

Prüfungsprotokoll Mathematik

Fach: Scientific Computing
Studiengang: Scientific Computing

☐ Bachelor
☒ Master

(Sonstiges bitte von Hand eintragen.)

Prüfer/in: Prof. Schwandt
Datum: November 2016
Prüfungsdauer: 30 Min.

Beisitzer/in: ?
Note: 2,0
Anzahl der Kandidaten: 1

Vorbereitungszeit: 2 Wochen

Literatur: Parallele Numerik - Eine Einführung
Hartmut Schwandt (2003)

Beurteilung der Prüfung und des/r Prüfers/in:

Entspannte Prüfungsumgebung

Prüfer stellt viele Zusatzfragen, besonders zur praktischen Umsetzung und Problemen

Fragen:

- Rechnerarchitekturen: Welche haben wir kennen gelernt, was sind die Unterschiede? Welche werden in der Praxis genutzt? Wie nutzt man konkret einen Cluster? Welche Probleme treten dabei auf?
- Was ist die Taktfrequenz, allgemein und speziell?
- Gauß-Algorithmus: Vorgehen, Komplexität, Warum lässt er sich schlecht parallelisieren? Wann ist Pivotisierung nötig? Welche Matrizen müssen nicht pivotisiert werden?
- Zyklische Reduktion: Vorkommen, kurz Methode, vor allem Probleme: welche und warum
- Binerman-Algorithmus: Wie löst es Probleme der Blockzyklischen Reduktion? Faktorisierung der Matrizen, warum und für welche Matrixklassen funktioniert das?
- Eigenwert-Eigenvektor-Zerlegung: Wie funktioniert die formale Umordnung? Wie berechnet man Q.H. effizient?
- FACR(2)-Algorithmus: nur Grundidee

Prüfungsprotokoll Mathematik

Fach: Wissenschaftliches Rechnen
Studiengang: Scientific Computing

☐ Vordiplom
☐ Zwischenprüfung
☐ Diplom

☐ Bachelor
☒ Master

Prüfer/in: Prof. Dr. Hartmut Schwandt
Datum: April 2014
Prüfungsdauer: 60 Minuten

Beisitzer/in:
Note: 1,7
Anzahl der Kandidaten: 1

Vorbereitungszeit:
14 Tage intensive Vorbereitung

Literatur:
Parallele Numerik - Eine Einführung (Hartmut Schwandt, 2003)

Beurteilung der Prüfung und des/r Prüfers/in:

Es herrschte eine entspannte Atmosphäre. Man hat viel Zeit zur Verfügung, um die Fragen zu beantworten und kann ruhig auch mal nachdenken.

Fragen:

Zum Beginn wird eine leichte Einstiegsfrage gestellt. Zu allen Antworten die man gibt, kommen immer weitere Fragen, die dann detaillierter werden. Die Prüfung ist ein Dialog und man kann nicht einfach nur gelerntes Wissen wiedergeben. Wenn man etwas nicht weiß, dann wird darüber trotzdem weiter geredet, bis einem etwas einfällt.

Rechnerarchitekturen:

- Welche Rechnerarchitekturen haben wir kennengelernt? Erläutern Sie auch die Unterschiede. → Vektorprozessor, Mikroprozessor, Parallelrechner mit verteiltem und geteiltem Speicher.
- Verteilter Speicher und geteilter Speicher:
 - Wie ist das mit der Arbeits- und Datenverteilung?
 - Muss in einem Programm direkt die Schleife angepasst oder nur die Daten anders verteilt werden?
- Was sind Probleme beim Zugriff auf den Hauptspeicher bei einem Vektorprozessor?
→ Speicherbankkonflikte.

Lineare Gleichungssysteme:

- Gaußalgorithmus:
 - Was sind die Nachteile?
 - Was ist Pivotsuche?
 - Wann muss man eine Pivotsuche durchführen?
 - Übertragen sich beim Gaußalgorithmus die Eigenschaften der ursprünglichen Matrix auf die Restmatrizen von Schritt zu Schritt?
 - Wie groß ist die arithmetische Komplexität?
 - Kann man den Gaußalgorithmus gut parallelisieren?

- Blocktridiagonalmatrizen:

- Was passiert, wenn man auf diese besondere Form von Matrizen den Gaußalgorithmus anwendet?
- Wie sieht die allgemeine Form solcher Matrizen aus?
- Wie kann man diese Matrizen weiter vereinfachen in den meisten Anwendungen?
→ Die Diagonalblöcke sind selbst Tridiagonalmatrizen.
- Wo werden solche Matrizen angewendet? → Finite Differenzen.
- Was muss gelten, damit die Blöcke so gleichmäßig strukturiert sind und auch immer die gleiche Größe haben? → Rechteckgebiet und gleichmäßiger Gitterabstand.

- Schnelle Löser:

- Wie sind schnelle Löser definiert? → Sie sind zum Lösen von LGS mit Blocktridiagonalmatrizen geeignet und haben eine bestimmte arithmetische Komplexität.
- Wie groß ist diese arithmetische Komplexität im Vergleich zum Blockgaußalgorithmus?

- Blockzyklische Reduktion:

- Was ist die blockzyklische Reduktion?
- Wie werden die Gleichungen entkoppelt? → Das sollte ich auch ausführlich aufschreiben.
- Was sind die Probleme bei der blockzyklischen Reduktion? → Durch Matrixmultiplikationen in den Reduktionsschritten werden die Werte in den Matrizen sehr groß. Dadurch entstehen Auslöschungseffekte und die Werte können auch so groß werden, dass diese gar nicht mehr richtig abgespeichert werden können. Außerdem bleibt die Struktur der Diagonalblöcke nicht erhalten.

- Buneman-Algorithmus:

- Was macht der Buneman-Algorithmus besser als die blockzyklischen Reduktion?
- Können Sie die Schritte erläutern, die dort durchgeführt werden? → Auch hier sollte ich das alles ausführlich aufschreiben und die Faktorisierung der Matrizen erklären. Dazu sollte ich die Tschebyschow-Polynome verwenden.
- Lässt sich der Buneman-Algorithmus parallelisieren und vektorisieren?