Jesse Thoren

April 24, 2017

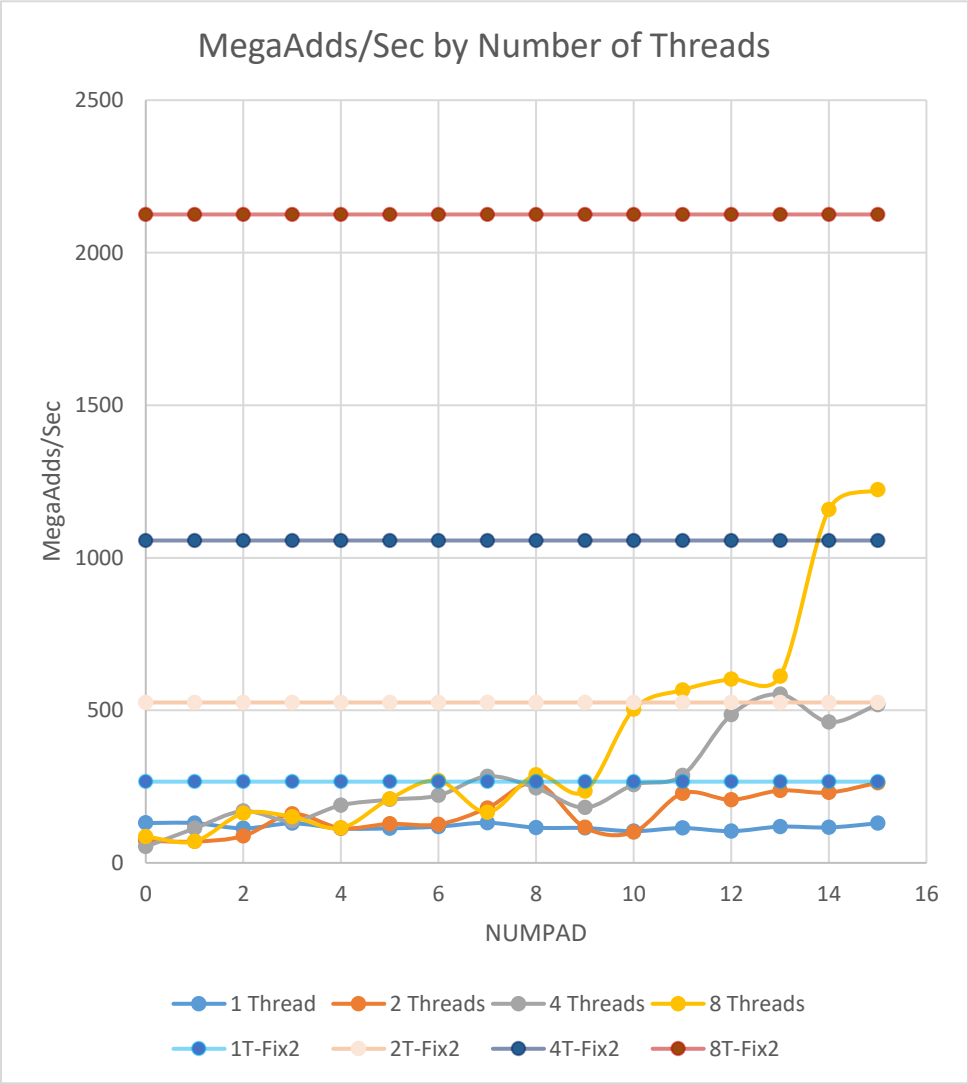
CS 475 – Parallel Programming

Project 3 – False Sharing – Commentary

1. I ran this on the OSU Rabbit machine at 1:20 PM on 4/24/17
2. Below is a table of my results. These are in MegaAdditions/sec. The leftmost column designates the value of NUMPAD that the structure was padded with. The rightmost 4 columns are single values that correspond to the value found by implementing Fix2 with the corresponding number of threads.

	1 Thread	2 Threads	4 Threads	8 Threads	1T-Fix2	2T-Fix2	4T-Fix2	8T-Fix2
0	130.63	72.38	53.07	85.93	266.25	525.75	1056.68	2125.5
1	130.4	69.88	112.89	70.61	266.25	525.75	1056.68	2125.5
2	112.91	88.12	169.62	162.18	266.25	525.75	1056.68	2125.5
3	129.57	159.96	136.91	151.32	266.25	525.75	1056.68	2125.5
4	112.8	113.76	188.84	114.32	266.25	525.75	1056.68	2125.5
5	112.99	127.67	207.16	210.12	266.25	525.75	1056.68	2125.5
6	118.82	126.25	221.03	269.5	266.25	525.75	1056.68	2125.5
7	131.44	179.78	283.48	165.86	266.25	525.75	1056.68	2125.5
8	115.28	262.13	245.49	288.3	266.25	525.75	1056.68	2125.5
9	114.12	116.06	181.26	234.56	266.25	525.75	1056.68	2125.5
10	103.69	100.94	255.87	503.31	266.25	525.75	1056.68	2125.5
11	114.21	227.99	286.56	566.75	266.25	525.75	1056.68	2125.5
12	103.84	207.08	485.12	602.46	266.25	525.75	1056.68	2125.5
13	118.7	237.09	552.5	611.51	266.25	525.75	1056.68	2125.5
14	116.06	230.61	461.71	1157.81	266.25	525.75	1056.68	2125.5
15	129.53	262.6	518.23	1223.26	266.25	525.75	1056.68	2125.5

3. This chart displays the data from the table in question 2. The horizontal lines correspond to the values from implementing fix 2.



4. Note that the values for 2, 4, and 8 threads begin slightly below the value for one thread. This suggests that there is some additional time being consumed by false sharing.
   As the amount padded by numpad increases, we see that the data settles into roughly n*performance for 1 thread, where n is the number of threads used. This is also seen by the numbers seen using fix 2, which have roughly 2x, 4x, and 8x performance from using 2, 4, and 8 threads.
5. For fix 1, there is additional overhead in the values due to false sharing. However, this isn't as nice as the expected values that I saw in the lecture graph. I'm not entirely sure why this is happening, but it may be in part due to my calculation of the performance values, and may be due to some variance that occurred during calculation on the server. However, I have no idea why that variance would happen, because I was likely the only person logged in at the time (from checking the active processes, and the current server load).