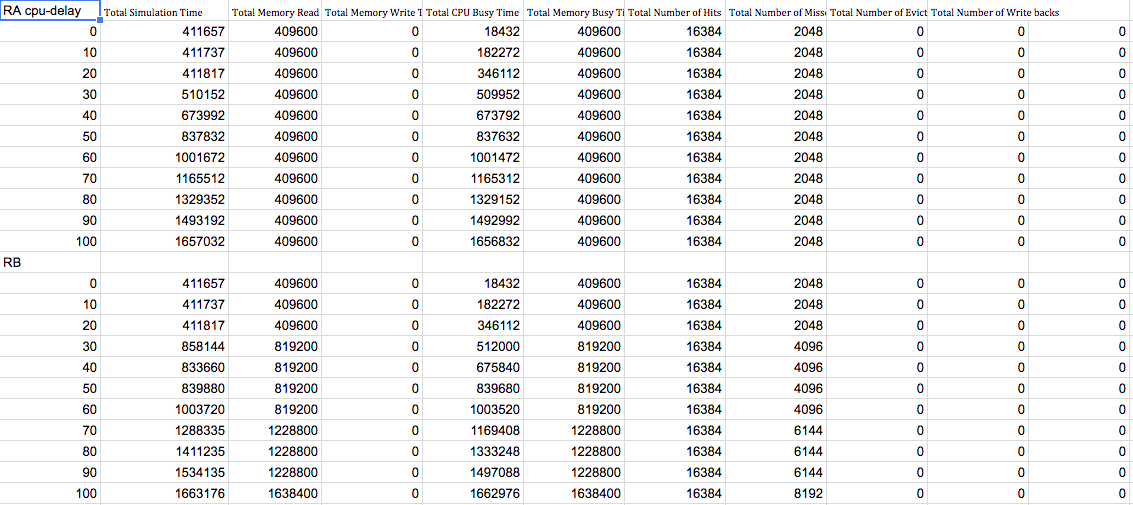
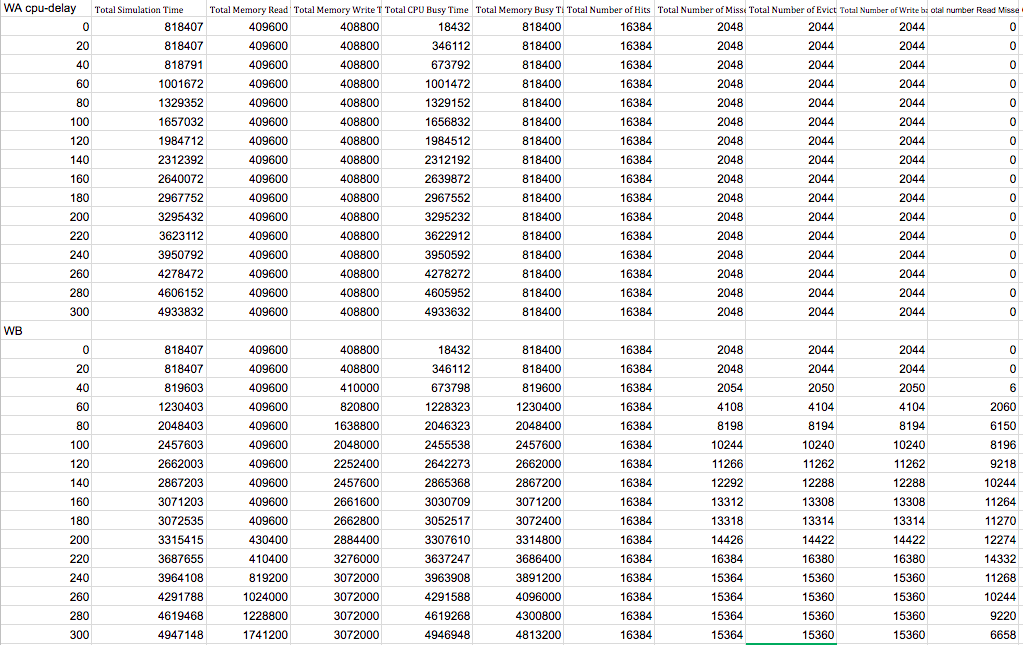
**Comp 526 HW2**

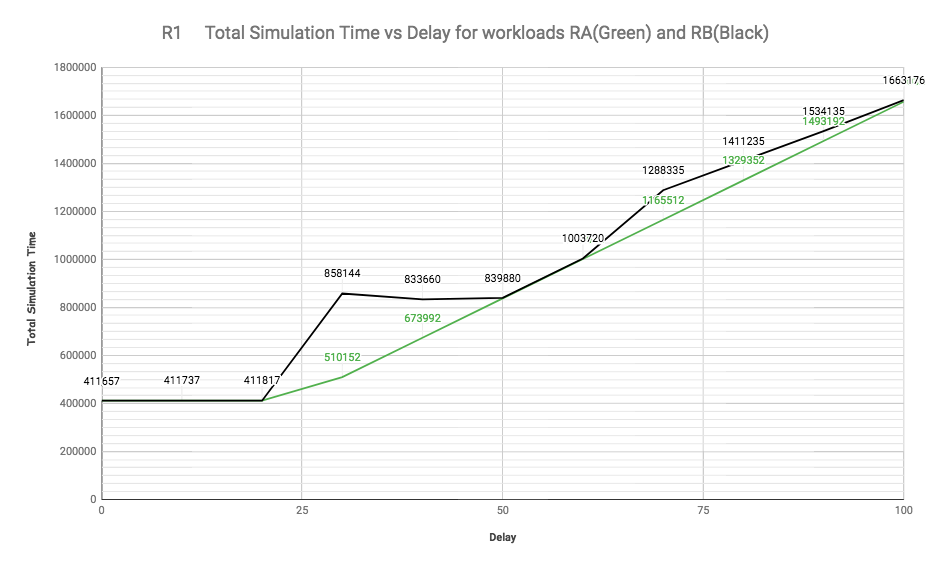
Mo Tang mt60

Data for RA/RB:

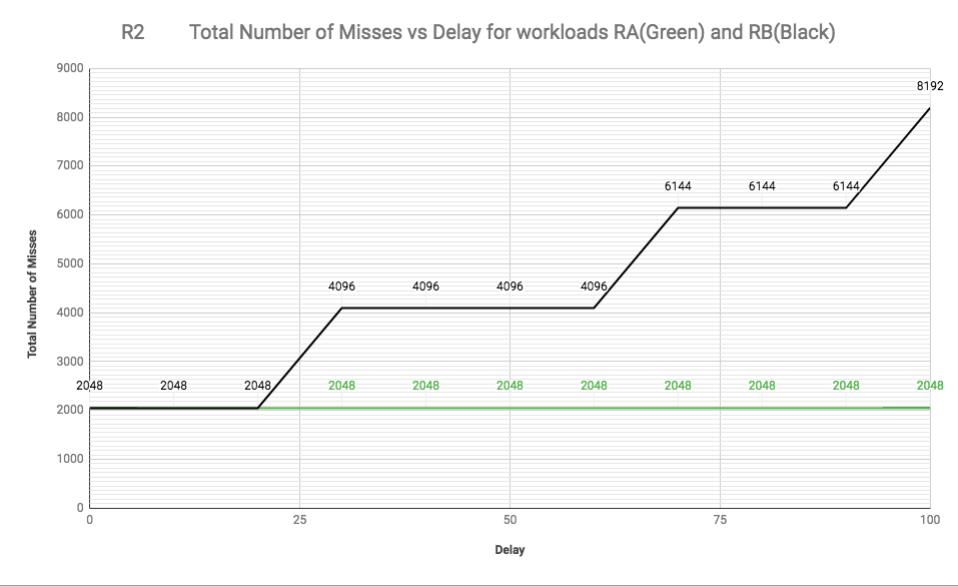


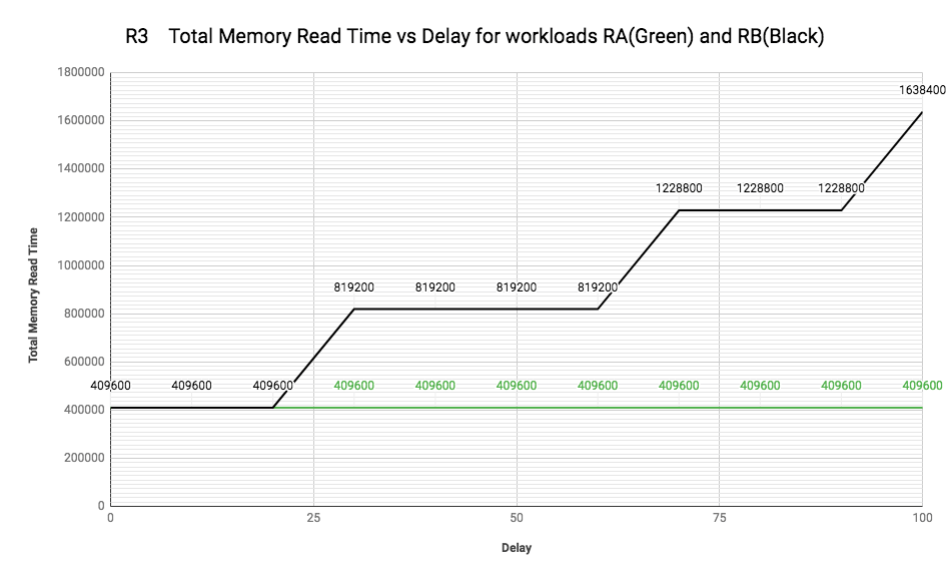
Data for WA/WB:



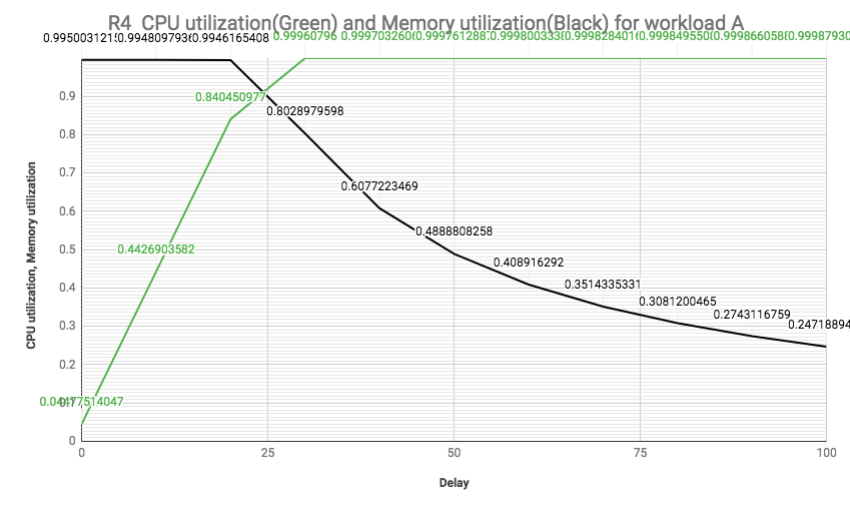


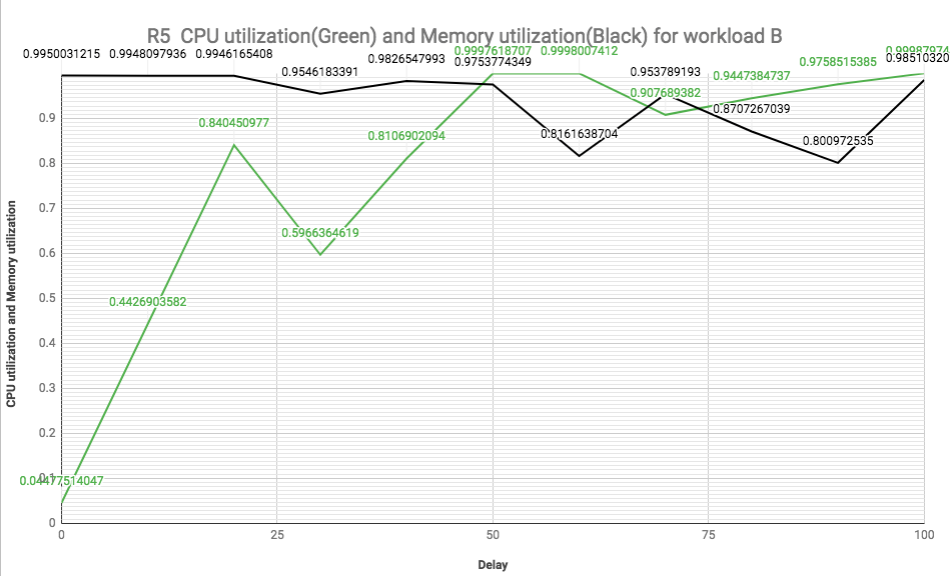
1. For RA and RB, the total CPU busy time is less than total simulation time, which causes the curves between delay0 to delay20 seem flat. Miss occurs when thread read the area(if succeed: cache hit) occupied by others. When the delay is small, the read can read the block and won’t conflict other thread’s reading. When the delay gets bigger, the other thread’s reading will block current thread’s reading. By analyzing the result of memtrace0 and memtrace1, we can find there are conflict between two threads. So the RB shows a staircase like pattern.



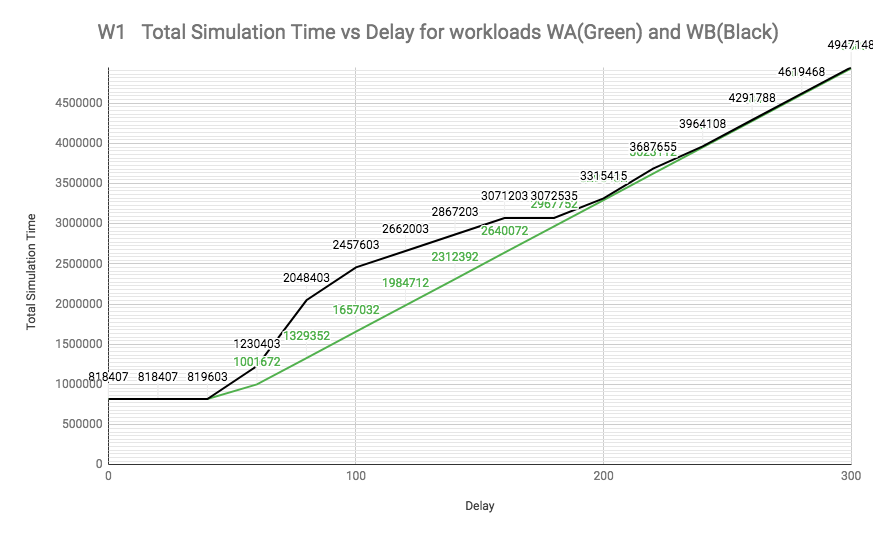


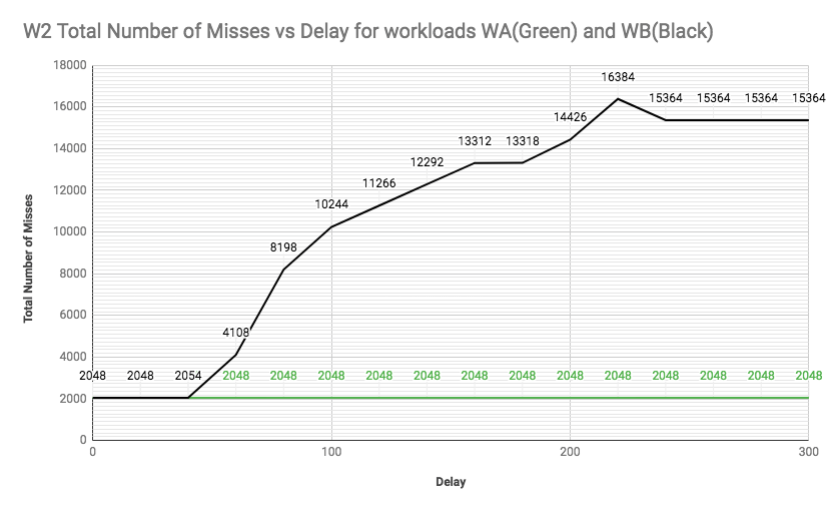
1. R2 and R3 have the same shape of curves. That is because once the miss occurs, data will be read from memory. Therefore, the total number of misses and total memory read time will increase simultaneously.

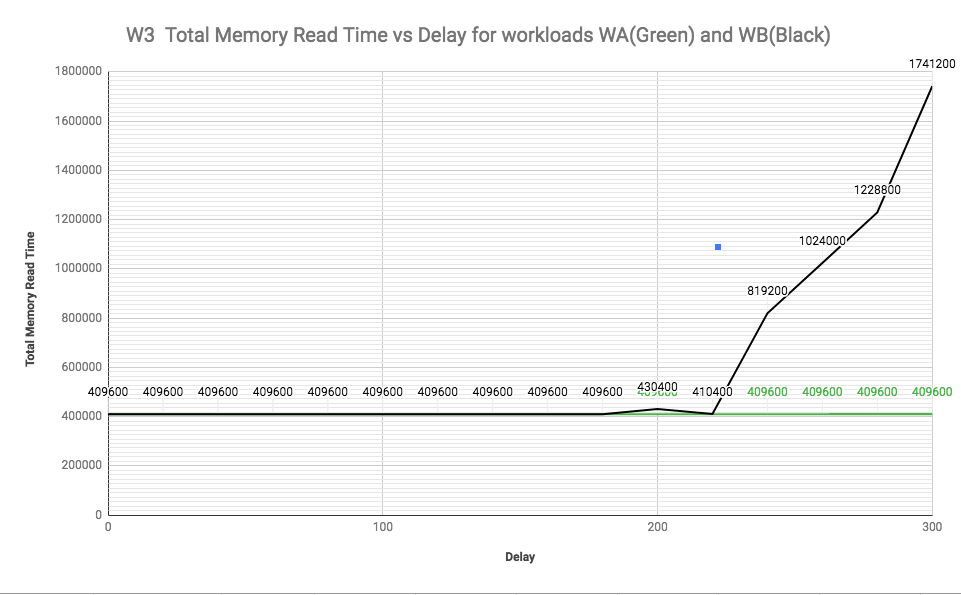


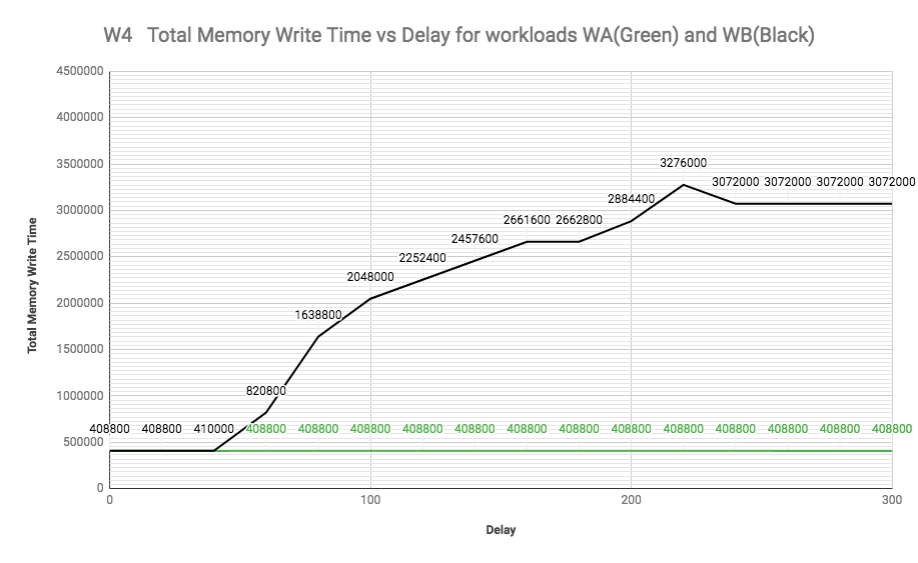


1. When the CPU delay gets bigger, it is obvious that the CPU utilization for both RA and RB will increase. As the total number of misses remains same for RA, the memory utilization will decrease on the contrast of CPU utilization increasement. At the same time, total number of misses increase for RB, there are more chance for miss and more chance for memory reading, then the memory utilization will increase for RB. When the delay parameter grows very large, the Memory utilization for RA would be 0, and the Memory utilization and CPU utilization for RA and RB would be 1.









1. As explained in question1, the increasement of CPU delay will lead to increasement of misses. The data is written into cache block when cache hit, and into memory when cache miss. So this time the total number of misses and total memory write time has the same curves. When the misses increase, the eviction and write back also increase. When misses occur, it will locate a cache block to use. And if it’s dirty, its previous data will be written back to memory, otherwise data will be read from memory into cache block. Only when the CPU delay gets large will the memory read time increases. So the total memory read time for RB will have a delay on the contrast of total number of misses.
2. For CPU delay of 100, the total time for WA and WA (concurrent design) are 1657032 and 2457603. If the two threads were run sequentially on CPU, the total running time for WA should be 1034240.