Kapitel 3: Adaptivität

Ziel: wähle Zeitschritt zu jedem Zeitpunkt
so groß wie möglich (Rechengeschwindigkeit!)
aber so klein wie nötig (Rechengenauigkeit! Konvergenz bei exp(Vors.!)

Erinnerung:

$$\|x - y\|_G \leq L \, \|E_G(y)\|_G$$

x: numerische Lsg, y: exakte Lsg, $E_G(y)$: bei korrektem Startwert

$$\max_{t \in (a,b]_G} \left\| \frac{\delta y(t)}{\delta t} - V(t,y) \right\|_\infty$$

Taylor Entwicklung

$$\tau(t) = \alpha(t)(\delta t)^p + O((\delta t)^{p+1})$$

p: Verfahren der Konsistenzordnung p

Ziel: sorge dafür (durch Wahl von $\delta t_k = t_k - t_{k-1}$) dass $\|\tau(t_k)\| \leq tol$ gilt

Dann wird auch $\|x - y\|_G \leq L \cdot tol$ gelten.

wegen $\tau(t_k) \approx \alpha(t_k)(\delta t_k)^p$ wähle $\delta t_k \leq \left(\frac{tol}{\|\alpha(t_k)\|_\infty}\right)^{\frac{1}{p}}$

wie kommt man an $\alpha(t_k)$?

Idee: benutze zwei Verfahren gleichzeitig ...

V : Verfahren mit Ordnung p

$\tilde{V}$: Verfahren mit Ordnung $p+1$

[Bem. man kann mit Lsg u des genaueren Verf arbeiten, da $\|u-y\| \lesssim \|x-y\| \leq L \cdot tol$ ↑ genauer ≙ kleinerer Fehler bei gen. kleiner Schr.weite]

rechne von t_{k-1} nach $t_k^\circ = t_{k-1} + h_0$ mit Testschrittweite h_0

$$\tau(t_k^\circ) = \alpha(t_k^\circ)\, h_0^p + \mathcal{O}(h_0^{p+1}) = \frac{\delta y(t_k^\circ)}{h_0} - V(t_k^\circ, y) = \frac{\delta y(t_k^\circ)}{h_0} - \tilde{V}(t_k^\circ, y) + \tilde{V}(t_k^\circ, y) - V(t_k^\circ, y)$$

$$= \beta(t_k^\circ)\, h_0^{p+1} + \mathcal{O}(h_0^{p+2}) + \tilde{V}(t_k^\circ, y) - V(t_k^\circ, y)$$

$$\Leftrightarrow \quad \alpha(t_k^\circ) = \frac{\tilde{V}(t_k^\circ, y) - V(t_k^\circ, y)}{h_0^p} + \mathcal{O}(h_0) \approx \frac{\tilde{V}(t_k^\circ, u) - V(t_k^\circ, u)}{h_0^p}$$

↑ u: Lsg des genaueren Verfahrens

liefert neue Testschrittweite h_1

$$h_1 = \left(\frac{tol}{\|\alpha(t_k^\circ)\|}\right)^{1/p} \approx \left(\frac{tol}{\|\tilde{V}(t_k^\circ, u) - V(t_k^\circ, u)\|}\right)^{1/p} \cdot h_0$$

bzw. numerisch sicherer: (Sicherheitsfaktor $\rho = 0{,}8 \dots 0{,}9$ um $\|\tau(t)\| \overset{!}{\leq} tol$ zu erreichen)

$$h_1 =: prop(h_0) := \min\left\{ \left(\frac{\rho \cdot tol}{\|\tilde{V}(t_k^\circ, u) - V(t_k^\circ, u)\| + \gamma}\right)^{1/p} \cdot h_0 \;,\; h_{max} \;,\; q \cdot h_0 \right\}$$

obere Schrittweitengrenze (Taylor!)

γ: z.B. 10^{-15} um Nulldivision zu vermeiden

z.B. $q = 5, 10$ um zu starke Vergrößerung zu vermeiden

Zeitschritt in schrittweitengesteuerten Algorithmus:

wähle Testschrittweite h (Vorschlag aus letztem Zeitschritt); $\varepsilon = \inf$

while $\varepsilon \leq tol$

$t_k = t_{k-1} + h$

berechne $u(t_k)$, $\varepsilon = \| \tilde{V}(t_k, u) - V(t_k, u) \|_\infty$, $h = prop(\varepsilon)$

end

Bem: wenn $\varepsilon \leq tol$ (zu genau) wird weiter gerechnet, aber der Vorschlag h für den nächsten Schritt ist dann größer

fehlt noch: Wahl der Verfahren Ziel: geringer Overhead durch Steuerung !

Idee: <u>eingebettete</u> Runge-Kutta-Verfahren

$$V: \begin{array}{c|c} c & A \\ \hline & b^T \end{array} \qquad \tilde{V}: \begin{array}{c|c} c & A \\ \hline & \tilde{b}^T \end{array}$$

Gleichungssystem identisch
⇔ geringe Zusatzkosten!

weitere Verbesserung mit Fehlberg Trick

$$c = \begin{pmatrix} \bar{c} \\ 1 \end{pmatrix} \qquad A = \begin{pmatrix} \bar{A} \\ \tilde{b}^T \end{pmatrix}$$

im GS wird

$g_s = F(t_{k-1} + h, \underbrace{u(t_{k-1}) + h \sum_{i=1}^{s} \tilde{b}_i g_i}_{})$ berechnet

dabei kommt $u(t_k)$ bereits vor

Notation: RK $p(q)$ — p: Ordnung des Verfahrens, q: Ordnung des Vergleichsverfahrens

Ordnung des Verfahrens mit dem weitergerechnet wird

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

← hiermit wird weitergerechnet
← hiermit wird verglichen

Beispiel RK 4(3) mit extrem geringen Overhead im Vergleich zu RK 4

0	0					
$\frac{1}{2}$	$\frac{1}{2}$	0				
$\frac{1}{2}$	0	$\frac{1}{2}$	0			
1	0	0	1	0		
1	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$	0	← Fehlberg Trick
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$	0	← Ordnung 4 (klassisches R.K)
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	0	$\frac{1}{6}$	← Ordnung 3

beachte: durch Fehlberg-Trick gleiche Anzahl F Auswertungen wie RK4

da $g_5^{(k+1)} = F(\underbrace{t_{k-1}+h}_{=t_k}, \underbrace{u(t_{k+1})+h\sum_{i=1}^{5}\tilde{b}_i g_i}_{u(t_k)}) = F(t_k, u(t_k)) = g_1^{(k)}$

d.h. g_1 ist g_5 des vorherigen Zeitschritts und muss nicht neu berechnet werden

außerdem muss $\frac{1}{6}g_1 + \frac{1}{3}g_2 + \frac{1}{3}g_3$ nur einmal berechnet werden