

Praktische Übungen zu Numerik I

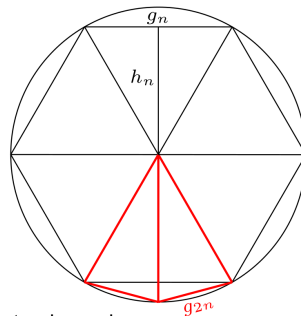
Projekt 7 – 22.01.2024

Abgabe: per E-Mail bis Freitag, den 02.02.2024, 10:00 Uhr

Homepage zur Vorlesung:

<https://aam.uni-freiburg.de/agru/lehre/ws23/num/index.html>

Projekt 1 (6 Punkte). Die Kreiszahl π ist bekanntlich der Quotient von Umfang und Durchmesser eines Kreises. Ist U der Umfang des Einheitskreises, so gilt also $\pi = \frac{U}{2}$. Wir wollen nun Näherungen des Umfangs U mit Hilfe von n gleichen dem Kreis einbeschriebenen Dreiecken mit Grundseite g_n bestimmen, d.h. wir approximieren $U \approx ng_n$. In jedem Schritt verdoppeln wir dabei die Anzahl der Dreiecke (siehe Abbildung unten mit $n = 6$, in der die Verdopplung durch die roten Dreiecke angedeutet ist).



Wir beobachten sowohl $g_6 = 1$ als auch

$$\text{(Pythagoras)} \quad 1 = h_n^2 + \left(\frac{g_n}{2}\right)^2 \quad \text{und} \quad g_{2n}^2 = (1 - h_n)^2 + \left(\frac{g_n}{2}\right)^2.$$

Aus den beiden Gleichungen folgt

$$g_{2n} = \sqrt{2 - \sqrt{4 - g_n^2}}$$

- (1) Implementieren Sie die Rekursion aus der letzten Gleichung mit Startwert $g_6 = 1$ und berechnen Sie damit Approximationen an $\pi_n = \frac{ng_n}{2}$ an die Kreiszahl π . Berechnen Sie in jedem Schritt den Fehler $|\pi_n - \pi|$ und geben Sie diesen in der Konsole aus (oder plotten Sie diese), wobei Sie als Referenzwert für π die Konstanten von Matlab/NumPy verwenden sollen. Was beobachten Sie?
- (2) Leiten Sie nun anstelle der obigen Rekursion eine stabilere Rekursionsvorschrift zur Berechnung der Grundseiten g_n her. Implementieren Sie auch diese. Berechnen Sie auch hier den Fehler und zeigen Sie so, dass Ihr Algorithmus für $n \rightarrow \infty$ immer bessere Näherungen für π berechnet.

Projekt 2 (6 Punkte). Implementieren Sie das Jacobi-Verfahren mit dem Abbruchkriterium $\mathcal{N}(A_k) \leq 10^{-4}$ 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 $i, j = 1, 2, \dots, n$ mit $n = 2, 4, 8, 16$. Modifizieren Sie das Programm, um eine Implementierung des zyklischen Jacobi-Verfahrens zu erhalten, das heißt auf die Suche des größten Eintrags wird verzichtet und alle Einträge werden sukzessive behandelt. Beobachten Sie grafisch die Größe der Einträge der Iterierten mit Hilfe der Matlab-Kommandos `meshgrid`, `surf` und `view(-270, 90)` bzw. den Python-Kommandos `np.meshgrid`, `plot_surface` (`matplotlib`). Betrachten Sie die Anzahl der benötigten Iterationsschritte in Abhängigkeit von n .

Achtung: Im Buch befindet sich bei der Definition der Givens Rotation ein Fehler. Es muss $(G_{pq})_{pq} = s$ und $(G_{pq})_{qp} = -s$ lauten.