Prof. Dr. Dirk Lebiedz
Dipl.-Math. oec. Klaus Stolle
Institut für Numerische Mathematik
Universität Ulm

WiMa-Praktikum II (Numerik)
Wintersemester 2016/2017

11. Januar 2017

Projekt 3 - Triangulierung
Datenstrukturen und uniforme Rotverfeinerung

In den letzten beiden Projekten haben wir bereits Gitter aus Dreiecken, sogenannte Triangulierungen, kennengelernt, durch die wir die Objekte der Szene dargestellt haben. Triangulierungen finden aber auch in anderen Gebieten Verwendung, beispielsweise werden sie bei der numerischen Lösung partieller Differentialgleichungen auf einem Gebiet $\Omega \subset \mathbb{R}^2$ mit der Finite-Elemente-Methode eingesetzt. In diesem Projekt wollen wir uns genauer mit den für Triangulierungen benötigten Datenstrukturen befassen und eine erste Möglichkeit zur Verfeinerung einer Triangulierung kennen lernen.

Datenstrukturen für die Triangulierung

Das folgende Beispiel einer Triangulierung von $\Omega = [0,1]^2$ soll die Verwendung der Datenstrukturen verdeutlichen:

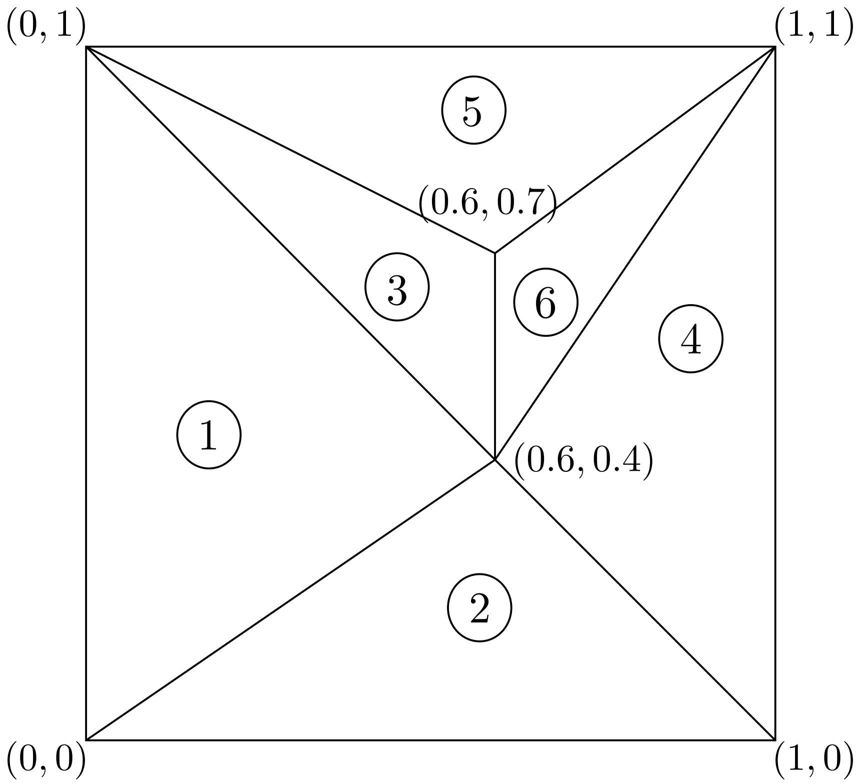

Abbildung 1: Triangulierung von $\Omega = [0,1]^2$

Um diese Triangulierung zu beschreiben, werden zwei Matrizen *coordinates* und *elements* benötigt. Die Matrix *coordinates* enthält die Koordinaten der Eckpunkte der Dreiecke und in der Matrix *elements* stehen die Nummern der Eckpunkte, die das jeweilige Dreieck begrenzen. Für unser Beispiel sehen die Matrizen wie folgt aus:

$$coordinates = \begin{pmatrix} 0.0 & 0.0 \\ 1.0 & 0.0 \\ 0.6 & 0.4 \\ 0.6 & 0.7 \\ 0.0 & 1.0 \\ 1.0 & 1.0 \end{pmatrix}, \qquad elements = \begin{pmatrix} 1 & 3 & 5 \\ 1 & 2 & 3 \\ 3 & 4 & 5 \\ 2 & 6 & 3 \\ 4 & 6 & 5 \\ 3 & 6 & 4 \end{pmatrix}.$$

Zur numerischen Lösung einer partiellen Differentialgleichung werden auch die Randdaten benötigt. Am unteren und oberen Rand des Quadrats geben wir *Dirichlet-Randbedingungen* vor, links und rechts *Neumann-Randbedingungen*. Entsprechend benötigen wir zwei Matrizen *dirichlet* und *neumann*, welche die Kanten des jeweiligen Randes enthalten. In unserem Fall sind die Matrizen gegeben durch

$$dirichlet = \begin{pmatrix} 1 & 2 \\ 6 & 5 \end{pmatrix}, \quad neumann = \begin{pmatrix} 2 & 6 \\ 5 & 1 \end{pmatrix}.$$

Aufgabenstellung zu den Datenstrukturen

Folgende Arbeitsschritte sollen ausgeführt werden:

a) Laden Sie das Material von der Homepage herunter.

b) Laden Sie mit Hilfe der Matlab-Funktion `load` die oben gegebenen Matrizen *coordinates*, *elements*, *dirichlet* und *neumann* aus den Files `coordinates.dat`, `elements.dat`, `dirichlet.dat` und `neumann.dat` in Ihr Hauptprogramm.

c) Schreiben Sie eine Funktion, die die Triangulierung mit der Matlab-Funktion `trisurf` zeichnet. Sie können hierzu den heruntergeladenen Rumpf der Funktion `zeigeTriangulierung` verwenden.

d) Berechnen Sie für alle Dreiecke die Koordinaten des jeweiligen Schwerpunktes. Die Koordinaten aller Schwerpunkte sollen in der Matrix `S` gespeichert werden.

e) Verwenden Sie die Matlab-Funktion `text`, um für jedes Dreieck die Nummer des Dreiecks an die Stelle seines Schwerpunktes zu schreiben.

f) Heben Sie den Rand der Triangulierung farbig hervor. Verwenden Sie für Randteile, auf denen Dirichlet-Randbedingungen gelten, eine andere Farbe als für Randstücke, für die Neumann-Randbedingungen gelten.

g) Schreiben Sie an die Kanten des Randes jeweils in die Mitte die Nummer der Dirichlet- bzw. Neumann-Kante.

Das Poisson-Problem

Stellvertretend für partielle Differentialgleichungen betrachten wir das Poisson-Problem auf dem Gebiet $\Omega \subset \mathbb{R}^2$.

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega \\ u &= u_D && \text{auf } \Gamma_D \subset \Gamma = \partial\Omega \quad \text{(Dirichlet Randbedingung)} \\ \frac{\partial u}{\partial n} &= g && \text{auf } \Gamma_N = \Gamma \setminus \Gamma_D \quad \text{(Neumann Randbedingung).} \end{aligned}$$

Dabei ist u die gesuchte Lösung des Problems und f eine gegebene Funktion.

Lösung des Poisson-Problems

a) Führen Sie die Funktion `fem2dLaplace` mit den entsprechenden Eingabe-Parametern aus.

b) Was können Sie auf der Lösung erkennen?

Uniforme Rot-Verfeinerung

Die in den heruntergeladenen Dateien enthaltene Triangulierung des Gebiets Ω ist offensichtlich für die numerische Lösung einer partiellen Differentialgleichung viel zu grob. Wir werden diese Triangulierung daher zunächst gleichmäßig durch eine *uniforme Rot-Verfeinerung* verfeinern. Bei der uniformen Rotverfeinerung wird jedes Dreieck in 4 kleinere Dreiecke zerlegt (siehe Abbildung 2).

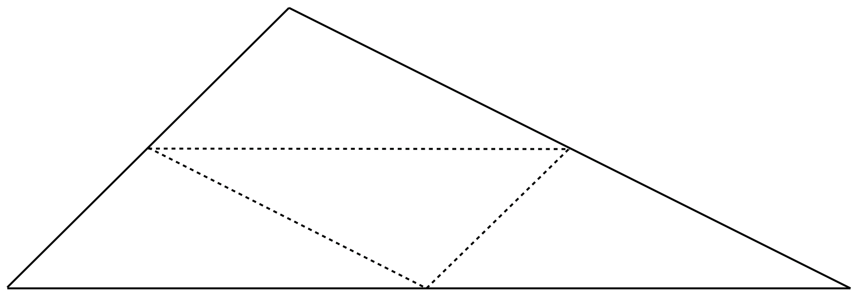

Abbildung 2: Rot-Verfeinerung

Aufgabenstellung zur uniformen Rot-Verfeinerung

Folgende Arbeitsschritte sollen ausgeführt werden:

a) Schreiben Sie eine Routine, welche eine uniforme Rot-Verfeinerung der Triangulierung vornimmt. Achten Sie dabei auf eine möglichst effiziente und elegante Programmierung. Sie können den heruntergeladenen Rumpf der Funktion `rotVerfeinerung` verwenden.

b) Verfeinern Sie in Ihrem Hauptprogramm die ursprüngliche Triangulierung mehrmals. Erstellen Sie mehrere Grafiken mit unterschiedlich feinen Triangulierungen.

c) Lösen Sie nun auf der verfeinerten Triangulierung das Poisson-Problem.