19 April 2018
Project #1
OSU CS475 Sp2018
Joshua L. Rogers
rogerjos@oregonstate.edu

Numeric Integration with OpenMP

I chose to run the experiment on flip, so I logged in and checked all of the flips for the lowest load via uptime. The best result was on flip3, so this was the server I used. In order to avoid variance due to processor load the C code runs a for loop from 1 to NUMT threads, within which it runs a for loop that iterates NUMTRIES times, within which the calculation takes place in a parallelized for loop using reduction. Then the necessary values are calculated both from the best performing iteration of the NUMTRIES loop and as an average of all tries for a given NUMNODE/NUMT combination, much like in Project 0. This assures that the timing is reliable as described in the project notes handout. The results are then output as comma-separated values. The bash script p1script serves to run the C code with the twelve NUMNODE values 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, and 16384 with NUMT of 12, which results in 144 test results, 132 of which are parallelized. These results are sent to a file.

The volume calculation converged rapidly on a value:

Nodes	8	16	32	64	128	256	512	1024	2056	4096	8192	16384
Volume	25.47064	25.34738	25.32069	25.31448	25.31299	25.31262	25.31253	25.31251	25.3125	25.3125	25.3125	25.3125

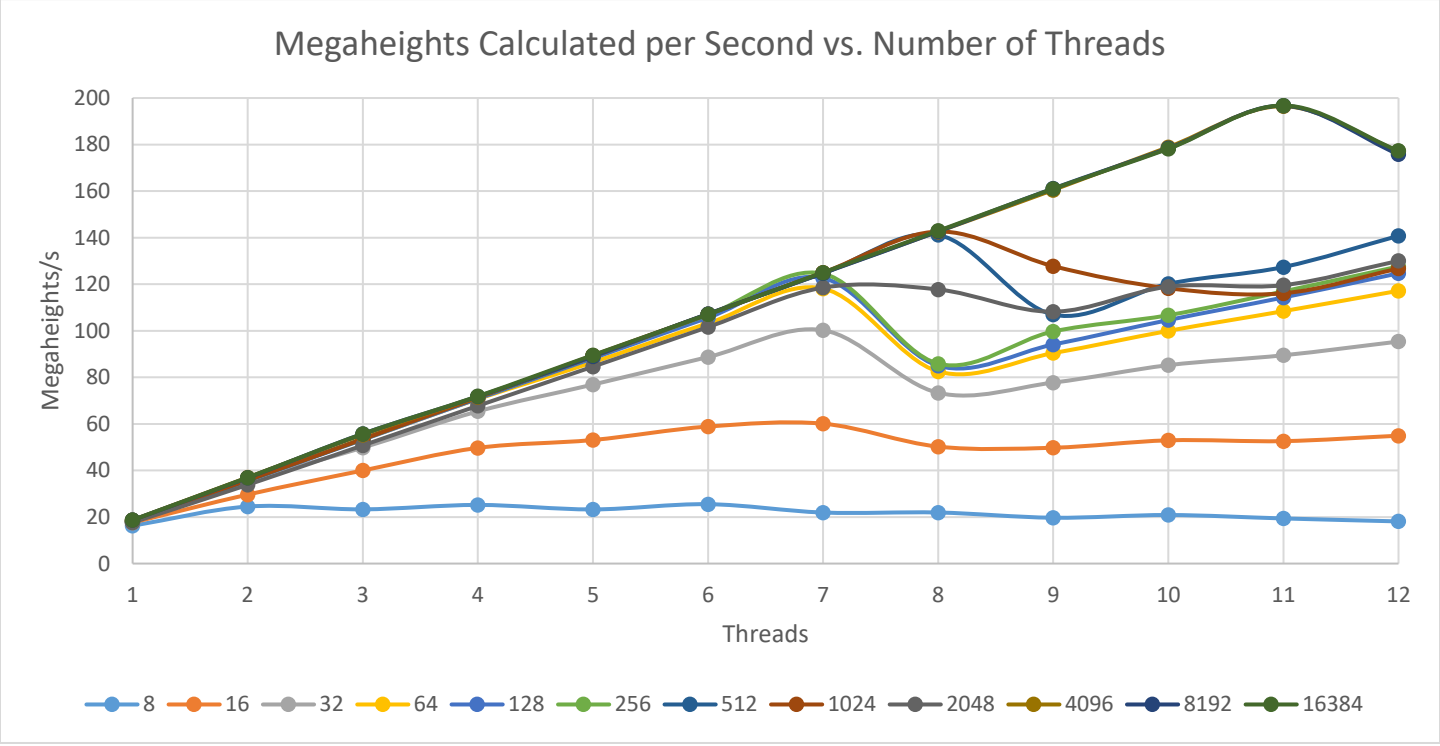
So it seems reasonable to conclude that the actual volume is 25.3125, or $25\frac{5}{16}$.

The average and peak performance values were close enough to be assured of reliability based on the project notes. The average performance values in Megaheights/s were as follows:

NUMNODES \ NUMT	1	2	3	4	5	6	7	8	9	10	11	12
8	15.07726	12.37402	13.61464	12.94726	12.42971	11.43441	11.3976	15.25624	14.38425	14.62043	14.06271	11.69564
16	16.95435	22.20813	31.02418	35.32628	34.66921	40.25815	39.13574	42.89684	39.2752	43.66444	43.2562	43.7716
32	17.78752	31.79049	44.43538	57.75727	63.55126	72.09634	83.18142	67.90438	71.23686	77.77643	80.58225	85.2016
64	17.9666	35.03684	51.19295	67.63893	81.03036	95.91684	109.1704	80.32901	87.54248	96.62963	104.3282	113.0617
128	17.71153	35.20122	52.15595	69.69839	86.45329	103.5112	119.9366	84.073	93.01715	103.2861	112.8452	123.0818
256	17.77751	35.51192	53.17477	70.8353	88.14992	105.6732	123.0062	85.03318	95.18453	101.5652	115.8999	126.5837
512	17.93944	35.49741	52.84412	70.67758	88.51767	105.3196	124.1671	90.25475	97.31679	113.0947	117.0802	127.6421
1024	17.82447	35.46373	53.24588	70.64566	88.69301	106.3436	123.3156	110.5743	101.4761	105.0523	79.29883	94.26001
2048	16.97127	33.8154	50.75344	66.61181	84.74735	100.1428	117.1739	86.20348	95.92958	109.2817	80.18491	85.77543
4096	17.6518	28.98472	54.22757	70.13478	88.61417	106.6715	124.2489	142.1871	145.6846	173.3659	171.8106	101.6446
8192	18.58016	35.12995	54.9896	69.42917	88.86729	103.7671	124.3267	139.6041	158.1468	176.2605	178.2786	109.1743
16384	18.31375	36.15912	54.84536	69.92183	89.1661	106.7004	124.1069	142.0001	158.9131	167.7529	192.7268	157.443

Thus, we will consider the peak performance numbers in Mh/s to be the effective best-case scenario:

NUMNODES \ NUMT	1	2	3	4	5	6	7	8	9	10	11	12
8	16.253424	24.490191	23.294738	25.190424	23.263195	25.508343	21.927083	21.941085	19.690394	20.862015	19.379435	18.112672
16	17.622638	29.633237	39.976426	49.688703	53.085729	58.860366	60.095738	50.251903	49.760664	52.942586	52.598145	54.931636
32	18.023008	35.470405	49.801233	65.392627	76.888925	88.727536	100.247231	73.359463	77.747958	85.23346	89.478485	95.377483
64	18.136424	36.112907	53.296734	70.667243	86.480386	103.206611	118.157179	82.536623	90.416646	99.96469	108.427753	117.181246
128	17.931559	35.594996	53.182982	70.826567	88.204374	105.615641	122.660303	84.992154	94.103035	104.621981	114.310687	124.682387
256	18.010332	35.8013	53.446997	71.355592	89.101429	106.858608	124.535429	85.866224	99.713966	106.685584	116.923066	127.637306
512	18.06269	35.755386	53.539347	71.313867	89.178382	106.977929	124.695201	141.173432	106.948094	120.178853	127.390874	140.792961
1024	18.494473	35.799556	53.456864	71.288834	89.404591	107.135548	124.99072	142.620326	127.765925	118.287398	116.113693	126.724599
2048	17.666446	33.827058	50.765571	67.76999	84.597713	101.606392	118.552199	117.752997	108.200758	118.974104	119.60361	130.051442
4096	18.672663	37.012656	55.353985	71.561832	88.911996	107.170354	124.879614	142.629543	160.50515	178.904758	196.486927	177.137212
8192	18.691369	36.840682	55.74891	71.770659	88.991625	107.30267	124.757715	142.660162	161.13538	178.394713	196.694118	175.820637
16384	18.704538	36.921951	55.700179	71.904142	89.561832	107.094247	125.054277	142.932365	161.023317	178.17652	196.790431	177.303098



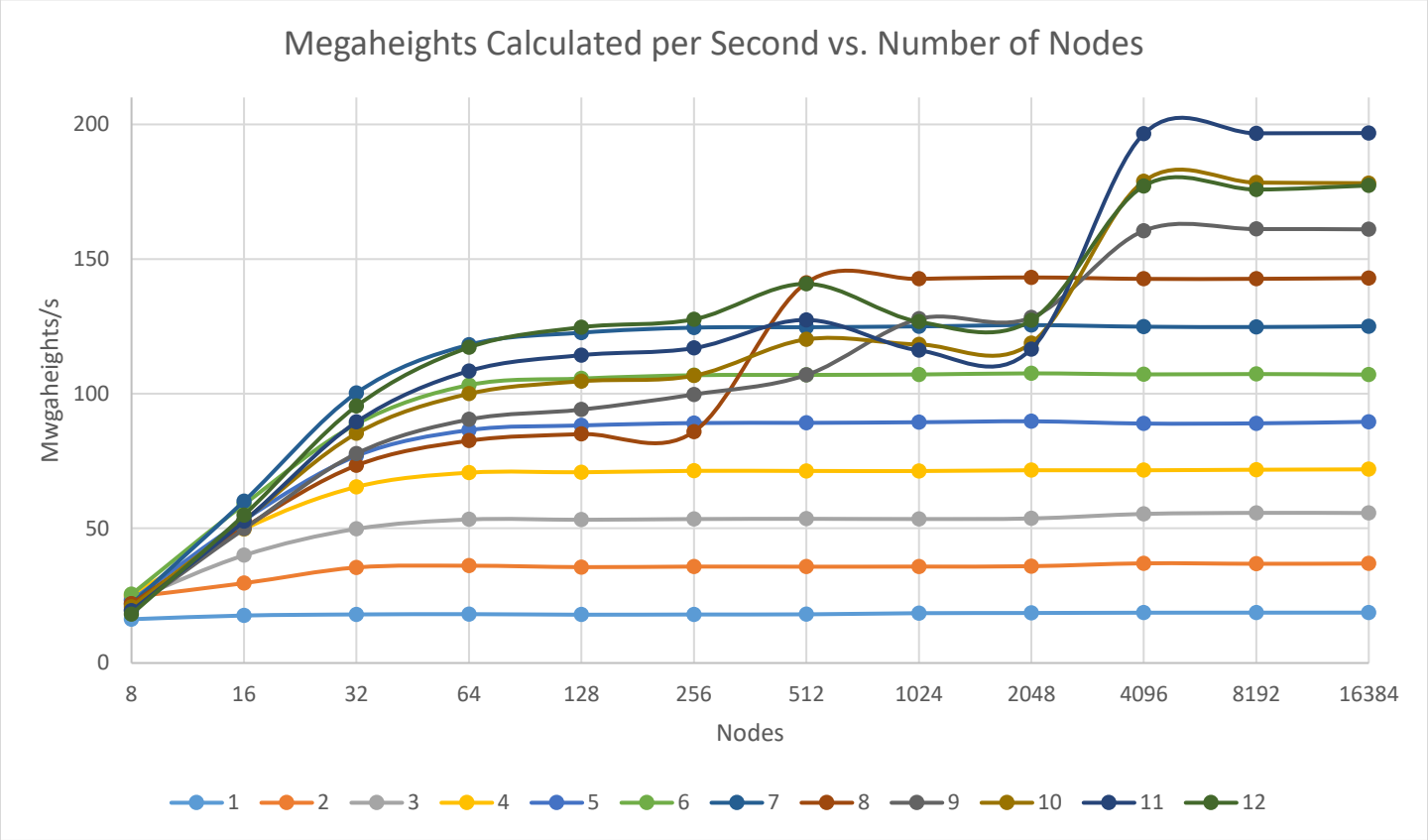
The Mh/s as a function of the number of threads, with each colored line indicating a given number of subdivisions (e.g., node count), reveals some interesting patterns. First, the performance for low node counts such as 8 (the light blue line) declines as thread count increases. This follows because a low number of nodes is easily evaluated by a single thread, and the performance gain from multiple threads is insufficient to offset the increased overhead involved in setting up and managing the threads.

More interestingly, there is a performance dip at some points on the graph under the following conditions:

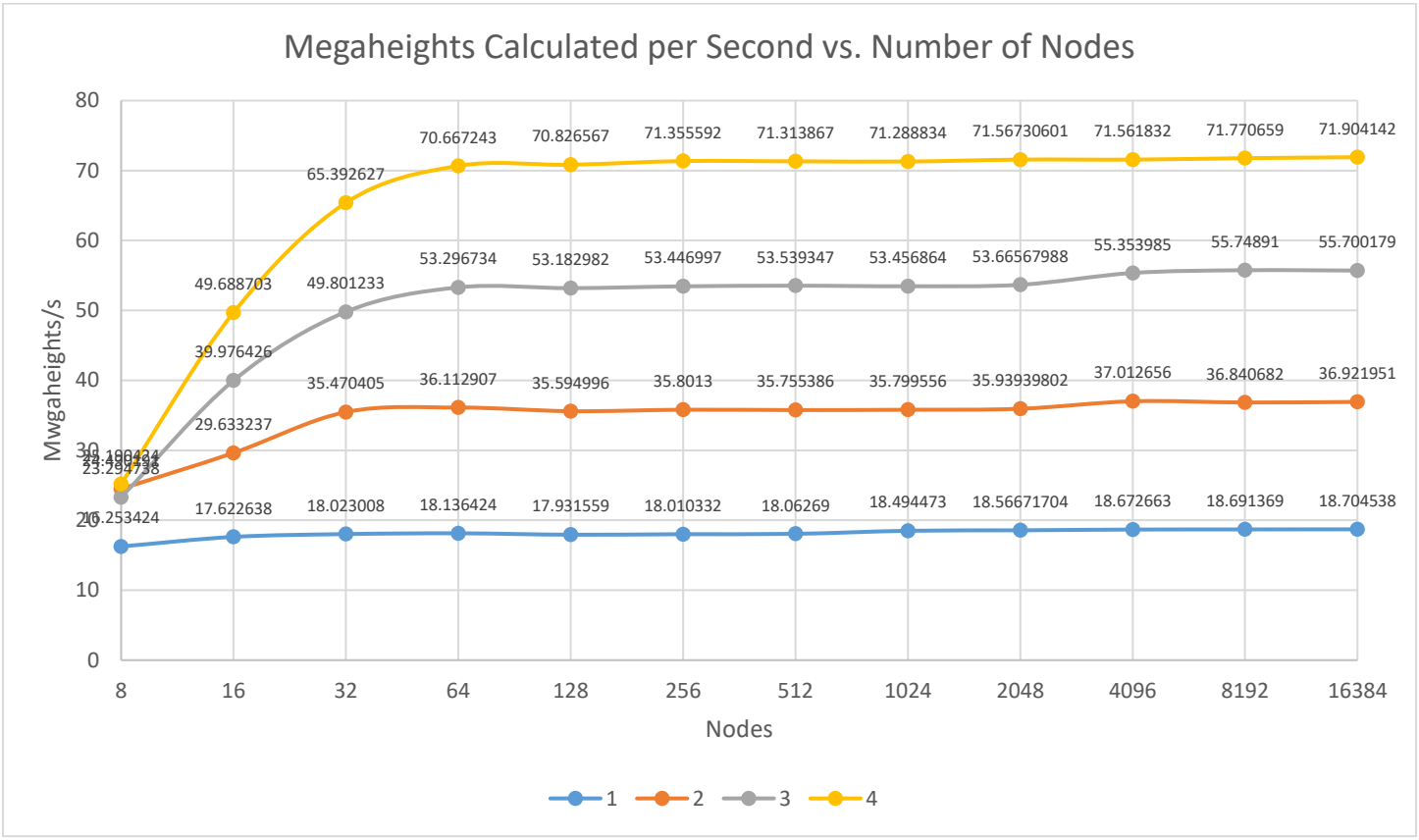
- node count is 16, 32, 64, 128, or 256 and thread count is increased from 7 to 8
- node count is 512 or 1024 and thread count is increased from 8 to 9
- node count is 4096, 8192, or 16384 and thread count is increased from 11 to 12

The dark gray line for 2048 nodes seems to be the only deviation from this pattern, but this could be explained by an increased load on the server during execution of that part of the code.

Ideally once the performance gains from threading are sufficient to overcome the threading overhead we would expect to see a linear increase in Mh/s as thread count increases. I'm not entirely certain what caused this performance dip. My best guess would be cache misses, since there doesn't seem to be any other explanation that fits the data. This is supported by the data graphed against number of subdivisions, e.g., node count:



Here we see the Mh/s as a function of the number of subdivisions, with each colored line indicating a given number of threads. Note that the x-axis is represented on a logarithmic scale since the node count doubles at each interval. For lower node counts the expected behavior seems to hold, but once the number of subdivisions increases to the point described on the previous page attempts to parallelize the process over eight or more threads has a negative impact on performance. The code is generating a lot of values, most of which will not be reused very often. Additionally, adding threads increases the number of values at any given time that will not be reused, filling shared cache space with useless information and triggering cache misses.



This is the same graph as the previous page, but only displaying results for 1, 2, 3, and 4 thread execution. This result reveals no inconsistencies, and is what I initially expected to see for all thread counts.

For each test I computed the following:

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

$$Efficiency_n = \frac{Speedup_n}{n}$$

$$F_{parallel} = \frac{n}{n-1}\left(1 - \frac{1}{Speedup}\right)$$

$$Speedup_{max} = \frac{1}{1 - F_{parallel}}$$

These values appear on the following pages. I note that the results when thread count is 2 appear to be inflated and unreliable. Additionally, there appears to be some variance which is probably a result of load on the server. Given the data, it would appear that the highest reasonable parallel fraction result is 0.904972 with 11 threads and 8192 nodes. This results in a maximum speedup of 10.52321421. It is interesting to note that generally $F_{parallel}$ does increase and so Gustafson's Observation holds, load-induced variance aside.

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
8	1	1	1	0	1
8	2	1.506771	0.753386	0.672658	3.054908933
8	3	1.43322	0.47774	0.302271	1.433221208
8	4	1.549853	0.387463	0.354778	1.549854159
8	5	1.43128	0.286256	0.301325	1.431280638
8	6	1.569414	0.261569	0.362819	1.569412773
8	7	1.349075	0.192725	0.258751	1.349074333
8	8	1.349936	0.168742	0.259224	1.349935743
8	9	1.211461	0.134607	0.174551	1.211461883
8	10	1.283546	0.128355	0.220908	1.283545461
8	11	1.192329	0.108394	0.161306	1.19232998
8	12	1.114391	0.092866	0.102649	1.114391136

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
16	1	1	1	0	1
16	2	1.681544	0.840772	0.810617	5.28030499
16	3	2.26847	0.756157	0.559174	2.268468738
16	4	2.819595	0.704899	0.645339	2.819593922
16	5	3.01236	0.602472	0.668034	3.012356687
16	6	3.340043	0.556674	0.700603	3.340046827
16	7	3.410144	0.487163	0.706757	3.410141078
16	8	2.851554	0.356444	0.649314	2.851553812
16	9	2.823678	0.313742	0.645852	2.823678236
16	10	3.004237	0.300424	0.667137	3.004238981
16	11	2.984692	0.271336	0.664957	2.984691517
16	12	3.117106	0.259759	0.67919	3.117109816

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
32	1	1	1	0	1
32	2	1.968062	0.984031	0.983772	61.62188809
32	3	2.763203	0.921068	0.638101	2.763201888
32	4	3.628286	0.907071	0.724388	3.628289044
32	5	4.266154	0.853231	0.765597	4.266157003
32	6	4.923015	0.820502	0.796872	4.923004214
32	7	5.562181	0.794597	0.820214	5.562168356
32	8	4.070323	0.50879	0.754319	4.070318828
32	9	4.313817	0.479313	0.768187	4.313821917
32	10	4.729147	0.472915	0.788545	4.729138587
32	11	4.964681	0.451335	0.798577	4.964676328
32	12	5.291985	0.440999	0.811035	5.291985288

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
64	1	1	1	0	1
64	2	1.991181	0.995591	0.995571	225.7846015
64	3	2.938657	0.979552	0.659709	2.938661322
64	4	3.896427	0.974107	0.743355	3.896432816
64	5	4.768326	0.953665	0.790283	4.768330655
64	6	5.690571	0.948429	0.824271	5.690580382
64	7	6.514911	0.930702	0.846506	6.514912635
64	8	4.550876	0.56886	0.780262	4.550874223
64	9	4.985362	0.553929	0.799413	4.985367945
64	10	5.511819	0.551182	0.818572	5.511828384
64	11	5.978453	0.543496	0.832733	5.978465567
64	12	6.4611	0.538425	0.845228	6.461116998

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
128	1	1	1	0	1
128	2	1.985048	0.992524	0.992467	132.7492367
128	3	2.965887	0.988629	0.662833	2.965889307
128	4	3.949828	0.987457	0.746824	3.949821468
128	5	4.918946	0.983789	0.796704	4.918935936
128	6	5.889931	0.981655	0.830219	5.88994057
128	7	6.840471	0.97721	0.853811	6.840459953
128	8	4.739808	0.592476	0.789021	4.739808227
128	9	5.2479	0.5831	0.809448	5.247911331
128	10	5.834517	0.583452	0.828606	5.83450996
128	11	6.374833	0.57953	0.843133	6.374827083
128	12	6.953238	0.579436	0.856182	6.953232558

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
256	1	1	1	0	1
256	2	1.98782	0.99391	0.993873	163.2120124
256	3	2.967574	0.989191	0.663024	2.967570391
256	4	3.961925	0.990481	0.747597	3.961918044
256	5	4.94724	0.989448	0.797867	4.94723771
256	6	5.933184	0.988864	0.831456	5.933168787
256	7	6.914666	0.987809	0.85538	6.914672936
256	8	4.767609	0.595951	0.790251	4.767603183
256	9	5.536487	0.615165	0.81938	5.536485439
256	10	5.923577	0.592358	0.831183	5.923574048
256	11	6.492	0.590182	0.845964	6.491988886
256	12	7.086894	0.590574	0.858894	7.086870863

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
512	1	1	1	0	1
512	2	1.979516	0.989758	0.989652	96.63703131
512	3	2.964085	0.988028	0.662628	2.96408712
512	4	3.948131	0.987033	0.746716	3.948137269
512	5	4.93716	0.987432	0.797454	4.937150079
512	6	5.922591	0.987099	0.831155	5.922591726
512	7	6.903468	0.98621	0.855145	6.903455179
512	8	7.815748	0.976969	0.872053	7.815736203
512	9	5.92094	0.657882	0.831108	5.920943562
512	10	6.653431	0.665343	0.849702	6.653448482
512	11	7.052708	0.641155	0.85821	7.052683546
512	12	7.794684	0.649557	0.871707	7.794657542

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
1024	1	1	1	0	1
1024	2	1.935689	0.967845	0.966776	30.09872381
1024	3	2.890424	0.963475	0.65403	2.890424025
1024	4	3.854602	0.963651	0.74057	3.854604325
1024	5	4.834125	0.966825	0.793137	4.834117266
1024	6	5.792841	0.965474	0.827373	5.792836578
1024	7	6.758274	0.965468	0.852033	6.758263667
1024	8	7.711511	0.963939	0.870324	7.711527191
1024	9	6.90833	0.767592	0.855247	6.908319689
1024	10	6.395824	0.639582	0.843648	6.395824806
1024	11	6.278292	0.570754	0.840721	6.278291551
1024	12	6.852025	0.571002	0.854058	6.852037111

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
2048	1	1	1	0	1
2048	2	1.914763	0.957382	0.955484	22.46383323
2048	3	2.873559	0.957853	0.651999	2.873554961
2048	4	3.836085	0.959021	0.739318	3.836091483
2048	5	4.788609	0.957722	0.791171	4.788606946
2048	6	5.751377	0.958563	0.826129	5.751390399
2048	7	6.710585	0.958655	0.850982	6.71059872
2048	8	6.665347	0.833168	0.84997	6.6653336
2048	9	6.124647	0.680516	0.836725	6.12463635
2048	10	6.734467	0.673447	0.85151	6.734460233
2048	11	6.7701	0.615464	0.852292	6.770114009
2048	12	7.361494	0.613458	0.864158	7.3614935

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
4096	1	1	1	0	1
4096	2	1.982184	0.991092	0.991012	111.2594571
4096	3	2.96444	0.988147	0.662668	2.964438595
4096	4	3.832438	0.95811	0.73907	3.832445483
4096	5	4.761613	0.952323	0.789987	4.761609996
4096	6	5.739425	0.956571	0.825767	5.739440864
4096	7	6.687831	0.955404	0.850475	6.687844842
4096	8	7.638415	0.954802	0.869083	7.638427401
4096	9	8.595729	0.955081	0.883663	8.595717613
4096	10	9.581106	0.958111	0.895628	9.581113709
4096	11	10.522705	0.95661	0.904967	10.52266055
4096	12	9.486446	0.790537	0.894586	9.48640598

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
8192	1	1	1	0	1
8192	2	1.971	0.9855	0.985286	67.96248471
8192	3	2.982602	0.994201	0.664722	2.982599514
8192	4	3.839775	0.959944	0.739568	3.839773914
8192	5	4.761108	0.952222	0.789965	4.761111243
8192	6	5.74076	0.956793	0.825807	5.740758813
8192	7	6.674616	0.953517	0.850179	6.674631727
8192	8	7.632408	0.954051	0.86898	7.632422531
8192	9	8.620844	0.957872	0.884002	8.62083829
8192	10	9.544229	0.954423	0.895225	9.544261513
8192	11	10.523259	0.95666	0.904972	10.52321421
8192	12	9.406515	0.783876	0.893691	9.406541309

Nodes	Threads	Speedup	Efficiency	Fp	Max Speedup
16384	1	1	1	0	1
16384	2	1.973957	0.986978	0.986807	75.79777155
16384	3	2.977897	0.992632	0.664193	2.977900997
16384	4	3.844208	0.961052	0.739868	3.844202174
16384	5	4.788241	0.957648	0.791155	4.788240082
16384	6	5.725576	0.954263	0.825345	5.725573273
16384	7	6.685772	0.95511	0.850429	6.68578802
16384	8	7.641588	0.955198	0.869137	7.641579362
16384	9	8.608784	0.956532	0.88384	8.608815427
16384	10	9.525845	0.952584	0.895022	9.525805407
16384	11	10.521	0.956455	0.904952	10.52099992
16384	12	9.479149	0.789929	0.894505	9.479122233