

Praktikum Multicore-Programmierung

Abschlussprojekt 1

Sarah Lutteropp und Johannes Sailer

31. Dezember 2015

Zusammenfassung

Ausarbeitung für das Abschlussprojekt “Projekt 1: Parallele Lösungsmethodik partieller Differentialgleichungen” für das Praktikum Multicore-Programmierung im WS 15/16.

1 Mathematischer Hintergrund

Wir haben folgendes Setting:

- $\Omega \cup \Gamma = [0, 1]^2$
- Rand $\Gamma = \{(x, y) \in \mathbb{R}^2 : x \in \{0, 1\} \text{ oder } y \in \{0, 1\}\}$
- $-\Delta u(x, y) = f(x, y) \quad \forall (x, y) \in (0, 1)^2$
- $u(x, y) = 0 \quad \forall (x, y) \in [0, 1]^2 \setminus (0, 1)^2$
- Diskretisierungsparameter h
- (x, y) sind also Koordinaten im diskretisierten Einheitsquadrat
- Vereinfachte Schreibweise: $u_{i,j} = u(x_i, y_j) \in \mathbb{R}$
- n ist die Anzahl der Gitterpunkte pro Dimension
- Im Vektor $u^k \in \mathbb{R}^{n^2}$ stehen die Werte von u für alle Gitterpunkte im Iterationsschritt k . Diese Notation ist echt doof, weil man das ständig mit der Funktion u selbst verwechselt. :-)
- Wir benötigen im Code keine einzige Matrix!

Blabla

1.1 Jacobi-Verfahren

Blabla

1.1.1 Herleitung

Blabla

1.1.2 Abbruchkriterium

Blabla

1.2 Gauß-Seidel-Verfahren

Blabla

1.2.1 Herleitung

Blabla

1.2.2 Abbruchkriterium

Blabla

1.3 Vergleich der Konvergenz und Stabilität beider Verfahren

Blabla

2 Serielle Implementierung

Blabla

2.1 Laufzeiten bei verschiedenen Verfeinerungen

Blabla

2.2 Approximationsfehler

Blabla

3 Parallelisierung

Blabla

3.1 Jacobi-Verfahren

Blabla

3.2 Gauß-Seidel-Verfahren

Blabla

3.2.1 Naiver Parallelisierungsansatz

Blabla

3.2.2 Erweiterter Parallelisierungsansatz

Blabla

4 Methodenwahl

Blabla