Assignment 4

Parallel Computing

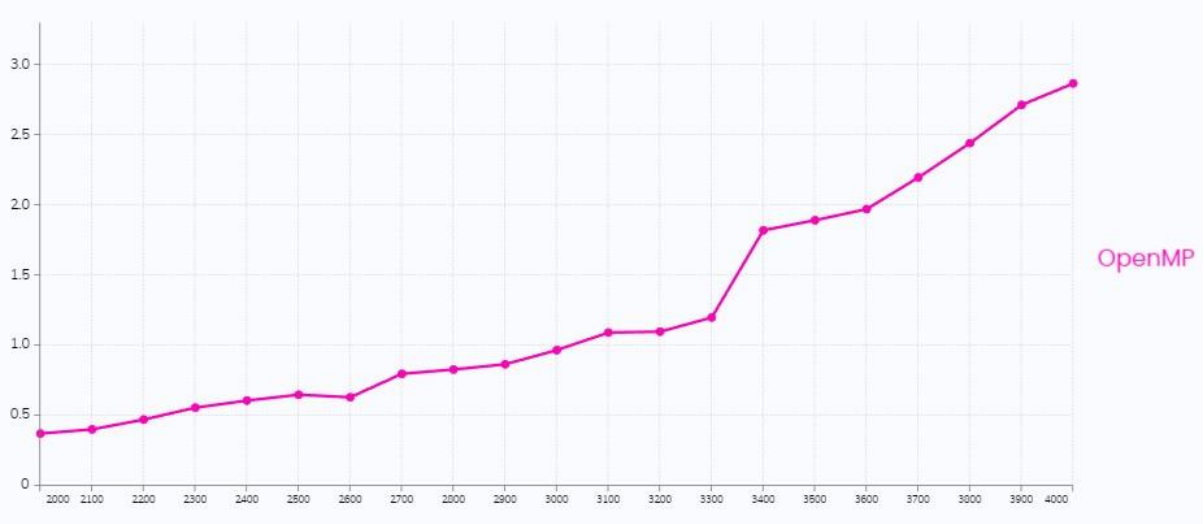
Ross Murphy

20207271

1) The dependence of the execution time of the program on the matrix size **n**

The graph below shows the speedup of the program against **n** which ranges from 2000 to 4000 and iterates by 100. The graph below uses no schedule type and 8 as the number of threads.

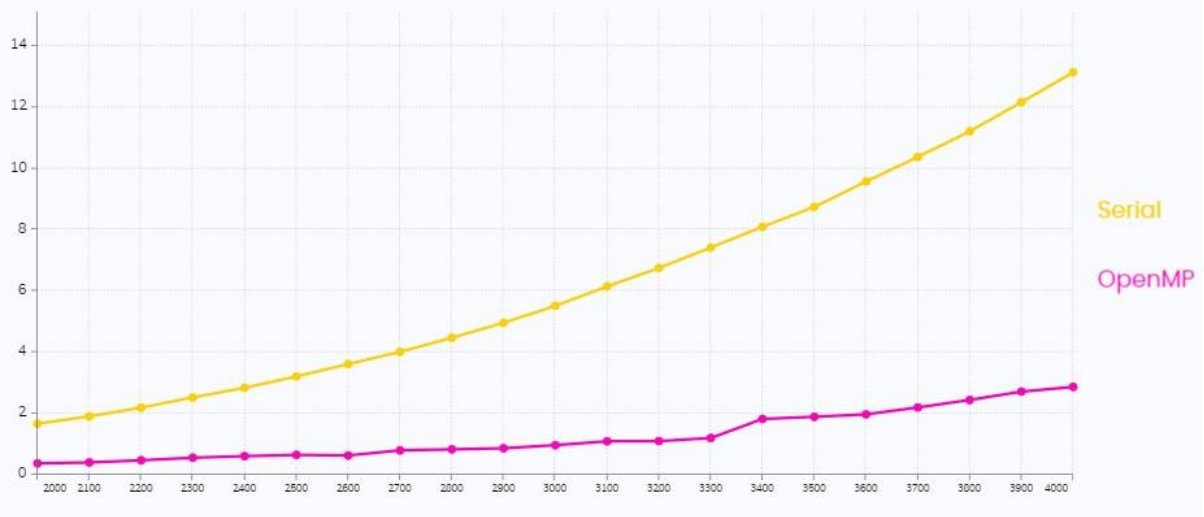
x-axis = N, y-axis=time



2) The speedup over a serial counterpart of the program

For the serial counterpart I used the regular BLAS multiplication without OpenMP. We can see from the graph below a dramatic speedup in using OpenMP.

x-axis = N, y-axis=time



3) The dependence of the execution time of the program on chunk size in static, dynamic and guided. Also, compare the best chunk sizes.

From the screenshots below, which were retrieved from the output of the program, we can see that 1 was the optimum chunk size for all schedule types. I compared chunk size 1, 2 and 4 and the time increased as the chunk size did.

Static:

Chunk size 1 Chunk size 2 Chunk size 4

N	parallel time
2000	0.440798
2100	0.516695
2200	0.579655
2300	0.681362
2400	0.742078
2500	0.848459
2600	0.950688
2700	1.053755
2800	1.223471
2900	1.324883
3000	1.453366
3100	1.600210
3200	1.752588
3300	1.940983
3400	2.144231
3500	2.270114
3600	2.530376
3700	2.731093
3800	2.945127
3900	3.187290
4000	3.538884

N	parallel time
2000	0.886380
2100	1.007210
2200	1.163746
2300	1.326333
2400	1.505357
2500	1.681158
2600	1.892478
2700	2.124420
2800	2.387423
2900	2.614791
3000	3.044255
3100	3.204894
3200	3.475024
3300	3.822162
3400	4.208147
3500	4.585136
3600	4.936039
3700	5.335634
3800	5.794250
3900	6.273131
4000	6.703651

N	parallel time
2000	0.368587
2100	0.359284
2200	0.438576
2300	0.419832
2400	0.511264
2500	0.603983
2600	0.648576
2700	0.727735
2800	0.812017
2900	0.890366
3000	0.905778
3100	0.991189
3200	1.141480
3300	1.286969
3400	1.332358
3500	1.420302
3600	1.739361
3700	1.635966
3800	1.831543
3900	1.925538
4000	2.086016

Dynamic:

Chunk size 1 Chunk size 2 Chunk size 4

N	parallel time
2000	0.341547
2100	0.394518
2200	0.428911
2300	0.517010
2400	0.574660
2500	0.644695
2600	0.692512
2700	0.701840
2800	0.775754
2900	0.846463
3000	0.900559
3100	1.093738
3200	1.127701
3300	1.252929
3400	1.406983
3500	1.433521
3600	1.597227
3700	1.722779
3800	1.776136
3900	1.988754
4000	2.321589

N	parallel time
2000	0.453479
2100	0.512190
2200	0.586556
2300	0.702371
2400	0.770666
2500	0.860602
2600	0.986454
2700	1.075892
2800	1.182996
2900	1.309371
3000	1.476040
3100	1.611637
3200	1.799314
3300	1.923163
3400	2.142325
3500	2.357385
3600	2.549284
3700	2.764697
3800	2.949835
3900	3.167777
4000	3.459731

N	parallel time
2000	0.895441
2100	1.066289
2200	1.157812
2300	1.320951
2400	1.495116
2500	1.699889
2600	1.897717
2700	2.120404
2800	2.436754
2900	2.608494
3000	2.927728
3100	3.163463
3200	3.458448
3300	3.781733
3400	4.184530
3500	4.517164
3600	4.919674
3700	5.420399
3800	5.774098
3900	6.220769
4000	6.728629

Guided:

Chunk size 1 Chunk size 2 Chunk size 4

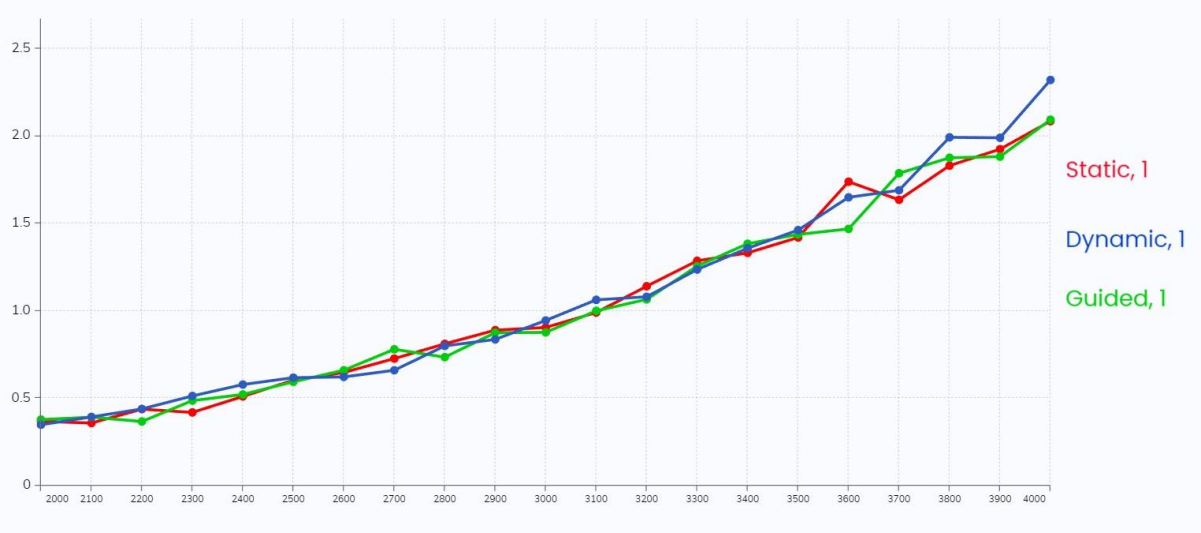
N	parallel time
2000	0.378712
2100	0.392696
2200	0.367912
2300	0.487362
2400	0.522471
2500	0.595031
2600	0.661151
2700	0.780848
2800	0.735989
2900	0.875809
3000	0.877657
3100	1.000773
3200	1.066476
3300	1.256565
3400	1.384892
3500	1.437791
3600	1.469716
3700	1.787833
3800	1.876434
3900	1.882887
4000	2.094023

N	parallel time
2000	0.447038
2100	0.545833
2200	0.588350
2300	0.681347
2400	0.808706
2500	0.891366
2600	0.963699
2700	1.062508
2800	1.188110
2900	1.380371
3000	1.449600
3100	1.654992
3200	1.765508
3300	1.969870
3400	2.251771
3500	2.297281
3600	2.506235
3700	2.779780
3800	3.144693
3900	3.181457
4000	3.488032

N	parallel time
2000	0.880155
2100	1.008570
2200	1.165505
2300	1.318952
2400	1.486435
2500	1.680893
2600	1.881506
2700	2.105959
2800	2.393027
2900	2.656724
3000	2.982273
3100	3.259501
3200	3.547752
3300	3.860896
3400	4.322529
3500	4.608951
3600	4.978643
3700	5.443602
3800	5.862262
3900	6.386899
4000	6.697861

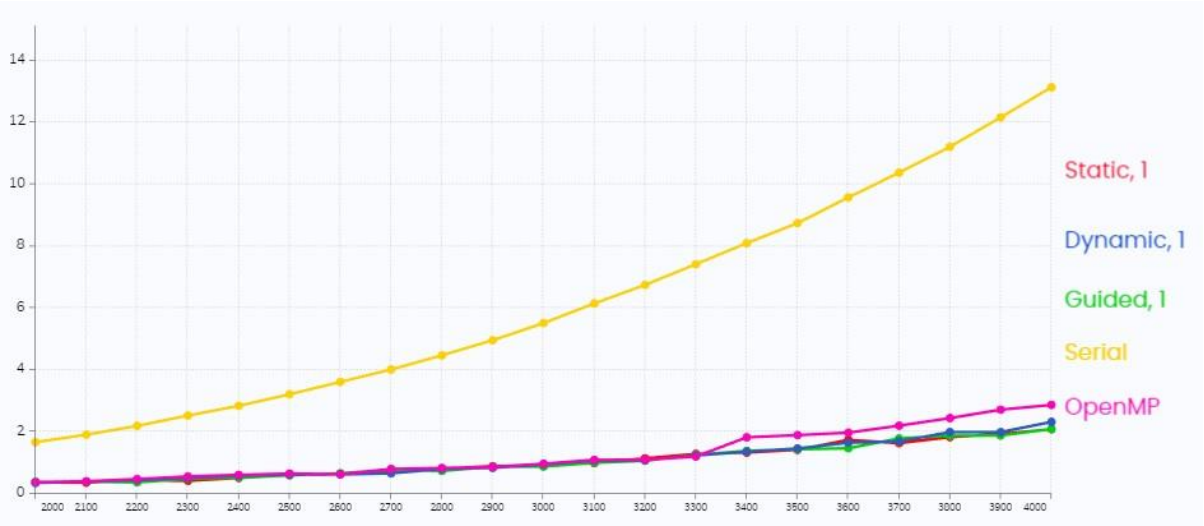
In the graph below we can see the 3 schedules compared.

x-axis = N, y-axis=time



We now compare the 3 schedule types to Serial and OpenMP with no schedule:

x-axis = N, y-axis=time



From the graph we can see that all OpenMP configurations were faster than regular BLAS. We can also see that using any schedule type with chunk size of 1 was faster than using no schedule type. Out of all 3 schedule types I found that guided was the fastest. I believe this is because guided scheduling dynamically decreases the chunk size at run time.