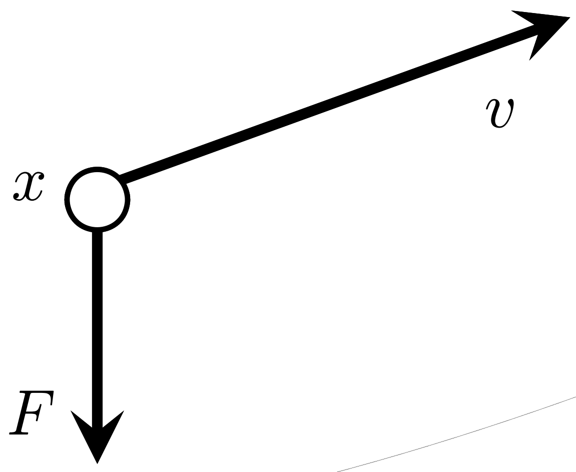
Prof. Dr. Jan Bender

Dynamische Simulation von Mehrkörpersystemen

# PARTIKEL

# Bewegung eines Partikels

- Physikalische Eigenschaften
  - Masse
  - Position
  - Geschwindigkeit
  - Kräfte

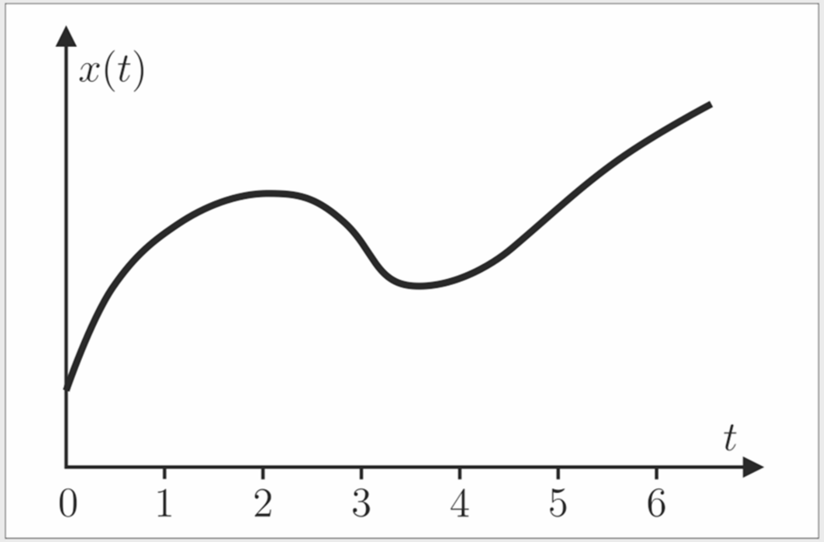


# Bewegung eines Partikels

- Kräfte können unabhängig von den Partikeln sein
  - z.B. Gravitation, Wind, …
- oder sie sind abhängig von Position, Geschwindigkeit, Zeit
  - z.B. Federkraft $F = -k\Delta l$

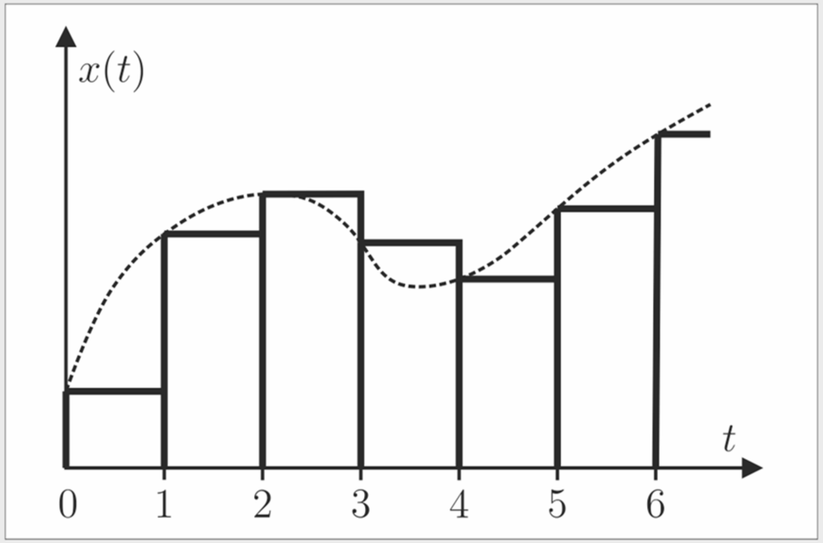
# Bewegung eines Partikels

kontinuierlich




$$\dot{x} = f(x)$$

diskret




$$x(t+1) = f(x(t))$$

# Newtonsche Gesetze

## *Das Trägheitsprinzip („lex prima“)*

*Ein Körper ändert seinen Bewegungszustand nicht von selbst, sondern nur durch die Einwirkung von äußeren Kräften.*

## *Das Aktionsprinzip („lex secunda“)*

*Die Änderung der Bewegung ist der Einwirkung der bewegenden Kraft proportional und geschieht nach der Richtung derjenigen geraden Linie, nach welcher jene Kraft wirkt.*

## *Das Reaktionsprinzip („lex tertia“)*

Übt ein Körper auf einen anderen Körper eine Kraft (actio) aus, dann wirkt eine gleich große Kraft (reactio) vom zweiten Körper auf den ersten in entgegengesetzte Richtung.

# Bewegungsgleichung

- Bewegung wird durch gewöhnliche Differential-gleichung zweiter Ordnung beschrieben:

$$\ddot{\mathbf{x}} = \frac{\mathbf{F}(\mathbf{x},\dot{\mathbf{x}},t)}{m}$$

- Umformulierung als gewöhnliche Differentialgleichungen erster Ordnung

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \frac{\mathbf{F}}{m} \end{aligned}$$

# Partikelsystem

- Differentialgleichung

$$\frac{d}{dt}\begin{pmatrix}\mathbf{x}\\ \dot{\mathbf{x}}\end{pmatrix} = \frac{d}{dt}\begin{pmatrix}\mathbf{x}\\ \mathbf{v}\end{pmatrix} = \begin{pmatrix}\mathbf{v}\\ \mathbf{M}^{-1}\mathbf{F}(\mathbf{x},\mathbf{v})\end{pmatrix}$$

- Massenmatrix

$$\mathbf{M}_{i,j} = \begin{cases} m_i\mathbf{E}_3 & \text{wenn } i = j\\ \mathbf{0} & \text{sonst}\end{cases}$$

# Beispiel

- *n* Partikel mit Gravitation (*m*=1)

$$\mathbf{M} = \mathbf{E}_{3n} \qquad \mathbf{F}_i = \begin{pmatrix} 0 & -9.81 & 0 \end{pmatrix}^T$$

- Bewegungsgleichungen

$$\begin{aligned} \mathbf{v}(t_0 + h) &= \mathbf{v}(t_0) + \mathbf{M}^{-1}\mathbf{F}h \\ \mathbf{x}(t_0 + h) &= \mathbf{x}(t_0) + \mathbf{v}(t_0)h + \frac{1}{2}\mathbf{M}^{-1}\mathbf{F}h^2 \end{aligned}$$

# Euler-Verfahren

- Anfangswertproblem

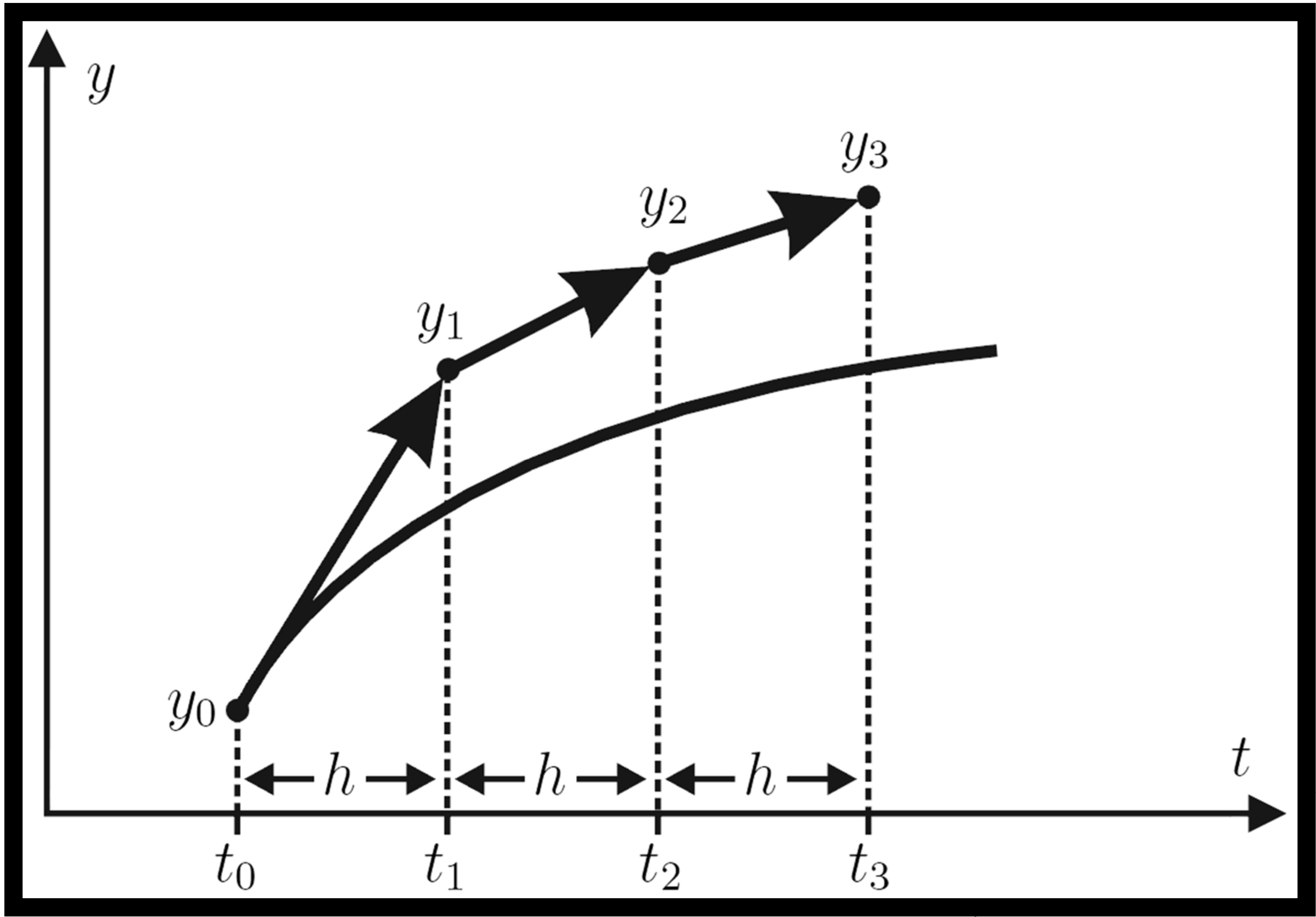
$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}), \qquad \mathbf{x}_0 = \mathbf{x}(t_0)$$

- Zeitschritt mit Schrittweite *h*

$$\mathbf{x}(t_0 + h) = \mathbf{x}_0 + h\mathbf{f}(t_0, \mathbf{x}_0)$$

- Fehler: $O(h^2)$

# Euler-Verfahren



# Verlet-Verfahren

- Geschwindigkeit, Fehler: $O(h)$

$$\mathbf{v}(t_0 + h) = \frac{\mathbf{x}(t_0+h) - \mathbf{x}(t_0)}{h}$$

- Position, Fehler: $O(h^4)$

$$\mathbf{x}(t_0 + h) = 2\mathbf{x}(t_0) - \mathbf{x}(t_0 - h) + \mathbf{a}(t_0)h^2$$

# Leapfrog-Verfahren

- Geschwindigkeit, Fehler: $O(h)$

$$\mathbf{v}(t_0 + h) = \mathbf{v}(t_0) + \frac{\mathbf{a}(t_0)+\mathbf{a}(t_0+h)}{2}h$$

- Position, Fehler: $O(h^4)$

$$\mathbf{x}(t_0 + h) = \mathbf{x}(t_0) + \mathbf{v}(t_0)h + \frac{1}{2}\mathbf{a}(t_0)h^2$$

# Runge-Kutta-Verfahren vierter Ordnung

- Zeitschritt mit Schrittweite *h*

$$
\begin{aligned}
\mathbf{k}_1 &= h\,\mathbf{f}(t_0,\ \mathbf{x}_0) \\
\mathbf{k}_2 &= h\,\mathbf{f}(t_0 + \frac{1}{2}\,h,\ \mathbf{x}_0 + \frac{1}{2}\,\mathbf{k}_1) \\
\mathbf{k}_3 &= h\,\mathbf{f}(t_0 + \frac{1}{2}\,h,\ \mathbf{x}_0 + \frac{1}{2}\,\mathbf{k}_2) \\
\mathbf{k}_4 &= h\,\mathbf{f}(t_0 + h,\ \mathbf{x}_0 + \mathbf{k}_3) \\
\mathbf{x}(t_0 + h) &= \mathbf{x}_0 + \frac{1}{6}\,(\mathbf{k}_1 + 2\,\mathbf{k}_2 + 2\,\mathbf{k}_3 + \mathbf{k}_4)
\end{aligned}
$$

- Fehler: $O(h^5)$

# Literatur

- William H. Press, „Numerical Recipes in C“,
  http://www.nrbook.com/a/bookcpdf.php
- David Baraff, „Physically Based Modeling“, Siggraph 2001 course notes,
  http://www.pixar.com/companyinfo/research/pbm2001