

Effiziente Lösung  
von  
singulären  
Differentialgleichungen

W. Auzinger  
**O. Koch**  
E. Weinmüller

Technische Universität Wien

## Singuläre Problemklasse

$$z'(t) = \underbrace{\frac{M(t)}{t}z(t) + f(t, z(t))}_{=:F(t, z(t))}, \quad t \in (0, 1],$$
$$B_a z(0) + B_b z(1) = \beta.$$

*Bemerkung:* Probleme zweiter Ordnung der Gestalt

$$y''(t) = \frac{A_1(t)}{t}y'(t) + \frac{A_0(t)}{t^2}y(t) + f(t, y(t))$$

können mittels der *Euler-Transformation* auf Systeme erster Ordnung transformiert werden:

$$z(t) := (y(t), ty'(t)).$$

## Anwendungen

- Dynamische Systeme — durch Bogenlänge parametrisierte Verbindungstrajektorien;  
z.B. Lorenz-Gleichung (Meteorologie)  
Kooperation Imperial College, London
- Computational Material Science — radial-symmetrische Schrödingergleichung;  
z.B. Dichtefunktionaltheorie  
Kooperation TU
- Lawinenschutz — Auslauflänge  
Kooperation Universität für Bodenkultur
- Mechanik — Beulverhalten von dünnwandigen Kugelschalen
- Algebra-Differentialgleichungen;  
z.B. Kontrolltheorie
- Unbeschränkte Intervalle;  
z.B. Strömungsdynamik

# Motivation

- Ordnungsreduktionen von Verfahren hoher Ordnung
  - de Hoog, Weiss (1985) — Explizite Runge-Kutta Verfahren
  - de Hoog, Weiss (1977) — Mehrschrittverfahren
  - Koch, Weinmüller (2000) — Beschleunigungsalgorithmen (IDeC, Extrapolation) versagen für allgemeine Problemklasse
- Lokale Fehlerschätzung unzuverlässig, z.B.
  - Kofler (1998) — Runge-Kutta Verfahren
  - Gräff, Weinmüller (1986) — Dreipunkt-Diskretisierung für Gleichungen zweiter Ordnung

## Unsere aktuellen Zugänge

- **Schießverfahren + IDeC** basierend auf **implizitem Eulerverfahren**

- Vorteile

- \* Klassische Konvergenzordnung
- \* Nur Vorwärtsintegration
- \* Strategienwechsel möglich

- Nachteil

- \* Eingeschränkte Problemklasse

- **Kollokation**

- Vorteile

- \* Stufenordnung gesichert für allgemeine Problemklasse
- \* Approximation gleichmäßig in  $t$

- Nachteile

- \* I.A. keine Superkonvergenz
- \* Größerer Rechenaufwand

# Schießverfahren

AWP äquivalent zu RWP:

$$\begin{aligned}z'_s(t) &= \frac{M(t)}{t} z_s(t) + f(t, z_s(t)), \\z_s(0) &= \tilde{E}s \in \ker M(0),\end{aligned}$$

wobei  $s$  so bestimmt wird, dass

$$B_a \tilde{E}s + B_b z_s(1) = \beta.$$

Dies geschieht mit dem Newtonverfahren  $\Rightarrow$   
das erfordert die Lösung mehrerer AWP!

Koch, Weinmüller (2001) —

- Der Prozess ist **wohldefiniert** und **konvergent** wenn die AWP sachgemäß gestellt sind.
- Numerisches Verfahren der Ordnung  $O(h^p)$  für AWP  $\Rightarrow$   **$O(h^p)$  Lösung** für RWP.

## Schießverfahren (2)

### Beweistechnik:

- **Wohldefiniiertheit** und **lokale Konvergenz** der (exakten) Iteration ist aus Eigenschaften eines wohldefinierten singulären RWP mit isolierter Lösung zu folgern.
- Numerische Approximation: Konvergenzresultat für **gestörte Newtoniteration**:
  - **Quadratische Konvergenz** weg vom Fixpunkt.
  - **Lineare Konvergenz** in der Nähe des Fixpunkts.
  - Iteration „landet“ in Umgebung des Fixpunkts, jedoch im Allgemeinen keine Konvergenz gegen festen Punkt.

Nachteil: Eingeschränkte Problemklasse

Benötigt: Verfahren hoher Ord. für AWP

## Iterierte Defektkorrektur

- Interpolation der numerischen Lösung  $z_h^{[0]}$ , die mit dem impliziten Eulerverfahren berechnet wurde, mit stetiger stückweiser Polynomfunktion vom Grad  $m$
- Aufstellen eines (analytischen!) Nachbarproblems (NP) mit bekannter Lösung
- Lösen des NP mit implizitem Euler
- Korrektur der Basislösung mittels bekanntem Fehler der numerischen Lösung des NP
- Interpolation der so gewonnenen Lösung
- Iterative Fortsetzung

Koch, Weinmüller (2000) — Die Ordnung der Lösung erhöht sich in jedem Schritt um 1, bis das theoretische Maximum  $m$  erreicht ist.



## Iterierte Defektkorrektur (2)

### Beweistechnik:

Asymptotische Fehlerentwicklung:

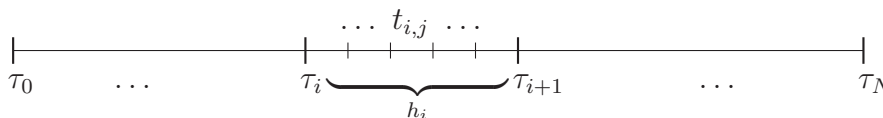
$$z_j^{[0]} = z(t_j) + \sum_{k=1}^m h^k e_k(t_j) + R_j, \quad |R_j| = O(h^{m+1}).$$

$e_k$  sind glatte Funktionen —

die Lösungen der **Variationsgleichungen**

- Im singulären Fall kann für die Fehlerentwicklung Faà di Bruno's Formel nicht direkt verwendet werden.
- Da die rechte Seite der Differentialgleichung nicht Lipschitz-stetig ist, muss an manchen Stellen eine gewisse Integraldarstellung der Lösung verwendet werden.

# Kollokation



$$p'(t_{i,j}) = F(t_{i,j}, p(t_{i,j})),$$

$$j = 1, \dots, m, \quad i = 0, \dots, N - 1,$$

$$B_a p(0) + B_b p(1) = \beta.$$

Die **Stufenordnung bleibt erhalten**, aber  
i.A. **keine Superkonvergenz**.

Tatsächlich tritt die Ordnungsreduktion für die  
meisten singulären Probleme nicht auf  $\implies$

- gerade Anzahl äquidistanter (innerer)  
Kollokationspunkte ..... **default**
- Gauß-Punkte ..... **optional**

## **Innere Kollokationspunkte:**

- Keine Auswertung in Singularität  $t = 0$
- Unser Fehlerschätzer  $\iff t_{i,m} \neq \tau_{i+1}$

## Globale Fehlerschätzung

- Wir konstruieren ein NP ähnlich der IDeC.
- Dann lösen wir für  $j = 1, \dots, m + 1$ ,  
 $i = 0, \dots, N - 1$ , die parallelen Differenzen-  
schemata (impliziter Euler!)

$$\frac{\pi_{i,j} - \pi_{i,j-1}}{t_{i,j} - t_{i,j-1}} = F(t_{i,j}, \pi_{i,j}) + \bar{d}_{i,j},$$
$$B_a \pi_{0,0} + B_b \pi_{N-1,m+1} = \beta,$$

$$\frac{\xi_{i,j} - \xi_{i,j-1}}{t_{i,j} - t_{i,j-1}} = F(t_{i,j}, \xi_{i,j}),$$
$$B_a \xi_{0,0} + B_b \xi_{N-1,m+1} = \beta,$$

wobei

$$t_{i,m+1} := \tau_{i+1}$$

den Endpunkt jedes Intervalls bezeichnet.

Der Fehlerschätzer ist nun gegeben als

$$\mathcal{E}_{i,j} := \pi_{i,j} - \xi_{i,j}.$$

## Globale Fehlerschätzung (2)

Der Defekt ist definiert als

$$\bar{d}_{i,j} := \frac{p(t_{i,j}) - p(t_{i,j-1})}{t_{i,j} - t_{i,j-1}} - \sum_{k=1}^{m+1} \alpha_{j,k} F(t_{i,k}, p(t_{i,k})).$$

Die  $\alpha_{j,k}$  charakterisieren dabei eine Quadraturformel,

$$\frac{1}{t_{i,j} - t_{i,j-1}} \int_{t_{i,j-1}}^{t_{i,j}} \varphi(\tau) d\tau = \sum_{k=1}^{m+1} \alpha_{j,k} \varphi(t_{i,k}) + O(h_i^{m+1}).$$

Auzinger, Koch, Weinmüller (2001) —

Für reguläre Probleme und Kollokation an einer geraden Anzahl von äquidistanten, inneren Kollokationspunkten ist dieser Fehlerschätzer asymptotisch korrekt,

$$|(p(t_{i,j}) - z(t_{i,j})) - (\pi_{i,j} - \xi_{i,j})| = O(\mathbf{h}^{m+1}), \\ j = 0, \dots, m+1, \quad i = 0, \dots, N-1.$$

## Globale Fehlerschätzung (3)

Beweisskizze für lineare Probleme

$$z'(t) = \underbrace{C(t)z(t) + f(t)}_{=:F(t,z(t))}.$$

Für die asymptotische Korrektheit ist die Differenz der folgenden Hilfsgrößen abzuschätzen:

$$\begin{aligned}\varepsilon_{i,j} &:= \xi_{i,j} - z(t_{i,j}), \\ \bar{\varepsilon}_{i,j} &:= \pi_{i,j} - p(t_{i,j}).\end{aligned}$$

Es gelten die folgenden parallelen Differenzenschemata (impliziter Euler!)

$$\begin{aligned}\frac{\varepsilon_{i,j} - \varepsilon_{i,j-1}}{t_{i,j} - t_{i,j-1}} &= C(t_{i,j})\varepsilon_{i,j} + F(t_{i,j}, z(t_{i,j})) - \\ &\quad - \sum_{k=1}^{m+1} \alpha_{j,k} F(t_{i,k}, z(t_{i,k})) + O(\mathbf{h}^{m+1}),\end{aligned}$$

$$\begin{aligned}\frac{\bar{\varepsilon}_{i,j} - \bar{\varepsilon}_{i,j-1}}{t_{i,j} - t_{i,j-1}} &= C(t_{i,j})\bar{\varepsilon}_{i,j} + F(t_{i,j}, p(t_{i,j})) - \\ &\quad - \sum_{k=1}^{m+1} \alpha_{j,k} F(t_{i,k}, p(t_{i,k})).\end{aligned}$$

## Globale Fehlerschätzung (4)

$$\left\{ \begin{array}{l} \text{Taylorentwicklung um } t_{i,j} \\ \sum_{k=1}^{m+1} \alpha_{j,k} = 1 \text{ für alle } j \end{array} \right. \implies$$

$$|\phi(t_{i,j}) - \sum_{k=1}^{m+1} \alpha_{j,k} \phi(t_{i,k})| \leq Ch_i \max_{\tau_i \leq t \leq \tau_{i+1}} |\phi'(t)|.$$

$$\phi(t) := F(t, p(t)) - F(t, z(t)),$$

$$\phi'(t) \dots \dots \dots \text{totales Differential.}$$

$\implies$

**Schranke für die Differenz der Inhomogenitäten** der Differenzgleichungen:

$$Ch(\|p-z\|_\infty + \|p'-z'\|_\infty) + O(\mathbf{h}^{m+1}) = O(\mathbf{h}^{m+1}).$$

Aus der **Stabilität des impliziten Eulerverfahrens** folgt das Resultat.

## Gleichverteilung des Fehlers

Wir verwenden bewährte Strategie für Gleichverteilung des lokalen Fehlers erfolgreich zur **Gleichverteilung des globalen Fehlers!**

**Prüffunktion**  $\Theta_{i,j}$  basierend auf unserem Fehlerschätzer,

$$\begin{aligned}\mathcal{E}_{i,j} &:= \pi_{i,j} - \xi_{i,j}, \\ \Theta_{i,j} &:= \sqrt[m]{|\mathcal{E}_{i,j}|}.\end{aligned}$$

$(\bar{\tau}_0, \dots, \bar{\tau}_{\bar{N}})$  ... gesuchtes Gitter, um das Integral der Prüffunktion am Intervall  $[0, 1]$  gleichzuveteilern:

$$\begin{aligned}I &:= \int_0^1 \Theta(s) ds := \sum_{i,j} \frac{\Theta_{i,j} + \Theta_{i,j-1}}{2} (t_{i,j} - t_{i,j-1}), \\ \int_{\bar{\tau}_i}^{\bar{\tau}_{i+1}} \Theta(s) ds &= \frac{I}{\bar{N}}, \quad i = 0, \dots, \bar{N} - 1.\end{aligned}$$

## Ein MATLAB 6.0 Code

- Wahl des Polynomgrads  $m = 1 \dots 8$
- Alternativ **äquidistante**/Gauß Knoten
- Wahl der polynomialen Basis:  
**Runge-Kutta, Lagrange, Legendre,**  
monomische Basis
- Gedämpftes Newtonverfahren (Deuffhard)  
Alternativ: **fsolve** (nicht empfohlen)
- Präkonditionierung von LGS  $\implies$   
Konditionszahl wächst linear mit Dimension
- Vektorisierter Input zur Effizienzsteigerung

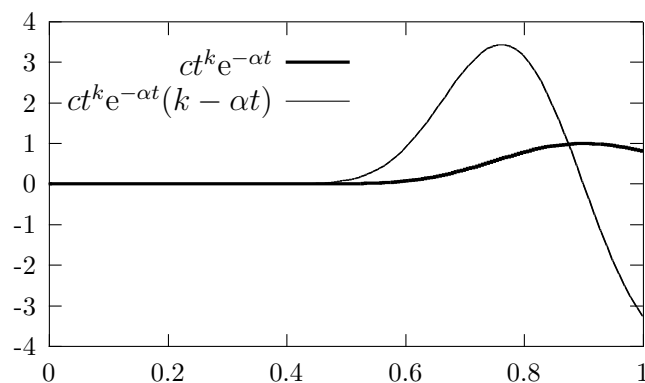


## Beispiel

$$z'(t) = \frac{1}{t} \begin{pmatrix} 0 & 1 \\ 1 + \alpha^2 t^2 & 0 \end{pmatrix} z(t) + \begin{pmatrix} 0 \\ ct^{k-1} e^{-\alpha t} (k^2 - 1 - \alpha t(1 + 2k)) \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} z(0) + \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} z(1) = \begin{pmatrix} 0 \\ ce^{-\alpha} \end{pmatrix},$$

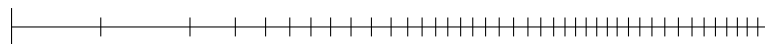
mit  $\alpha = 40$ ,  $k = 36$  und  $c = \left(\frac{\alpha}{k}\right)^k e^k$ .



Gemischte Toleranzen mit  $rTOL=aTOL=10^{-6}$ .

Polynomgrad  $m = 4$ .

Erzeugtes Gitter:



Auswertung am Gitter: \* exakter globaler Fehler,

● Fehlerschätzer.

