Institut für Numerische Simulation
Rheinische Friedrich-Wilhelms-Universität Bonn

universitätbonn

Übungen zu Algorithmische Mathematik II (V1G6)

Sommersemester 2015

Prof. Dr. Martin Rumpf — Dr. Orestis Vantzos — Dipl.-Math. Behrend Heeren

Programmieraufgabe 3 — Abgabe: freiwillig

In Programmieraufgabe 2 wurde die Auswertung des Laplace-Operators Δ auf einer Gitterfunktion u_h behandelt, indem die Anwendung der Matrix $A_h \approx -\Delta$ mit Hilfe des Finite-Differenzen-Sterns programmiert wurde. Nun sollen die niedrig frequenten Eigenschwingungen einer Membran (Grundton, Obertöne) als (approximative) Eigenvektoren von A_h zu kleinen Eigenwerten mit Hilfe der *inversen Vektoriteration* (2.26) berechnet werden.

Seien $0 < \lambda_1 \leq \ldots \leq \lambda_n$ die Eigenwerte der positiv-definiten Matrix $A_h \in \mathbb{R}^{n,n}$ und $v^1, \ldots, v^n$ dazugehörige Eigenvektoren. Für $m \leq n$ berechne nun eine Folge $(\lambda_1^\epsilon, v^{1,\epsilon}), \ldots, (\lambda_m^\epsilon, v^{m,\epsilon})$ von approximativen Eigenwerten bzw. Eigenvektoren:

Algorithmus: Sei $l \geq 0$. Seien $(\lambda_1^\epsilon, v^{1,\epsilon}), \ldots, (\lambda_l^\epsilon, v^{l,\epsilon})$ berechnet, $x^0 \in \mathbb{R}^n$ ein zufällig initialisierter Vektor. Setze $k = 0$ und iteriere:

(1.) $\hat{x}^k = x^k - \sum_{j=1}^{l-1} \langle x^k, v^{j,\epsilon} \rangle v^{j,\epsilon}$

(2.) Löse $A_h \tilde{x}^{k+1} = \hat{x}^k$, setze $\mu_k = \|\tilde{x}^{k+1}\|$ und $x^{k+1} = \dfrac{\tilde{x}^{k+1}}{\mu_k}$.

(3.) Falls $|\mu_k - \mu_{k-1}| < \epsilon$ setze $\lambda_{l+1}^\epsilon = \mu_k^{-1}$ und $v^{l+1,\epsilon} = x^{k+1}$, sonst setze $k = k+1$ und gehe zu (1.)

(4.) Output: (approx.) Eigenvektor $v^{l+1,\epsilon}$ zum (approx.) Eigenwert λ_{l+1}^ϵ

Aufgabe:

- Programmieren Sie die inverse Vektoriteration mit Projektion zur Berechnung der $m = 12$ kleinsten Eigenwerte von A_h. Wählen Sie dazu $\epsilon = 10^{-5}$.
- Benutzen Sie aus der letzten Programmieraufgabe das CG-Verfahren zur Lösung des Systems $A_h u_h = f$, mit $f = \hat{x}^k$ und $x^{k+1} = u_h$.
- Wählen Sie als Definitionsgebiet Ω (welches die Form der Membran beschreibt) z.B. $\Omega = [0,1]^2$ oder das L-Gebiet aus der letzten Programmieraufgabe. Verwenden Sie wiederum die Maske χ mit $\chi(x) = 1$ für $x \in \Omega$.

Hinweis: Benutzen Sie das auf der Homepage bereitgestellte C++ Codefragment.