MATHEMATISCHES INSTITUT
DER UNIVERSITÄT ZU KÖLN

Sommersemester 2018

Prof. Dr. A. Klawonn
J. Knepper, M. Sc.
M. Kühn, M. Sc.

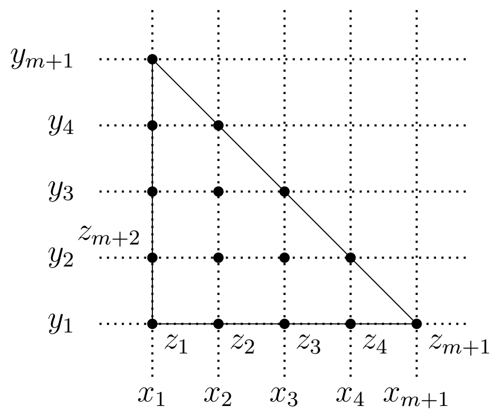
29. Mai 2018

7. Übung zur Numerik partieller Differentialgleichungen I

Hinweis: Schreiben Sie bitte auf **jedes Blatt ihren Namen**. Auf die erste Seite Ihrer Übung schreiben Sie bitte zusätzlich Ihre Matrikelnummer und die Nummer der Ihnen zugewiesenen Übungsgruppe.

Aufgabe 1: (5 Punkte)
Betrachten Sie Satz 6.1 aus der Vorlesung.

> *Sei $m \geq 0$ und T ein beliebiges Dreieck. In T seien auf $m+1$ parallelen Linien $l = 1 + 2 + \cdots + (m+1)$ Punkte $z_1, \ldots, z_l$ so angeordnet, wie in der Abbildung. Dann gibt es zu jedem $f \in \mathcal{C}^0(T)$ genau ein Polynom $p \in \mathcal{P}_m$, sodass*
> $$p(z_i) = f(z_i) \quad i = 1, \ldots, l.$$



$m = 4$

In der Vorlesung wurde die Existenzaussage bewiesen. Zeigen Sie die Eindeutigkeit des Polynoms.

Aufgabe 2: (8 Punkte)
Sei $\mathcal{P}_k^n$ der Raum der Polynome vom Grad $\leq k$ in n Variablen. Zeigen Sie

$$\dim(\mathcal{P}_k^n) = \binom{n+k}{k}.$$

Aufgabe 3: (5 Punkte)
Sei $\Omega \subset \mathbb{R}^2$ ein Polygon und, darauf gegeben, τ_h eine Triangulierung mit $r := |\tau_h|$ Dreiecken und q Knoten. Es seien φ_i nodale Basisfunktionen des Finite-Elemente-Raumes

$$H_0^1(\Omega) \supset V^h := \{v \in \mathcal{C}^0(\overline{\Omega})\colon v|_T \in \mathcal{P}_k(T),\ \forall T \in \tau_h\}.$$

Vergleichen Sie die knoten- und elementbasierten Ansätze (d.h. ohne („naiv“) bzw. mit („geschickt“) Transformation auf das Referenzelement) zur Assemblierung der Steifigkeitsmatrix zur Poissongleichung,

$$K = \Big(\int_\Omega \nabla\varphi_i \nabla\varphi_j^T \, \mathrm{d}x \Big)_{i,j=1}^q,$$

und der Massenmatrix

$$M = \Big(\int_\Omega \varphi_i \varphi_j \, \mathrm{d}x \Big)_{i,j=1}^q.$$

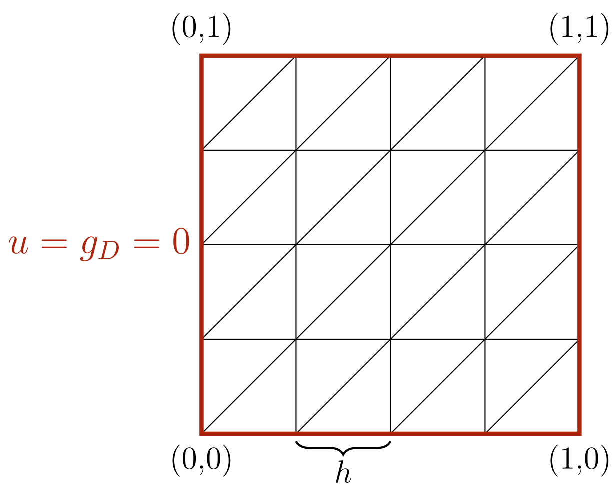
Quantifizieren Sie dabei die Unterschiede im Aufwand in Abhängigkeit des Polynomgrades k, der Anzahl Dreiecke r und der Knotenanzahl q.

Programmieraufgabe 1: (10+5 = 15 Punkte) **Abgabe bis 05. Juni 2018, 12:00 Uhr**

Schreiben Sie in Matlab ein Finite-Elemente-Programm zur Lösung des Randwertproblems

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega = (0,1)^2, \\ u &= g_D = 0 && \text{auf } \partial\Omega. \end{aligned}$$

Zerlegen Sie das Einheitsquadrat (wie in der Abbildung) gleichmäßig in Dreiecke und verwenden Sie $\mathcal{P}_1$-Ansatzfunktionen.



a)
1. Definieren Sie in Matlab zwei `logical` Vektoren (*true/false*), um die Zugehörigkeit von Knoten zu charakterisieren (innere Knoten, Dirichletrandknoten).
2. Schreiben Sie eine Funktion, die für ein Dreieck T die Matrix und den Vektor der affin-linearen Abbildung $B\hat{x} + d$ aufstellt, welche Punkte des Referenzdreiecks $\hat{T}$ auf T abbildet.
3. Modifizieren Sie Ihre Berechnung der Elementsteifigkeitsmatrizen, Elementlastvektoren und Elementmassenmatrizen aus Programmieraufgabe 2, Blatt 5, und integrieren Sie die Transformation auf das Referenzelement.
4. Stellen Sie die Steifigkeits- und Massenmatrix sowie den Lastvektor global auf. Verwenden Sie für die Matrizen das `sparse`-Format, um deren Dünnbesetztheit auszunutzen.
5. Setzen Sie die Randbedingungen in der Steifigkeitsmatrix und dem Lastvektor.
6. Berechnen Sie die Galerkin-Lösung.

b) Testen Sie Ihr Programm mit

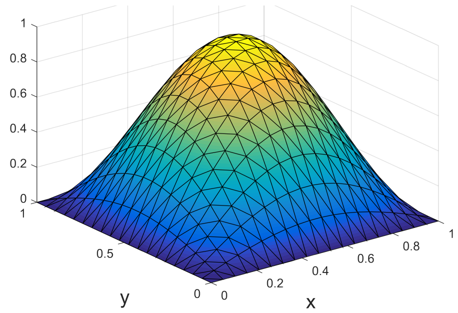
$$f(x,y) := 2\pi^2 \sin(\pi x)\sin(\pi y) \quad \text{in } \Omega,$$

für $h = 1/4, 1/16, 1/64$ und $1/256$. Die exakte Lösung ist durch $u = \sin(\pi x)\sin(\pi y)$ gegeben.

Mit $I^h(u)$ bezeichnen wir die Interpolation der exakten Lösung in den Finite-Elemente-Raum: Sei zu den Knoten $\{x_i\}_{i=1}^m$ die nodale Basis $\{\varphi_i\}_{i=1}^m$ des Finite-Elemente-Raumes gegeben, dann gilt $I^h(u) = \sum_{i=1}^m u(x_i)\varphi_i$. Vergleichen Sie die Galerkin-Lösungen u_h mit $I^h(u)$:

1. Stellen Sie für $h = 1/16, 1/64$ die Galerkin-Lösung u_h sowie die Differenz $I^h(u) - u_h$ grafisch dar.
2. Berechnen Sie für alle h die Fehlermaße $||I^h(u) - u_h||_{L^2(\Omega)}$, $|I^h(u) - u_h|_{H^1(\Omega)}$, $||I^h(u) - u_h||_{H^1(\Omega)}$, $||I^h(u) - u_h||_{L^\infty(\Omega)}$ und stellen Sie deren Verlauf in einem Plot gegenüber.

Hinweis:



Galerkin-Lösung für $h = 1/19$.

Programmieraufgabe 2: (5 Punkte) **Abgabe bis 12. Juni 2018, 12:00 Uhr**

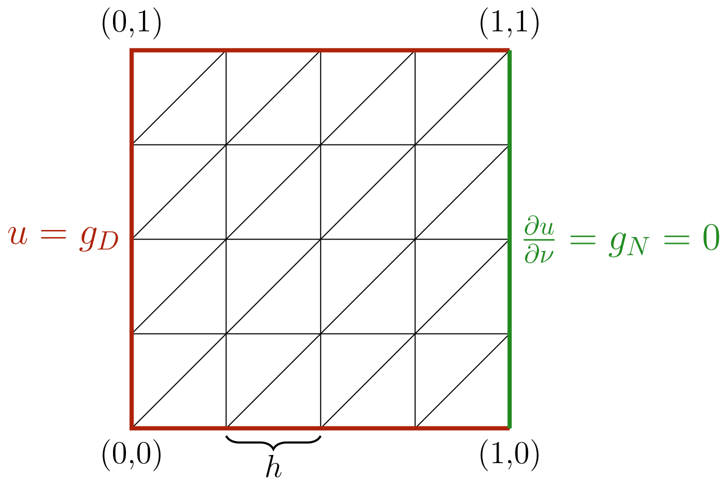
Sei $\Omega = (0,1)^2$ und der Rand von Omega aufgeteilt in

$$\partial\Omega_N := \{(1,y) \in \mathbb{R}^2 : 0 < y < 1\},$$
$$\partial\Omega_D := \partial\Omega \setminus \partial\Omega_N.$$

Erweitern Sie Ihr Programm aus der ersten Aufgabe zur Lösung folgendes Randwertproblems:

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega = (0,1)^2, \\ u &= g_D && \text{auf } \partial\Omega_D, \\ \frac{\partial u}{\partial \nu} &= g_N = 0 && \text{auf } \partial\Omega_N, \end{aligned}$$

wobei ν der äußere Einheitsnormalenvektor von $\partial\Omega_N$ ist.



Berechnen Sie für $h = 1/16$ und

$$\begin{aligned} f(x, y) &:= 1 && \text{in } \Omega, \\ g_D(x, y) &:= 0.1(x-1)^2 + 0.1\sin(-\pi y) && \text{auf } \partial\Omega_D, \end{aligned}$$

die Galerkin-Lösung und stellen Sie diese grafisch dar.

Allgemeine Hinweise zum Programmierteil

- Schreiben Sie das Programm in Matlab.
- Der Code **muss sinnvoll kommentiert** sein. Ein nicht kommentiertes Programm gilt als nicht erfolgreich bearbeitet.
- Das Programm muss ausführbar sein, ohne Änderungen am Code vornehmen zu müssen (d.h. ein Klick auf „Ausführen“ muss ausreichen). Schreiben Sie daher ein oder mehrere Skripte für die Teilaufgabe(n). Benennen Sie das Skript / die Skripte sinnvoll (z.B. `aufg1c.m`).
- Schreiben Sie bitte Funktionen in eigene Dateien und nicht in Skriptdateien (*Ausnahme:* anonyme Funktionen der Art `f = @(x) x.^2;`).
- Enthält ihr Code mehrere Funktionen, so ist jede Funktion in eine eigene Datei zu schreiben. *Ausnahme:* Die Funktion wird ausschließlich von anderen Funktionen derselben Datei aufgerufen. In diesem Fall steht an oberster Stelle der Funktionsdatei die Funktion, welche von außerhalb (z.B. von einem Skript) aufgerufen wird.

Abgabe des Programmierteils

- Packen Sie Ihre Dateien in ein Archiv (Formate: .zip, oder .tar.gz) mit einem Dateinamen der Art:

  `ueb01_nachname_vorname.zip`

- Den Quellcode schicken Sie bitte an die E-Mail-Adresse Ihrer Übungsgruppenleiter / Übungsgruppenleiterinnen, mit einem Betreff der Art:

  `Betreff: Uebung1, Nachname, Vorname`

- Geben Sie bitte immer eine **ausgedruckte Version** Ihrer Programmcodes mit den schriftlichen Aufgaben ab ($\rightarrow$ Kasten), sofern dies in der Aufgabenstellung nicht eindeutig anders vermerkt wurde.
- Sofern es zur sinnvollen Lösung der Aufgabenstellung nötig ist, drucken Sie bitte auch die Ausgabe von Matlab aus. Dies sollte nicht zwei DIN-A4-Seiten überschreiten. Gleiches gilt für Grafiken.

Abgabe im entsprechenden Kasten in Raum 3.01 des Mathematischen Instituts.

- Theorie und Programmieraufgabe 1: Bis Dienstag, 05. Juni 2018, 12:00 Uhr.
- Programmieraufgabe 2: Bis Dienstag, 12. Juni 2018, 12:00 Uhr.