# Serie 4

1.

Bestimmen Sie die Fehlerkonstante für das explizite Euler-Verfahren

$$y_1 = y_0 + hf(x_0, y_0).$$

2. (P)*

Schreiben Sie eine Matlab-Funktion `[t,y]=RK4Adapt(t0,Tend,y0,Tol)` zur numerischen Lösung der Anfangswertaufgabe

$$\begin{aligned} \dot{y}(t) &= f(t,y) \\ y(t_0) &= y_0 \end{aligned}$$

mit dem «klassischen» vierstufigen Runge-Kutta-Verfahren. Verwenden Sie dazu die Richardson-Extrapolation mit den Parametern $\eta_{min} = 0.2, \eta_{max} = 5$ und einer vorgegebenen Fehlertoleranz `Tol > 0`.
Testen Sie das Programm mit der Anfangswertaufgabe

$$\begin{aligned} \dot{y}(t) &= -200ty^2(t),\ 0 \le t \le 3 \\ y(0) &= 1 \end{aligned}$$

mit der exakten Lösung

$$y(t) = \frac{1}{1 + 100t^2}.$$

Reduzieren Sie die `Tol`, bis der globale Fehler

$$\max_i |y(t_i) - y_i| < 10^{-9}$$

ist. Wieviele Zeitschritte benötigt das Verfahren dazu? Wieviele Schritte werden verworfen? Vergleiche den Aufwand (Anzahl Auswertungen der Funktion $f$) mit dem Aufwand bei konstanter Schrittweite, um eine vergleichbare (globale) Fehlerschranke zu erreichen.
*Hinweis: Die Struktur des Matlab-Programms könnte z. B. so aussehen:*

```
f.m:
function yp=f(t,y)
...
RK4.m:
function y=RK4(tk,yk,h) % siehe Serie 3
...
```

```
RK4Adapt.m:
function [t,y]=RK4Adapt(t0,Tend,y0,Tol)
Eta_min = 0.2;
...
t(1)=t0;
y(1,:)=y0;
k=1;
h=1;
while (t(k) < Tend),
     y1=RK4(t(k),y(k,:),h);
     y2=...
     y_hat=...
     err=...
     h_opt=...
     y2_hat=...
     if err < Tol
       k=k+1;
       t(k)=t(k-1)+2*h;
       y(k,:)=y2_hat;
     end
     h=h_opt;
end
```

3. (P)*

Sei $y_1 = y_0 + h(b_1 k_1 + \ldots + b_s k_s)$ ein $s$-stufiges RK-Verfahren der Ordnung $p$. Hier betrachen wir ein sog. *eingebettetes RK-Verfahren* der Ordnung $\hat{p} < p$, das die gleichen Werte von $f$ benutzt, d.h.

$$\hat{y}_1 = y_0 + h(\hat{b}_1 k_1 + \ldots + \hat{b}_s k_s). \quad (*)$$

Um mehr Freiheiten zu erhalten, nimmt man noch den Term $f(x_1, y_1)$ dazu in $(*)$:

$$\hat{y}_1 := y_0 + h(\hat{b}_1 k_1 + \ldots + \hat{b}_s k_s + \hat{b}_{s+1} f(x_1, y_1)). \quad (**)$$

$(a)$ Warum kostet dieser Term gar nichts?

Wir benutzen das Verfahren $(**)$, um eine Abschätzung des lokalen Fehlers zu bekommen:

$$err \approx ||y_1 - \hat{y}_1|| := \sqrt{\frac{1}{n}\sum_{i=1}^{n}\Big(\frac{y_{i1} - \hat{y}_{i1}}{sc_i}\Big)^2},$$

mit $sc_i = 1 + max(|y_{i0}|, |y_{i1}|)$. Also eine Mischung zwischen relativem Fehler und absolutem Fehler.

$(b)$ Wir betrachten jetzt das RK-Verfahren mit $p = s = 4$ (siehe Vorlesung und Serie 3). Wir suchen ein *eingebettetes RK-Verfahren* der Ordnung $\hat{p} = 3$. Geben Sie die

Ordnungsbedingung von diesem Verfahren (wir setzen $a_{5,i} = b_i$ für $i = 1, \ldots, 4$).

$(c)$ Wir bekommen ein lineares System von 4 Gleichungen mit 5 Unbekannten. Wählen Sie $\hat{b}_5 = 1/6$ und lösen Sie das System.

$(d)$ Für eine gegebene Toleranz `Tol` bestimmen Sie (wie in der Vorlesung) eine optimale Schrittweite $h_{opt}$.

$(e)$ Wenden Sie dieses eingebette RK-Verfahren mit `Tol=10^(-8)` an, um das folgende Problem (Van der Pol-Gleichungen) zu lösen:

$$\begin{aligned} \ddot{y}(t) &= 8(1 - y(t)^2)\dot{y}(t) - y(t), \text{ für } t \in [0, 30] \\ y(0) &= 2 \\ \dot{y}(0) &= 0. \end{aligned}$$