Numerik II mit Matlab

Hans Werner Borchers
Duale Hochschule Mannheim

Okt. – Nov. 2017

(Version 0.5-1)

Gewöhnliche Differenzialgleichungen

Eine (gewöhnliche) **Differenzialgleichung** (DGL) erster Ordnung ist ein Ausdruck der Form

$$y' = \frac{dy}{dx} = f(x, y), \quad y(x0) = y0$$

Gesucht ist eine Funktion $y = y(x)$ in einer Veränderlichen, welche in einem Intervall $[x0, x1]$ definiert ist und die **Anfangsbedingung** $y(x0) = y0$ erfüllt.
Gewöhnlich: y ist eine Funktion *einer* Veränderlichen, es treten keine partiellen Ableitungen auf.
Erster Ordnung: Es treten nur Ableitungen erster Ordnung auf: y'.

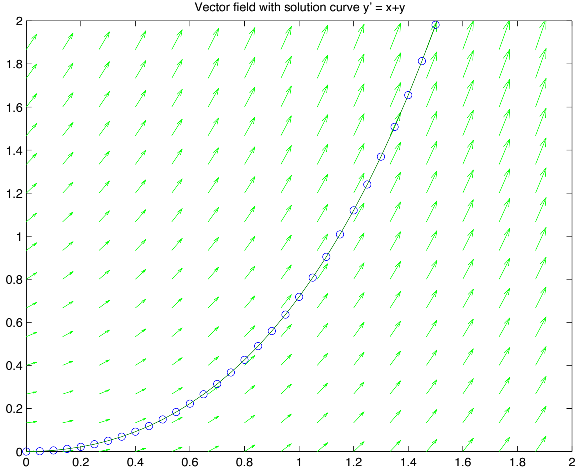
Einige einfache Beispiele

- $y' = x^2$
- $y' = \cos(x)$
- $y' = y$
- $y' = x \cdot y$
- $y' = \frac{1+y}{x}$
- $y' = sin(\frac{y}{x})$
- $y' = y^2$
- $y' = e^y \sin(x)$
- $y' = x^2 + y^2$

Anwendungsbeispiele

- Wachstums-/Zerfallprozesse: $y' = \pm ky$
- Ausbreitung (einer Krankheit): $P' = kP(P_0 - P)$
- Konzentration (chemische Prozesse): $\frac{dC}{dt} = \frac{-k_1 C}{1+k_2 C}$
- Auslaufender Wassertank: $y' = -k\sqrt{y}A(y)$
- Freier Fall mit Luftwiderstand: $v' = -g + k/m\,|v|^p$
- Biologische Raubtier-Beute Modelle
- Federpendel / Mathematisches Pendel
- Elektrischer Schwingkreis

Vektorfeld mit DGL Lösung



Plotten eines Vektorfeldes

Die Funktion `quiver(x,y,u,v)` stellt Vektoren dar mit Komponenten `(u,v)` in den Punkten `(x,y)`. Um ein Vektorfeld anzuzeigen, wird zunächst ein *Grid* erstellt.

```
% Erzeuge ein 22 x 22 Grid with X-, Y-Komponenten
[X, Y] = meshgrid(-2.1:0.2:2.1, -2.1:0.2:2.1);

% Erzeuge Pfeile der Länge 1/5
U = ones(22, 22);
V = -X ./ Y;                          % DGL y' = -x/y
D = 5 * sqrt(U.^2 + V.^2);
U = U ./ D; V = V ./ D;

% Aufruf des Vektorplots
quiver(X, Y, U, V)
```

Eulersches Lösungsverfahren

"Das Eulersche Polygonzugverfahren oder explizite Euler-Verfahren ist das einfachste Verfahren zur numerischen Lösung eines Anfangswertproblems."
Wikipedia

Gegeben eine (gewöhnliche) DGL $y' = f(x, y)$ mit Anfangswertbedingung $y(a) = y0$. Bestimme eine Lösung $y(x)$ im Intervall $[a = x_0, b]$ durch:

1. Unterteile das Intervall $[a, b]$ in $n - 1$ gleichmässig grosse Teilintervalle $[x_1 = a, x_2, \ldots, x_n = b]$ der Länge $h = (b - a)/n$.
2. Allokiere einen Vektor $(y_1, \ldots, y_n)$ der Länge n mit $y(1) = y0$.
3. Für $i = 2$ bis n setze $y_i = y_{i-1} + h \cdot s$, wobei s die Steigung der Lösungskurve in (x_{i-1}, y_{i-1}) ist.

Als Ergebnis werden die Vektoren x und y zurückgegeben.

Euler-Heun Verfahren (in MATLAB)

```
function [x, y] = euler(f, a, b, y0, n)
% Explizites Euler Verfahren
h = (b-a)/n;
x = a:h:b;     % n+1 Punkte
y(1) = y0;
for i = 1:n
  y(i+1) = y(i) + h * f(x(i), y(i));
  % Heun Korrektur
  y(i+1) = y(i) + ...
           h * (f(x(i),y(i)) + f(x(i+1),y(i+1)))/2.0
end
end %Function
```

Symbolische Lösung (mit MATLAB)

MATLAB kann **einige** DGLen lösen mit einem symbolischen Löser in der ‘Symbolic’ Toolbox.

```
dsolve('Dy = f(x, y)', 'y(x0) = y0', 'x')
```

Beispiel: $y' = x + y$

```
dsolve('Dy = x + y', 'y(0) = 0', 'x')
ans =
exp(x) - x - 1
```

Numerische Lösung (mit MATLAB)

MATLAB besitzt verschiedene Solver für DGLen und DGLs-Systeme, etwa ode23, ode45 oder ode15s. Für viele Anwendungen ist **ode45** geeignet, der aufgerufen wird mit:

```
options = odeset('MaxStep', 0.1, 'RelTol', 1e-07)
[x, y] = ode45(f, [a, b], y0, options)
plot(x, y, 'b-o')

% y Lösung der DGL, an den Stellen x berechnet
% f Funktion welche die GGL y' = f(x, y) bestimmt
% [a, b] Integrationsgrenzen
% Anfangsbedingung: y(a) = y0
% Mögliche Optionen, z. B. Genauigkeit
```

Ohne `[x, y]` wird nur die Lösungskurve geplottet.

Numerische Lösung als Struktur

Alle MATLAB Solver können auch in der folgenden Form aufgerufen werden:

```
sol = ode45(f, [a, b], y0)
plot(sol.x, sol.y, 'b-o')
y = deval(sol, x)      % or: [y, yp] = deval(sol, x)
```

`sol` ist eine *Struktur* mit folgenden Komponenten

```
%  solver: 'ode45'
% extdata: [1x1 struct]
%       x: x-Werte
%       y: y-Werte
%   stats: [1x1 struct]
%   idata: [1x1 struct]
```

`deval` berechnet die Lösung der DGL an bestimmten Stellen.

Numerische Lösung mit Ereignis

Definiere Aktionen, die beim Eintreten eines *Ereignisses* auszuführen sind, in einer eigenen *event* Funktion:

```
function [fun, act, dir] = myevent(t, y)
  fun = ...  % Ereignis als Nullstelle
  act = ...  % = 1 Beenden, = 0 Fortsetzen
  dir = ...  % = 1/-1/0 steigend/fallend/immer
end
```

und Aufruf des Lösers mit

```
options = odeset('Events', 'myevent');
ode45(f, [a, b], y0, options)
```

Beispiel: Ausbreitung einer Krankheit

In einer Stadtbevölkerung von $P0$ Personen breite sich eine Krankheit (z. B. Influenza) mit einem Ausbreitungsfaktor k pro Begegnung aus. Ist die Anzahl der anfänglich infizierten Personen P dann gilt die DGL

$$P' = kP(P0 - P)$$

Für $P0 = 25000$, $P = 250$ und $k = 0.00001$ pro Tag.
Bestimme die Anzahl der infizierten Personen nach 30 Tagen;
vergleiche mit einer grösseren Stadt: $P0 = 250000$.

Beispiel: Konzentration in einem chemischen Reaktor

Bei einer chemischen Reaktion erfolge die Veränderung der Konzentration einer chemischen Substanz folgendem dynamischen Verhalten:

$$\frac{dC}{dt} = \frac{-k_1 C}{1 + k_2 C}$$

Bestimme die Endkonzentration für die folgenden beiden Fälle:
(1) $k_1 = 1.0$, $k_2 = 0.8$, Anfangskonzentration: $C(0) = 0.8$
(2) $k_1 = 2.0$, $k_2 = 0.1$, Anfangskonzentration: $C(0) = 1.0$

Beispiel: Auslaufender Wassertank

Toricellis Gesetz: Ein Wassertank (beliebiger Form) habe an seinem Boden eine Öffnung der Grösse a (als Fläche). Beim Auslaufen/Entleeren verhält sich die Füllstandshöhe y nach folgendem Gesetz

$$y' = -k\sqrt{y}A(y)$$

Dabei ist $A(y)$ ie Querschnittsfläche in Höhe des Wasserstandes y, $k = a\sqrt{2g}$, $g = 9.81 m/s^2$, a der Querschnitt das Auslaufs.

Beispiel: "Badewanne" mit a als 6 runde Löcher mit Durchmesser 1 cm, $A(y)$ konstant 1,45 x 0,55 qm, und Füllhöhe 30 cm. Wie lange dauert es in etwa, bis die Badewanne ausgelaufen ist (in min).

Beispiel: Aufflammender Funken

Ein Funke breitet seinen inneren Brennbereich aus, bis ein Gleichgewicht zwischen Verbrauch von Sauerstoff (Volumen) und nachstömendem Sauerstoff von aussen (Oberfläche) erreicht ist.
Bezeichne mit y den Radius des Brennbereiches, dann sollte folgende Differenzialgleichung erfüllt sein (mit nicht spezifizierten Koeffizienten):

$$y' = y^2 - y^3$$

Löse diese DGL mit Anfangsbedingung $y(0) = 1/250$ (man kann nicht $y(0) = 0$ ansetzen – warum nicht?) in den ersten 500 Millisekunden.

Vergleiche die Ergebnisse der Solver `ode45` und `ode15s`!

Beispiel: Freier Fall mit Luftwiderstand

Luftwiderstand: $F = k\, v^p, 1 \leq p \leq 2$, k eine Konstante.
v die Fallgeschwindigkeit (in negativer y-Richtung)

$$v' = -g + \frac{k}{m}|v|^p$$

Beispiel: Für einen Fallschirmspringer gilt ungefähr: $p = 1.1$ und $cw = k/m = 1.6$ bei einem Gewicht von $k = 80$ kg.

- Welche Endgeschwindigkeit wird er erreichen?
- Spielt das Gewicht eine Rolle?
- Wann erreicht er aus 1000 m den Erdboden?
- Und wenn er eine Anfangsgeschwindigkeit von 240 km/Std. hat (z.B. schon in 2000 m aus dem Flugzeug gesprungen ist)?

Lösung freier Fall

- **Datei** `fall.m`

```
function dv = fall(t, v)
% constants
g = 9.81;
k = 130;
m = 80;
% change
dv = -g + k/m * abs(v)^1.1;
end
```

- **Aufruf des Lösers**

```
[t, v] = ode45(@fall, [0, 10], -240000/(60*60));
plot(t, v, '-or'), grid on
```

Systeme von Differenzialgleichungen

Ein **System von** zwei **Differenzialgleichungen** besteht aus *gekoppelten* Differenzialgleichungen der Form

$$y1' = f1(x, y1, y2)$$

$$y2' = f2(x, y1, y2)$$

mit Anfangsbedingungen $y1(x0) = y1$, $y2(x0) = y2$.

Es wird in Matlab gelöst durch folgenden Aufruf

```
ode45(@f, [x0, x1], [y1, y2])
```

wobei die Funktion `f(x, y)` y als Vektor `[y(1), y(2)]` erwartet und die Änderung dy als Spaltenvektor zurück geben muss.
(Siehe das folgende Beispiel.)

Lotka-Volterra (LV) Modell

Raubtier-Beute Modell
Gegeben ein (geschlossenes) “Habitat” mit Füchsen F und Hasen R

$$dR/dt = aR - bRF$$

$$dF/dt = cRF - dF$$

$a = 1.5$ Reproduktionsrate R; $b = 0.02$ Sterberate R
$c = 0.02$ Fressrate F, $d = 3.0$ Aussterberate F

Zeitverlauf: 12 Vierteljahre.
Anfangsbedingung: 100 Hasen und 10 Füchse.

Lösung des Raubtier-Beute-Modells

- **Datei** `ppm.m`

```
function dy = ppm(t, y)
a = 1.5; b = 0.02;
c = 0.02; d = 3.0;
R = y(1); F = y(2);
dR = a*R - b*R*F;
dF = c*R*F - d*F;
dy = [dR; dF];  % Spaltenvektor !!
end
```

- Löse das System wie üblich:
  `ode45(@ppm, [0, 12], [100, 10])`
  und probiere verschiedene Anfangsbedingungen aus.

Beispiel: Modifiziertes LV Modell

Löse das Raubtier-Beute Modell, wenn sich in dem Habitat maximal 300 Hasen ernähren können.
Diese Beschränkung lässt sich ausdrücken in der Form

$$\frac{dr}{dt} = a(1 - \frac{R}{Rmax})R - bRF)$$

Löse dieses Problem mit den genannten Anfangsbedingungen.

Beispiel: Chemische Reaktionskinetik

In einem Tank befinden sich drei Substanzen, die miteinander reagieren in der Intensität bestimmt durch die jeweils vorhandenen Konzentrationen. Die Raktionsgleichungen lauten:

$$y_1' = -k_1 y_1 + k_3 y_3$$

$$y_2' = k_1 y_1 - k_2 y_2$$

$$y_3' = k_2 y_2 - k_3 y_3$$

Reaktionskoeffizienten: $k_1 = 0.3$, $k_2 = 0.2$, $k_3 = 0.5$
Anfangskonzentrationen: $[0.6, 0.2, 0.2]$

Wie verläuft die Reaktion in einem Zeitraum von 10 Minuten?

Differenzialgleichung zweiter Ordnung

Eine (gew.) **Differenzialgleichung** zweiter Ordnung
ist ein Ausdruck der Form

$$y'' = f(x, y, y'), \quad y(x0) = y1,\, y'(x0) = y2$$

Gesucht ist eine Funktion $y = y(x)$ einer Veränderlichen, welche in einem Intervall $[x0, x1]$ definiert ist, deren erste und zweite Ableitungen die Gleichung $y'' = f(x, y, y')$ erfüllen, und die auch die **Anfangsbedingungen** $y(x0) = y1$ und $y'(x0) = y2$ erfüllt.

Bemerkung:
Entsprechend werden DGL höherer Ordnung definiert als:

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

DGLen zweiter Ordnung (in MATLAB)

Gegeben die DGL zweiter Ordnung $y'' = f(x, y, y')$.

Definiere zwei neue Funktionen $y_1 = y$ und $y_2 = y'$,
dann gilt $y_1' = y'$ und $y_2' = y''$, also

$$y_1' = y_2$$

$$y_2' = f(x, y_1, y_2)$$

Die Matlab Funktion für diese System von DGLen enthält folgende Zeile:

```
dy = [ y(2)
       f(x, y(1), y(2) ]
```

`f` bezeichnet die in Matlab umgewandelte mathematische Funktion f.

Beispiel: Schwingung

Die einfachste Beschreibung einer Schwingung als DGL lautet:

$$y'' = -y$$

Die entsprechende MATLAB Funktion als *anonyme* Funktion ist dann

```
f = @(t, y) [y(2); -y(1)];
```

Löse die DGL und stelle die Funktion dar mit den Anfangsbedingungen $y(0) = 0$, $y'(0) = 1$ im Intervall $[0, 10]$.

Beispiel: Mathematisches Pendel

Das *mathematische* (oder ideale) Pendel wird unter dem Einfluss der Gravitation beschrieben durch eine DGL zweiter Ordnung:

$$u'' + \frac{g}{L} sin(u) = 0$$

Dabei sei u der Winkel der Auslenkung vom Ruhezustand.
Löse dieses Gleichung mit den Anfangsbedingungen $u(0) = \pi/2$, $u'(0) = 0$ und den Konstanten $g = 9.81$ m/s^2 und $L = 1$ m.

Vergleiche dies mit der Vereinfachung $sin(u) \approx u$ für $u << 1$ und veranschauliche beide Lösungen in einem gemeinsamen Plot.

Unterscheiden sich die maximalen Auslenkungen?
Welches Pendel schwingt langsamer?

Lösung Mathematisches Pendel

- Löse das (ungedämpfte) ideale Pendel $y'' = -g/L \cdot sin(y)$ durch Einführung zweier Funktionen $y1 = y$, $y2 = y'$ als

$$y1' = y2, y2' = -g/L \cdot sin(y1)$$

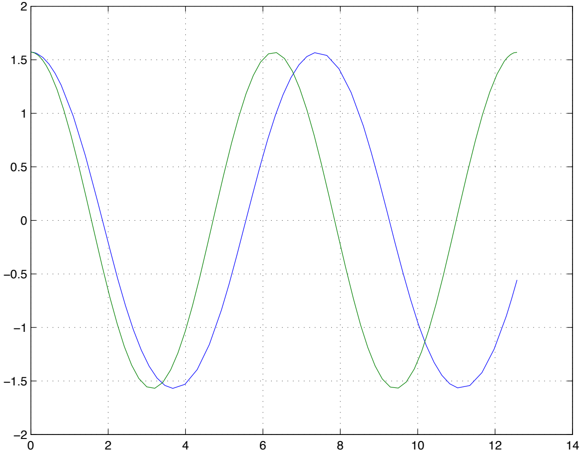
- **Datei** `pendel.m`

```
function dy = pendel(t, y)
g = 9.81; L = 0.1;
y1 = y(1); y2 = y(2);
dy1 = y2;  dy2 = -g/L * sin(y1);
dy = [dy1; dy2];
```

- Löse das System mit Auslenkung 90 Grad:

```
ode45(@pendel, [0, 10], [pi/2, 0])
```

Vergleich der Pendelmodelle



Gedämpftes Federpendel

Die Bewegungsgleichung für ein *gedämpftes* Federpendel lautet (Rückstellkraft proportional zur Auslenkung):

$$\frac{d^2 z}{dt^2} + 2\delta \frac{dz}{dt} + \frac{k}{m} z = 0$$

- Lösung ohne Dämpfungsterm: $z = a\cos(\omega t + \phi)$ mit $\omega = \sqrt{k/m}$.
- Mit Dämpfungsterm unterscheidet man die Fälle $\delta < \omega$, $\delta = \omega$, und $\delta > \omega$.
- Die *erzwungene* Schwingung

$$\frac{d^2 z}{dt^2} + 2\delta \frac{dz}{dt} + \frac{k}{m} z = \frac{F_0}{m} \sin(\omega_e t)$$

hat eine Resonanz bei $\omega_e = \sqrt{\omega^2 - 2\delta^2}$.

Beispiel: Federpendel

Löse die DGL für eine gedämpfte Schwingung

$$\frac{d^2z}{dt^2} + 2\delta\frac{dz}{dt} + \frac{k}{m}z = 0$$

- mit Dämpfungsfaktor $\delta = 1$ und folgenden Konstanten: $k = 5$ N/m, $m = 0.1$ kg, $z(0) = 0.1$ m und $z'(0) = 0$ m/s.
- als erzwungene Schwingung mit einer äusseren Kraft $\frac{F_0}{m}sin(\omega_e t)$ wobei $F_0 = 1$ N und $\omega_e = 7$ bzw $\omega_e = \sqrt{50}$.
- Was geschieht, wenn die Dämpfung wegfällt?

Freier Fall (noch einmal)

Der freie Fall mit Luftwiderstand kann besser beschrieben werden als eine DGL zweiter Ordnung mit folgender Gleichung:

$$y'' = -g + c_w \rho A v^2 / 2$$

Es ist $\rho = 1.3$ kg/m^3 der Luftdruck (in Meereshöhe) und A die Projektionsfläche in Bewegungsrichtung.

Ein Basketball wird vom obersten Stockwerk des One World Trade Center in New York heruntergeworfen (Höhe 387 m). Der Ball hat einen Durchmesser von 24 cm und als Kugelform den c_w-Wert 0.1.

Nach wieviel Sekunden genau erreicht der Ball den Erdboden?