Abteilung für Angewandte Mathematik 21.01.2019
Prof. Dr. S. Bartels
M.Sc. C. Palus

Übung zur Vorlesung

Einführung in Theorie und Numerik partieller Differentialgleichungen

Wintersemester 2018/2019 – Blatt 13

Abgabe: Briefkästen RZ bis Montag, den 28.01.2019 um 12:00 Uhr

Aufgabe 1 (5 Punkte)**.** Betrachten Sie den MATLAB-Code zum $P1$-Finite-Elemente-Verfahren für das Poisson-Problem (s. Rückseite) und erläutern Sie Zeile für Zeile, wie das Verfahren im Code realisiert wird.

Aufgabe 2 (5 Punkte)**.** Für $n \in \mathbb{N}$ sei $f_n \colon [0,1] \to \mathbb{R}$ gegeben durch

$$f_n(x) = \begin{cases} x - x_k, & \text{falls } x \in (x_k, x_{k+1}) \text{ für } k \text{ gerade,} \\ x_{k+1} - x, & \text{falls } x \in (x_k, x_{k+1}) \text{ für } k \text{ ungerade,} \end{cases}$$

mit $x_k = k2^{-n}$ für $k = 0, 1, \ldots, 2^n$. Zeigen Sie, dass die Folge (f_n) stark in $L^2([0,1])$ und schwach in $H^1([0,1])$ konvergiert, jedoch nicht stark in $H^1([0,1])$.

Aufgabe 3 (5 Punkte)**.** Sei $\Omega \subset \mathbb{R}^d$ ein beschränktes Lipschitzgebiet. Beweisen Sie konstruktiv die Existenz einer Konstanten $c_p > 0$, sodass

$$\|v\|_{L^2(\Omega)} \le c_p \|\nabla v\|_{L^2(\Omega)}$$

für alle $v \in H^1(\Omega)$ mit $\int_\Omega v \,\mathrm{d}x = 0$ gilt.
Hinweis: Verwenden Sie den Mittelwertsatz, um $v(x)$ für $x \in \Omega$ als Integral über Ω darzustellen.

Aufgabe 4 (3+2 Punkte)**.** Für ein Dreieck $T \subset \mathbb{R}^2$ mit Ecken $z_0, z_1, z_2 \in \mathbb{R}^2$ seien $z_3, z_4, z_5 \in \mathbb{R}^2$ die Mittelpunkte der Seiten von T.
(i) Zeigen Sie, dass $(T, \mathcal{P}_2(T), \mathcal{K})$ mit $\mathcal{K} = \{\chi_j : j = 0, 1, \ldots, 5\}$ für $\chi_j(\phi) = \phi(z_j)$, $j = 0, 1, \ldots, 5$, ein finites Element ist.
(ii) Konstruieren Sie die nodale Basis des Elements $(T, \mathcal{P}_2(T), \mathcal{K})$.

```matlab
function p1_poisson(d,red)
[c4n,n4e,Db,Nb] = triang_cube(d);
for j = 1:red
    [c4n,n4e,Db,Nb,P0,P1] = red_refine(c4n,n4e,Db,Nb);
end
[nC,d] = size(c4n); nE = size(n4e,1); nNb = size(Nb,1);
dNodes = unique(Db); fNodes = setdiff(1:nC,dNodes);
u = zeros(nC,1); tu_D = zeros(nC,1); b = zeros(nC,1);
ctr = 0; ctr_max = (d+1)^2*nE;
I = zeros(ctr_max,1); J = zeros(ctr_max,1); X = zeros(ctr_max,1);
for j = 1:nE
    X_T = [ones(1,d+1);c4n(n4e(j,:),:)'];
    grads_T = X_T\[zeros(1,d);eye(d)];
    vol_T = det(X_T)/factorial(d);
    mp_T = sum(c4n(n4e(j,:),:),1)/(d+1);
    for m = 1:d+1
        b(n4e(j,m)) = b(n4e(j,m))+(1/(d+1))*vol_T*f(mp_T);
        for n = 1:d+1
            ctr = ctr+1; I(ctr) = n4e(j,m); J(ctr) = n4e(j,n);
            X(ctr) = vol_T*grads_T(m,:)*grads_T(n,:)';
        end
    end
end
s = sparse(I,J,X,nC,nC);
for j = 1:nNb
    if d == 1
        vol_S = 1;
    elseif d == 2
        vol_S = norm(c4n(Nb(j,1),:)-c4n(Nb(j,2),:));
    elseif d == 3
        vol_S = norm(cross(c4n(Nb(j,3),:)-c4n(Nb(j,1),:),...
            c4n(Nb(j,2),:)-c4n(Nb(j,1),:)),2)/2;
    end
    mp_S = sum(c4n(Nb(j,:),:),1)/d;
    for k = 1:d
        b(Nb(j,k)) = b(Nb(j,k))+(1/d)*vol_S*g(mp_S);
    end
end
for j = 1:nC
    tu_D(j) = u_D(c4n(j,:));
end
b = b-s*tu_D; u(fNodes) = s(fNodes,fNodes)\b(fNodes); u = u+tu_D;
if d == 1; plot(c4n(n4e),u(n4e));
elseif d == 2; trisurf(n4e,c4n(:,1),c4n(:,2),u);
elseif d == 3; trisurf([Db;Nb],c4n(:,1),c4n(:,2),c4n(:,3),u);
end

function val = f(x); val = 1;
function val = g(x); val = 1;
function val = u_D(x); val = sin(2*pi*x(:,1));
```

MATLAB-Implementierung der $P1$-Finite-Elemente-Methode für das Poisson-Problem.