|  |  |
| --- | --- |
|  | tud_logo |
|  | Hochleistungssimulation im Ingenieurwesen 3. Hausübung |
|  |  |
|  |  |
|  | Xiangjun Wei 2594637 Bauingenieurwesen M.Sc  Guanlin Wang 2872498 Bauingenieurwesen M.Sc  Wintersemester 2018/19  07. Feb 2019 |
|  |  |

Inhaltsverzeichnis

[1. Einleitung 3](#_Toc389534)

[2. PCG-Solver Programmen 3](#_Toc389535)

[2.1. Sequentiell 3](#_Toc389536)

[2.2. Parallel 3](#_Toc389537)

[3. Ergebnisse 3](#_Toc389539)

[3.1. Sequentielles Ergebnis 3](#_Toc389540)

[3.2. Parallelisiertes Ergebnis 3](#_Toc389541)

[4. Vergleichen 3](#_Toc389542)

[5. Aufgetretene Probleme 4](#_Toc389543)

# Einleitung

In der dritten Übung programmieren wir für PCG-Solver in sequentiellen und parallelen Form. Dann geben wir die Ergebnisse, und zwar Anzahl der Iterationen, Wert des Abbruchkriteriums und Ergebnisvektor, von den beiden Programmierung, um zu vergleichen.

# PCG-Solver Programmen

## Sequentiell

Wir geben die Initialisierung zuerst, dann mit dem PCG-Solver Algorithmus bekommen die Ergebnisse. Und das größte Abbruchkriterium ist 10E-7.

## Parallel

Für die Parallele Programmierung machen zuerst die Verteilung der Matrizen. Wir benutzen nur 2 Prozessoren. D.h. es gibt insgesamt zwei 6\*6 Matrizen von K, und zwei 6\*1 Vektoren von f. Außerdem sind die Anzahl der überlappenden Teile der beiden Matrizen alle durch zwei geteilt. Dann schreiben wir die Codes mit dem Formel 2(in Hausübungaufgabeblätter). Das Abbruchkriterium ist gleich wie in Sequentiell.

# Ergebnisse

Ergebnisvektor: [0.0, 0.0625, 0.1875, 0.0, 0.28125, 0.5625, 0.0, 0.5, 1.0]

## Sequentielles Ergebnis

Anzahl der Iteration: 3

Wert des Abbruchkriteriums: 8.7871E-34

Ergebnisvektor: [0.0, 0.06250000000000003, 0.1875, 0.0, 0.28125, 0.5625, 0.0, 0.5, 1.0]

## Parallelisiertes Ergebnis

Anzahl der Iteration: 3

Wert des Abbruchkriteriums: 7.9746E-34

Ergebnisvektor: [0.0, 0.062499999999999986, 0.18749999999999997, 0.0, 0.28125, 0.5625, 0.0, 0.5, 1.0]

# Vergleichen

Die Anzahl der Iteration von den beiden Programmierungen sind gleich.

Aber das Abbruchkriterium von parallelen Programmierung ist kleiner als sequentiellen Programmierung. Meiner Meinung nach ist das wegen, dass für die parallele Programmierung jeder Prozessor kleinere Matrix berechnet, und alle Prozessoren gleichzeitig das Abbruchkriterium erreichen müssen.

Und der Ergebnisvektor von sequentiell ist exakter als parallel. Es gibt eine Abweichung, wenn der Computer die Methode „Allreduce“ durchführt.

# Aufgetretene Probleme

Zuerst können wir nicht aus der Schleife springen. Dann machen wir mit „break“, um die Schleife zu springen. Dann unsere Methode für Multiplizierung der Matrizen ist nicht exakt. Es gibt einige Fehler darin.

Weil die parallele Programmierung ähnlich wie die erste Hausübung ist, machen wir flüssig in dem zweiten Teil.