Andrew Frost
CS415
PA4 Report
04/12/2017

Overview:

In this programming assignment matrices were multiplied using sequential and parallel algorithms. The sequential program made us of a simplistic matrix multiplication algorithm that ran in O(n^3). Cannon's algorithm was used in the parallel program. In this report the parallel run times and sequential run times are compared. The complete data for this report can be found in data.xlsx. Table 1 shows a sample result of matrix multiplication.

Matrix Multiplication Results

Mat A:		
1	2	3
2	3	4
3	4	5
Mat B:		
3	2	1
4	3	2
5	4	3
Mat C:		
26	20	14
38	29	20
50	38	26

Table 1: an example of multiplying two 3x3 matrices using the sequential algorithm.

Table 1 shows a sample result from multiplying two matrices together using the sequential algorithm. The sequential algorithm will be discussed in depth below.

Sequential:

The sequential algorithm run times are based off of single sample run times. On select tests, multiple time trials were ran and little deviation was found. Based on the low variance found in select sample run times, the data used in this report for sequential run times was not averaged or processed. The sequential algorithm used had a time complexity of O(n^3) and the run time results reflect this curve. Figure 1 graphs the run times of the sequential algorithm.

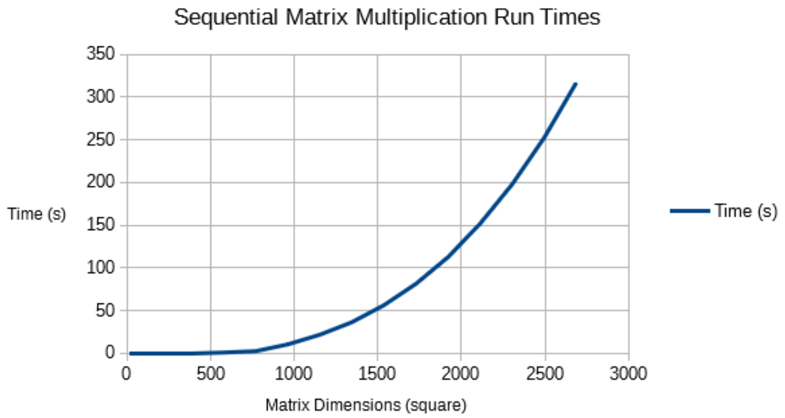

Figure 1: a graph of sequential run times in seconds. As the size of the matrix increases, the run time required increases dramatically.

As seen in the graph in Figure 1, the run time of the sequential algorithm increases dramatically as the matrix dimension size increases. The maximum tested dimension of 2688x2688 exceeded the 5 minute run time. This not only shows the undesirability of an $O(n^3)$, but highlights the need for an efficient parallel algorithm. The run times of the sequential algorithm are included in Table 2.

Sequential Matrix Multiplication Run Times

Matrix Dimensions (Square)	Time (s)
12	5E-06
24	3E-05
48	0.000282
96	0.001602
192	0.010534
384	0.128293
576	0.669576
768	2.31288
960	10.2022
1152	21.5373
1344	36.2078
1536	56.1688
1728	81.2236
1920	112.514
2112	151.711
2304	197.9
2496	252.654
2688	317.111

Table 2: the run time in seconds of the sequential matrix multiplication algorithm. The lack of fluctuation can be attributed to consistency provided by the algorithm running only on a single processor core of the same make and model.

Nothing unusual was encountered when testing the sequential algorithm. The sequential algorithm will now be compared to the parallel algorithm.