

# PVK Numerische Methoden

## Tag 4

Lucas Böttcher

ETH Zürich  
Institut für Baustoffe  
Wolfgang-Pauli-Str. 27  
HIT G 23.8  
8093 Zürich

*lucasb@ethz.ch*

June 22, 2017

## **Kurstag**    **Inhalt**

Tag 1    Differenzialgleichungen

Tag 2    Quadratur, Nichtlineare algebraische Gleichungen

Tag 3    Ausgleichsrechnung, Eigenwerte, Interpolation

Tag 4    **Lineare ODEs, Exponentielle Integratoren, Splitting-Verfahren**

Tag 5    Erweiterte Übungen

## Kapitel VII

Lineare ODEs (pp. 214–216 und 271–275 im Skript)

Gegebene lineare, zeitabhängige Differentialgleichungen der Form  $\dot{\mathbf{y}} = A(t)\mathbf{y}$  mit  $\mathbf{y}(0) = \mathbf{y}_0$  sollen mit geeigneten Verfahren gelöst werden.

**Lernziele:** Kennen und Anwenden von Magnus-Integratoren, um lineare, zeitabhängige Differentialgleichungen der Form  $\dot{\mathbf{y}} = A(t)\mathbf{y}$  mit  $\mathbf{y}(0) = \mathbf{y}_0$  zu lösen.

## Lineare ODEs

**Grundidee:** Als Beispiel eines Systems  $\dot{\mathbf{y}} = A(t)\mathbf{y}$ , schauen wir uns die Mathieu Differentialgleichung an, cf. p. 274 im Skript.

Die **Mathieu Differentialgleichung** ist  $\ddot{y} + (\omega^2 + \epsilon \cos(t))y = 0$  mit  $y(0) = 1, \dot{y}(0) = 0$ .

## Lineare ODEs

**Grundidee:** Als Beispiel eines Systems  $\dot{\mathbf{y}} = A(t)\mathbf{y}$ , schauen wir uns die Mathieu Differentialgleichung an, cf. p. 274 im Skript.

Die **Mathieu Differentialgleichung** ist  $\ddot{y} + (\omega^2 + \epsilon \cos(t))y = 0$  mit  $y(0) = 1, \dot{y}(0) = 0$ .

Ein **Magnus-Verfahren** 4. Ordnung sieht zum Beispiel wie folgt aus  $\mathbf{y}_{n+1} = e^{\Omega^{[4]}} \mathbf{y}_n$ .

Hier ist  $\Omega^{[4]} = \frac{h}{6} (A_1 + 4A_2 + A_3) - \frac{h^2}{72} [A_1 + 4A_2 + A_3, A_3 - A_1]$  wobei

$$A_1 = A(t_n),$$

$$A_2 = A(t_n + h/2),$$

$$A_3 = A(t_n + h).$$

## Lineare ODEs

**Grundidee:** Als Beispiel eines Systems  $\dot{\mathbf{y}} = A(t)\mathbf{y}$ , schauen wir uns die Mathieu Differentialgleichung an, cf. p. 274 im Skript.

Die **Mathieu Differentialgleichung** ist  $\ddot{y} + (\omega^2 + \epsilon \cos(t))y = 0$  mit  $y(0) = 1, \dot{y}(0) = 0$ .

Ein **Magnus-Verfahren** 4. Ordnung sieht zum Beispiel wie folgt aus  $\mathbf{y}_{n+1} = e^{\Omega^{[4]}} \mathbf{y}_n$ .

Hier ist  $\Omega^{[4]} = \frac{h}{6} (A_1 + 4A_2 + A_3) - \frac{h^2}{72} [A_1 + 4A_2 + A_3, A_3 - A_1]$  wobei

$$A_1 = A(t_n),$$

$$A_2 = A(t_n + h/2),$$

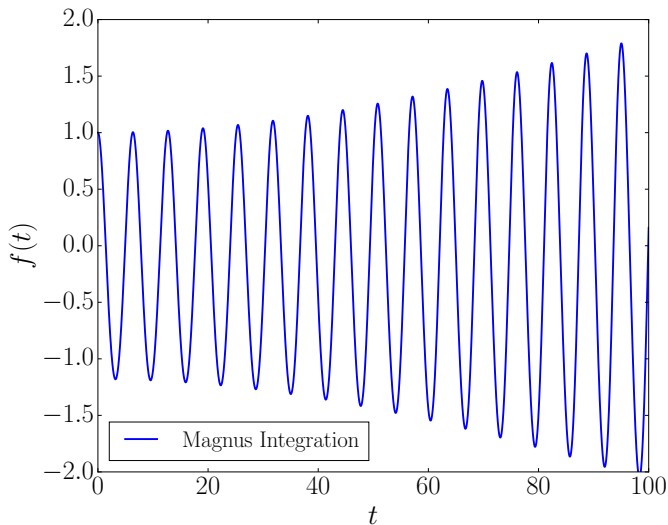
$$A_3 = A(t_n + h).$$

Für nicht-zeitabhängige Matrizen  $A$  nutzt man das Matrixexponential [scipy.linalg.expm](#).

## Lineare ODEs

Nutze die Implementierung vom Magnus-Verfahren 4. Ordnung im Skript auf Seite 274, um die Mathieu Differentialgleichung für  $\omega = 1$ ,  $\epsilon = 0.25$  zu lösen.

Plote das Ergebnis im Interval  $t \in [0, 100]$ .



## Kapitel VIII

### Exponentielle Integratoren (pp. 293–296 im Skript)

## Exponentielle Integratoren

**Grundidee:** Man betrachtet eine ODE der Form  $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$  mit  $f \in C^1$ .

## Exponentielle Integratoren

**Grundidee:** Man betrachtet eine ODE der Form  $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$  mit  $f \in C^1$ .

Die ODE wird linearisiert durch  $\dot{\mathbf{y}} = J\mathbf{y}$  mit  $J = D\mathbf{f}(\mathbf{y}_0)$ .

Nun kann man die ODE umschreiben  $\dot{\mathbf{y}} = J\mathbf{y} + \mathbf{g}(\mathbf{y})$  wobei  $\mathbf{g}(\mathbf{y}) = \mathbf{f}(\mathbf{y}) - J\mathbf{y}$ .

Es ist dann  $\mathbf{y}(h) = e^{Jh}\mathbf{y}_0 + \int_0^h e^{J(h-z)}\mathbf{g}(\mathbf{y}(z))dz$ .

Es gibt verschiedene Möglichkeiten das obere Integral numerisch zu berechnen.

## Beispiel Exponentielles Euler-Verfahren

Das exponentielle Euler-Verfahren ist gegeben durch

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \phi(h\mathbf{J}_f) h\mathbf{f}(\mathbf{y}_k).$$

## Beispiel Exponentielles Euler-Verfahren

Das **exponentielle Euler-Verfahren** ist gegeben durch

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \phi(h\mathbf{J}_f) h\mathbf{f}(\mathbf{y}_k).$$

Dabei ist  $\mathbf{J}_f := D\mathbf{f}(\mathbf{y}_k)$  und  $\phi(z) = \frac{e^z - 1}{z}$ .

## Beispiel Exponentielles Euler-Verfahren

Das **exponentielle Euler-Verfahren** ist gegeben durch

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \phi(hJ_f) h\mathbf{f}(\mathbf{y}_k).$$

Dabei ist  $J_f := D\mathbf{f}(\mathbf{y}_k)$  und  $\phi(z) = \frac{e^z - 1}{z}$ .

Die Stabilitätsfunktion bekommt man durch die Anwendung des Verfahrens auf die Testgleichung  $\dot{\mathbf{y}} = \lambda\mathbf{y}$ .

## Beispiel Exponentielles Euler-Verfahren

Das **exponentielle Euler-Verfahren** ist gegeben durch

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \phi(hJ_f) h\mathbf{f}(\mathbf{y}_k).$$

Dabei ist  $J_f := D\mathbf{f}(\mathbf{y}_k)$  und  $\phi(z) = \frac{e^z - 1}{z}$ .

Die Stabilitätsfunktion bekommt man durch die Anwendung des Verfahrens auf die Testgleichung  $\dot{\mathbf{y}} = \lambda\mathbf{y}$ .

Damit erhält man  $\mathbf{y}_{k+1} = \mathbf{y}_k + h \frac{\exp(h\lambda) - 1}{h\lambda} \lambda\mathbf{y}_k = \exp(h\lambda)\mathbf{y}_k$ .

## Beispiel Exponentielles Euler-Verfahren

Das **exponentielle Euler-Verfahren** ist gegeben durch

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \phi(hJ_f) h\mathbf{f}(\mathbf{y}_k).$$

Dabei ist  $J_f := D\mathbf{f}(\mathbf{y}_k)$  und  $\phi(z) = \frac{e^z - 1}{z}$ .

Die Stabilitätsfunktion bekommt man durch die Anwendung des Verfahrens auf die Testgleichung  $\dot{\mathbf{y}} = \lambda\mathbf{y}$ .

Damit erhält man  $\mathbf{y}_{k+1} = \mathbf{y}_k + h \frac{\exp(h\lambda) - 1}{h\lambda} \lambda\mathbf{y}_k = \exp(h\lambda)\mathbf{y}_k$ .

Die **Stabilitätsfunktion** ist dann  $S(z) = \exp(z)$  mit  $z := \lambda h$ .

## Beispiel Exponentielles Euler-Verfahren

Nutze das exponentielle Euler-Verfahren um die folgende ODE zu lösen:

$$\dot{\mathbf{y}} = \begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \end{pmatrix} = \begin{pmatrix} -\frac{y_1^2}{y_2} + y_2 \log(y_2) \\ -y_1 \end{pmatrix} \text{ mit } \mathbf{y}(0) = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

# Beispiel exponentielles Euler-Verfahren

```
# Exponentielles Euler-Verfahren
```

```
def expEV(n_steps, t0, tf, y0, f, Df):
```

```
    t_arr, h = linspace(t0, tf, n_steps+1, retstep=True)
```

```
    y = []
```

```
    y.append(y0)
```

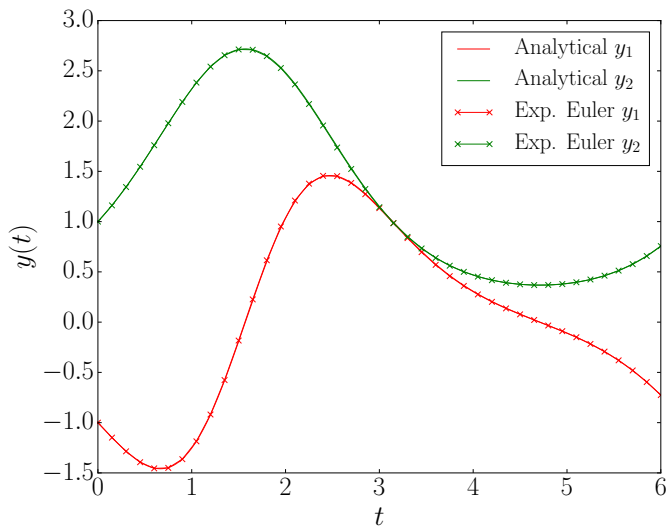
```
    for i in range(n_steps):
```

```
        DF = Df(y[-1])
```

```
        y.append(y[-1] + dot(expm(h*DF) - eye(*DF.shape),  
                             solve(DF, f(y[-1]))))
```

```
    return t_arr, asarray(y)
```

# Beispiel exponentielles Euler-Verfahren



## Kapitel IX

### Splitting-Verfahren (pp. 228–251 im Skript)

# Splitting-Verfahren

Für die numerische Lösung gewisser Differentialgleichung eignen sich Splitting-Verfahren, welche die “rechte Seite” einer gegebenen Differentialgleichung aufteilen und gesondert behandeln.

# Splitting-Verfahren

Für die numerische Lösung gewisser Differentialgleichung eignen sich Splitting-Verfahren, welche die “rechte Seite” einer gegebenen Differentialgleichung aufteilen und gesondert behandeln.

**Lernziele:** Kennen und Anwenden geläufiger Splitting-Verfahren, insbesondere mit Blick auf Energieerhaltung bei physikalischen Problemen.

## Splitting-Verfahren

**Grundidee:** Wir betrachten autonome Anfangswertprobleme der Form  $\dot{\mathbf{y}} = \mathbf{f}_a(\mathbf{y}) + \mathbf{f}_b(\mathbf{y})$  mit  $\mathbf{y}(0) = \mathbf{y}_0$ .

## Splitting-Verfahren

**Grundidee:** Wir betrachten autonome Anfangswertprobleme der Form  $\dot{\mathbf{y}} = \mathbf{f}_a(\mathbf{y}) + \mathbf{f}_b(\mathbf{y})$  mit  $\mathbf{y}(0) = \mathbf{y}_0$ .

Die **Evolutionsoperatoren** für die Funktionen  $\mathbf{f}_a(\mathbf{y})$  und  $\mathbf{f}_b(\mathbf{y})$  sind  $\Phi_a^t$  und  $\Phi_b^t$ .

## Splitting-Verfahren

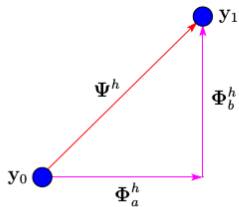
**Grundidee:** Wir betrachten autonome Anfangswertprobleme der Form  $\dot{\mathbf{y}} = \mathbf{f}_a(\mathbf{y}) + \mathbf{f}_b(\mathbf{y})$  mit  $\mathbf{y}(0) = \mathbf{y}_0$ .

Die **Evolutionsoperatoren** für die Funktionen  $\mathbf{f}_a(\mathbf{y})$  und  $\mathbf{f}_b(\mathbf{y})$  sind  $\Phi_a^t$  und  $\Phi_b^t$ .

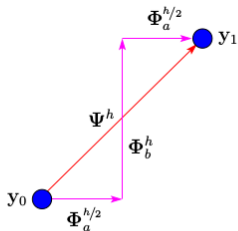
**Lie-Trotter-Splitting** ( $\mathcal{O}(h)$ ):  $\Psi_1^h = \Phi_a^h \circ \Phi_b^h$  oder  $\Psi_1^h = \Phi_b^t \circ \Phi_a^t$ ,

**Strang-Splitting** ( $\mathcal{O}(h^2)$ ):  $\Psi_2^h = \Phi_a^{h/2} \circ \Phi_b^h \circ \Phi_a^{h/2}$ .

Zum Beispiel entspricht das Strang-Splitting der Newton-Gleichungen dem **Störmer-Verlet-Verfahren**, cf. p. 239 im Skript.



Lie-Trotter-Splitting



Strang-Splitting

Aus dem Skript von Seite 287.

# Beispiel Splitting-Verfahren

## Beispiel Splitting-Verfahren

Wir betrachten ein Teilchen der Masse  $m = 1$  im Gravitationsfeld eines Körpers mit Masse  $M = 1$  und Gravitationskonstante  $G = 1$ .

## Beispiel Splitting-Verfahren

Wir betrachten ein Teilchen der Masse  $m = 1$  im Gravitationsfeld eines Körpers mit Masse  $M = 1$  und Gravitationskonstante  $G = 1$ .

Der Hamiltonian ist  $H(\mathbf{q}, \mathbf{p}) = \frac{1}{2m} \|\mathbf{p}\|^2 + U(\mathbf{q})$  mit  $U(\mathbf{q}) = -\frac{GM}{\|\mathbf{q}\|}$ .

## Beispiel Splitting-Verfahren

Wir betrachten ein Teilchen der Masse  $m = 1$  im Gravitationsfeld eines Körpers mit Masse  $M = 1$  und Gravitationskonstante  $G = 1$ .

Der Hamiltonian ist  $H(\mathbf{p}, \mathbf{q}) = \frac{1}{2m} \|\mathbf{p}\|^2 + U(\mathbf{q})$  mit  $U(\mathbf{q}) = -\frac{GM}{\|\mathbf{q}\|}$ .

Die Bewegungsgleichungen sind gegeben durch:

$$\begin{aligned}\dot{\mathbf{q}} &= \nabla_{\mathbf{p}} \cdot H(\mathbf{q}, \mathbf{p}), \\ \dot{\mathbf{p}} &= -\nabla_{\mathbf{q}} \cdot H(\mathbf{q}, \mathbf{p}).\end{aligned}$$

## Beispiel Splitting-Verfahren

Finde die Bewegungsgleichungen und integriere diese mit dem Strang-Splitting Verfahren für  $\mathbf{p}(0) = (0, 1)^T$ ,  $\mathbf{q}(0) = (1, 0)^T$  und  $t \in [0, 50]$ .

# Beispiel Splitting-Verfahren

## Beispiel Splitting-Verfahren

Finde die Bewegungsgleichungen und integriere diese mit dem Strang-Splitting Verfahren für  $\mathbf{p}(0) = (0, 1)^T$ ,  $\mathbf{q}(0) = (1, 0)^T$  und  $t \in [0, 50]$ .

Wir finden  $\dot{\mathbf{q}} = \mathbf{p}$  und  $\dot{\mathbf{p}} = -\frac{\mathbf{q}}{\|\mathbf{q}\|^3}$ .

Der Hamiltonian  $H(\mathbf{q}, \mathbf{p}) = T(\mathbf{p}) + V(\mathbf{q})$  kann in einen kinetischen  $T(\mathbf{p})$  und einen Potentialteil  $V(\mathbf{q})$  aufgeteilt werden.

# Beispiel Splitting-Verfahren

## Beispiel Splitting-Verfahren

Finde die Bewegungsgleichungen und integriere diese mit dem Strang-Splitting Verfahren für  $\mathbf{p}(0) = (0, 1)^T$ ,  $\mathbf{q}(0) = (1, 0)^T$  und  $t \in [0, 50]$ .

Wir finden  $\dot{\mathbf{q}} = \mathbf{p}$  und  $\dot{\mathbf{p}} = -\frac{\mathbf{q}}{\|\mathbf{q}\|^3}$ .

Der Hamiltonian  $H(\mathbf{q}, \mathbf{p}) = T(\mathbf{p}) + V(\mathbf{q})$  kann in einen kinetischen  $T(\mathbf{p})$  und einen Potentialteil  $V(\mathbf{q})$  aufgeteilt werden.

Damit sind die Evolutionsoperatoren gegeben durch:

$$\Phi_T^t \begin{pmatrix} \mathbf{q} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{q} + \mathbf{t} \nabla T(\mathbf{p}) \\ \mathbf{p} \end{pmatrix}$$
$$\Phi_V^t \begin{pmatrix} \mathbf{q} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{q} \\ \mathbf{p} + \mathbf{t} \nabla V(\mathbf{q}) \end{pmatrix}$$

# Beispiel exponentielles Euler-Verfahren

---

`U = lambda r: -G*M / r`

`dUdr = lambda r: G*M / r**2`

`T = lambda p: 0.5*np.sum(p**2, axis=1)`

`V = lambda q: U(np.linalg.norm(q, axis=1))`

`H = lambda q, p: T(p) + V(q)`

`gradT = lambda p: p`

`gradV = lambda q: -1.0*dUdr(np.linalg.norm(q))  
* q/np.linalg.norm(q)`

---

# Beispiel exponentielles Euler-Verfahren

```
def PhiT(y, dt):  
    q, p = y[0,:], y[1,:]   
    y1 = np.empty_like(y)  
    y1[0,:] = q + dt*gradT(p)  
    y1[1,:] = p
```

```
return y1
```

```
def PhiV(y, dt):  
    q, p = y[0,:], y[1,:]   
    y1 = np.empty_like(y)  
    y1[0,:] = q  
    y1[1,:] = p + dt*gradV(q)
```

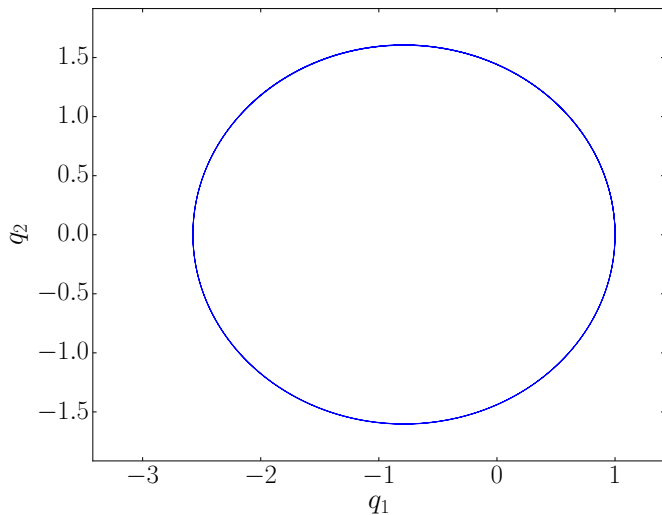
```
return y1
```

```
def strang_splitting_step(Phi_a, Phi_b, y0, dt):
    y_a = Phi_a(y0, 0.5*dt)
    y_ba = Phi_b(y_a, dt)
    y_aba = Phi_a(y_ba, 0.5*dt)
    return y_aba

def integrate(Phi_a, Phi_b, y0, t_end, n_steps):
    t, dt = np.linspace(0, t_end, n_steps+1, retstep=True)
    y = []
    y.append(y0)
    for i in range(n_steps):
        y.append(strang_splitting_step(Phi_a, Phi_b,
                                       y[-1], dt))

    return t, np.asarray(y)
```

# Beispiel Splitting-Verfahren



# Beispiel Splitting-Verfahren

