

# Homework 2 - Experiments with Caching

Robert Smith

September 30, 2013

## 1 Experimental Design

The matrices were implemented as 1-dimensional short int arrays, and were initialized outside of the timer. A function was created so that the rows & columns could be easily accessed, and the blocking loop was implemented similarly to the example. The timer was begun after the matrix initialization, and ends immediately after the matrix multiplication is completed. Finally, the experiment was run on the Tesla server.

## 2 Output

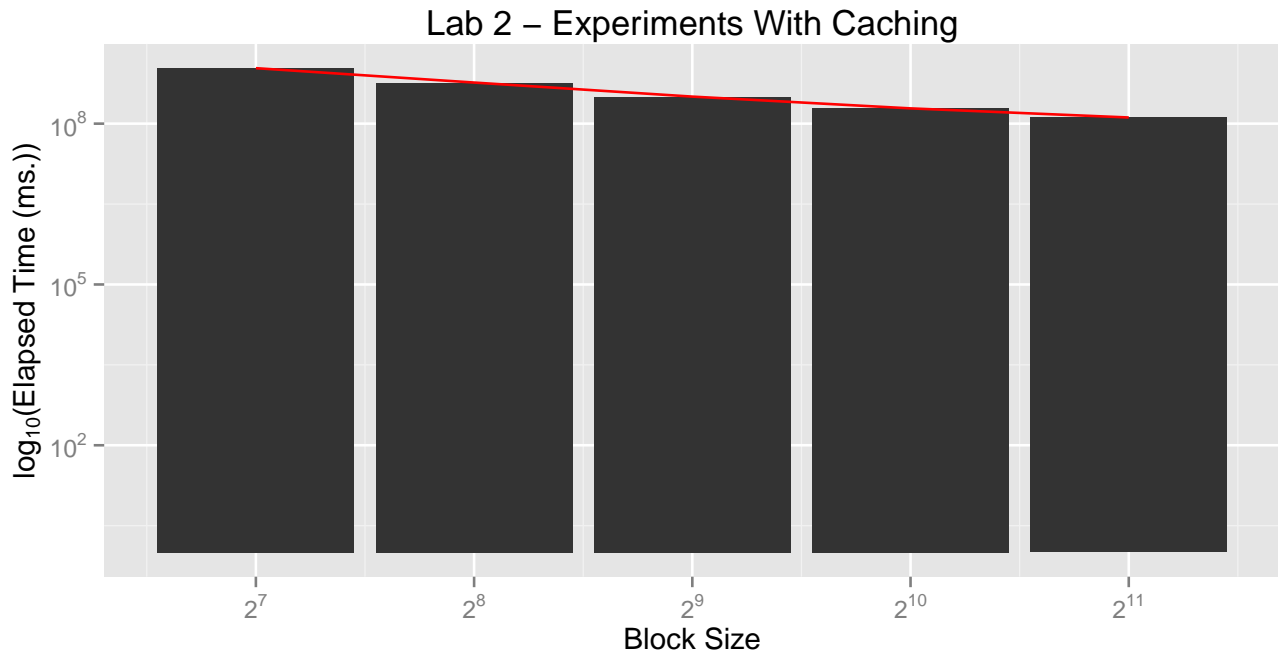
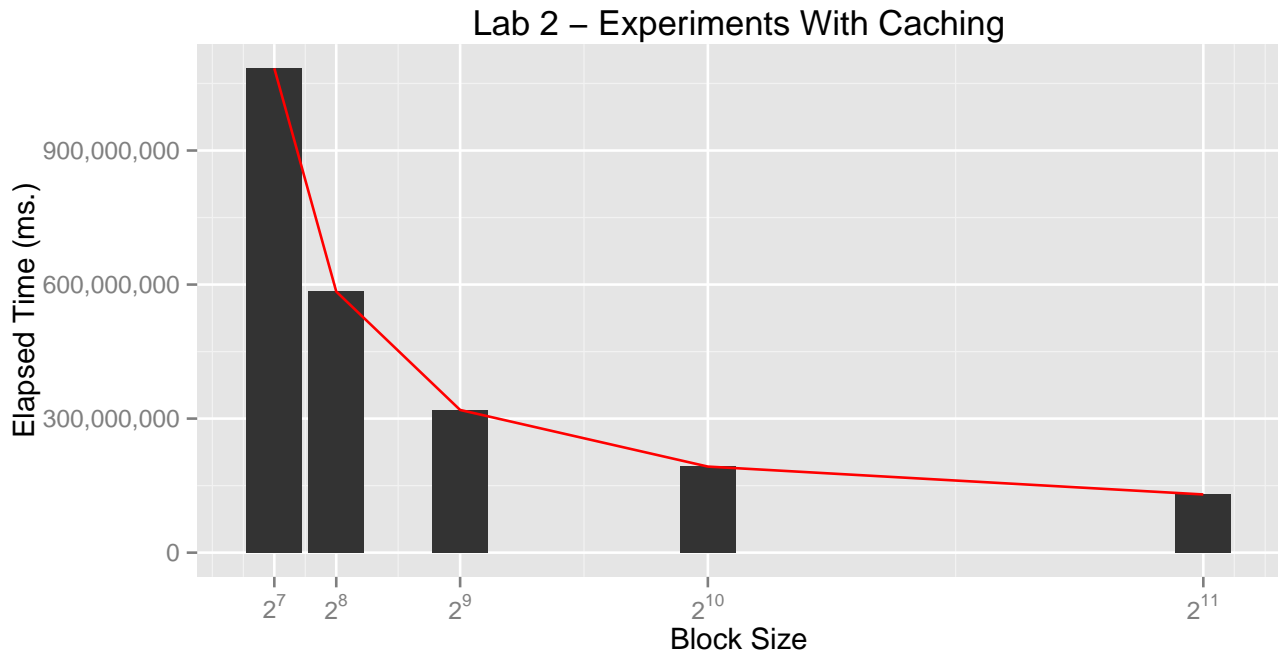
Based on three trials, I found a quite profound & repeatable linear speedup between Block Sizes of  $2^7$  and  $2^8$ , with slower speedups in the ranges of  $2^9$  to  $2^{11}$ . This speedup, when viewed after a logarithmic transformation appears to be linear and therefore as block size increases by a power of X, time to complete the process in milliseconds appears to fall by a power of Y. From the table below, none of the times were outside a certain range so I don't expect that there will be any extraordinary speedup achieved by moving towards larger block sizes.

Table 1: Table of Minimum Elapsed Time

	Block_Size	Elapsed_Time	exp
1	128	1083894512	1
2	256	584329083	2
3	512	319391493	3
4	1024	192616222	4
5	2048	130219659	5

Table 2: Table of Elapsed Times from All Runs

	Block_Size	Run_1	Run_2	Run_3
1	128	1083894512	1105853807	1104741789
2	256	623339217	585105284	585309110
3	512	319391493	325777344	325506858
4	1024	192616222	195882200	195787772
5	2048	130653006	130632564	130503220



### 3 Conclusions

Based on the trials, I estimate that the cache size on my laptop Intel Core i5 is around  $2 * 2^{11} = 2^{12}$  bytes (as short ints are stored as 2-byte data types). To find this estimate, we see a fairly linear time trend when we look at the logarithmic value of the time it takes to complete the process which suggests that since we don't see a local minimum somewhere outside the left and right side of the graph that we've approached the cache size but not actually surpassed it. While the cache maximum was not proven to be obtained, by looking at the slope of the untreated data we can see that we are rapidly approaching the system minimum and as such should not be much more than a few extra powers of two.