

Bewegung von Himmelskörpern beschrieben durch Gravitationsgesetz

$$F_G = G \cdot \frac{m_1 \cdot m_2}{r^2} \hat{r}$$

Gemäß Newton gilt:

$$F = m \cdot a$$

Differentialgleichung eines Bewegungssystem von N -Körpern:

$$m \cdot \frac{d^2 r}{dt^2} = F \quad (\text{DGL 2. Ordnung})$$

Für die numerische Lösung von DGL 2. Ordnung in Python wird die DGL 2. Ordnung in zwei DGL 1. Ordnung umgewandelt:

$$\frac{dr}{dt} = v = \text{Geschwindigkeit} \quad r = \text{Abstand}$$

$$\frac{dv}{dt} = a = \text{Beschleunigung}$$

Numerische Methoden zur Lösung von DGL

- Euler Methode $\rightarrow$ einfach, aber ungenau
- Runge-Kutta-Verfahren
- in Python nutzt man `scipy.integrate.odeint`, um gewöhnliche DGL 1. Ordnung der Form

$$\frac{dy}{dt} = f(y, t) \quad \text{numerisch zu lösen}$$