

Project # 1

Mazen Alotaibi

Tell what machine you ran this on

I am running on the ENGR Server.

To compile program, run `./runPro`

I have made the **number of tries** to be fixed at **100**.

Check the end of the document for the actual output from running the program.

What do you think the actual volume is?

Estimation of the actual volume from running the program:

NUMT / NUMNODES	10	20	40	80	160	320	640	1280
1	26.77	26.12	25.73	25.53	25.42	25.37	25.34	25.33
2	26.77	26.12	25.73	25.53	25.42	25.37	25.34	25.33
4	26.77	26.12	25.73	25.53	25.42	25.37	25.34	25.33

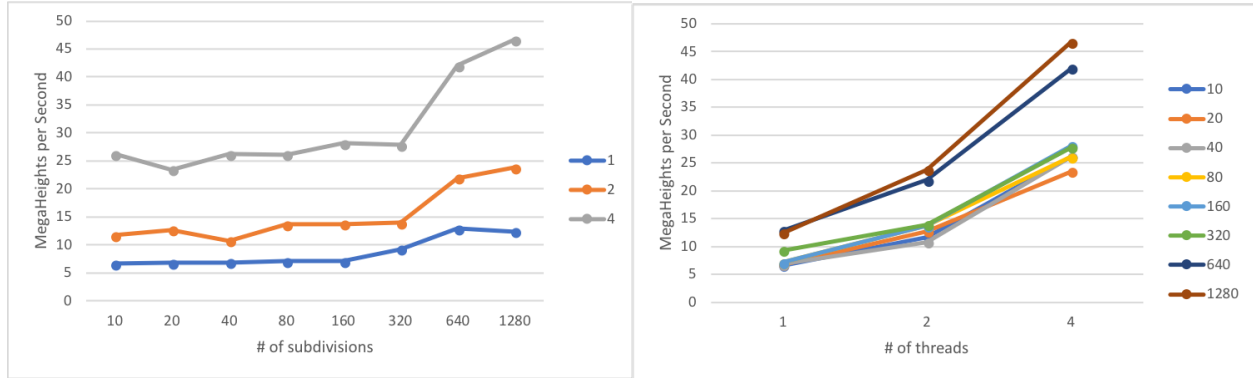
As the number of subdivision increases the approximation for the Trapezoidal rule increases, so I would assume that the actual volume is 25.33.

Show the performances you achieved in tables and graphs as a function of NUMNODES and NUMT

Performances Table:

NUMT / NUMNODES	10	20	40	80	160	320	640	1280
1	6.62	6.8	6.84	7.1	7.11	9.28	12.89	12.41
2	11.71	12.71	10.77	13.69	13.74	13.92	21.92	23.82
4	26.21	23.48	26.18	26.14	28.15	27.82	42.02	46.7

Performances Graph:



What patterns are you seeing in the speeds?

It seems that as the number of subdivisions increases the speeds increases for all threads. In addition, the speeds for 4 threads is greater than 2 threads and 1 thread, and the speeds for 2 threads is greater than 1 thread.

$T_i = \frac{\sum_{j=1}^s t_j}{s}$, where T_i is the average of all subdivision time, t , for a certain i number of threads.

$$S_n = \frac{T_1}{T_n}$$

$$S_2 = \frac{T_1}{T_2} = 1.83$$

$$S_4 = \frac{T_1}{T_4} = 3.55$$

Why do you think it is behaving this way?

The speedup ratio is greater than 1 due to the average elapsed time when using one thread is greater than the average elapsed time when using two or four threads, which means that **when using two or four threads it takes less time to process compare to using one thread to do the same task**. In addition, the speedup when using 4 threads is greater than the speedup when using 2 threads because, when using 4 threads, it takes less time to process same task compared to the timing when using 2 threads.

What is the Parallel Fraction for this application, using the Inverse Amdahl equation?

$$F_n = \frac{n}{(n-1)}(1 - \frac{1}{S_n})$$

$$F_2 = \frac{2}{(1)}(1 - \frac{1}{1.83}) = 0.90$$

$$F_4 = \frac{4}{(3)}(1 - \frac{1}{3.55}) = 0.96$$

$$F = \frac{F_2 + F_4}{2} = 0.93$$

The Parallel Fraction is 0.93.

Given that Parallel Fraction, what is the maximum speed-up you could ever get?

$$(S_{max}) = \frac{1}{(1-F)}$$

$$(S_{max}) = \frac{1}{(1-0.93)} \approx 14$$

The maximum speed-up is ~14.

Output files

output.txt:

```
## _NUMT(Number of Threads) = 1
## |
## .---_NUMS(Number of subdivisions) = 10
##   Peak Performance =      6.69 MegaHeights/Sec
## Average Performance =      6.62 MegaHeights/Sec
## Average Elapsed Time =   15.20 microseconds
## |
## .---_NUMS(Number of subdivisions) = 20
##   Peak Performance =      6.85 MegaHeights/Sec
## Average Performance =      6.80 MegaHeights/Sec
## Average Elapsed Time =   58.87 microseconds
## |
## .---_NUMS(Number of subdivisions) = 40
##   Peak Performance =      6.90 MegaHeights/Sec
## Average Performance =      6.84 MegaHeights/Sec
## Average Elapsed Time =  233.99 microseconds
## |
## .---_NUMS(Number of subdivisions) = 80
##   Peak Performance =      7.14 MegaHeights/Sec
## Average Performance =      7.10 MegaHeights/Sec
## Average Elapsed Time =  901.21 microseconds
## |
## .---_NUMS(Number of subdivisions) = 160
##   Peak Performance =      7.13 MegaHeights/Sec
## Average Performance =      7.11 MegaHeights/Sec
## Average Elapsed Time = 3599.12 microseconds
## |
## .---_NUMS(Number of subdivisions) = 320
##   Peak Performance =     13.69 MegaHeights/Sec
## Average Performance =      9.28 MegaHeights/Sec
## Average Elapsed Time = 12089.73 microseconds
## |
## .---_NUMS(Number of subdivisions) = 640
##   Peak Performance =     13.69 MegaHeights/Sec
## Average Performance =     12.89 MegaHeights/Sec
## Average Elapsed Time = 32931.59 microseconds
## |
## .---_NUMS(Number of subdivisions) = 1280
##   Peak Performance =     13.27 MegaHeights/Sec
## Average Performance =     12.41 MegaHeights/Sec
```

```

## Average Elapsed Time = 136234.50 microseconds
## -----
## _NUMT(Number of Threads) = 2
## |
## .---_NUMS(Number of subdivisions) = 10
##     Peak Performance =    12.14 MegaHeights/Sec
## Average Performance =    11.71 MegaHeights/Sec
## Average Elapsed Time =    10.38 microseconds
## |
## .---_NUMS(Number of subdivisions) = 20
##     Peak Performance =    13.20 MegaHeights/Sec
## Average Performance =    12.71 MegaHeights/Sec
## Average Elapsed Time =    32.90 microseconds
## |
## .---_NUMS(Number of subdivisions) = 40
##     Peak Performance =    13.71 MegaHeights/Sec
## Average Performance =    10.77 MegaHeights/Sec
## Average Elapsed Time =   153.28 microseconds
## |
## .---_NUMS(Number of subdivisions) = 80
##     Peak Performance =    14.25 MegaHeights/Sec
## Average Performance =    13.69 MegaHeights/Sec
## Average Elapsed Time =   469.20 microseconds
## |
## .---_NUMS(Number of subdivisions) = 160
##     Peak Performance =    14.26 MegaHeights/Sec
## Average Performance =    13.74 MegaHeights/Sec
## Average Elapsed Time =  1884.28 microseconds
## |
## .---_NUMS(Number of subdivisions) = 320
##     Peak Performance =    14.26 MegaHeights/Sec
## Average Performance =    13.92 MegaHeights/Sec
## Average Elapsed Time =  7389.82 microseconds
## |
## .---_NUMS(Number of subdivisions) = 640
##     Peak Performance =    27.31 MegaHeights/Sec
## Average Performance =    21.92 MegaHeights/Sec
## Average Elapsed Time = 20448.96 microseconds
## |
## .---_NUMS(Number of subdivisions) = 1280
##     Peak Performance =    26.06 MegaHeights/Sec
## Average Performance =    23.82 MegaHeights/Sec
## Average Elapsed Time = 71530.82 microseconds
## -----
## _NUMT(Number of Threads) = 4
## |
## .---_NUMS(Number of subdivisions) = 10
##     Peak Performance =    29.61 MegaHeights/Sec
## Average Performance =    26.21 MegaHeights/Sec
## Average Elapsed Time =     5.03 microseconds
## |
## .---_NUMS(Number of subdivisions) = 20
##     Peak Performance =    24.51 MegaHeights/Sec
## Average Performance =    23.48 MegaHeights/Sec

```

```

## Average Elapsed Time = 18.50 microseconds
## |
## .---_NUMS(Number of subdivisions) = 40
##     Peak Performance = 26.84 MegaHeights/Sec
## Average Performance = 26.18 MegaHeights/Sec
## Average Elapsed Time = 62.48 microseconds
## |
## .---_NUMS(Number of subdivisions) = 80
##     Peak Performance = 28.34 MegaHeights/Sec
## Average Performance = 26.14 MegaHeights/Sec
## Average Elapsed Time = 251.37 microseconds
## |
## .---_NUMS(Number of subdivisions) = 160
##     Peak Performance = 28.52 MegaHeights/Sec
## Average Performance = 28.15 MegaHeights/Sec
## Average Elapsed Time = 909.95 microseconds
## |
## .---_NUMS(Number of subdivisions) = 320
##     Peak Performance = 28.48 MegaHeights/Sec
## Average Performance = 27.82 MegaHeights/Sec
## Average Elapsed Time = 3681.06 microseconds
## |
## .---_NUMS(Number of subdivisions) = 640
##     Peak Performance = 52.34 MegaHeights/Sec
## Average Performance = 42.02 MegaHeights/Sec
## Average Elapsed Time = 10670.42 microseconds
## |
## .---_NUMS(Number of subdivisions) = 1280
##     Peak Performance = 50.68 MegaHeights/Sec
## Average Performance = 46.70 MegaHeights/Sec
## Average Elapsed Time = 36847.66 microseconds
## -----
##           Speedup-2 = 1.83
##           Speedup-4 = 3.55
## Parallel Fraction-2 = 0.90
## Parallel Fraction-3 = 0.96

```

data1.csv

```

## Estimated Volume, Max. MegaHeights per Seconds, Avg. MegaHeights per Second, and Avg. Time
## 26.77, 6.69, 6.62, 15.20
## 26.12, 6.85, 6.80, 58.87
## 25.73, 6.90, 6.84, 233.99
## 25.53, 7.14, 7.10, 901.21
## 25.42, 7.13, 7.11, 3599.12
## 25.37, 13.69, 9.28, 12089.73
## 25.34, 13.69, 12.89, 32931.59
## 25.33, 13.27, 12.41, 136234.50
## 26.77, 12.14, 11.71, 10.38
## 26.12, 13.20, 12.71, 32.90
## 25.73, 13.71, 10.77, 153.28
## 25.53, 14.25, 13.69, 469.20
## 25.42, 14.26, 13.74, 1884.28
## 25.37, 14.26, 13.92, 7389.82
## 25.34, 27.31, 21.92, 20448.96

```

```
## 25.33, 26.06, 23.82, 71530.82
## 26.77, 29.61, 26.21, 5.03
## 26.12, 24.51, 23.48, 18.50
## 25.73, 26.84, 26.18, 62.48
## 25.53, 28.34, 26.14, 251.37
## 25.42, 28.52, 28.15, 909.95
## 25.37, 28.48, 27.82, 3681.06
## 25.34, 52.34, 42.02, 10670.42
## 25.33, 50.68, 46.70, 36847.66
## 1.83, 0.90
## 3.55, 0.96
## S_n and F_n
```