Question 6: OpenMp

Sean Marshallsay: sm1183

Table 1: At a low number of iterations the cost of setting up a three- or four-core process outweighs the gains of using parallelism.

	time (s)			
number of iterations	1 core	2 cores	3 cores	4 cores
1000	0.000248	7.9e-05	0.001125	0.000555
2000	0.000375	6e-05	6e-05	6.6e-05
5000	0.000933	0.000152	0.000147	0.000162
10000	0.0019	0.000304	0.000297	0.000326
20000	0.004278	0.000661	0.000683	0.00076
50000	0.005868	0.001569	0.001684	0.001858
100000	0.006589	0.003144	0.004609	0.004842
200000	0.012461	0.006792	0.006818	0.010621
500000	0.030382	0.016011	0.011097	0.010773
1000000	0.060543	0.030596	0.021261	0.019381
2000000	0.119939	0.061187	0.041904	0.037598
5000000	0.300413	0.151024	0.103298	0.090076

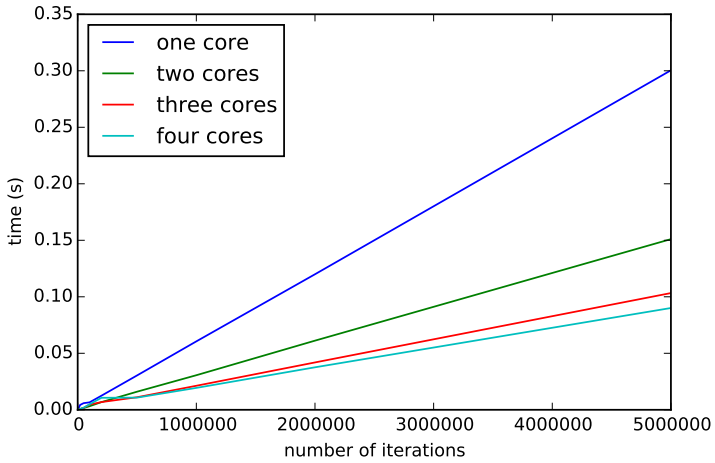

Figure 1: A linear graph of the time taken to sum with various numbers of cores. This shows that as the number of iterations gets large more cores produces faster code. At this number of iterations ($> 1,000,000$) using c cores will give a factor of c speed up.

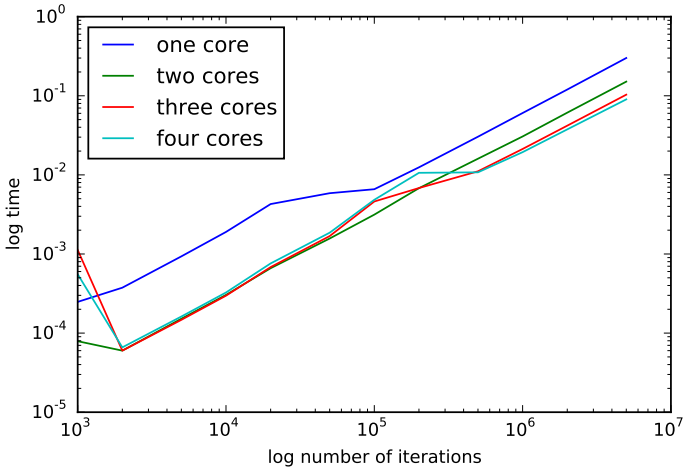

Figure 2: A log graph of figure 1, emphasising the speed of multiple cores at lower numbers of iterations. This shows that at lower numbers of iterations more cores can have an adverse affect since the cost of setting up the parallelism is greater than the gains of using parallel code.