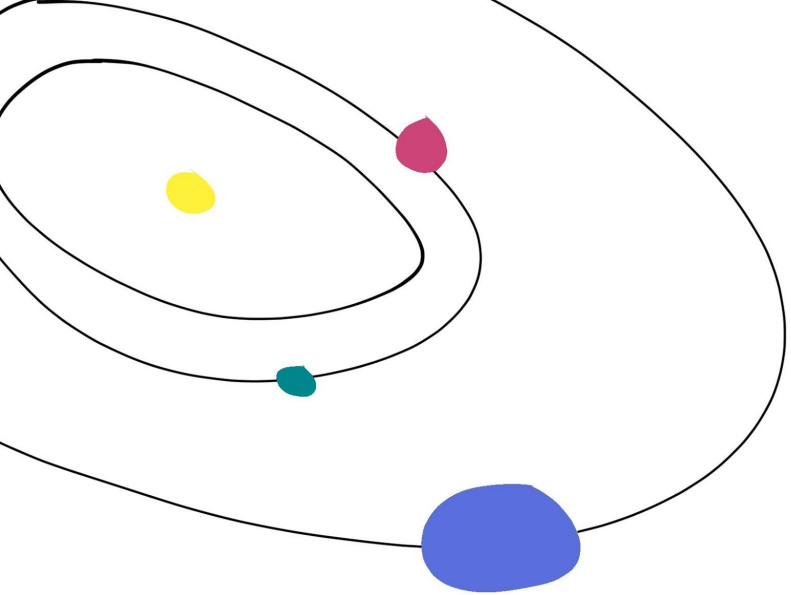
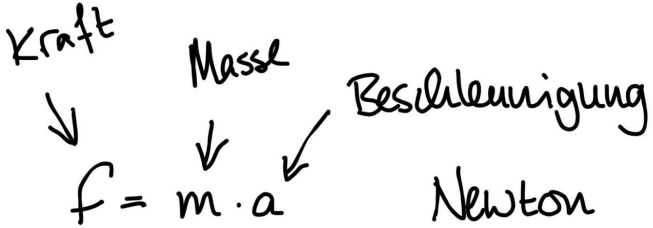


Numerische Ansätze für physikalische Probleme

Dynamik in der Physik

- Zusammenhänge werden meist über Formeln ausgedrückt.
- Ein paar Beispiele:



$$i\hbar \frac{\partial}{\partial t} \Psi(x,t) = -\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} \Psi(x,t)$$

$$\vec{\nabla} \cdot \vec{E} = \rho / \varepsilon_0$$

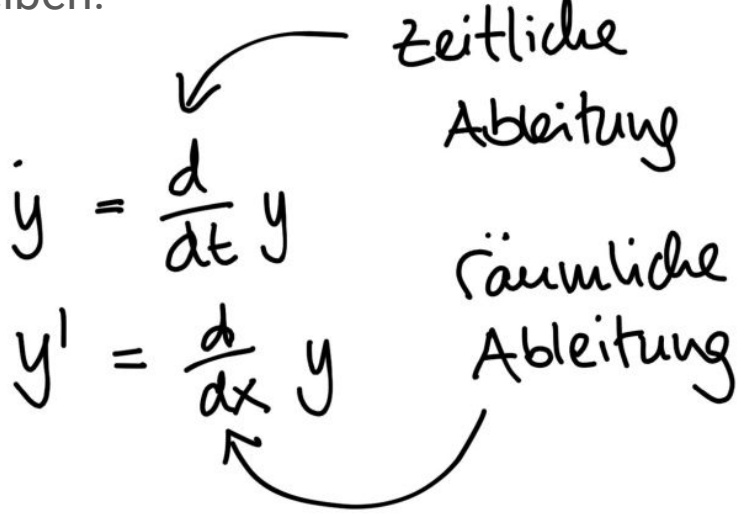
Es handelt sich um sogenannte **Differenzialgleichungen**

Differentialgleichungen (DGL)

- Es wird die **Änderung einer Größe** mit sich selbst oder anderen Größen in Zusammenhang gebracht.

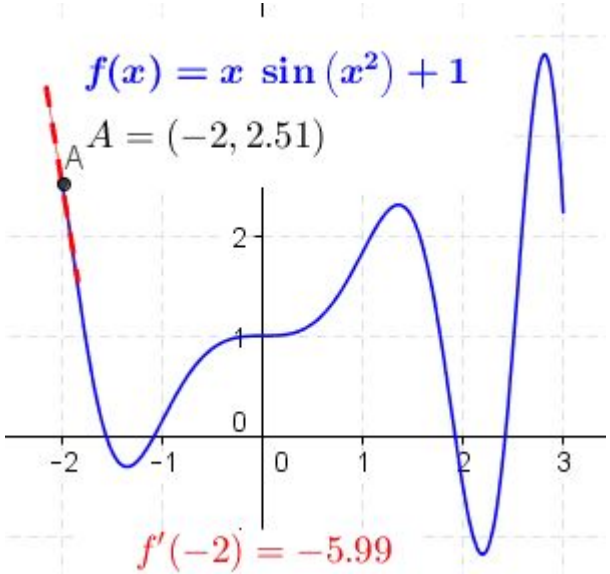
- Es gibt verschiedene Arten das aufzuschreiben.

$$f(y) = f(y', y'', \ldots)$$



Ableitung?!!?!

- Die **Ableitung** einer Funktion gibt deren **Steigung** (Änderung) an.
- Es gibt diverse Regeln, mit denen man Ableitungen berechnen kann - einige Ableitungen sollte man kennen.



$$\frac{d}{dt}\sin(t) = \cos(t)$$

$$\frac{d}{dx}e^x = e^x$$

$$\frac{d}{dt}\cos(t) = -\sin(t)$$

Die Lösung von DGLs

- DGLs lassen sich **nicht durch einzelne Zahlenwerte** lösen, wie andere Gleichungen.
- Lösungen von DGLs sind selbst **Funktionen!**

$$y = \dot{y}$$

$$y = e^t \quad \text{da} \quad \dot{y} = e^t$$

$$y = -\ddot{y}$$

$$y = \sin(t) \quad \dot{y} = \cos(t) \quad \ddot{y} = -\sin(t)$$

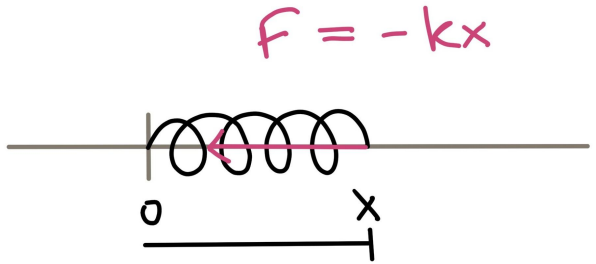
Zurück zur Physik: Dynamik einer Feder

Newton $f = ma = m\dot{v} = m\ddot{x}$

Hook'sches Gesetz $f = -kx$

Federkonstante

Auslenkung

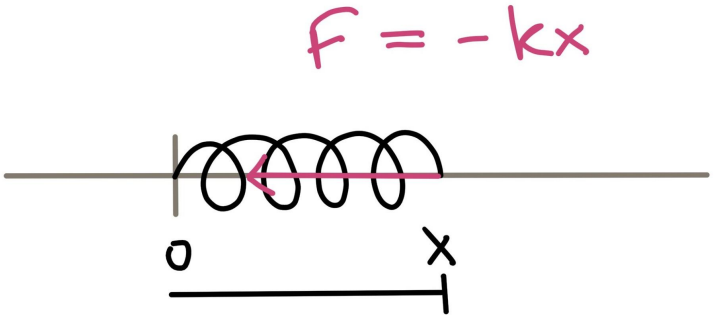


… noch mehr Formeln …

$$m\ddot{x} = -kx$$

$$\ddot{x} = -\frac{k}{m}x$$

$$x(t) = \cos\left(\sqrt{\frac{k}{m}}x\right)$$

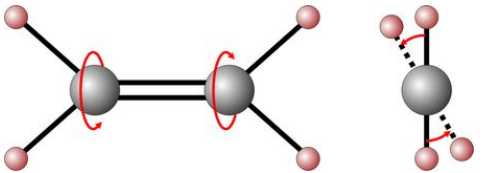


Harmonischer Oszillator

$$\ddot{x} = -\omega^2 x$$

$$x(t) = \cos(\omega t)$$

"Harmonischer Oszillator"



Step by Step: das Euler-Verfahren

$$m\ddot{x} = -kx$$

$$x_1 = x_0 + v_0 \Delta t$$

$$v_1 = v_0 + a_0 \Delta t$$

$$a_0 = -\frac{k}{m} x_0$$

Zeitschritte Δt

Anfangsbed. $t_0 = 0$, $x_0 = 1$, $v_0 = 0$

$m = 1$

$k = 1$

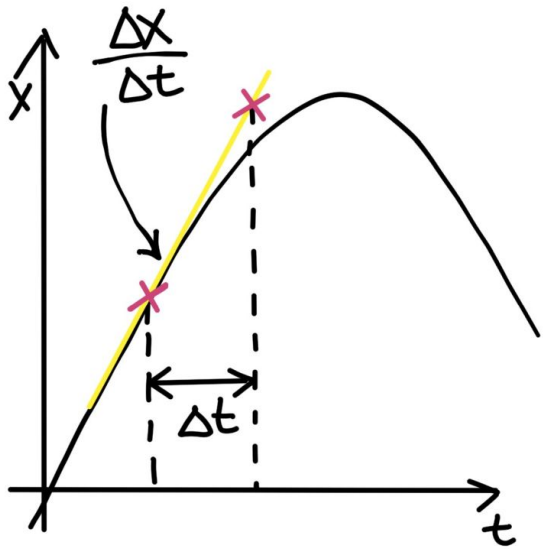
Step by Step: das Euler-Verfahren

$$m\ddot{x} = -kx$$

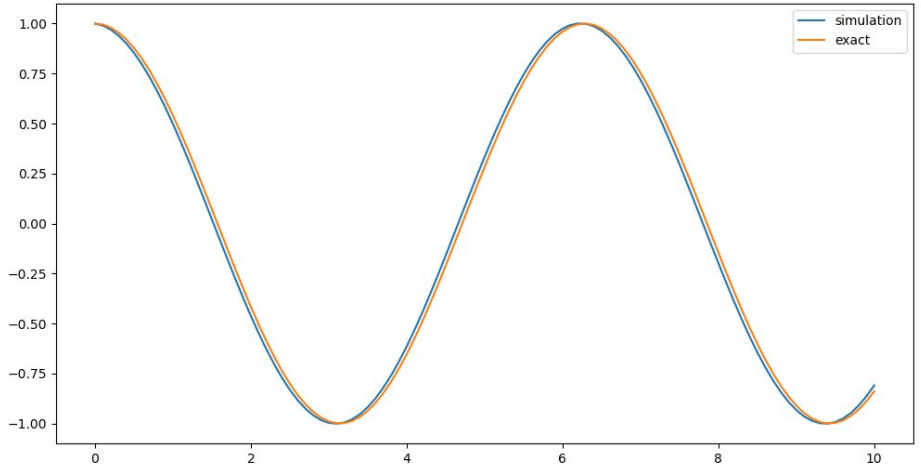
$$a_t = -\frac{k}{m} x_t$$

$$v_{t+1} = v_t + a_t \cdot \Delta t$$

$$x_{t+1} = x_t + v_t \cdot \Delta t$$



Implementieren …




```python
t_0, x_0, v_0 = 0, 1, 0
m, k = 1, 1

def acc(x_t, m, k):
    return -k*x_t/m

def v_step(v_t, a_t, dt):
    return v_t + (a_t*dt)

def x_step(x_t, v_t, dt):
    return x_t + v_t*dt
```

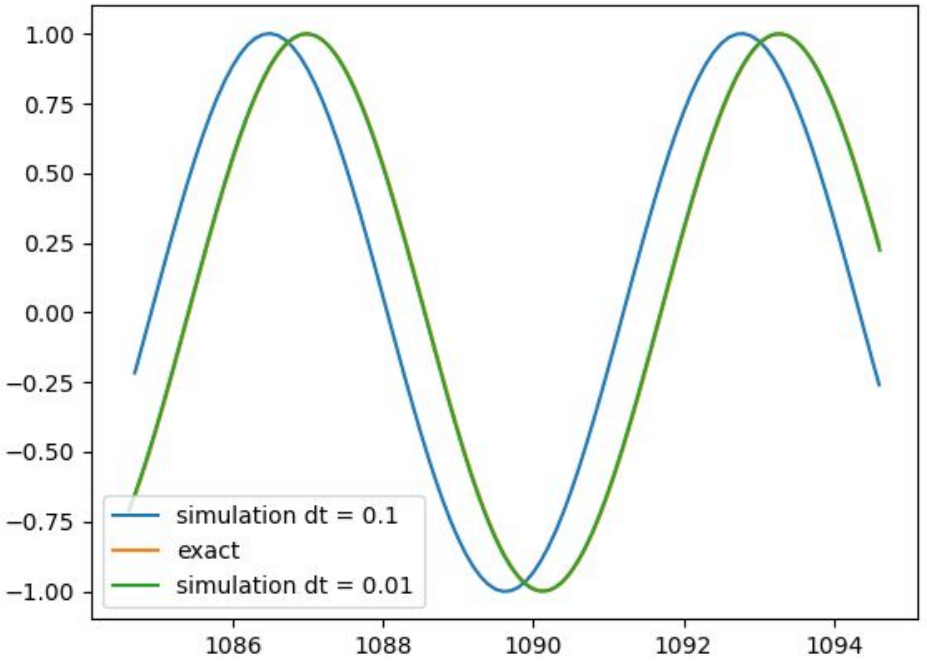
```python
steps = 100
dt = 0.1

t = t_0
x = x_0
v = v_0
a = acc(x, m, k)
```

```python
for step in range(steps):
    t = t + dt
    a = acc(x, m, k)
    v = v_step(v, a, dt)
    x = x_step(x, v, dt)
    numeric.append([t, x, v, a])
    exact.append([t, math.cos(t)])
```

Ergebnisse … wenn man länger wartet

- Nach etwa 10000 Schritten ist die Abweichung von numerischer und exakter Lösung größer als 0.5

- Um bessere Ergebnisse zu erzielen kann man die Schrittweite verkleinern, z.B. von 0.1 auf 0.01

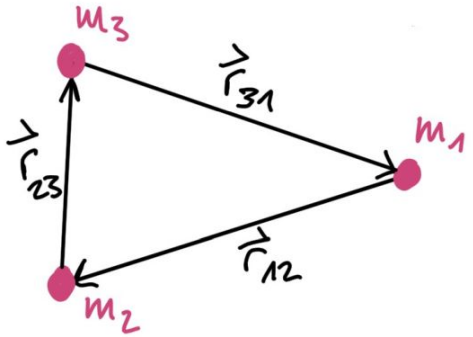


Wann man Numerik wirklich braucht

- gekoppelte, nichtlineare Systeme
  - Die zugehörigen Gleichungen lassen sich nicht mehr analytisch lösen!
  - Die Physik der komplexen Systeme wurde erst mit leistungsfähigen Rechnern möglich.

- Beispiele
  - Doppelpendel
  - Planetensysteme (mehr als 2 Körper)
  - Wetter

Drei Planeten - das “Dreikörperproblem”



$$F_{12} = G \frac{m_1 m_2}{r_{12}^2}$$

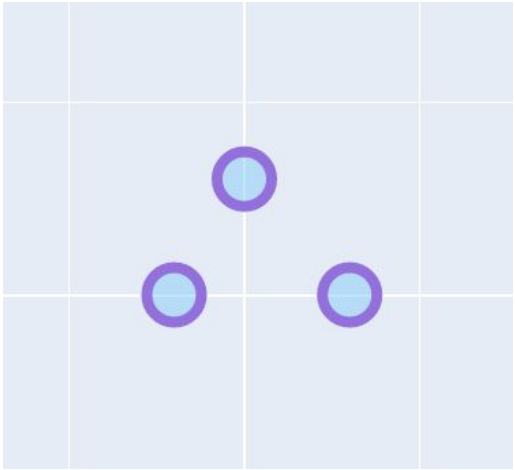
2D: Achtung Vektoren!

$$m_3 \vec{a}_3 = G \left(\frac{m_3 \cdot m_1}{r_{31}^2} \vec{r}_{31} + \frac{m_2 \cdot m_3}{r_{23}^2} \vec{r}_{32} \right)$$

Simulation

```
def gravitational_force(x1, y1, m1, x2, y2, m2):
    force = -1* m1*m2/((x2-x1)**2 + (y2-y1)**2)
    norm_factor = 1/((x2-x1)**2 + (y2-y1)**2)**0.5
    x, y = (x2-x1)*norm_factor, (y2-y1)*norm_factor
    return force*x, force*y
```

```
def step(self, force_x, force_y, dt):
    self.vx = self.vx + dt* force_x/self.m
    self.vy = self.vy + dt* force_y/self.m
    self.xt = self.x + dt* self.vx
    self.yt = self.y + dt* self.vy
```



Zusammenfassung

- Numerische Simulationen sind ein wichtiges Tool in der Physik
- Jedes numerische Verfahren erzeugt langfristig **Fehler**.
- Neben dem einfachen Euler-Verfahren gibt es auch etwas geschicktere Algorithmen, die physikalische Gesetze wie Energie- und Impulserhaltung gewährleisten.
- Bessere Verfahren sind z.B. das **Leap-Frog-Verfahren** und **Runge-Kutta-Verfahren höherer Ordnung** (zu dieser Klasse von Verfahren gehört auch das Euler-Verfahren).