Universität Hamburg, Fachbereich Informatik
Arbeitsbereich Wissenschaftliches Rechnen
Prof. T. Ludwig, Dr. M. Kuhn
M. Blesel, T. Jammer, J. Plehn

Übungsblatt 9 zur Vorlesung
Hochleistungsrechnen
im WiSe 2018/2019
Abgabe: 15.12.2018 und 12.01.2019, 23:59

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 Abbruch nach Iterationszahl und Abbruch nach Genauigkeit!

Im ersten Fall muss das Ergebnis identisch zum sequentiellen Programm sein. Bei dem Abbruchkriterium „erreichte Genauigkeit“ muss die parallele Variante nicht unbedingt bei derselben Iteration wie die sequentielle abbrechen. Das Ergebnis muss bei gleicher 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.

Für dieses Aufgabenblatt findet die Abgabe in zwei Stufen statt. In der ersten Übung sollen die bisherigen Überlegungen vorgestellt werden. Hierzu soll aus der `calculate`-Funktion ein übersichtlicher Pseudocode mit den nötigen MPI-Aufrufen erstellt werden. Die Angabe der Parameter im Pseudocode kann auf die relevanten Informationen (z. B. beteiligte Prozesse) reduziert werden. Abhängigkeiten, die Ihnen bekannt sind aber sich nicht durch MPI-Funktionsaufrufe ausdrücken lassen, merken Sie als Kommentare im Pseudocode an. Für das Vorstellen des Pseudocodes in der Übung erhalten Sie bis zu 90 der ingesamt 600 möglichen Punkte. Jede Gruppe wird die Möglichkeit bekommen ihren Entwurf vorzustellen.

Vorgaben & Hinweise

- Das Programm darf nicht langsamer als die sequentielle Variante sein.
- Zu keinem Zeitpunkt darf ein Prozess die gesamte Matrix im Speicher halten.
- Das Programm muss weiterhin mit einem Prozess funktionieren.
- Das Programm muss mit beliebigen Prozesszahlen funktionieren.
- Erstellen Sie eine eigene Funktion für die MPI-Version des Gauß-Seidel-Verfahrens.

Abgabe des Entwurfs (15.12.2018)

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

- Eine Datei `entwurf.txt`, welche den Pseudocode der parallelisierten `calculate`-Funktion enthält.

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

Abgabe des Programms (12.01.2019)

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

- Alle Quellen, aus denen Ihr Programm besteht; gut dokumentiert (Kommentare bei geänderten Code-Teilen!)
  - Erwartet werden die Dateien `Makefile`, `askparams.c`, `partdiff.c` und `partdiff.h`.
- 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`. Auch für das parallele Programm irrelevante Parameter müssen erhalten bleiben. `make clean` und `make` sollen erwartungsgemäß funktionieren. `make` soll dabei `partdiff-par` erzeugen.
- **Keine** Binärdateien!

Senden Sie das Archiv 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!