D-MATH Numerische Methoden FS 2020
Dr. Vasile Gradinaru
Dr. Adrian Montgomery Ruf

Serie 13

Best before: Di. 02.06. / Mi. 03.06, in den Übungsgruppen
Koordinatoren: Adrian Montgomery Ruf, HG G 54.1, `adrian.ruf@sam.math.ethz.ch`
Webpage: `http://metaphor.ethz.ch/x/2020/fs/401-1662-10L/#exercises`

1. *Lineare ODEs mit Krylov-Verfahren*

Gegeben sei das lineare System gewöhnlicher Differentialgleichungen:

$$\begin{aligned}\dot{\underline{y}}(t) &= -i\mathbf{A}\underline{y}(t)\\ \underline{y}(0) &= \underline{y}_0\end{aligned}$$

mit der Hermite-symmetrischen Matrix $\mathbf{A} \in \mathbb{C}^{d\times d}$.

a) Zeigen Sie, dass die $\|\cdot\|_2$-Norm der Lösung durch die Evolution erhalten bleibt:

$$\|\underline{y}(t)\|_2 = \|\underline{y}(0)\|_2\,.$$

Hierbei ist:

$$\|\underline{u}\|_2^2 = \langle \underline{u}, \underline{u}\rangle = \sum_{j=1}^{d} \bar{u}_j u_j\,.$$

b) Verwenden Sie die Diagonalisierung von $\mathbf{A}$, um eine formale Lösung des Differentialgleichungssystems zu finden.

Hinweis: Siehe Beispiel 4.1.3 im Skript.

c) Verwenden Sie ein Krylov-Verfahren um eine numerische Lösung des Differentialgleichungssystems zu finden. Suchen Sie dafür eine Lösung:

$$\begin{aligned}\underline{u}_m(t) &\in \mathcal{K}_m(\mathbf{A}, \underline{y}_0)\\ \underline{u}_m(0) &= \underline{y}_0\,,\end{aligned}$$

so dass das Residuum $\dot{\underline{u}}_m(t) + i\mathbf{A}\underline{u}_m(t)$ orthogonal auf dem Krylov-Raum $\mathcal{K}_m(\mathbf{A}, \underline{y}_0)$ steht.

d) Wir wollen nun unsere Methoden für die Matrizen $\mathbf{A}$

1. `sqrt`
2. `minij`
3. `dvr`

Bitte wenden!

welche im Template `krylov.py` implementiert sind, anwenden. Die Lösung soll mit

$$\underline{y}_0 = \frac{1}{\sqrt{d}}[1, 1, \ldots, 1]^{\mathrm{T}}$$

bis zur Zeit $t = 10^{-2}$ numerisch berechnet werden. Für das Krylov-Verfahren verwenden Sie sowohl das *Arnoldi-* als auch das *Lanczos-*Verfahren. Benutzen Sie Als Referenzlösung das Ergebnis welches `expm` aus `scipy.linalg` liefert. Geben Sie die Rechenzeiten und die Fehler bezüglich der Referenzlösung aus. Welchen Einfluss hat der Parameter m (Dimension des Krylov-Raums) auf die Lösung?

2. *Exponentielles Euler-Verfahren (Prüfungsaufgabe FS14)*

Betrachten Sie das *exponentielle Euler-Verfahren* mit konstanter Schrittweite:

$$\underline{y}_{k+1} = \underline{y}_k + h\varphi(h\mathbf{J_f})f(\underline{y}_k), \quad k = 0, \ldots, N \tag{1}$$

wobei:

$$\mathbf{J_f} := \mathrm{D}\, f(\underline{y}_k), \quad \varphi(z) = \frac{e^z - 1}{z}$$

a) Leiten Sie die Stabilitätsfunktion $S(z)$ von (1) her.

b) Schreiben Sie eine `Python`-Funktion `expEV` die das nichtlineare Anfangswertproblem

$$\dot{\underline{y}} = \begin{bmatrix} \dot{y_1} \\ \dot{y_2} \end{bmatrix} = \begin{bmatrix} -\frac{y_1^2}{y_2} + y_2 \log(y_2) \\ -y_1 \end{bmatrix}, \quad \underline{y}(0) = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

mit dem exponentiellen Eulerverfahren (1) mit konstanter Schrittweite löst.

Hinweis: Verwenden Sie das Template `exp_euler.py`.

Hinweis: Die Aufgabe wird viel leichter wenn Sie beachten, dass $Df \in \mathbb{R}^{2\times 2}$ klein ist. Sie können also hier einfach `expm` verwenden.

c) Bestimmen Sie empirisch die Konvergenzordnung des Verfahrens. Betrachten Sie das Zeitintervall $[0, 6]$ und berechnen Sie den Fehler bezüglich der exakten Lösung:

$$\underline{y}(t) = \begin{bmatrix} -\cos(t)\exp(\sin(t)) \\ \exp(\sin(t)) \end{bmatrix}$$

für verschiedene Anzahl von Zeitschritten $N = 24$, 48, 96, 192, 384.

3. *Stationäre Zustände der Schrödingergleichung*

Wir betrachten die zeitunabhängige oder stationäre Schrödingergleichung:

$$\mathcal{H}\Psi = E\Psi$$

wobei $\Psi(\underline{x})$ die Wellenfunktion, E die Energie und:

$$\mathcal{H} := -\frac{1}{2}\Delta + V(\underline{x})$$

Siehe nächstes Blatt!

der Hamilton-Operator ist. Wir wollen nun für ein gegebenes Potential $V(\underline{x})$ den Grundzustand $\Psi_0(\underline{x})$ sowie ein paar weitere Zustände $\Psi_n(\underline{x})$ niedriger Energie finden. Man diskretisiert die Gleichung indem man auf dem Intervall $[a, b]$ genau N Punkte:

$$a = x_0 < \ldots < x_i < \ldots < x_{N-1} = b$$

gleichmässig verteilt und für den Laplace-Operator Δ eine Approximation mit finiten Differenzen verwendet. Dann kann die Gleichung als lineares Eigenwertproblem:

$$\mathbf{H}\underline{\psi} = E\underline{\psi}$$

geschrieben werden wobei die Wellenfunktion zu einem Vektor von Punktauswertungen:

$$\underline{\psi} = [\ldots, \psi_i, \ldots]^{\mathrm{T}} = [\ldots, \psi(x_i), \ldots]^{\mathrm{T}} \in \mathbb{R}^N$$

und der diskretisierte Hamilton-Operator $\mathbf{H} \in \mathbb{R}^{N\times N}$ zu einer Matrix wird.

Betrachten Sie als erstes, simples Beispiel den *harmonischen Oszillator* gegeben durch das Potential $V(x) = \frac{1}{2}x^2$ auf dem Intervall $x \in [-10, 10]$.

a) Leiten Sie die Matrix $\mathbf{H}$ für diesen Fall explizit her und formulieren Sie danach das diskrete Eigenwertproblem. Benutzen Sie zentrale finite Differenzen im Innern des Intervalls sowie die passende asymmetrische Form am Rande.

Hinweis: Finite Differenzen:

- Vorwärts: $f''(x_i) \approx \frac{f(x_i)-2f(x_{i+1})+f(x_{i+2})}{h^2} + \mathcal{O}(h)$
- Zentral: $f''(x_i) \approx \frac{f(x_{i-1})-2f(x_i)+f(x_{i+1})}{h^2} + \mathcal{O}(h^2)$
- Rückwärts: $f''(x_i) \approx \frac{f(x_{i-2})-2f(x_{i-1})+f(x_i)}{h^2} + \mathcal{O}(h)$

Bemerkung: Verwendet man die Randbedingung $\Psi(\text{rand}) = 0$, ist $\mathbf{H}$ symmetrisch.

b) Berechnen Sie die Eigenvektoren $\underline{\psi_n}$ und Eigenwerte E_n für $N = 32, 64, \ldots, 1024$ Punkte im Intervall $[-10, 10]$ mit `eig`.

Hinweis: Sortieren Sie die Eigenwerte und Eigenvektoren. Beispielsweise mit einer geschickten Anwendung von `argsort`.

c) Plotten Sie, für alle N, die ersten $0 \leq n < 32$ Energien $E_n^{(N)}$ gegen n. Berechnen und Plotten Sie ebenso den Fehler $|E_n^{(N)} - E_n^{\text{exact}}|$ der gefundenen Energien.

Hinweis: Die exakten Energien sind: $E_n^{\text{exact}} = n + \frac{1}{2}$.

d) Sei $N = 1024$ fix. Plotten Sie die ersten $0 \leq n \leq 6$ Eigenfunktionen $\psi_n(x)$ gegen x. Berechnen und Plotten Sie den Fehler $\||\psi_n^N| - |\psi_n^{\text{exact}}|\|_2$.

Hinweis: Die exakte Lösung ist: $\psi_n^{\text{exact}}(x) = \pi^{-\frac{1}{4}} \frac{1}{\sqrt{2^n n!}} H_n(x) e^{-\frac{1}{2}x^2}$ wobei $H_n(x)$ das (physicists') Hermite Polynom ist.

Als Nächstes betrachten wir das *Morse-Potential*:

$$V(x) := V_0 \left(e^{-2\beta x} - 2e^{-\beta x}\right)$$

mit den Parametern $V_0 = 16$ und $\beta = 1$. Dieses wichtige Potential ist auf der einen Seite asymptotisch flach und ermöglicht so die Simulation der Aufspaltung zweiatomiger Moleküle.

Bitte wenden!

e) Berechnen Sie die Eigenwerte und Eigenvektoren mittels `eig`. Verwenden Sie $N = 256$ Punkte im Intervall $[-2, 8]$. Plotten Sie die ersten vier Eigenfunktionen $\psi_0(x)$ bis $\psi_3(x)$. (Achtung, $\psi_{n\geq 6}(x)$ existieren aus quantenmechanischen Gründen nicht.)

f) Implementieren Sie ein Arnoldi-Verfahren um die kleinsten Eigenwerte einer Matrix zu approximieren. Testen Sie das Verfahren am Morse-Potential mit einem Krylov-Raum der Grösse $k = 150$ Iterationen.

Bemerkung: Für eine echte Anwendung soll man `eigs`, `eigsh` verwenden. Dies ist der `scipy` Wrapper um die `Arpack`[1] Library, die sehr effiziente und robuste Krylov-Verfahren implementiert. Für die Beispiele hier sind weniger als 20 Iterationen notwendig.

Zum Schluss wollen wir noch ein zweidimensionales Problem berechnen. Gegeben sei das *Henon–Heiles*-Potential:

$$V(x, y) := \frac{a}{2}(x^2 + y^2) + b\left(x^2 y - \frac{y^3}{3}\right)$$

mit $a = 2.0$ und $b = 0.4$. Wir verwenden ein zweidimensionales Gitter auf $[-3, 3] \times [-3, 3]$ mit je $N = 32$ Punkten in jede Richtung. Ordnen Sie die Gitterpunkte $\underline{x}_{i,j} := (x_i, y_j)$ in einen Spaltenvektor der Länge N^2. Die Wellenfunktion ist dann wie folgt diskretisiert:

$$\begin{aligned}\underline{\psi} &= [\ldots, \psi_{i,j}, \ldots]^{\mathrm{T}} \\ &= [\psi(x_0, y_0), \psi(x_1, y_0), \ldots, \psi(x_{N-1}, y_0), \psi(x_0, y_1), \ldots, \psi(x_{N-1}, y_1), \ldots, \psi(x_{N-1}, y_{N-1})]^{\mathrm{T}}\end{aligned}$$

Das benötigte Gitter kann mit `meshgrid` erzeugt werden.

g) Diskretisieren Sie den Hamilton-Operator $\mathcal{H}$ und plotten Sie die linke obere 50×50 Ecke von $\mathbf{H}$ mit Hilfe von `matshow`.

Hinweis: Es gilt $\Delta f = f_{xx} + f_{yy}$ und somit:

$$\Delta f(\underline{x}_{i,j}) \approx \frac{f(\underline{x}_{i-1,j}) - 2f(\underline{x}_{i,j}) + f(\underline{x}_{i+1,j})}{h^2} + \frac{f(\underline{x}_{i,j-1}) - 2f(\underline{x}_{i,j}) + f(\underline{x}_{i,j+1})}{h^2}$$

h) Berechnen Sie die Eigenwerte und Eigenvektoren mittels `eig` und plotten Sie die ersten sechs Eigenfunktionen $\psi_0(x, y)$ bis $\psi_5(x, y)$.[2]

i) Berechnen Sie die Eigenwerte und Eigenvektoren mit dem Arnoldi-Verfahren und einem Krylov-Raum der Grösse $k = 220$ Iterationen. Plotten Sie wiederum die ersten sechs Eigenfunktionen $\psi_0(x, y)$ bis $\psi_5(x, y)$.

4. *Vibration einer Saite*

Die Vibration einer Saite, die an beiden Enden fixiert ist und unter gleichmässiger Spannung T steht, wird durch folgende Differentialgleichung beschrieben:

$$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{T}{m(x)} \frac{\partial^2 u(x,t)}{\partial x^2},$$

[1] http://www.caam.rice.edu/software/ARPACK/

[2] Ein Plot der Eigenfunktionen, berechnet mit einer wesentlich genaueren Methode, ist hier zu finden: http://raoulbq.github.io/WaveBlocksND/_images/henon_heiles_eigenstates.png
Die Lösungen dieser Aufgabe sollten ähnlich aber aufgrund anderer Parameter nicht identisch aussehen.

Siehe nächstes Blatt!

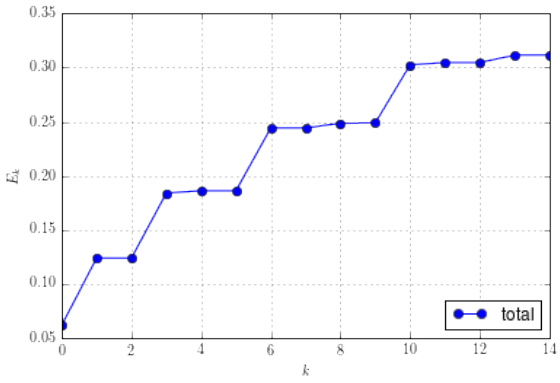

Abbildung 1 – Energielevels des Henon–Heiles-Potential.

wobei $m(x)$ die Masse ist. Die Methode der Separation der Variablen liefert:

$$\frac{T}{m(x)}\frac{d^2\psi(x)}{dx^2} + \omega^2\psi(x) = 0\ ,$$

wobei ω durch die Randbedingungen gegeben ist. Für die Approximation der Ableitungen wollen wir finite Differenzen verwenden. Wir unterteilen das Intervall in $N = 513$ Stücke der Länge h indem wir $N + 1$ Punkte gleichmässig verteilen:

$$0 = x_0 < \ldots < x_i < \ldots < x_N = L$$

Dann approximieren wir die exakte Lösung in diesen Punkten mit $\psi_i \approx \psi(x_i)$.

Die Saite sei an den beiden Endpunkten $x = 0$ und $x = L$ fest eingespannt. Die Spannung T und die Masse $m(x)$ sind hier fix auf 1 gesetzt.

a) Stellen Sie die Matrix $\mathbf{A}$ *effizient* auf und lösen Sie das Eigenwertproblem:

$$\mathbf{A}\underline{\nu_n} = \lambda_n\underline{\nu_n}$$

per `eigh` aus `scipy.linalg`. Berechnen Sie sowohl die Eigenwerte λ_n als auch die Eigenvektoren $\underline{\nu_n}$.

b) Warum verwenden wir besser `eigh` als `eig`? Beide Funktionen sind in `scipy.linalg` zu finden. Welche Funktionen aus diesem Modul könnten hier auch noch nützlich sein?

c) Stellen Sie $\mathbf{A}$ *effizient* als dünnbesetzte Matrix auf.

Hinweis: Nutzen Sie dazu die Funktion `diags` aus `scipy.sparse`.

d) Berechnen Sie die 50 kleinsten Eigenwerte λ_n sowie die dazugehörigen Eigenvektoren $\underline{\nu_n}$ von $\mathbf{A}$. Benutzen Sie dafür die Funktion `eigsh` aus `scipy.sparse.linalg`

e) Plotten Sie die ersten 10 Eigenschwingungen $\psi_n(x)$.

f) Plotten Sie die Energien E_n der ersten 20 Eigenschwingungen gegen n.

g) Wiederholen Sie die Aufgabe (ohne den Teil für dünnbesetzte Matrizen) für eine inhomogene Massenverteilung:

$$m(x) := \frac{1}{2}\left(m_1(L - x) + m_2 x\right)$$

mit $m_1 = 1 - \delta m$, $m_2 = 1 + \delta m$ und $\delta m = 0.99$. Welche Routinen zur Berechnung der Eigenwerte dürfen wir in diesem Fall verwenden?

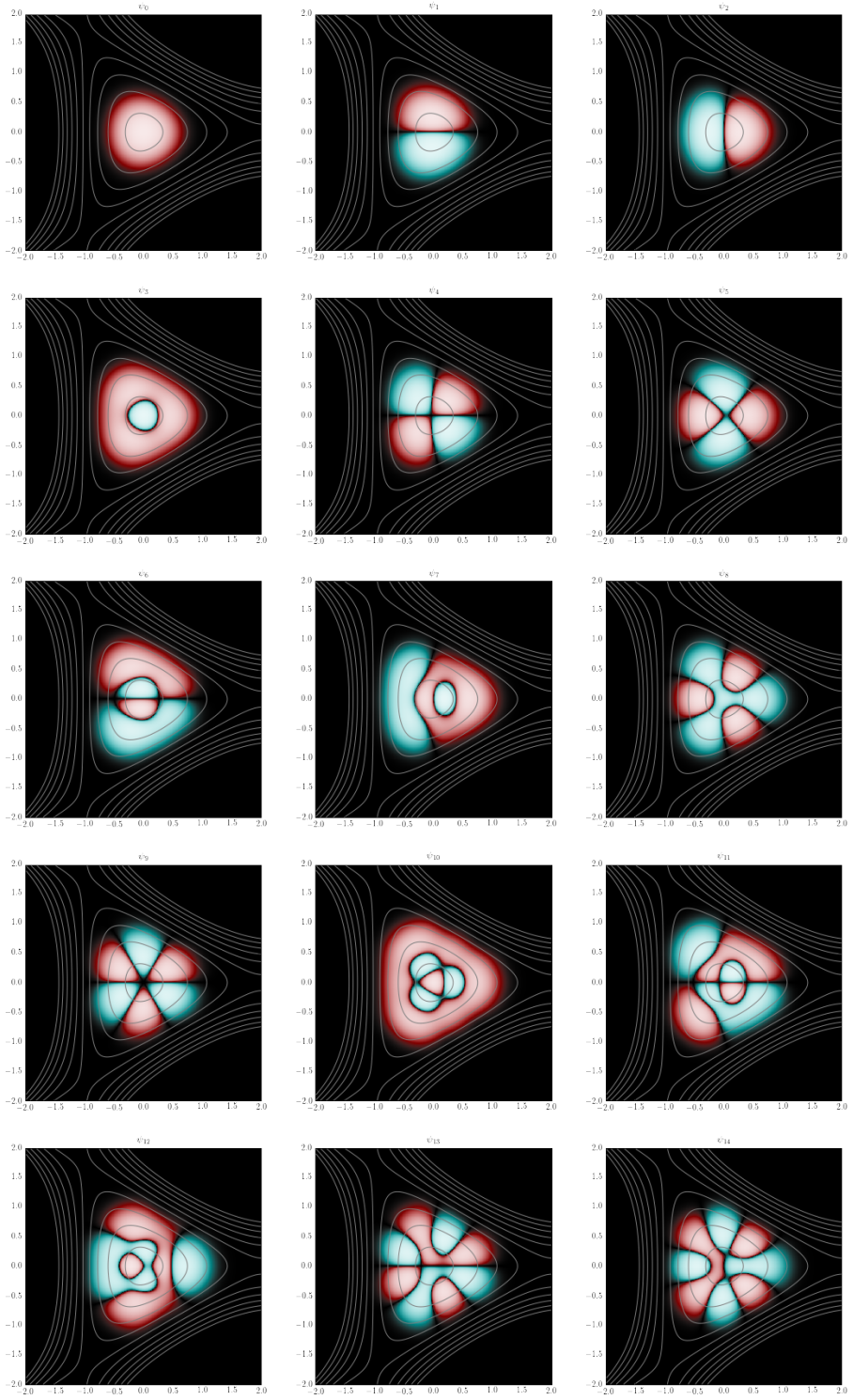

Abbildung 2 – Eigenzustände des Henon–Heiles-Potential.