

VL3

- Start recording
- Fragen?

1.3) Nullstellsuche

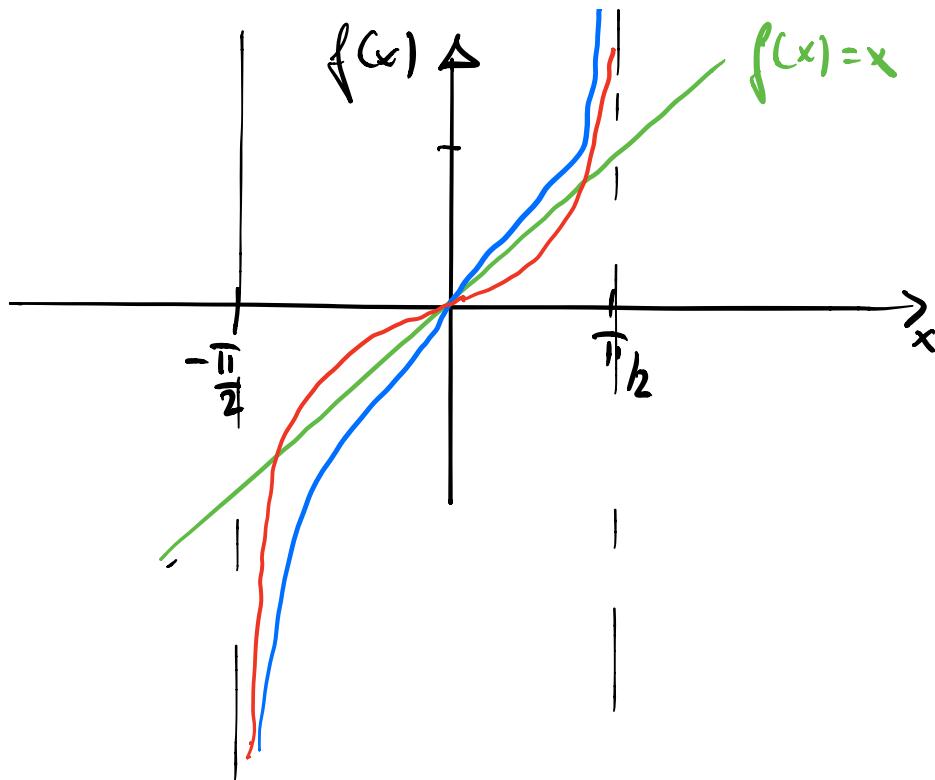
Problem: gegeben $f(x)$, gesucht Stellen x_0 mit

$$f(x_0) = 0$$

Wo kommt das vor in der Physik?

- Eigenwerte: charakteristisches Polynom
- Extremstellen
- gew. DGL (linear, konst. Koeffizienten)
- Nicht explizit auflösbare Gleichungen

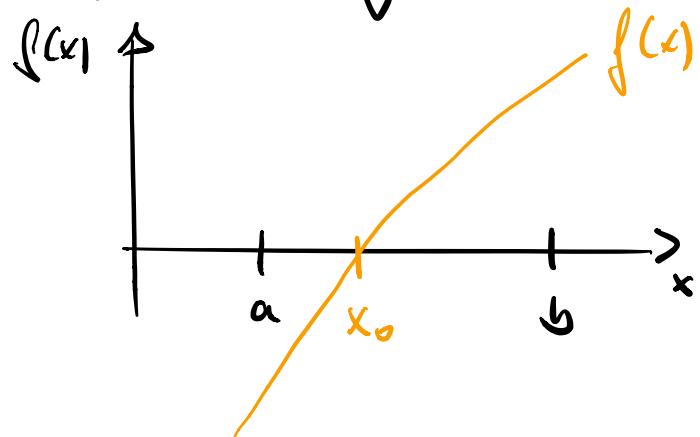
$$x = \tan(\alpha x)$$



1. Schritt: Graphisch Lösen

2. Schritt: numerisch Nullstellen x_0

Allg. Zugang



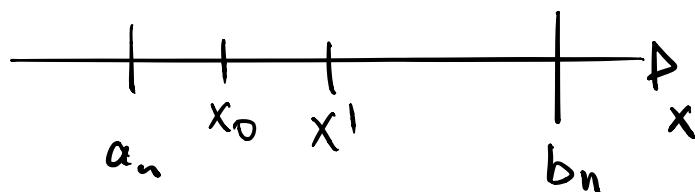
Für die Intervalle $[a, b]$ mit:

$f(a) \cdot f(b) < 0$ \Rightarrow \exists mindestens eine Nullstelle x_0
 $x_0 \in (a, b)$
 $(f$ stetig)

a) Bisektion (Intervallschachtung)

Gegeben $f(a_n) f(b_n) < 0$

$$\Rightarrow x^1 = \frac{a_n + b_n}{2} \quad \begin{array}{l} \text{[Intervall)} \\ \text{halbieren} \end{array}$$



Falls:

$$f(a) f(x^1) < 0$$

$$\Rightarrow b_{n+1} = x^1$$

sonst

$$a_{n+1} = x^1$$

Abbrechekriterium:

$$|f(a_{n+1})|, |f(b_{n+1})| < \varepsilon$$

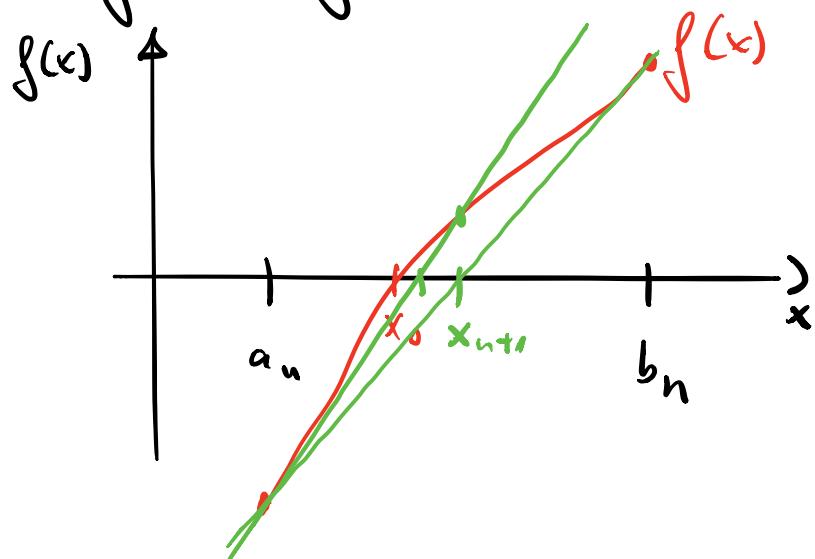
gewünschte
Genauigkeit

$$\rightarrow x_0 \approx a_{n+1}, b_{n+1}$$

- stabil
- nicht effizient

b) Regulär falsi

Gegeben: $f(a_n) f(b_n) < 0$



Lineare Interpolation:

$$f(x) \approx f(a_n) + \frac{f(b_n) - f(a_n)}{b_n - a_n} (x_{n+1} - a_n) \stackrel{!}{=} 0$$

$$\Rightarrow x_{n+1} = a_n + \frac{a_n - b_n}{f(b_n) - f(a_n)} \cdot f(a_n)$$

$$f(a) \cdot f(x_{n+1}) < 0$$

$$\Rightarrow b_{n+1} = x_{n+1}$$

somit

$$a_{n+1} = x_{n+1}$$

Abbruch vorher:

$$|f(x_{n+1})| < \varepsilon$$

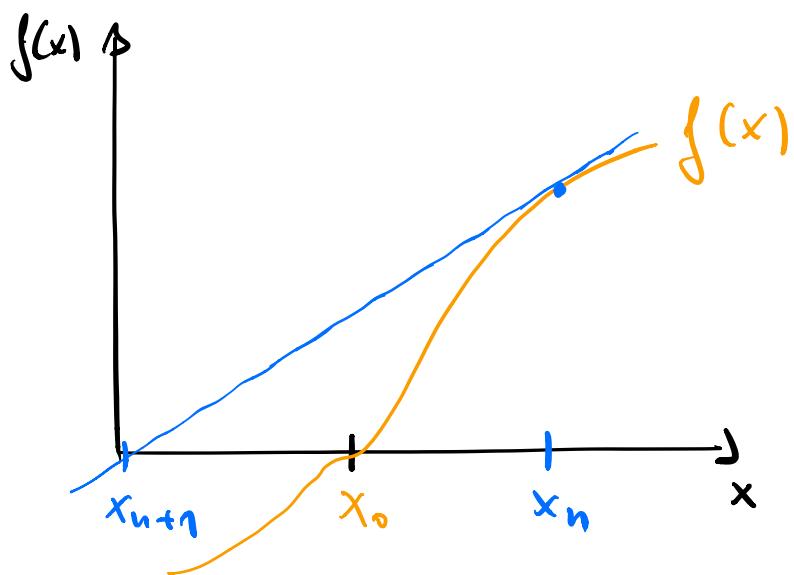
Bem.:

- schneller als Intervallschachtelung
- stabil
- Anpassen, falls $f(b_n) - f(a_n) < \varepsilon$!

c) Newton-Raphson Verfahren

x_n : n-te Abschätzung der Nullstelle

$$\Rightarrow x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$



Bem:

- Schnelle Konvergenz
- $\exists f'(x)$ Nullstellen in der Nähe von x_0
 $\rightarrow x_{n+1}$ geht
- Stark oszillierende Funktion

Diskrete Variante:

- Bsp: $f'(x)$ nicht bekannt
 $f(x_i)$ an diskreten Stellen x_i

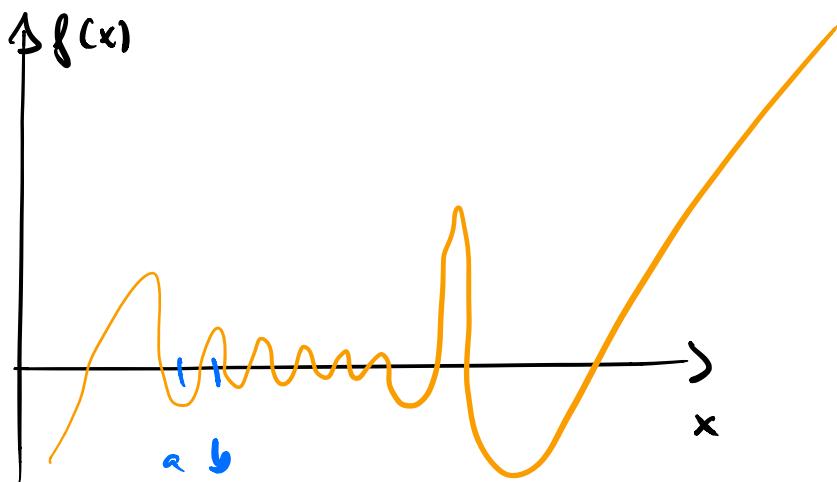
$$f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

(oder besser abschätzungen für f')

Empfehlung:

- Mehrere Verfahren implementieren

- Konsistente Ergebnisse



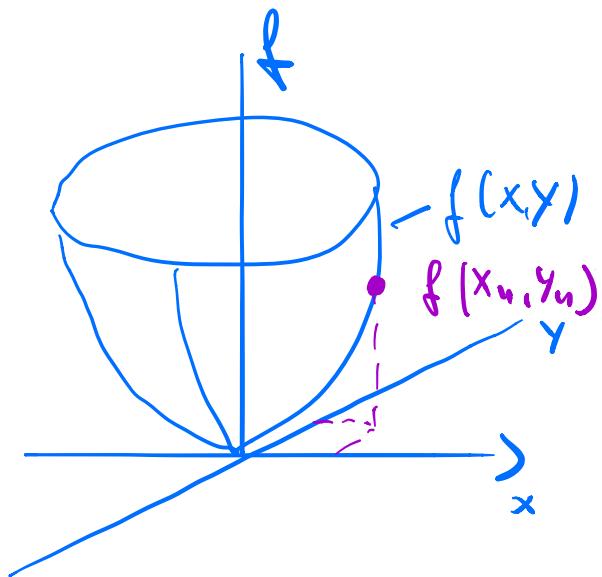
→ Wutige Abbrüche:

$\forall n \quad x_n \in [a, b]$ sonst Neustart mit
neuem Startwert.

Pause: 8:52 .

ausgelassen:

→ Suche nach Minima, wenn z.B.
skalar Fkt. $f(x, y, z)$ gegeben.



→ Berechne Gradienten, gehen in diesem
neg. Rtg. weiter (steepest descent)

- ~> lokale Minima problematisch.
- ~> Wie stellt man sicher, dass Minimum global ist?

3) gewöhnliche DGLn

Anfangswertprobleme

Auftritten:

* • Klassische Mechanik

• QM

Wellenmechanik: Eigenwertprobleme,
Randbedingungen

↳ später



• Partielle Gleichungen

(radioaktiver Zerfall, Biophysik, ...)

3.1) Darstellung als System von DGLn

1. Ordnung

Allg.: DGL n-ter Ordnung; $y = y(x)$

$$f(y^{(n)}(x), y^{(n-1)}(x), \dots, y(x), x) = 0$$

Explizit DGL: nach $y^{(n)}$ auflösbar

$$y^{(n)}(x) = g(y^{(n-1)}(x), \dots, y(x), x) \quad (*)$$

Hier: $x \in \mathbb{R}$

Überführen von (*) in System von DGLn

1. Ordnung:

Definieren Vektor von Funktionen:

$$\vec{y}(x) = \begin{pmatrix} y_1(x) \\ y_2(x) \\ \vdots \\ y_n(x) \end{pmatrix} := \begin{pmatrix} y(x) \\ y'(x) \\ \vdots \\ y^{(n-1)}(x) \end{pmatrix}$$

$$\Rightarrow \frac{d\vec{y}}{dx} = \begin{pmatrix} y'(x) \\ y''(x) \\ \vdots \\ y^{(n)}(x) \end{pmatrix} = \begin{pmatrix} y_2(x) \\ y_3(x) \\ \vdots \\ y_n(x) \\ y^{(n+1)}(x) \end{pmatrix}$$

$\vec{y} = \begin{pmatrix} y_1(x) \\ y_2(x) \\ y_3(x) \\ \vdots \\ y_n(x) \\ \underbrace{g(y_n(x), \dots, y_1(x), x)}_{g(\vec{y}, x)} \end{pmatrix} =: \tilde{G}(\vec{y}, x)$

 ↗ Def.

DGL
 n-fach
 Ordnung

$$\Rightarrow \frac{d\vec{y}}{dx} = \tilde{G}(\vec{y}, x)$$

Beispiel: gedämpfter harmonischer
 Oszillations, $x = x(t)$
 Koordinate \uparrow Zeit \uparrow

$$\ddot{x} + 2\gamma\dot{x} + \omega_0^2 x = 0 \quad ; \quad \gamma > 0$$

Homogene, lineare DGL 2. Ordnung,
konst. Koeffizienten

Def:

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} := \begin{pmatrix} x \\ \dot{x} \end{pmatrix}$$

$$\Rightarrow \frac{d}{dt} \vec{x} = \begin{pmatrix} \dot{x} \\ \ddot{x} \end{pmatrix} = \begin{pmatrix} x_2 \\ -\underline{2\gamma x_2 - \omega_0^2 x_1} \end{pmatrix}$$

$$= \begin{pmatrix} 0 & 1 \\ -\omega_0^2 & -2\gamma \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$\frac{d\vec{x}}{dt} = A \vec{x}$

→ Matrix A,
kone Fkt. von $\underline{\underline{\underline{t}}}$

Allg. lineare, homogene DGL (Mathe-Notation)

$$\frac{d\vec{y}}{dx} = A(x) \vec{y} \quad ; \quad \vec{y} = \vec{y}(x)$$

A

Matrix A hängt von x ab,

→ Matrix - Vektor Multiplikation

Wozu diese Umformung?

- numerische Verfahren basieren auf Näherungen für 1. Ableitung

(→ vermeidet numerische Berechnung

höherer Ableitungen, wenn nur

$\vec{y} = \vec{y}(x_i)$ nur an diskreten

Stellen x_i bekannt ist). numerisch!

3.2) Euler-Cauchy Verfahren

Hier: explizites Eulerverfahren

↪ DGL nach $\vec{y}^{(n)}$ auflösbar.

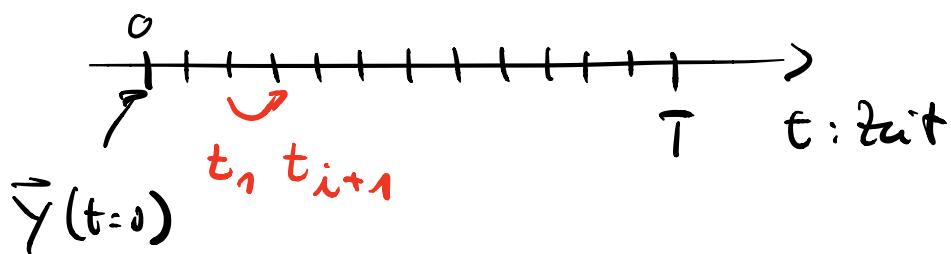
$$\frac{d\vec{y}}{dx} = \vec{g}(\vec{y}, x)$$

Näherung für 1. Ableitung:

Schrittweite: $\Delta x = x_{i+1} - x_i$

(Diskretisierung des Arguments)

Beisp.:



$$\frac{d\vec{y}}{dx} \underset{\approx}{=} \frac{\vec{y}(x_i + \Delta x) - \vec{y}(x_i)}{\Delta x} \stackrel{!}{=} \vec{g}(\vec{y}(x_i), x_i)$$

Vorwärtsintegration der DGL: iterativ

$$\vec{y}(x_0) \xrightarrow{\text{Euler}} \vec{y}(x_1) \rightarrow \dots \rightarrow \vec{y}(x_i) \rightarrow \vec{y}(x_{i+1})$$

$\rightarrow \dots$

$$\Rightarrow \boxed{\vec{y}(x_{i+1}) = \vec{y}(x_i + \Delta x)} \\ = \boxed{\vec{g}(\vec{y}(x_i), x_i) \Delta x + \vec{y}(x_i)}$$

Konkurrenz (siehe Kap. 1.2) $\sim O(\Delta x)$

pro Integrations -
schnitt

$$\vec{y}(x_i) = \begin{pmatrix} y(x_i) \\ \vdots \\ \vdots \\ y^{(n-1)}(x_i) \end{pmatrix}$$

Fr. Euler Verfahren funktioniert auch für
allg. \vec{y} ?

1. Beispiel: Radioaktiver Zerfall

$$\frac{dN}{dt} = -\gamma N \quad \begin{aligned} N &: \text{Zahl Isotope} \\ \gamma &: \text{Zerfalls-} \\ &\text{konstante} \end{aligned}$$

$$\Rightarrow \frac{N(t + \Delta t) - N(t)}{\Delta t} = -\gamma N(t)$$

$$\begin{aligned} \Rightarrow N(t + \Delta t) &= -\gamma N(t) \Delta t + N(t) \\ &= [-\gamma \Delta t + 1] N(t) \end{aligned}$$

$$\Rightarrow N(t_{i+1}) = \underbrace{[-\gamma \Delta t + 1]}_{= T(\Delta t) : \text{Skalar}} N(t_i)$$

2. Beispiel: Gedämpftes harmonisches Oszillieren (siehe oben)

$$\ddot{\vec{x}} = \vec{x}(t)$$

$$\frac{d}{dt} \ddot{\vec{x}} = A \ddot{\vec{x}} ; \quad A = \begin{pmatrix} 0 & 1 \\ -\omega_0^2 & -2\gamma \end{pmatrix}$$

$$\Rightarrow \frac{\ddot{\vec{x}}(t + \Delta t) - \ddot{\vec{x}}(t)}{\Delta t} \stackrel{!}{=} A \ddot{\vec{x}}(t)$$

$$\begin{aligned} \Rightarrow \boxed{\ddot{\vec{x}}(t_{i+1})} &= \Delta t A \ddot{\vec{x}}(t_i) + \ddot{\vec{x}}(t_i) \\ &= (\Delta t A + \underline{\underline{I}}) \ddot{\vec{x}}(t_i) \\ &= \boxed{\overline{T(\Delta t)} \ddot{\vec{x}}(t_i)} \end{aligned}$$

Matrix: $\overline{T}(\Delta t) = \underbrace{(\Delta t A + \underline{\underline{I}})}_{\uparrow}$

Einfachmatrix

Dgl 2. Ordnung: (Frage aus Publikum)

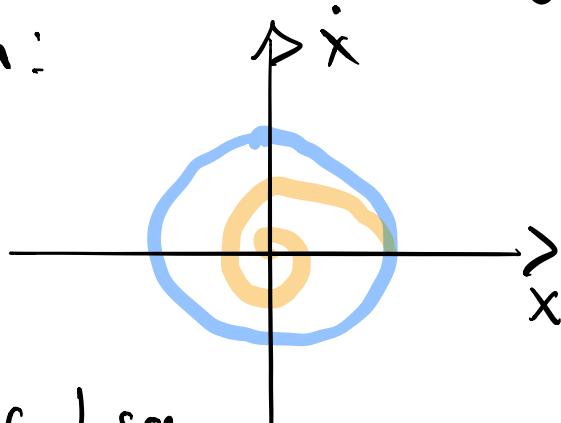
$$\begin{aligned} x(t=0) &= x_1(0) \\ \dot{x}(t=0) &= x_2(0) \end{aligned} \quad \left. \begin{array}{l} \text{Anfangs} \\ \text{bed.} \end{array} \right\}$$

$$\Rightarrow \vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \text{ bei } t=0 \text{ festgelegt.}$$

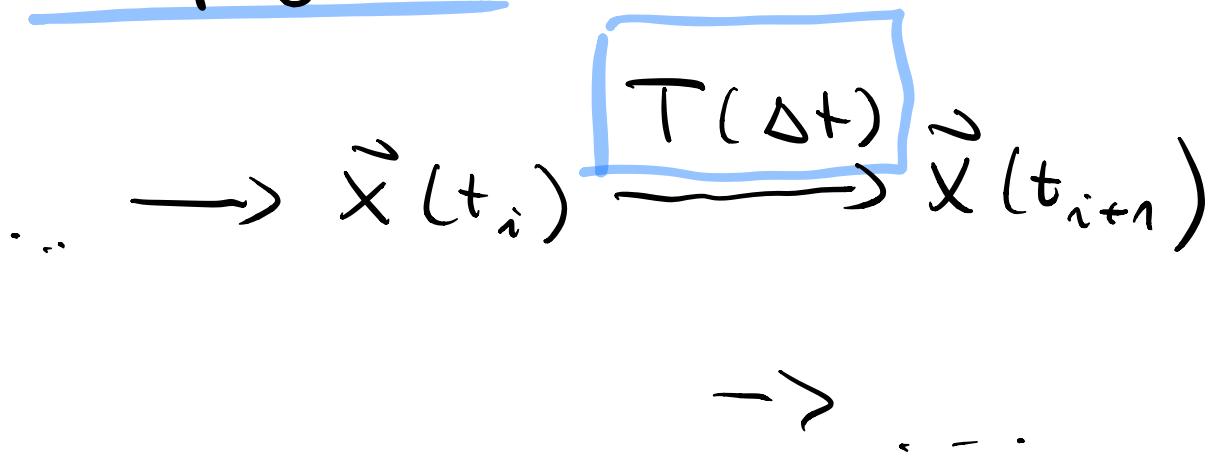
$$\vec{x}(0) \xrightarrow{T} \vec{x}(\Delta t) = T \vec{x}(0)$$

Wozu \dot{x}, x ? Beispiel: Darstellung im Phasenraum:

→ numerische Stabilitätsanalyse der phys. Lsg.



Propagator:



T : hängt ab von

- DGL (via A, Γ)
- Zeitschritt Δt (allg. Δx)

- Integrationsverfahren:

Euler: $(\Delta t A + I)$

Ziel: VL4

Was bedingen Eigenschaften von T:

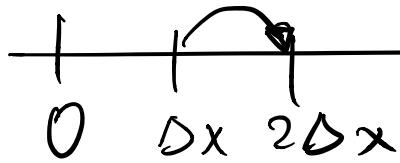
- Konvergenz
- Fehl.
- Stabilität

Discussion: Euler

- einfach zu programmieren
- schnell (wenige Operationen pro Schritt)
- niedrige Genauigkeit:
 - Fehler / Schritt Δx
 - $\sim \Theta(\Delta x^2)$

Globaler Fehler nach N Schritten

$\tilde{y}(w)$ Fehler/ Δx



Globaler Fehler



x_{\max}

$$x_{\max} = N \cdot \Delta x$$

$$\Rightarrow \frac{1}{N} \propto \Delta x$$

$$\boxed{\Delta} = N \mathcal{O}(\Delta^2 x)$$

$$= N \mathcal{O}\left(\left(\frac{x_{\max}}{N}\right)^2\right)$$

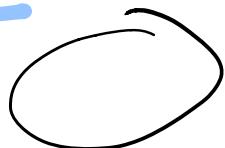
$$\sim \mathcal{O}\left(\frac{1}{N}\right) \sim \mathcal{O}(\Delta x)$$

- Stabilitätsprobleme (VL4 + UE)

• Nutzen: einfache
Test komplizierterer
Verfahren

102375 81, ...

102376 30, ...



$\approx O(1)$

$$\frac{1}{N} \sum_i x_i \rightarrow \sum_i \frac{x_i}{N}$$