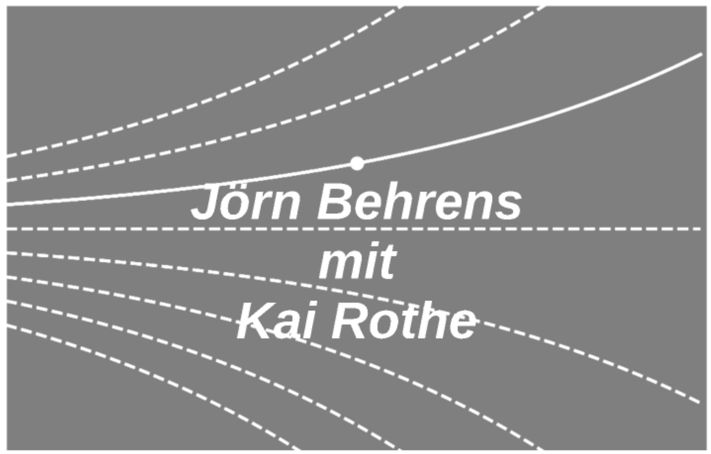
Differentialgleichungen I



Numerische Verfahren

Buch Kapitel 6.10

Einführung

Motivation:

- Wir haben eine Reihe von analytischen Lösungsmethoden kennen gelernt. Was aber, wenn die DGL zu kompliziert?
- Mit den bisherigen Sätzen können wir trotzdem Aussagen über die Läsbarkeit machen, allerdings vielleicht nicht die Lösung berechnen.
- Idee: Finde eine (Näherungs-) Lösung mit Hilfe numerischer Verfahren!

Idee:

- Betrachte DGL 1. Ordnung:

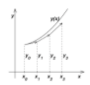
$$y'(x) = f(x, y(x))$$

- Anfangswerte seien

$$y(x_0) = y_0$$

- Beobachtung: In (x_0, y_0) ist mit $y'(x_0) = F(x_0, y_0)$ die Steigung der Lösungsfunktion y bekannt!
- Idee: Nähere die Lösung mit Hilfe der Tangente (Linearisierung) an.

Verfahren: (Integrationsmethode von Euler)



- Betrachte Anfangswertproblem 1. Ordnung:

$$y'(x) = f(x, y(x)), \quad y(x_0) = y_0$$

- Definiere äquidistante Stützstellen zur Schrittweite h:

$$x_k = x_0 + kh \quad (k = 1, 2, \ldots).$$

- Dann erhält man eine Näherung y_k an die exakten Lösungswerte $y(x_k)$ durch

$$y_{k+1} = y_k + hf(x_k, y_k), \quad k = 0, 1, 2, \ldots$$

- Die so definierte Methode heißt Integrationsmethode von Euler.
  Andere Bezeichnungen: *Polygonzugmethode*, *Euler-Verfahren*.

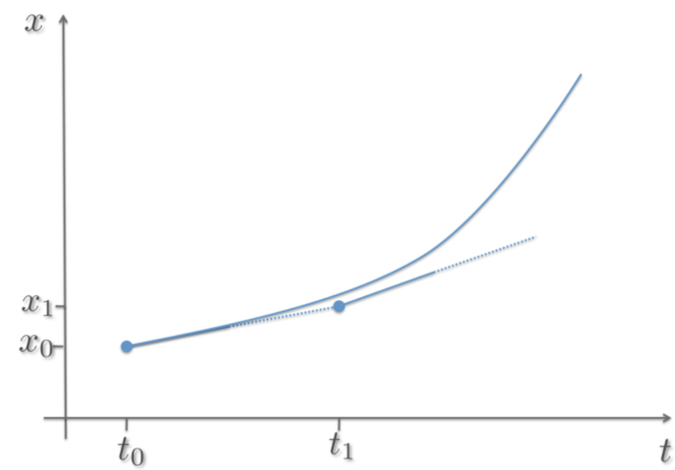
Bemerkungen:

- Das Polygonzugverfahren ist nur für sehr kleine h zu verwenden.
- Das Verfahren ist das einfachste *explizite Einschritt-Verfahren*.

Motivation:

- Wir haben eine Reihe von analytischen Lösungsmethoden kennen gelernt. Was aber, wenn die DGL zu kompliziert?
- Mit den bisherigen Sätzen können wir trotzdem Aussagen über die Läsbarkeit machen, allerdings vielleicht nicht die Lösung berechnen.
- Idee: Finde eine (Näherungs-) Lösung mit Hilfe numerischer Verfahren!

Idee:



- Betrachte DGL 1. Ordnung:

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

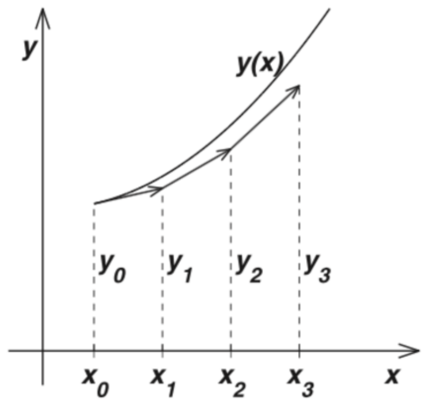
- Anfangswerte seien

$$y(x_0) = y_0$$

- Beobachtung: In (x_0, y_0) ist mit $y'(x_0) = F(x_0, y_0)$ die Steigung der Lösungsfunktion y bekannt!

- Idee: Nähere die Lösung mit Hilfe der Tangente (Linearisierung) an.

Verfahren: (Integrationsmethode von Euler)



- Betrachte Anfangswertproblem 1. Ordnung:

$$y'(x) = f(x, y(x)), \quad y(x_0) = y_0$$

- Definiere äquidistante Stützstellen zur Schrittweite h:

$$x_k = x_0 + kh \quad (k = 1, 2, \ldots).$$

- Dann erhält man eine Näherung y_k an die exakten Lösungswerte $y(x_k)$ durch

$$y_{k+1} = y_k + hf(x_k, y_k), \quad k = 0, 1, 2, \ldots$$

- Die so definierte Methode heißt Integrationsmethode von Euler.
  Andere Bezeichnungen: *Polygonzugmethode*, *Euler-Verfahren*.

Bemerkungen:

- Das Polygonzugverfahren ist nur für sehr kleine h zu verwenden.
- Das Verfahren ist das einfachste *explizite Einschritt-Verfahren*.

Diskretisierungsfehler

Notationen:

- Verwende die allgemeine (implizite) Form der Rechenvorschrift

$$y_{k+1} = y_k + h\Phi(x_k, y_k, y_{k+1}, h).$$

- Hängt Φ nur von x_k, y_k und h ab, so ist eine *explizite* Rechenvorschrift gegeben.
- Hängt Φ auch von y_{k+1} ab, so muss in jedem Zeitschritt i.A. eine nichtlineare Gleichung (*implizit*) gelöst werden.
- Beispiel: Das Euler-Verfahren verwendet $\Phi(x_k, y_k, y_{k+1}, h) = f(x_k, y_k)$.

Definition: (lokaler Diskretisierungsfehler)
Der lokale Diskretisierungsfehler an der Stelle x_{k+1} ist definiert als

$$d_{k+1} := y(x_{k+1}) - y(x_k) - h\Phi(x_k, y(x_k), y(x_{k+1}), h).$$

Bemerkung: Es handelt sich also um den Fehler, der in einem einzelnen Schritt $x_k \to x_{k+1}$ gegenüber der exakten Lösung verursacht wird.

Definition: (Fehlerordnung)
Ein Einschrittverfahren $y_{k+1} = y_k + h\Phi(x_k, y_k, y_{k+1}, h)$ besitzt die Fehlerordnung p, falls für den lokalen Diskretisierungsfehler d_k gilt:

$$\max_{1 \leq k \leq n} |d_k| \leq D = \text{const.} \cdot h^{p+1} = \mathcal{O}(h^{p+1}).$$

Satz: (Abschätzung des globaler Diskretisierungsfehlers)
Für den globalen Fehler g_n an der festen Stelle $x_n = x_0 + nh$ gilt

- Für eine explizite Einschrittmethode:

$$|g_n| \leq \frac{D}{hL}\left(e^{nhL} - 1\right) \leq \frac{D}{hL} e^{nhL}.$$

- Für eine implizite Methode:

$$|g_n| \leq \frac{D}{hK(1-hL)}\left(e^{nhL} - 1\right) \leq \frac{D}{hK(1-hL)} e^{nhL}.$$

Dabei ist D eine obere Schranke für den lokalen Disrektisierungsfehler, L eine (Lipschitz-) Konstante die von der Rechenvorschrift Φ abhängt, und K eine Konstante.

Bemerkungen:

- Für Lösungsfunktionen mit genügenden Eigenschaften (z.B. zweimal stetig diff'bar) kann man zeigen, dass $D \leq \frac{1}{2}h^2 M$ gilt, wobei M eine obere Schranke von y'' ist.
- Damit ergibt sich beispielsweise für das Polygonzugverfahren:

$$|g_n| \leq h\frac{M}{2L} e^{L(x_n - x_0)} =: hC, \quad C \in \mathbb{R}.$$

- Interpretation: Bei fester Stelle x_n und verkleinerter Schrittweite $h = \frac{x_n - x_0}{n}$ (also zunehmendem n) nimmt der globale Fehler proportional zur Schrittweite ab.

Notationen:

- Verwende die allgemeine (implizite) Form der Rechenvorschrift

$$y_{k+1} = y_k + h\Phi(x_k, y_k, y_{k+1}, h).$$

- Hängt Φ nur von x_k, y_k und h ab, so ist eine *explizite* Rechenvorschrift gegeben.
- Hängt Φ auch von y_{k+1} ab, so muss in jedem Zeitschritt i.A. eine nichtlineare Gleichung (*implizit*) gelöst werden.
- Beispiel: Das Euler-Verfahren verwendet $\Phi(x_k, y_k, y_{k+1}, h) = f(x_k, y_k)$.

Definition: (lokaler Diskretisierungsfehler)
Der lokale Diskretisierungsfehler an der Stelle x_{k+1} ist definiert als

$$d_{k+1} := y(x_{k+1}) - y(x_k) - h\Phi(x_k, y(x_k), y(x_{k+1}), h).$$

Bemerkung: Es handelt sich also um den Fehler, der in einem einzelnen Schritt $x_k \to x_{k+1}$ gegenüber der exakten Lösung verursacht wird.

Definition: (globaler Diskretisierungsfehler)
Der globale Diskretisierungsfehler an der Stelle x_k ist definiert als

$$g_k := y(x_k) - y_k.$$

Bemerkung: Es handelt sich also um den Fehler zwischen der exakten Lösung $y(x_k)$ und der vom numerischen Lösungsverfahren berechneten Lösung y_k.

1

Satz: (Abschätzung des globaler Diskretisierungsfehlers)
Für den globalen Fehler g_n an der festen Stelle $x_n = x_0 + nh$ gilt

- Für eine explizite Einschrittmethode:

$$|g_n| \leq \frac{D}{hL}\left(e^{nhL} - 1\right) \leq \frac{D}{hL} e^{nhL}.$$

- Für eine implizite Methode:

$$|g_n| \leq \frac{D}{hK(1-hL)}\left(e^{nhL} - 1\right) \leq \frac{D}{hK(1-hL)} e^{nhL}.$$

Dabei ist D eine obere Schranke für den lokalen Disrektisierungsfehler, L eine (Lipschitz-) Konstante die von der Rechenvorschrift Φ abhängt, und K eine Konstante.

Bemerkungen:

- Für Lösungsfunktionen mit genügenden Eigenschaften (z.B. zweimal stetig diff'bar) kann man zeigen, dass $D \leq \frac{1}{2}h^2 M$ gilt, wobei M eine obere Schranke von y'' ist.

- Damit ergibt sich beispielsweise für das Polygonzugverfahren:

$$|g_n| \leq h \frac{M}{2L} e^{L(x_n - x_0)} =: hC, \quad C \in \mathbb{R}.$$

- Interpretation: Bei fester Stelle x_n und verkleinerter Schrittweite $h = \frac{x_n - x_0}{n}$ (also zunehmendem n) nimmt der globale Fehler proportional zur Schrittweite ab.

Definition: (Fehlerordnung)
Ein Einschrittverfahren $y_{k+1} = y_k + h\Phi(x_k, y_k, y_{k+1}, h)$ besitzt die Fehlerordnung p, falls für den lokalen Diskretisierungsfehler d_k gilt:

$$\max_{1\leq k\leq n} |d_k| \leq D = \text{const.} \cdot h^{p+1} = \mathcal{O}(h^{p+1}).$$

Folgerung:
Der globale Fehler g_n einer expliziten Methode mit Fehlerordnung p ist beschränkt durch

$$|g_k| \leq \frac{\text{const.}}{L} e^{nhL} \cdot h^p = \mathcal{O}(h^p).$$

Trapezmethode

Idee: (Methode aus Richardson-Extrapolation)

- Ziel: Methode mit Fehlerordnung $p > 1$.
- Berechne y_n mit Schrittweite $h_1 = h$ und dieselbe Stelle y_{2n} mit $h_2 = \frac{h}{2}$, erhalte

$$\begin{aligned} y_n &\approx y(x) + c_1 h + \mathcal{O}(h^2) \\ y_{2n} &\approx y(x) + c_1 \frac{h}{2} + \mathcal{O}(h^2) \end{aligned}$$

- Richardson-Extrapolation ergibt dann

$$\tilde{y} = 2y_{2n} - y_n \approx y(x) + \mathcal{O}(h^2).$$

Herleitung: (Verbesserte Polygonzug-Methode)

- Statt Anwendung der Richardson-Extrapolation auf das Ergebnis von zwei Polygonzug-Verfahren (PZV), wende Extrapolation in jedem Schritt an!
- **Normalschritt**: PZV mit h:

$$y^{(1)}_{k+1} = y_k + hf(x_k, y_k)$$

- **Doppelschritt**: PZV zweimal ausgeführt mit $\frac{h}{2}$:

$$\begin{aligned} y^{(2)}_{k+\frac{1}{2}} &= y_k + \frac{h}{2} f(x_k, y_k) \\ y^{(2)}_{k+1} &= y^{(2)}_{k+\frac{1}{2}} + \frac{h}{2} f(x_k + \frac{h}{2}, y^{(2)}_{k+\frac{1}{2}}) \end{aligned}$$

- **Richardson-Extrapolation**:

$$\begin{aligned} y_{k+1} &= 2y^{(2)}_{k+1} - y^{(1)}_{k+1} \\ &= 2y^{(2)}_{k+\frac{1}{2}} + hf(x_k + \frac{h}{2}, y^{(2)}_{k+\frac{1}{2}}) - y_k - hf(x_k, y_k) \\ &= 2y_k + hf(x_k, y_k) + hf(x_k + \frac{h}{2}, y^{(2)}_{k+\frac{1}{2}}) - y_k - hf(x_k, y_k) \\ &= y_k + hf(x_k + \frac{h}{2}, y_k + \frac{h}{2} f(x_k, y_k)). \end{aligned}$$

Algorithmus: (Verbesserte Polygonzug-Methode)

$$\begin{aligned} k_1 &= f(x_k, y_k) \\ k_2 &= f(x_k + \frac{h}{2}, y_k + \frac{h}{2} k_1) \\ y_{k+1} &= y_k + hk_2 \end{aligned}$$

Bemerkung: die Verbesserte Polygonzugmethode verwendet

$$\Phi(x_k, y_k, y_{k+1}, h) = f(x_k + \frac{h}{2}, y_k + \frac{h}{2} f(x_k, y_k)).$$

Idee: (Methode von Heun)

- Iteriere in der Fixpunktiteration lediglich einen Schritt:

$$\begin{aligned} y^{(p)}_{k+1} &= y_k + f(x_k, y_k) \\ y_{k+1} &= y_k + \frac{h}{2} \left[f(x_k, y_k) + f(x_{k+1}, y^{(p)}_{k+1}) \right]. \end{aligned}$$

- D.h. Euler Methode ermittelt Prediktorwert $y^{(p)}_{k+1}$, Trapezmethode bestimmt *korrigierten* Wert y_{k+1}.
- Das folgende Heun-Verfahren ist eine Prädiktor-Korrektor-Methode:

$$\begin{aligned} k_1 &= f(x_k, y_k) \\ k_2 &= f(x_k + h, y_k + hk_1) \\ y_{k+1} &= y_k + \frac{h}{2}[k_1 + k_2] \end{aligned}$$

Idee: (Trapezmethode)

- Neue Idee: Integriere die DGL $y'(x) = f(x, y(x))$ für einen Zeitschritt und erhalte

$$y(x_{k+1}) - y(x_k) = \int_{x_k}^{x_{k+1}} f(x, y(x))\, dx$$

- Löse das Integral mit Hilfe einer (numerischen) Quadraturformel (hier Trapezregel):

$$y_{k+1} = y_k + \frac{h}{2} [f(x_k, y_k) + f(x_{k+1}, y_{k+1})]$$

- Erhalte implizites Verfahren, die Trapezmethode.

Bemerkungen: (Trapezmethode)

- Da die implizite Lösung meist ein nichtlineares Problem enthält, löse mittels Fixpunkt-Iteration:

$$\begin{aligned} y^{(0)}_{k+1} &= y_k + f(x_k, y_k) \\ y^{(s+1)}_{k+1} &= y_k + \frac{h}{2} \left[f(x_k, y_k) + f(x_{k+1}, y^{(s)}_{k+1}) \right] \quad s = 0, 1, 2, \dots \end{aligned}$$

- Erhalte Konvergenz, falls $|f(x, y) - f(x, y^*)| \le L|y - y^*|$ (Lipschitz-stetig) und $\frac{hL}{2} < 1$ (Banachscher Fixpunktsatz).

Idee: (Methode aus Richardson-Extrapolation)

- Ziel: Methode mit Fehlerordnung $p > 1$.
- Berechne y_n mit Schrittweite $h_1 = h$ und dieselbe Stelle y_{2n} mit $h_2 = \frac{h}{2}$, erhalte

$$\begin{aligned} y_n &\approx y(x) + c_1 h + \mathcal{O}(h^2) \\ y_{2n} &\approx y(x) + c_1 \frac{h}{2} + \mathcal{O}(h^2) \end{aligned}$$

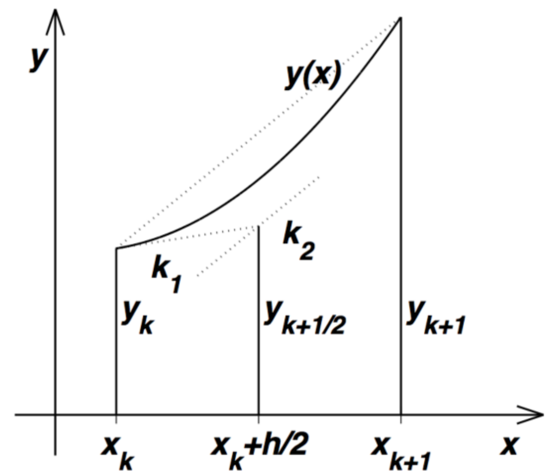
- Richardson-Extrapolation ergibt dann

$$\tilde{y} = 2y_{2n} - y_n \approx y(x) + \mathcal{O}(h^2).$$

Herleitung: (Verbesserte Polygonzug-Methode)

- Statt Anwendung der Richardson-Extrapolation auf das Ergebnis von zwei Polygonzug-Verfahren (PZV), wende Extrapolation in jedem Schritt an!
- **Normalschritt**: PZV mit h:

$$y_{k+1}^{(1)} = y_k + hf(x_k, y_k)$$



- **Doppelschritt**: PZV zweimal ausgeführt mit $\frac{h}{2}$:

$$\begin{aligned} y_{k+\frac{1}{2}}^{(2)} &= y_k + \frac{h}{2} f(x_k, y_k) \\ y_{k+1}^{(2)} &= y_{k+\frac{1}{2}}^{(2)} + \frac{h}{2} f(x_k + \frac{h}{2}, y_{k+\frac{1}{2}}^{(2)}) \end{aligned}$$

- **Richardson-Extrapolation**:

$$\begin{aligned} y_{k+1} &= 2y_{k+\frac{1}{2}}^{(2)} - y_{k+1}^{(1)} \\ &= 2y_{k+\frac{1}{2}}^{(2)} + hf(x_k + \frac{h}{2}, y_{k+\frac{1}{2}}^{(2)}) - y_k - hf(x_k, y_k) \\ &= 2y_k + hf(x_k, y_k) + hf(x_k + \frac{h}{2}, y_{k+\frac{1}{2}}^{(2)}) - y_k - hf(x_k, y_k) \\ &= y_k + hf(x_k + \frac{h}{2}, y_k + \frac{h}{2} f(x_k, y_k)). \end{aligned}$$

Algorithmus: (Verbesserte Polygonzug-Methode)

$$
\begin{aligned}
k_1 &= f(x_k, y_k) \\
k_2 &= f(x_k + \frac{h}{2}, y_k + \frac{h}{2} k_1) \\
y_{k+1} &= y_k + h k_2
\end{aligned}
$$

Bemerkung: die Verbesserte Polygonzugmethode verwendet

$$\Phi(x_k, y_k, y_{k+1}, h) = f(x_k + \frac{h}{2}, y_k + \frac{h}{2} f(x_k, y_k)).$$

Idee: (Trapezmethode)

- Neue Idee: Integriere die DGL $y'(x) = f(x, y(x))$ für einen Zeitschritt und erhalte
$$y(x_{k+1}) - y(x_k) = \int_{x_k}^{x_{k+1}} f(x, y(x))\; dx$$
- Löse das Integral mit Hilfe einer (numerischen) Quadraturformel (hier Trapezregel):
$$y_{k+1} = y_k + \frac{h}{2}\left[f(x_k, y_k) + f(x_{k+1}, y_{k+1})\right]$$
- Erhalte implizites Verfahren, die Trapezmethode.

Bemerkungen: (Trapezmethode)

- Da die implizite Lösung meist ein nichtlineares Problem enthält, löse mittels Fixpunkt-Iteration:

$$\begin{aligned} y_{k+1}^{(0)} &= y_k + f(x_k, y_k) \\ y_{k+1}^{(s+1)} &= y_k + \frac{h}{2}\left[f(x_k, y_k) + f(x_{k+1}, y_{k+1}^{(s)})\right] \quad s = 0, 1, 2, \ldots \end{aligned}$$

- Erhalte Konvergenz, falls $|f(x, y) - f(x, y^*)| \leq L|y - y^*|$ (Lipschitz-stetig) und $\frac{hL}{2} < 1$ (Banachscher Fixpunktsatz).

Idee: (Methode von Heun)

- Iteriere in der Fixpunktiteration lediglich einen Schritt:

$$\begin{aligned} y_{k+1}^{(p)} &= y_k + f(x_k, y_k) \\ y_{k+1} &= y_k + \frac{h}{2}\left[f(x_k, y_k) + f(x_{k+1}, y_{k+1}^{(p)})\right]. \end{aligned}$$

- D.h. Euler Methode ermittelt Prediktorwert $y_{k+1}^{(p)}$, Trapezmethode bestimmt *korrigierten* Wert y_{k+1}.
- Das folgende Heun-Verfahren ist eine Prädiktor-Korrektor-Methode:

$$\begin{aligned} k_1 &= f(x_k, y_k) \\ k_2 &= f(x_k + h, y_k + hk_1) \\ y_{k+1} &= y_k + \frac{h}{2}[k_1 + k_2] \end{aligned}$$

Runge-Kutta Verfahren

Vorbemerkungen:

- Die Methode von Heun und das verbesserte Polygonzugverfahren sind Beispiele für explizite zweistufige Runge-Kutta Verfahren mit Fehlerordnung 2.
- Für die Beschreibung von Runge-Kutta Verfahren mit höheren Fehlerordnungen starten wir von der Integralgleichung

$$y(x_{k+1}) - y(x_k) = \int_{x_k}^{x_{k+1}} f(x, y(x))\, dx.$$

- Verwende zur Berechnung des Integrals eine allgemeine Quadraturformel mit 3 Stützstellen im Intervall $[x_k, x_{k+1}]$. Das führt auf den Ansatz:

$$y_{k+1} = y_k + h[c_1 f(\xi_1, y(\xi_1)) + c_2 f(\xi_2, y(\xi_2)) + c_3 f(\xi_3, y(\xi_3))].$$

- Dabei seien $c_1 + c_2 + c_3 = 1$, ξ_i die Stützstellen.

Bestimmung der Stützstellen und Werte:

- Verwende Stützstellen

$$\xi_1 = x_k, \quad \xi_2 = x_k + a_2 h, \quad \xi_3 = x_k + a_3 h, \quad 0 < a_2, a_3 \leq 1.$$

- Wegen $\xi_1 = x_k$ ist $y(\xi_1) = y_k$.
- Für $y(\xi_2)$ und $y(\xi_3)$ verwende Prädiktor-Ansatz:

$$\begin{aligned} y(\xi_2): \quad y_2^* &= y_k + hb_{21} f(x_k, y_k) \\ y(\xi_3): \quad y_3^* &= y_k + hb_{31} f(x_k, y_k) + hb_{32} f(x_k + a_2 h, y_2^*). \end{aligned}$$

- Man erhält Parameter $a_1, a_2, b_{21}, b_{31}, b_{32}, c_1, c_2, c_3$, die so gewählt werden, dass eine optimale Fehlerordnung erreicht wird.

Algorithmus: (3-Stufiges Runge-Kutta Verfahren)

$$\begin{aligned} k_1 &= f(x_k, y_k) \\ k_2 &= f(x_k + a_2 h, y_k + hb_{21} k_1) \\ k_3 &= f(x_k + a_3 h, y_k + h(b_{31} k_1 + b_{32} k_2)) \\ y_{k+1} &= y_k + h[c_1 k_1 + c_2 k_2 + c_3 k_3]. \end{aligned}$$

Beispiel: Das Heun-Verfahren dritter Ordnung erhält man durch die folgende Wahl der Parameter:

$$a_1 = \frac{1}{3},\ a_2 = \frac{2}{3},\ c_1 = \frac{1}{4},\ c_2 = 0,\ c_3 = \frac{3}{4}, b_{32} = \frac{2}{3}, b_{31} = a_3 - b_{32} = 0.$$

Vorbemerkungen:

- Die Methode von Heun und das verbesserte Polygonzugverfahren sind Beispiele für explizite zweistufige Runge-Kutta Verfahren mit Fehlerordnung 2.

- Für die Beschreibung von Runge-Kutta Verfahren mit höheren Fehlerordnungen starten wir von der Integralgleichung

$$y(x_{k+1}) - y(x_k) = \int_{x_k}^{x_{k+1}} f(x, y(x))\, dx.$$

- Verwende zur Berechnung des Integrals eine allgemeine Quadraturformel mit 3 Stützstellen im Intervall $[x_k, x_{k+1}]$. Das führt auf den Ansatz:

$$y_{k+1} = y_k + h[c_1 f(\xi_1, y(\xi_1)) + c_2 f(\xi_2, y(\xi_2)) + c_3 f(\xi_3, y(\xi_3))].$$

- Dabei seien $c_1 + c_2 + c_3 = 1$, ξ_i die Stützstellen.

Bestimmung der Stützstellen und Werte:

- Verwende Stützstellen

$$\xi_1 = x_k, \quad \xi_2 = x_k + a_2 h, \quad \xi_3 = x_k + a_3 h, \quad 0 < a_2, a_3 \leq 1.$$

- Wegen $\xi_1 = x_k$ ist $y(\xi_1) = y_k$.
- Für $y(\xi_2)$ und $y(\xi_3)$ verwende Prädiktor-Ansatz:

$$\begin{aligned} y(\xi_2): \quad y_2^* &= y_k + hb_{21} f(x_k, y_k) \\ y(\xi_3): \quad y_3^* &= y_k + hb_{31} f(x_k, y_k) + hb_{32} f(x_k + a_2 h, y_2^*). \end{aligned}$$

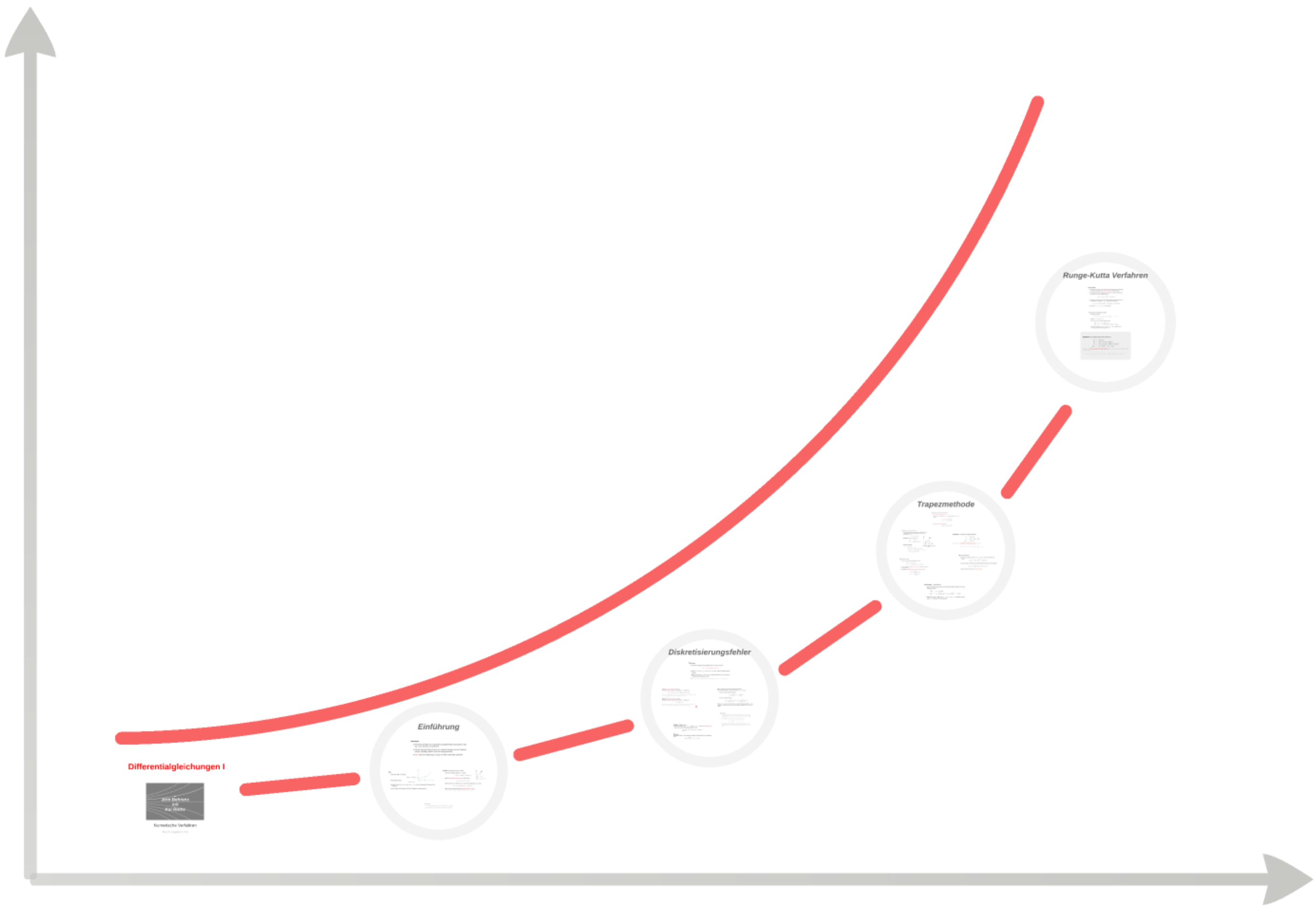
- Man erhält Parameter $a_1, a_2, b_{21}, b_{31}, b_{32}, c_1, c_2, c_3$, die so gewählt werden, dass eine optimale Fehlerordnung erreicht wird.

Algorithmus: (3-Stufiges Runge-Kutta Verfahren)

$$
\begin{aligned}
k_1 &= f(x_k, y_k) \\
k_2 &= f(x_k + a_2 h, y_k + h b_{21} k_1) \\
k_3 &= f(x_k + a_3 h, y_k + h(b_{31} k_1 + b_{32} k_2)) \\
y_{k+1} &= y_k + h[c_1 k_1 + c_2 k_2 + c_3 k_3].
\end{aligned}
$$

Beispiel: Das Heun-Verfahren dritter Ordnung erhält man durch die folgende Wahl der Parameter:

$$
a_1 = \frac{1}{3},\; a_2 = \frac{2}{3},\; c_1 = \frac{1}{4},\; c_2 = 0,\; c_3 = \frac{3}{4}, b_{32} = \frac{2}{3}, b_{31} = a_3 - b_{32} = 0.
$$


Differentialgleichungen I
Einführung
Diskretisierungsfehler
Trapezmethode
Runge-Kutta Verfahren