Numerisches Programmieren (IN0019)

Frank R. Schmidt

Winter Semester 2016/2017

11. Gewöhnliche Differenzialgleichungen

Differenzialgleichungen

Differenzialgleichungen (Physik)

Physikalische Prozesse lassen sich mit **Differenzialgleichungen** beschreiben.

Das **2. Newtonsche Gesetz** $F = m_0 \cdot a$ geht davon aus, dass auf einen Körper der Ruhemasse m_0 eine konstante Kraft F ausgeübt wird. Da die Beschleunigung $a(t)$ die Zeit-Ableitung der Geschwindigkeit $v(t)$ ist, erhält man folgende Beschreibung

$$v'(t) = \frac{F}{m_0} \qquad v(0) = 0$$

Modelliert man außerdem noch, dass sich die Masse mit der Geschwindigkeit verändert (Spezielle Relativität), so erhält man

$$v'(t) = \frac{F}{m_0 \cdot c}\sqrt{c^2 - v(t)^2} \qquad v(0) = 0$$

Man erhält also eine **Gleichung**, die von einer Funktion und ihren **Ableitungen (Differenzialen)** abhängt.

Differenzialgleichungen (Biologie)

Auch Wachstumsprozesse lassen sich mit **Differenzialgleichungen** darstellen.

Setzt man eine Bakterienkultur auf einem Nährboden aus, so kann man mit $p(t)$ die Population zum Zeitpunkt t angeben. Da die Reproduktionsrate $p'(t)$ von der aktuellen Population abhängt, erhält man

$$p'(t) = \alpha \cdot p(t) \qquad p(0) = 1$$

Modelliert man außerdem, dass die Petrischale nur Platz für eine Population $p \leqslant K$ zulässt, kann man das wie folgt modellieren

$$p'(t) = \alpha \cdot p(t) \cdot \left(1 - \frac{p(t)}{K}\right) \qquad p(0) = 1$$

Es ist oft schwer, diese Probleme analytisch zu lösen.

Anfangswertproblem

Wir suchen eine Funktion $y\colon \mathbb{R} \to \mathbb{R}$, die folgende Differenzialgleichung erfüllt:

$$y'(x) = F(x, y(x)) \qquad y(x_0) = y_0,$$

wobei $F\colon \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ stetig ist.

Wir nennen dieses Problem ein **Anfangswertproblem (AWP)**, da neben der Differenzialgleichung auch der **Anfangswert** y_0 an der Stelle x_0 gegeben ist.

Man kann zeigen, dass das AWP eindeutig lösbar ist, wenn $F(x, y)$ in y differenzierbar ist und diese Ableitungen durch eine Zahl L beschränkt sind.

Ein AWP kann auch für Funktionen $y\colon \mathbb{R} \to \mathbb{R}^n$ beschrieben werden:

$$y'(x) = F(x, y(x)) \qquad y(x_0) = y_0 \in \mathbb{R}^n,$$

wobei $F\colon \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ stetig ist.

AWP m-ter Ordnung

Weiter kann man Differenzialgleichungen auch für höhere Ableitungen beschreiben. Wir erhalten dann das AWP m**-ter Ordnung**

$$y^{(m)}(x) = F(x, y^{(0)}(x), \dots, y^{(m-1)}(x)) \qquad y^{(k)}(x_0) = y_{0,k} \qquad \forall k < m$$

mit $F\colon \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$.

Ein AWP m-ter Ordnung kann in ein vektorwertiges AWP **erster Ordnung transformiert** werden:

$$\begin{aligned} z_k' &= z_{k+1} && k < m-1 \\ z_{m-1}' &= F(x, z_0(x), \dots, z_{m-1}(x)) \\ z(x_0) &= (y_{0,0}, \dots, y_{0,m-1})^\top \end{aligned}$$

Es reicht also, Lösungen für das AWP erster Ordnung zu finden.

Spezialfall: Integration

Hängt F nicht von $y(x)$ ab, so erhalten wir das AWP

$$y'(x) = F(x) \qquad y(x_0) = y_0,$$

Dieses Problem lässt sich mit Hilfe der **Integration** eindeutig lösen

$$f(x) = y_0 + \int_{x_0}^{x} F(t)\,\mathrm{dt}$$

Hierzu gehört z.B. das Bewegungsgesetz von Newton und wir erhalten für

$$v'(x) = \frac{F}{m_0} \qquad v(0) = 0$$

gerade die **eindeutige Lösung**

$$v(x) = 0 + \int_0^x \frac{F}{m_0}\,\mathrm{dt} = \frac{F}{m_0} \cdot x$$

Picard-Lindelöf

Iteratives Verfahren

Wir wollen nun mit der Integration eine **Fixpunkt-Gleichung** formulieren.

Ist z.B. y die Lösung, die wir suchen, so gilt für dieses y gerade

$$y(x) = y_0 + \int_{x_0}^{x} y'(t)\,\mathrm{dt} = y_0 + \int_{x_0}^{x} F(t, y(t))\,\mathrm{dt} =: \Phi(y)$$

Das **Theorem von Picard-Lindelöf** sagt aus, dass die **Iterationsvorschrift** Φ die Voraussetzung des **Banachschen Fixpunktsatzes** erfüllt.

Damit wissen wir zum Einen, dass die Lösung y^* des AWP eindeutig ist.

Außerdem erhalten wir durch wiederholtes Anwenden von Φ eine Funktionsfolge $y^{[k]}$ die gegen die eindeutige Lösung y^* des AWP konvergiert.

Für den Spezialfall des Integrationsproblem haben wir bereits gesehen, dass eine einmalige Anwendung von Φ zur Lösung y^* führt.

Iterative Funktionenfolge (Bsp.)

Betrachten wir nun das AWP

$$y'(x) = y(x) \qquad y(0) = 1.$$

Wir erhalten die folgende Iterationsfolge $y^{[k+1]}(x) = 1 + \int_0^x y^{[k]}(t)\mathrm{dt}$.

Starten wir mit der Einsfunktion, $y^{[0]} := 1$, so erhalten wir

$$\begin{aligned} y^{[1]}(x) &= 1 + \int_0^x 1\,\mathrm{dt} = 1 + x \\ y^{[2]}(x) &= 1 + \int_0^x 1 + t\,\mathrm{dt} = 1 + x + \frac{1}{2}x^2 \\ y^{[k+1]}(x) &= 1 + \int_0^x \sum_{n=0}^{k} \frac{t^n}{n!}\,\mathrm{dt} = \sum_{n=0}^{k+1} \frac{x^n}{n!}, \end{aligned}$$

d.h. die Funktionsfolge konvergiert zur eindeutigen Lösung $y^*(x) = \exp(x)$.

Zusammenfassung

Viele naturwissenschaftliche Probleme lasses sich als Lösungen von Differenzialgleichungen $y'(x) = F(x, y(x))$ darstellen.

Das AWP mit $y(x_0) = y_0$ lässt sich immer eindeutig lösen, wenn F in y differenzierbar ist und diese Ableitung beschränkt ist.

Die eindeutige Lösung lässt sich mit Hilfe eines iterativen Verfahrens lösen, das in jedem Schritt ein Integral berechnet.

Ist F von der Gestalt $F(x)$, lässt sich das Problem auf ein Integrationsproblem zurückführen, das von dem iterativen Verfahren in einem Schritt gelöst wird.

Anstelle der Iterationsvorschrift werden wir im Folgenden die Fixpunktgleichung $y = \Phi(y)$ direkt benutzen.

Die Integralberechung wird mit Hilfe von **Quadratur-Formeln** durchgeführt. Unterschiedliche Quadratur-Formeln führen zu unterschiedlichen Verfahren.

Einschritt-Verfahren

Einschritt-Verfahren

Wir sind an dem Verlauf der Funktion $y\colon [x_0; x_0 + T] \to \mathbb{R}$ interessiert. Dazu zerlegen wir das Intervall $[x_0; x_0 + T]$ in N äquidistante Abschnitte

$$x_k = x_0 + h \cdot k \qquad h := \frac{T}{N} \qquad k = 0, \ldots, N$$

Jedes Verfahren brechnet $y_k \approx y(x_k)$ nachdem alle y_j mit $j < k$ bereits berechnet wurden.

Wir starten mit der exakten Lösung des Anfangswertes $y_0 = y(x_0)$ und berechnen zuerst y_1, y_2, etc. bis $y_N \approx y(x_0 + T)$ berechnet ist.

Hängt y_{k+1} nur von y_k ab, sprechen wir von einem **Einschritt-Verfahren**. Andernfalls von einem **Mehrschritt-Verfahren**.

Alle Verfahren basieren üblicherweise auf Quadratur-Formeln.

Euler-Verfahren

Die einfachste Quadratur-Formel geht davon aus, dass der Integrand konstant ist. Wir erhalten damit das **explizite Euler-Verfahren**

$$\begin{aligned} y(x_0 + h) &= y_0 + \int_0^h F(x_0 + t, y(x_0 + t))\mathrm{dt} \approx y_0 + \int_0^h F(x_0, y_0)\mathrm{dt} \\ &= y_0 + h \cdot F(x_0, y_0) =: y_1 \end{aligned}$$

Approximieren wir den Integranden mit dem konstanten Wert $F(x_0, y(x_0 + h))$, erhalten wir das **implizite Euler-Verfahren**

$$y(x_0 + h) = y_0 + h \cdot F(x_0, y(x_0 + h))$$

Während im expliziten Verfahren, der neue Wert y_1 explizit gegeben ist, muss beim impliziten Verfahren die folgende Gleichung gelöst werden:

$$y_1 = y_0 + h \cdot F(x_0, y_1)$$

Implizite Verfahren

Um y_1 zu erhalten, ist die Nullstelle der Funktion

$$f(y_1) = (y_0 - y_1) + h \cdot F(x_0, y_1)$$

zu finden.

Hier kann man z.B. das Newton-Verfahren benutzen.

Da implizite Verfahren oft exaktere Werte als explizite Verfahren liefern, ist es sinnvoll, das Newton-Verfahren mit dem Wert zu initialisieren, der das Ergebnis des expliziten Verfahrens ist.

Man spricht dann von dem expliziten Euler-Schritt als **Predictor** und von dem Newton-Verfahren als **Corrector**.

Nach der Berechnung von y_1 kann man $y_2 \approx y(x_1 + h)$ berechnen, etc.

Verbessertes Euler-Verfahren

Die einfachste Gauß-Quadratur-Formel ist die Mittelpunktsregel. Mit ihr erhält man das **verbesserte Euler-Verfahren**

$$\begin{aligned} y(x_0+h) =& y_0 + \int_0^h F(x_0+t, y(x_0+t))\mathrm{dt} \approx y_0 + \int_0^h F\left(x_0+\frac{h}{2}, y_{0.5}\right)\mathrm{dt} \\ =& y_0 + h\cdot F\left(x_0+\frac{h}{2}, y_{0.5}\right) \end{aligned}$$

Da $y_{0.5}$ nicht bekannt ist, wird hierfür das explizite Euler-Verfahren benutzt, d.h. wir erhalten insgesamt folgende Vorschrift:

$$\begin{aligned} y_{0.5} =& y_0 + \frac{h}{2}\cdot F(x_0, y_0) \\ y_1 =& y_0 + h\cdot F\left(x_0+\frac{h}{2}, y_{0.5}\right) \end{aligned}$$

Da die benutzte Quadratur eine Gauß-Quadratur ist, erwarten wir eine bessere Konvergenz als bei dem expliziten Euler-Verfahren mit halber Schrittweite.

Euler-Verfahren (Exp. Wachstum)

Betrachten wir die drei Euler-Verfahren für das folgende AWP

$$y'(x) = y(x) \qquad y_0 = 1$$

Das **explizite Euler-Verfahren** berechnet

$$y_1 = y_0 + h\cdot y_0 = y_0\cdot(1+h)$$

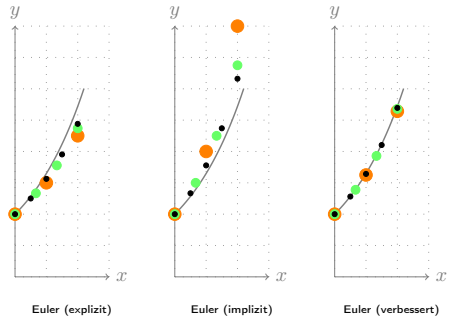
Das **implizite Euler-Verfahren** berechnet

$$y_1 = y_0 + h\cdot y_1 = y_0\cdot\frac{1}{1-h}$$

Das **verbesserte Euler-Verfahren** berechnet

$$y_{0.5} = y_0\cdot\left(1+\frac{h}{2}\right) \qquad y_1 = y_0\cdot\left[1+h\cdot\left(1+\frac{h}{2}\right)\right]$$

Euler-Verfahren (Bsp.)



Lösungen der drei Euler-Verfahren für $x_0 = 0$, $T = 1$ und $N = 2, 3, 4$.

Klassisches Runge-Kutta-Verfahren

Das **klassische Runge-Kutta-Verfahren** benutzt die Simpson-Regel als Quadratur-Formel:

$$y_1 = y_0 + \int_0^h F(x_0+t, y(x_0+t))\mathrm{dt} \approx y_0 + \frac{h}{6}\left(k_1 + 4\frac{k_2+k_3}{2} + k_4\right),$$

wobei k_1,k_4 den Integrand an den Rändern approximiert und k_2 sowie k_3 den zentralen Punkt des Integranden approximiert.

Runge schlug 1895 folgende Werte für k_1 bis k_4 vor:

$k_1 = F(x_0, y_0)$	(Exakte Berechnung)
$k_2 = F(x_0+0.5h, y_0+0.5h\cdot k_1)$	("Expliziter" Euler-Schritt)
$k_3 = F(x_0+0.5h, y_0+0.5h\cdot k_2)$	("Impliziter" Euler-Schritt)
$k_4 = F(x_0+h, y_0+h\cdot k_3)$	("Verbesserter" Euler-Schritt)

Runge-Kutta (Exp. Wachstum)

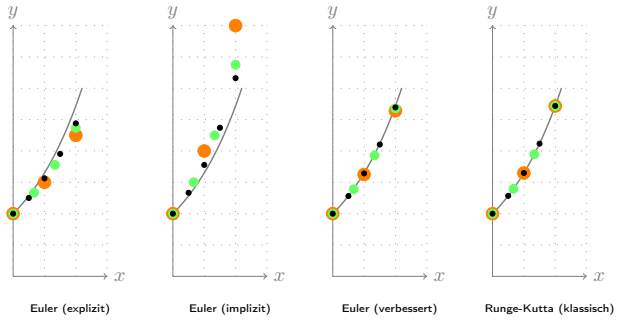
Betrachten wir erneut das AWP

$$y'(x) = y(x) \qquad y_0 = 1$$

Dann gilt für k_1 bis k_4 sowie y_1 gerade

$$\begin{aligned} k_1 =& F(x_0, y_0) = y_0 \\ k_2 =& y_0 + 0.5h\cdot k_1 = y_0\cdot\left(1+\frac{1}{2}h\right) \\ k_3 =& y_0 + 0.5h\cdot k_2 = y_0\cdot\left(1+\frac{1}{2}h+\frac{1}{4}h^2\right) \\ k_4 =& y_0 + h\cdot k_3 = y_0\cdot\left(1+h+\frac{1}{2}h^2+\frac{1}{4}h^3\right) \\ y_1 =& y_0 + \frac{h}{6}(k_1+2k_2+2k_3+k_4) = y_0\cdot\left(1+h+\frac{1}{2}h^2+\frac{1}{6}h^3+\frac{1}{24}h^4\right) \end{aligned}$$

Einschritt-Verfahren (Exp.)



Lösungen von Einschritt-Verfahren für $x_0 = 0$, $T = 1$ und $N = 2, 3, 4$.

Fehleranalyse

Diskretisierungsfehler

Bei der Definition des lokalen Diskretisierungsfehlers ε_k beschreiben wir nur den Fehler, der entsteht, wenn wir den Übergang von $(x_k, y(x_k))$ nach $(x_{k+1}, \hat{y}_{k+1})$ berechnen.

$y(x_k)$ ist also der exakte Wert von y und nicht die Approximation y_k.

Der **lokale Diskretisierungsfehler** ist somit:

$$e_k := \hat{y}_{k+1} - y(x_{k+1}).$$

Als **globalen Diskretisierungsfehler** bezeichnen wir:

$$\varepsilon_k := y_k - y(x_k).$$

Dieser Fehler berücksichtigt alle Fehler bei der Berechnung von y_k.

Konsistenz

Bei Einschritt-Verfahren wollen wir also erreichen, dass in jedem Schritt ein möglichst kleiner Fehler entsteht.

Wir sagen, dass ein Verfahren die **Konsistenzordnung** $p \in \mathbb{N}$ besitzt, falls

$$y_1 - y_0 = \mathcal{O}(h^{1+p}).$$

Für den globalen Diskretisierungsfehler gilt nun bzgl. einer Konstanten $\overline{L} > L$:

$$\begin{aligned}|\varepsilon_N| &= |y(x_N) - \hat{y}_N| + |\hat{y}_N - y_N| \leqslant |e_{N-1}| + (1 + \overline{L}h)\,|\varepsilon_{N-1}| \\ &\leqslant \ldots \leqslant e^{\overline{L}T} \sum_{k=0}^{N-1} |e_k| = \mathcal{O}(h^p)\end{aligned}$$

Wir nennen ein Verfahren **konsistent**, wenn $p \geqslant 1$ gilt.

Dann verbessert sich der globale Diskretisierungsfehler bei jeder Verfeinerung.

Konsistenzordnung (Bsp.)

Die Konsistenzordnung des expliziten Euler-Verfahrens ist $p = 1$.

Die Konsistenzordnung des impliziten Euler-Verfahrens ist $p = 1$.

Die Konsistenzordnung des verbesserten Euler-Verfahrens ist $p = 2$.
Hierdurch ist der Name des "verbesserten" Euler-Verfahrens gerechtfertigt.

Die Konsistenzordnung des klassischen Runge-Kutta-Verfahrens ist $p = 4$.

Wir haben das bereits am Beispiel des exponentiellen Wachstums gesehen:

$$\begin{aligned}\text{(expl. Euler:)} \quad y_1 &= y_0 \cdot \sum_{k=0}^{1} \frac{h^k}{k!} & \text{(impl. Euler:)} \quad y_1 &= y_0 \cdot \sum_{k=0}^{1} \frac{h^k}{k!} + \mathcal{O}(h^2) \\ \text{(verb. Euler:)} \quad y_1 &= y_0 \cdot \sum_{k=0}^{2} \frac{h^k}{k!} & \text{(Runge-Kutta:)} \quad y_1 &= y_0 \cdot \sum_{k=0}^{4} \frac{h^k}{k!}\end{aligned}$$

Konvergenz

Ein konsistentes Verfahren stellt sicher, dass wir für $h \to 0$ die korrekte Lösung berechnen, d.h. wir betrachten folgende Folgen

$$y_0 \qquad y_1^{(h)} \qquad \ldots \qquad y_N^{(h)} \qquad \text{für } h = \frac{T}{N}$$

Dann ist $y_N^{(h)}$ eine Approximation von $y(x_0 + T)$ und es gilt

$$\lim_{N \to \infty} y_N^{(\frac{T}{N})} = y(x_0 + T)$$

Dieses Verhalten nennen wir **Konvergenz** des Verfahrens.

Man kann zeigen, dass ein Verfahren mit der **Konsistenzordnung** p auch mit der **Ordnung** p **konvergiert**, d.h. eine höhere Konsistenzordnung ist grundsätzlich besser, um eine Differenzialgleichung approximativ zu lösen.

Stabilität

Die Konsistenz beantwortet die Frage nach der Konvergenz für $h \to 0$. Nun beschäftigen wir uns mit der Frage, welche h "sinnvolle" Ergebnisse liefern.

Betrachten wir hierzu erneut das AWP

$$y'(x) = \lambda \cdot y(x) \qquad y(0) = 1$$

Wir sagen, dass ein Verfahren bzgl. $h > 0$ stabil ist, wenn die Folge $y_1, \ldots, y_n, \ldots$ folgende Eigenschaften hat

$$\begin{aligned}\lim_{n \to \infty} y_n &= \infty & (\lambda > 0) \\ \lim_{n \to \infty} y_n &= 0 & (\lambda < 0)\end{aligned}$$

Wir definieren die Stabilität nur bzgl. des exponentiellen Wachstums, da man hierfür die exakte Lösung der Differenzialgleichung kennt.

Expl. Euler (Stabilität)

Das explizite Euler-Verfahren berechnet die folgende Folge

$$\begin{aligned}y_{k+1} &= y_k + h\lambda y_k = y_k \cdot (1 + h\lambda) \\ y_k &= (1 + h\lambda)^k\end{aligned}$$

Für $\lambda > 0$ erhalten wir die Stabilitätsbedingung

$$|1 + h\lambda| > 1,$$

die wegen $h > 0$ immer erfüllt ist.

Für $\lambda < 0$ erhalten wir die Stabilitätsbedingung

$$|1 + h\lambda| < 1,$$

die nur für $h < \frac{2}{|\lambda|}$ erfüllt ist.

Impl. Euler (Stabilität)

Das implizite Euler-Verfahren berechnet die folgende Folge

$$\begin{aligned}y_{k+1} &= \frac{y_k}{1 - h\lambda} \\ y_k &= \frac{1}{(1 - h\lambda)^k}\end{aligned}$$

Für $\lambda > 0$ erhalten wir die Stabilitätsbedingung

$$|1 - h\lambda| < 1,$$

die nur für $h < \frac{2}{\lambda}$ erfüllt ist.

Für $\lambda < 0$ erhalten wir die Stabilitätsbedingung

$$|1 - h\lambda| > 1,$$

die wegen $h > 0$ immer erfüllt ist.

Stabilität in der Praxis

Das explizite Euler-Verfahren ist stabil für alle Schrittweiten, falls $\lambda > 0$ gilt.

Das implizite Euler-Verfahren ist stabil für alle Schrittweiten, falls $\lambda < 0$ gilt.

In der Praxis ist der Fall $\lambda < 0$ wichtiger als $\lambda > 0$, da $\lambda < 0$ eine Dämpfung beschreibt. Wenn man z.B. bei statischen Problemen, Spannungen korrekt beschreiben möchte, sollte auf jeden Fall die Situation $\lambda < 0$ korrekt dargestellt werden.

Aus diesen Gründen wird das implizite Euler-Verfahren oft dem expliziten Euler-Verfahren vorgezogen.

Logistisches Wachstum

Im Folgenden betrachten wir das AWP des logistischen Wachstums

$$p'(t) = p(t) \cdot \left(1 - \frac{p(t)}{5}\right) \qquad p(0) = 1.$$

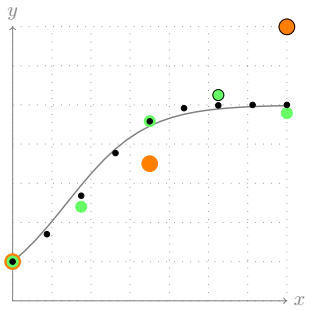
Der explizite Euler-Schritt ist

$$p_1 = p_0(1 + h) - \frac{h}{5} p_0^2$$

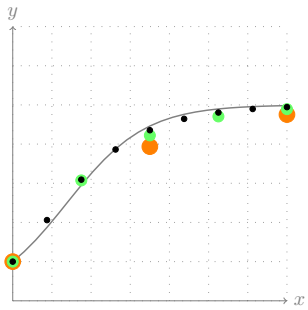
Der implizite Euler-Schritt ist

$$p_1 = \frac{\sqrt{[2.5(1-h)]^2 + 5hp_0} - 2.5(1-h)}{h}$$

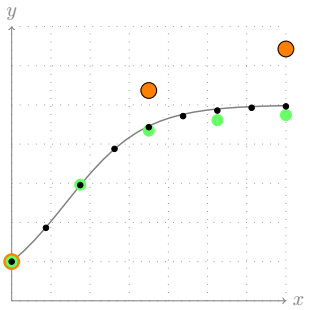
Der verbesserte Euler-Schritt ist ein Polynom 4. Grades in y_0 und h.
Das klassische Runge-Kutta-Verfahren benutzt ein Polynom 16. Grades in y_0 und 15. Grades in h.



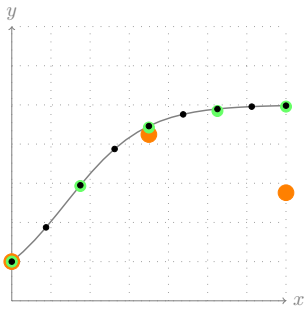
Euler (explizit)



Euler (implizit)



Euler (verbessert)



Runge-Kutta (klassisch)