3 Kinematik und Dynamik

Computer Animation & Kollisionsdetektion

Wintersemester 2011/12

3.6 Integrationsverfahren

- (Numerische) Lösungsverfahren für *gewöhnliche Differentialgleichung* (*Anfangswertproblem*):

$$y'(t) = F\big(t, y(t)\big), \quad y(t_0) = y_0$$

- Ziel: Suche zu vorgegebenen Zeitpunkten $t_1 < t_2 < t_3 < t_4 < \ldots$ Näherungen y_n für die exakte Lösung $y(t_n)$
  Spezialfall: *äquidistante Schrittweite*: $t_n = t_0 + nh$

- Es gibt verschiedene Ansätze:
  - Explizite und implizite
  - Einschritt-Verfahren und Mehrschritt-Verfahren

	explizit	implizit
Einschritt	$y_{n+1} = \varphi_F(t_n, y_n)$	$\Phi_F(t_n, t_{n+1}, y_n, y_{n+1}) = 0$
Mehrschritt	$y_{n+1} = \psi_F(t_n, t_{n-1}, \ldots, y_n, y_{n-1}, \ldots)$	$\Psi_F(t_{n+1}, t_{n-1}, \ldots, y_{n+1}, y_n, y_{n-1}, \ldots) = 0$

- Einfache Integrationsverfahren für Systeme 1.Ordnung für
  $y'(t) = F(t, y(t))\ ,\quad y'(0) = y_0$

- **Explizites Eulerverfahren** (für feste Schrittweite $h>0$)
  $t_0 = 0\ ;\ y_0 = y_0\ ;$
  do while $\{\ t_i < t_{end}\}$
  $\quad t_{i+1} = t_i + h\ ,\quad y_{i+1} = y_i + h\cdot F(t_i,y_i)$

- **Implizites Eulerverfahren** (für feste Schrittweite $h>0$)
  $t_0 = 0\ ;\ y_0 = y_0\ ;$
  do while $\{\ t_i < t_{end}\}$
  $\quad t_{i+1} = t_i + h\ ,\quad y_{i+1} = y_i + h\cdot F(t_{i+1},y_{i+1})$

- Beim impliziten Verfahren muss ein GS gelöst werden !!

- besser: Heun explizit
  - bestimme Ableitung am Punkt (t_i,y_i) : $k_1 = F(t_i,y_i)$
  - mache Eulerschritt
  - bestimme Ableitung am Zielpunkt : k_2
  - mache einen Schritt mit Ableitung $(k_1+k_2)/2$
- PseudoCode:
  - **do while** $\{ t_i < t_{end} \}$
    $t_{i+1} = t_i + h$,
    $k_1 = F(t_i, y_i)$, $k_2 = F(t_{i+1}, y_i + hk_1)$
    $y_{i+1} = y_i + h \cdot (k_1+k_2)/2$
- oder: Heun implizit

$$y_{i+1} = y_i + h \cdot (F(t_i,y_i) + F(t_{i+1},y_{i+1}))/2$$

- noch besser: Runge-Kutta 4. Ordnung
  - $k_1 = F(t_i, y_i)$
  - $k_2 = F(t_i + h/2, y_i + h/2\ k_1)$
  - $k_3 = F(t_i + h/2, y_i + h/2\ k_2)$
  - $k_4 = F(t_i + h, y_i + h\ k_3)$
  - $y_{i+1} = y_i + h(k_1 + 2k_2 + 2k_3 + k_4)/6$

3.6 Integrationsverfahren

- lokaler Fehler:
  Fehler, der in einem Schritt gemacht wird
- globaler Fehler : (→ Konvergenzordnung-Ordnung)
  Fehler, der bis zum Zielpunkt gemacht wird.
  Unter guten Bedingungen („Stabilität“) gilt:
  Konsistenzordnung = Konvergenzordnung
- Bsp. Euler:
  - man berücksichtigt den linearen Anteil
  - es bleibt quadratischer Fehler
  - → lokaler Fehler Euler ist $O(h^2)$
  - Zielpunkt sei x_{end}, d.h. man braucht x_{end}/h Schritte
  - wenn man also h halbiert, hat man den Fehler pro Schritt geviertelt, braucht aber doppelt soviele Schritte
    → globaler Fehler $O(h)$

- Fehlerordnung: falls Fehler = $O(h^p)$ dann Ordnung p ;

- Fehlerordnung - im günstigsten Fall (bei **Stabilität**!)

  Ordnung(globale Fehler) = Ordnung(lokaler Fehler)-1

	lokal	global
Euler	$O(h^2)$	$O(h)$
Heun	$O(h^3)$	$O(h^2)$
Runge-Kutta	$O(h^5)$	$O(h^4)$

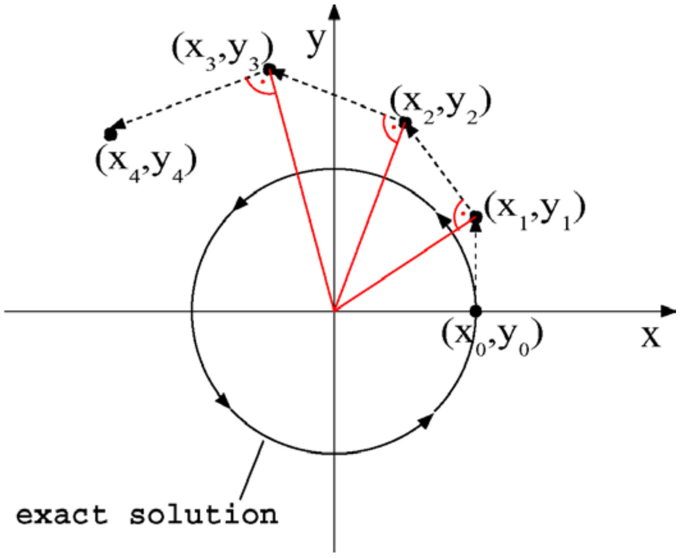
3.6 Integrationsverfahren

- bisher: explizite Verfahren
- Alternative: implizite Verfahren (sind i.d.R. stabiler)
- impliziter Euler:
  - suche Zielpunkt (t_{i+1},y_{i+1}), der eine Steigung besitzt, die der Verbindung von (t_i,y_i) nach (t_{i+1},y_{i+1}) entspricht
  - math.: $y_{i+1} = y_i + hF(t_{i+1},y_{i+1})$
  - Problem: dies erfordert Suche
    - Lösen eines Gleichungssystems
    - oder Iteration (Predictor – Corrector)
- ähnlich für Heun und Runge-Kutta
- warum implizit?
  $\rightarrow$ stabiler (d.h. globale Fehlerordnung = lokale -1 auch
  auch bei ‘großen‘ Schrittweiten)

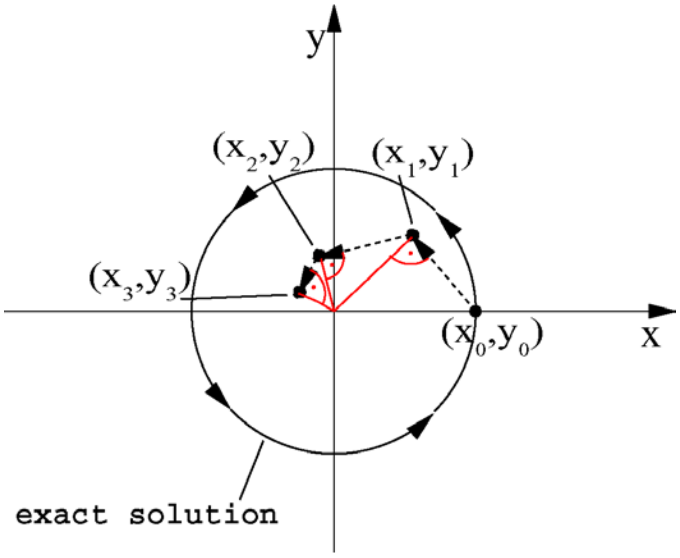
- 2D-Beispiel:

$$\frac{d}{dt}\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = F(t,[x(t), y(t)]) = \begin{bmatrix} -y(t) \\ x(t) \end{bmatrix}$$

exakte Lösung ist Kreisbahn $[R\cdot\cos(t-t_0), R\cdot\sin(t-t_0)]$



explizit Euler



implizit Euler

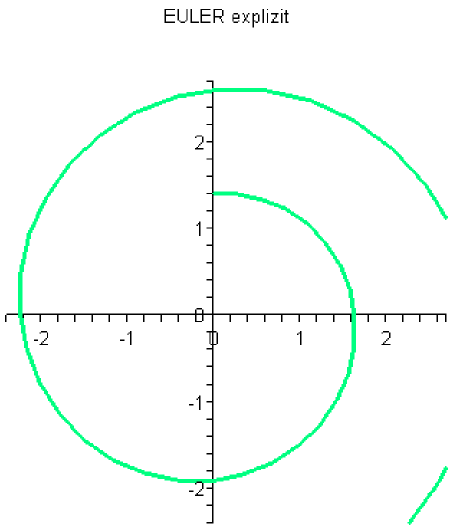
- Prädiktor-Korrektor-Methode für implizite Verfahren

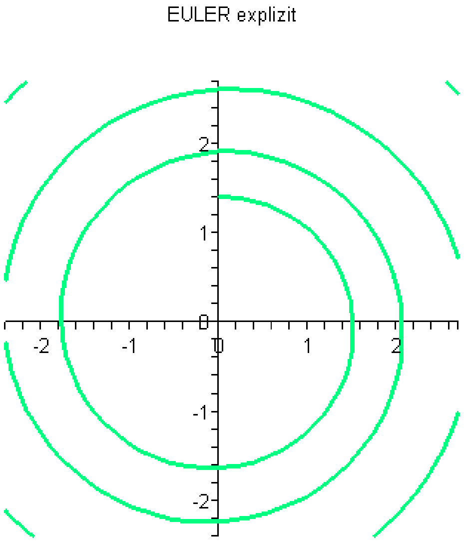
- Iteration für impliziten Euler
  - $y_{i+1}^{(0)} = y_i + h \cdot F(t_i, y_i)$ // (Prädiktor)
    do until convergence (k=0,1,2,…)
    $y_{i+1}^{(k+1)} = y_i + h \cdot F(t_{i+1}, y_{i+1}^{(k)})$ // (Korrektor)

- Iteration für impliziten Heun
  - $\mathbf{k_1 = F(t_i, y_i)}$
    $\mathbf{k_2 = F(t_{i+1}, y_i + hk_1)}$
    $\mathbf{y_{i+1}^{(0)} = y_i + h\ (k_1+k_2)/2}$ // (Prädiktor)
    do until convergence (k=0,1,2,…)
    $\mathbf{y_{i+1}^{(k+1)} = y_i + h\ (k_1 + F(t_{i+1}, y_{i+1}^{(k)}))/2}$ // (Korrektor)

3.6 Beispiel: explizites Euler : O(τ)

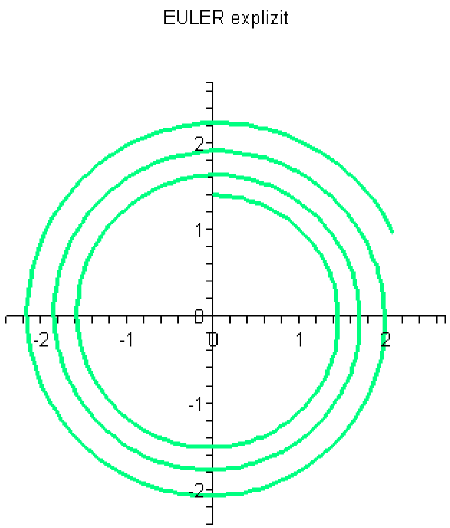
τ = 0.200

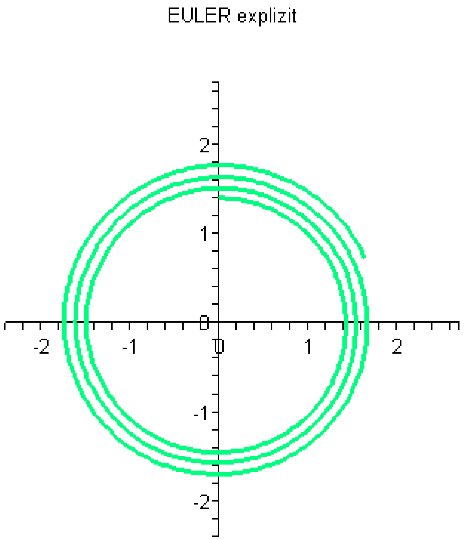




τ = 0.100

τ = 0.050

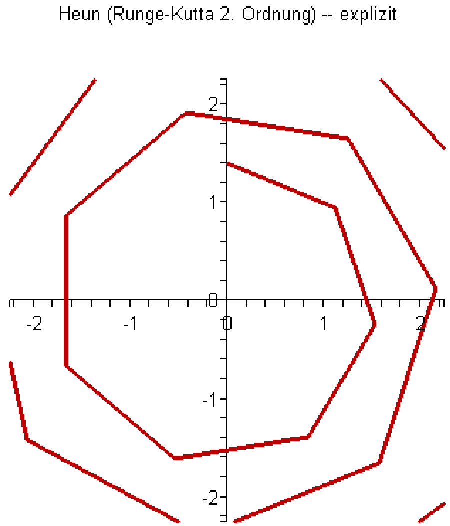


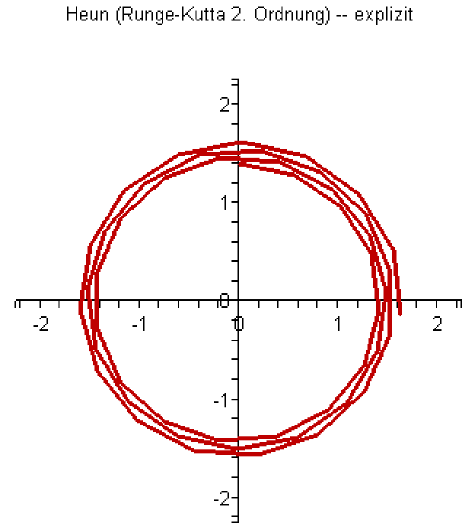


τ = 0.025

3.6 Beispiel: explizites Heun $O(\tau^2)$

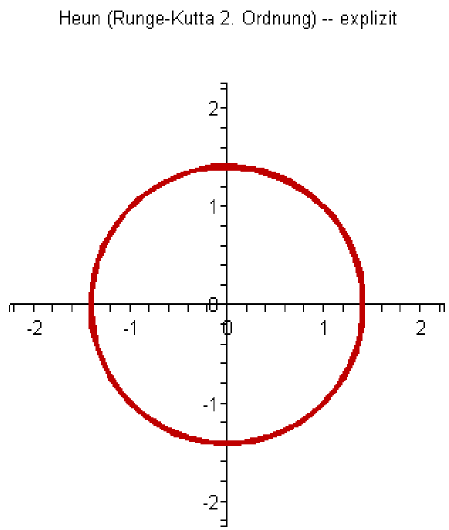
τ = 0.800

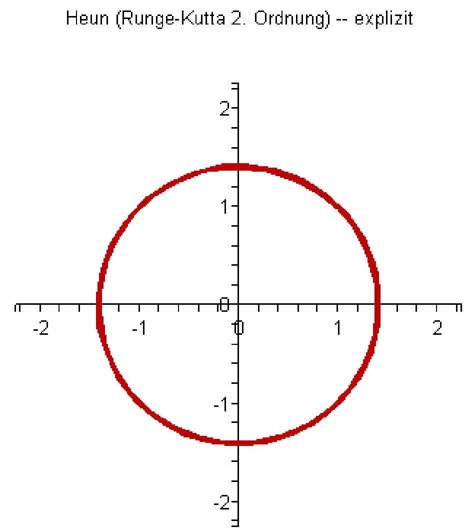




τ = 0.400

τ = 0.200

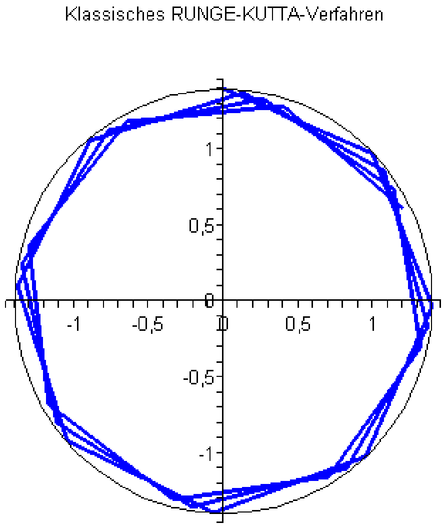


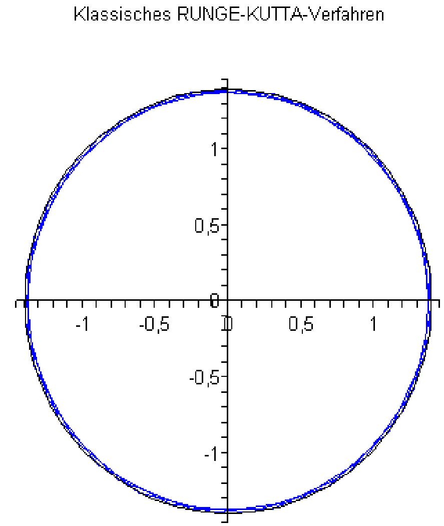


τ = 0.100

3.6 Beispiel: klassisches Runge-Kutta $O(\tau^4)$

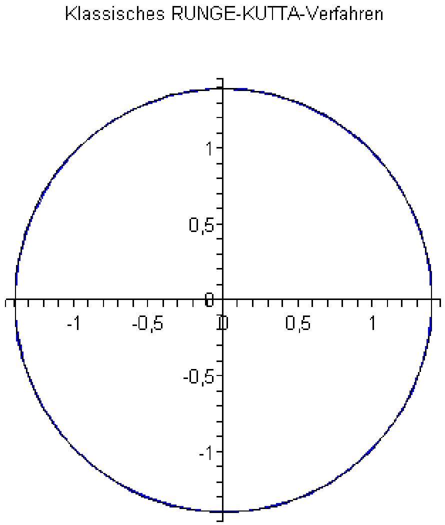
τ = 0.800

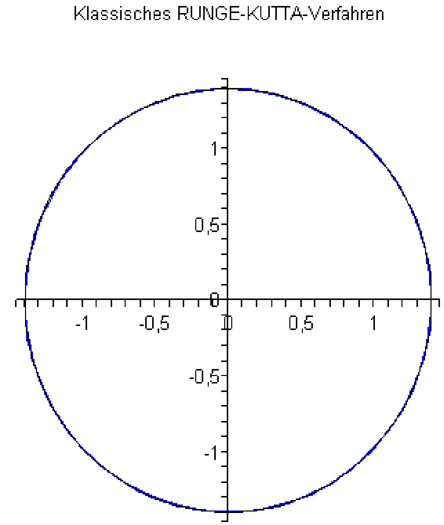




τ = 0.400

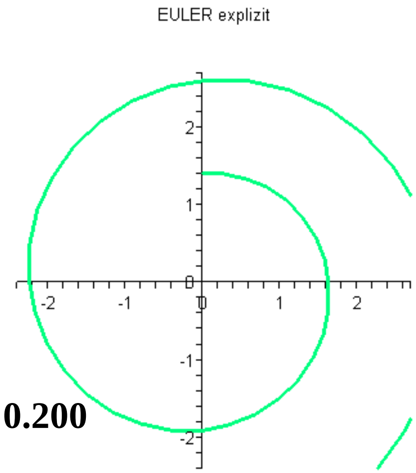
τ = 0.200



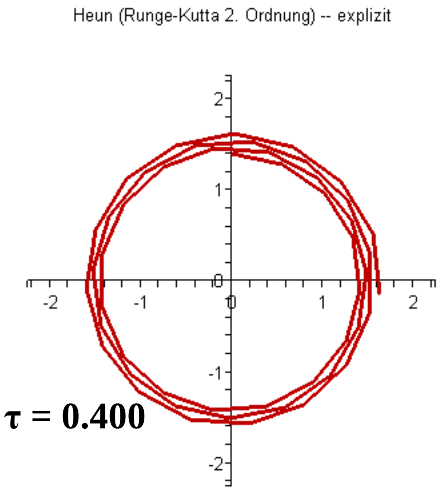


τ = 0.100

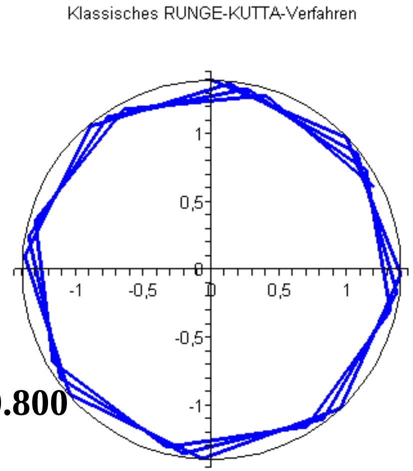
3.6 Beispiel: Vergleich ($O(\tau^1)$ $O(\tau^2)$ $O(\tau^4)$)



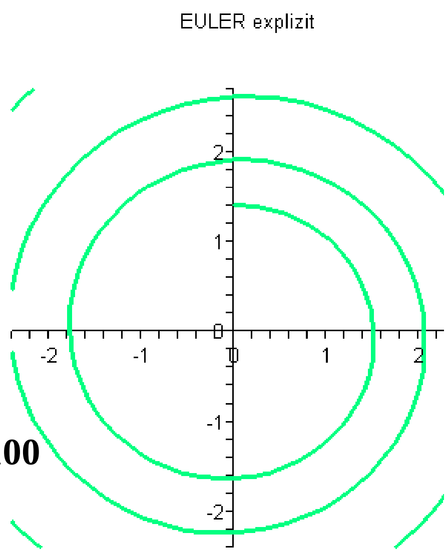
$\tau = 0.200$



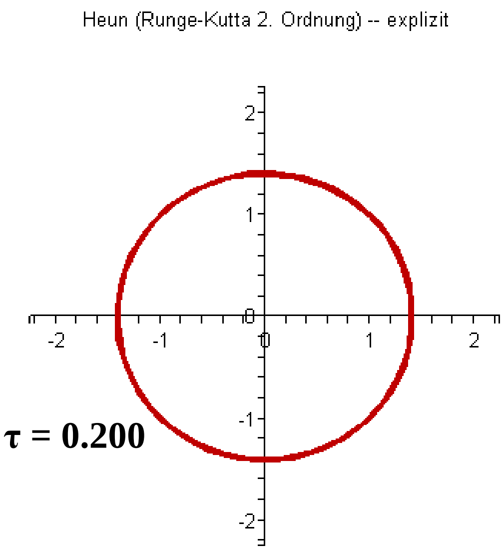
$\tau = 0.400$



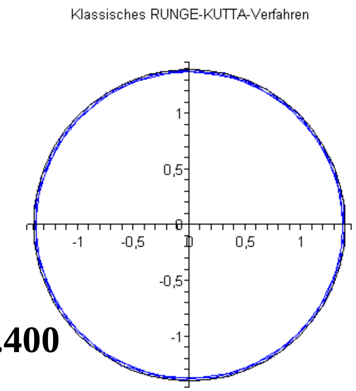
$\tau = 0.800$



$\tau = 0.100$



$\tau = 0.200$



$\tau = 0.400$

3.6 Integrationsverfahren - Mehrschrittverfahren

- explizite Mehrschritt-Verfahren: Adams-Bashforth

$$t_{i+1} = t_i + h; \quad y_{i+1} = y_i + h \cdot [\tfrac{3}{2} F(t_i, y_i) - \tfrac{1}{2} F(t_{i-1}, y_{i-1})]$$

$$t_{i+1} = t_i + h; \quad y_{i+1} = y_i + h \cdot [\tfrac{23}{12} F(t_i, y_i) - \tfrac{16}{12} F(t_{i-1}, y_{i-1}) + \tfrac{5}{12} F(t_{i-2}, y_{i-2})]$$

- Vorteil: pro Schritt nur eine Auswertung von $F(t,y)$!
- Nachteile
  - Anfangswerte
  - Schrittweiten-Steuerung ist schwierig
- implizite Mehrschritt-Verfahren: Adams-Moulton

$$t_{i+1} = t_i + h; \quad y_{i+1} = y_i + h \cdot [\tfrac{5}{12} F(t_{i+1}, y_{i+1}) + \tfrac{8}{12} F(t_i, y_i) - \tfrac{1}{12} F(t_{i-1}, y_{i-1})]$$

- Kombination: Prädiktor-Korrektor

3.6 Integrationsverfahren - Zusammenfassung

- Fehleranalyse:
  - Lokaler Fehler (Konsistenz)
  - Globaler Fehler (Konvergenz)
  - Falls Stabiltät:
    Konvergenzordnung = Konsistenzordnung (= lok. Fehlerordnung -1)
  - Gute Einschritt-Verfahren: Konv.-Ord. = #(Auswertung von F)
  - Gute Mehrschritt-Verfahren: Konv.-Ord = „Tiefe“
- Explizit vs. implizit
  - Implizite Verfahren sind „stabiler“ (unabhängig von Schrittweite)
  - Implizite erforden Lösen eines nichtlinearen Gleichungssystems
  - Gute implizite haben u.U. bessere Fehlerordnung
  - Kompromiss: Prädiktor-Korrektor-Verfahren
- Einschritt vs. Mehrschritt
  - Bei gleicher Ordnung pro nur eine Auswertung von F
  - Einschritt erlaubt Schrittweiten-Steuerung