Übung 4

Aufgabe 4.1

Die Gesamtzeit des sequentiellen Programms beträgt T(1) = 11,1s.

7,8s werden für die Matrixmultiplikation benötigt, welche parallelisiert werden kann(Tp).

3,3s werden für die Ein- und Ausgabe gebraucht (Ts). Diese muss sequentiell erfolgen.

Für n gegen Unendlich geht T(n) gegen 3,3s.

Der maximale Speedup beträgt somit also S(n) = T(1)/T(n) = 11,1s/3,3s = 3,36.

Aufgabe 4.2

Speedup S(n) = T(1)/T(n)

T(n) = 5,55s

Tp = 7,8s

Ts = 3,3s

**Speedup 2**:

S(n) = 11,1s / T(n) = 2 🡪 Die parallele Gesamtprogrammlaufzeit müsste

T(n) = Ts + (Tp/n)

5,55s = 3,3s + (7,8s/n) 🡪 n = 3,467

Es müssen mindestens 4 Prozesse verwendet werden um einen Speedup von 2 zu erreichen.

**Speedup 3**:

S(n) = 11,1s / T(n) = 3 🡪 T(n) = 3,7s

3,7s = 3,3s + (7,8s/n) 🡪 n = 19,5

Es müssen mindestens 20 Prozesse verwendet werden um einen Speedup von 3 zu erreichen.

**Speedup 10**:

Es ist nicht möglich einen Speedup von 10 zu erreichen, da der sequentielle Anteil zu groß ist und ein maximaler Speedup von 3,36 zu erreichen ist.

Aufgabe 4.3.2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **OPENMP** |  |  |  |  |  |
|  | Threads1 | Threads2 | Threads4 | Threads8 | Threads16 |
| Iteration 1 | 20,37 | 10,18 | 5,11 | 2,57 | 2,62 |
| Iteration 2 | 20,37 | 10,18 | 5,1 | 2,58 | 2,79 |
| Iteration 3 | 23,38 | 10,18 | 5,55 | 2,56 | 2,68 |
| Iteration 4 | 20,37 | 10,18 | 5,1 | 2,56 | 2,75 |
| Iteration 5 | 20,9 | 10,18 | 5,09 | 2,56 | 2,72 |
| Average | 21,078 | 10,18 | 5,19 | 2,566 | 2,712 |
| Standard Deviation | 1,307161046 | 0 | 0,2013703057 | 0,00894427191 | 0,06534523701 |

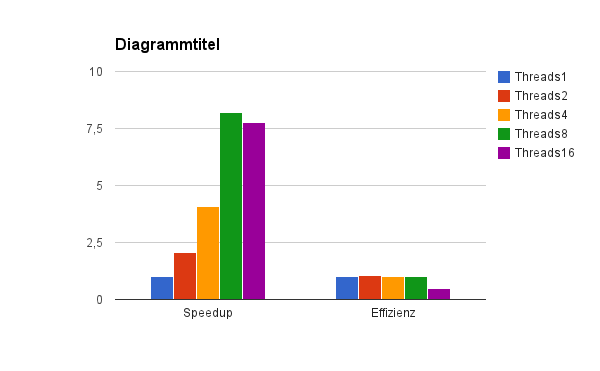
Aufgabe 4.3.4 + 4.3.5

Speedup S(n) = T(1)/T(n)

Effizienz E(n) = S(n) / n

Karp-Flatt metric e(n) = (((1/S(n)) – 1/n) / (1-(1/n))

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **OPENMP** |  |  |  |  |  |
| Threads | Threads1 | Threads2 | Threads4 | Threads8 | Threads16 |
| Speedup | 1 | 2,070530452 | 4,061271676 | 8,214341387 | 7,772123894 |
| Effizienz | 1 | 1,035265226 | 1,015317919 | 1,026792673 | 0,4857577434 |
| Karp-Flatt metric |  | -0,03406395294 | -0,005028940127 | -0,003727651038 | 0,07057595597 |



Aufgabe 4.3.6

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **OPENMP** |  |  |  |  |  |
|  | Threads1 | Threads2 | Threads4 | Threads8 | Threads16 |
| Speedup | 1 | 2,070530452 | 4,061271676 | 8,214341387 | 7,772123894 |
| linearer Speedup | 1 | 2 | 4 | 8 | 16 |

Für die Threads 2-8 skaliert das Programm nahezu linear. Bei 16 Threads wir die Ausführung wieder langsamer, da auf einem Zuseknoten nur 8 CPUs vorhanden und sich somit mehrere Threads eine CPU teilen müssen. Das Scheduling führt zu der höheren Programmzeit im Vergleich zu 8 Threads.