Abteilung für
Angewandte Mathematik

Sommersemester 2020
Prof. Dr. Sören Bartels, Pascal Weyer

Praktikum zur Numerik Teil 2

Projektblatt 1 – 11. Mai 2020

Abgabe: per E-Mail an die Tutoren bis Montag, den 25. Mai 2020, 9:00 Uhr

---

Homepage zur Vorlesung:

`https://aam.uni-freiburg.de/agba/lehre/ss20/num/index.html`

Hinweis: Bitte erläutern Sie bei allen Aufgaben Ihre Beobachtungen in einigen Sätzen.

Projekt 1 (4 Punkte)**.** Implementieren Sie das Neville-Schema in nichtrekursiver Form und verwenden Sie es, um das Interpolationspolynom der Funktion $f(x) = (1 + 25x^2)^{-1}$ bezüglich äquidistanter Stützstellen $-1 = x_0 < x_1 < \cdots < x_n = 1$ sowie Tschebyscheff-Knoten $-1 < t_0 < t_1 < \cdots < t_n < 1$ an den Punkten $x_a = 0{,}4$, $x_b = 0{,}8$ und $x_c = 0{,}99$ für $n = 1, 2, 4, 8, 16, 32$ auszuwerten.

Projekt 2 (4 Punkte)**.** Implementieren Sie das Jacobi-Verfahren zur Bestimmung von Eigenwerten mit dem Abbruchkriterium $\mathcal{N}(A_k) \leq 10^{-4}$ in `Matlab` und testen Sie es für die Matrix $A \in \mathbb{R}^{n\times n}$ definiert durch

$$a_{ij} = \sin(|i - j|\pi/n) - 2\delta_{ij}$$

für $i, j = 1, 2, \ldots, n$ mit $n = 2, 4, 8, 16$. Modifizieren Sie das Programm, um eine Implementation des zyklischen Jacobi-Verfahrens zu erhalten, das heißt auf die Suche des größten Eintrags wird verzichtet und alle Einträge werden sukzessive behandelt. Betrachten Sie die Anzahl der benötigten Iterationsschritte in Abhängigkeit von n. Beobachten Sie grafisch die Größe der Einträge der Iterierten, beispielsweise mit Hilfe der `Matlab`-Kommandos

```
imagesc(Ak)
title(sprintf('n=%d, N(A)=%f', steps, NA))
colorbar
pause(0.05)
```

an passender Stelle in Ihrer Schleife für die Matrix im aktuellen Schritt `Ak`.
Bitte verwenden Sie in Def. 8.2 in »Numerik 3x9« die Fallunterscheidung auf der linken Seite zur Implementierung der Givens-Rotation. In der dort angegebenen Matrix-Darstellung müssen die Vorzeichen von s vertauscht werden. Weiterhin benutzen Sie bitte $s = +\operatorname{sign}(a_{pq})\sqrt{(1-D)/2}$ in Lemma 8.3.

Projekt 3 (2 Bonus-Punkte)**.** Wir betrachten die numerische Bestimmung der Eulerschen Zahl e, die sich durch die Grenzwerte

$$e = \lim_{n\to\infty} (1 + 1/n)^n, \quad e = \lim_{n\to\infty} \sum_{k=0}^{n} \frac{1}{k!}$$

charakterisieren lässt. Verwenden Sie ausschließlich arithmetische Grundoperationen und endliche Approximationen der obigen Grenzwerte mit $n = 10^j$, $j = 1, 2, \ldots, 10$, um e zu approximieren. Bestimmen Sie jeweils die Approximationsfehler mit Hilfe der Annäherung $e \approx 2{,}718281828459045$ und stellen Sie diesen mit 15 Nachkommastellen in einer Tabelle dar.

Praktikum zur Numerik Teil 2

Projektblatt 2 – 22. Mai 2020

Abgabe: per E-Mail an die Tutoren bis Montag, den 8. Juni 2020, 9:00 Uhr

---

Homepage zur Vorlesung:

`https://aam.uni-freiburg.de/agba/lehre/ss20/num/index.html`

Hinweis: Bitte erläutern Sie bei allen Aufgaben Ihre Beobachtungen in einigen Sätzen.

Projekt 4 (4 Punkte)**.** Schreiben Sie ein `Matlab`-Programm zur Bestimmung der Koeffizienten eines Interpolationspolynoms bezüglich der Newton-Basis für gegebene Stützstellen $x_0 < x_1 < \cdots < x_n$ und zugehörige -werte $y_0, \dots, y_n$. Testen Sie Ihr Programm für die Funktionen $f(x) = \sin(\pi x)$, $g(x) = (1 + 25x^2)^{-1}$ und $h(x) = |x|$ im Intervall $[-1, 1]$ bei Verwendung von äquidistanten Stützstellen und Tschebyscheff-Knoten. Werten Sie die Interpolationspolynome an den Punkten $z_j = -1 + 2j/100$, $j = 0, 1, \dots, 100$ mit dem Horner-Schema aus und plotten Sie damit die Interpolationspolynome für $n = 1, 2, 4, 8$. Es bietet sich hier an, für jede Funktion eine `figure` zu erstellen, die vier Koordinatensysteme enthält – eines je Wert von n. Sie können dies durch den `subplot(4,1,i)`-Befehl erreichen, wobei `i` das aktuelle Koordinatensystem angibt.

Projekt 5 (2+2 Punkte)**.** (i) Der `Matlab`-Befehl `plot(X,Y)` stellt einen durch die Vektoren X und Y definierten Polygonzug grafisch dar. Ist $X = [x_0, x_1, \dots, x_n]^\top$ und $Y = [f(x_0), f(x_1), \dots, f(x_n)]^\top$, so wird eine stetige, stückweise lineare Interpolation der Funktion f dargestellt. Illustrieren Sie grafisch die stückweise lineare Approximation der Funktion $f(x) = x^{1/2}$ auf dem Intervall $[0, 1]$ mit den Gitterpunkten

$$\text{(a)} \quad x_i = i/n, \qquad \text{(b)} \quad x_i = (i/n)^4$$

für $i = 0, 1, \dots, n$ und $n = 2, 4, 8, 16$, indem Sie diese mit der Darstellung von f auf einem sehr feinen Gitter vergleichen.
(ii) Schreiben Sie eine Routine zur Berechnung eines interpolierenden kubischen Splines mit natürlichen Randbedingungen. Testen Sie die Routine mit den Partitionierungen aus (i) für die Funktion $f(x) = \sin(2\pi x)$.

Praktikum zur Numerik Teil 2

Projektblatt 3 – 8. Juni 2020

Abgabe: per E-Mail an die Tutoren bis Montag, den 22. Juni 2020, 9:00 Uhr

---

Homepage: `aam.uni-freiburg.de/agba/lehre/ss20/num`

Hinweis: Bitte erläutern Sie bei allen Aufgaben Ihre Beobachtungen in einigen Sätzen.

Wir benutzen folgende Funktionen $f_r : [0, 2\pi] \to \mathbb{C}$ in den ersten beiden Projekten:

$$f_1(x) = \sin(5x) + 0.5\mathrm{i}\cos(x) \quad f_2(x) = \begin{cases} 1, & x \in [\pi - 1/4, \pi + 1/4], \\ 0, & x \notin [\pi - 1/4, \pi + 1/4], \end{cases}$$

$$f_3(x) = \begin{cases} 1, & x \in [0, \pi), \\ -1, & x \in [\pi, 2\pi], \end{cases} \quad f_4(x) = \begin{cases} x, & x \in [0, \pi), \\ 2\pi - x, & x \in [\pi, 2\pi]. \end{cases}$$

Projekt 6 (4 Punkte)**.** Implementieren Sie die schnelle komplexe Fourier-Synthese als rekursive Funktion `my_ifft` und benutzen Sie diese sowie Bemerkung 13.2(i) in »Numerik 3x9«, um eine Funktion `my_fft` für die komplexe Fourier-Transformation zu definieren. Verwenden Sie Ihre Routine, um die Fourier-Transformation der Vektoren $y \in \mathbb{C}^n$ definiert durch $y_j = f_r(2\pi j/n)$, $j = 0, 1, \dots, n-1$, $r = 1, 2, 3, 4$, mit $n = 2^s$, $s = 1, 2, \dots, 5$, zu berechnen. Stellen Sie die zugehörigen komplexen trigonometrischen Polynome grafisch dar. Nutzen Sie bei der Erstellung Ihres Programms die `Matlab`-Realisierung komplexer Zahlen.

Projekt 7 (4 Punkte)**.** Verwenden Sie Ihre im vorigen Projekt erhaltene Routine `my_fft` oder die `Matlab`-Routine `fft`, um für $n = 2^s$, $s = 1, 2, \dots, 5$, komplexe Koeffizienten $(\beta_k)_{k=0,1,\dots,n-1}$ und $(\delta_k)_{k=0,1,\dots,n-1}$ zu berechnen, sodass für die Funktionen

$$p(x) = \sum_{k=0}^{n-1} \beta_k e^{\mathrm{i}kx}, \quad q(x) = \sum_{k=-n/2}^{n/2-1} \delta_{k+n/2} e^{\mathrm{i}kx}$$

die Interpolationseigenschaft $p(x_j) = f_r(x_j)$ beziehungsweise $q(x_j) = f_r(x_j)$ für $j = 0, 1, \dots, n-1$, $r = 1, 2, 3, 4$ und mit $x_j = 2\pi j/n$ erfüllt ist. Hierbei muss das Ergebnis von `fft` noch durch die Länge des Eingabevektors dividiert werden. Plotten Sie jeweils den Real- und Imaginärteil der Funktionen p und q. Stellen Sie auch das Frequenzspektrum $|\beta_k|$ bzw. $|\delta_k|$ als Funktion von k graphisch dar.

Projekt 8 (3 Bonus-Punkte)**.** Modifizieren Sie Ihr Programm zu Berechnung interpolierender kubischer Splines aus Projekt 5(ii) dahingehend, dass es periodische Randbedingungen anwendet. Interpolieren Sie damit die folgende Funktion $f : [0, 1] \to \mathbb{R}^3$:

$$f(x) = \begin{bmatrix} (2 + \cos(2q\pi x))\cos(2p\pi x) \\ (2 + \cos(2q\pi x))\sin(2p\pi x) \\ \sin(2q\pi x) \end{bmatrix}$$

für $p = 1$ und eine natürliche Zahl $q > 2$ Ihrer Wahl und $2^s q + 1$ äquidistante Stützstellen für $s = 0, 1, 2, 3$. Wenden Sie Ihre Interpolations-Routine dafür auf alle Komponenten separat an und stellen Sie das Ergebnis graphisch mithilfe der Funktion `plot3` dar. Probieren Sie gerne auch andere natürliche Zahlen p und q aus. Sie erhalten dabei sogenannte »Torus-Knoten«.

Praktikum zur Numerik Teil 2

Projektblatt 4 – 22. Juni 2020

Abgabe: per E-Mail an die Tutoren bis Montag, den 6. Juli 2020, 9:00 Uhr

---

Hinweis: Bitte erläutern Sie bei allen Aufgaben Ihre Beobachtungen in einigen Sätzen.

Projekt 9 (4 Punkte)**.** Verwenden Sie die summierten Trapez- und Simpson-Regeln sowie eine summierte Gaußsche 3-Punkt-Quadraturformel, um die Integrale im Intervall $[0, 1]$ der Funktionen

$$f(x) = \sin(\pi x)e^x, \quad g(x) = x^{1/3}$$

mit Schrittweiten $h = 2^{-\ell}$, $\ell = 1, 2, \ldots, 10$, zu approximieren. Berechnen Sie jeweils den Fehler e_h und bestimmen Sie eine experimentelle Konvergenzrate γ aus dem Ansatz $e_h \approx c_1 h^\gamma$ und der daraus folgenden Formel

$$\gamma \approx \frac{\log(e_h/e_H)}{\log(h/H)}$$

für zwei aufeinanderfolgende Schrittweiten $h, H > 0$. Vergleichen Sie die experimentellen Konvergenzraten mit den theoretischen Konvergenzraten der Verfahren und kommentieren Sie Ihre Ergebnisse. Stellen Sie die Paare (h, e_h) für die verschiedenen Quadraturformeln vergleichend als Polygonzüge grafisch in logarithmischer Achsenskalierung mit Hilfe des `Matlab`-Befehls `loglog` dar.

Projekt 10 (2+2 Punkte)**.** (i) Aus der Taylor-Formel ergibt sich, dass die Quotienten

$$d_h^+ f(x) = \frac{f(x+h) - f(x)}{h}, \quad \hat{d}_h f(x) = \frac{f(x+h) - f(x-h)}{2h}$$

für eine gegebene Schrittweite $h > 0$ Approximationen von $f'(x)$ mit der Fehlerordnung $\mathcal{O}(h)$ beziehungsweise $\mathcal{O}(h^2)$ definieren. Überprüfen Sie diese Eigenschaft experimentell am Beispiel $f(x) = \tan(x)$ für $x = 1/2$ mit den Schrittweiten $h = 2^{-\ell}$, $\ell = 1, 2, \ldots, 30$.
(ii) Konstruieren Sie (ohne Beweis) durch Extrapolation einen Quotienten $\hat{d}_h^* f(x)$, der die Ableitung $f'(x)$ bis auf einen Fehler der Ordnung $\mathcal{O}(h^4)$ approximiert. Überprüfen Sie das Ergebnis anhand des Tangens wie in Teil (i). Welche Vor- und Nachteile besitzt diese Approximation im Vergleich zu derjenigen in Teil (i)?
Orientieren Sie sich bei der Konstruktion bitte an derjenigen von $T^*(h)$ in Kapitel 14.5 im Buch »Numerik 3x9«. Die dort beschriebenen Rechnungen sind nicht nur für Quadraturen gültig, sondern auch für andere numerische Approximationen wie den hiesigen Differenzenquotienten.

Projekt 11 (3 Bonus-Punkte)**.** In dieser Aufgabe wollen wir mithilfe der (schnellen) Fourier-Transformation drei Instrumente stimmen – Cello, Querflöte und Oboe. Dafür sollen die gespielten Frequenzen mithilfe der FFT berechnet und dann harmonisiert werden. Den Erfolg Ihrer Arbeit können Sie dabei am Ende selbst akustisch beurteilen und gleichzeitig sehen, warum diese Instrumente als verschieden klingend empfunden werden.
Laden Sie bitte die Datei
`aam.uni-freiburg.de/agba/lehre/ss20/num/materials/p11.zip`
herunter und entpacken Sie diese in Ihren `Matlab`-Arbeitsordner. Die Zugangsdaten sind identisch zu denen der übrigen Vorlesungsmaterialien. Bearbeiten Sie bitte die dort enthaltene `.m`-Datei entsprechend den im Quellcode vermerkten Arbeitsaufträgen (markiert mit `%TODO`).

Abteilung für
Angewandte Mathematik

Sommersemester 2020
Prof. Dr. Sören Bartels, Pascal Weyer

Praktikum zur Numerik Teil 2

Projektblatt 5 – 6. Juli 2020

Abgabe: per E-Mail an die Tutoren bis Montag, den 20. Juli 2020, 9:00 Uhr

---

Hinweis: Bitte erläutern Sie bei allen Aufgaben Ihre Beobachtungen in einigen Sätzen.

Projekt 12 (4+4 Punkte + 4 Bonus-Punkte)**.** Wir wollen die Nullstellen der Funktion

$$f(x) = \exp(x) + x^2 - 2$$

mithilfe dreier numerischer Verfahren bestimmen. Vergleichen Sie bitte am Ende die Ergebnisse der von Ihnen implementierten Verfahren untereinander.

(i) Implementieren Sie das Sekanten-Verfahren zur Nullstellensuche und testen Sie es mit der Funktion f, den Start-Intervallen $[-2, 0]$ und $[0, 2]$ sowie dem Abbruchkriterium $|x_{k+1} - x_k| \leq 10^{-12}$.

(ii) Implementieren Sie das Newton-Verfahren und testen Sie es mit dem Startwert $x_0 \in \{-1, 0, 1\}$ sowie dem Abbruchkriterium $|x_{k+1} - x_k| \leq 10^{-12}$. Beenden Sie das Newton-Verfahren bei Nichterreichen des Abbruchkriteriums mit 100 Iterationen.

(iii) Realisieren Sie die Nullstellenbestimmung von f durch ein Abstiegsverfahren für die Funktion $g(x) = |f(x)|^2$. Wählen Sie für das Armijo-Kriterium zum Beispiel $\sigma = 0.5$. Testen Sie erneut die Startwerte $x_0 \in \{-1, 0, 1\}$.

Projekt 13 (4 Bonus-Punkte)**.** Für eine holomorphe Funktion $f : \mathbb{C} \to \mathbb{C}$ mit Nullstellen $z_1, z_2, \ldots, z_n \in \mathbb{C}$ kann die komplexe Ebene in Einzugsbereiche $E_j \subset \mathbb{C}$, die für $j = 1, 2, \ldots, n$, durch

$$E_j = \{z \in \mathbb{C} : \text{Newton-Verfahren mit Startwert } z \text{ konvergiert gegen } z_j\}$$

definiert sind, sowie die Restmenge $X = \mathbb{C} \setminus \cup_{j=1}^n E_j$, partitioniert werden. Betrachten Sie die Funktion $f(z) = z^3 - 1$ und verwenden Sie als Startwerte Gitterpunkte $z_\ell = x_\ell + \mathrm{i} y_\ell$ im Bereich $[-1, 1]^2 \subset \mathbb{C}^2$, die im Abstand $h = 1/200$ angeordnet sind.

Markieren Sie die Punkte unterschiedlich entsprechend der Zugehörigkeit zum Einzugsbereich einer Nullstelle und stellen Sie diese grafisch dar. Verwenden Sie dazu die Matlab-Befehle

```
[X,Y]=meshgrid(-1:h:1,-1:h:1);
scatter(X(:),Y(:),5,C(:));
```

mit einer geeignet definierten Matrix `C`.

Für ein anderes Polynom ergibt sich dabei nebenstehendes Bild. Die Nullstellen welcher Funktion wurden hier approximiert?