1. Überblick

- Modelltausch: kontinuierliches Modell (gew. Dgln) ⇨ diskretes Modell

- Aufgabe: Wahl eines geeigneten Integrationsverfahrens und Anpassung der Schrittweite

  ↳ t_i ; t_n, t_{n+1}

  ↳ $h_n = t_{n+1} - t_n$

- Klassifizierung:

  $\dot{\vec{y}} = \vec{f}(\vec{y}, t)$ $\quad t \to t_0, \dots t_n, t_{n+1} \dots$

  $\vec{y}(t_0)$ geg.

  Nach der Integrationsart:
  - Explizite Verfahren $\quad \vec{y}(t_{n+1}) = F_e(\vec{y}(t_0), \vec{y}(t_1) \dots \vec{y}(t_n))$
  - Implizite Verfahren $\quad 0 = F_i(\vec{y}(t_0), \dots, \vec{y}(t_n), \vec{y}(t_{n+1}))$

    $\vec{y}(t_0) \to \vec{y}_0, \quad \vec{y}(t_n) \to \vec{y}_n, \quad \vec{y}(t_{n+1}) \to \vec{y}_{n+1}$

  Nach Anzahl der verwendeten Stützstellen:
  - Einschrittverfahren $\quad \vec{y}_{n+1} = F_e(\vec{y}_n)$, $0 = F_i(\vec{y}_{n+1}, \vec{y}_n)$
  - Mehrschrittverfahren

1. Überblick:

1. Integrationsformeln

$\dot{y} = f(t, y)$

$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} f(t, y(t))\, dt$

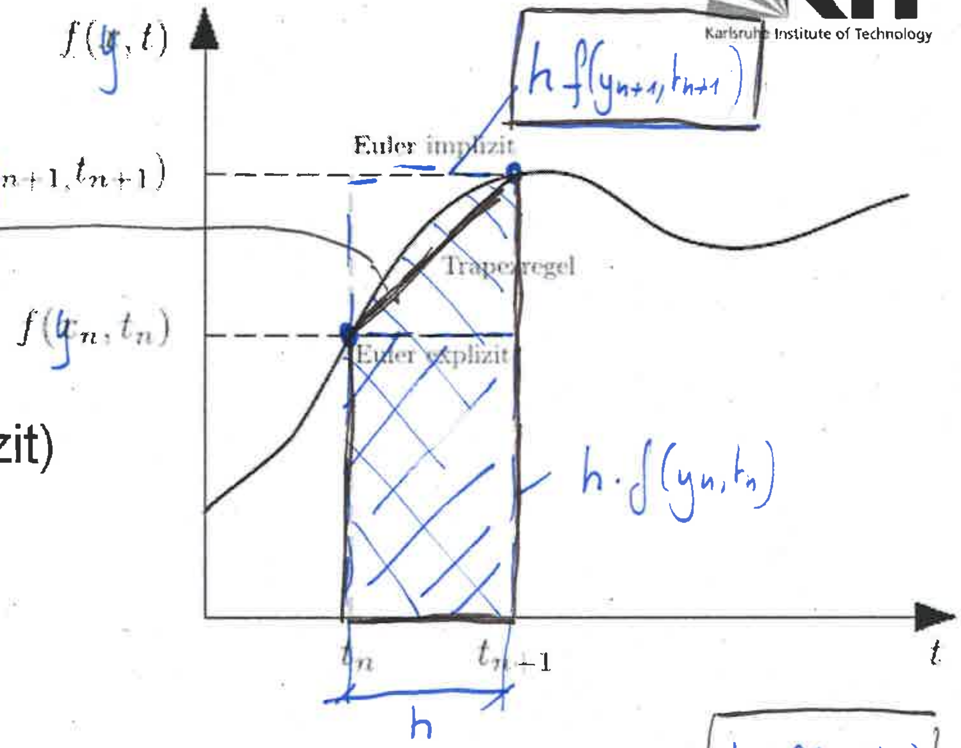
z.B. Rechteckregel

- $y_{n+1}=y_n + f(t_n,y_n)\ h$ (Euler explizit)
- $y_{n+1}=y_n + f(t_{n+1},y_{n+1})\ h$ (Euler implizit)

Trapezregel:

$y_{n+1}=y_n +(f(t_{n+1},y_{n+1})+f(t_n,y_n))\ h/2$

-> liefert offenbar ausschließlich Einschrittverfahren



$y_{n+1} = y_n + \boxed{h \cdot f(y_n,t_n)}$

1. Überblick

2. Näherung von f(t,y(t)) durch Interpolationspolynom p(t) und Lösen des Integrals $\int_{t_n}^{t_{n+1}} p(t)\,\mathrm{d}t$

$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} p(t)\,dt$, $p(t) \approx f(t,y)$

a) Lagrange-Interpolation $p(t) = \sum_{j=0}^{k-1} f(t_{n-j}, y_{n-j}) \frac{\prod_{i=0, i\neq j}^{k-1} (t - t_{n-i})}{\prod_{i=0, i\neq j}^{k-1} (t_{n-j} - t_{n-i})}$

Adams-Bashforth-Verfahren

offenbar explizit

k-Schrittverfahren

für k=1: Euler explizit

Nachteil: Interpolation erfolgt außerhalb des Integrationsintervalls

$k=1,\ j=0 \rightarrow f(t_n,y_n)\,1 = p(t) \rightsquigarrow \int_{t_n}^{t_{n+1}} p(t)\,dt = f(t_n,y_n)\underbrace{(t_{n+1}-t_n)}_{h}$

$y_{n+1} = y_n + f(t_n,y_n)\,h$

b) Lagrange-Interpolation $p(t) = \sum_{j=-1}^{k-1} f(t_{n-j}, y_{n-j}) \frac{\prod_{i=-1, i\neq j}^{k-1} (t - t_{n-i})}{\prod_{i=-1, i\neq j}^{k-1} (t_{n-j} - t_{n-i})}$

$k=0$, $j=-1$: $f(t_{n+1},y_{n+1})$

Adams-Bashforth-Moulton-Verfahren

offenbar implizit, k-Schrittverfahren

(außer k=0, Einschrittverfahren)

für k=0: Euler implizit

für k=1: Trapezregel

$k=0$: $f(t_{n+1},y_{n+1})\frac{1}{1} = p(t) \rightsquigarrow \int_{t_n}^{t_{n+1}} p(t) = f(t_{n+1},y_{n+1})(t_{n+1}-t_n)$

$k=1$: Interpolationsgerade

Institut für Technische Mechanik
Abteilung Dynamik / Mechatronik

1. Überblick

3. Numerische Differentiation

$\dot{y} = f(y,t)$

Ersetze $\dot{y}(t)$ durch $\sum_{i=0}^{k} \alpha_i y_{n+1-i}$

$\dot{y}_n$ oder $\dot{y}_{n+1} \approx \dot{y} \approx \frac{y_{n+1} - y_n}{t_{n+1} - t_n} = \frac{y_{n+1} - y_n}{h}$ $\qquad k = 1,\ \alpha_0 = \frac{1}{h},\ \alpha_1 = -\frac{1}{h}.$

$\frac{y_{n+1}}{h} = \frac{y_n}{h} + f(y,t) \mid h \quad \sim \quad y_{n+1} = y_n + f(y,t)\,h$ $\quad$ Euler

$\dot{y}_n = f(y_n, t_n) \quad \sim \quad y_{n+1} = y_n + f(y_n, t_n)\,h$ $\quad$ explizit

$\dot{y}_{n+1} = f(y_{n+1}, t_{n+1}) \quad y_{n+1} = y_n + f(y_{n+1}, t_{n+1})\,h$ $\quad$ implizit

2. Prädiktor-Korrektor-Verfahren

Nachteil impliziter Verfahren: in jedem Schritt ist ein nichtlineares Gleichungssystem zu lösen!

Ausweg: 2stufiges Verfahren

1. Näherung für y_{n+1} mit explizitem Verfahren
2. einsetzen in implizites Verfahren

Beispiel: Kombination aus explizitem Euler-Verfahren und Trapezregel (Verfahren von Heun)

$$y_{n+1}^{(1)}=y_n+h\ f(t_n,y_n)$$
$$\boxed{y_{n+1}}=y_n+h/2\ (f(t_{n+1},y_{n+1}^{(1)})+f(t_n,y_n))$$

Notation:

$$y_{n+1}=y_n+1/2\ (k_n^{(1)}+k_n^{(2)})$$
$$k_n^{(1)}=h\ \underline{f(t_n,y_n)}$$
$$k_n^{(2)}=h\ \underline{f(t_n+h,y_n+k_n^{(1)})}$$

Institut für Technische Mechanik
Abteilung Dynamik / Mechatronik

3. Fehleranalyse

KIT Karlsruhe Institute of Technology

Fehlerarten:

- **lokaler Diskretisierungsfehler l(h):**
  Differenz zwischen Differentialquotient und genähertem Differentialquotient für die Lösung des Anfangswertproblems, d.h., das, was in jedem Zeitschritt neu an Fehler entsteht.

$O(h^p)$

- Ist $l(h)/h^p$ beschränkt für $h \to 0$, so wird das Diskretisierungsschema **konsistent zur Ordnung p** genannt.

- **Konsistenzordnungen:**
  - Euler-Verfahren: p=1
  - Trapezregel: p=2
  - Verfahren von Heun: p=2
  - Runge-Kutta-Verfahren: p=4

Funktionsauswertung

1
2
2
4

4. Stabilität der Integrationsverfahren

Karlsruhe Institute of Technology

- Modellproblem zur Stabilitätsuntersuchung:

$$\boxed{\dot{y}(t)=\lambda y(t)},\ \lambda\in\mathbb{C},\ y(t_0=0)=y_0$$

Re(λ) > exp. aufklingt; = konstant; < abklingt.

- Exakte Lösung: $y(t)=\exp(\lambda t)\,y_0$

  stabil für $\mathrm{Re}(\lambda)\le 0$, sonst instabil.

abklingend

- Verfahren $\boxed{y_{n+1}=F(\lambda h)\,y_n}$ stabil, wenn $|F(\lambda h)|<1$

  aufklingend für $|F(\lambda h)|>1$

- Beispiele:

$f(y_n,t_n)=\lambda y_n$

explizites Euler-Verfahren:

$$y_{n+1}=y_n+h\lambda y_n$$

$$y_{n+1}=\underbrace{(1+h\lambda)}\,y_n \qquad \sim\ |1+h\lambda|<1\ ?$$

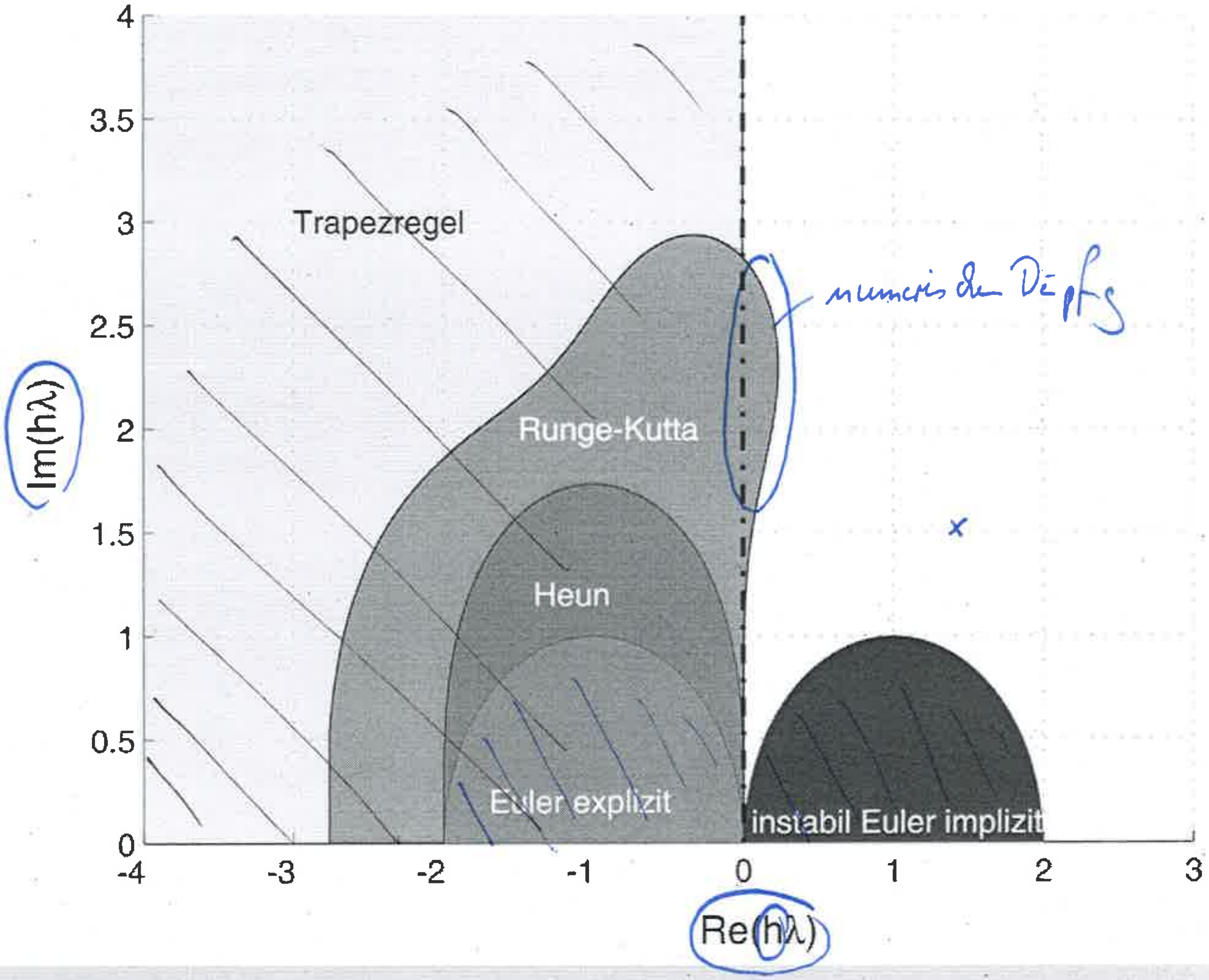
implizites Euler-Verfahren:

$$y_{n+1}=y_n+h\lambda y_{n+1} \qquad f(y_{n+1},t_{n+1})$$

$$\underline{y_{n+1}}=y_n/(1-h\lambda) \qquad \frac{1}{1-h\lambda} \qquad \sim\ \left|\frac{1}{1-h\lambda}\right|<1.$$

4. Stabilität der Integrationsverfahren

Stabilitätskarte der behandelten Verfahren



Institut für Technische Mechanik
Abteilung Dynamik / Mechatronik

5. Schrittweitensteuerung

- Schrittweitensteuerung
- Idee: lokalen Fehler messen und Schrittweite entsprechend anpassen
- Benötigt: Kontrollrechnung mit einem Verfahren der Ordnung p+1

$$\overset{p}{y_{n+1}} - \overset{p+1}{\tilde{y}_{n+1}}$$

$$|y_{n+1} - \tilde{y}_{n+1}| \leq Ch^p + \tilde{C}h^{p+1} \approx Ch^p$$

Kontrollrechnung sollte mit bereits vorgenommenen Funktionsauswertungen auskommen

(z.B. Runge-Kutta-Fehlberg-Verfahren)