns wie oben im Allgemeinen die Abhängigkeit des zweiten $\mathcal{O}$-Terms von C_0 ein. Die Aussage des Lemmas folgt nun, indem wir dies in die Ausgangsiteration (3.26) einsetzen.

□

Beweis von Lemma 3.13: Die Vorgehensweise basiert auf dem Beweis von Lemma 4.5 aus [HLR89].

Sei zunächst M nur als stabil vorausgesetzt. Aufgrund der Stabilität der Matrizen V und M existieren Normen $\|.\|^*$ und $\|.\|_*$ auf $\mathbb{R}^m$ mit

$$\|V\|^* = 1, \qquad \|M\|_* \leq \rho(M) \leq 1$$

(vgl. [SB] Satz 6.9.2).
Wir wenden nun die Projektionen P_{j+1} und Q_{j+1} jeweils auf die obige Iteration (3.30) an und finden mit den vorausgesetzten Eigenschaften der Projektionen

$$\begin{aligned} P_{j+1}\xi_{j+1} &= VP_j\xi_j + \mathcal{O}(\delta\|Q_j\xi_j\|) + \mathcal{O}(h\|\xi_j\|) + hP_{j+1}r_{j+1}, \\ Q_{j+1}\xi_{j+1} &= MQ_j\xi_j + \mathcal{O}(\delta\|Q_j\xi_j\|) + \mathcal{O}(h\|\xi_j\|) + hQ_{j+1}r_{j+1}. \end{aligned} \tag{3.35}$$

Wir definieren $v_j := \|P_j\xi_j\|^*$ und $w_j := \|Q_j\xi_j\|_*$ und erhalten mit der Dreiecksungleichung

$$\begin{pmatrix} v_{j+1} \\ w_{j+1} \end{pmatrix} \leq \left[\begin{pmatrix} 1 & \mathcal{O}(\delta) \\ 0 & \|M\|_* + \mathcal{O}(\delta) \end{pmatrix} + \mathcal{O}(h) \right] \begin{pmatrix} v_j \\ w_j \end{pmatrix} + h \begin{pmatrix} D_1 \\ D_2 \end{pmatrix}.$$

Wir betrachten im Folgenden den Fall $\|M\|_* < 1$. Für hinreichend kleine δ existiert ein $\rho < 1$ mit

$$\|M\|_* + \mathcal{O}(\delta) \leq \rho. \tag{3.36}$$

Hier sei darauf hingewiesen, dass im Fall $\delta = \mathcal{O}(h)$ nur $\|M\|_* \leq \rho$ garantiert werden muss, so dass für $M = 0$ auch $\rho = 0$ ausreichend ist.
Die Eigenwerte der Matrix

$$B := \begin{pmatrix} 1 & \mathcal{O}(\delta) \\ 0 & \rho \end{pmatrix} + \mathcal{O}(h)$$

sind $\lambda_1 = 1+\mathcal{O}(h)$ und $\lambda_2 = \rho+\mathcal{O}(h)$ und somit für hinreichend kleine h verschieden. Für solche h können wir B diagonalisieren mit Transformationsmatrix

$$S := \begin{pmatrix} 1 & \mathcal{O}(\delta) + \mathcal{O}(h) \\ \mathcal{O}(h) & 1 \end{pmatrix},$$

d.h. es gilt

$$SBS^{-1} = D \text{ mit } D := \operatorname{diag}(\lambda_1, \lambda_2).$$

Wir finden daher mit obiger Ungleichung

$$\begin{pmatrix} v_n \\ w_n \end{pmatrix} \leq S^{-1}D^nS \begin{pmatrix} v_0 \\ w_0 \end{pmatrix} + h\sum_{j=0}^{n-1} S^{-1}D^jS \begin{pmatrix} D_1 \\ D_2 \end{pmatrix}. \tag{3.37}$$

Für $j = 0, ..., n$ finden wir die Darstellung

$$\begin{aligned} S^{-1}D^jS &= \begin{pmatrix} 1 & \mathcal{O}(\delta) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \lambda_1^j & 0 \\ 0 & \lambda_2^j \end{pmatrix} \begin{pmatrix} 1 & \mathcal{O}(\delta) \\ 0 & 1 \end{pmatrix} + \mathcal{O}(h) \\ &= \begin{pmatrix} \lambda_1^j & \mathcal{O}(\delta) \\ 0 & \lambda_2^j \end{pmatrix} + \mathcal{O}(h). \end{aligned}$$

Aus $\lambda_1 = 1 + \mathcal{O}(h)$ und $\lambda_2 = \rho + \mathcal{O}(h)$ folgt für hinreichend kleine h

$$h\sum_{j=0}^{n-1} S^{-1}D^jS = \begin{pmatrix} \mathcal{O}(1) & \mathcal{O}(\delta) \\ 0 & 0 \end{pmatrix} + \mathcal{O}(h).$$

Setzen wir dies in (3.37) ein, so erhalten wir für hinreichend großes $C > 0$ die Abschätzung

$$\begin{pmatrix} v_n \\ w_n \end{pmatrix} \leq \left[\begin{pmatrix} \lambda_1^n & \mathcal{O}(\delta) \\ 0 & \lambda_2^n \end{pmatrix} + \mathcal{O}(h) \right] \begin{pmatrix} v_0 \\ w_0 \end{pmatrix} + C \begin{pmatrix} D_1 + (\delta + h)D_2 \\ h(D_1 + D_2) \end{pmatrix}.$$

Nach eventueller Vergrößerung von C schreiben wir dies für die einzelnen Komponenten:

$$\begin{aligned} \|P_n\xi_n\|^* &\leq C[\|P_0\xi_0\|^* + (\delta + h)\|Q_0\xi_0\|_* + D_1 + (\delta + h)D_2], \\ \|Q_n\xi_n\|_* &\leq C[h\|P_0\xi_0\|^* + (\rho^n + h)\|Q_0\xi_0\|_* + hD_1 + hD_2]. \end{aligned}$$

Wir vergrößern eventuell abschließend $C > 0$ erneut, so dass mit der Äquivalenz der Normen auf $\mathbb{R}^m$ und

$$\|\xi_n\| \leq \|P_n\xi_n\| + \|Q_n\xi_n\|$$

die geforderten Abschätzungen für eine beliebige Norm gelten.

Um den Zusatz zu beweisen, gehen wir von $\delta = \mathcal{O}(h)$ aus. Sei nun M sogar nilpotent mit Index k_0, d.h. k_0 ist die kleinste Zahl mit

$$M^{k_0} = 0.$$

Wir erinnern an die untere Gleichung in (3.35), die für $\delta = \mathcal{O}(h)$ lautet:

$$Q_{j+1}\xi_{j+1} = MQ_j\xi_j + \mathcal{O}(h\|\xi_j\|) + hQ_{j+1}r_{j+1}.$$

Daraus folgt

$$Q_n\xi_n = M^nQ_0\xi_0 + \sum_{i=0}^{n-1} M^{n-1-i}(\mathcal{O}(h\|\xi_i\|) + hQ_ir_i).$$

Für $n = k_0$ finden wir mit der Nilpotenz der Matrix M die Abschätzung

$$\|Q_{k_0}\xi_{k_0}\| = Ch(\max_{0\le i<k_0} \|\xi_i\| + D_2)$$

für eine positive Konstante C. Nach eventueller Vergrößerung von C folgt mit den bereits bewiesenen Abschätzungen für $\|\xi_i\|$ die Ungleichung

$$\|Q_{k_0}\xi_{k_0}\| = Ch(\|\xi_0\| + D_1 + D_2).$$

Besitzt M einen Eigenwert auf dem Rand des Einheitskreises, d.h. gilt $\|M\|_* = 1$, so kann die Stabilität der Matrix

$$B := \begin{pmatrix} 1 & \mathcal{O}(\delta) \\ 0 & 1 + \mathcal{O}(\delta) \end{pmatrix}$$

nur garantiert werden, wenn $(1 + \mathcal{O}(\delta))^n$, also $n\delta$ beschränkt ist. Dies erklärt die Forderung $\delta = \mathcal{O}(h)$. In diesem Fall liefert die Beschränktheit von B unmittelbar die Abschätzung

$$\|\xi_n\| \;\le\; C\Big(\|\xi_0\| + D_1 + D_2\Big).$$

□

Stabilität der z-Komponente
Wir definieren mit Hilfe des Iterationsoperators

$$N^z(u, v) = (M(\infty) \otimes I)v + (BA^{-1} \otimes I)Z(h, u)$$

den Operator

$$T_h^z : U_\delta \to (\mathbb{R}^{rl})^{\Omega_h}$$

durch

$$T_h^z(u,v)(x_j) = \begin{cases} v(0) - \varphi^z(h, y_0, z_0) \text{ für } j = 0, \\ v(x_j) - N^z(v(x_{j-1}), u(x_{j-1})) \text{ für } j > 0. \end{cases} \tag{3.38}$$

Das folgende Lemma entspricht Lemma 3.12 und 3.13:

Lemma 3.16 *Angenommen für die Index-2 DAE* (3.5) *gelte* (DAE1)-(DAE3). *Für das allgemeine lineare Verfahren setzen wir* (GLM1),(GLM3) *und* (GLM4) *voraus. Seien* $(\hat{y}, \hat{z})$ *und* $(\bar{y}, \bar{z})$ *aus der Umgebung* U_δ. *Dann existieren positive Konstanten* C *und* D, *so dass für die Differenzen*

$$\Delta z_j = \hat{z}(x_j) - \bar{z}(x_j)$$

folgt:

(i) *Für* $\rho(M(\infty)) < 1$ *und* $nh \leq$ *konst. gilt die Abschätzung*

$$\|\Delta z_n\| \leq D\Big(\|\Delta T^y\|_\infty + \|Q_0 \frac{\Delta T_0^y}{h}\|\Big) + C\|\Delta T^z\|_\infty.$$

Im Spezialfall $M(\infty) = 0$ *gilt unter der zusätzlichen Voraussetzung* $\delta_i = \mathcal{O}(h)$ *sogar*

$$\|\Delta z_n\| \leq D\|\Delta T^y\|_\infty + C\|\Delta T^z\|_\infty.$$

(ii) *Für* $\rho(M(\infty)) = 1$ *und* $nh \leq$ *konst. gilt die Abschätzung*

$$\|\Delta z_n\| \leq D\Big(\|\frac{\Delta T^y}{h}\|_\infty + \|Q\frac{\Delta T^y}{h^2}\|_\infty\Big) + C\|\frac{\Delta T^z}{h}\|_\infty.$$

Beweis: Seien also $(\hat{y}, \hat{z})$ und $(\bar{y}, \bar{z})$ zwei Elemente aus U_δ. Mit der Definition von T_h^z folgt unmittelbar die Darstellung

$$\Delta z_{j+1} = M(\infty)\Delta z_j + BA^{-1}\Delta Z_j + \Delta T_{j+1}^z.$$

Hier haben wir dieselben Schreibweisen wie im Fall der y-Komponente benutzt. Mit der Dreiecksungleichung und der Abschätzung (3.24) aus Korollar 3.10 folgt

$$\|\Delta z_{j+1}\| \leq \|M(\infty)\|\|\Delta z_j\| + \frac{K_1}{h}\|Q_j\Delta y_j\| + K_2\|\Delta y_j\| + \|\Delta T^z\|_\infty$$

für zwei positive Konstanten K_1 und K_2. Dabei spielen die Eigenschaften der Konstanten K_i, welche im Zusatz von Korollar 3.10 formuliert sind, hier keine Rolle.

Zu (i): Gelte für die Norm $\|.\|$ ohne Einschränkung

$$\|M(\infty)\| =: \rho < 1$$

(vgl. zur Wahl der Norm [SB] Satz 6.9.2). Da ρ kleiner als eins ist, existieren positive Konstanten K_3 und K_4 mit

$$\|\Delta z_{j+1}\| \leq \rho^{j+1}\|\Delta z_0\| + K_3\|\Delta T^y\|_\infty + K_4\|\Delta T^z\|_\infty + \frac{K_1}{h}\sum_{i=0}^{j}\rho^{j-i}\|Q_i\Delta y_i\|. \quad (3.39)$$

Hier haben wir zusätzlich die Stabilität des Operators T_h^y ausgenutzt.
Offenbar gilt

$$\|\Delta z_0\| = \|\Delta T_0^z\|.$$

Der letzte Term in der Abschätzung (3.39) wird uns aufgrund des Faktors $1/h$ etwas Arbeit abverlangen. Nach der Ungleichung (3.32) aus dem Stabilitätslemma 3.13, welches wir oben auf die Iterationen Δy_j angewendet haben, gilt

$$\|Q_i\Delta y_i\| \leq K_5\Big((\rho^i + h)\|Q_0\Delta y_0\| + h\|\Delta T^y\|_\infty\Big). \quad (3.40)$$

Hier ist K_5 eine entsprechend große positive Konstante. Das ρ in dieser Abschätzung kann im Grunde mit dem aus (3.39) identifiziert werden. Einzige Ausnahme bildet der Spezialfall $M(\infty) = 0$. Während das ρ in Abschätzung (3.39) in diesem Fall identisch Null ist, kann das ρ in (3.40) nur unter der zusätzlichen Annahme $\delta_i = \mathcal{O}(h)$ gleich Null gesetzt werden (vgl. die Bemerkung nach (3.36) im Beweis von Lemma 3.13). Damit ist jedoch die Aussage des Lemmas für den Spezialfall bereits bewiesen. Setzen wir die Abschätzung (3.40) in die Summe von (3.39) ein, so folgt

$$\frac{1}{h}\sum_{i=0}^{j}\rho^{j-i}\|Q_i\Delta y_i\| \leq K_6\|\Delta T^y\|_\infty + \frac{K_5}{h}\rho^j(j+1)\|Q_0\Delta y_0\|$$

für eine positive Konstante K_6. Tatsächlich ist die Folge $(\rho^j(j+1))_{j\geq 0}$ beschränkt, aber weiterhin macht der Faktor $1/h$ für $\rho > 0$ Probleme. Setzen wir dies in (3.39) ein, so werden wir eine h Potenz bei x_0 opfern müssen, um Stabilität zu erhalten:

$$\|\Delta z_{j+1}\| \leq D\Big(\|\Delta T^y\|_\infty + \|Q_0\frac{\Delta T_0^y}{h}\|\Big) + C\|\Delta T^z\|_\infty.$$

Dabei sind C und D entsprechend groß gewählte positive Konstanten.

Zu (ii): Im Fall $\rho(M(\infty)) = 1$ gehen wir zu einer entsprechenden Norm $\|.\|_*$ über mit

$$\|M(\infty)\|_* = 1$$

(vgl. [SB] Satz 6.9.2). Wir erhalten also:

$$\|\Delta z_{j+1}\|_* \leq \|\Delta z_j\|_* + \frac{K_1}{h}\|Q_j\Delta y_j\|_* + K_2\|\Delta y_j\| + \|\Delta T^z\|_\infty.$$

Opfern wir eine h Potenz, d.h. betrachten wir $\frac{1}{h}T_h$, so sind aufgrund der Stabilität von T_h^y die beiden letzten Terme unproblematisch:

$$\|\Delta z_{j+1}\|_* \leq \|\Delta z_0\|_* + \frac{K_1}{h}\sum_{i=0}^{j}\|Q_i\Delta y_i\|_* + K_3\|\frac{\Delta T^y}{h}\|_\infty + K_4\|\frac{\Delta T^z}{h}\|_\infty.$$

Leider reicht dies nicht aus, um den Term mit dem Faktor $1/h$ in den Griff zu bekommen: Eine h Potenz des Operators T_h^y müssen wir investieren, da wir für $Q_i\Delta y_i$ nach dem Stabilitätslemma keinen Faktor h mehr gewinnen:

$$\|Q_i\Delta y_i\| \leq K_5\Big(\|\Delta y_0\| + \|\Delta T^y\|_\infty\Big).$$

Zusätzlich wird auch die Summe eine h Potenz verschlucken, da $\rho^j(j+1) = j+1$ nicht länger beschränkt ist. Insgesamt folgt:

$$\|\Delta z_n\| \leq D\Big(\|\frac{\Delta T^y}{h}\|_\infty + \|Q\frac{\Delta T^y}{h^2}\|_\infty\Big) + C\|\frac{\Delta T^z}{h}\|_\infty$$

für hinreichend große positive Konstanten C und D.

□

Insgesamt sehen wir, dass streng genommen nur im Spezialfall $M(\infty) = 0$ das allgemeine lineare Verfahren in seiner ursprünglichen Form stabil ist. Für $\rho(M(\infty)) < 1$ erhält man Stabilität nach einer kleinen Modifikation des Operators T_h, genauer von T_h^y. Für $\rho(M(\infty)) = 1$ genügt der Operator T_h keiner Stabilitätsungleichung gleichmäßig für kleine h. Wir nehmen diese Überlegungen zum Anlass für die Definition:

Definition 3.17 *Wir nennen ein allgemeines lineares Verfahren für Index-2 DAEs in Hessenberg Form stabil in z, wenn es nullstabil ist und $\rho(M(\infty)) < 1$ gilt. Zusätzlich bezeichnen wir ein allgemeines lineares Verfahren als stabil, wenn es sowohl in y als auch in z stabil ist.*

Wir fassen nun alle Beobachtungen in einem Satz zusammen. Dabei werden wir neben den beiden Variablen y und z auch die Fälle $\rho(M(\infty))$ gleich oder kleiner eins getrennt betrachten müssen. Zudem brachte die Annahme $\delta_i = \mathcal{O}(h)$ in manchen Fällen eine Verbesserung. Wir bezeichnen daher den Definitionsbereich des Operators $T_h = (T_h^y, T_h^z)$ etwas genauer mit U_h, falls sich die δ_i wie $\mathcal{O}(h)$ verhalten. Um nun den Satz formulieren zu können, definieren wir die verallgemeinerte Norm

$$|(u,v)| = (\|u\|_\infty, \|v\|_\infty)^T$$

für $u \in (\mathbb{R}^{rN})^{\Omega_h}$ und $v \in (\mathbb{R}^{rl})^{\Omega_h}$. Zudem wiederholen wir die wichtigsten Definitionen: Für $C_0 > 0$ definieren wir die Umgebung

$$U_\delta = U^y \times U^z$$

durch

$$\begin{aligned} U^y &:= \{u \in (\mathbb{R}^{rN})^{\Omega_h} |\ \|u - y_h\|_\infty \le C_0 h, \|(U \otimes Q)(u - y_h)\|_\infty \le \delta_1 h\}, \\ U^z &:= \{v \in (\mathbb{R}^{rl})^{\Omega_h} |\ \|v - z_h\|_\infty \le \delta_2\}. \end{aligned}$$

Gilt $\delta_i = \mathcal{O}(h)$, so ersetzen wir den Index durch h.
Wir definieren den Operator

$$T_h : U_\delta \to (\mathbb{R}^{r(N+l)})^{\Omega_h}, \qquad T_h(u,v) := (T_h^y(u), T_h^z(u,v))$$

durch

$$\begin{aligned} T_h^y u(x_j) &= \begin{cases} u(0) - \varphi(h, y_0) \text{ für } j = 0, \\ h^{-1}(u(x_j) - N^y u(x_{j-1})) \text{ für } j > 0, \end{cases} \\ T_h^z(u,v)(x_j) &= \begin{cases} v(0) - \varphi(h, z_0) \text{ für } j = 0, \\ v(x_j) - N^z(v(x_{j-1}), u(x_{j-1})) \text{ für } j > 0. \end{cases} \end{aligned}$$

Zudem definieren wir den Operator $\tilde{T}_h$, der bis auf

$$\tilde{T}_h^y u(x_0) = P_0 T_h^y u(x_0) + Q_0 \frac{T_h^y u(x_0)}{h}$$

mit T_h übereinstimmt.

Satz (über die Stabilität) 3.18
Angenommen für die Index-2 DAE (3.5) *gelte* (DAE1)-(DAE3). *Für das allgemeine lineare Verfahren setzen wir* (GLM1)-(GLM4) *voraus.*
Dann ist der Operator T_h für den Spezialfall $M(\infty) = 0$ auf U_h für hinreichend kleine h stabil. Tatsächlich gilt auf U_h eine Stabilitätsungleichung der Form

$$|(\hat{u}, \hat{v}) - (\bar{u}, \bar{v})| \le \begin{pmatrix} C^y & 0 \\ D^y & C^z \end{pmatrix} |T_h(\hat{u}, \hat{v}) - T_h(\bar{u}, \bar{v})| \tag{3.41}$$

mit positiven Konstanten C^y, D^y und C^z. Weiter gilt:
(i) *Im Fall $0 \le \rho(M(\infty)) < 1$ erhalten wir im Allgemeinen nur die Stabilität des Operators $\tilde{T}_h$ auf U_δ für hinreichend kleine h und δ_i, das heißt es gilt eine Stabilitätsungleichung der Form* (3.41) *auf U_δ mit T_h ersetzt durch $\tilde{T}_h$. Dies ist durch die z-Komponente bedingt. Tatsächlich gilt für die y-Komponente weiterhin*

$$\|\hat{u} - \bar{u}\|_\infty \le C^y \|T_h^y \hat{u} - T_h^y \bar{u}\|_\infty \tag{3.42}$$

und auf U_h sogar

$$\|\hat{u} - \bar{u}\|_\infty \le C^y \Big(\|P(\tilde{T}_h^y \hat{u} - \tilde{T}_h^y \bar{u})\|_\infty + h \|Q(\tilde{T}_h^y \hat{u} - \tilde{T}_h^y \bar{u})\|_\infty \Big).$$

(ii) *Im Fall* $\rho(M(\infty)) = 1$ *ist für die y-Komponente eine Abschätzung der Form* (3.42) *auf* U_h *für hinreichend kleine* h *weiterhin möglich.*
Für die z-Komponente gilt jedoch:

$$\begin{aligned}\|\hat{v} - \bar{v}\| \quad \leq \quad & D^y\Big(\|\frac{T_h^y\hat{u} - T_h^y\bar{u}}{h}\|_\infty + \|Q\frac{T_h^y\hat{u} - T_h^y\bar{u}}{h^2}\|_\infty\Big) \\ & + C^z\|\frac{T_h^z(\hat{u},\hat{v}) - T_h^z(\bar{u},\bar{v})}{h}\|_\infty.\end{aligned}$$

□

3.2.3 Konvergenz und Konsistenz

Wir untersuchen in diesem Abschnitt die Konvergenz allgemeiner linearer Verfahren angewendet auf Index-2 DAEs in Hessenberg Form. Dabei betrachten wir die Variablen y und z wieder weitgehend getrennt. Aufgrund des letzten Abschnitts über die Stabilität beschränken wir uns auf den Fall $\rho(M(\infty)) < 1$, dass heißt wir betrachten nur stabile allgemeine lineare Verfahren im Sinne von Definition 3.17. Zudem geben wir eine Darstellung des lokalen Fehlers für allgemeine lineare Verfahren, welche für gewöhnliche Differentialgleichungen die Stufenordnung q und die Ordnung $p \geq q$ besitzen. Das Phänomen der Ordnungsreduktion, welches von steifen Differentialgleichungen her bekannt ist, ist auch bei Differential-Algebraischen Gleichungen erkennbar: Die Konvergenzordnung ist von der Größe der Stufenordnung. Wir ziehen daraus die entsprechenden Konsequenzen.

Wir betrachten wieder eine Index-2 DAE der Form (3.5), wobei die Voraussetzungen (DAE1)-(DAE3) gelten. Tatsächlich gehen wir in diesem Abschnitt von folgender Verschärfung aus:

Die Funktion f ist r-mal und g sogar $(r+1)$-mal stetig differenzierbar. (DAE4)

Dabei ist $r \geq 1$ die Konsistenzordnung der y-Komponente des allgemeinen linearen Verfahrens angewendet auf Index-2 DAEs in Hessenberg Form, welche wir im Anschluss an den Konvergenzsatz 3.25 definieren.

Ein allgemeines lineares Verfahren für diese DAE lautet:

$$\begin{aligned} y^{[n+1]} &= (V \otimes I_N) y^{[n]} + h(B \otimes I_N) f(Y, Z), \\ z^{[n+1]} &= (V \otimes I_l) z^{[n]} + h(B \otimes I_l) Z', \\ \\ Y &= (U \otimes I_N) y^{[n]} + h(A \otimes I_N) f(Y, Z), \\ Z &= (U \otimes I_l) z^{[n]} + h(A \otimes I_l) Z', \\ 0 &= g(Y). \end{aligned} \tag{3.43}$$

Dabei sind die Startwerte der Iteration durch eine Startprozedur φ gegeben:

$$\begin{pmatrix} y^{[0]} \\ z^{[0]} \end{pmatrix} = \begin{pmatrix} \varphi^y(h, y_0, z_0) \\ \varphi^z(h, y_0, z_0) \end{pmatrix} = \varphi(h, y_0, z_0).$$

Für das allgemeine lineare Verfahren sei (GLM1)-(GLM4) vorausgesetzt.

Was genau sind der lokale Fehler, die Konsistenz und die Konvergenz eines allgemeinen linearen Verfahrens angewendet auf eine Index-2 DAE (3.5)? Während eine

Definition des lokalen und globalen Fehlers und somit der Konvergenz auf natürliche Weise gegeben ist, wird die geeignete Definition der Konsistenz erst später nach einigen Überlegungen erfolgen können.

Definition 3.19 *Bezeichne $y_c(x,h), z_c(x,h)$ die correct value Funktionen. Dann definieren wir den lokalen Fehler der y-Komponente eines allgemeinen linearen Verfahrens für Index-2 DAEs in Hessenberg Form* (3.43) *durch*

$$\begin{aligned} \delta_0^y &= \varphi^y(h, y_0, z_0) - y_c(x_0, h), \\ \delta_{n+1}^y &= V y_c(x_n, h) + hBf(Y(h, y_c(x_n,h)), Z(h, y_c(x_n,h))) - y_c(x_{n+1}, h), \end{aligned}$$

wobei wir hier die Lösbarkeit des Gleichungssytems der Stufenwerte für hinreichend kleine h garantieren können (vgl. Bemerkung 3.8 (ii)).
Entsprechend definieren wir den lokalen Fehler der z-Komponente durch

$$\begin{aligned} \delta_0^z &= \varphi^z(h, y_0, z_0) - z_c(x_0, h), \\ \delta_{n+1}^z &= M(\infty) z_c(x_n, h) + BA^{-1} Z(h, y_c(x_n,h)) - z_c(x_{n+1}, h). \end{aligned}$$

Die Konvergenz und Konvergenzordnung ist wie gewöhnlich über den globalen Fehler definiert:

Definition 3.20 *Ein allgemeines lineares Verfahren für Index-2 DAEs in Hessenberg Form* (3.43) *ist konvergent in y, falls für den globalen Fehler*

$$\Delta_n := y^{[n]} - y_c(x_n, h)$$

folgende Darstellung gilt:

$$\Delta_n = o(1)$$

mit $x = x_n = nh \leq x_e$. Ein solches allgemeines lineares Verfahren ist konvergent von der Ordnung p in y, falls für den globalen Fehler sogar gilt:

$$\Delta_n = \mathcal{O}(h^p).$$

Entsprechend ist die Konvergenz(ordnung) in der z-Komponente definiert.
Wir nennen ein allgemeines lineares Verfahren für Index-2 DAEs in Hessenberg Form konvergent (von der Ordnung p), falls es sowohl in y als auch in z konvergent (von der Ordnung p) ist.

Konvergenz der y-Komponente
In einem ersten Schritt erhalten wir Konvergenzresultate mit den Abschätzungen aus dem Stabilitätssatz 3.18:

Korollar 3.21
Angenommen die Voraussetzungen von Satz 3.18 gelten. Zudem sei $\rho(M(\infty)) < 1$.
(i) *Erfüllt der lokale Fehler für* $0 \le nh \le konst.$ *die "minimalen" Voraussetzungen*

$$\delta_n^z = o(1), \qquad Q_0\delta_0^y = ho(1)$$

und gilt für den lokalen Fehler in y die Darstellung

$$\delta_0^y = \mathcal{O}(h^r), \qquad \delta_{n+1}^y = \mathcal{O}(h^{r+1}) \tag{3.44}$$

für ein $r \ge 1$*, so ist das allgemeine lineare Verfahren für hinreichend kleine o-Terme in der y-Komponente konvergent von der Ordnung r.*

(ii) *Erfüllt der lokale Fehler für* $0 \le nh \le konst.$ *die "minimalen" Voraussetzungen*

$$\delta_n^z = \mathcal{O}(h), \qquad Q_0\delta_0^y = \mathcal{O}(h^2),$$

so brauchen wir nur

$$\delta_n^y = \mathcal{O}(h^r), \qquad P(x_{n+1})\delta_{n+1}^y = \mathcal{O}(h^{r+1}) \tag{3.45}$$

voraussetzen, um in der y-Komponente Konvergenz von der Ordnung r zu erhalten.

Bemerkungen:
(i) *Die minimalen Voraussetzungen in (i) und die Darstellungen in* (3.44) *mit* $r = 1$ *garantieren, dass die Lösung von* $T_h(u, v) = 0$ *in der Umgebung* U_δ *liegt. Entsprechend sichern im Fall (ii) die minimalen Voraussetzungen und wiederum die Darstellungen in* (3.45) *mit* $r = 1$*, dass diese Lösung in* U_h *liegt. In beiden Fällen kann durch Anwendung von Satz 3.18 mit den Voraussetzungen in* (3.44) *bzw.* (3.45) *die Konvergenz der Ordnung r geschlossen werden.*

(ii) *Bei Runge-Kutta Verfahren sind alle Voraussetzungen an* δ_0^y *und* δ_0^z *per Definition erfüllt, da von exakten konsistenten Anfangswerten gestartet wird. Eine Startprozedur ist nicht nötig. Die Voraussetzungen an* δ_{n+1}^z *dagegen sind weiterhin nötig (vgl. dazu [HLR89], wo diese Voraussetzungen nicht die nötige Beachtung finden; erst in der neueren Formulierung von Theorem 4.2 S.33 in dem Buch [HW] auf Seite 493 wird eine Bedingung an die z-Komponente formuliert).*

(iii) *Für* $r \ge 2$ *ist die Voraussetzung an* $Q_0\delta_0^y$ *jeweils durch die Darstellungen* (3.44) *bzw. in (ii) durch* (3.45) *implizit gegeben.*

(iv) *Ist die Matrix* $M(\infty)$ *sogar nilpotent mit Index* k_0*, so brauchen wir in* (3.44) *und* (3.45) *jeweils für* δ_0^y *nur*

$$P_0\delta_0^y = \mathcal{O}(h^r), \qquad Q_0\delta_0^y = \mathcal{O}(h^{r-1})$$

voraussetzen (vgl. Kapitel 4 Lemma 4.1). Wir gewinnen nämlich nach den

Abschätzungen (3.33) *und* (3.34) *aus Lemma 3.13, welches wir im Beweis unten auf die Entwicklung des globalen Fehlers anwenden, nach k_0 Iterationen eine h-Potenz in dem projizierten globalen Fehler $Q_n\Delta y_n$. Wir erhalten somit die Konvergenz der Ordnung r nach k_0 Schritten, während vorher die Ordnung $r-1$ beträgt. Aufgrund der minimalen Voraussetzungen in (i) und (ii) bringt dies nur für $r \geq 3$ einen Vorteil.*

Beweis: Der Beweis besteht in der Anwendung des Stabilitätssatzes 3.18: Wir bezeichnen dazu mit $\bar{y}_h, \bar{z}_h$ die Lösung des numerischen Modells $T_h(u,v) = 0$, das heißt es gilt:

$$(\bar{y}_h, \bar{z}_h)(x_n) = (y^{[n]}, z^{[n]}).$$

Zudem sei an die Bezeichnung (y_h, z_h) der Restriktionen der *correct value* Funktionen auf das Gitter Ω_h erinnert. Da die Mengen U_δ und U_h Umgebungen von (y_h, z_h) sind, gilt:

$$(y_h, z_h) \in U_\delta \cap U_h.$$

Wir zeigen, dass unter den Voraussetzungen von (i) auch die numerische Lösung in der Umgebung U_δ liegt:

$$(\bar{y}_h, \bar{z}_h) \in U_\delta.$$

Satz 3.18 (i) liefert in diesem Fall für hinreichend kleine h und δ_i die Konvergenz der Ordnung r in der y-Komponente:

$$\|\bar{y}_h - y_h\|_\infty \leq C^y \underbrace{\|T_h^y y_h\|_\infty}_{=\mathcal{O}(h^r)}.$$

Dabei gilt:

$$hT_h^y y_h(x_{n+1}) = \delta_{n+1}^y, \qquad T_h^y y_h(x_0) = \delta_0^y.$$

Unter den Voraussetzungen von (ii) finden wir sogar

$$(\bar{y}_h, \bar{z}_h) \in U_h.$$

Eine Anwendung von Satz 3.18 (i) liefert nun

$$\|\bar{y}_h - y_h\|_\infty \leq C^y \Big(\|P\tilde{T}_h^y y_h\|_\infty + h\|Q\tilde{T}_h^y y_h\|_\infty \Big).$$

Aufgrund der Definition des Operators $\tilde{T}_h$ und den Voraussetzungen an den lokalen Fehler gibt dies genau die Konvergenz der entsprechenden Ordnung:

$$\|\bar{y}_h - y_h\|_\infty \leq C^y \mathcal{O}(h^r).$$

Zu (i): Wir zeigen für $C_0 > 0$ induktiv

$$\begin{aligned} \|y^{[n]} - y_c(x_n, h)\| &\leq C_0 h, \\ \|Q_n(y^{[n]} - y_c(x_n, h))\| &\leq \delta_1 h, \\ \|z^{[n]} - z_c(x_n, h)\| &\leq \delta_2. \end{aligned} \tag{3.46}$$

Wir müssen diese Abschätzungen für hinreichend kleine δ_i und h garantieren, damit wir Satz 3.18 (i) auch wirklich anwenden können.
Für $r \geq 2$ ist es nicht weiter schwer (3.46) zu zeigen. Alle nötigen Abschätzungen können ohne genauere Überlegungen angewendet werden. Im Fall $r = 1$ ist dies nicht möglich. Das Ziel ist ja (3.46) gleichmäßig für n zu zeigen. Insbesondere müssen C_0 und die δ_i unabhängig von n sein. In diesem Fall zahlt sich das genaue Vorgehen der letzten beiden Abschnitte aus, wo wir die Abhängigkeiten entsprechender Konstanten genau untersucht haben (vgl. die Zusätze von Satz 3.7 und Korollar 3.10 und die daraus resultierende Bemerkung 3.11).

Induktionsanfang: Für beliebig vorgegebene δ_i können wir den Induktionsanfang für hinreichend kleine h und $o(1)$ mit der Voraussetzung $Q_0\delta_0^y = ho(1)$ garantieren:

$$\begin{array}{rcl} y^{[0]} - y_c(x_0,h) & = & \varphi^y(h,y_0,z_0) - y_c(x_0,h) \;=\; \delta_0^y \;=\; \mathcal{O}(h), \\ z^{[0]} - z_c(x_0,h) & = & \varphi^z(h,y_0,z_0) - z_c(x_0,h) \;=\; \delta_0^z \;=\; o(1). \end{array}$$

Hier müssen wir C_0 aufgrund der Konstanten, die implizit durch den $\mathcal{O}$-Term gegeben ist, entsprechend groß wählen.
Wir gehen nun davon aus, dass (3.46) für alle $j = 0, \ldots, n$ gilt. Gilt (3.46) dann auch für $j = n+1$? Die Antwort ist natürlich: Ja! Um dies zu zeigen, erinnern wir zunächst an die Iterationsoperatoren

$$\begin{array}{rcl} y^{[n+1]} & = & N^y y^{[n]}, \\ z^{[n+1]} & = & N^z(y^{[n]}, z^{[n]}), \end{array}$$

die definiert sind durch

$$\begin{array}{rcl} N^y y^{[n]} & = & V y^{[n]} + hBf(Y(h,y^{[n]}), Z(h,y^{[n]})), \\ N^z(y^{[n]}, z^{[n]}) & = & M(\infty) z^{[n]} + BA^{-1} Z(h, y^{[n]}). \end{array}$$

Aus der Definition des lokalen Fehlers folgt unmittelbar die Darstellung

$$\begin{array}{rcl} \delta_{n+1}^y & = & N^y y_c(x_n,h) - y_c(x_{n+1},h), \\ \delta_{n+1}^z & = & N^z(y_c(x_n,h), z_c(x_n,h)) - z_c(x_{n+1},h). \end{array}$$

Induktionsschluss: Mit der nahrhaften Null und dem Iterationsoperator der y-Komponente finden wir für den globalen Fehler die Darstellung

$$\begin{array}{rcll} y^{[n+1]} - y_c(x_{n+1},h) & = & y^{[n+1]} - N^y y_c(x_n,h) + N^y y_c(x_n,h) - y_c(x_{n+1},h) & \\ & = & V(y^{[n]} - y_c(x_n,h)) + h\Delta\Phi_n + \delta_{n+1}^y. & (3.47) \end{array}$$

Hier haben wir die abkürzende Schreibweise aus Abschnitt 3.2.2 benutzt:

$$\Delta\Phi_j := \Phi(h, y^{[j]}) - \Phi(h, y_c(x_j,h)).$$

Die Verfahrensfunktion ist dabei gegeben durch:

$$\Phi(h,u) := Bf(Y(h,u), Z(h,u)).$$

Analog zum Beweis von Lemma 3.12 erhalten wir für $j = 0, \ldots, n$ und $\delta = \max \delta_1$ die Darstellung (ohne das Kronecker Produkt)

$$\Delta\Phi_j = BA^{-1}UQ_j\Delta y_j + \mathcal{O}(\delta\|Q_j\Delta y_j\|) + \mathcal{O}(h\|\Delta y_j\|), \tag{3.48}$$

wobei diesmal alle $\mathcal{O}$-Terme unabhängig von C_0 sind. In Abschnitt 3.2.2 besaß C_0 als Konstante in der Definition von U_δ genau die gleiche Funktion wie hier. Die Unabhängigkeit ist garantiert, da wir bei der Linearisierung 3.29 die Abhängigkeit vermeiden können (vgl. Bemerkung 3.11). Das Einsetzen von (3.48) in die Darstellung des globalen Fehlers (3.47) ergibt folgende Fehlerentwicklung:

$$\Delta y_{j+1} = VP_j\Delta y_j + M(\infty)Q_j\Delta y_j + \mathcal{O}(\delta\|Q_j\Delta y_j\|) + \mathcal{O}(h\|\Delta y_j\|) + \delta^y_{n+1}. \tag{3.49}$$

Dabei sind weiterhin alle $\mathcal{O}$-Terme unabhängig von C_0. Eine Anwendung von Lemma 3.13 mit $D_i = \|T^y_h\|_\infty$ liefert eine von n und C_0 unabhängige Konstante $K > 0$ mit

$$\|\Delta y_{n+1}\| \leq Kh.$$

Ohne Einschränkung sei C_0 in (3.46) größer gewählt worden als K. Zusätzlich finden wir mit der Abschätzung (3.32) aus Lemma 3.13 ohne Einschränkung für dasselbe K die Ungleichung

$$\|Q_{n+1}\Delta y_{n+1}\| \leq Kho(1).$$

Hier ist $o(1)$ der o-Term aus der Darstellung $Q_0\delta^y_0 = ho(1)$. Ist dieser o-Term hinreichend klein, so kann

$$Ko(1) < \delta_1$$

garantiert werden.

Mit der nahrhaften Null und dem Iterationsoperator der z-Komponente finden wir

$$\begin{aligned} & z^{[n+1]} - z_c(x_{n+1}, h) \\ &= z^{[n+1]} - N^z(y_c(x_n,h), z_c(x_n,h)) + N^z(y_c(x_n,h), z_c(x_n,h)) - z_c(x_{n+1},h) \\ &= M(\infty)(z^{[n]} - z_c(x_n,h)) + BA^{-1}(Z(h, y^{[n]}) - Z(h, y_c(x_n,h))) + \delta^z_{n+1}. \end{aligned}$$

Mit Korollar 3.10 können wir die Differenz der Stufenwerte abschätzen:

$$\|BA^{-1}(Z(h,y^{[n]}) - Z(h,y_c(x_n,h)))\| \leq \frac{K_1}{h}\|Q_n(y^{[n]} - y_c(x_n,h))\| + K_2\|y^{[n]} - y_c(x_n,h)\|.$$

Dabei ist die Konstante K_1 unabhängig von C_0 und den δ_1. Somit folgt

$$\|z^{[n+1]} - z_c(x_{n+1},h)\|_* \leq \|M(\infty)\|_*\|z^{[n]} - z_c(x_n,h)\|_* + K_1\delta_1 + K_2hC_0 + o(1).$$

Die Norm sei so gewählt, dass gilt:

$$\|M(\infty)\|_* < 1$$

(vgl. [SB] Satz 6.9.2). Dies ist aufgrund der Voraussetzung $\rho(M(\infty)) < 1$ möglich. Für hinreichend kleine h,δ_1 und $o(1)$ kann daher

$$\|z^{[n+1]} - z_c(x_{n+1}, h)\| \leq \delta_2$$

garantiert werden.

zu (ii): Der Beweis ist analog: Wir ersetzen in (3.46) die δ_i durch $C_i h$. Der Induktionsanfang ist wieder durch die Voraussetzungen gegeben. Die Konstanten C_i müssen nur entsprechend groß gewählt werden. Im Induktionsschluss folgt

$$\|\Delta y_{n+1}\| \leq C_0 h$$

in genau derselben Weise wie in (i). Für die projizierte Komponente folgt nun mit der Abschätzung (3.32) aus Lemma 3.13

$$\|Q_{n+1}\Delta y_{n+1}\| \leq Kh^2.$$

Die Konstante K ist dabei unabhängig von n und den C_i. Dabei gelte ohne Einschränkung $K \leq C_1$.
Für die z-Komponente gilt nun

$$\|z^{[n+1]} - z_c(x_{n+1}, h)\|_* \leq \|M(\infty)\|_* \|z^{[n]} - z_c(x_n, h)\|_* + K_1 C_1 h + K_2 h C_0 + \mathcal{O}(h).$$

Für hinreichend kleine h gilt somit

$$\|z^{[n+1]} - z_c(x_{n+1}, h)\| \leq C_2 h,$$

falls C_2 hinreichend groß gewählt worden ist.

□

Besitzt der lokale Fehler allgemeiner linearer Verfahren, welche für gewöhnliche Differentialgleichungen konzipiert wurden, jetzt aber auf Index-2 DAEs in Hessenberg Form angewendet werden, die im Korollar geforderte Form? Wir werden im folgenden Satz sehen, dass dies nicht immer der Fall ist. Zudem ist die Ordnung des lokalen Fehlers von der Größe der Stufenordnung!

Wir erinnern an die Blockdiagonalisierung der stabilen Matrix V aus Kapitel 2 im Anschluss an die Definition 2.13:

$$T^{-1}VT = \text{diag}\{1, \ldots, 1, \zeta_2, \ldots, \zeta_l, J_s\} \tag{3.50}$$

mit $\zeta_i \neq 1$ und $|\zeta_i| = 1$. Das Spektrum der Matrix J_s liegt dabei innerhalb des Einheitskreises. Zudem hatten wir dort die Projektion

$$E := T\text{diag}\{I, 0, \ldots, 0\}T^{-1} \tag{3.51}$$

definiert.

Satz (über den lokalen Fehler) 3.22
Für die Index-2 DAE (3.5) *setzen wir* (DAE1)-(DAE3) *bzw. die Verschärfung* (DAE4) *für* $q \geq 1$ *voraus. Angenommen ein allgemeines lineares Verfahren mit invertierbarer Koeffizientenmatrix A besitzt für gewöhnliche Differentialgleichungen die Konsistenzordnung* p *und die Stufenordnung* q *mit* $p \geq q \geq 1$. *Dann gelten für den lokalen Fehler der y-Komponente des Verfahrens angewendet auf die DAE* (3.5) *die Darstellungen*

- $\boldsymbol{q+1 < p}$*:* $\quad \delta_{n+1}^y = \mathcal{O}(h^{q+1}), \qquad P(x_{n+1})\delta_{n+1}^y = \mathcal{O}(h^{q+2}).$
- $\boldsymbol{q+1 = p}$*:* $\quad \delta_{n+1}^y = \mathcal{O}(h^{q+1}), \qquad EP(x_{n+1})\delta_{n+1}^y = \mathcal{O}(h^{q+2}).$
- $\boldsymbol{q = p}$*:* $\quad \delta_{n+1}^y = \mathcal{O}(h^{p}), \qquad E\delta_{n+1}^y = \mathcal{O}(h^{p+1}).$

Für den lokalen Fehler der z-Komponente gilt immer

$$\delta_{n+1}^z = \mathcal{O}(h^q).$$

Zusatz: Angenommen der lokale Fehler des allgemeinen linearen Verfahrens für gewöhnliche Differentialgleichungen besitze die Ordnung p. Dann gelten für den lokalen Fehler der y-Komponente des Verfahrens angewendet auf die DAE (3.5) *die Darstellungen*

- $\boldsymbol{q+1 = p}$*:* $\quad \delta_{n+1}^y = \mathcal{O}(h^{q+1}), \qquad P(x_{n+1})\delta_{n+1}^y = \mathcal{O}(h^{q+2}).$
- $\boldsymbol{q = p}$*:* $\quad \delta_{n+1}^y = \mathcal{O}(h^{p+1}).$

Bemerkungen:
(i) *Die Darstellungen des lokalen Fehlers sind unabhängig von der Matrix* $M(\infty)$.

(ii) *Insgesamt ist die Ordnung des lokalen Fehlers von der Größe der Stufenordnung!*

Beweis: Die Idee des Beweises liegt darin, die Lösung $y(x), z(x)$ der Index-2 DAE (3.5) als Lösung gewisser gewöhnlicher Differentialgleichungen zu interpretieren, um die Voraussetzungen an das allgemeine lineare Verfahren ausnutzen zu können.

Die Lösungskomponente $y(x)$ ist Lösung der gewöhnlichen Differentialgleichung

$$u' = f(u, z(x)).$$

Hier wurde die exakte Lösung $z(x)$ eingesetzt. Die Definition der Konsistenz 2.13 liefert nun

$$\begin{aligned} \delta_{n+1}^u := Vy_c(x_n,h) + hBf(U(h,y_c(x_n,h)), z(x_n+ch)) - y_c(x_{n+1},h) &= \mathcal{O}(h^p), \\ E(\delta_1^u + \ldots + \delta_{n+1}^u) &= \mathcal{O}(h^p). \end{aligned}$$

Im Fall des Zusatzes gilt sogar

$$\delta_{n+1}^u = \mathcal{O}(h^{p+1}).$$

Es sei an die Schreibweise

$$h(x_n + ch) = \begin{bmatrix} h(x_n + c_1 h) \\ \vdots \\ h(x_n + c_s h) \end{bmatrix}$$

erinnert. Betrachten wir den lokalen Fehler der y-Komponente des allgemeinen linearen Verfahrens angewendet auf die Index-2 DAE

$$\delta_{n+1}^y = V y_c(x_n, h) + hBf(Y(h, y_c(x_n, h)), Z(h, y_c(x_n, h))) - y_c(x_{n+1}, h)$$

so ähneln sich die Darstellungen der lokalen Fehler oben sehr. Tatsächlich gilt

$$\delta_{n+1}^y = \delta_{n+1}^u + hB\Big(f(Y(h, y_c(x_n, h)), Z(h, y_c(x_n, h))) - f(U(h, y_c(x_n, h)), z(x_n + ch))\Big).$$

Um jetzt die Differenz in den Griff zu bekommen, müssen wir die Differenz der Argumente abschätzen: Da die Stufenordnung q ist, gilt per Definition

$$\mathcal{O}(h^{q+1}) = y(x_n + ch) - U y_c(x_n, h) - hAf(y(x_n + ch), z(x_n + ch)).$$

Aus Lemma 2.3 folgt damit

$$U(h, y_c(x_n, h)) - y(x_n + ch) = \mathcal{O}(h^{q+1}).$$

Offenbar gilt zudem

$$0 = g(y(x_n + ch)).$$

Wir wenden nun Satz 3.7 auf

$$\eta = U y_c(x_n, h), \qquad \xi = U z_c(x_n, h)$$

an und erhalten für die Lösung $Y(h, y_c(x_n, h)), Z(h, y_c(x_n, h))$ die Darstellungen

$$\begin{aligned} Y(h, y_c(x_n, h)) - \mathbb{1} \otimes y(x) &= \mathcal{O}(h), \\ Z(h, y_c(x_n, h)) - \mathbb{1} \otimes z(x) &= \mathcal{O}(h) \end{aligned}$$

(vgl. Bemerkung 3.8 (ii)). Insbesondere erhalten wir für hinreichend kleine h mit Korollar 3.9 die Darstellungen

$$\begin{aligned} Y(h, y_c(x_n, h)) - y(x_n + ch) &= \mathcal{O}(h^{q+1}), \\ Z(h, y_c(x_n, h)) - z(x_n + ch) &= \mathcal{O}(h^q). \end{aligned}$$

Mit diesen Überlegungen können wir die Argumente der Differenz in der Darstellung des lokalen Fehlers oben abschätzen. Eine Taylorentwicklung liefert dann

$$\begin{aligned}\delta^y_{n+1} &= \delta^u_{n+1} \\ &\quad + hB\frac{\partial f}{\partial Z}(\mathbb{1}\otimes y(x_n), \mathbb{1}\otimes z(x_n))(Z(h, y_c(x_n,h)) - z(x_n+ch)) + \mathcal{O}(h^{q+2}).\end{aligned}$$

Nutzen wir die tatsächliche Struktur aus, indem wir das Kronecker Produkt verwenden, so erhalten wir

$$\delta^y_{n+1} = \delta^u_{n+1} + h(B\otimes f^n_z)(Z(h, y_c(x_n,h)) - z(x_n+ch)) + \mathcal{O}(h^{q+2}).$$

Hier ist f^n_z wieder eine Abkürzung der partiellen Ableitung von f ausgewertet in $(y(x_n), z(x_n))$. Eine letzte Taylorentwicklung liefert die Darstellung

$$\delta^y_{n+1} = \delta^u_{n+1} + h(B\otimes f^{n+1}_z)(\underbrace{Z(h, y_c(x_n,h)) - z(x_n+ch)}_{\mathcal{O}(h^q)}) + \mathcal{O}(h^{q+2}).$$

Mit $P(x_{n+1})f^{n+1}_z = 0$ und den Eigenschaften von δ^u_{n+1} folgen alle Aussagen des Satzes bezüglich der y-Komponente unmittelbar.

Um den lokalen Fehler der z-Komponente zu untersuchen, erinnern wir daran, dass auch $z(x)$ Lösung einer gewöhnlichen Differentialgleichung

$$v' = \varphi(x, v)$$

ist. Diese Gleichung erhält man durch zweimaliges Differenzieren der algebraischen Gleichung, auflösen nach z' und einsetzen der exakten y-Komponente $y(x)$. Wir verwenden zur Unterscheidung die Variable v. Mit der Stufenordnung q und der Konsistenzordnung p erhalten wir

$$\begin{aligned}z(x_n+ch) &= Uz_c(x_n,h) + hA\varphi(x_n+ch, z(x_n+ch)) + \mathcal{O}(h^{q+1}),\\ z_c(x_{n+1},h) &= Vz_c(x_n,h) + hB\varphi(x_n+ch, V(h, z_c(x_n,h))) + \mathcal{O}(h^p).\end{aligned}$$

Man beachte, dass wir hier nicht die Matrix E und die exakte Definition der Konsistenzordnung benutzt haben (vgl. Definition 2.13), da wir nur an Ordnung p interessiert sind. Eine weitere einfache Folgerung aus der Stufenordnung q ist

$$z(x_n+ch) - V(h, z_c(x_n,h)) = \mathcal{O}(h^{q+1})$$

(vgl. Lemma 2.3). Lösen wir die erste Gleichung oben nach der Funktion $h\varphi$ auf und investieren diese Folgerung

$$h\varphi(x_n+ch, Z(h, z_c(x_n,h))) = A^{-1}(z(x_n+ch) - Uz_c(x_n,h)) + \mathcal{O}(h^{q+1}),$$

so finden wir

$$z_c(x_{n+1},h) = M(\infty)z_c(x_n,h) + BA^{-1}z(x_n+ch) + \mathcal{O}(h^{\min\{p,q+1\}}).$$

Der lokale Fehler der z Komponente ist gegeben durch

$$z_c(x_{n+1}, h) = M(\infty) z_c(x_n, h) + BA^{-1} Z(h, y_c(x_n, h)).$$

Der Aussage des Satzes über den lokalen Fehler der z-Komponente folgt nun aus der bereits oben hergeleiteten Darstellung

$$Z(h, y_c(x_n, h)) - z(x_n + ch) = \mathcal{O}(h^q).$$

□

Wie wir bereits in der Bemerkung im Anschluss an den Satz festgestellt haben, ist die Ordnung des lokalen Fehlers sowohl in der y- als auch in der z-Komponente von der Größenordnung der Stufenordnung. Dies bedeutet, dass vor allem Verfahren mit hoher Stufenordnung zum Beispiel mit $p = q + 1$ oder $p = q$ zum Lösen von Index-2 DAEs in Hessenberg Form geeignet scheinen. Solche Verfahren sind auch zum Lösen steifer Differentialgleichungen aufgrund des Phänomens der Ordnungsreduktion entwickelt worden (vgl. zum Beispiel [Huang05, Wright02]).
Über den lokalen Fehler δ_0^y und δ_0^z zu Beginn der Iteration kann der Satz natürlich keine Aussage machen, da wir die Startprozedur nicht festgelegt haben. Wir werden bei der Implementierung darauf zu achten haben, dass die entsprechende Ordnung gewährleistet werden kann (vgl. Kapitel 4).
Korollar 3.21 liefert in den meisten Fällen entsprechende Konvergenzaussagen der allgemeinen linearen Verfahren. Nur das Auftreten der Matrix E ist in den Voraussetzungen des Korollars, vorallem in (ii), nicht berücksichtigt. Können diese Voraussetzungen dahingehend reduziert werden? Die Antwort ist: Ja! Um dies auch wirklich einzusehen, beweisen wir eine Verallgemeinerung von Lemma 3.13:

Konvergenzlemma 3.23 *Gegeben sei eine Iteration in $\mathbb{R}^m$ der Form*

$$\xi_{j+1} = V P_j \xi_j + M Q_j \xi_j + \mathcal{O}(h\|\xi_j\|) + h r_{j+1}$$

für $0 \leq jh \leq konst.$ Zudem gelte (für solche j):

- *Die Matrizen M und V sind stabil; es gilt sogar $\rho(M) < 1$ und $1 \in \sigma(V)$.*
- *Die Projektionen P_j, $Q_j = I_m - P_j$ auf $\mathbb{R}^m$ kommutieren mit V und M und es gilt*
$$P_{j+1} = P_j + \mathcal{O}(h).$$
- *Es existiert eine stetig differenzierbare Funktion r mit*
$$r(x_j) = r_{j+1}.$$
- *Es existieren nicht negative Konstanten D und D' mit*
$$\|r_j\| \leq D, \qquad \|r'\| \leq D'.$$

Dann existieren $h_0 > 0$ und eine Konstante $C > 0$, so dass die folgende Abschätzung

$$\|\xi_n\| \leq C\Big(\|\xi_0\| + hD + h\|\sum_{k=1}^{n} EP_k r_k\| + hD'\Big) \tag{3.52}$$

für alle $h \leq h_0$ und $nh \leq$ konst. gilt.
Zusatz: Setzen wir $hr_0 := \xi_0$, so existiert ein $K > 0$ mit

$$\begin{aligned}
\max_{0\leq j\leq n} \|\sum_{i=0}^{j} V^{j-i} hP_i r_i\| &\leq K \max_{0\leq j\leq n} \|\xi_j\|, \\
\max_{0\leq j\leq n} \|\sum_{i=0}^{j} M^{j-i} hQ_i r_i\| &\leq K \max_{0\leq j\leq n} \|\xi_j\|.
\end{aligned}$$

Beweis: Wir hatten bereits in (3.51) die Projektion E auf den Eigenraum zum Eigenwert 1 mit Hilfe der Transformation (3.50) eingeführt:

$$E := T\mathrm{diag}\{I, 0, \ldots, 0\}T^{-1}.$$

Entsprechend definieren wir die Projektionen:

$$\begin{aligned}
E_2 &= T\mathrm{diag}\{0, I, 0, \ldots, 0\}T^{-1}, \\
&\vdots \\
E_l &= T\mathrm{diag}\{0, \ldots, 0, I, 0\}T^{-1}.
\end{aligned}$$

Wir definieren weiter die zu J_s äquivalente Matrix

$$V_s := T\mathrm{diag}\{0, \ldots, 0, J_s\}T^{-1}.$$

Insbesondere liegt auch das Spektrum von V_s innerhalb des Einheitskreises. Die Eigenwerte von V_s sind genau die Eigenwerte von V die nicht auf dem Rande des Einheitskreises liegen. Bezeichne E_s die Projektion auf die direkte Summe der zugehörigen verallgemeinerten Eigenräume parallel zu der direkten Summe der restlichen verallgemeinerten Eigenräume.
Mit diesen Definitionen erhalten wir die Zerlegung

$$V = E + \zeta_2 E_2 + \ldots + \zeta_l E_l + V_s. \tag{3.53}$$

Wir gehen im Folgenden sehr ähnlich zum Beweis des Stabilitätslemmas 3.13 vor, vermeiden aber zunächst die Anwendung der Dreiecksungleichung. Die Idee besteht darin, die Iteration durch Anwendung verschiedener Projektionen zu zerlegen und dann eine Iteration in einem entsprechend höher dimensionalen Raum zu betrachten. Für diese Iteration lässt sich eine Stabilitätsungleichung zeigen. Schreiben wir anschließend ξ_{j+1} als Summe unter diesen Projektionen, so erhalten

wir die Stabilitätsungleichung des Lemmas.

Wir finden wie im Beweis des Stabilitätslemmas 3.13 die Gleichungen

$$\begin{aligned} P_{j+1}\xi_{j+1} &= VP_j\xi_j + \mathcal{O}(h\|\xi_j\|) + hP_{j+1}r_{j+1}, \\ Q_{j+1}\xi_{j+1} &= MQ_j\xi_j + \mathcal{O}(h\|\xi_j\|) + hQ_{j+1}r_{j+1}. \end{aligned} \tag{3.54}$$

Multiplizieren wir die erste dieser Gleichungen von links mit E, E_s und E_i, so erhalten wir mit der Darstellung (3.53) die Gleichungen

$$\begin{aligned} EP_{j+1}\xi_{j+1} &= EP_j\xi_j + \mathcal{O}(h\|\xi_j\|) + hEP_{j+1}r_{j+1}, \\ E_sP_{j+1}\xi_{j+1} &= V_sE_sP_j\xi_j + \mathcal{O}(h\|\xi_j\|) + hE_sP_{j+1}r_{j+1}, \\ E_iP_{j+1}\xi_{j+1} &= \zeta_iE_iP_j\xi_j + \mathcal{O}(h\|\xi_j\|) + hE_iP_{j+1}r_{j+1} \end{aligned}$$

für $i = 2, \ldots, l$. Wir haben nun die Iteration durch Anwendung verschiedener Projektionen zerlegt und betrachten diese Zerlegung bei entsprechender Definition von $\bar{v}_j$ und $\bar{r}_{j+1}$ als Iteration in einem $(l+2)m$-dimensionalen Raum:

$$\bar{v}_{j+1} = B\bar{v}_j + h\bar{r}_{j+1} = B^{j+1}\bar{v}_0 + h\sum_{k=0}^{j} B^k\bar{r}_{j+1-k}.$$

Dabei ist die Matrix B definiert durch

$$B = \operatorname{diag}\{I, \zeta_2 I, \ldots, \zeta_l I, V_s, M\} + \mathcal{O}(h).$$

Die Stabilität dieser Iteration ist für hinreichend kleine h mit der Stabilität der Matrix B gewährleistet. Insbesondere ist B^n normbeschränkt und zwar gleichmäßig für alle n mit $nh \leq konst.$ Tatsächlich sind wir jedoch daran interessiert, in welchen Komponenten wir den Faktor h vor der Summe investieren müssen, um deren Beschränktheit zu erhalten, und in welchen Komponenten dies nicht nötig ist.
Wir blockdiagonalisieren dazu die Matrix B. Es existiert eine Transformationsmatrix S der Form

$$S = I + \mathcal{O}(h)$$

mit

$$SBS^{-1} = J := \operatorname{diag}\{I + \mathcal{O}(h), \zeta_2 I + \mathcal{O}(h), \ldots, \zeta_l I + \mathcal{O}(h), \operatorname{diag}\{V_s, M\} + \mathcal{O}(h)\}.$$

Wir finden damit

$$\bar{v}_n = B^n\bar{v}_0 + h\sum_{k=0}^{n-1} S^{-1}J^kS\bar{r}_{n-k}. \tag{3.55}$$

Aufgrund der Blockdiagonalstruktur der Matrix J gilt:

$$J^n = \operatorname{diag}\{I + \mathcal{O}(1), \zeta_2^n(I + \mathcal{O}(1)), \ldots, \zeta_l^n(I + \mathcal{O}(1)), (\operatorname{diag}\{V_s, M\} + \mathcal{O}(h))^n\}.$$

Wir definieren nun eine Norm $\|.\|_*$ auf $\mathbb{R}^{(l+2)m}$ wie folgt:

$$\| \begin{pmatrix} v_1 \\ \vdots \\ v_{l+2} \end{pmatrix} \|_* := \sum_{k=1}^{l+2} \|v_k\|$$

mit einer beliebigen Norm auf $\mathbb{R}^m$. Wenden wir diese Norm auf (3.55) an, so finden wir mit der Dreiecksungleichung und der Stabilität der Matrix B die Abschätzung

$$\begin{aligned} \|\xi_n\| &\leq \|\bar{v}_n\|_* \\ &\leq C\Big(\|\xi_0\| + hD + h\|\sum_{k=0}^{n-1} J^k \bar{r}_{n-k}\|_*\Big). \end{aligned} \tag{3.56}$$

Wir untersuchen die Blöcke der Matrix J^k etwas genauer. Der untere rechte Block von J ist

$$\mathrm{diag}\{V_s, M\} + \mathcal{O}(h).$$

Für hinreichend kleine h liegen alle Eigenwerte dieser Matrix im Einheitskreis. Also ist

$$\sum_{k=0}^{n-1} (\mathrm{diag}\{V_s, M\} + \mathcal{O}(h))^k$$

beschränkt. Wir müssen keinen Faktor h investieren, um diese Beschränktheit zu erhalten.
Dies ist natürlich bei dem ersten Block

$$I + \mathcal{O}(1)$$

der Matrix J^k nicht möglich. Daher bleibt die Summe in der Abschätzung des Lemmas erhalten.
Betrachten wir nun noch die Blöcke zu den Eigenwerten auf dem Rand des Einheitskreises, die ungleich 1 sind. Mit der Abelschen partiellen Summation folgt leicht (vgl. [H] S.91):

$$\sum_{k=0}^{n-1} \zeta_i^k E_i P_{n-k} r_{n-k} = \frac{1-\zeta_i^n}{1-\zeta} E_i P_1 r_1 + \sum_{k=1}^{n} \frac{1-\zeta_i^{n-k}}{1-\zeta} E_i (P_{k+1} r_{k+1} - P_k r_k)$$

für $i = 2, \ldots, l$. Daher folgt

$$\|\sum_{k=0}^{n-1} \zeta_i^k E_i P_{n-k} r_{n-k}\| \leq K(D + \sum_{k=1}^{n} \|r_{k+1} - r_k\|)$$

für eine positive Konstante K. Hier haben wir wieder die Voraussetzung $P_{k+1} = P_k + \mathcal{O}(h)$ ausgenutzt. Durch die Beschränktheit der Ableitung r' gewinnen wir einen Faktor h, den wir investieren, um die Summe unabhängig von n

und h abzuschätzen.
Mit diesen Überlegungen folgt die Abschätzung des Lemmas nach einer eventuellen Vergrößerung von C leicht aus (3.56).

Zum Beweis des Zusatzes: Mit $hr_0 = \xi_0$ und der abkürzenden Schreibweise

$$\tilde{S} := \mathrm{diag}\{V, M\}$$

folgt unmittelbar aus der Darstellung von $P_{j+1}\xi_{j+1}$ und $Q_{j+1}\xi_{j+1}$ in (3.54):

$$\begin{aligned}
\begin{pmatrix} P_{j+1}\xi_{j+1} \\ Q_{j+1}\xi_{j+1} \end{pmatrix} &= \begin{pmatrix} V & 0 \\ 0 & M \end{pmatrix} \begin{pmatrix} P_j\xi_j \\ Q_j\xi_j \end{pmatrix} + \mathcal{O}(h\|\xi_j\|) + h \begin{pmatrix} P_{j+1}r_{j+1} \\ Q_{j+1}r_{j+1} \end{pmatrix} \\
&= \tilde{S}^{j+1} \begin{pmatrix} P_0\xi_0 \\ Q_0\xi_0 \end{pmatrix} + \sum_{i=0}^{j} \tilde{S}^{j-i} \left[\mathcal{O}(h\|\xi_i\|) + h \begin{pmatrix} P_{i+1}r_{i+1} \\ Q_{i+1}r_{i+1} \end{pmatrix} \right] \\
&= \sum_{i=0}^{j} \tilde{S}^{j-i}\mathcal{O}(h\|\xi_i\|) + \sum_{i=0}^{j+1} \tilde{S}^{j+1-i} h \begin{pmatrix} P_i r_i \\ Q_i r_i \end{pmatrix}.
\end{aligned}$$

Also gilt nach entsprechender Auflösung:

$$\sum_{i=0}^{j} \tilde{S}^{j-i} h \begin{pmatrix} P_i r_i \\ Q_i r_i \end{pmatrix} = \begin{pmatrix} P_j r_j \\ Q_j r_j \end{pmatrix} - \sum_{i=0}^{j-1} \tilde{S}^{j-1-i}\mathcal{O}(h\|\xi_i\|).$$

Mit der Stabilität der Matrizen V folgt:

$$\begin{aligned}
\|\sum_{i=0}^{j} V^{j-i} h P_i r_i\| &\leq \|P_j\xi_j\| + \mathcal{O}(h) \sum_{i=0}^{j-1} \|V\|^{j-1-i}\|\xi_i\| \\
&\leq (1 + \mathcal{O}(h)j) \max_{0\leq j\leq n} \|\xi_j\| \\
&\leq K \max_{0\leq j\leq n} \|\xi_j\|
\end{aligned}$$

für hinreichend großes $K > 0$. Der Beweis der zweiten Ungleichung des Zusatzes folgt analog.

□

Bemerkung 3.24 *Der Beweis macht deutlich, dass die Aussagen des Korollars auch für eine stabile Matrix M mit $\rho(M) = 1$ richtig bleiben. Die Abschätzung* (3.52) *lautet dann*

$$\|\xi_n\| \leq C\Big(\|\xi_0\| + hD + h\|\sum_{k=1}^{n} E_V P_k r_k\| + h\|\sum_{k=1}^{n} E_M Q_k r_k\| + hD'\Big),$$

wobei die Projektion E nun mit E_V bezeichnet wurde und E_M die entsprechende Projektion auf den Eigenraum zum Eigenwert 1 der Matrix M ist.

Eine leichte Folge des Konvergenzlemmas ist:

Satz (über die Konvergenz in y) 3.25
Angenommen für die Index-2 DAE (3.5) *gelte* (DAE1)-(DAE3) *bzw. es gelte die Verschärfung* (DAE4) *für $r \geq 1$. Für das allgemeine lineare Verfahren setzen wir* (GLM1)-(GLM4) *bzw. sogar $\rho(M(\infty)) < 1$ voraus.*
Erfüllt der lokale Fehler für $0 \leq nh \leq konst.$ die "minimalen" Voraussetzungen

$$\delta_n^z = \mathcal{O}(h), \qquad Q_0\delta_0^y = \mathcal{O}(h^2)$$

und gilt für den lokalen Fehler in y die Darstellung

$$\begin{aligned} \delta_0^y &= \mathcal{O}(h^r), \\ \delta_{n+1}^y &= d_r(x_n)h^r + \mathcal{O}(h^{r+1}), \\ EP(x_{n+1})\delta_{n+1}^y &= \mathcal{O}(h^{r+1}) \end{aligned} \tag{3.57}$$

für eine stetig differenzierbare Funktion $d_r(x)$, so ist das allgemeine lineare Verfahren in der y-Komponente konvergent von der Ordnung r.
Zusatz: Seien auch die correct value Funktionen r-mal stetig differenzierbar. Unter den minimalen Voraussetzungen oben sind die Bedingungen in (3.57) *nicht nur hinreichend für die Konvergenz der Ordnung r, sondern auch notwendig.*

Beweis: Der Beweis kann völlig analog zum Beweis von Korollar 3.21 (ii) geführt werden. Tatsächlich kann Lemma 3.13, insbesondere die Abschätzung (3.32), weiterhin angewendet werden. Statt (3.31) muss jedoch die entsprechende Abschätzung (3.52) aus Lemma 3.23 verwendet werden.
Beweis des Zusatzes (vgl. [HW] S.437ff für gewöhnliche Differentialgleichungen): Nach Voraussetzung sind die rechte Seite f der Differential-Algebraischen Gleichung und die *correct value* Funktion r-mal stetig differenzierbar sind. Die Funktion g der algebraischen Gleichung ist sogar $r+1$-mal stetig differenzierbar. Wir können in diesem Fall den lokalen Fehler der y-Komponente in eine Taylorreihe in h entwickeln:

$$\delta_{n+1}^y = d_0^y(x_n) + d_1^y(x_n)h + \ldots + d_r^y(x_n)h^r + \mathcal{O}(h^{r+1}).$$

Die Funktionen $d_j^y(x)$ sind dabei $(r-j+1)$-mal stetig differenzierbar (vgl. im Fall von gewöhnlichen Differentialgleichungen (2.6)).
Die Anwendung von Lemma 3.23 auf die Darstellung des globalen Fehlers (3.49) liefert nach dem Zusatz des Konvergenzlemmas:

$$\begin{aligned} \max_{0\leq j\leq n} \| \sum_{i=0}^{j} V^{j-i} P_i \delta_i^y \| &\leq K \max_{0\leq j\leq n} \|\Delta y_j\|, \\ \max_{0\leq j\leq n} \| \sum_{i=0}^{j} M^{j-i} Q_i \delta_i^y \| &\leq K \max_{0\leq j\leq n} \|\Delta y_j\|. \end{aligned}$$

Setzen wir die Konvergenz der Ordnung r voraus, so folgt unmittelbar

$$\begin{aligned} \delta_0^y &= \mathcal{O}(h^r), \\ \delta_{n+1}^y &= d_r^y(x_n)h^r + \mathcal{O}(h^{r+1}) \end{aligned}$$

für alle $0 \leq nh \leq konst$. Aufgrund der Zerlegung (3.53) gilt $EV = E$. Daraus wiederum folgt die Identität

$$E(P_0\delta_0^y + \ldots + P_n\delta_n^y) + P_{n+1}\delta_{n+1}^y = \sum_{j=0}^{n+1} V^{n+1-j}P_j\delta_j^y - (V-E)\sum_{j=0}^{n} V^{n-j}P_j\delta_j^y.$$

Die Anwendung der Dreiecksungleichung liefert:

$$\begin{aligned} \|E(P_0\delta_0^y + \ldots + P_n\delta_n^y) + P_{n+1}\delta_{n+1}^y\| &\leq (1 + \|V-E\|)K \max_{0\leq j\leq n} \|\Delta y_j\| \\ &= \mathcal{O}(h^r). \end{aligned} \tag{3.58}$$

Der Beweis wird nun völlig analog zum Beweise von Lemma 8.11 in [HW] S. 437 geführt: Wir betrachten Paare (n, h) für fest gewähltes $x = nh$. Mit der Darstellung des lokalen Fehlers finden wir

$$E(P_1\delta_1^y + \ldots + P_n\delta_n^y) = h^r E\sum_{j=1}^{n} P_j d_r^y(x_{j-1}) + \mathcal{O}(h^r).$$

Wir investieren $P_j = P_{j-1} + \mathcal{O}(h)$ und approximieren die erhaltene Summe durch das entsprechende Riemann-Integral:

$$E(P_1\delta_1^y + \ldots + P_n\delta_n^y) = h^{r-1}\int_0^x EP(s)d_r^y(s) + \mathcal{O}(h^r).$$

Aufgrund der Abschätzung (3.58) ist das Integral gleich 0. Da aber x beliebig gewählt werden kann, verschwindet der Integrand, das heißt es gilt:

$$EP(x)d_r^y(x) = 0,$$

womit der Zusatz vollständig bewiesen ist.

□

Aufgrund dieses Satzes definieren wir die Konsistenz der y-Komponente für allgemeine lineare Verfahren angewendet auf DAEs der Form (3.5) wie folgt (vgl. Definition 2.13 bzw. die äquivalente Formulierung (2.7)):

Definition 3.26 *Ein allgemeines lineares Verfahren angewendet auf Index-2 DAEs in Hessenberg Form ist in der y-Komponente konsistent der Ordnung p, falls gilt:*

$$\begin{aligned} \delta_n^y &= \mathcal{O}(h^p), \\ (E \otimes P(x_{n+1}))\delta_{n+1}^y &= \mathcal{O}(h^{p+1}) \end{aligned}$$

für $0 \leq nh \leq konst$.

Konvergenz der z-Komponente
Als Anwendung des Stabilitätssatzes 3.18 erhalten wir das folgende Konvergenzresultat:

Satz (über die Konvergenz in z) 3.27
Angenommen für die Index-2 DAE (3.5) *gelte* (DAE1)-(DAE3). *Für das allgemeine lineare Verfahren setzen wir* (GLM1)-(GLM4) *bzw. sogar* $\rho(M(\infty)) < 1$ *voraus. Erfüllt der lokale Fehler für* $0 \le nh \le konst.$ *die Voraussetzungen*

$$\begin{aligned} \delta_0^y &= \mathcal{O}(h^r), & Q_0\delta_0^y &= \mathcal{O}(h^{r+1}), \\ \delta_n^z &= \mathcal{O}(h^r), & \delta_{n+1}^y &= \mathcal{O}(h^{r+1}) \end{aligned}$$

so ist das allgemeine lineare Verfahren (in der z-Komponente konvergent) von der Ordnung r.

Beweis: Unter diesen Voraussetzungen liegt die Lösung des numerischen Modells $T_h(u, v) = 0$ nach dem Beweis von Korollar 3.21 (i) in der Umgebung U_δ. Somit liefert Satze 3.18 (i) die Konvergenz der Ordnung r.

□

Bemerkung: *Für nilpotente Matrix $M(\infty)$ erhalten wir unter weniger starken Voraussetzungen an δ_0^y und δ_0^z nach k_0 Schritten Konvergenz der Ordnung r, wobei k_0 den Index der Nilpotenz bezeichnet (vgl. Bemerkung (iv) im Anschluss an Korollar 3.21). Wir setzen in diesem Fall nur voraus:*

$$\begin{aligned} \delta_0^z &= o(1), & Q_0\delta_0^y &= ho(1), \\ \delta_0^y &= \mathcal{O}(h^r), & Q_0\delta_0^y &= \mathcal{O}(h^{r-1}), \\ \delta_{n+1}^z &= \mathcal{O}(h^r), & \delta_{n+1}^y &= \mathcal{O}(h^{r+1}). \end{aligned}$$

Dabei ist in Abhängigkeit von r jeweils eine Bedingung an $Q_0\delta_0^y$ hinfällig.

Diese Aussage ist leicht einzusehen. Sie folgt unmittelbar aus der Bemerkung (iv) im Anschluss an Korollar 3.21 und aus der folgenden Darstellung aus dem Beweis von Lemma 3.16:

$$\begin{aligned} \Delta z_{k_0} &= M(\infty)\Delta z_{k_0-1} + BA^{-1}\Delta Z_{k_0-1} + \delta_{k_0}^z \\ &= \underbrace{M(\infty)^{k_0}}_{=0}\Delta z_0 + \sum_{i=0}^{k_0-1} M^{k_0-1-i}[BA^{-1}\Delta Z_i + \delta_{i+1}^z]. \end{aligned}$$

3.3 Semi-explizite Index-2 DAEs

In diesem Unterkapitel übertragen wir die Konvergenzresultate allgemeiner linearer Verfahren für Index-2 DAEs in Hessenberg Form auf andere semi-explizite DAEs mit Störungsindex 2. Eine solche Übertragung ist in [HW] (S. 456) angedeutet und in [BP89] sehr kurz skizziert. Wir werden dabei zunächst eine leichte Verallgemeinerung von Index-2 DAEs in Hessenberg Form betrachten. Diese verallgemeinerte Klasse von DAEs enthält die augmentierte Index-2 Formulierung mechanischer Mehrkörpersysteme (vgl. Beispiel 3.28). Zum anderen bildet sie die Grundlage, die Konvergenzresultate auf eine noch umfassendere Klasse von semi-expliziten DAEs mit Störungsindex 2 zu übertragen, welche auch typische Gleichungen der Schaltkreissimulation enthält.

Augmentierung mechanischer Mehrkörpersysteme

Beispiel 3.28 *Die Bewegungsgleichungen eines mechanischen Mehrkörpersystems auf dem Geschwindigkeitslevel sind gegeben durch (vgl. Kapitel 1):*

$$\begin{aligned} \dot{p} &= v, \\ M(p)\dot{v} &= f(t,p,v) - G(p)^T\lambda, \\ 0 &= G(p)v. \end{aligned} \tag{3.59}$$

Dabei sei die Massematrix $M(p)$ invertierbar und der Rang von $G(p)$ maximal. Streng genommen handelt es sich bei (3.59) *um ein implizites System. Durch Multiplikation der zweiten Gleichung von links mit $M(p)^{-1}$ erhalten wir eine Index-2 DAE in Hessenberg Form. Da der Index offensichtlich invariant ist unter einer solchen Multiplikation, handelt es sich bei* (3.59) *um eine implizite Index-2 DAE. Wollen wir jedoch diese Multiplikation vermeiden und* (3.59) *direkt durch ein allgemeines lineares Verfahren diskretisieren, so bildet nach Unterkapitel 3.1 das folgende augmentierte System die Grundlage:*

$$\begin{aligned} \dot{p} &= v, \\ \dot{v} &= w, \\ 0 &= M(p)w - f(t,p,v) + G(p)^T\lambda, \\ 0 &= G(p)v. \end{aligned}$$

Diese augmentierte Formulierung mechanischer Mehrkörpersysteme auf dem Geschwindigkeitslevel und ebenso die Augmentierung der entsprechenden GGL-Formulierung gehören zu der Klasse der DAEs

$$\begin{aligned} y' &= f(y,w,z), & y(0) &= y_0 \in \mathbb{R}^N, \\ 0 &= k(y,w,z), & w(0) &= w_0 \in \mathbb{R}^M, \\ 0 &= g(y), & z(0) &= z_0 \in \mathbb{R}^l \end{aligned} \tag{3.60}$$

mit invertierbarer partieller Ableitung $k_w(y, w, z)$ in der Nähe der Lösung. Für die mechanischen Mehrkörpersysteme gilt $N = M$. Nach dem Satz über implizite Funktionen lässt sich die zweite Gleichung nach w auflösen und wir erhalten mit einer entsprechenden auflösenden Funktion H zumindest auf einem kompakten Intervall

$$w(x) = H(y(x), z(x)). \tag{3.61}$$

Hier bezeichne $(y(x), w(x), z(x))$ die Lösung der DAE auf $[0, x_e]$, von deren Existenz wir ausgehen. Des Weiteren sei neben k auch die Funktion f stetig differenzierbar. Die Funktion g sei sogar zweimal stetig differenzierbar.
Ersetzen wir $w(x)$ in der Differentialgleichung gemäß Gleichung (3.61), so erhalten wir die folgende DAE in Hessenberg Form:

$$\begin{array}{rclcrcl} y' & = & f(y, H(y,z), z), & & y(0) & = & y_0 \in \mathbb{R}^N, \\ 0 & = & g(y), & & z(0) & = & z_0 \in \mathbb{R}^l. \end{array} \tag{3.62}$$

Gehen wir nun davon aus, dass diese DAE den Index-2 besitzt, dass also die Matrix

$$Dg(y)\frac{\partial f}{\partial z}(y, H(y,z), z)$$

in einer Umgebung der Lösung invertierbar ist, so besitzt auch die DAE (3.60) den Index 2. Zusammenfassend setzen wir voraus:

Die Matrix $\boldsymbol{k_w(y, w, z)}$ ist invertierbar. (DAE0)

Die Matrix $\boldsymbol{Dg(y)\dfrac{\partial f}{\partial z}(y, H(y,z), z)}$ ist invertierbar. (DAE1)

Die Funktionen $\boldsymbol{f, k}$ sind einmal stetig differenzierbar, $\boldsymbol{g}$ sogar zweimal. (DAE2)

Es existiert eine eind. Lösung $\boldsymbol{(y(x), w(x), z(x))}$ auf $\boldsymbol{[0, x_e]}$. (DAE3)

Die entsprechende Verschärfung von (DAE2), welche wir in Satz 3.31 benötigen, lautet:

Die Funktionen $\boldsymbol{f, k}$ sind $\boldsymbol{r}$-mal, $\boldsymbol{g}$ sogar $\boldsymbol{(r+1)}$-mal stetig differenzierbar. (DAE4)

Welche Aussagen lassen sich für allgemeine lineare Verfahren angewendet auf DAEs der Form (3.60) unter diesen Voraussetzungen treffen? Sind die Konvergenzresultate des letzten Abschnitts übertragbar bzw. genauer verallgemeinerbar?

Alle nun folgenden Aussagen lassen sich ohne großen Aufwand auch für nichtautonome Index-2 DAEs der Form (3.60) formulieren. Wir behandeln jedoch aufgrund einer übersichtlicheren Darstellung wiederum den autonomen Fall.

Wir betrachten ein allgemeines lineares Verfahren, für welches wir wie im letzten Unterkapitel (GLM1)-(GLM3) voraussetzen. Angewandt auf DAEs der Form (3.60) lautet ein solches Verfahren:

$$\begin{aligned} y^{[n+1]} &= Vy^{[n]} + hBf(Y,W,Z), \\ w^{[n+1]} &= M(\infty)w^{[n]} + BA^{-1}W, \\ z^{[n+1]} &= M(\infty)z^{[n]} + BA^{-1}Z, \\ \\ Y &= Uy^{[n]} + hAf(Y,W,Z), \\ 0 &= k(Y,W,Z), \\ 0 &= g(Y). \end{aligned} \tag{3.63}$$

Liegen die Stufenwerte Y, W und Z bzw. deren s Subvektoren hinreichend nahe der Lösung, so lässt sich die zweite Gleichung des Gleichungssystems der Stufenwerte nach W auflösen:

$$W = H(Y,Z). \tag{3.64}$$

Das Gleichungssystem lässt sich unter dieser Voraussetzung reduzieren zu

$$\begin{aligned} Y &= Uy^{[n]} + hAf(Y,H(Y,Z),Z), \\ 0 &= g(Y). \end{aligned}$$

Wir betrachten nun die Anwendung des allgemeinen linearen Verfahrens auf die Index-2 DAE in Hessenberg Form (3.62):

$$\begin{aligned} y^{[n+1]} &= Vy^{[n]} + hBf(Y,W,Z), \\ z^{[n+1]} &= M(\infty)z^{[n]} + BA^{-1}Z, \\ \\ Y &= Uy^{[n]} + hAf(Y,H(Y,Z),Z), \\ 0 &= g(Y). \end{aligned} \tag{3.65}$$

Unter der Auflösbarkeitsvoraussetzung (3.64) ist das Gleichungssystem der Stufenwerte eines allgemeinen linearen Verfahrens angewendet auf die DAE (3.60) und die reduzierte DAE in Hessenberg Form (3.62) bis auf die zusätzlichen Stufenwerte W identisch (vgl. jeweils die unteren Gleichungen in (3.63) und (3.65)). Ebenso verhält es sich mit den Iterationen (vgl. jeweils die y- und z-Iteration in (3.63) und (3.65)). Wir erhalten somit entsprechende Konvergenzaussagen für die y- und z-Komponente aus den Konvergenzaussagen allgemeiner linearer Verfahren angewendet auf Index-2 DAEs in Hessenberg Form (vgl. Abbildung 3.4).

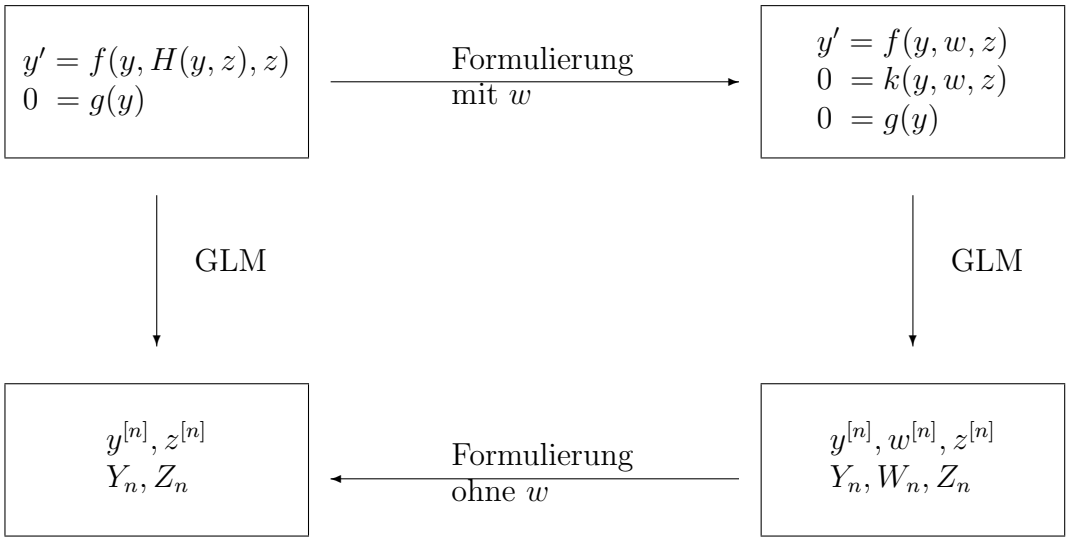

Abbildung 3.4: Äquivalenz der y- und z-Komponente allgemeiner linearer Verfahren mit invertierbarer Verfahrensmatrix A angewendet auf DAEs der Form (3.62) und auf (3.60) unter der Auflösbarkeitsvoraussetzung (3.64).

Wir untersuchen im Folgenden, unter welchen Voraussetzungen die Auflösbarkeit in (3.64) möglich ist und welche Konvergenzaussagen für die w-Komponente gelten. Dabei lassen sich die Ergebnisse aus dem letzten Unterkapitel nicht einfach übertragen, da es sich bei (3.60) um eine Verallgemeinerung der Index-2 DAEs in Hessenberg Form handelt. Wir formulieren im Folgenden die wichtigsten verallgemeinerten Resultate, ohne sie im einzelnen zu beweisen.

Wir definieren den Operator F wie folgt (vgl. zur Motivation dieser Definition Abschnitt 3.2.1):

$$F(h,\eta,Y,W,Z) = \begin{pmatrix} Y - \eta - hAf(Y,Z) \\ k(Y,W,Z) \\ g(\eta)/h + \int_0^1 Dg(\eta + s(Y-\eta))ds Af(Y,Z) \end{pmatrix}.$$

Entsprechend zu Satz 3.7 gilt:

Satz (über die lokale Lösbarkeit des Gleichungssystems) 3.29
Wir setzen (DAE0)-(DAE3) *und* (GLM1) *voraus. Angenommen* $\eta = \eta(h), \zeta$ *und* ξ *genügen*

$$\begin{aligned} \eta - \mathbb{1} \otimes y(x) &= o(1), \\ \zeta - \mathbb{1} \otimes w(x) &= o(1), \\ \xi - \mathbb{1} \otimes z(x) &= o(1), \\ g(\eta)/h &= o(1), \end{aligned} \tag{3.66}$$

wobei die $o(1)$*-Terme hinreichend klein sind und nicht notwendiger Weise von* h *abhängen.*

Dann existieren $h_0 > 0$ und ein "Radius" $r > 0 \in \mathbb{R}^3$ unabhängig von h_0 und den $o(1)$-Termen, so dass das System

$$\begin{aligned} 0 &= Y - \eta - hAf(Y, W, Z), \\ 0 &= k(Y, W, Z), \\ 0 &= g(Y) \end{aligned}$$

für $h \leq h_0$ eine eindeutige Lösung

$$(Y(h,\eta), W(h,\eta), Z(h,\eta)) \in B_r = \{V \in \mathbb{R}^{s(N+M+l)} |\, |V - (\eta, \zeta, \xi)| \leq r\}$$

besitzt.
Weiterhin existieren nicht negative Matrizen $P, K \in \mathbb{R}^{3,3}$ für solche h mit $I - PK$ regulär und

$$((I - PK)^{-1}P)_{12} = \mathcal{O}(h), \qquad ((I - PK)^{-1}P)_{13} = \mathcal{O}(h),$$

so dass die folgende Stabilitätsungleichung in B_r gilt:

$$|(Y, W, Z) - (\bar{Y}, \bar{W}, \bar{Z})| \leq (I - PK)^{-1}P|F(h, \eta, Y, W, Z) - F(h, \eta, \bar{Y}, \bar{W}, \bar{Z})|. \quad (3.67)$$

Insbesondere gilt

$$Y(h,\eta) - \eta = \mathcal{O}(h), \qquad W(h,\eta) - \zeta = o(1), \qquad Z(h,\eta) - \xi = o(1).$$

Zusatz: Ist die Funktion k unabhängig von der algebraischen Variablen z, so gilt ebenfalls

$$((I - PK)^{-1}P)_{23} = \mathcal{O}(h).$$

Bemerkung: *Wie im Anschluß an den Beweis von Satz 3.7 lässt sich die lokale Unabhängigkeit der Lösung von ζ und ξ begründen. Vor diesem Hintergrund ist die im Satz oben verwendete Schreibweise der Lösung zu verstehen, die nur die Abhängigkeit von η verdeutlicht.*

Beweis: Der Beweis wird analog zum Beweis von Satz 3.7 geführt. Das einzige neue Argument bringt die Anwendung des Satzes über implizite Funktionen: In der Nähe der Lösung gilt:

$$0 = k(y, w, z) \Longleftrightarrow w = H(y, z).$$

Differenzieren nach z liefert die Identität

$$H_z(y, z) = -(k_w(y, H(y, z), z))^{-1} k_z(y, H(y, z), z).$$

□

Wie formulieren nun zwei Anwendungen der Stabilitätsungleichung 3.67, die völlig analog zur entsprechenden Anwendung in Unterkapitel 3.2 zu beweisen sind. Dies ist zum einen ein Analogon zu Korollar 3.10:

Korollar 3.30
Es seien Vektoren $y, \bar{y} \in \mathbb{R}^{rN}, w, \bar{w} \in \mathbb{R}^{rM}$ *und* $z, \bar{z} \in \mathbb{R}^{rl}$ *gegeben. Zudem setzen wir* (DAE0)-(DAE3) *und* (GLM1) *voraus. Angenommen*

$$\begin{array}{lll} \eta = (U \otimes I)y, & \zeta = (U \otimes I)w, & \xi = (U \otimes I)z, \\ \bar{\eta} = (U \otimes I)\bar{y}, & \bar{\zeta} = (U \otimes I)\bar{w}, & \bar{\xi} = (U \otimes I)\bar{z} \end{array}$$

erfüllen die Voraussetzungen von Satz 3.29, so dass die Lösungen $Y(h,y), W(h,y)$, $Z(h,y)$ *und* $Y(h,\bar{y}), W(h,\bar{y}), Z(h,\bar{y})$ *der entsprechenden Gleichungssysteme definiert sind. Zudem gelte*

$$y - \bar{y} = \mathcal{O}(h), \qquad w - \bar{w} = o(1), \qquad z - \bar{z} = o(1). \tag{3.68}$$

Dann existieren für hinreichend kleinen o-Term positive Konstanten K_1, K_2 *und* $h_0 > 0$, *so dass für* $h \leq h_0$ *die folgenden Abschätzungen gelten:*

$$\begin{aligned} \|Y(h,y) - Y(h,\bar{y})\| &\leq K_1 \|y - \bar{y}\|, \\ \|W(h,y) - W(h,\bar{y})\| &\leq \frac{K_1}{h}\|Dg(\eta)(y - \bar{y})\| + K_2\|y - \bar{y}\|, \\ \|Z(h,y) - Z(h,\bar{y})\| &\leq \frac{K_1}{h}\|Dg(\eta)(y - \bar{y})\| + K_2\|y - \bar{y}\|. \end{aligned}$$

Ist die Funktion k *unabhängig von der algebraischen Variablen* z, *so ist die folgende strengere Abschätzung möglich:*

$$\|W(h,y) - W(h,\bar{y})\| \leq K_1\|y - \bar{y}\|.$$

Zusatz: Die Konstante K_1 *hängt dabei nur von Schranken gewisser Ableitungen von* f, k *und* g *ab, nicht jedoch von den Konstanten, die implizit durch die* $\mathcal{O}$- *und o-Terme in (3.66) und (3.68) gegeben sind.* K_2 *hingegen hängt in linearer Weise von der Konstanten aus dem* $\mathcal{O}$*-Term in (3.68) ab.*

Zum anderen finden wir wie im Beweis von Satz 3.22 durch eine Anwendung der Stabilitätsungleichung 3.67 die Darstellungen

$$\begin{aligned} Y(h, y_c(x_n,h)) - y(x_n + ch) &= \mathcal{O}(h^{q+1}), \\ W(h, y_c(x_n,h)) - w(x_n + ch) &= \mathcal{O}(h^q), \\ Z(h, y_c(x_n,h)) - z(x_n + ch) &= \mathcal{O}(h^q). \end{aligned} \tag{3.69}$$

Dabei gilt für eine von z unabhängige Funktion k sogar:

$$W(h, y_c(x_n,h)) - w(x_n + ch) = \mathcal{O}(h^{q+1}).$$

Mit diesen Abschätzungen erhalten wir für den lokalen Fehler ergänzend zu Satz 3.22 im Allgemeinen:

Satz (über den lokalen Fehler) 3.31
Für die Index-2 DAE (3.60) *setzen wir* (DAE0)-(DAE3) *bzw. die Verschärfung* (DAE4) *für* $q \geq 1$ *voraus. Angenommen ein allgemeines lineares Verfahren mit invertierbarer Koeffizientenmatrix A besitzt für gewöhnliche Differentialgleichungen die Konsistenzordnung p und die Stufenordnung q mit* $p \geq q \geq 1$. *Dann gilt für den lokalen Fehler der w-Komponente die Darstellung*

$$\delta_{n+1}^{w} = \mathcal{O}(h^q).$$

Fazit: Die Iteration der w-Komponente ist von der gleichen Form wie die der z-Komponente. Auch sind für die Stufenwerte im Allgemeinen nach Korollar 3.30 und Darstellung (3.69) die gleichen Abschätzungen möglich. Aus diesen Aussagen lässt sich folgern: Die Voraussetzungen an ein allgemeines lineares Verfahren angewendet auf (3.62), die eine Konvergenz einer gewissen Ordnung in y und z garantieren, liefern eine entsprechende Konvergenz des Verfahrens angewendet auf (3.60). Die Auflösbarkeitvoraussetzung (3.64) ist für hinreichend kleine h erfüllt. Fordern wir für die w-Komponente diegleichen Voraussetzungen wie für die z-Komponente, so erhalten wir in w dieselbe Konvergenzordnung wie in der z-Komponente. Desweiteren besitzt der lokale Fehler eines allgemeinen linearen Verfahrens, welches für gewöhnliche Differentilagleichungen die Stufenordnung q und die Konsistenzordnung $p \geq q$ hat, in der w-Komponente dieselbe Ordnung wie in der z-Komponente.

Abschließend treffen wir Aussagen für den Fall, dass die Funktion k unabhängig von z ist. Es handelt sich dann bei

$$w(x) = H(y(x))$$

nicht um eine „algebraische“ sondern vielmehr um eine „differentielle“ Variable. Die Konvergenzordnung kann genauso groß wie die Konvergenzordnung in y sein: Betrachten wir zum Beispiel ein steif genaues Runge-Kutta Verfahren, das heißt die letzten und s-ten Stufenwerte Y_s, W_s und Z_s bilden die neuen Iterierten, so folgt:

$$\begin{aligned} w^{[n+1]} - w(x_{n+1}) &= W_s(h, y^{[n]}) - w(x_{n+1}) \\ &= H(Y_s(h, y^{[n]})) - H(y(x_{n+1})) \\ &= H(y^{[n+1]}) - H(y(x_{n+1})). \end{aligned}$$

Eine Lipschitz-Stetigkeit der auflösenden Funktion H liefert in w dieselbe Konvergenzordnung wie in y. Auch für projizierte Kollokationsmethoden kann bei Unabhängigkeit der Funktion k von z die Konvergenzordnung in w auf die der y-Komponente gebracht werden (vgl. zur Definition [HW, AP91]). Das Gleichungssystem für die Projektion lautet hier

$$\begin{aligned} \hat{y}^{[n]} &= y^{[n]} + f_z(\hat{y}^{[n]}, \hat{w}^{[n]}, z^{[n]})\lambda_n, \\ 0 &= k(\hat{y}^{[n]}, \hat{w}^{[n]}), \\ 0 &= g(\hat{y}^{[n]}). \end{aligned}$$

Im Allgemeinen hingegen ist die Projektion durch

$$\begin{aligned} \hat{y}^{[n]} &= y^{[n]} + f_z(\hat{y}^{[n]}, w^{[n]}, z^{[n]})\lambda_n, \\ 0 &= g(\hat{y}^{[n]}) \end{aligned}$$

gegeben. Es wird also nur die y-Komponente projiziert.

Spezielle semi-explizite Index-2 DAEs

Wir betrachten nun semi-explizite Gleichungen

$$\begin{aligned} y' &= f(y,z), & y(0) &= y_0 \in \mathbb{R}^N, \\ 0 &= g(y,z), & z(0) &= z_0 \in \mathbb{R}^l \end{aligned} \tag{3.70}$$

mit konstantem Rang der Matrix $\frac{\partial g}{\partial z}(y,z)$ kleiner als l. Bezeichne $y(x), z(x)$ die Lösung der DAE von deren Existenz wir auf dem Intervall $[0, x_e]$ ausgehen. Seien f einmal stetig differenzierbar und g zweimal stetig differenzierbar. Wir nehmen zudem an, der Störungsindex dieser DAE sei 2. Zusammenfassend setzen wir also voraus:

Die Matrix $g_z(y,z)$ besitzt konstanten Rang. (DAE0)

Der Störungsindex der DAE (3.70) **ist 2.** (DAE1)

Die Funktion f ist einmal stetig differenzierbar, g zweimal. (DAE2)

Es existiert eine eindeutige Lösung $(y(x), z(x))$ auf $[0, x_e]$. (DAE3)

Beispiel 3.32 *Wir betrachten die lineare DAE*

$$A(t)[D(t)x(t)]' + B(t)x(t) = q(t), \quad t \in \mathcal{I} \tag{3.71}$$

mit auf einem kompakten Intervall $\mathcal{I}$ definierten matrixwertigen Funktionen $A \in C^1(\mathcal{I}, \mathbb{R}^{m,n})$, $D \in C^1(\mathcal{I}, \mathbb{R}^{n,m})$ und $B \in C^1(\mathcal{I}, \mathbb{R}^{m,m})$. Der Hauptterm dieser DAE sei proper formuliert (vgl. Anhang A.1). Zudem sei der Traktabilitätsindex dieser DAE 2 (vgl. zur Definition dieses Indexes [M02a]).
Numerische Verfahren werden auf die äquivalente Formulierung

$$\begin{aligned} A(t)y'(t) + B(t)x(t) &= q(t), \\ 0 &= y(t) - D(t)x(t) \end{aligned}$$

angewendet. Das entsprechend augmentierte System lautet:

$$\begin{aligned} y'(t) &= w(t), \\ 0 &= A(t)w(t) + B(t)x(t) - q(t), \\ 0 &= y(t) - D(t)x(t). \end{aligned}$$

Nach Theorem 2.3 in [M02a] besitzt dieses System unter den Voraussetzungen des proper formulierten Hauptterms und des Traktabilitätsindexes 2 auch den Störungsindex 2. Zudem lässt sich unter diesen Voraussetzungen zeigen, dass die Matrix

$$\begin{pmatrix} B(t) & A(t) \\ -D(t) & 0 \end{pmatrix}$$

konstanten Rang besitzt (vgl. Lemma A.2 in Anhang A). Die Matrix entspricht genau der partiellen Ableitung $g_z(y, z)$ von oben. Die DAE aus Beispiel 3.2 ist von der Form (3.71).

Aufgrund der konstanten Rang Bedingung (DAE0) existieren invertierbare Matrizen $S(x)$ und $T(x)$ mit

$$S(x)g_z(y(x), z(x))T(x) = \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix}$$

(vgl. [D64]). Tatsächlich hängen diese Matrizen von der Lösung $y(x), z(x)$ ab. Sie sind so glatt wie $g_z(y(x), z(x))$.
Wir multiplizieren die algebraische Gleichung mit S und nehmen die folgende Koordinatentransformation in z vor:

$$z = T(x)w.$$

Für die Variablen y und w erhalten wir

$$\begin{aligned} y' &= f(y, T(x)w), & y(0) &= y_0 \in \mathbb{R}^N, \\ 0 &= S(x)g(y, T(x)w), & w(0) &= T^{-1}(0)z_0 \in \mathbb{R}^l. \end{aligned} \tag{3.72}$$

Diese DAE ist von der Form (3.60). Der Störungsindex bleibt unter der Multiplikation und der Transformation erhalten. Ein allgemeines lineares Verfahren ist zumindest bei Multiplikation der algebraischen Gleichung mit einer invertierbaren Matrix invariant. Doch wie verhält sich das Verfahren unter der Transformation?

Wir wenden ein allgemeines lineares Verfahren auf die DAE (3.72) an:

$$\begin{aligned} y^{[n+1]} &= Vy^{[n]} + hBf(Y, T(x_n + ch)W), \\ w^{[n+1]} &= M(\infty)w^{[n]} + BA^{-1}W, \\ Y &= Uy^{[n]} + hAf(Y, T(x_n + ch)W), \\ 0 &= S(x_n + ch)g(Y, T(x_n + ch)W). \end{aligned} \tag{3.73}$$

Dabei haben wir die Schreibweise

$$T(x_n + ch)W := \mathrm{diag}\{T(x_n + c_1 h), \ldots, T(x_n + c_s h)\} \cdot W$$

verwendet. Die Multiplikation mit $S(x_n + ch)$ ist entsprechend zu verstehen. Definieren wir $Z := T(x_n + ch)W$, so ist (3.73) äquivalent zu:

$$\begin{aligned} y^{[n+1]} &= Vy^{[n]} + hBf(Y, Z), \\ w^{[n+1]} &= M(\infty)w^{[n]} + BA^{-1}T(x_n + ch)^{-1}Z, \end{aligned}$$

$$\begin{aligned} Y &= Uy^{[n]} + hAf(Y, Z), \\ 0 &= g(Y, Z). \end{aligned}$$

Dies entspricht jedoch bis auf die Iteration in w einem allgemeinen linearen Verfahren angewendet auf (3.70). Wir erhalten also Konvergenzaussagen eines allgemeinen linearen Verfahrens angewendet auf (3.70) bezüglich der y-Komponente aus den entsprechenden Aussagen eines allgemeinen linearen Verfahrens angewendet auf eine Index-2 DAE in Hessenberg Form.

Bemerkung 3.33 *Ist die Koordinatentransformation oben nicht nötig, das heißt es gilt $T(x) = I$, so erhalten wir auch in der z-Komponente eines allgemeinen linearen Verfahrens angewendet auf* (3.70) *entsprechende Konvergenzresultate aus Abschnitt 3.2.3.*

Beispiel 3.34 *Sei eine allgemeine implizite Index-1 DAE gegeben:*

$$0 = F(y', y), \qquad y(0) = y_0 \in \mathbb{R}^N,$$

wobei die Matrix $\frac{\partial F}{\partial y'}(y', y)$ konstanten Rang besitze. Das um $y' = w$ augmentierte System ist von der Form (3.70)*. Somit lassen sich insbesondere die Konvergenzresultate auf diese DAEs übertragen.*

4 Implementierung

In diesem Kapitel geben wir Hilfen für die Implementierung eines allgemeinen linearen Verfahrens angewandt auf Index-2 DAEs in Hessenberg Form. Dabei gehen wir auf die Herleitung von Startprozeduren und die geeignete Anwendung der vereinfachten Newton-Verfahrens ein.

Zur Erinnerung: Es sei eine Index-2 DAEs in Hessenberg Form gegeben:

$$\begin{array}{rcll} y' & = & f(x,y,z), & y(x_0) = y_0 \in \mathbb{R}^N, \\ 0 & = & g(x,y), & z(x_0) = z_0 \in \mathbb{R}^l, \end{array} \tag{4.1}$$

wobei wir die Invertierbarkeit von $Dg(y)f_z(y,z)$ in einer Umgebung der Lösung voraussetzen. Desweiteren seien die Anfangswerte konsistent.
Ein allgemeines lineares Verfahren für diese DAE lautet:

$$\begin{array}{rcl} y^{[n+1]} & = & (V \otimes I_N) y^{[n]} + h(B \otimes I_N) f(x_n + ch, Y, Z), \\ z^{[n+1]} & = & (V \otimes I_l) z^{[n]} + h(B \otimes I_l) Z', \end{array}$$

$$\begin{array}{rcl} Y & = & (U \otimes I_N) y^{[n]} + h(A \otimes I_N) f(x_n + ch, Y, Z), \\ Z & = & (U \otimes I_l) z^{[n]} + h(A \otimes I_l) Z', \\ 0 & = & g(x_n + ch, Y). \end{array}$$

Dabei sind die Startwerte der Iteration durch eine Startprozedur φ gegeben:

$$\begin{pmatrix} y^{[0]} \\ z^{[0]} \end{pmatrix} = \begin{pmatrix} \varphi^y(h, y_0, z_0) \\ \varphi^z(h, y_0, z_0) \end{pmatrix} = \varphi(h, y_0, z_0).$$

Doch wie könnte konkret eine solche Startprozedur aussehen?

4.1 Startprozedur

Wir beschäftigen uns in diesem Abschnitt mit der Berechnung der Startwerte $y^{[0]}$ und $z^{[0]}$. Bei der Theorie in Kapitel 3 haben wir die Existenz einer Startprozedur $\varphi(h, y_0, z_0)$ vorausgesetzt, welche Näherungen von $y_c(x_0, h)$ und $z_c(x_0, h)$ hinreichender Güte lieferte. Jetzt, wo wir allgemeine lineare Verfahren implementieren wollen, müssen wir Farbe bekennen und eine Startprozedur bereitstellen.
Tatsächlich ist dies streng genommen nicht ganz richtig. Viele Computerprogramme zum Lösen gewöhnlicher Differentialgleichungen besitzen neben einer variablen Schrittweite auch eine variable Ordnung. Es werden also verschiedene numerische Verfahren ein und derselben Klasse verwendet, die aber unterschiedliche Ordnungen besitzen. Typischerweise benötigten die Verfahren der Klasse mit niedriger Ordnung höchstens zwei Näherungen, welche durch den gegebenen Anfangswert y_0 und dessen "Ableitung " $f(y_0)$ berechnet werden können. Man startet also problemlos ohne Startprozedur mit einem Verfahren niedriger Ordnung und berechnet weitere Näherungen. Durch die zusätzlichen Näherungen können nun auch Verfahren höherer Ordnung verwendet werden. Diese Vorgehensweise ist ohne größeren Programmieraufwand möglich, da eine Strategie zur Wahl der Ordnung in diesen Programmen bereits realisiert ist. Theoretisch ist dadurch natürlich nur die minimale auftretende Ordnung garantiert. Praktisch gleicht man dies durch entsprechend kleine Schrittweiten aus, um eine vorgegebene Genauigkeit garantieren zu können. Der auf diese Weise beschriebene Aufbau der gewünschten Ordnung ist jedoch nicht sehr effizient. Zudem sind wir nicht an einem Lösungsverfahren mit variabler Ordnung interessiert und müssen daher eine Startprozedur implementieren.

Die wohl einfachste Vorgehensweise, Näherungen als Startwerte zum Beispiel für ein lineares Mehrschrittverfahren zu berechnen, liegt in der Verwendung von Runge-Kutta Verfahren entsprechend hoher Ordnung. Da die Konvergenzordnung von impliziten Runge-Kutta Verfahren angewendet auf Index-2 DAEs in Hessenberg Form im Allgemeinen nur von der Größe der Stufenordnung q ist (vgl. Abschnitt 3.2.3), muss die Stufenordnung des Runge-Kutta Verfahrens so groß sein wie die Konvergenz p des allgemeinen linearen Verfahrens, für welches die Näherungen bereitgestellt werden sollen. Dies bedeutet im Allgemeinen für Index-2 DAEs höhere Kosten als für gewöhnliche Differentialgleichungen, da in der Regel die Stufenordnung wesentlich kleiner als die Konvergenzordnung eines Runge-Kutta Verfahrens ist. Sollen jedoch für ein allgemeines lineares Verfahren mit nilpotenter Matrix $M(\infty)$ Startwerte berechnet werden, können dazu steif genaue Kollokationsmethoden verwendet werden. Dies liegt zum einen daran, dass wegen der Nilpotenz von $M(\infty)$ auf eine Berechnung der Startwerte für die z-Komponente höherer Güte verzichtet werden kann. Die Ordnung q ist hier ausreichend. Besitzt nämlich $M(\infty)$ den Index k_0, das heißt

es gilt $M(\infty)^{k_0} = 0$, so finden wir:

$$\begin{aligned} z^{[k_0]} &= M(\infty) z^{[k_0-1]} + BA^{-1} Z(h, y^{[k_0-1]}) \\ &= \sum_{i=0}^{k_0-1} M(\infty)^i BA^{-1} Z(h, y^{[k_0-1-i]}). \end{aligned}$$

Da die Stufenwerte $Z(h, y^{[k_0-1-i]})$ unabhängig von den Näherungen $z^{[k_0-1-i]}$ sind (solange der lokale Fehler der z-Komponente des allgemeinen linearen Verfahrens und die Stufenordnung der Kollokationsmethode größer 1 sind), ist $z^{[k_0]}$ unabhängig von den Anfangsiterationen $z^{[0]}, \ldots, z^{[k_0-1]}$. Es ist also unproblematisch mit Startwerten niedriger Ordnung in der z-Komponente zu rechnen. Nach k_0 Schritten erhalten wir die Ordnung in z, welche das allgemeine lineare Verfahren garantiert. Zum anderen können steif genaue Kollokationsmethoden verwendet werden, da sie in der y-Komponente dieselbe Konvergenzordnung p wie für gewöhnliche Differentialgleichungen aufweisen (vgl. [HW] S. 500). Dies bedeutet konkret, dass zum Beispiel ein Radau IIA Runge-Kutta Verfahren der Ordnung 5 und der Stufenordnung 3 verwendet werden kann, um Startwerte für ein allgemeines lineares Verfahren der Ordnung 5 mit nilpotenter Matrix $M(\infty)$ zu berechnen.
Für eine allgemeine Matrix $M(\infty)$ lässt sich die Berechnung durch *singly implicit* Runge-Kutta Verfahren mit $p = q$ realisieren, deren Existenz gewährleistet ist (vgl. [BBC80]). Dabei ist p sowohl die Ordnung des Runge-Kutta als auch des allgemeinen linearen Verfahrens. Ein Nachteil liegt bei der Berechnung der Stufenwerte: Für diese Verfahren gilt $s = q$, so dass relativ viele Stufenwerte (s·(Anzahl der Näherungen)) berechnet werden müssen.

Gear schlug 1980 *Runge-Kutta ähnliche* Methoden als Startprozedur für lineare Mehrschrittverfaren vor, welche schon für gewöhnliche Differentialgleichungen kostengünstiger als die oben skizzierte Verwendung klassischer Runge-Kutta Verfahren sind (vgl. [G80]). Die Motivation dieser Arbeit lag hauptsächlich im Lösen gewöhnlicher Differentialgleichungen, deren rechte Seite viele Unstetigkeiten aufweist. Runge-Kutta Verfahren lösen solche Gleichungen problemlos, so lange die Stellen der Unstetigkeit auf Gitterpunkte fallen. Lineare Mehrschrittverfahren, welche in jeder Iteration auf der Glattheit der Gleichung auf einem Intervall $[x_{n-1}, x_n]$ ausgehen, müssen bei jeder Unstetigkeitsstelle neu gestartet werden. Gear betrachtete ausführlich explizite Startprozeduren. Dies liegt daran, dass bei vielen steifen Differentialgleichungen unmittelbar nach einer Unstetigkeit die schnellen transienten Lösungskomponenten dominant sind (vgl. [G80]), das heißt die Probleme sind dort nicht steif. Zur Erinnerung: Differentialgleichungen werden als steif bezeichnet, wenn schnelle transiente Lösungskomponenten vernachlässigt werden können. Unsere Erfahrung aus Abschnitt 3.2.1 lässt aber vermuten, dass wir nur implizite Startprozeduren für DAEs verwenden können.

Wir greifen nun die Idee von Gear auf und entwickeln implizite Runge-Kutta ähnliche Startprozeduren für allgemeine lineare Verfahren angewendet auf Index-2 DAEs

in Hessenberg Form. Wir tun dies zum einen, da diese Startprozeduren kostengünstiger sind als die oben angesprochene Verwendung klassischer Runge-Kutta Verfahren, und zum anderen, da diese Prozeduren auch für DAEs mit Unstetigkeiten verwendet werden könnten. Der Startvektor wird in Nordsieck-Form berechnet. Andere Typen von *correct value* Funktionen zum Beispiel

$$y_c(x,h) = \begin{bmatrix} y(x) \\ y(x-h) \\ \vdots \\ y(x-(r-1)h) \end{bmatrix}$$

können bei entsprechender Größe von r aus einfachen Transformationen gewonnen werden (vgl. [G80, Sjö99]).

Sei eine gewöhnliche Differentialgleichung gegeben:

$$y' = f(x,y), \qquad y(x_0) = y_0.$$

Wir berechnen Stufenwerte Y als Lösung des Gleichungssystems

$$Y = \mathbb{1} \otimes y_0 + h(\hat{A} \otimes I) f(x_0 + \hat{c}h, Y). \tag{4.2}$$

Dabei ist $\hat{A}$ eine $\hat{s} \times \hat{s}$-Matrix und $\hat{c}$ ein $\hat{s}$-dimensionaler Vektor. Wir bestimmen $\hat{s}$, die Matrizen $\hat{A}, \hat{B}$ und den Knotenvektor $\hat{c}$, so dass wir

$$y_c(x_0,h) = \begin{bmatrix} y(x_0) \\ hy'(x_0) \\ \vdots \\ h^{r-1}y^{(r-1)}(x_0) \end{bmatrix} = e_1 \otimes y_0 + h(\hat{B} \otimes I) f(x_0 + \hat{c}h, Y) + \mathcal{O}(h^{\hat{p}+1})$$

für vorgegebenes $\hat{p}$ garantieren können. Hier ist e_1 der erste r-dimensionale Einheitsvektor und $\hat{B} \in \mathbb{R}^{r,\hat{s}}$. Es werden also Näherungen des Nordsieckvektors $y_c(x_0,h)$ berechnet. Wir bezeichnen die Güte dieser Approximation, gemessen in Potenzen von h, als die Ordnung $\hat{p}$ der Startprozedur. Dabei kann die erste Zeile von $\hat{B}$ identisch Null gewählt werden.
Will man in dieser Rechnung ebenfalls ausnutzen, dass $hy'(x_0)$ durch die Differentialgleichung gegeben ist, so wählt man $c_1 = 0$ und auch die erste Zeile der Matrix $\hat{A}$ identisch Null. Zusätzlich garantiert man

$$e_2^T \hat{B} = e_1^T$$

für entsprechend dimensionale Einheitvektoren (vgl. [Wright02] S.108f, wo einige solcher Startprozeduren angegeben sind und auch eine systematische Berechnung beschrieben ist). Die Singularität der Matrix $\hat{A}$ ist aber bei DAEs nicht erwünscht. Eine Runge-Kutta ähnliche Startprozedur ist gegeben durch

$$\left[\begin{array}{c|c} \hat{c} & \hat{A} \\ \hline e_1 & \hat{B} \end{array}\right].$$

Analog zu den Überlegungen am Anfang von Kapitel 3, wo wir die Anwendung allgemeine lineare Verfahren für Index-2 DAEs der Form (4.1) beschrieben haben, formulieren wir eine Runge-Kutta ähnliche Startprozedur für diese DAEs. Das Gleichungssystem der Stufenwerte besitzt dabei eine aus Abschnitt 3.2.1 bekannte Struktur

$$\begin{aligned} Y &= \mathbb{1} \otimes y_0 + h(\hat{A} \otimes I) f(x_0 + \hat{c}h, Y, Z), \\ Z &= \mathbb{1} \otimes z_0 + h(\hat{A} \otimes I) Z', \\ 0 &= g(x_0 + \hat{c}h, Y). \end{aligned}$$

Bevor wir auch die Iteration der Startprozedur angeben, wiederholen wir einige Aussagen aus Kapitel 3 über die Stufenwerte: Besitzt die Runge-Kutta ähnliche Startprozedur für gewöhnliche Differentialgleichungen die Stufenordnung $\hat{q}$, das heißt erfüllt die Lösung der Differentialgleichung ausgewertet bei $x_0 + \hat{c}h$ die Gleichung (4.2) bis auf einen Fehler der Größe $\mathcal{O}(h^{\hat{q}+1})$, so löst auch $y(x_0 + \hat{c}h), z(x_0 + \hat{c}h)$ die erste Gleichung des Systems oben mit einem Fehler dieser Größe. Hier bezeichnet $y(x), z(x)$ die Lösung der Index-2 DAE (4.1). Also besitzt die Runge-Kutta ähnliche Prozedur auch für Index-2 DAEs in Hessenberg Form die Stufenordnung $\hat{q}$. Satz 3.7 liefert zudem für hinreichend kleine h und invertierbarer Matrix $\hat{A}$ die lokale Lösbarkeit dieses Gleichungssystems und mit der Stabilitätsungleichung (3.14) des Satzes folgt wieder

$$\begin{aligned} Y - y(x_0 + \hat{c}h) &= \mathcal{O}(h^{\hat{q}+1}), \\ Z - z(x_0 + \hat{c}h) &= \mathcal{O}(h^{\hat{q}}). \end{aligned}$$

Dabei bezeichne Y, Z die lokale Lösung. Mit Hilfe der Stufenwerte berechnen wir anschließend Näherungen der Nordsieckvektoren $y_c(x_0, h)$ und des entsprechend definierten Vektors $z_c(x_0, h)$ wie folgt:

$$\begin{aligned} y_c(x_0, h) &= e_1 \otimes y_0 + h(\hat{B} \otimes I) f(x_0 + \hat{c}h, Y, Z) + \mathcal{O}(h^{\hat{p}_1+1}), \\ z_c(x_0, h) &= e_1 \otimes z_0 + h(\hat{B} \otimes I) Z' + \mathcal{O}(h^{\hat{p}_2+1}) \\ &= (e_1 - \hat{B}\hat{A}^{-1}\mathbb{1}) \otimes z_0 + (\hat{B}\hat{A}^{-1} \otimes I) Z + \mathcal{O}(h^{\hat{p}_2+1}). \end{aligned}$$

Dabei sind $\hat{p}_i$ vorgegebene Ordnungen. Wieder kann die erste Zeile von $\hat{B}$ aus lauter Nullen bestehen.
An dieser Stelle gehen wir nochmals auf die Berechnung der zweiten Komponente von $y_c(x_0, h)$ ein. Sie ist aufgrund der Differentialgleichung der DAE (4.1) gegeben durch

$$hy'(x_0) = hf(y_0, z_0).$$

Natürlich kann sie auf diese Weise berechnet werden. Das Problem ist jedoch, dass die entsprechende Berechnung von $hz'(x_0)$ nicht möglich ist. Für die algebraische Variable liegt jedoch explizit keine Differentialgleichung vor. Diese Unbestimmtheit wird auch durch die Wahl $c_1 = 0$ und $e_1^T \hat{A} = 0$ deutlich: Setzen wir also den ersten Stufenwert Y_1, Z_1 mit dem Anfangswert y_0, z_0 gleich, so können bei geeigneter Wahl der restlichen Koeffizienten von $\hat{A}$ die weiteren Stufenwerte eindeutig bestimmt werden (vgl. [HLR89] S.46 zu den Lobatto IIIA Methoden), aber Z' bleibt weiterhin

uneindeutig. Somit ist die Berechnung von $z_c(x_0, h)$ nicht geklärt. Wir gehen daher im Folgenden von einer invertierbaren Matrix $\hat{A}$ aus und berechnen die Stufenwerte Y_1, Z_1 ebenfalls über das Gleichungssystem der Stufenwerte ohne sie mit y_0, z_0 zu identifizieren.

Wir definieren

$$\begin{aligned} \delta_0^y &:= e_1 \otimes y_0 + h(\hat{B} \otimes I) f(x_0 + \hat{c}h, Y, Z) - y_c(x_0, h), \\ \delta_0^z &:= (e_1 - \hat{B}\hat{A}^{-1}\mathbb{1}) \otimes z_0 + (\hat{B}\hat{A}^{-1} \otimes I) Z - z_c(x_0, h) \end{aligned}$$

und finden analog zu Satz 3.22 die Aussage:

Lemma 4.1 *Besitzt die Runge-Kutta ähnliche Startprozedur mit invertierbarer Verfahrensmatrix $\hat{A}$ die Stufenordnung $\hat{q}$ und die Ordnung $\hat{p} \geq \hat{q}$ für gewöhnliche Differentialgleichungen, so gilt für diese Startprozedur angewendet auf Index-2 DAEs in Hessenberg Form*

$$\begin{aligned} \delta_0^y &= \mathcal{O}(h^{\hat{q}+1}), & P(x_0)\delta_0^y &= \mathcal{O}(h^{\min\{\hat{p}+1, \hat{q}+2\}}), \\ \delta_0^z &= \mathcal{O}(h^{\hat{q}}). \end{aligned}$$

Beweis: Der Beweis wird analog zum Beweis des Satzes 3.22 geführt.

□

Wir leiten nun *singly implicit* Runge-Kutta ähnliche Startprozeduren mit $\hat{p} = \hat{q} = \hat{s} - 1$ für gewöhnliche Differentialgleichungen her. Wenden wir diese Startprozedur auf Index-2 DAEs in Hessenberg Form an, so erhalten wir nach Lemma 4.1 Startwerte der Ordnung $\hat{p}+1$ in y und der Ordnung $\hat{p}$ in z. Wir definieren zu dieser Herleitung die Shiftmatrizen

$$K = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & 1 \\ 0 & \cdots & \cdots & \cdots & 0 \end{pmatrix} \in \mathbb{R}^{\hat{s},\hat{s}}, \qquad J = K^T$$

und für einen gegebenen Knotenvektor $\hat{c}$ die skalierte Vandermondematrix

$$\hat{C} = \begin{pmatrix} 1 & \hat{c}_1 & \frac{\hat{c}_1^2}{2!} & \cdots & \frac{\hat{c}_1^{\hat{s}-1}}{(\hat{s}-1)!} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \hat{c}_{\hat{s}} & \frac{\hat{c}_{\hat{s}}^2}{2!} & \cdots & \frac{\hat{c}_{\hat{s}}^{\hat{s}-1}}{(\hat{s}-1)!} \end{pmatrix} \in \mathbb{R}^{\hat{s},\hat{s}}.$$

Eine Runge-Kutta ähnliche Startprozedur besitzt genau dann die Ordnung $\hat{p} = \hat{q}$, wenn für die Matrizen $\hat{A}$ und $\hat{B}$ folgende Gleichungen gelten

$$\begin{aligned} \mathbb{1} \cdot e_1^T &= \hat{C} - \hat{A}\hat{C}K, \\ e_1 \cdot e_1^T &= I - \hat{B}\hat{C}K \end{aligned} \tag{4.3}$$

(vgl. [Wright02] S.108).

Wright leitet Startprozeduren her, indem er

$$\begin{aligned} \hat{A}\hat{C} &= \hat{C}J, \\ \hat{B}\hat{C} &= J \end{aligned}$$

löst (vgl. ebd.). Dies ist für paarweise verschiedene Knoten $\hat{c}_i$ möglich, da $\hat{C}$ in diesem Fall invertierbar ist. Im Folgenden gehen wir von dieser Invertierbarkeit aus. Eine einfache Rechnung zeigt, dass die auf diese Weise bestimmten Matrizen $\hat{A}$ und $\hat{B}$ tatsächlich (4.3) erfüllen. Das Problem dabei ist jedoch, dass mit J auch $\hat{A}$ einen eindimensionalen Kern besitzt, also insbesondere nicht invertierbar ist. Für Index-2 DAEs ist dies, wie wir bereits oben erwähnten, nicht erwünscht.

Die Idee besteht nun darin, eine Matrix L mit der Eigenschaft

$$L\hat{C}K = 0 \tag{4.4}$$

zu $\hat{C}J\hat{C}^{-1}$ zu addieren, so dass

$$\hat{A} := \hat{C}J\hat{C}^{-1} + L$$

invertierbar ist und ein einpunktiges Spektrum besitzt. Die Eigenschaft (4.4) garantiert, dass $\hat{A}$ ebenfalls die erste Gleichung in (4.3) erfüllt.

Wie ist nun L zu berechnen? Es gilt äquivalent zu oben

$$\hat{A} = \hat{C}(J + \hat{C}^{-1}L\hat{C})\hat{C}^{-1}.$$

Besitzt nun die Matrix in der Klammer ein Einpunktspektrum ungleich Null, so können wir aufgrund der Ähnlichkeit der Matrizen dies auch für $\hat{A}$ garantieren. Eigenschaft (4.4) liefert

$$L\hat{C} = l \cdot e_s^T$$

für einen Vektor $l \in \mathbb{R}^s$. Dies ist offensichtlich, da $e_{\hat{s}}$ im Kern von $K^T = J$ liegt. Also gilt

$$\tilde{J} := J + \hat{C}^{-1}L\hat{C} = \begin{pmatrix} 0 & \cdots & \cdots & 0 & a_1 \\ 1 & \ddots & & \vdots & \vdots \\ 0 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & 1 & a_{\hat{s}} \end{pmatrix}$$

mit $\hat{C}a = l$. Wir haben das Problem nun auf die Bestimmung des Vektors a zurückgeführt. Kennen wir a, so lässt sich l und mit der Invertierbarkeit von $\hat{C}$ auch die Matrix L berechnen.
Wie ist der Vektor a bestimmt, wenn wir für $\tilde{J}$ ein einpunktiges Spektrum $\sigma(\tilde{J}) = \{\mu\}$ für ein $\mu \neq 0$ fordern? Ein Koeffizientenvergleich von

$$\det(\tilde{J} - \lambda I) = (-\lambda + \mu)^{\hat{s}}$$

liefert

$$a_i = (-1)^{\hat{s}-i} \begin{pmatrix} \hat{s} \\ i-1 \end{pmatrix} \mu^{\hat{s}+1-i}.$$

Für vorgegebene $\hat{s}$, Knotenvektor $\hat{c} \in \mathbb{R}^{\hat{s}}$ und Eigenwert $\mu \neq 0$ lassen sich $\hat{A}$ und $\hat{B}$ also durch

$$\begin{aligned} \hat{A} &= \hat{C}J\hat{C}^{-1} + L, \\ \hat{B} &= J\hat{C}^{-1} \end{aligned}$$

mit

$$L = \hat{C} \cdot l \cdot e_{\hat{s}}^T \cdot \hat{C}^{-1}$$

bestimmen. In Anhang B ist ein Matlabprogramm zur Berechnung beliebiger *singly implicit* Runge-Kutta ähnlicher Startprozeduren mit $\hat{p} = \hat{q} = \hat{s} - 1$ angegeben.

Beispiel 4.2

(i) *Die Koeffizienten für die Parameter* $\hat{p} = \hat{q} = 1 = \hat{s} - 1$, $\mu = 1$ *und* $c = [0\ 1]$ *lauten:*

$$\left[\begin{array}{c|c} \hat{c} & \hat{A} \\ \hline e_1 & \hat{B} \end{array}\right] = \left[\begin{array}{c|cc} 0 & 1 & -1 \\ 1 & 0 & 1 \\ \hline 1 & 0 & 0 \\ 0 & 1 & 0 \end{array}\right].$$

(ii) *Die Koeffizienten für die Parameter* $\hat{p} = \hat{q} = 2 = \hat{s} - 1$, $\mu = 1$ *und* $c = [0\ \frac{1}{2}\ 1]$ *lauten:*

$$\left[\begin{array}{c|c} \hat{c} & \hat{A} \\ \hline e_1 & \hat{B} \end{array}\right] = \left[\begin{array}{c|ccc} 0 & 4 & -8 & 4 \\ \frac{1}{2} & -\frac{3}{8} & \frac{3}{2} & -\frac{5}{8} \\ 1 & -\frac{5}{2} & 6 & -\frac{5}{2} \\ \hline 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -3 & 4 & -1 \end{array}\right].$$

Beispiel 4.3 *Die Koeffizienten für die Parameter* $\hat{p} = \hat{q} = 3 = \hat{s} - 1$, $\mu = 1$ *und* $c = [0\ \frac{1}{3}\ \frac{2}{3}\ 1]$ *lauten:*

$$\left[\begin{array}{c|c} \hat{c} & \hat{A} \\ \hline e_1 & \hat{B} \end{array}\right] = \left[\begin{array}{c|cccc} 0 & 27 & -81 & 81 & -27 \\ \frac{1}{3} & -\frac{19}{36} & \frac{20}{9} & -\frac{73}{36} & \frac{2}{3} \\ \frac{2}{3} & -14 & \frac{385}{9} & -\frac{380}{9} & \frac{127}{9} \\ 1 & -\frac{67}{4} & 51 & -\frac{201}{4} & 17 \\ \hline 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & -\frac{11}{2} & 9 & -\frac{9}{2} & 1 \\ 0 & 18 & -45 & 36 & -9 \end{array}\right].$$

Wir werden im nächsten Kapitel über das Newton-Verfahren sehen, dass es sinnvoll ist, die Matrix $\hat{A}$ oder alternativ $\hat{A}^{-1}$ auf "einfache" Gestalt zu transformieren. Dies ist per Konstruktion für die oben hergeleiteten Startprozeduren leicht zu realisieren, da eine Transformationsmatrix für die Jordansche Normalform der Koeffizientenmatrix $\hat{A}$ leicht zu ermitteln ist.

Unabhängig vom Vektor a hat die Matrix $\tilde{J} - \mu I$ und somit auch $\hat{A} - \mu I$ den Rang $\hat{s} - 1$. Daher besitzt der Eigenwert μ der Matrix $\hat{A}$ die algebraische Vielfachheit $\hat{s}$ und die geometrische Vielfachheit 1. Es existiert somit eine Transformationsmatrix T mit

$$T^{-1}\hat{A}T = J + \mu I.$$

Durch eine leichte Rechnung lässt sich mit der Definition der a_i oben zeigen, dass

$$(\tilde{J} - \mu I)^{\hat{s}-1} e_1$$

ungleich dem Nullvektor ist. Aus der Theorie der linearen Algebra wissen wir dann, dass dieses Produkt ein Eigenvektor der Matrix $\tilde{J}$ ist. Der Vektor e_1 ist also ein Hauptvektor der Ordnung $\hat{s}$. Wiederum folgt aus der linearen Algebra, dass dann $Ce_1 = \mathbb{1}$ ein Hauptvektor der Ordnung $\hat{s}$ und

$$(\hat{A} - \mu I)^{\hat{s}-1} \mathbb{1}$$

ein Eigenvektor der Matrix $\hat{A}$ ist. Daher lassen sich die Spalten der Transformationsmatrix leicht berechnen:

$$T^i = (\hat{A} - \mu I)^{i-1} \mathbb{1}.$$

Dabei bezeichne T^i die i-te Spalte der Matrix T.

Mit einer anderen Transformationsmatrix lässt sich natürlich auch $\hat{A}^{-1}$ auf Jordansche Normalform bringen. Die Berechnung der dazu nötigen Transformationsmatrix

lässt sich analog zu oben herleiten. Die entsprechenden Hauptvektoren von $\tilde{J}^{-1}$ und $\hat{A}^{-1}$ lauten nun $e_{\hat{s}}$ und $\hat{C} \cdot e_{\hat{s}}$. Dies ist leicht einzusehen, wenn man

$$\tilde{J}^{-1} = K + a^{-1} \cdot e_1^T$$

mit $a^{-1} \in \mathbb{R}^s$ definiert durch

$$\begin{aligned} a_{\hat{s}}^{-1} &= 1/a_1, \\ a_i^{-1} &= -a_{i+1}/a_1 \end{aligned}$$

für $i = 1, \ldots, \hat{s} - 1$ beachtet.

Bemerkung: *Huang verwendet in ihrer Dissertation singly diagonally implicit Runge-Kutta ähnliche Startprozeduren für allgemeine lineare Verfahren angewendet auf steife Differentialgleichungen (vgl. [Huang05] S. 92ff). Die Matrix $\hat{A}$ hat dabei die folgende Struktur:*

$$\hat{A} = \begin{bmatrix} \lambda & 0 & \ldots & 0 \\ \hat{a}_{21} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \hat{a}_{s1} & \ldots & \hat{a}_{ss-1} & \lambda \end{bmatrix}.$$

Dabei ist λ natürlich ungleich 0. Die Stufenordnung ist tatsächlich nur 1! Die Idee ist nun Folgende: Man berechnet sukzessive Stufenwerte höherer Stufenordnung. Dabei werden bei der Berechnung von Stufenwerten der Stufenordnung q nur solche mit Stufenordnung $q - 1$ verwendet. Dies wird durch Nullen in der Matrix $\hat{A}$ an entsprechender Stelle realisiert. Bei der Berechnung der Näherung von $y_c(x_0, h)$ werden anschließend nur die Stufenwerte maximaler Ordnung verwendet. Dies wirkt sich durch Nullspalten der Matrix $\hat{B}$ aus. Bei diesem sukzessiven Aufbau der Stufenordnung spielt der Faktor h vor der Funktion f in (4.2) eine wesentliche Rolle. Er trägt dazu bei, dass aus der Stufenordnung $q - 1$ die Stufenordnung q werden kann. Dies ist bei Index-2 DAEs in Hessenberg Form nicht möglich, da wir bei den Stufenwerten der z-Komponente eine Ordnung verlieren. Eine solche singly diagonally implicit Runge-Kutta ähnliche Startprozedur besitzt also angewendet auf eine Index-2 DAE in Hessenberg Form tatsächlich in allen Stufenwerten die Ordnung 1! Dies lässt vermuten, dass diese Startprozeduren vom Phänomen der Ordnungsreduktion betroffen sind und sich daher auch für sehr steife Gleichungen nicht eignen.

Wir haben *singly implicit* Runge-Kutta ähnliche Startprozeduren mit den Koeffizienten aus Beispiel 4.3 zum einen auf die gewöhnliche Differentialgleichung von Prothero und Robinson (2.3) mit $\varphi(x) = \sin(x)$ und $x_0 = 0$ angewendet:

$$y' = \lambda(y - \sin(x)) + \cos(x), \qquad y(0) = 0. \tag{4.5}$$

$p = 3,\ s = 4$	$h = 0.01$	$h = 0.001$	$h = 0.0001$
$\lambda = -1.D3$	0.00195724035368	$5.324314297244625D - 4$	0.00738111151562
$\lambda = -1.D4$	0.00149058145125	$3.466439322807839D - 4$	0.06733421195465
$\lambda = -1.D5$	0.00144328606726	$3.003838035462664D - 4$	0.13044832427098
$\lambda = -1.D6$	0.00143855603818	$2.919855007881206D - 4$	0.13485892038968
$\lambda = -1.D7$	0.00143809885579	$2.848948540384958D - 4$	0.14077210542590

Tabelle 4.1: Fehler der *singly implicit* Runge-Kutta ähnlichen Startprozedur aus Beispiel 4.3 angewendet auf die gewöhnliche Differentialgleichung (4.5) geteilt durch h^4 für verschiedene Schrittweiten und Steifheiten.

Bei der Berechnung des Fehlers wurde als Norm die euklidische Norm verwendet, das heißt es gilt für die Einträge in Tabelle 4.1 jeweils:

$$\sqrt{(\varphi(h,0)_2 - h)^2 + \varphi(h,0)_3^2 + (\varphi(h,0)_4 + h^3)^2}/h^4,$$

wobei $\varphi(h, x_0)$ die Startprozedur bezeichnet und der Index die jeweilige Komponente. Dabei ist zu beachten, dass bei den oben hergeleiteten Startprozeduren

$$\varphi(h, x_0)_1 = x_0$$

gilt, also die erste Komponente exakt ist.

Es ist klar erkennbar, dass die Ordnung unabhängig von der Steifheit der Gleichung gewährleistet ist (vgl. Tabelle 4.1). Diese Startprozeduren sind daher auch für steife Differentialgleichungen geeignet.

Zum anderen haben wir die Startwerte für die DAE aus Beispiel 3.2 für $f(x) = \exp(x)$ und $g = 0$ zum Zeitpunkt $x_0 = 0$ berechnet. Ohne die Variable y lautet diese einfache Index-2 DAE:

$$\begin{aligned} u' &= -z, \\ 0 &= u - \exp(x). \end{aligned} \tag{4.6}$$

Die analytische Lösung ist offenbar:

$$u(x) = \exp(-x), \qquad z(x) = \exp(-x).$$

Wiederum wurde die euklidische Norm des Fehler durch entsprechende h-Potenzen dividiert. Dabei ist die Ordnung gemäß Lemma 4.1 sowohl für $s = 3$ als auch für $s = 4$ erkennbar (vgl. Tabelle 4.2 und 4.3).

$p = 2,\ s = 3$	Fehler in u/h^3	Fehler in z/h^2
$h = 0.1$	0.49231346	0.797679266
$h = 0.01$	0.505410213	0.791292722
$h = 0.001$	0.506747973	0.790641688
$h = 0.0001$	0.506535303	0.791094542

Tabelle 4.2: Fehler der Startprozedur aus Beispiel 4.3 angewendet auf die DAE (4.6) sowohl der differentiellen Variable u geteilt durch h^3 als auch der algebraischen Variable z geteilt durch h^2 für verschiedene Schrittweiten.

$p = 3,\ s = 4$	Fehler in u/h^4	Fehler in z/h^3
$h = 0.1$	0.496284148	1.0590095
$h = 0.01$	0.508922019	1.06219195
$h = 0.001$	0.511489801	1.06237163

Tabelle 4.3: Fehler der Startprozedur aus Beispiel 4.3 angewendet auf die DAE (4.6) sowohl der differentiellen Variable u geteilt durch h^4 als auch der algebraischen Variable z geteilt durch h^3 für verschiedene Schrittweiten.

Die verwendete Startprozedur für Index-2 DAEs in Hessenberg Form der Ordnung 2 mit $s = 3$ Stufen ist in Anhang B als Fortran-Programm angegeben.

4.2 Vereinfachtes Newton-Verfahren

In diesem Unterkapitel untersuchen wir die Anwendung des vereinfachten Newton-Verfahrens auf das Gleichungssystem der Stufenwerte. Dieses Gleichungssystem eines allgemeinen linearen Verfahrens angewendet auf eine Index-2 DAE in Hessenberg Form ist gegeben durch

$$\begin{aligned} 0 &= Y - Uy^{[n]} - hAf(x_n + ch, Y, Z) \in \mathbb{R}^{sN}, \\ 0 &= g(x_n + ch, Y) \qquad\qquad\qquad\quad \in \mathbb{R}^{sl}. \end{aligned}$$

Wir haben in dieser Darstellung auf das Kroneckerprodukt der Matrizen U und A mit der Identität verzichtet. Eine Anwendung des vereinfachten Newton-Verfahrens benötigt in jeder Iteration die Lösung eines linearen Gleichungssystems mit der Matrix

$$DF := \begin{pmatrix} I - hA\frac{\partial f}{\partial Y}(x_n + ch, Uy^{[n]}, Zy^{[n]}) & -hA\frac{\partial f}{\partial Z}(x_n + ch, Uy^{[n]}, Zy^{[n]}) \\ \\ \frac{\partial g}{\partial Y}(x_n + ch, Uy^{[n]}) & 0 \end{pmatrix}.$$

Das vereinfachte Newton-Verfahren lautet mit diesen Bezeichnungen

$$\begin{aligned} DF\Delta \begin{pmatrix} Y \\ Z \end{pmatrix}_{k+1} &= -\begin{pmatrix} Y_k - Uy^{[n]} \\ 0 \end{pmatrix} + \begin{pmatrix} hAf(x_n + ch, Y_k, Z_k) \\ -g(x_n + ch, Y_k) \end{pmatrix}, \\ \begin{pmatrix} Y_{k+1} \\ Z_{k+1} \end{pmatrix} &= \begin{pmatrix} Y_k \\ Z_k \end{pmatrix} + \Delta \begin{pmatrix} Y \\ Z \end{pmatrix}_{k+1}. \end{aligned}$$

Die Struktur der Matrix DF ist entscheidend für den Aufwand zum Lösen des linearen Gleichungssystems. Die partiellen Ableitungen in der Matrix DF haben mehr Struktur, als die Darstellung oben vermuten lässt. Zum Beispiel gilt

$$\frac{\partial g}{\partial Y}(x_n + ch, Uy^{[n]}) = \begin{pmatrix} \frac{\partial g}{\partial y}^1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \frac{\partial g}{\partial y}^s \end{pmatrix}$$

mit

$$\frac{\partial g}{\partial y}^i = \frac{\partial g}{\partial y}(x_n + c_i h, U_i y^{[n]}).$$

Dabei bezeichne U_i die i-te Zeile der Matrix U. Dieselbe Form besitzen die partiellen Ableitungen von f. Diese Struktur wird im Allgemeinen durch die Multiplikation mit der Matrix A in DF zerstört.

Viel bessere Strukturen der entsprechend definierten Matrix DF sind möglich für eine äquivalente Formulierung des Gleichungssystems der Stufenwerte:

$$\begin{aligned} 0 &= Y_1 - (U_1 \otimes I_N) y^{[n]} - h(A_1 \otimes I_N) f(x_n + ch, Y, Z), \\ 0 &= -h(A_1 \otimes I_l) g(x_n + ch, Y), \\ &\vdots \\ 0 &= Y_s - (U_s \otimes I_N) y^{[n]} - h(A_s \otimes I_N) f(x_n + ch, Y, Z), \\ 0 &= -h(A_s \otimes I_l) g(x_n + ch, Y). \end{aligned}$$

Hier bezeichnet der Index an den Matrizen A und U jeweils die entsprechende Zeile. Die Multiplikationen der "algebraischen Gleichung" $0 = g(x_n+ch, Y)$ mit $-h(A_i \otimes I_l)$ sind unproblematisch, da das vereinfachte Newton-Verfahren invariant unter solchen Multiplikationen ist. Bis auf diese Multiplikationen geht diese äquivalente Formulierung nur aus einer Permutation der Gleichungen oben hervor.
Die Matrix des Newton Verfahrens lautet nun

$$DF := (I \otimes M) - h(A \otimes I_{N+l}) J, \tag{4.7}$$

wobei J die folgende Bandstruktur besitzt:

$$J = \begin{pmatrix} J^1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & J^s \end{pmatrix}$$

mit

$$J^i = \begin{pmatrix} \frac{\partial f}{\partial y}^i & \frac{\partial f}{\partial z}^i \\ \frac{\partial g}{\partial y}^i & 0 \end{pmatrix}.$$

Die Massematrix M ist definiert durch

$$M \begin{pmatrix} y' \\ z' \end{pmatrix} = \begin{pmatrix} f(x, y, z) \\ g(x, y) \end{pmatrix}$$

und lautet daher

$$M = \begin{pmatrix} I_N & 0 \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{N+l, N+l}.$$

Wiederum wird durch die Multiplikation der Matrix J mit $(A \otimes I_{N+l})$ in (4.7) die Bandstruktur im Allgemeinen zerstört. Der entscheidende Unterschied ist aber nun, dass durch eine Koordinatentransformation die Matrix A auf einfache Gestalt gebracht werden kann. Dies ist zum Beispiel die Jordansche Normalform. Doch bevor wir dies skizzieren, geben wir das vereinfachte Newton-Verfahren an:

$$\begin{aligned} DF \Delta W_{k+1} &= \\ &\quad -(I \otimes M) \left[W_k - (U \otimes I_{N+l}) \begin{pmatrix} y \\ z \end{pmatrix}^{[n]} \right] + h(A \otimes I_{N+l}) \begin{pmatrix} f \\ g \end{pmatrix} (W_k), \\ W_{k+1} &= W_k + \Delta W_{k+1}. \end{aligned}$$

Hier sind die Supervektoren W, $\begin{pmatrix} y \\ z \end{pmatrix}^{[n]}$ und $\begin{pmatrix} f \\ g \end{pmatrix}(W)$ definiert durch

$$W = \begin{pmatrix} Y_1 \\ Z_1 \\ \vdots \\ Y_s \\ Z_s \end{pmatrix}, \quad \begin{pmatrix} y \\ z \end{pmatrix}^{[n]} = \begin{pmatrix} y_1^{[n]} \\ z_1^{[n]} \\ \vdots \\ y_s^{[n]} \\ z_s^{[n]} \end{pmatrix}, \quad \begin{pmatrix} f \\ g \end{pmatrix}(W) = \begin{pmatrix} f(x_n + c_1 h, Y_1, Z_1) \\ g(x_n + c_1 h, Y_1) \\ \vdots \\ f(x_n + c_s h, Y_s, Z_s) \\ g(x_n + c_s h, Y_s) \end{pmatrix}.$$

Um Rundungsfehler zu vermeiden, führen wir die Variable

$$\bar{W} = W - (U \otimes I_{N+l}) \begin{pmatrix} y \\ z \end{pmatrix}^{[n]}$$

ein (vgl. [HW] S.118 für das Gleichungssystem der Stufenwerte von Runge-Kutta Verfahren). Das vereinfachte Newton-Verfahren lautet nun

$$\begin{aligned} DF\Delta\bar{W}_{k+1} &= -(I \otimes M)\bar{W}_k + h(A \otimes I_{N+l}) \begin{pmatrix} f \\ g \end{pmatrix}(W_k), \qquad (4.8) \\ \bar{W}_{k+1} &= \bar{W}_k + \Delta\bar{W}_{k+1}. \end{aligned}$$

Es ist nun üblich die Gleichung (4.8) von links mit $(T^{-1}A^{-1} \otimes I_{N+l})$ zu multiplizieren, wobei T eine Transformationsmatrix ist, so dass die Matrix

$$T^{-1}A^{-1}T = \Lambda$$

möglichst einfache Gestalt besitzt ([HW] S.121 und [B76]) . Der Vorteil liegt auf der Hand, wenn wir

$$\hat{W} = (T^{-1} \otimes I_{N+l})\bar{W}$$

setzen und die entsprechende Matrix des Newton-Verfahrens in den $\hat{W}$-Koordinaten betrachten:

$$(\Lambda \otimes M) - hJ.$$

Diese Matrix besitzt mit Λ und J Bandstruktur und eignet sich daher gut zum Lösen linearer Gleichungssysteme. Das Newton-Verfahren selbst lautet schließlich

$$\begin{aligned} ((\Lambda \otimes M) - hJ)\Delta\hat{W}_{k+1} &= -(\Lambda \otimes M)\hat{W}_k + h(T^{-1} \otimes I_{N+l}) \begin{pmatrix} f \\ g \end{pmatrix}(W_k), \\ \hat{W}_{k+1} &= \hat{W}_k + \Delta\hat{W}_{k+1}. \end{aligned}$$

Wir haben aufgrund einer besseren Darstellung darauf verzichtet, das Argument der Funktionen f und g ebenfalls durch $\hat{W}$-Koordinaten darzustellen.

Alternativ wäre auch eine Transformation der Matrix A selbst möglich gewesen, d.h wir hätten (4.8) nur mit der entsprechenden Matrix T^{-1} multipliziert. Das Ergebnis

wäre dann eine Matrix des Newton-Verfahrens von der Form (4.7) mit A ersetzt durch Λ mit

$$T^{-1}AT = \Lambda.$$

Man erreicht jedoch bei der Verwendung der Inversen von A eine dünner besetzte Matrix des Newton-Verfahrens, da M in der Regel dünner besetzt ist als die Submatrizen J^i der Matrix J.

Wir haben nun gesehen, wie durch geeignete Formulierung des Gleichungssystems und Transformation der Koordinaten die Matrix des vereinfachten Newton-Verfahrens einfache Gestalt und Struktur bekommt. Diese Struktur reduziert die Kosten beim Lösen des linearen Gleichungssystems des Newton-Verfahrens zum Beispiel durch eine LU-Zerlegung. Um die Kosten noch weiter zu reduzieren, wollen wir die Submatrizen der Matrix J identisch wählen, das heißt nur eine Auswertung der Jacobi-Matrix der rechten Seite der Index-2 DAE investieren. Dies ist für allgemeine lineare Verfahren vom Nordsieck-Typ möglich. Die *correct value* Funktion lautet bei diesen Verfahren

$$y_c(x,h) = \begin{bmatrix} y(x) \\ hy'(x) \\ \vdots \\ h^{r-1}y^{(r-1)}(x) \end{bmatrix}.$$

Insbesondere gilt

$$(U \otimes I_N)y^{[n]} = \mathbb{1} \otimes y_n + \mathcal{O}(h). \tag{4.9}$$

Zur Erinnerung: Der Präkonsistenzvektor u ist bei GLM vom Nordsieck-Typ gleich dem ersten Einheitvektor und die Präkonsistenzbedingung garantiert $Ue_1 = \mathbb{1}$.
In der Darstellung (4.9) ist y_n die erste Komponente des Nordsieckvektors und stellt somit eine Approximation der Lösung $y(x_n)$ dar. Entsprechend gilt dies bei Index-2 DAEs in Hessenberg Form auch für die z-Komponente. Es folgt bei entsprechender Güte dieser Approximationen:

$$J^i = \begin{pmatrix} \frac{\partial f}{\partial y}(x_n, y_n, z_n) & \frac{\partial f}{\partial z}(x_n, y_n, z_n) \\ \frac{\partial g}{\partial y}(x_n, y_n) & 0 \end{pmatrix} + \mathcal{O}(h).$$

Wir vernachlässigen den $\mathcal{O}$-Term und bezeichnen die Matrix rechts ebenfalls mit J. Für die Matrix des vereinfachten Newton-Verfahrens finden wir somit

$$(\Lambda \otimes M) - h(I \otimes J).$$

Diese Überlegungen sind nicht auf GLMs vom Nordsieck-Typ beschränkt. Liefert ein allgemeines lineares Verfahren eine "gute" Näherung y_n von $y(x_n)$, so dass eine Darstellung (4.9) möglich ist, so können die Matrizen J^i durch J ersetzt werden.

Wir betrachten abschließend allgemeine lineare Verfahren mit Verfahrensmatrix

$$A = \begin{bmatrix} \lambda & 0 & \dots & 0 \\ a_{21} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ a_{s1} & \dots & a_{ss-1} & \lambda \end{bmatrix}.$$

Diese *singly diagonally implicit* GLMs weisen einen weiteren großen Vorteil auf: Die Stufenwerte lassen sich sukzessive durch s Gleichungssysteme der Dimension $N + l$ berechnen. Genau aus diesem Grund sind diese Verfahren entwickelt worden. Im Fall von Runge-Kutta Verfahren heißen sie SDIRK Methoden und besitzen die Stufenordnung 1 ! Eine wichtige Klasse von allgemeinen linearen Verfahren mit Stufenordnung $p = q$ oder $p = q + 1$ mit einer Verfahrensmatrix dieser Form sind die Typ-2 DIMSIMs. Für Index-2 DAEs lautet das Gleichungssystem des i-ten Stufenwertes dieser Verfahren:

$$\begin{aligned} 0 &= Y_i - \eta_i - h\lambda f(x_0 + c_i h, Y_i, Z_i), \\ 0 &= g(x_0 + c_i h, Y_i) \end{aligned} \tag{4.10}$$

mit

$$\eta_i := (U_i \otimes I_N) y^{[n]} + h \sum_{j=1}^{i-1} a_{ij} f(x_n + c_j h, Y_j, Z_j).$$

Man beachte, dass in der Summe nur bereits berechnete Stufenwerte auftreten. Lässt sich diese sukzessive Berechnung denn wirklich durchführen?
Für allgemeine lineare Verfahren der Ordnung 1, die eine Näherung

$$y_n = y(x_n) + \mathcal{O}(h)$$

mit einer Darstellung (4.9) liefern, ist diese Berechnung durchführbar:
Mit Satz 3.7 lässt sich die lokale Lösbarkeit von (4.10) bei $(\eta_i, (U_i \otimes I_l) z^{[n]})$ mit

$$Y_i - \eta_i = \mathcal{O}(h), \qquad Z_i - (U_i \otimes I_l) z^{[n]} = \mathcal{O}(h) \tag{4.11}$$

sukzessive beweisen: Wir gehen dabei induktiv vor und betrachten daher zunächst den Fall $k = 1$. Es gilt

$$\eta_1 = (U_1 \otimes I_N) y^{[n]}.$$

Mit den Überlegungen aus Kapitel 3 Abschnitt 3.2.3 liegen für ein allgemeines lineares Verfahren der Ordnung 1 die Näherungen hinreichend nahe der Lösung, so dass Satz 3.7 angewendet werden kann. Tatsächlich müssen wir streng genommen für die Anfangswerte

$$Q_0(\varphi^y(h, y_0, z_0) - y_c(x_0, h)) = \mathcal{O}(h^2)$$

garantieren. Diese Bedingung und auch die Voraussetzung der Ordnung 1 sind für alle in Betracht kommenden Verfahren erfüllt.

Somit ist das Gleichungssystem (4.10) für $i = 1$ nach Satz 3.7 für hinreichend kleine h lokal bei $(\eta_1, (U_1 \otimes I_l)z^{[n]})$ lösbar. Zusätzlich folgen aus dem Satz die Darstellungen von (4.11) für $i = 1$.
Wir nehmen im Folgenden an, dass das Gleichungssystem (4.10) für $i = k$ lokal bei $(\eta_k, (U_k \otimes I_l)z^{[n]})$ lösbar ist und dass für die Lösungen die Darstellungen in (4.11) mit $i = k$ gelten. Wir wollen zeigen, dass dies dann auch entsprechend für $k = i + 1$ gilt.
Der Induktionschluss wäre durch eine Anwendung von Satz 3.7 vollständig bewiesen. Die beiden ersten Bedingungen der Voraussetzungen (3.13) des Satzes sind mit

$$\eta_k - (U_k \otimes I_N)y^{[n]} = \mathcal{O}(h)$$

und der Bemerkung 3.8 (iii) offenbar erfüllt. Die dritte und letzte Bedingung in (3.13) folgt aus den Gleichungen (ohne das Kronecker-Produkt)

$$\begin{aligned}
g(x_n, \eta_k) &= g(x_n, U_k y^{[n]}) + h \sum_{j=1}^{i} a_{ij} \frac{\partial g}{\partial y}(x_n, U_k y^{[n]}) f(x_n + c_j h, Y_j, Z_j) + \mathcal{O}(h^2) \\
&= g(x_n, U_k y^{[n]}) + h \sum_{j=1}^{i} a_{ij} (I \otimes \frac{\partial g}{\partial y}(x_n, y_n)) f(x_n, \eta_j, U_j z^{[n]}) + \mathcal{O}(h^2) \\
&= g(x_n, U_k y^{[n]}) + h \sum_{j=1}^{i} a_{ij} (I \otimes \frac{\partial g}{\partial y}(x_n, y_n) f(x_n, y_n, z_n)) + \mathcal{O}(h^2) \\
&= g(x_n, U_k y^{[n]}) + h \sum_{j=1}^{i} a_{ij} (I \otimes \underbrace{\frac{\partial g}{\partial y}(x_n, y(x_n)) f(x_n, y(x_n), z(x_n)))}_{=0} + \mathcal{O}(h^2) \\
&= g(x_n, U_k y^{[n]}) + \mathcal{O}(h^2).
\end{aligned}$$

Die Kriterien zum Stoppen des Newton Verfahrens von Kapitel IV.8 des Buchs „Solving Ordinary Differential Equation II“ von Hairer und Wanner können generell für allgemeine lineare Verfahren übernommen werden. Es ist jedoch zu bemerken, dass bei einer Verkleinerung der Schrittweite h aufgrund einer schlechten Konvergenz, die sukzessive Berechnung der Stufenwerte bei *singly diagonally implicit* GLMs erneut gestartet werden muss!

4.3 Beispiele

In diesem Unterkapitel präsentieren wir numerische Resultate von allgemeinen linearen Verfahren der Ordnung 1,2 und 3 angewendet auf verschiedene Differential-Algebraische Gleichungen. Dabei liegt der Schwerpunkt im Nachweis der Konvergenzresultate aus Abschnitt 3.2.3 bzw. des Unterkapitels 3.3. Konkret wurden Typ-2 DIMSIMs realisiert (vgl. zur Einteilung der *Diagonally Implicit Multistage Integration Methods* [B]). Insbesondere besitzt also die Koeffizientenmatrix A untere Dreiecksstruktur und nur einen einzigen Eigenwert.

Verfahren 1 *(vgl. [Huang05]) Das zweistufige und zweiwertige Verfahren besitzt sowohl die Ordnung als auch die Stufenordnung 1:*

$$\begin{aligned} p &= 1, \\ q &= 1, \\ r &= 2, \\ s &= 2. \end{aligned}$$

Der Knotenvektor und die Koeffizientenmatrizen sind gegeben durch:

$$c = \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix}, \qquad \left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|cc} \frac{3}{10} & 0 & 1 & \frac{1}{5} \\ \frac{21}{50} & \frac{3}{10} & 1 & \frac{7}{25} \\ \hline \frac{21}{50} & \frac{3}{10} & 1 & \frac{7}{25} \\ 0 & 1 & 0 & 0 \end{array}\right].$$

Verfahren 2 *(vgl. [Wright02]) Das Verfahren besitzt die Ordnung und die Stufenordnung 2. Es werden $s = 3$ innere und $r = 3$ äußere Stufen berechnet:*

$$\begin{aligned} p &= 2, \\ q &= 2, \\ r &= 3, \\ s &= 3. \end{aligned}$$

Der Knotenvektor und die Koeffizientenmatrizen sind gegeben durch:

$$c = \begin{bmatrix} 0 \\ \frac{1}{2} \\ 1 \end{bmatrix}, \qquad \left[\begin{array}{ccc|ccc} \frac{1}{4} & 0 & 0 & 1 & -\frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 1 & 0 & \frac{1}{8} \\ \hline \frac{1}{2} & -\frac{1}{8} & \frac{1}{2} & 1 & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{2} & -\frac{1}{2} & 1 & 0 & 0 & \frac{1}{4} \\ 0 & -2 & 2 & 0 & 0 & 0 \end{array}\right].$$

Verfahren 3 *Das folgende Verfahren wurde von Butcher und Podhaisky entwickelt (vgl. [BP06]). Es besitzt die Konvergenz- und Stufenordnung 3. Zudem werden $s = 4$ innere und $r = 4$ äußere Stufen berechnet:*

$$\begin{aligned} p &= 3, \\ q &= 3, \\ r &= 4, \\ s &= 4. \end{aligned}$$

Der Knotenvektor und die Koeffizientenmatrizen sind gegeben durch:

$$c = \begin{bmatrix} \frac{1}{4} \\ \frac{1}{2} \\ \frac{3}{4} \\ 1 \end{bmatrix}, \qquad \left[\begin{array}{cccc|cccc} \frac{9}{40} & 0 & 0 & 0 & 1 & \frac{1}{40} & -\frac{1}{40} & -\frac{17}{3840} \\ \frac{52443}{17200} & \frac{9}{40} & 0 & 0 & 1 & -\frac{47713}{17200} & -\frac{51583}{68800} & -\frac{169369}{1651200} \\ -\frac{1996641}{1350200} & -\frac{387}{1570} & \frac{9}{40} & 0 & 1 & \frac{759579}{337550} & \frac{3269871}{5400800} & \frac{907929}{10801600} \\ -\frac{59481}{40000} & -\frac{189}{800} & \frac{1413}{8000} & \frac{9}{40} & 1 & \frac{46433}{20000} & \frac{50593}{80000} & \frac{9659}{120000} \\ \hline -\frac{59481}{40000} & -\frac{189}{800} & \frac{1413}{8000} & \frac{9}{40} & 1 & \frac{46433}{20000} & \frac{50593}{80000} & \frac{9659}{120000} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -5 & \frac{38}{45} & -\frac{157}{45} & \frac{40}{9} & 0 & \frac{16}{5} & 0 & -\frac{137}{720} \\ \frac{2672}{81} & -\frac{256}{27} & -\frac{2512}{81} & \frac{1600}{81} & 0 & -\frac{992}{81} & 0 & 0 \end{array}\right].$$

Nach Kapitel 3 Abschnitt 3.2.3 besitzen alle diese Verfahren angewendet auf Index-2 DAEs in Hessenberg Form (3.5) sowohl für die differentielle Variable y als auch für die algebraische Variable z die Ordnung p, falls die Startprozedur Startwerte hinreichender Güte liefert (vgl. die Sätze 3.25 und 3.27). Da wir die in Unterkapitel 4.1 hergeleiteten *singly implicit* Runge-Kutta ähnlichen Prozeduren verwenden, kann die Ordnung p gewährleistet werden (vgl. für die Koeffizienten der Startprozeduren Beispiel 4.2 und 4.3). Die Koeffizienten der entsprechenden Startprozedur sind für $\mu = 1$ und äquidistant verteilte Knoten c_i mit $c_1 = 0,\ c_s = 1$ durch das Matlabprogramm in Anhang B berechnet worden.

Als Beispielprogramme sind Fortran-Codes einer Startprozedur der Ordnung 2 und des Verfahrens 2 in Anhang B angegeben. Alle zur Berechnung der hier präsentierten Daten verwendeten Fortran-Programme sind mit dem GNU Fortran Compiler übersetzt worden. Die Rechnungen wurden auf einem Server Sun Fire 3800 mit einem UltraSparc III-Prozessor durchgeführt. Für das Lösen der Gleichungssysteme des vereinfachten Newton-Verfahrens sind die Subroutinen DGETRF.f und DGETRS.f verwendet worden. Diese Subroutinen sowie alle weiteren für diese Routinen benötigten Programme sind dem LAPACK (Linear Algebra PACKage) entnommen (vgl. [Lapack]).

4.3.1 Das Pendel

Als klassisches Beispiel einer Index-2 DAE in Hessenberg Form dient das Pendel aus Beispiel 1.5. Es lautet in der GGL-Formulierung (vgl. Kapitel 1) :

$$\begin{aligned} \dot{x}_1 &= y_1 - 2x_1\mu, \\ \dot{x}_2 &= y_2 - 2x_2\mu, \\ m\dot{y}_1 &= -2x_1, \lambda \\ m\dot{y}_2 &= -mg - 2x_2\lambda, \\ 0 &= x_1^2 + x_2^2 - l^2, \\ 0 &= 2x_1y_1 + 2x_2y_2. \end{aligned}$$

Für die Berechnungen haben wir die Parameter für die Masse, Länge und Gravitation auf den Wert 1 normiert. Der verwendete Vektor konsistenter Anfangswerte lautet:

$$\begin{bmatrix} x_1(0) \\ x_2(0) \\ y_1(0) \\ y_2(0) \\ \lambda(0) \\ \mu(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{4.12}$$

Die bekannte Dynamik der Position und der Geschwindigkeit des Massepunktes ist in Abbildungen 4.1 verdeutlicht. Des Weiteren sind in Abbildung 4.2 die zeitlichen Entwicklungen der Lagrange-Multiplikatoren beschrieben. Die den Abbildungen zugrunde liegenden Daten wurden mit dem Verfahren 3 für konstante Schrittweite $h = 0.001$ und den Toleranzen $RTOL = ATOL = 1.D - 8$ berechnet.

Zusätzlich haben wir mit dem Code RADAU5 die Lösung des Pendels ebenfalls mit $RTOL = ATOL = 1.D-8$ berechnet. Dabei ist die Konvergenzordnung von diesem Verfahren in den differentiellen Variablen 5 und in den algebraischen Variablen 3. Die Größenordnungen des Lagrange-Multiplikators μ in Abbildung 4.2 und in 4.3 machen deutlich, dass das Verfahren 3 die Lösung von $\mu = 0$ bei gleicher konstanter Schrittweite wesentlich besser löst als das RADAU5-Verfahren, obwohl beide Verfahren für die algebraischen Variablen einer Index-2 DAE in Hessenberg Form die Konvergenzordung 3 besitzen!

Für die Verfahren 1 und 2 haben wir zum Nachweis der entsprechenden Ordnungen eine Referenzlösung mit dem Code RADAU5 für sehr kleine konstante Schrittweite $h = 1.D - 6$ berechnet. Der Fehler wird durch den euklidischen Abstand dieser Lösung von den Approximationen, welche durch die Verfahren berechnet wurden, in den differentiellen Variablen $y = (x_1, x_2, y_1, y_2)$ bzw. in den algebraischen Variablen $z = (\lambda, \mu)$ gemessen. Wir haben beide Verfahren auf die GGL-Formulierung des Pendels mit den konsistenten Anfangswerten in (4.12) angewendet. Für den Endpunkt

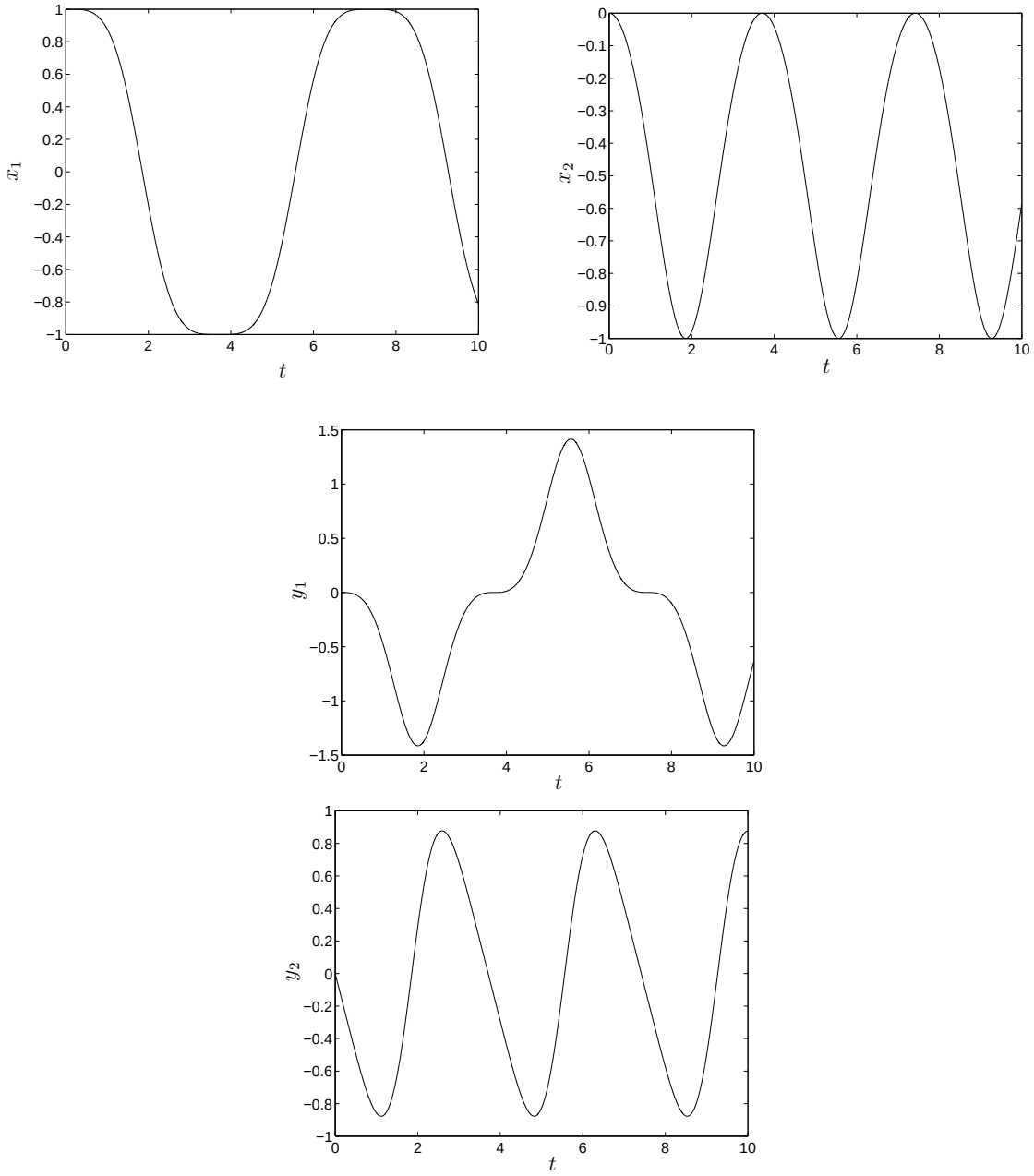

Abbildung 4.1: Zeitliche Entwicklung der Position und der Geschwindigkeit des Pendels berechnet mit dem Verfahren 3 für $h = 0.001$.

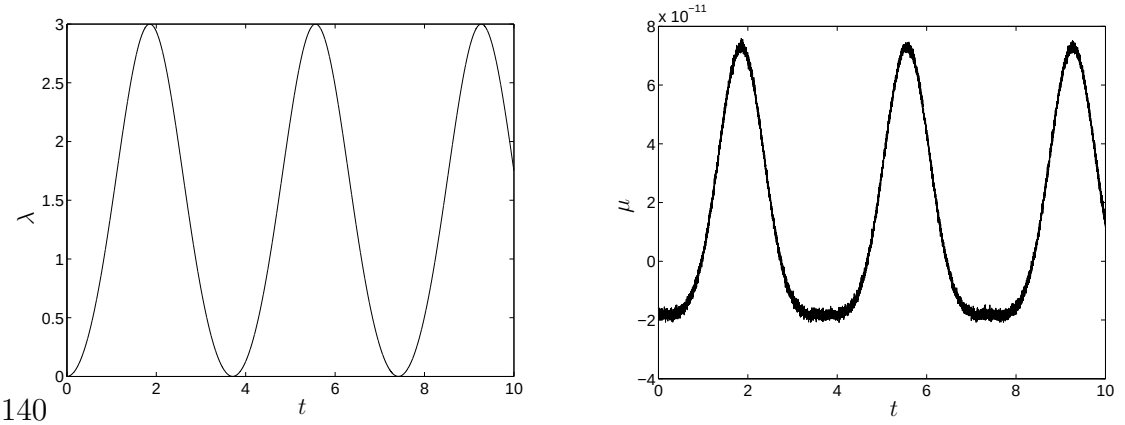

Abbildung 4.2: Zeitliche Entwicklung der Lagrange-Multiplikatoren der GGL-Formulierung des Pendels berechnet mit dem Verfahren 3 für $h = 0.001$:

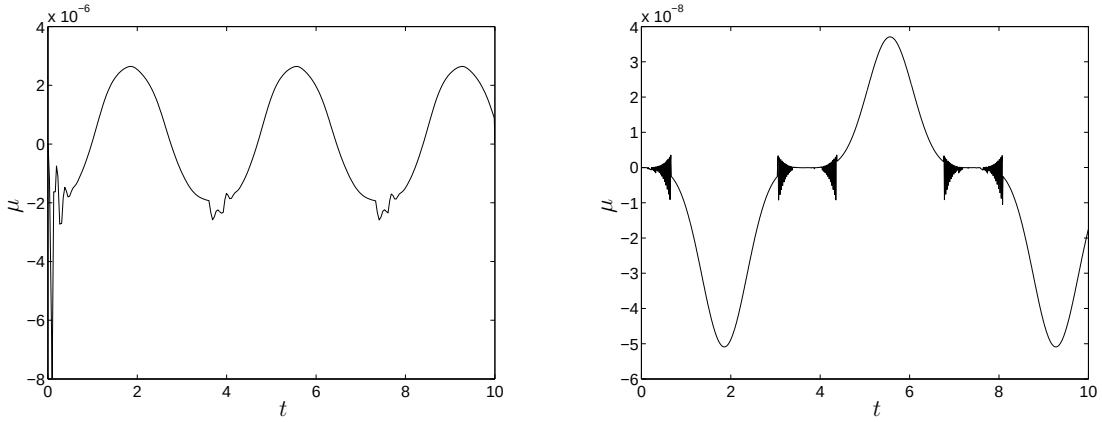

Abbildung 4.3: Zeitliche Entwicklung des Lagrange-Multiplikators μ der GGL-Formulierung des Pendels berechnet mit dem Code RADAU5 (vgl. [HW]): links mit variabler, rechts mit konstanter Schrittweite $h = 0.001$. Dabei ist bei konstanter Schrittweite h aufgrund von starken Oszillationen nur jede zehnte Approximation zum Zeichnen des Graphen verwendet worden.

$p = 1,\ s = 2$	Fehler in y/h	Fehler in z/h
$h = 0.001$	0.0131250118	0.161654503
$h = 0.0001$	0.0129863715	0.16130143
$h = 1.D - 5$	0.0114787645	0.154092469
$h = 1.D - 6$	0.00982087178	0.0924549797

Tabelle 4.4: Fehler des Verfahrens 1 angewendet auf die GGL-Formulierung des Pendels sowohl in den differentiellen Variablen $y = (x_1, x_2, y_1, y_2)$ als auch in den algebraischen Variablen $z = (\lambda, \mu)$ geteilt durch h für verschiedene Schrittweiten.

$p = 2,\ s = 3$	Fehler in y/h^2	Fehler in z/h^2
$h = 0.01$	0.09628015837153	4.93576828742608
$h = 0.001$	0.11676032794535	4.83785623124812
$h = 0.0005$	0.18134995560176	4.75594255334421
$h = 0.0001$	2.27816622429148	8.97545925082043

Tabelle 4.5: Fehler des Verfahrens 2 angewendet auf die GGL-Formulierung des Pendels sowohl in den differentiellen Variablen $y = (x_1, x_2, y_1, y_2)$ als auch in den algebraischen Variablen $z = (\lambda, \mu)$ geteilt durch h^2 für verschiedene Schrittweiten.

der Integration gilt $x_e = 1$. Die Berechnungen wurden mit denselben Toleranzen wie oben durchgeführt. Die Ergebnisse sind in den Tabellen 4.4 und 4.5 aufgeführt. Für beide Verfahren ist die in Kapitel 3 Abschnitt 3.2.3 bewiesene Konvergenzordnung sichtbar!

4.3.2 Eine lineare DAE mit properem Hauptterm

Eine wirklich bemerkenswerte Differential-Algebraische Gleichung, wie wir bereits in Kapitel 3 feststellten, ist die lineare DAE aus Beispiel 3.1. Die Schwierigkeiten beim Lösen dieser DAE für gewisse Parameterwerte ist wohl bekannt (vgl. [GP83]). In [HLR89] wird dieses System sogar als Index-2 Problem bezeichnet, bei dem numerische Verfahren scheitern. Die kritischen Parameterwerte sind dabei $-1 < \eta < -0.5$. In Kapitel 3.3 haben wir gesehen, woran dies genau liegt: Der Index des augmentierten System ist 3! Bei der äquivalenten Formulierung aus Beispiel 3.2 erhöhte die Augmentierung den Index nicht. Es handelt sich bei dieser Formulierung um eine DAE mit properem Hauptterm und Traktabilitätsindex 2, wie wir sie in Unterkapitel 3.3 betrachtet haben. Die Augmentierung lautet für $f(x) = exp(x)$ und $g = 0$ mit anderen Bezeichnungen der Variablen als in Beispiel 3.2:

$$\begin{aligned} y' &= z_1, \\ 0 &= z_2 + \eta x z_3 - \exp(-x), \\ 0 &= z_1 + z_3, \\ 0 &= y - z_2 - \eta x z_3. \end{aligned} \tag{4.13}$$

Dies ist eine semi-explizite DAE der Form (3.70) mit Index 2, wobei die partielle Ableitung $\frac{\partial g}{\partial z}$ den konstanten Rang 2 besitzt. Noch wichtiger ist die Beobachtung, dass

$p = 1,\ s = 2$	Fehler in y/h	Fehler in z_2/h	Fehler in z_3/h
$h = 0.001$	$1.09287579D - 11$	0.00934908291	0.00415514796
$h = 0.0001$	$9.49171297D - 10$	0.00933646624	0.00414954045
$h = 1.D - 5$	$5.56929503D - 08$	0.00933459944	0.00414870467
$h = 1.D - 6$	$3.25754701D - 06$	0.0091848254	0.00408178265
$h = 1.D - 7$	0.000648497506	0.0188195987	0.00843632143

Tabelle 4.6: Fehler des Verfahrens 1 angewendet auf die lineare DAE (4.13) sowohl in der differentiellen Variablen y als auch in den algebraischen Variablen z_2 und z_3 geteilt durch h für verschiedene Schrittweiten.

keine Koordinatentransformation, wie sie in Unterkapitel 3.3 angesprochen wurde, nötig ist, um nach den Variablen z_1 und z_2 aufzulösen. Wir erhalten also nicht nur für die differentielle Variable die Konvergenzordnung, wie wir sie in Abschnitt 3.2.3 bewiesen haben, sondern theoretisch auch für alle algebraischen Variablen (vgl. Bemerkung 3.33).

Wir wollen nun nachweisen, dass auch für kritische Parameterwerte der Formulierung aus Beispiel 3.1 die äquivalente DAE (4.13) durch die Verfahren 1-3 gelöst werden kann und dass die in Kapitel 3 bewiesenen Konvergenzordnungen auch praktisch beobachtet werden können. Wie wir oben bereits begründet haben, sollte die Ordnung in allen Variablen gleich p sein.

Wir wählen als konsistente Anfangswerte:

$$\begin{bmatrix} y \\ z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 1 \\ 1 \end{bmatrix}.$$

Die exakte Lösung des Anfangswertproblems ist offenbar gegeben durch:

$$\begin{bmatrix} y(x) \\ z_1(x) \\ z_2(x) \\ z_3(x) \end{bmatrix} = \begin{bmatrix} \exp(x) \\ -\exp(x) \\ (1 - \eta x)\exp(x) \\ \exp(x) \end{bmatrix}.$$

$p = 2,\ s = 3$	Fehler in y/h^2	Fehler in z/h^2
$h = 0.001$	$1.24497912D - 05$	0.23742606
$h = 0.0001$	$1.07379383D - 05$	0.237396994
$h = 1.D - 5$	0.00556929503	0.14999581

Tabelle 4.7: Fehler des Verfahrens 2 angewendet auf die lineare DAE (4.13) sowohl in der differentiellen Variablen y als auch in den algebraischen Variablen $z = (z_1, z_2, z_3)$ geteilt durch h^2 für verschiedene Schrittweiten.

$p = 3,\ s = 4$	Fehler in y/h^3	Fehler in z/h^3
$h = 0.01$	$9.92261828D - 10$	0.000667539357
$h = 0.001$	$1.09287579E - 05$	0.00077720951
$h = 1.D - 4$	0.0949240686	0.34511097

Tabelle 4.8: Fehler des Verfahrens 3 angewendet auf die lineare DAE (4.13) sowohl in der differentiellen Variablen y als auch in den algebraischen Variablen $z = (z_1, z_2, z_3)$ geteilt durch h^3 für verschiedene Schrittweiten.

Als Integrationsintervall haben wir $[0, 3]$ festgelegt, die Toleranzen sind in diesem Fall $RTOL = ATOL = 1.D{-}9$. Der Fehler ist wiederum durch den euklidischen Abstand der Approximation, bzw. von Komponenten der Approximation, zur exakten Lösung definiert. Während die entsprechenden Ordnungen der algebraischen Komponenten schön zu erkennen sind, scheint in der y-Variablen keine entsprechende Konvergenz vorzuliegen (vgl. die Tabellen 4.6 - 4.8). Es sei jedoch bemerkt, dass der Fehler in y trotz Division durch h^p sehr klein ist! Vielleicht ist die Konvergenz dieses Quotienten gegen eine Konstante nur für noch kleinere Schrittweiten h zu beobachten.

4.3.3 Der "Andrews' squeezer Mechanismus"

Der *Andrews' squeezer* Mechanismus ist ein Mehrkörpersystem in der Ebene bestehend aus sieben Körpern, welche durch Gelenke ohne Reibung miteinander verbunden sind (vgl. Abbildung 7.1 in [HW] S.531). Die Bewegungsgleichungen der Index-3

Formulierung sind von der bekannten Form

$$\begin{aligned} M(q)\ddot{q} &= f(q,\dot{q}) - G^T(q)\lambda, \\ 0 &= g(q). \end{aligned}$$

Dabei repräsentiert der Vektor $q(t) \in \mathbb{R}^7$ sieben verschiedene Winkel des Systems und der Vektor $\lambda \in \mathbb{R}^6$ die Lagrange-Multiplikatoren (vgl. für eine detaillierte Beschreibung [HW, BTS]). Die augmentierte Formulierung auf dem Geschwindigkeitslevel lautet nach Beispiel 3.28 wie folgt:

$$\begin{aligned} \dot{q} &= v, \\ \dot{v} &= w, \\ 0 &= M(q)w - f(t,q,v) + G(q)^T\lambda, \\ 0 &= G(q)v. \end{aligned} \tag{4.14}$$

Die Dimension dieser Index-2 DAE ist 27. Sie besteht aus 14 Differential- und 13 algebraischen Gleichungen.

Die Anfangswerte sind bis auf die folgenden Komponenten gleich Null:

$$\begin{aligned} q_1(0) &= -0.06171389001427644963589484580 01D0 \\ q_3(0) &= 0.45527981916307038025591238244 9D0 \\ q_4(0) &= 0.22266839016588588467447318560 9D0 \\ q_5(0) &= 0.48736497954384255022559895353 0D0 \\ q_6(0) &= -0.22266839016588588467447318560 9D0 \\ q_7(0) &= 1.23054744454982119249735015568D0 \\ w_1(0) &= 14222.44391995411387059116258 87D0 \\ w_2(0) &= -10666.83293996558540294337194 15D0 \\ \lambda_1(0) &= 98.56687039624108960576549821 70D0 \\ \lambda_2(0) &= -6.12268834425566265503114393122D0 \end{aligned}$$

Wir haben das Verfahren 1 auf dieses Anfangswertproblem mit $RTOL = ATOL = 1.D-2$ und der Schrittweite $h = 1.D-8$ angewendet. Der Endpunkt der Integration ist wie üblich 0.03. Als Approximation der Jacobi-Matrix der rechten Seite haben wir die folgende Matrix verwendet (vgl. [HW] S540):

$$\begin{pmatrix} 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & M & G^T \\ 0 & G & 0 & 0 \end{pmatrix}.$$

Die Dynamik der verschiedenen Winkel ist in Abbildung 4.4 aufgetragen.

Fazit der numerischen Berechnungen: Die Konvergenzordnung allgemeiner linearer Verfahren, welche wir theoretisch in Kapitel 3 sowohl in Abschnitt 3.2.3 als auch in Unterkapitel 3.3 bewiesen haben, lassen sich auch praktisch nachweisen. Beim Vergleich des RADAU5 Verfahrens mit konstanter Schrittweite und des Verfahrens 3, welcher nur in den algebraischen Variablen möglich ist, lieferte das Verfahren 3 bessere Approximationen. Dabei sei ausdrücklich darauf hingewiesen, dass wir hier nicht den Code RADAU5 als Maß gewählt haben, sondern aufgrund eines fairen Vergleichs das RADAU5-Verfahren mit konstanter Schrittweite ohne jegliche Optimierung. Ebenso sind im Fortran-Code des Verfahrens 3 neben dem eigentlichen Algorithmus keine weiteren Optimierungen implementiert worden (vgl. das Beispielprogramm in Anhang B). Zudem konnte auch das nicht-triviale Beispiel des *Andrews' squeezer* durch DIMSIMs vom Typ-2, wenn auch mit sehr kleinen Schrittweiten, gelöst werden. Die Lösung einer steifen Formulierung des Mechanismus konnte jedoch mit den vergleichsweise einfachen Implementierungen der Verfahren nicht zufrieden stellend berechnet werden (vgl. zur steifen Formulierung des Mechanismus [HW] VII.7). Die Realisierung einer variablen Schrittweite und weiterer Optimierungen scheinen unabdingbar zu sein. Insgesamt wird aber deutlich, dass sich zum Beispiel Typ-2 DIMSIMS zum Lösen Differential-Algebraischer Gleichungen eignen.

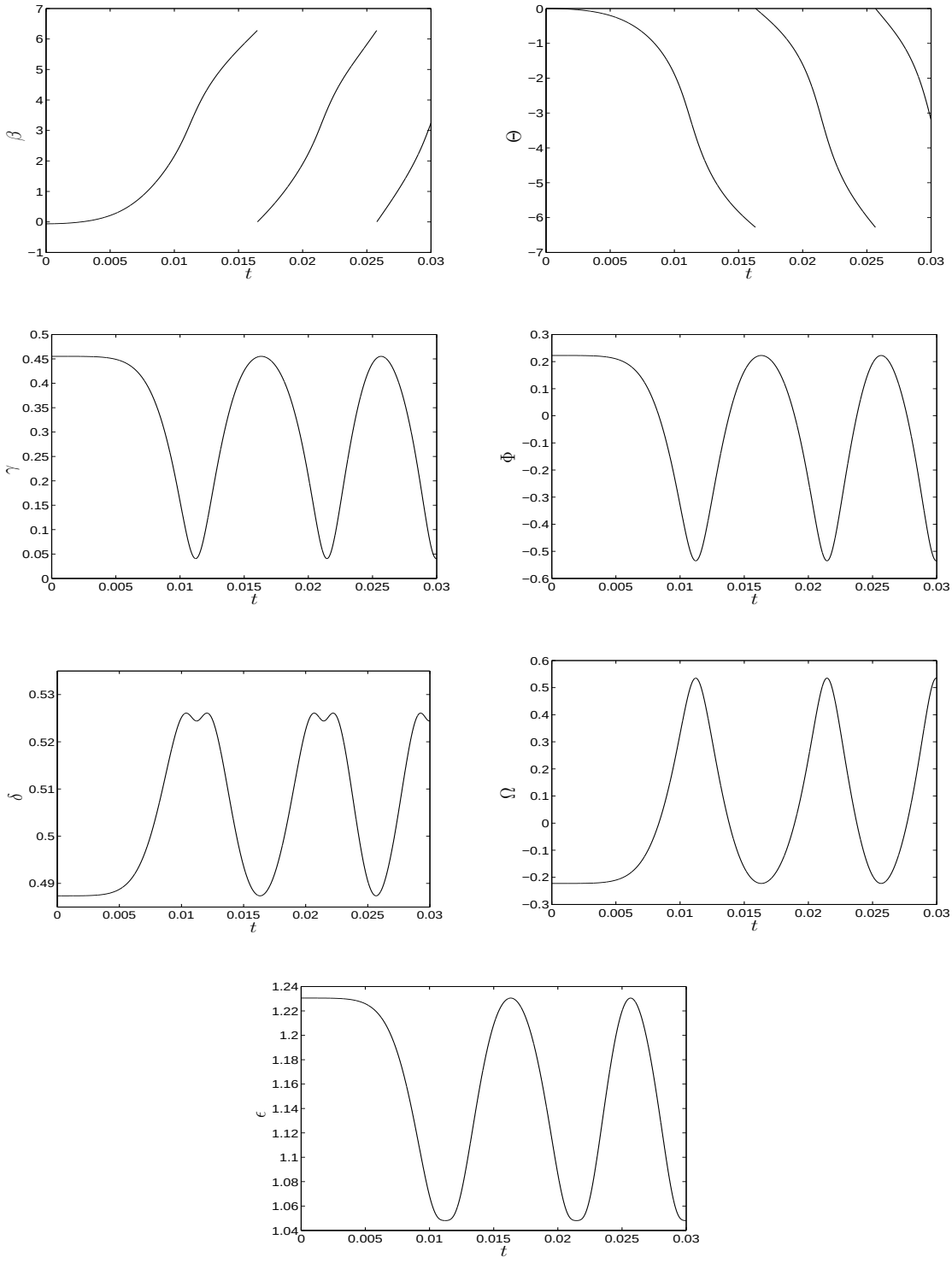

Abbildung 4.4: Zeitliche Entwicklung der Winkel des *Andrews' squeezer* Mechanismus berechnet mit dem Verfahren 1 für $h = 1.D - 8$. Es gilt $q_1 = \beta,\ q_2 = \Theta,\ q_3 = \gamma,\ q_4 = \Phi,\ q_5 = \delta,\ q_6 = \Omega,\ q_7 = \epsilon$.

A DAEs mit properem Hauptterm

Wir betrachten die lineare DAE

$$A(t)[D(t)x(t)]' + B(t)x(t) = q(t), \quad t \in \mathcal{I} \tag{A.1}$$

mit matrixwertigen Funktionen $A \in C(\mathcal{I}, \mathbb{R}^{m,n})$, $D \in C(\mathcal{I}, \mathbb{R}^{n,m})$ und $B \in C(\mathcal{I}, \mathbb{R}^{m,m})$. Der Hauptterm dieser DAE sei proper formuliert (vgl. [M02b]):

Definition A.1 *Der Hauptterm von* (A.1) *ist proper formuliert, wenn der Kern der Koeffizientenmatrizen $A(t)$ und das Bild von $D(t)$ in der direkten Sumnme den gesamten $\mathbb{R}^n$ bilden,*

$$\ker A(t) \oplus \mathrm{im} D(t) = \mathbb{R}^n,$$

und ein stetig differenzierbarer Projektor $R(t) \in C(\mathcal{I}, \mathbb{R}^{n,n})$ existiert mit

$$\ker A(t) = \ker R(t), \qquad \mathrm{im} D(t) = \mathrm{im} R(t), \quad t \in \mathcal{I}.$$

Folgerungen (vgl. [M02b]):

(i) Die Koeffizientenmatrizen $A(t)$ und $D(t)$ besitzen einen gemeinsamen konstanten Rang.

(ii) Es existiert eine verallgemeinerte Inverse von D mit

$$\begin{array}{rclcrcl} DD^- &=& R, & & DD^-D &=& D, \\ D^-D &=& P_0, & & D^-DD^- &=& D^-, \end{array} \tag{A.2}$$

wobei $P_0 = I - Q_0$ gilt und Q_0 eine beliebige Projektion auf den Kern von AD ist. Wir haben dabei die Abhängigkeit von t vernachlässigt.

(iii) Es gilt (ohne das Argument t):

$$DQ_0 = 0, \qquad Q_0 D^- = 0. \tag{A.3}$$

Lemma A.2 *Besitze die DAE* (A.1) *einen proper formulierten Hauptterm und sei der Traktabilitätsindex dieser DAE 2 (vgl. zur Definition dieses Indexes [M02a]). Dann besitzt die Matrix*

$$M(t) := \begin{pmatrix} B(t) & A(t) \\ -D(t) & 0 \end{pmatrix}$$

konstanten Rang.

Beweis: Aus der Definition des Traktabilitätsindexes folgt:

$$G_1(t) := A(t)D(t) + B(t)Q_0(t) \text{ besitzt konstanten Rang } r_1 < m$$

(vgl. [M02b]). Insbesondere besitzt auch $\ker G_1(t)$ konstanten Rang, nämlich $m - r_1$. Wir untersuchen nun den Kern der Matrix $M(t)$ für festes t. Sei $y \in \mathrm{Ker} A(t)$ gegeben. Offenbar gilt:

$$M(t) \begin{pmatrix} 0 \\ y \end{pmatrix} = 0. \tag{A.4}$$

Mit Folgerung (i) oben erhalten wir die Existenz von r linear unabhängigen Vektoren, welche den Kern von $A(t)$ aufspannen. Die Dimension r ist dabei unabhängig von der Zeit t. Wir erweitern diese Vektoren um Nullen, wie in (A.4) geschehen, und erhalten als Spann dieser erweiterten Vektoren den r-dimensionalen Raum $\{0\} \times \ker A(t)$, welcher im Kern der Matrix $M(t)$ liegt. Sei nun

$$\begin{pmatrix} x \\ y \end{pmatrix} \in \ker M(t),$$

wobei wir ohne Einschränkung aufgrund des proper formulierten Hauptterms von $y \in \mathrm{im} D(t)$ ausgehen können. Der Kern von $A(t)$ und das Bild von $D(t)$ bilden ja in der direkten Sumnme den gesamten $\mathbb{R}^n$. Wir definieren

$$z = D^-(t)y + Q_0(t)x. \tag{A.5}$$

Aufgrund der Definition von G_1 und z folgt mit den Folgerungen oben (ohne das Argument t):

$$\begin{aligned} G_1 z &= (AD + BQ_0)(D^- y + Q_0 x) \\ &= ARy + BQ_0 x \\ &= Ay + BQ_0 x. \end{aligned}$$

Zusätzlich haben wir ausgenutzt, dass y im Bild von D liegt, das heißt es gilt:

$$Ry = y.$$

Da Q_0 eine Projektion auf den Kern von $A(t)D(t)$ ist und x nach Voraussetzung im Kern von $D(t)$ liegt, gilt:

$$Q_0 x = x.$$

Somit folgt:

$$\begin{aligned} G_1 z &= Ay + BQ_0 x \\ &= Ay + Bx. \end{aligned}$$

Der Vektor $(x, y)^T$ ist nach Voraussetzung aus dem Kern von $M(t)$. Wir finden daher

$$z \in \ker G_1(t).$$

Seien nun linear unabhängige Vektoren

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} \in \ker M$$

gegeben, wobei y_i jeweils im Bild von $D(t)$ liegt. Zusätzlich seien Vektoren z_i gemäß (A.5) definiert. Sind dann auch die Vektoren z_i linear unabhängig?
Wir betrachten eine Linearkombination dieser Vektoren und zeigen, dass der Nullvektor nur mit der trivialen Linearkombination gebildet werden kann. Nach Definition der z_i gilt

$$\sum \lambda_i z_i = 0 \Longleftrightarrow \sum \lambda_i D^- y_i + \sum \lambda_i Q_0 x_i = 0.$$

Wir multiplizieren die rechte Gleichung jeweils mit DP_0 und Q_0 und erhalten

$$\begin{aligned} \sum \lambda_i D D^- y_i &= 0, \\ \sum \lambda_i Q_0 x_i &= 0. \end{aligned}$$

Hier haben wir die Gleichungen

$$\begin{aligned} Q_0 D^- &= 0, & \qquad P_0 D^- &= D^-, \\ Q_0^2 &= Q_0, & \qquad P_0 Q_0 &= 0 \end{aligned}$$

investiert. Wiederum gilt:

$$\begin{aligned} D D^- y_i &= R y_i = y_i, \\ Q_0 x_i &= x_i. \end{aligned}$$

Somit erhalten wir

$$\sum \lambda_i \begin{pmatrix} y_i \\ x_i \end{pmatrix} = 0,$$

woraus nach der Voraussetzung der linearen Unabhängigkeit der Vektoren $(x_i, y_i)^T$ unmittelbar $\lambda_i = 0$ folgt.
Sei nun z ein Vektor aus dem Kern von $G_1(t)$. Wir definieren

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} Q_0 z \\ D z \end{pmatrix}. \tag{A.6}$$

Eine leichte Rechnung zeigt

$$\begin{pmatrix} x \\ y \end{pmatrix} \in \ker M.$$

Seien z_i linear unabhängige Vektoren aus $\ker G_1(t)$ und seien $(x_i, y_i)^T$ gemäß (A.6) definiert. Es gilt:

$$\sum \lambda_i \begin{pmatrix} x_i \\ y_i \end{pmatrix} = 0 \Longleftrightarrow \begin{cases} \sum \lambda_i Q_0 z_i &= 0, \\ \sum \lambda_i D z_i &= 0. \end{cases}$$

Multiplizieren wir die untere rechte Gleichung mit D^- und investieren $D^-D = P_0$, so folgt wiederum $\lambda_i = 0$.
Insgesamt haben wir gezeigt, wie sich eine Basis von ker$M(t)$ aus Basis Vektoren von ker$A(t)$ und ker$G_1(t)$ konstruieren lässt. Die Konstruktion macht deutlich, dass die Dimension des Kerns von $M(t)$ unabhängig von t ist. Sie ist genau $r + m - r_1$. Damit besitzt aber auch $M(t)$ selbst konstanten Rang.

□

Bemerkung: Das Lemma gilt auch für allgemeinere nichtlineare Differential-Algebraische Gleichungen

$$A(x(t), t)[d(x(t), t)]' + B(x(t), t) = q(t), \quad t \in \mathcal{I}$$

mit proper formuliertem Hauptterm und Traktabilitätsindex 2 (vgl. zu den Definitionen im nichtlinearen Fall [M01]). Alle oben verwendeten Größen werden nämlich über die Linearisierung definiert (vgl. ebd.).

B Programme

In diesem Anhang stellen wir einige Programmcodes bereit. Dabei lag der Schwerpunkt der Programmierung in übersichtlichen Programmen, nicht in der Optimierung, wie sie für einen professionellen Code nötig wären. Die Fortran Programme sind dabei stark an dem RADAU5-Code orientiert (vgl. [HW] Appendix. Fortran Codes)

Zur Berechnung der Koeffizientenmatrizen $\hat{A}$ und $\hat{B}$ einer *singly implicit* Runge-Kutta ähnlicher Startprozedur für vorgegebenen Knotenvektor $\hat{c}$, Eigenwert μ und vorgegebene Ordnung $\hat{p} = \hat{s} - 1$ dient das folgende Matlabprogramm (vgl. Unterkapitel 4.1):

```
function [A,B,T,LAMBDA,BAI] = startcoefficient(c,mu,s)
% ------------------------------------------------------%
% BERECHNUNG DER KOEFFIZIENTENMATRIZEN EINER           %
% SINGLY IMPLICIT RUNGE-KUTTA ÄHNLICHEN STARTPROZEDUR %
%-------------------------------------------------------%
% ------ DEFINITION DER SHIFTMATRIX J
  J=[zeros(1,s-1),0;eye(s-1),zeros(s-1,1)];
% ------ BERECHNUNG DER SKALIERTEN VANDERMONDE C
  C = ones(s,1);
  fac=1;
  for i=1:s-1
      fac=fac*i;
      C=[C,c.^(i)/fac];
  end
% ------  BERECHNUNG DER MATRIX L
  a=zeros(s,1);
  l=zeros(s,1);
  for i=s:-1:1
      a(i)=(-1)^(s-i)*binom(s,i-1)*mu^(s+1-i);
  end
  l=C*a;
  L=[zeros(s,s-1),l]/C;
% ------ BERECHNUNG DER MATRIZEN A und B
  A=C*J/C+L;
```

```
  B=J/C;
  AI=A\eye(s);
% ------ BERECHNUNG DER TRANSFORMATIONSMATRIX T
  hilf=C(:,s);
  T = hilf;
  fac=1/mu;
  matrix=AI-fac*eye(s);
  for i=1:s-1
      hilf=matrix*hilf;
      T=[T,hilf];
  end
% ------ BERECHNUNG VON T^-1*A^-1*T=LAMBDA
  LAMBDA= (A*T)\T;
% ------ BERECHNUNG VON BA^-1
  BAI=B*AI;
```

Dabei ist binom eine Subfunktion zur Berechnung des Binomialkoeffizienten:

```
function [y] =binom(n,k)
prod=1;
if (k>n) y=0;
elseif (k<=n/2)
    for i=1:k
        prod=prod*(n+1-i)/i;
    end
    y=prod;
else
    y=binom(n,n-k);
end
```

Startprozedur der Ordnung 2 in y und der Ordnung 1 in z, deren Koeffizienten durch das Matlabprogramm oben berechnet worden sind:

```
C-----------------------------------------------------------------
C STARTPROZEDUR DER ORDNUNG 2 (s=3)
C-----------------------------------------------------------------
      SUBROUTINE START3(N,L,X0,Y0,Z0,H,FCN,JAC,
     &                  SCAL,NIT,UROUND,FNEWT,
     &                  YNOR1,YNOR2,YNOR3,
```

```
     &                   ZNOR1,ZNOR2,ZNOR3,
     &                   NFCN,NJAC,NDEC,NSOL,IDID)
C-----------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION Y0(N), YNOR1(N), YNOR2(N), YNOR3(N)
      DIMENSION Z0(L), ZNOR1(L), ZNOR2(L), ZNOR3(L)
      DIMENSION IPIV(N+L,N+L),E(N+L,N+L),FJAC(N+L,N+L)
      DIMENSION CONTY(N), CONTZ(L)
      DIMENSION WF1(N),WF2(N),WF3(N),WG1(L),WG2(L),WG3(L)
      DIMENSION F1(N),F2(N),F3(N),G1(L),G2(L),G3(L)
      DIMENSION RHS1(N+L),RHS2(N+L),RHS3(N+L),SCAL(N+L)
      EXTERNAL FCN,JAC
C ------ INITIALISIERUNG
      IDID  = 0
      NFCN  = 0
      NJAC  = 0
      NDEC  = 0
      NSOL  = 0
C ------ DIMENSION DES GESAMTSYSTEMS
      NL = N+L
      NL3 = NL*3
C ------ KOEFFIZIENTEN DER STARTPROZEDUR
      T21 =  0.125D0
      T22 =  0.375D0
      T23 =  0.125D0
      T31 =  0.5D0
      T32 =  0.5D0
      T33 = -0.5D0
      TI11 =  2.0D0
      TI12 = -4.0D0
      TI13 =  3.0D0
      TI21 = -1.0D0
      TI22 =  4.0D0
      TI23 = -1.0D0
      TI31 =  1.0D0
      BAI22 =  4.0D0
      BAI23 = -1.0D0
      BAI31 =  1.0D0
      BAI32 = -8.0D0
      BAI33 =  4.0D0
      c2 = 0.5D0
      FAC=1/H
C ------ BERECHNUNG DER JACOBI-MATRIX
      CALL JAC(N,L,X0,Y0,Z0,FJAC)
```

```
      NJAC=NJAC+1
C ------ BERECHNUNG DER LU-ZERLEGUNG
      DO I=1,NL
         DO J=1,NL
            E(I,J)   = -FJAC(I,J)
         END DO
      END DO
      DO I=1,N
         E(I,I) = E(I,I) + FAC
      END DO
      CALL DGETRF(NL,NL,E,NL,IPIV,INFO)
      IF (INFO.GT.0) GOTO 78
      IF (INFO.LT.0) GOTO 78
      NDEC=NDEC+1
C ------ STARTWERTE DES NEWTON-VERFAHRENS
      DO I=1,N
         WF1(I) = 0.0D0
         WF2(I) = 0.0D0
         WF3(I) = 0.0D0
      END DO
      DO J=1,L
         WG1(J) = 0.0D0
         WG2(J) = 0.0D0
         WG3(J) = 0.0D0
      END DO
C ------ SCHLEIFE DES NEWTON-VERFAHRENS
      NEWT = 0
      DYNO   = 1.0D0
      FACCON = 1.0D0
  40  CONTINUE
        NEWT = NEWT+1
        IF (NEWT.GE.NIT) GOTO 79
C -------- BERECHNUNG DER FUNKTIONSWERTE DER RECHTEN SEITE
        DO I=1,N
           CONTY(I)=WF3(I)+Y0(I)
        END DO
        DO J=1,L
           CONTZ(J)=WG3(J)+Z0(J)
        END DO
        CALL FCN(N,L,X0,CONTY,CONTZ,F1,G1)
        DO I=1,N
           CONTY(I)=T21*WF1(I)+T22*WF2(I)+T23*WF3(I)+Y0(I)
        END DO
        DO J=1,L
```

```
            CONTZ(J)=T21*WG1(J)+T22*WG2(J)+T23*WG3(J)+Z0(J)
         END DO
         CALL FCN(N,L,X0+c2*H,CONTY,CONTZ,F2,G2)
         DO I=1,N
            CONTY(I)=T31*WF1(I)+T32*WF2(I)+T33*WF3(I)+Y0(I)
         END DO
         DO J=1,L
            CONTZ(J)=T31*WG1(J)+T32*WG2(J)+T33*WG3(J)+Z0(J)
         END DO
         CALL FCN(N,L,X0+H,CONTY,CONTZ,F3,G3)
         NFCN=NFCN+3
C -------- BERECHNUNG DER RECHTEN SEITE
         DO I=1,N
            RHS1(I) = -FAC*WF1(I)
     &               + TI11*F1(I)+TI12*F2(I)+TI13*F3(I)
         END DO
         DO J=1,L
            RHS1(J+N) = TI11*G1(J)+TI12*G2(J)+TI13*G3(J)
         END DO
C -------- LÖSEN DES LINEAREN GLEICHUNGSSYSTEMS
         CALL DGETRS('N',NL,1,E,NL,IPIV,RHS1,NL,INFO)
C -------- BERECHNUNG DER RECHTEN SEITE
         DO I=1,N
            RHS2(I) = -FAC*(WF1(I)+WF2(I)+RHS1(I))
     &               + TI21*F1(I)+TI22*F2(I)+TI23*F3(I)
         END DO
         DO J=1,L
            RHS2(J+N) = TI21*G1(J)+TI22*G2(J)+TI23*G3(J)
         END DO
C -------- LÖSEN DES LINEAREN GLEICHUNGSSYSTEMS
         CALL DGETRS('N',NL,1,E,NL,IPIV,RHS2,NL,INFO)
C -------- BERECHNUNG DER RECHTEN SEITE
         DO I=1,N
            RHS3(I) = -FAC*(WF2(I)+WF3(I)+RHS2(I))
     &              + TI31*F1(I)
         END DO
         DO J=1,L
            RHS3(J+N) = TI31*G1(J)
         END DO
C -------- LÖSEN DES LINEAREN GLEICHUNGSSYSTEMS
         CALL DGETRS('N',NL,1,E,NL,IPIV,RHS3,NL,INFO)
         NSOL=NSOL+3
C -------- BERECHNUNG DER FEHLERNORM
         DYNO=0.0D0
```

```
         DO I=1,NL
            DENOM=SCAL(I)
            DYNO = DYNO + (RHS1(I)/DENOM)**2 + (RHS2(I)/DENOM)**2
     &                  + (RHS3(I)/DENOM)**2
         END DO
         DYNO = DSQRT(DYNO/NL3)
C -------- SCHLECHTE KONVERGENZ ODER ZAHL DER SCHRITTE ZU HOCH
         IF (NEWT.GT.1.AND.NEWT.LT.NIT) THEN
             THQ=DYNO/DYNOLD
             IF (NEWT.EQ.2) THEN
                 THETA=THQ
             ELSE
                 THETA=SQRT(THQ*THQOLD)
             END IF
             THQOLD=THQ
             IF (THETA.LT.0.99D0) THEN
                 FACCON=THETA/(1.0D0-THETA)
                 DYTH=FACCON*DYNO*THETA**(NIT-1-NEWT)/FNEWT
                 IF (DYTH.GE.1.0D0) GOTO 177
             ELSE
                 GOTO 178
             END IF
         END IF
         DYNOLD=MAX(DYNO,UROUND)
C -------- BERECHNUNG DER NEUEN ITERIERTEN W^i+1
         DO I=1,N
            WF1(I) = WF1(I) + RHS1(I)
            WF2(I) = WF2(I) + RHS2(I)
            WF3(I) = WF3(I) + RHS3(I)
         END DO
         DO J=1,L
            WG1(J) = WG1(J) + RHS1(N+J)
            WG2(J) = WG2(J) + RHS2(N+J)
            WG3(J) = WG3(J) + RHS3(N+J)
         END DO
       IF (FACCON*DYNO.GT.FNEWT) GOTO 40
C ------- BERECHNUNG DES START(NORDSIECK)VEKTORS
       DO I=1,N
          hilf1 =                                  WF3(I)
          hilf2 = T21*WF1(I) + T22*WF2(I) + T23*WF3(I)
          hilf3 = T31*WF1(I) + T32*WF2(I) + T33*WF3(I)
          WF1(I) = hilf1
          WF2(I) = hilf2
          WF3(I) = hilf3
```

```
      END DO
      DO I=1,N
         YNOR1(I) = Y0(I)
         YNOR2(I) =          BAI22*WF2(I) + BAI23*WF3(I)
         YNOR3(I) = WF1(I) + BAI32*WF2(I) + BAI33*WF3(I)
      END DO
      DO J=1,L
         hilf1 =                                WG3(J)
         hilf2 = T21*WG1(J) + T22*WG2(J) + T23*WG3(J)
         hilf3 = T31*WG1(J) + T32*WG2(J) + T33*WG3(J)
         WG1(J) = hilf1
         WG2(J) = hilf2
         WG3(J) = hilf3
      END DO
      DO J=1,L
         ZNOR1(J) = Z0(J)
         ZNOR2(J) =          BAI22*WG2(J) + BAI23*WG3(J)
         ZNOR3(J) = WG1(J) + BAI32*WG2(J) + BAI33*WG3(J)
      END DO
      RETURN
C ------ LU-ZERLEGUNG NICHT MÖGLICH
  78  CONTINUE
      WRITE(6,*) ' MATRIX IST SINGULAR', INFO
      IDID=-4
      RETURN
C ------ ZU VIELE SCHRITTE
  79  CONTINUE
      WRITE(6,*) ' ANZAHL DER ITERATIONEN ZU HOCH, NEWT=', NEWT
      IDID=-3
      RETURN
C ------ KEINE KONVERGENZ
 177  CONTINUE
      WRITE(6,*) ' NEWTON WIRD NICHT KONVERGIEREN, DYTH=', DYTH
      IDID=-2
      RETURN
C ------ SCHLECHTE KONVERGENZRATE
 178  CONTINUE
      WRITE(6,*) ' KONVERGENZRATE ZU SCHLECHT, THETA=', THETA
      IDID=-1
      RETURN
      END
C
C     END OF SUBROUTINE START3
```

Allgemeines lineares Verfahren der Typ-2 DIMISIM Klasse mit Koeffizienten aus [Wright02]:

```
C-----------------------------------------------------------------
C DIMSIM DER ORDNUNG 2 (s=3)
C-----------------------------------------------------------------
      SUBROUTINE GLM3(N,L,X,Y0,Z0,XEND,H,FCN,JAC,
     &                RTOL,ATOL,ITOL,
     &                NIT,UROUND,FNEWT,NMAX,
     &                YNOR1,YNOR2,YNOR3,ZNOR1,ZNOR2,ZNOR3,
     &                NFCN,NJAC,NDEC,NSOL,NSTEP,IDID)
C-----------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION Y0(N), YNOR1(N), YNOR2(N), YNOR3(N)
      DIMENSION Z0(L), ZNOR1(L), ZNOR2(L), ZNOR3(L)
      DIMENSION SCAL(N+L)
      DIMENSION FJAC(N+L,N+L)
      DIMENSION IPIV(N+L,N+L), E(N+L,N+L)
      DIMENSION UY1(N), UY2(N), UY3(N)
      DIMENSION UZ1(L), UZ2(L), UZ3(L)
      DIMENSION WF1(N), WF2(N), WF3(N)
      DIMENSION WG1(L), WG2(L), WG3(L)
      DIMENSION F1(N), F2(N), F3(N)
      DIMENSION G1(L), G2(L), G3(L)
      DIMENSION CONTY(N), CONTZ(L)
      DIMENSION RHS1(N+L), RHS2(N+L), RHS3(N+L)
      EXTERNAL FCN,JAC,NEWTON
C ------ INITIALISIERUNG
      NSTEP = 0
C ------ DIMENSION DES GESAMTSYSTEMS
      NL = N+L
C ------ KOEFFIZIENTEN DES VERFAHRENS
      A11 = 0.25D0
      A21 = 0.25D0
      A31 = 0.5D0
      A32 = 0.25D0
      V12 = 0.125D0
      V13 = 0.0625D0
      V23 = 0.25D0
      U12 = -0.25D0
      U33 = 0.125D0
```

```
      c1   = 0.0D0
      c2   = 0.5D0
      c3   = 1.0D0
      B11 =  0.5D0
      B12 = -0.125D0
      B13 =  0.5D0
      B21 =  0.5D0
      B22 = -0.5D0
      B23 =  1.0D0
      B31 =  0.0D0
      B32 = -2.0D0
      B33 =  2.0D0
      BAI11 = 0.5D0
      BAI12 = -2.5D0
      BAI13 = 2.0D0
      BAI22 = -6.0D0
      BAI23 = 4.0D0
      BAI32 = -16.0D0
      BAI33 = 8.0D0
      VBAU11 = 1.0D0
      VBAU12 = 0.25D0
      VBAU13 = -3.D0/16.D0
      VBAU21 = 2.0D0
      VBAU22 = 0.0D0
      VBAU23 = -0.25D0
      VBAU31 = 8.0D0
      VBAU32 = 0.0D0
      VBAU33 = -1.0D0
      FAC = H*A11
      FAC = 1/FAC
C ------ SKALIERUNG
      DO I=1,N
         SCAL(I)=ATOL+RTOL*ABS(Y0(I))
      END DO
      DO J=1,L
         SCAL(N+J)=(ATOL+RTOL*ABS(Z0(J)))/H
      END DO
C ------ BERECHNUNG DES ANFANGS(NORDSIECK)VEKTORS ----------
      CALL START3(N,L,X,Y0,Z0,H,FCN,JAC,
     &            SCAL,NIT,UROUND,FNEWT,
     &            YNOR1,YNOR2,YNOR3,
     &            ZNOR1,ZNOR2,ZNOR3,
     &            NFCN,NJAC,NDEC,NSOL,IDID)
C -----------------------------------------------------------
```

```
      IF (IDID.LT.0) GOTO 77
C ------ INTEGRATIONSSCHRITT
  10  CONTINUE
C ------ BERECHNUNG DER JACOBI-MATRIX
      CALL JAC(N,L,X,YNOR1,ZNOR1,FJAC)
      NJAC=NJAC+1
  20  CONTINUE
C ------ BERECHNUNG VON (U\otimes)Y^n UND (U\otimes)Z^n
      DO I=1,N
         UY1(I) = YNOR1(I) + YNOR2(I)*U12
         UY2(I) = YNOR1(I)
         UY3(I) = YNOR1(I) + YNOR3(I)*U33
      END DO
      DO J=1,L
         UZ1(J) = ZNOR1(J) + ZNOR2(J)*U12
         UZ2(J) = ZNOR1(J)
         UZ3(J) = ZNOR1(J) + ZNOR3(J)*U33
      END DO
C ------ BERECHNUNG DER LU-ZERLEGUNG
      DO I=1,NL
         DO J=1,NL
            E(I,J)   = -FJAC(I,J)
         END DO
      END DO
      DO I=1,N
         E(I,I) = E(I,I) + FAC
      END DO
      CALL DGETRF(NL,NL,E,NL,IPIV,INFO)
      IF (INFO.GT.0) GOTO 78
      IF (INFO.LT.0) GOTO 78
      NDEC=NDEC+1
  30  CONTINUE
      NSTEP=NSTEP+1
      IF (NSTEP.GT.NMAX) GOTO 179
C ------ STARTWERTE DES NEWTON-VERFAHRENS
      DO I=1,N
         WF1(I) = 0.0D0
         WF2(I) = 0.0D0
         WF3(I) = 0.0D0
      END DO
      DO J=1,L
         WG1(J) = UZ1(J)
         WG2(J) = UZ2(J)
         WG3(J) = UZ3(J)
```

```
      END DO
C ------ AUFRUF DES NEWTON-VERFAHRENS
      CALL NEWTON(N,L,X,H,FCN,c1,FAC,E,IPIV,
     &            WF1,WG1,UY1,
     &            NIT,UROUND,FNEWT,SCAL,
     &            NFCN,NSOL,IDID)
      IF (IDID.LT.0) GOTO 77
C ------ AUFRUF DES NEWTON-VERFAHRENS
      DO I=1,N
         UY2(I) = UY2(I) + WF1(I)
      END DO
      CALL NEWTON(N,L,X,H,FCN,c2,FAC,E,IPIV,
     &            WF2,WG2,UY2,
     &            NIT,UROUND,FNEWT,SCAL,
     &            NFCN,NSOL,IDID)
      IF (IDID.LT.0) GOTO 77
C ------ AUFRUF DES NEWTON-VERFAHRENS
      DO I=1,N
         UY3(I) = UY3(I) + 2*WF1(I) + WF2(I)
      END DO
      CALL NEWTON(N,L,X,H,FCN,c3,FAC,E,IPIV,
     &            WF3,WG3,UY3,
     &            NIT,UROUND,FNEWT,SCAL,
     &            NFCN,NSOL,IDID)
      IF (IDID.LT.0) GOTO 77
      DO I=1,N
         WF1(I) = WF1(I)/A11
         WF2(I) = WF2(I)/A11
         WF3(I) = WF3(I)/A11
      END DO
C ------ BERECHNUNG DER NEUEN ITERATIONEN y^n+1,z^n+1
      DO I=1,N
         YNOR1(I) = YNOR1(I) + V12*YNOR2(I) + V13*YNOR3(I)
     &            + B11*WF1(I) + B12*WF2(I) + B13*WF3(I)
         YNOR2(I) = V23*YNOR3(I) + B21*WF1(I) + B22*WF2(I)
     &                                        + B23*WF3(I)
         YNOR3(I) =                B32*WF2(I) + B33*WF3(I)
      END DO
      DO J=1,L
         hilf1 = ZNOR1(J)
         ZNOR1(J) = ZNOR1(J)+VBAU12*ZNOR2(J)+VBAU13*ZNOR3(J)
     &            + BAI11*WG1(J)+BAI12*WG2(J)+BAI13*WG3(J)
         ZNOR2(J) = VBAU21*hilf1+VBAU23*ZNOR3(J)
     &            + BAI22*WG2(J)+BAI23*WG3(J)
```

```
         ZNOR3(J) = VBAU31*hilf1+VBAU33*ZNOR3(J)
     &            + BAI32*WG2(J)+BAI33*WG3(J)
      END DO
C ------ RESKALIERUNG
      DO I=1,N
         SCAL(I)=ATOL+RTOL*ABS(YNOR1(I))
      END DO
      DO J=1,L
         SCAL(N+J)=(ATOL+RTOL*ABS(ZNOR1(J)))/H
      END DO
      X = X+H
      IF ((X+H).LE.XEND+1.D-9) GOTO 10
      RETURN
  77  CONTINUE
      IF (IDID.EQ.-3) THEN
          GOTO 79
      ELSE
          IF (IDID.EQ.-2) THEN
               GOTO 177
          ELSE
               IF (IDID.EQ.-1) THEN
                   GOTO 178
               END IF
          END IF
      END IF
C ------ LU-ZERLEGUNG NICHT MÖGLICH
  78  CONTINUE
      WRITE(6,*) ' MATRIX IS SINGULAR', INFO
      IDID=-4
      RETURN
C ------ ZU VIELE SCHRITTE
  79  CONTINUE
      WRITE(6,*) ' ANZAHL DER ITERATIONEN ZU HOCH '
      RETURN
C ------ KEINE KONVERGENZ
 177  CONTINUE
      WRITE(6,*) ' NEWTON WIRD NICHT KONVERGIEREN, DYTH=', DYTH
      RETURN
C ------ SCHLECHTE KONVERGENZRATE
 178  CONTINUE
      WRITE(6,*) ' KONVERGENZRATE ZU SCHLECHT, THETA=', THETA
      RETURN
C ------ ANZAHL DER SCHRITTE ZU HOCH
 179  CONTINUE
```

```
      WRITE(6,*) ' ANZAHL DER SCHRITTE GRÖßER ALS, NMAX=', NMAX
      IDID = -5
      RETURN
      END
C
C     END OF SUBROUTINE GLM3
```

Die Subroutine NEWTON:

```
C-----------------------------------------------------------------
C DAS VEREINFACHTE NEWTON-VERFAHREN
C-----------------------------------------------------------------
      SUBROUTINE NEWTON(N,L,X,H,FCN,c,FAC,E,IPIV,
     &                  WF,WG,UY,
     &                  NIT,UROUND,FNEWT,SCAL,
     &                  NFCN,NSOL,IDID)
C-----------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION SCAL(N+L), RHS(N+L)
      DIMENSION IPIV(N+L,N+L), E(N+L,N+L)
      DIMENSION UY(N), WF(N), WG(L)
      DIMENSION F(N), G(L), CONTY(N), CONTZ(L)
      EXTERNAL FCN
C ------ INITIALISIERUNG
      IDID = 0
      NEWT = 0
      DYNO   = 1.0D0
      FACCON = 1.0D0
C ------ DIMENSION DES GESAMTSYSTEMS
      NL = N+L
  40  CONTINUE
        NEWT = NEWT+1
        IF (NEWT.GE.NIT) THEN
           IDID=-3
           RETURN
        END IF
C -------- BERECHNUNG DER FUNKTIONSWERTE DER RECHTEN SEITE
        DO I=1,N
           CONTY(I) = WF(I) + UY(I)
        END DO
        DO J=1,L
           CONTZ(J) = WG(J)
```

```
         END DO
         CALL FCN(N,L,X+c*H,CONTY,CONTZ,F,G)
         NFCN=NFCN+1
C -------- BERECHNUNG DER RECHTEN SEITE
         DO I=1,N
            RHS(I) = -FAC*WF(I) + F(I)
         END DO
         DO J=1,L
            RHS(J+N) = G(J)
         END DO
C -------- LÖSEN DES LINEAREN GLEICHUNGSSYSTEMS
         CALL DGETRS('N',NL,1,E,NL,IPIV,RHS,NL,INFO)
         NSOL=NSOL+1
C -------- BERECHNUNG DER FEHLERNORM
         DYNO=0.0D0
         DO I=1,NL
            DENOM=SCAL(I)
            DYNO=DYNO+(RHS(I)/DENOM)**2
         END DO
         DYNO = DSQRT(DYNO/NL)
C -------- SCHLECHTE KONVERGENZ ODER ANZAHL DER SCHRITTE ZU HOCH
         IF (NEWT.GT.1.AND.NEWT.LT.NIT) THEN
             THQ=DYNO/DYNOLD
             IF (NEWT.EQ.2) THEN
                 THETA=THQ
             ELSE
                 THETA=SQRT(THQ*THQOLD)
             END IF
             THQOLD=THQ
             IF (THETA.LT.0.99D0) THEN
                 FACCON=THETA/(1.0D0-THETA)
                 DYTH=FACCON*DYNO*THETA**(NIT-1-NEWT)/FNEWT
                 IF (DYTH.GE.1.0D0) THEN
                     IDID=-2
                     RETURN
                 END IF
             ELSE
                IDID=-1
                RETURN
             END IF
         END IF
         DYNOLD=MAX(DYNO,UROUND)
C -------- BERECHNUNG DER NEUEN ITERIERTEN W^i+1
         DO I=1,N
```

```
            WF(I) = WF(I) + RHS(I)
          END DO
          DO J=1,L
             WG(J) = WG(J) + RHS(N+J)
          END DO
        IF (FACCON*DYNO.GT.FNEWT) GOTO 40
        RETURN
        END
C
C     END OF SUBROUTINE NEWTON
```

Dieses Programm ist der Treiber für die Differential-Algebraische Gleichung (4.13):

```
C-----------------------------------------------------------------
C TREIBER FÜR DIE DAE (4.13)
C-----------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER (ND=1,LD=3)
      DIMENSION Y0(ND),Z0(LD)
      DIMENSION YNOR1(ND),YNOR2(ND),YNOR3(ND)
      DIMENSION ZNOR1(LD),ZNOR2(LD),ZNOR3(ND)
      EXTERNAL FBCP,JBCP
C-----------------------------------------------------------------
C ------ DIMENSION DER Y VARIABLEN
      N=1
C ------ DIMENSION DER Z VARIABLEN
      L=3
C ------ ANFANGSWERTE
      X     =  0.0D0
      Y0(1) =  1.0D0
      Z0(1) = -1.0D0
      Z0(2) =  1.0D0
      Z0(3) =  1.0D0
C ------ ENDPUNKT DER INTEGRATION
      XEND = 3.0D0
C ------ TOLERANZEN
      RTOL = 1.0D-9
      ATOL = RTOL
      ITOL = 0
C ------ MAXIMALE ANZAHL DER SCHRITTE
      NMAX = 10000000
C ------ KONSTANTEN FÜR DIE NEWTON-ITERATION
```

```
      NIT = 7
      UROUND = 1.D-16
      FNEWT = MIN(0.03,RTOL**0.5)
C ------ SCHRITTWEITE
      H = 1.D-5
C ------ AUFRUF DER INTEGRATIONSMETHODE -------
      CALL GLM3(N,L,X,Y0,Z0,XEND,H,FBCP,JBCP,
     &          RTOL,ATOL,ITOL,
     &          NIT,UROUND,FNEWT,NMAX,
     &          YNOR1,YNOR2,YNOR3,
     &          ZNOR1,ZNOR2,ZNOR3,
     &          NFCN,NJAC,NDEC,NSOL,NSTEP,IDID)
C ---------------------------------------
      STOP
      END
C
C ------ RECHTE SEITE DER DAE
      SUBROUTINE FBCP(N,L,X,Y,Z,F,G)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION Y(N),F(N),Z(L),G(L)
        ETA = -0.75D0
        F(1) = Z(1)
        G(1) = Z(2) + ETA*X*Z(3)-exp(-X)
        G(2) = Z(1) + Z(3)
        G(3) = Y(1) - Z(2)-ETA*X*Z(3)
      RETURN
      END
C
C ------ JACOBI-MATRIX DER RECHTEN SEITE
      SUBROUTINE JBCP(N,L,X,Y,Z,FJAC)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION Y(N),FJAC(N+L,N+L),Z(L)
        ETA = -0.75D0
        FJAC(1,1) =  0.0D0
        FJAC(1,2) =  1.0D0
        FJAC(1,3) =  0.0D0
        FJAC(1,4) =  0.0D0
        FJAC(2,1) =  0.0D0
        FJAC(2,2) =  0.0D0
        FJAC(2,3) =  1.0D0
        FJAC(2,4) =  ETA*X
        FJAC(3,1) =  0.0D0
        FJAC(3,2) =  1.0D0
        FJAC(3,3) =  0.0D0
```

```
      FJAC(3,4) =  1.0D0
      FJAC(4,1) =  1.0D0
      FJAC(4,2) =  0.0D0
      FJAC(4,3) =  -1.0D0
      FJAC(4,4) =  -ETA*X
    RETURN
    END
```

Literaturverzeichnis

[Alt] H. W. Alt, *Lineare Funktionalanalysis, eine anwendungsorientierte Einführung*, Springer, Berlin-Heidelberg-New York-London-Paris-Tokyo-Hong Kong-Barcelona-Budapest, 1992[2].

[A95] M. Arnold, „A perturbation analysis for the dynamical simulation of mechanical multibody systems,“*Appl. Numer. Math.* **18** (1995), 37-56.

[ASW95] M. Arnold, K. Strehmel, R. Weiner, „Errors in the numerical solution of nonlinear differential-algebraic systems of index 2,“Martin-Luther-University Halle, Department of Mathematics and Computer Science. - Technical Report **11** (1995).

[AP91] U. M. Ascher, L. R. Petzold, „Projected implicit Runge-Kutta Methods for differential-algebraic equations,“*SIAM J.Numer.Anal.* **28** (1991), 1097-1120.

[AP] U. M. Ascher, L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM, 1998.

[AC98] A. Aubry, P. Chartier, „On improving the Convergence of Radau IIA Methods applied to Index-2 DAEs,“*SIAM J. Numer. Anal.* **35** (1998), 1347-1367.

[Beyn94] W.-J. Beyn, „Numerical analysis of homoclinic orbits emanating a Takens-Bogdanov point,“*IMA J. of Numer. Anal.* **14** (1994), 381-410.

[BS00] W.-J. Beyn, J. Schropp, „Runge-Kutta discretizations of singularly perturbed gradient equations,“*BIT* **40** (2000), 415-433.

[BP89] K. Brenan, L. R. Petzold, „The numerical solution of higher index differential/algebraic equations by imlicit methods,“*SIAM J. Numer. Anal* **26** (1989), 976-996.

[BCP] K. Brenan, S. Campbell, L. R. Petzold, *Numerical Solution of Initial-Value Problemms in Differential-Algebraic Equations*, North Holland Publishing Co., 1989.

[BB80] K. Burrage, J. C. Butcher, „Non-linear stability of a general class of differential equation methods,“*BIT* **20** (1980), 185-203.

[BBC80] K. Burrage, J. C. Butcher, F. H. Chipman „An Implementation of singly-implicit Runge-Kutta methods,“ *BIT* **20** (1980), 326-340.

[B64] J. C. Butcher, „Implicit Runge-Kutta processes,“ *Math. Comp.* **18** (1964), 50-64.

[B66] J. C. Butcher, „On the convergence of numerical solutions to ordinary differential equations,“ *Math. Comp.* **20** (1966), 1-10.

[B76] J. C. Butcher, „On the implementation of implicit Runge-Kutta methods,“ *BIT* **16** (1976), 237-240.

[B93] J. C. Butcher, „Diagonally-implicit multi-stage integration methods,“ *Appl. Numer. Math.* **11** (1993), 347-363.

[B01] J. C. Butcher, „General linear methods for stiff differential equations,“ *BIT* **41** (2001), 240-264.

[B] J. C. Butcher, *Numerical methods for ordinary differential equations*, John Wiley Sons, 2003.

[BJ04a] J. C. Butcher, Z. Jackiewicz, „Unconditionally stable general linear methods for ordinary differential equations,“ *BIT* **44** (2004), 557-570.

[BJ04b] J. C. Butcher, Z. Jackiewicz, „Construction of general linear methods with Runge-Kutta stability properties,“ *Numer. Algorithms* **36** (2004), 53-72.

[BP06] J. C. Butcher, H. Podhaisky, „On error estimation in general linear methods for stiff ODEs,“ *Appl. Numer. Math.* **56** (2006), 345-357.

[BR05] J. C. Butcher, N. Rattenbury, „ARK methods for stiff problems,“ *Appl. Numer. Math.* **53** (2005), 165-181.

[BW03] J. C. Butcher, W. Wright, „The construction of practical general linear methods,“ *BIT* **43** (2003), 695-721.

[CG95] S. L. Campbell, C. W. Gear, „The index of general nonlinear DAEs,“ *Numer. Math.* **72** (1995), 173-196.

[D64] V. Doležal, „The existence of a continuous basis of a certain linear subspace of E_r, which depends on a parameter,“ Č*asopis pro pěstováni matematiky*, roč. **89** (1964), 466-468.

[EF] E. Eich-Soellner, C. Führer, *Numerical methods in Multibody Dynamics*, Teubner, 1998.

[EFLR90] E. Eich, C. Führer, B. Leimkuhler, S. Reich *Stabilization and projection methods for multibody dynamics*, Helsinki University of Technology, Institute of Mathematics. - Research Reports, 1990.

[G71] C. W. Gear, „Simultaneous numerical solution of differential-algebraic equations,“*IEEE Trans. Circuits and Theory CT-18* **1** (1971), 89-95.

[G80] C. W. Gear, „Runge-Kutta Starters for Multistep Methods,“*ACM Trans. Math. Softw.* **6** (1980), 263-279.

[G86] C. W. Gear, „Maintaining solution invariants in the numerical solution of ODEs,“*SIAM J. Sci. Stat. Comput.* **7** (1986), 734-743.

[G88] C. W. Gear, „Differential-Algebraic Equation Index Transformation,“*SIAM J. Sci. Stat. Comput.* **9** (1988), 39-47.

[GGL85] C. W. Gear, G. K. Gupta, B. Leimkuhler, „Automatic integration of Euler-Lagrange equations with constraints,“*J. Comp. Appl. Math.* **12/13** (1985), 77-90.

[GP83] C. W. Gear, L. R. Petzold, „Differential/algebraic systems and matrix pencils“in *Matrix Pencils* (B. Kagstrom, A. Ruhe, ed.), Lecture Notes in Mathematics 973, Springer, 1983, 75-89.

[GP84] C. W. Gear, L. R. Petzold, „ODE methods for the solution of differential/algebraic systems,“*SIAM J. Numer. Anal.* **21** (1984), 716-728.

[GM] E. Griepentrog, R. März, *Differential-Algebraic Equations and their Numerical Treatment*, Teubner Verlagsgesellschaft, 1986.

[HNW] E. Hairer, S. P. Norsett, G. Wanner, *Solving ordinary differential equations I, Nonstiff problems*, Springer, 2000^2.

[HW] E. Hairer, G. Wanner, *Solving ordinary differential equations II, Stiff and differential-algebraic problems*, Springer, 2002^2.

[HLR89] E. Hairer, C. Lubich, M. Roche, *The numerical solution of differential-algebraic systems by Runge-Kutta methods*, Lecture Notes in Mathematics 1409, Springer, 1989.

[H] H. Heuser, *Lehrbuch der Analysis, Teil 1*, Teubner, Stuttgart , 1990^{11}.

[Huang05] S. J. Y. Huang, *Implementation of General Linear Methoda for Stiff Ordinary Differential Equations*, Ph.D. thesis, The University of Auckland, New Zealand, 2005, http://www.math.auckland.ac.nz/ butcher/theses/shirleyhuang.pdf.

[KM] P. Kunkel, V. Mehrmann, *Differential-Algebraic Equations. Analysis and Numerical Solution*, EMS, 2006.

[LP86] P. Lötstedt, L. Petzold „Numerical solution of nonlinear differential equations with algebraic constraints I: Convergence results for Backward Differentiation Formulas,“*Math. Comp.* **46** (1986), 491-516.

[L89] C. Lubich, „Linearly implicit extrapolation methods for differential-algebraic systems,“*Numer. Math.* **55** (1989), 197-211.

[M01] R. März „Nonlinear differential-algebraic equations with properly formulated leading terms,“*Tech. Report 01-3* (2001), Humboldt Universität zu Berlin, http://www.math.hu-berlin.de/publ/pre/2001/P-01-3.ps.

[M02a] R. März „Differential algebraic systems anew,“*Appl. Numer. Math.* **42** (2002), 315-335.

[M02b] R. März „The index of linear differential algebraic equations with properly stated leading terms,“*Results Math.* **42** (2002), 308-338.

[P86] L. R. Petzold „Order results for implicit Runge-Kutta methods applied to differential/algebraic systems,“*SIAM J. Numer. Anal.* **23** (1986), 837-852.

[PR74] A. Prothero, A. Robinson „On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations,“*Math. Comp.* **28** (1974), 145-162.

[Sjö99] A. Sjö, *Analysis of Computational Algorithms for Linear Multistep Methods*, Ph.D. thesis, Lund University, Sweden, 1999, http://www.maths.lth.se/na/Publications/Sjo99.pdf

[Sch97] S. Schneider, „Convergence of genral linear methods on differential-algebraic systems of index 3,“*BIT* **37** (1997), 424-441.

[S76] R. Skeel, „Analysis of fixed-stepsize methods,“*SIAM J. Numer. Anal.* **13** (1976), 664-685.

[SB] J. Stoer und R. Bulirsch, *Numerische Mathematik 2*, Springer, Berlin-Heidelberg-New York-London- Paris-Tokyo-Hong Kong,1990[3].

[Voigtmann06] S. Voigtmann, *General Linear Methods for Integrated Circuit Design, Draft version*, Ph.D. thesis, Humboldt Universität zu Berlin, 2006, http://edoc.hu-berlin.de/dissertationen/voigtmann-steffen-2006-06-26/PDF/voigtmann.pdf.

[Wright02] W. Wright, *General linear methods with inherent Runge-Kutta stability*, Ph.D. thesis, The University of Auckland, New Zealand, 2002, http://www.math.auckland.ac.nz/ butcher/theses/willwright.pdf.

[Lapack] http://www.netlib.org/lapack.

[BTS] http://pitagora.dm.uniba.it/ testset.

Eigenständigkeitserklärung

Ich versichere, dass ich die von mir vorgelegte Dissertation selbständig angefertigt, die benutzten Quellen und Hilfsmittel vollständig angegeben und die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken im Wortlaut oder dem Sinn nach entnommen sind, in jedem Einzelfall als Entlehnung kenntlich gemacht habe; dass diese Dissertation noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen hat; dass sie - abgesehen von unten angegebenen Teilpublikationen - noch nicht veröffentlicht worden ist sowie, dass ich eine solche Veröffentlichung vor Abschluss des Promotionsverfahrens nicht vornehmen werde.
Die Bestimmungen der Promotionsordnung sind mir bekannt. Die von mir vorgelegte Dissertation ist von Herrn Prof. Dr. Johannes Schropp betreut worden.

Teilpublikationen: keine

Köln, 30. April 2007 Daniel Weiß