
Parallelisierung mit MPI (Gauß-Seidel: 600 Punkte)

Nun soll auch das Gauß-Seidel-Verfahren mittels Nachrichtenaustausch mit MPI parallelisiert werden. Dies soll natürlich so geschehen, dass auch ein Aufruf des Programms im Jacobi-Modus noch dieselben Resultate wie im sequentiellen Fall liefert.

Überprüfen Sie zunächst die Korrektheit der Parallelisierung des Gauß-Seidel-Verfahrens für die beiden Fälle (1) Abbruch nach Iterationszahl und (2) Abbruch nach Genauigkeit!

Im ersten Fall muss das Ergebnis identisch zum sequentiellen Programm sein. Bei dem Abbruchkriterium „erreichte Genauigkeit“ muss das Programm nicht notwendigerweise bei der selben Iteration abbrechen. Das Ergebnis muss bei fester Iterationszahl jedoch nach wie vor unabhängig von der Prozessanzahl sein (ansonsten ist das Programm wieder **falsch**).

Desweiteren stellen Sie bitte sicher, dass auch nach diesem Arbeitsschritt das Jacobi-Verfahren einwandfrei funktioniert.

Vorgaben & Hinweise

- Das Programm darf nicht langsamer als die sequentielle Variante sein.
- Das Programm muss weiterhin mit einem Prozess funktionieren.
- Erstellen Sie eine eigene Funktion für die MPI-Parallelisierung des Gauß-Seidel-Verfahrens.

Abgabe des Programms

Abzugeben ist ein gemäß den bekannten Richtlinien erstelltes und benanntes Archiv (`.tar.gz`). Das enthaltene und gewohnt benannte Verzeichnis soll folgenden Inhalt haben:

- Alle Quellen, aus denen Ihr Programm besteht; gut dokumentiert (Kommentare bei geänderten Code-Teilen!)
- Ein Makefile derart, dass `make partdiff-par` ihr parallelisiertes Programm mit dem Namen `partdiff-par` generiert, das sich **genauso** aufrufen lässt wie das vorgegebene `partdiff-seq`. Auch für das parallele Programm irrelevante Parameter müssen erhalten bleiben. `make clean` und `make` sollen erwartungsgemäß funktionieren. `make` soll dabei natürlich `partdiff-par` erzeugen.
- **Keine** Binärdateien!

Senden Sie das Archiv per Mail an `hr-abgabe@wr.informatik.uni-hamburg.de`.

Hinweis: *Die Bearbeitungszeit ist schwer zu schätzen, da sie sehr stark von Ihren Vorkenntnissen und dem Glück, mit dem Sie auf Anhieb eine einigermaßen fehlerfreie MPI-Implementierung hinbekommen, abhängt. Bei komplexen Fehlern kann sich der Aufwand aber leicht stark erhöhen, fangen Sie deshalb **frühzeitig** an. Das bedeutet: sofort in diesen Tagen!*

Bewertungskriterien

- Strukturierter, gut dokumentierter Programmcode
- Fehlerfrei kompilierbares und lauffähiges Programm
- Pünktliche Abgabe der Ergebnisse
- Vollständigkeit der geforderten Materialien
- Korrektheit der Form der Abgabe