

Numerische Softwareentwicklung in C und C++ Wintersemester 2016/17

Projekt elliptische Randwertprobleme

Ziel des Projekts ist die Entwicklung eines C- oder C++-Programms, welches numerische Lösungen von elliptischen Randwertproblemen in zwei Raumdimensionen der Form

$$-\Delta u(x, y) + \lambda u(x, y) = f(x, y) \quad \text{in } \Omega \subseteq \mathbb{R}^2, \quad (1a)$$

mit Dirichlet-Randbedingungen

$$u(x, y) = g(x, y) \quad \text{auf } \Gamma_D, \quad (1b)$$

und Neumann-Randbedingungen

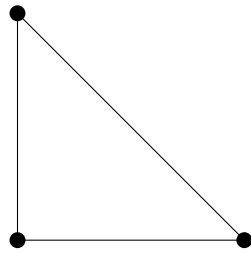
$$\frac{\partial u(x, y)}{\partial n} = h(x, y) \quad \text{auf } \Gamma_N, \quad (1c)$$

in einem polygonalen Gebiet $\Omega \subseteq \mathbb{R}^2$ mit Dirichletrand Γ_D und Neumannrand Γ_N , wobei $\Gamma_D \cup \Gamma_N = \partial\Omega$ und $\Gamma_D \cap \Gamma_N = \emptyset$, berechnet. Dabei seien $\lambda \in \mathbb{R}$, $\lambda \geq 0$ konstant, n der äußere Normalenvektor an $\partial\Omega$ und die Funktionen f , g und h seien gegeben.

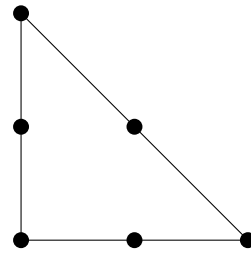
Bemerkung: Für $\lambda = 0$ erhält man die Poisson-Gleichung.

Verwenden Sie die Finite Elemente Methode (FEM) zur numerischen Lösung:

- Betrachten Sie dabei die folgenden zwei FE-Typen:
 1. **Lineare Lagrange- (Dreiecks-)elemente**, d. h. die Formfunktionen sind linear und die lokalen Freiheitsgrade sind die Werte der Formfunktionen an den Eckpunkten des Dreiecks (vgl. Abbildung 1a) sowie
 2. **Quadratische Lagrange- (Dreiecks-)elemente**, d. h. die Formfunktionen sind quadratisch und die lokalen Freiheitsgrade sind die Werte der Formfunktionen an den Eckpunkten sowie an den Seitenmittelpunkten des Dreiecks (vgl. Abbildung 1b)
- Lösen Sie das diskrete lineare Gleichungssystem
 1. mit einem Löser Ihrer Wahl und
 2. mit einem iterativen Verfahren mit Vorkonditionierung Ihrer Wahl.



(a) Lineares Lagrange-Element



(b) Quadratisches Lagrange-Element

Abbildung 1: Freiheitsgrade des linearen und des quadratischen Lagrange-Elements

Die Projektarbeit besteht aus drei Teilen,

- einer Implementierung in C oder C++,
- dessen schriftlicher Ausarbeitung sowie
- einer Präsentation (im Rahmen eines Vortrags).

Allgemeine Hinweise:

- Testen Sie Ihr Programm mindestens mit den folgenden drei Testproblemen:

Problem *Dirichlet1*:

- $\Omega = [0, 1] \times [0, 1]$ (Einheitsquadrat)
- $\Gamma_N = \emptyset, \Gamma_D = \partial\Omega$
- $\lambda = 1$
- Rechte Seite und homogene Dirichlet-Randbedingung

$$f(x, y) = (1 + 2\pi^2) \sin(\pi x) \sin(\pi y)$$

$$g(x, y) \equiv 0$$

Problem *Dirichlet2*:

- Ω ist ein L-förmiges Gebiet mit 3 Löchern (vgl. Abbildung 2)
- $\Gamma_N = \emptyset, \Gamma_D = \partial\Omega$
- $\lambda = 0.5$
- Rechte Seite und Dirichlet-Randbedingung

$$f(x, y) = 100(x^2 + y^2)$$

$$g(x, y) \equiv 1$$

Problem Neumann1:

- $\Omega = [0, 1] \times [0, 1]$ (Einheitsquadrat)
- $\Gamma_D = \emptyset, \Gamma_N = \partial\Omega$
- $\lambda = 1$
- Rechte Seite und Neumann-Randbedingung

$$f(x, y) = (1 + 2\pi^2) \sin(\pi x) \sin(\pi y)$$

$$h(x, y) = -\pi(\sin(\pi x) + \sin(\pi y))$$

Bemerkung: Sie dürfen zusätzlich auch andere Testprobleme Ihrer Wahl betrachten.

- Visualisieren Sie Ihre Ergebnisse mit *Paraview*.
- Fertigen Sie eine schriftliche Ausarbeitung an, in der Sie Ihr Vorgehen beschreiben. Die Ausarbeitung sollte mit einer kurzen Problembeschreibung beginnen. Erläutern Sie im Folgenden insbesondere Ihre Diskretisierungen, die Programm-Module Ihrer Implementierung, verwendete Bibliotheken und interpretieren Sie Ihre Ergebnisse. Untersuchen Sie auch die Approximationsgenauigkeit der zwei FE-Typen (z. B. in Form einer Tabelle). Nutzen Sie dabei folgende analytische Lösung für die Testprobleme *Dirichlet1* und *Neumann1*, um für diese Probleme den Fehler in geeigneten Normen zu berechnen

$$u(x, y) = \sin(\pi x) \sin(\pi y) \tag{2}$$

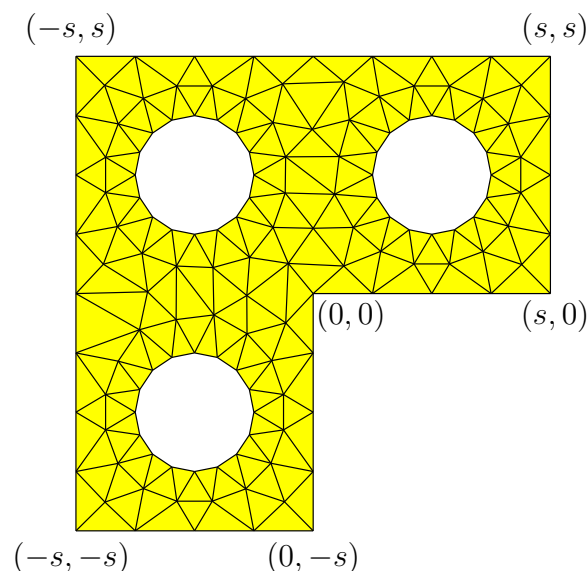


Abbildung 2: L-förmiges Gebiet des Testproblems *Dirichlet2* mit Seitenlängen $2s$ bzw. s , $s = 0.64$ und 3 Löchern (16-Ecke) mit Radii $r = s/4$.

Hinweise zur Implementierung:

- Schreiben Sie ein Makefile, um Ihr Programm zu übersetzen.
- Kommentieren Sie Ihren Quellcode.
- Sie dürfen sich an Implementierungen von frei verfügbarer FE-Software orientieren, müssen aber ein eigenes Programm schreiben.
- Für einzelne Programm-Module (z. B. für die Triangulierung) dürfen Sie frei verfügbare Bibliotheken verwenden.

Hinweise zur schriftlichen Ausarbeitung:

- Der Umfang sollte maximal 10 Seiten betragen.
- Verwenden Sie, falls möglich LaTeX.
- Achten Sie auf eine ordentliche Zitierweise, wenn Sie Referenzen verwenden.

Hinweise zur Präsentation:

- Die Vorstellung Ihrer Projektarbeit sollte 10-15 Minuten nicht überschreiten.
- Verwenden Sie zur Darstellung Folien und halten Sie Ihre Implementierung bereit.

Hinweise zur Abgabe:

- Reichen Sie Ihre Projektarbeit **vollständig** (C/C++-Dateien inkl. Makefile aber ohne ausführbare Datei(en), schriftliche Ausarbeitung und Präsentation) in Form eines Archivs per E-Mail ein.
- Geben Sie zusätzlich Ihre schriftliche Ausarbeitung in gedruckter Form bei Ihrer Präsentation ab.
- Fügen Sie Ihrer Abgabe die auf der Webseite verfügbare Erklärung bei.

<p>Abgabe bis 15.02.2017 um 15:00 Uhr per E-Mail an sfriedho@math.uni-koeln.de. Die Abgabe soll in Form eines Archivs erfolgen. Bitte beachten Sie obige Hinweise zur Abgabe.</p>
--