

12

DGLn höherer Ordnung. Lösung mit Standardsoftware

Jörn Loviscach

Versionsstand: 21. März 2014, 21:09

Die nummerierten Felder sind absichtlich leer, zum Ausfüllen beim Ansehen der Videos:
<http://www.j3L7h.de/videos.html>



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Germany License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/de/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Bitte hier notieren, was beim Bearbeiten unklar geblieben ist

1 Idee

Gegeben sei die Differentialgleichung

$$y''' + y'' y' + \sin(x)y^2 + x^3 = 0.$$

Was für ein Typ an DGL ist das?

Was müsste man als Anfangsbedingung zum Beispiel vorgeben?

Wie sieht also der Zustandsraum (= Phasenraum) aus?

3

Diese Differentialgleichung kann man – wie jede andere dieser Art – mit einem Trick als Vektor-Differentialgleichung (= DGL-System) *erster* Ordnung schreiben:

4

Programme zum numerischen Lösen von Differentialgleichungen wollen meist genau diese Form haben: explizit, erste Ordnung.

2 Differentialgleichungen mit Software lösen

In Wolfram Alpha tippt man `solve y'+5y=x^2` für die allgemeine Lösung oder `solve y'+5y=x^2 and y(13)=42` für eine spezielle Lösung ein. Wenn Alpha das nicht mehr symbolisch (also mit einer „analytischen“ Formel) lösen kann, gibt es eine numerisch berechnete Kurve aus.

MATLAB[®] schaltet nicht automatisch auf numerische Lösung um. Symbolisch arbeitet `dsolve('Dy+5*y=x^2','x')` für die allgemeine Lösung oder `dsolve('Dy+5*y=x^2','y(13)=42','x')` für eine spezielle Lösung. Das 'x' am Ende gibt die unabhängige Variable an; ohne diese Angabe ist die unabhängige Variable t . Wenn das Programm keine symbolische Lösung findet, liefert es eine leere Antwort. Wie in Wolfram Alpha sind auch Differentialgleichungen höherer Ordnung erlaubt:

```
dsolve('D2x+x=t^3','x(0)=3','Dx(0)=7')
```

Weil hier am Ende keine unabhängige Variable ausdrücklich angegeben ist, nimmt das Programm t als unabhängige Variable.

Für die numerische Lösung hat MATLAB[®] diverse Funktionen, von denen `ode45` die übliche ist. Diese nimmt nur explizite Differentialgleichungen erster Ordnung entgegen. Also muss man umformen. Aus $y' + 5y = x^2$ macht man:

5

Die Eingabe sieht dann so aus:

```
[x, y]=ode45(@ (x, y) -5*y+x^2, [13, 17], 42);
```

Der Ausdruck $@(x, y) -5*y+x^2$ ist eine Funktion von x und y , die $-5y + x^2$ berechnet. Die beiden Werte $[13, 17]$ geben das erste und das letzte x an, die in der Wertetabelle auftauchen sollen. Die 42 ist der Startwert $y(13)$. Ausgegeben werden zwei Arrays (in diesem Fall x und y) mit der Wertetabelle. Diese kann man zum Beispiel mit `plot(x, y)` darstellen.

Für eine Differentialgleichung zweiter Ordnung wird es aufwendiger. Sei die Differentialgleichung $\ddot{x} + 0,01\dot{x} + 0,2x = \cos(0,1t)$ zum Anfangswert $x(2) = 3$, $\dot{x}(2) = 1$ zu lösen. Dann muss man diese in ein explizites Differentialgleichungssystem erster Ordnung umwandeln:

6

Einzutippen ist dann:

```
[t, x]=ode45(@ (t, x) [x(2); cos(0.1*t)-0.01*x(2)-0.2*x(1)], ...
    [2, 1000], [3; 1]);
```

Das Array x wird nun eine Matrix mit zwei Spalten. In der ersten Spalte stehen die Werte von x , in der zweiten von \dot{x} . Das Ergebnis kann man nun zum Beispiel mit `plot(t, x(:, 1), 'k', t, x(:, 2), 'b')` ausgeben.

Ohne die äußere Anregung ergeben sich schöne Phasenplots:

```
[t, x]=ode45(@ (t, x) [-0.01*x(2)-0.2*x(1)], [2, 1000], [3; 1]);
plot(x(:, 1), x(:, 2))
```

Vorsicht: An Stellen wie den Differentialgleichungslösern hier sieht man in den Clones von MATLAB® oft andere Schreibweisen.