Math. Einschub: DGL (Differentialgleichungen)

Gewöhnliche Gleichung:

y = f(x)

Abbildung: $x \rightarrow y$

x ist vorgegeben +
Vorschrift, wie daraus y berechnet wird

DGL:

Gleichungen, die sowohl x und y als auch Ableitungen von y enthalten.

gesucht: Alle Funktionen $y(x)$, die diese DGL **für alle** x erfüllen.

Beispiel:

$y' = 2$

d.h. gesucht sind alle Funktionen $y(x)$, für die y' gleich 2 ist, für alle x.

Lösung:

$y(x) = 2\,x + c$, c beliebig reell

Probe: $y' = 2$, für alle x

Als Lösungen bekommen wir eine ganze Schar von Kurven, die sich durch die beliebige Verschiebung c unterscheiden.

Klassifikation der DGLn:

a) gewöhnliche <-> partielle DGL

unbekannte Funktion $y(x)$ hat ein Argument x, bzw. mehrere Argumente $y(x_1, x_2, \ldots)$

b) linear <-> nichtlinear

nur lineare Terme in y und seinen Ableitungen bzw. mindestens ein nichtlinearer Term

c) homogen <-> nichthomogen

nur Terme mit y und seinen Ableitungen bzw. auch andere additive Terme, z.B. Konstante

Ordnung der DGL:
höchste auftretende Ableitung von y

Grad der DGL:
höchste Potenz von y bzw. y-Ableitungen

Aufgabe:

Konstante Kraft in z-Richtung: $F = -m\,g$

Bewegungsgleichungen:

$$m\, d^2z / dt^2 = -m\,g$$

Lösung der DGL

Lösung:

DGL: $d^2z / dt^2 = -g$

ist lineare DGL für $z(t)$.

Methode: zweifache Integration

=> $dz/dt = -\int g\,dt = -g\,t + c1$

=> $z(t) = -g/2\,t^2 + c1\,t + c2$

ist Lösung für freien Fall

Beispiel: Springender Ball

Newton: $\boldsymbol{F} = m\,\boldsymbol{a} = m\,d^2\boldsymbol{r}\,/\,dt^2$

Beteiligte Kräfte:

a) Gewichtskraft: $-\,m\,g\,\boldsymbol{e}_z$

b) Elastische Kraft (Feder) beim Aufprall: $-\,k\,\boldsymbol{z}$

c) Energieverlust bei Verformung: $-\,c\,\boldsymbol{v}_z$

ergibt folgendes DGL-System:

$m\,d^2x\,/\,dt^2 = 0$ (Annahme: keine Luftreibung)

$m\,d^2z\,/\,dt^2 = -\,m\,g - \Theta(-z)\,(\,k\,z + c\,dz/dt\,)$

$\Theta(z)$: Theta-Funktion, 1 für $z>0$, sonst 0

Problem: geschlossene analytische Lösung

Numerische Lösung mit MATLAB:

für unterschiedliche Anfangsbedingungen

function movie_Ball(*v*x0, *v*y0, *t*m, *k*, *v*)

>> movie_Ball(10, 0, 15, 1000, 0.5)