Sreepathy Jayanand

CED17I038

High Performance Computing Lab - 9

Analysis of parallelization factor

No of Grids used = 1

No of blocks used per grid = 1

No of threads per block = 1, 2, 4, 8, 16, 32, 64, 128, 256, 500

Problem statement 1: Find sum of array

Input size = 10 ^ 8 (n)

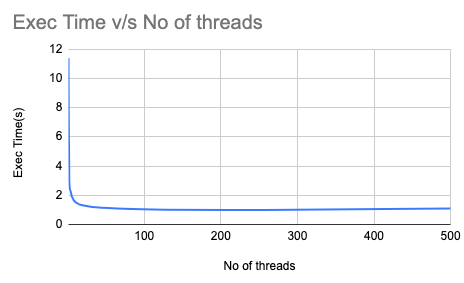
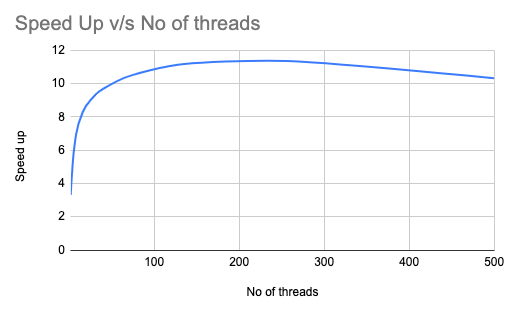
Segmentation fault = 2 \* 10 ^ 8

Let the number of threads = t

Strategy: Each t threads get ‘n / t’ elements to calculate the sum of. Each thread calculates the sum and puts the value to ‘partialsum’ array.

We then synchronize the threads so that all the results are calculated before the next step, and then the addition of the partial sum array is performed by the thread 0 and the result is returned.

| No of threads | Exec time (s) |
| --- | --- |
| 1 | 11.36 |
| 2 | 3.43 |
| 4 | 2.25 |
| 8 | 1.64 |
| 16 | 1.37 |
| 32 | 1.21 |
| 64 | 1.10 |
| 128 | 1.02 |
| 256 | 1.00 |
| 500 | 1.10 |



The most efficient thread size for this program is 256

According to Amdahl’s law :

Speed up = 1/((1-p) + p/N)

Or

Parallelization factor = (N - N/S.U)/(N - 1)

So here N = 256;

Speed Up = 11.36/1.00 = 11.36

PF = (256 - 256/11.36)/255 = 0.91

Problem statement 2: Find dot product of 2 arrays

Input size = 10 ^ 8 (n)

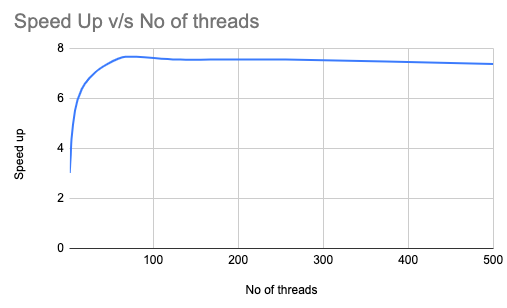
Segmentation fault = 2 \* 10 ^ 8

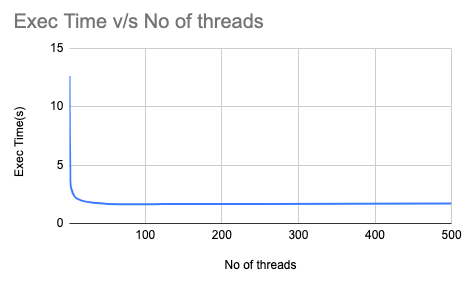
Let the number of threads = t

Strategy: Each t threads get ‘n / t’ elements to calculate the product of elements and sum them. Each thread calculates the sum and puts the value to ‘partialsum’ array.

We then synchronize the threads so that all the results are calculated before the next step and then the addition of the partial sum array is performed by the thread 0 and the result is returned.

| No of threads | Exec time (s) |
| --- | --- |
| 1 | 12.61 |
| 2 | 4.19 |
| 4 | 2.89 |
| 8 | 2.29 |
| 16 | 1.98 |
| 32 | 1.79 |
| 64 | 1.65 |
| 128 | 1.67 |
| 256 | 1.67 |
| 500 | 1.71 |





The most efficient thread size for this program is 64

According to Amdahl’s law :

Speed up = 1/((1-p) + p/N)

Or

Parallelization factor = (N - N/S.U)/(N - 1)

So here N = 64;

Speed Up = 12.61/1.71 = 7.36

PF = (64 - 64/7.36)/63 = 0.87