

Serie 1

1. *Programming for fun: Conways Spiel des Lebens*¹

Betrachten Sie ein 8×8 Gitter von quadratischen Zellen in einem periodischen Gebiet. Jede Zelle hat zwei mögliche Zustände - *lebendig* oder *tod* - und interagiert nur mit ihren acht direkten Nachbarzellen.

Ausgehend von einer Anfangskonfiguration der Zell-Zustände entwickelt sich das System in jedem Zeitschritt nach folgenden Regeln:

- a) Eine lebende Zelle mit weniger als zwei lebenden Nachbarn stirbt an Einsamkeit.
- b) Eine lebende Zelle mit zwei oder drei lebenden Nachbarn bleibt am Leben.
- c) Eine lebende Zelle mit mehr als drei lebenden Nachbarn stirbt an Überbevölkerung.
- d) Eine tote Zelle mit genau drei lebenden Nachbarn wird neu geboren.

Implementieren Sie diesen Evolutionsalgorithmus im MATLAB Template `conwayGameOfLife.m`. Die Anfangskonfiguration und eine Plotting Funktion `plotBoard` sind gegeben.

2. In dieser Aufgabe leiten wir eine der populärsten Quadraturregeln her: die Simpson-Regel

$$S[f] = \sum_{j=0}^2 \omega_j f(x_j) \approx \int_a^b f(x) dx = I[f].$$

Die Simpson-Regel verwendet als Quadratur Stützstellen/Knoten

$$x_0 = a, \quad x_1 = \frac{a+b}{2}, \quad x_2 = b.$$

- a) Berechnen Sie die Lagrange-Polynome $L_0^2(x)$, $L_1^2(x)$ und $L_2^2(x)$ passend zu den Knoten x_0 , x_1 und x_2 .
- b) Berechnen Sie mit a) die Quadratur Gewichte ω_0 , ω_1 und ω_2 .

Hinweis: $S[f] = \frac{b-a}{6} (f(a) + 4f(\frac{a+b}{2}) + f(b))$.

¹https://de.wikipedia.org/wiki/Conways_Spiel_des_Lebens

- c) Nun untersuchen wir den sogenannten Genauigkeitgrad, d.h. der maximale Polynom Grad welche eine Quadraturregel exakt integrieren kann, der Simpson-Regel. Ohne Beschränkung der Allgemeinheit setzen wir $a = -1$ und $b = 1$ (jedes 1-dimensionale Intervall kann auf $[-1, 1]$ transformiert werden). Berechnen Sie

$$S[1], S[x], S[x^2], S[x^3] \text{ und } S[x^4],$$

und vergleichen Sie diese mit den exakten Integralen

$$I[1] = 2, I[x] = 0, I[x^2] = \frac{2}{3}, I[x^3] = 0 \text{ und } I[x^4] = \frac{2}{5}.$$

Was beobachten Sie?

3. In der Vorlesung haben wir polynomiale Interpolation zur Approximation von Funktionen kennengelernt. In dieser Aufgabe wollen wir qualitativ die Güte der Approximation untersuchen und dabei ein paar nützliche MATLAB Befehle kennenlernen. Hierzu betrachten wir die sogenannte Runge-Funktion

$$f(x) = \frac{1}{1 + 25x^2}, \quad x \in [-1, 1].$$

- a) Verwenden Sie jeweils 4, 6, 8 und 20 äquidistante Stützstellen und interpolieren Sie die Runge-Funktion. Plotten Sie die Interpolationspolynome zusammen mit der Runge-Funktion. Verwenden Sie `polyfit` zur Berechnung der Interpolationspolynome und `polyval` zu deren Auswertung. Die äquidistanten Stützstellen können bequem mit dem MATLAB Befehl `linspace` erstellt werden.

Was beobachten Sie?

- b) Anstatt äquidistanter Stützstellen verwenden Sie die sogenannten Tschebyschow-Stützstellen

$$x_j = \cos\left(\frac{2j+1}{2n+2}\pi\right), \quad j = 0, 1, \dots, n,$$

und wiederholen Sie a). Was beobachten Sie?

Dieses Beispiel sollte darauf hinweisen, dass man polynomiale Interpolation nicht blind verwenden sollte. Wir werden dies nicht weiter in dieser Vorlesung diskutieren und für weitere Informationen empfehlen wir die zusätzlich angegebene Literatur.

Abgabe: Bis Freitag, den 05.03.2021.

Laden Sie Ihre MATLAB -Programme unter `sam-up.math.ethz.ch` hoch.

Die schriftlichen Ergebnisse sollten Sie separat in den jeweiligen Übungsgruppen abgeben.