

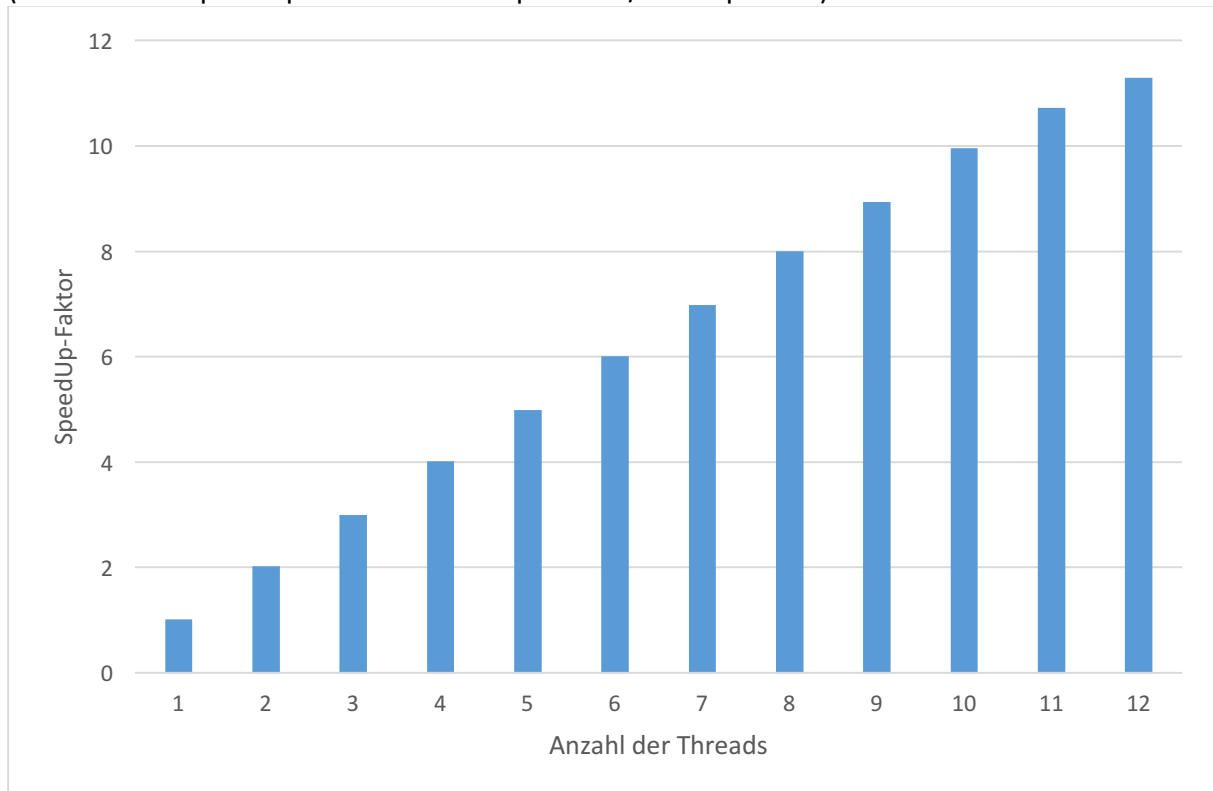
## Leistungsanalyse

Messung 1:

Aufruf: `./partdiff-openmp 12 2 512 2 2 1200`

**Laufzeit des Programmes bei verschiedenen Thread-Anzahlen:**

(Verwendete SpeedUp-Formel:  $\text{Zeit Sequentiell} / \text{Zeit OpenMP}$ )



Bei der Analyse des Graphen zeigt sich, dass die OpenMP-Version des Programmes mit einem Thread ein wenig schneller läuft als die sequentielle Version. Wenn man einen zweiten Thread hinzunimmt, läuft das Programm mehr als doppelt so schnell wie das Sequentielle. Bei der Ausführung mit 6 Threads wird der SpeedUp-Faktor 6,01 erreicht und in der finalen Ausführung mit 12 Threads erreicht der Wert 11,29.

Der SpeedUp-Faktor nimmt zu Beginn linear zu. Ab dem 9 Thread nimmt die Zunahme des Faktors etwas ab, jedoch ist die Zunahme auf jeden Fall noch ersichtlich.

**Tabelle: Laufzeit und SpeedUp-Faktor**

Programm-Version	Laufzeit (s)	SpeedUp-Faktor
Sequentiell	671,254	
OpenMP: 1 Thread	664,454	1,01
OpenMP: 2 Threads	332,406	2,02
OpenMP: 3 Threads	224,246	2,99
OpenMP: 4 Threads	167,277	4,01
OpenMP: 5 Threads	134,607	4,99
OpenMP: 6 Threads	111,667	6,01
OpenMP: 7 Threads	96,077	6,99
OpenMP: 8 Threads	83,937	8,00
OpenMP: 9 Threads	75,119	8,94
OpenMP: 10 Threads	67,388	9,96
OpenMP: 11 Threads	62,588	10,72
OpenMP; 12 Threads	59,443	11,29

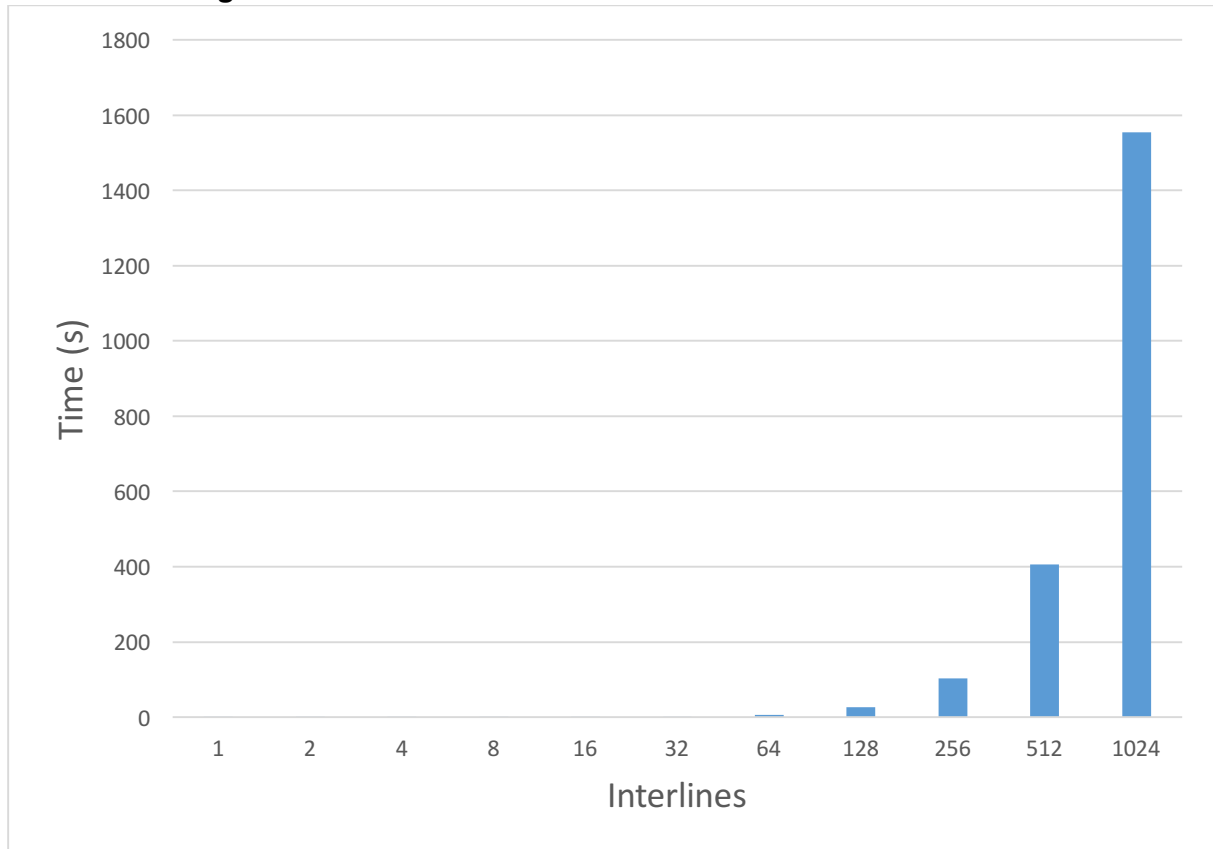
**Messdaten:**

1 Thread(s): 664.454 s  
2 Thread(s): 332.406 s  
3 Thread(s): 224.246 s  
4 Thread(s): 167.277 s  
5 Thread(s): 134.607 s  
6 Thread(s): 111.667 s  
7 Thread(s): 96.077 s  
8 Thread(s): 83.937 s  
9 Thread(s): 75.119 s  
10 Thread(s): 67.388 s  
11 Thread(s): 62.588 s  
12 Thread(s): 59.443 s

## Messung 2:

Aufruf: `./partdiff-openmp 12 2 2i 2 2 8192 (0 ≤ i ≤ 10)`

## Laufzeit des Programmes bei verschiedenen Interlines:



Der Interlines Parameter hat einen großen Einfluss auf die Laufzeit des Programmes. Bei kleinen Werten von Interlines macht sich das noch nicht bemerkbar, jedoch nimmt die Bedeutung mit dem Ansteigen zu.

Bei dem Sprung von 1 auf 2 Interlines steigt die Laufzeit um 5,8% und bei dem Sprung von 32 auf 64 Interlines bereits um 261,07%. Den höchsten Steigerungswert konnten wir bei dem Übergang von 64 auf 128 Interlines feststellen, dieser beträgt 325,56%. Bei den folgenden Übergängen liegt die Steigerung im Bereich von 279,4% bis 293,38%.

Es gibt also ab 128 Interlines eine relativ gleichbleibende Steigerung in der Laufzeit. Wir vermuten, dass das Erzeugen der Matrix ab dem Zeitpunkt so effektiv aufgeteilt ist, dass das Verdoppeln des Interlines-Wertes eine konstante prozentuale Laufzeiterhöhung mit sich bringt.

**Tabelle: Mittelwerte und Prozentuale Steigerung**

Interlines	Mittelwerte (s)	Prozentuale Steigerung zum vorherigen Wert (in %)
1	0,051	
2	0,054	4,55
4	0,091	69,57
8	0,231	153,48
16	0,823	256,94
32	1,767	114,66
64	6,381	261,07
128	27,157	325,56
256	103,032	279,40
512	405,305	293,38
1024	1553,657	283,33

**Messdaten:**

Interlines = 1:

1. Lauf: 0.054 s
2. Lauf: 0.050 s
3. Lauf: 0.050 s

Interlines = 2:

1. Lauf: 0.056 s
2. Lauf: 0.050 s
3. Lauf: 0.055 s

Interlines = 4:

1. Lauf: 0.091 s
2. Lauf: 0.080 s
3. Lauf: 0.102 s

Interlines = 8:

1. Lauf: 0.215 s
2. Lauf: 0.259 s
3. Lauf: 0.218 s

Interlines = 16:

1. Lauf: 0.526 s
2. Lauf: 0.574 s
3. Lauf: 1.370 s

Interlines = 32:

1. Lauf: 1.707 s
2. Lauf: 1.780 s
3. Lauf: 1.815 s

Interlines = 64:

1. Lauf: 6.436 s
2. Lauf: 6.407 s
3. Lauf: 6.301 s

Interlines = 128:

- 1. Lauf: 30.358 s
- 2. Lauf: 26.770 s
- 3. Lauf: 24.342 s

Interlines = 256:

- 1. Lauf: 101.848 s
- 2. Lauf: 104.718 s
- 3. Lauf: 102.529 s

Interlines = 512:

- 1. Lauf: 388.247 s
- 2. Lauf: 425.048 s
- 3. Lauf: 402.619 s

Interlines = 1024:

- 1. Lauf: 1555.951 s
- 2. Lauf: 1553.574 s
- 3. Lauf: 1551.445 s