

AUFGABE 20 PARALLELES MEHRGITTER MIT MINIMALER ÜBERLAPPUNG

Wir beschränken uns der Einfachheit halber auf 1D.

Sei $\hat{\Omega}_i^0 = \{x \in \Omega_1^0 \mid \text{dist}(x, \Omega_i^0) < H\}$ die überlappende Zerlegung des Grobgitters der Gitterweite H . Sei $\tilde{I}_i^0 = \{j \in I^0 \mid x_j \in \hat{\Omega}_i^0\}$. Dieses soll nun zur Implementierung eines parallelen Mehrgitterverfahrens benutzt werden. Als Diskretisierungsverfahren werden lineare Finite Element mit Standard-Lagrange-Basisfunktionen (P_1) verwendet.

- a) Laut Vorlesungsskript muss die Datenverteilung folgendes erfüllen: Für Paare $\tilde{I}_i^l, \tilde{I}_i^{l+1}$, $l = 0, \dots, L-1$ (feinstes Level ist L) muss

$$j \in \tilde{I}_i^{l+1} \wedge \theta_{k,l}^l \neq 0 \implies k \in \tilde{I}_i^l$$

gelten.

Beschreiben sie die $\hat{\Omega}_i^l$ mit minimaler Überlappung und skizzieren Level 0, 1, 2 für zwei Prozessoren und aus nur einem Element bestehenden Ω_i^0 .

- b) Wir betrachten das Laplace Problem. Prozessor i berechnet und speichert die Korrektur für alle Indices \tilde{I}_i^l . Die lokalen Anteile am Gleichungssystem im Prozessor i seien:

$$(A_i^l)_{\alpha,\beta} = \int_{\Omega_i} \nabla \varphi_\alpha^l \cdot \nabla \varphi_\beta^l dx$$

$$(b_i^l)_\alpha = \int_{\Omega_i} f \varphi_\alpha^l dx.$$

Stellen sie A_i^1 für zwei Prozessoren mit obiger Datenverteilung des Grobgitters auf.

- c) Sei $A^l x^l = b^l$ das sequentielle Gleichungssystem. Zeigen Sie das gilt:

$$b^l = \sum_{i=1}^p (\tilde{R}_i^l)^T b_i^l$$

$$A^l = \sum_{i=1}^p (\tilde{R}_i^l)^T A_i^l \tilde{R}_i^l,$$

sowie für den Defekt $d^l = b^l - A^l x^l$

$$d^l = \sum_{i=1}^p (\tilde{R}_i^l)^T d_i^l,$$

falls $(x_i^l)_k = (x^l)_k$ für $k \in I_i^l$ gilt.

10 Punkte

AUFGABE 21 BPX-VERFAHREN

Nach einem Update Ihrer DUNE Module (`svn update` im Verzeichnis `dune-parsolve`) finden Sie ein neues Programm

`src/parallel_bpx.cc`, welches einen paralleles Multilevel-Diagonal-Scaling-Verfahren (MDS oder auch BPX) implementiert, in ihrem Verzeichnes `dune-parsolve`.

Das Programm bekommt 1-2 Parameter auf der Kommandozeile übergeben:

- a) Das maxlevel bis zu dem verfeinert wird.
- b) Die Anzahl elemente auf dem Grobgitter in jeder Dimension ($1/h$) (optional, wenn nicht angegeben, wird ein Element pro Prozess angenommen).

Führen Sie Testrechnungen mit dem Programm aus. Lassen Sie einmal die globale Grobgittergröße (starke Skalierbarkeit) und einmal die Grobgittergröße pro Prozess (schwache Skalierbarkeit) fest. Nehmen sie als Standard ein maximales Verfeinerungslevel von 8. Messen Sie dabei folgendes für 1, 4, 16, 64 Prozessoren:

- Die benötigte Anzahl an Iterationsschritten.
- 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.
- Den erreichten minimalen Defekt.

Interpretieren Sie die Ergebnisse. Wie erklären Sie sich das beobachtete Verhalten? *10 Punkte*