Universität Hamburg, Fachbereich Informatik
Arbeitsbereich Wissenschaftliches Rechnen
Prof. T. Ludwig, Dr. M. Kuhn
K. Duwe, K. Tesch, E. Zickler

Übungsblatt 8 zur Vorlesung
Hochleistungsrechnen
im WiSe 2016/2017
Abgabe: 10.12.2016, 23:59

Parallelisierung mit MPI (Jacobi: 300 Punkte)

Parallelisieren Sie in einem ersten Schritt das Jacobi-Verfahren in dem Programm. Dabei soll nach gleicher Iterationszahl das Ergebnis identisch bleiben. Außerdem soll bei Abbruch nach Genauigkeit im parallelen Programm nach derselben Iterationszahl wie im sequentiellen abgebrochen werden.

Es gibt hier zwei Fälle, die auf Korrektheit der Parallelisierung zu prüfen sind:

1. Jacobi-Verfahren mit Abbruch nach fester Iterationszahl
2. Jacobi-Verfahren mit Abbruch nach Genauigkeit

Überprüfen Sie, dass die Ergebnisse mit 24 Prozessen auf zwei Knoten identisch zum sequentiellen Fall sind. **Wichtig: wenn dies nicht der Fall ist, ist Ihr Programm falsch! Wichtig: wenn dies nicht der Fall ist, ist Ihr Programm falsch!** ☺ Es steht hier doppelt, weil die Chance, dass zunächst ein abweichendes Ergebnis herauskommt, wohl gegen 100% geht.

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-Parallelisierung des Jacobi-Verfahrens.
- Benutzen Sie die in den Materialien bereitgestellte `DisplayMatrix`-Funktion als Grundlage für die parallele Ausgabe der Matrix.

Hybride Parallelisierung (60 Bonuspunkte)

Parallelisieren Sie Ihre MPI-Version des Jacobi-Verfahrens zusätzlich mittels OpenMP.

Leistungsanalyse

Ermitteln Sie die Leistungsdaten Ihres Hybrid-Programms und vergleichen Sie die Laufzeiten für folgende Konfigurationen in einem Diagramm:

- 3 Knoten $\times$ 12 Prozesse
- 3 Knoten $\times$ 24 Prozesse

- 3 Knoten $\times$ 1 Prozess $\times$ 12 Threads
- 3 Knoten $\times$ 1 Prozess $\times$ 24 Threads
- 3 Knoten $\times$ 2 Prozesse $\times$ 6 Threads
- 3 Knoten $\times$ 2 Prozesse $\times$ 12 Threads
- 3 Knoten $\times$ 12 Prozesse $\times$ 2 Threads

Verwenden Sie hierzu 512 Interlines. Der kürzeste Lauf sollte mindestens 50 Sekunden rechnen; wählen Sie geeignete Parameter aus!

Schreiben Sie ein halbe Seite Interpretation zu diesen Ergebnissen. Wiederholen Sie jede Messung mindestens 3 Mal, um aussagekräftige Mittelwerte bilden zu können.

Hinweis: Es ist empfehlenswert die Störfunktion $f(x,y) = 2\pi^2 \sin(\pi x) \sin(\pi y)$ zu verwenden, da der erhöhte Rechenaufwand das Skalierungsverhalten verbessert.

Abgabe des Programms

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, in einem Verzeichnis `pde`; **gut** dokumentiert! (Kommentare bei geänderten Code-Teilen!)
  - Erwartet werden die Dateien `Makefile`, `askparams.c`, `partdiff-par.c` und `partdiff-par.h`.
  - **Optional:** Eine Ausarbeitung `leistungsanalyse.pdf` mit den ermittelten Laufzeiten und der Leistungsanalyse.
- 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 `partdiff-par` erzeugen.
  - **Optional:** Ein Target `partdiff-par-hybrid` für die Binärdatei `partdiff-par-hybrid`, welche die Hybrid-Parallelisierung umsetzt.
- **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!