ÜBUNG 16 POINCARESCHE UNGLEICHUNG

Beweisen Sie die Poincaresche Ungleichung in 1D: Sei $\Omega = (0,1)$, dann gibt es eine Konstante $C > 0$, so dass

$$\|u\|_{0,\Omega}^2 \leqslant C \left\{ |u|_{1,\Omega}^2 + \left(\int_\Omega u\,dx \right)^2 \right\}$$

4 Punkte

ÜBUNG 17 INVERSE ABSCHÄTZUNG

In der Vorlesung Numerik partieller Differentialgleichungen wurden die Fehlerabschätzungen für Finite Elemente hergeleitet, siehe Skript (Kapitel 8)
`http://cox.iwr.uni-heidelberg.de/teaching/numerik2_ss2012/num2.pdf`

Satz 1 (Fehlerabschätzung)**.** Sei $\Omega \subset \mathbb{R}^n$ ein Gebiet mit Lipschitz-stetigem Rand und $\mathcal{T}_h$ eine quasiuniforme Triangulierung von Ω. Ferner seien $k, m \in \mathbb{N}$, $k \geqslant m > \frac{n}{2}$ und $\mathcal{I}_h : H^m(\Omega) \to \mathbb{P}_{k-1}^n$ die Lagrange-Interpolation für das Gitter $\mathcal{T}_h$. Dann ist für $u \in H^m(\Omega)$ der Interpolationsfehler in der $H^m(\Omega)$-Norm beschränkt:

$$\|u - \mathcal{I}_h u\|_{m,\Omega} \leqslant ch^{k-m} |u|_{k,\Omega} \qquad \forall u \in H^m(\Omega) \; .$$

In dieser Abschätzung ist m kleiner gleich k. Der Approximationsfehler wird also in einer gröberen Norm gemessen als die gegebene Funktion.

Ziel ist es, die feinere Norm der finiten Elemente durch eine gröbere Norm abgeschätzen.

Satz 2 (Inverse Abschätzung)**.** Sei $\{\mathcal{T}_\nu\}$ eine affine Familie von Finiten Elementen mit uniformen Zerlegungen, die stückweise aus Polynomen vom Grad k bestehen. Dann gibt es eine Konstante c, so dass für $0 \leqslant s \leqslant m$

$$\|u_h\|_m \leqslant ch^{s-m} \|u_h\|_s \qquad \forall u_h \in P_k\left(\mathcal{T}_\nu\right) \; . \tag{1}$$

Beweisen Sie zuerst die Formel auf einem Referenzelement:

$$|v|_{m,\mathcal{T}_{ref}} \leqslant c|v|_{s,\mathcal{T}_{ref}} \qquad \forall v \in P_k\left(\mathcal{T}_{ref}\right) \; .$$

Hinweis: Nehmen Sie die Norm $|||\cdot|||$ wie im Beweis von Proposition 8.4.

Wie muss man fortfahren, um (1) beweisen zu können?

6 Punkte

ÜBUNG 18 PARALLELES ZWEIGITTERVERFAHREN MIT GROBGITTERKORREKTUR

Nach einem Update Ihres `dune-parsolve` Moduls finden Sie ein neues Programm `uebung/uebung07/uebung07.cc`, welches einen parallelen überlappenden Schwarzlöser mit Grobgitterkorrektur bereit stellt (additive Version). Als Standard werden hier exakte Teilgebietslöser verwendet. Das Programm löst ein 2D-Problem wie in den vorherigen Übungen.

Es erwartet vier Parameter:

1. Die Anzahl der Zellen in jeder Richtung
2. Den gewünschten Überlapp auf dem grobsten Gitter
3. Den gewünschten Überlapp auf dem feinsten Gitter
4. Das Verfeinerungslevel L

Auf dem verwendeten `YaspGrid` wird L-mal `globalRefine()` aufgerufen, d.h. es wird L-mal global uniform verfeinert. Die Aufteilung des Gitters auf Level L auf die Prozessoren stellt die lokalen Teilgebiete dar. Als Grobgitter wird das ursprüngliche Gitter verwendet. Auf diesem Gitter wird das Grobgitterproblem gelöst und als Grobgitterkorrektur verwendet.

Aufgabe 1 Schauen Sie die Dateien `geometric_multigrid_components.hh`, `two_level_schwarz.hh` und `uebung07.cc` an. Beschreiben Sie mit eigenen Worten, was alles zu tun ist, um das Problem mit Grobgitterkorrektur lösen zu können. Ein genauer Blick ist zu empfehlen, da wir in den nächsten Übungen dieses Program weiter entwickeln werden!

Aufgabe 2 Schätzen Sie die Kosten im Vergleich zur Variante ohne Grobgitterkorrekturen. Kann man beliebig das Gitter vergröbern?

Aufgabe 3 Führen Sie mit dem Programm Testrechnungen durch, und zwar auf einem Gitter der Größe 512×512. Messen Sie dabei Folgendes:

- Die benötigte Anzahl an Iterationsschritten für $1, 4, 16, 64$ Prozessoren. Nehmen Sie verschiedene Overlaps. Gerne können Sie auch mit den Werten spielen. Vergleichen Sie mit der Variante ohne Grobgitterorrektur.
  (siehe `uebung/loesung04/additive_schwarz.cc`
- Den Speedup der benötigten Zeit pro Iterationsschritt. Hier müssen Sie natürlich sicher stellen, dass nur ein Prozess pro Prozessor benutzt wird. Der Pool hat maximal 50 Rechner mit je 2 Prozessoren.

Stellen Sie Ihre Ergebnisse graphisch (z.b. mit gnuplot) dar.

10 Punkte