NC State University

Department of Electrical and Computer Engineering

ECE 463/521: Fall 2015 (Rotenberg)

Project #1: Cache Design, Memory Hierarchy Design

by

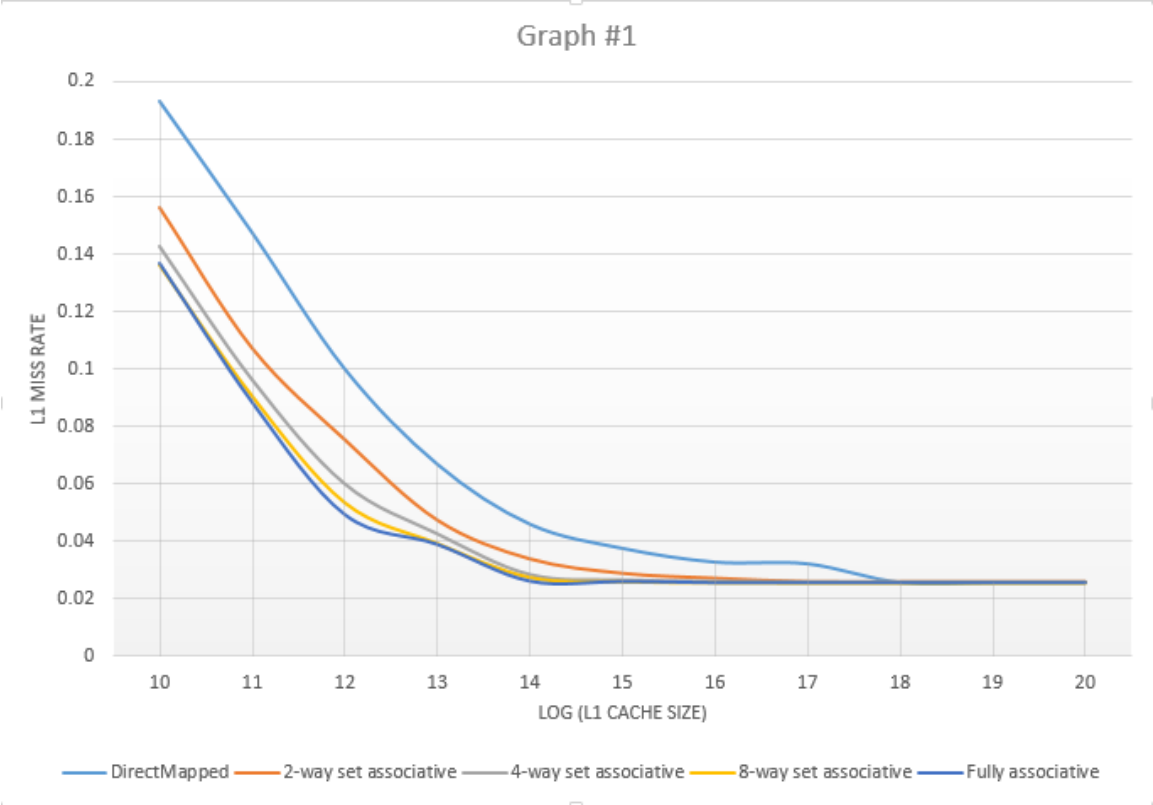
Denil Vira

NCSU Honor Pledge: "I have neither given nor received unauthorized aid on this test or assignment."

Student's electronic signature: _____Denil Vira_____
(sign by typing your name)

Course number: _____521____
(463 or 521 ?)

Graph # 1



1. Discuss trends in the graph. For a given associativity, how does increasing cache size affect miss rate? For a given cache size, what is the effect of increasing associativity?

For given associativity, increasing cache size leads to reduced miss rate. Increasing cache size eliminates capacity and conflict misses but not the compulsory misses. After certain point, in this case, once the L1 cache size is 2^18 bytes, increasing the size of cache beyond that does not reduce the miss rate since, the only misses remaining are the compulsory misses.

For the given cache size, increasing associativity reduces the miss rate. By increasing associativity, we get rid of conflict misses. Hence, the miss rate reduces as we move from direct mapped cache to a fully associative cache.

2. Estimate the compulsory miss rate from the graph.

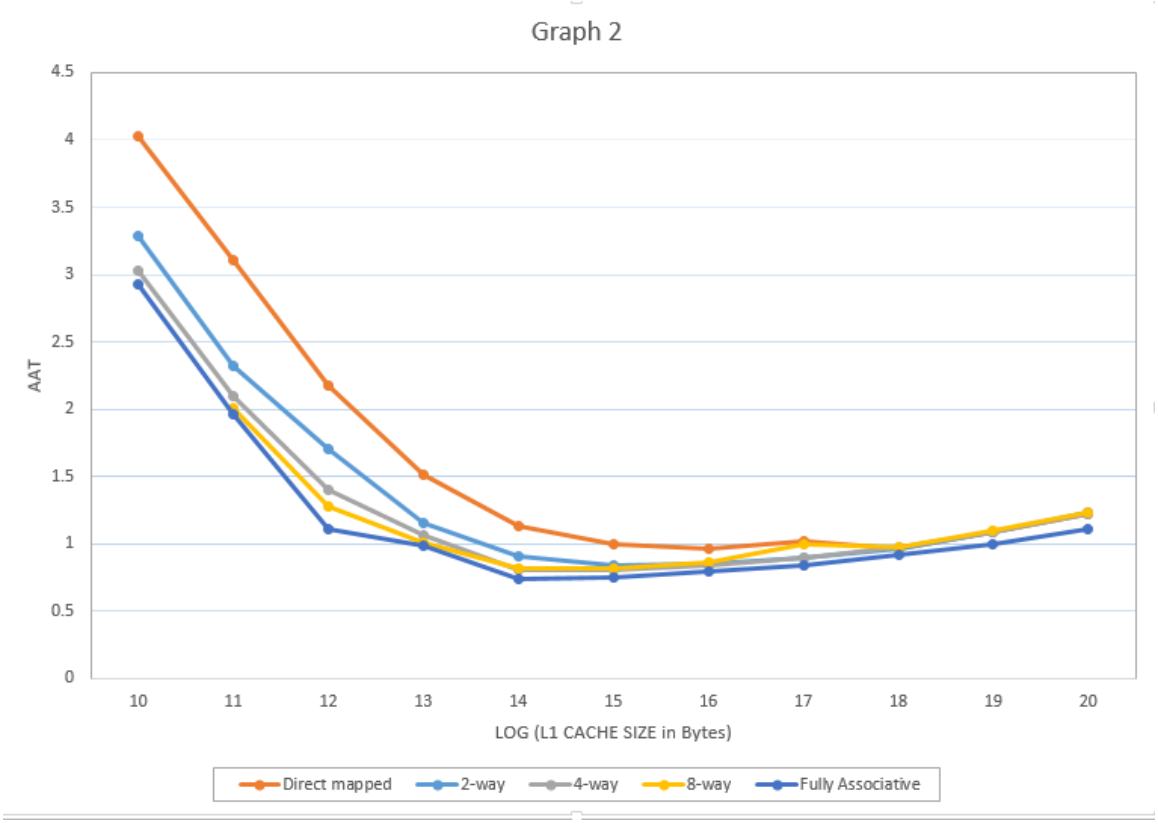
As described in point 1, we can observe from the graph that compulsory miss rate is about 0.025 since the miss rate stays constant irrespective of increasing the cache size.

3. For each associativity, estimate the conflict miss rate from the graph.

To get estimate of conflict miss rate, we compare each associativity with the full associative cache. A fully associatve cache has no conflict misses. So the average difference in miss rate between an associative cache and the fully associative cache of same size will give us the estimate of conflict miss rate for that associativity.

1. Direct mapped : 0.03
2. 2-way set associative : 0.01
3. 4-way set associative : 0.003
4. 8-way set associative : 0.0004

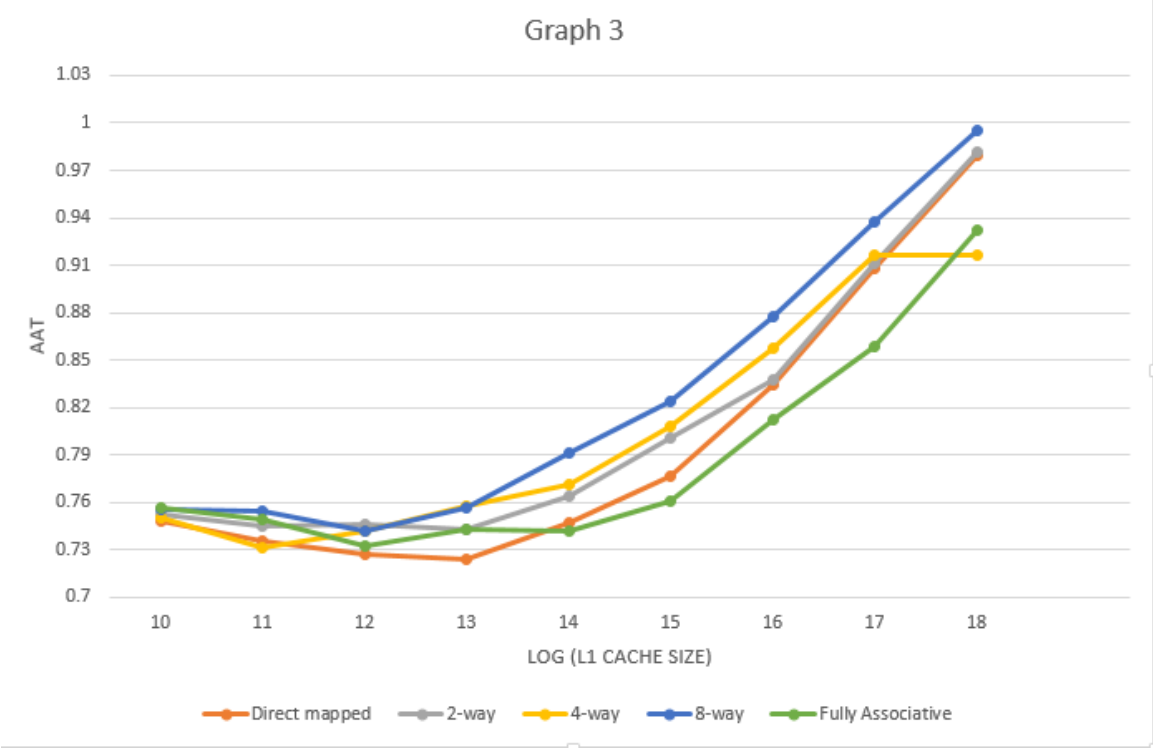
Graph # 2



1. For a memory hierarchy with only an L1 cache and BLOCKSIZE = 32, which configuration yields the best (i.e., lowest) AAT?

As seen from the graph, Cache size of 16KB and fully associative mapping leads to best (lowest) AAT.
This is most likely because the cache has less miss penalties occurring due to conflict and capacity misses.

Graph # 3



1. With the L2 cache added to the system, which L1 cache configurations result in AATs close to the best AAT observed in GRAPH #2 (e.g., within 5%)?

Following are some cache configurations with close to best AAT in Graph # 2
1. Fully associative L1 cache of 4KB size
2. 4-way set associative L1 cache of 2KB size
3. Direct mapped L1 cache of 2KB size

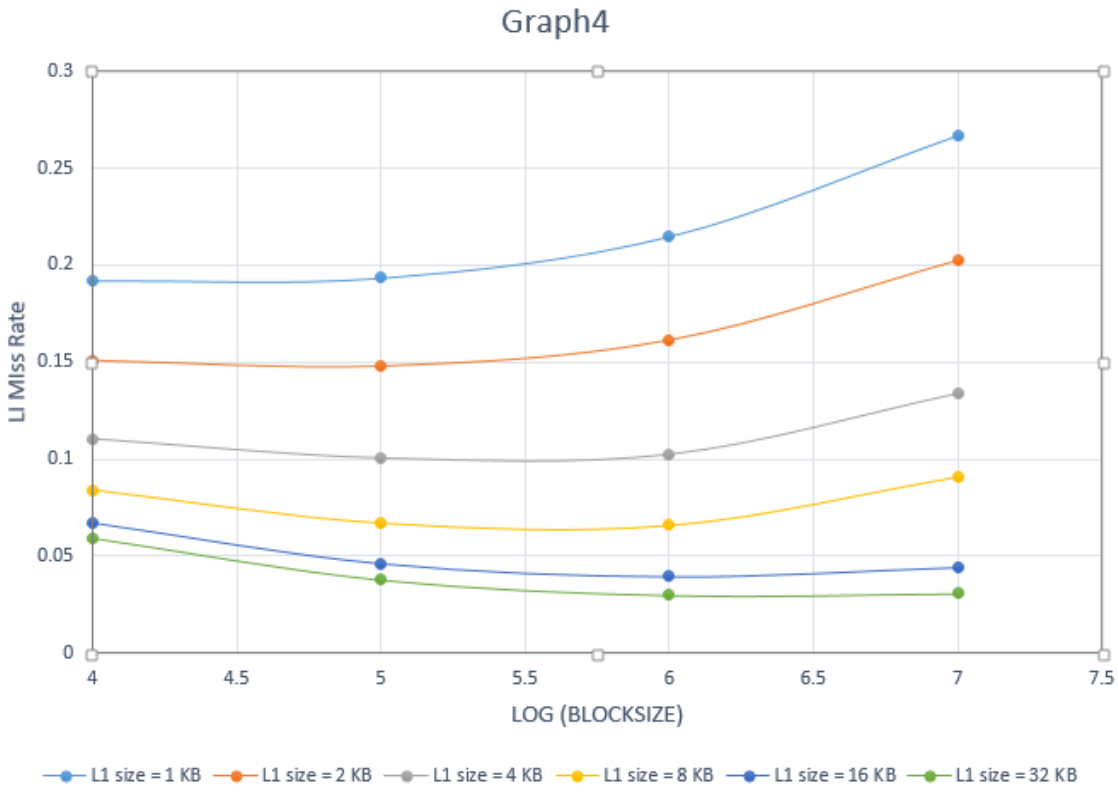
2. With the L2 cache added to the system, which L1 cache configuration yields the best (i.e., lowest) AAT? How much lower is this optimal AAT compared to the optimal AAT in GRAPH #2?

With L2 cache added, lowest AAT is 0.7241 cycles. 8KB direct mapped L1 cache yields the best AAT. The difference between the optimal configurations of graph#2 and #3 is 0.0128.

3. Compare the total area required for the optimal-AAT configurations with L2 cache (GRAPH #3) versus without L2 cache (GRAPH #2).

The area of cache in graph #2 is 0.063446019. I n Graph #3, we have both L1 and L2 caches. The area of L1 cache is 0.053293238 and the area of L2 cache is 2.640142073 thus the total area is addition of these 2 values = 2.6934 (approx).

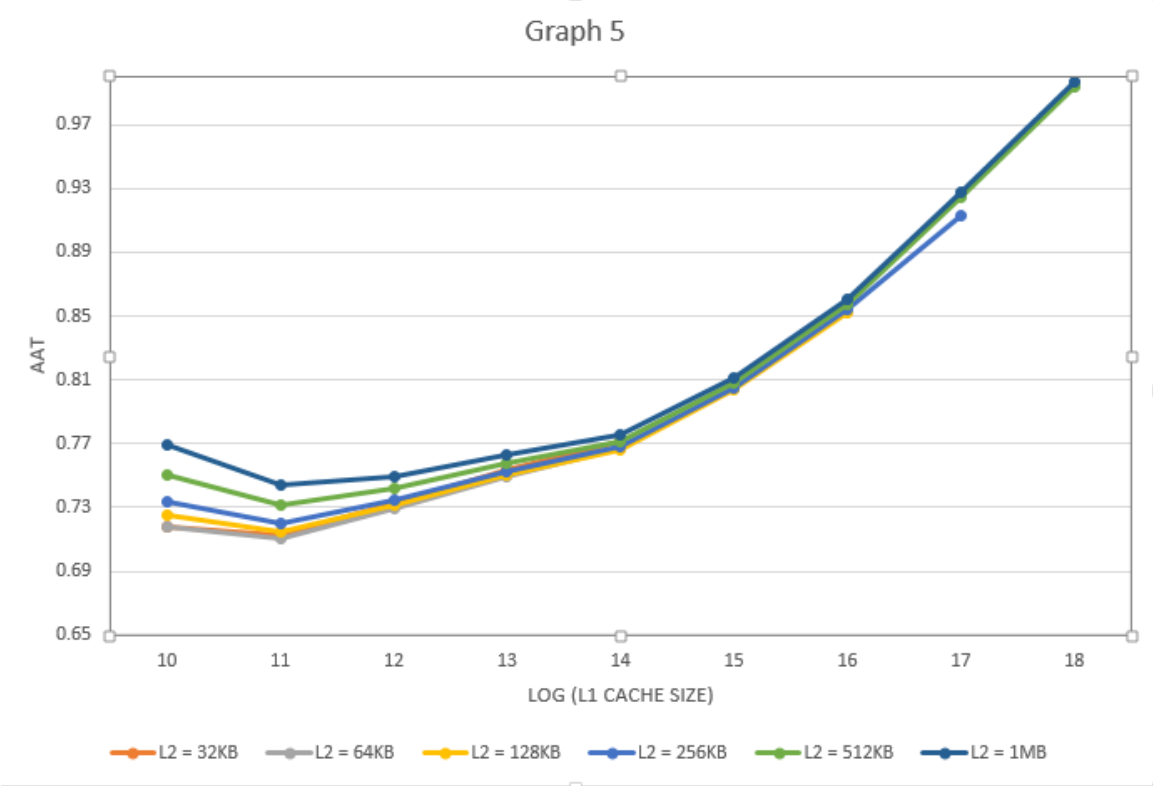
Graph #4



1. Discuss trends in the graph. Do smaller caches prefer smaller or larger block sizes? Do larger caches prefer smaller or larger block sizes? Why? As block size is increased from 16 to 128, is the tradeoff between exploiting more spatial locality versus increasing cache pollution evident in the graph, and does the balance between these two factors shift with different cache sizes?

As observed from the graph, smaller caches prefer smaller block sizes. Miss rate increases faster for smaller caches with increase in blocksize as compared to larger caches. Larger caches prefer larger block sizes to an extent since it allows to exploit more spatial locality. After certain point, increasing blocksize does not improve miss rate rather large blocksizes cause a phenomenon called cache pollution where in unnecessary blocks occupy the cache leading to higher miss rate. Yes, as the blocksize is increased from 16 to 128, the tradeoff between exploiting spatial locality versus cache pollution is evident from the graph. Yes, the balance shifts with different cache sizes. Smaller caches get polluted sooner than larger caches with increase of the block size.

Graph # 5



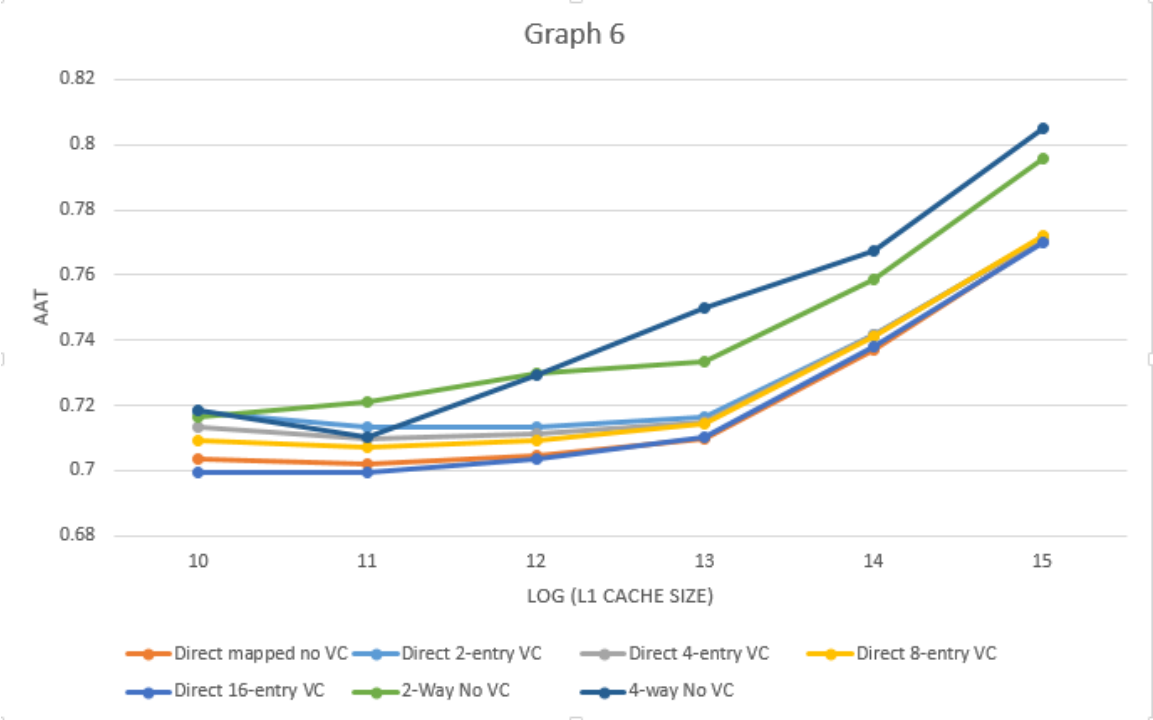
1. Which memory hierarchy configuration yields the best (i.e., lowest) AAT?

L1 : 2KB cache with 4-way set associative
L2 : 64KB 8-way set associative

2. Which memory hierarchy configuration has the smallest total area, that yields an AAT within 5% of the best AAT?

L1 : 1KB cache with 4-way set associative
L2 : 32KB cache with 8-way set associative

Graph # 6



1. Discuss trends in the graph. Does adding a Victim Cache to a direct-mapped L1 cache yield performance comparable to a 2-way set-associative L1 cache of the same size? …for which L1 cache sizes? …for how many Victim Cache entries?

Ideally a victim cache adds associativity to a direct mapped cache thus reducing the conflict misses and hence the AAT. However, the general trend in the graph is that adding Victim cache to direct mapped L1 does not yield performance comparable to 2-way set associative L1 cache of same size. This could possibly be due to nature of trace file where in addresses do not conflict and hence adding the victim cache provides no improvement in AAT. As seen from the graph, the green curve is 2-way set associative AAT and the direct mapped cache that yields the closest performance comparable to it is Direct mapped L1 cache with 2 VC entries (light blue curve).

2. Which memory hierarchy configuration yields the best (i.e., lowest) AAT?

The best value of AAT found is 0.7022 with following memory hierarchy with blocksize of 32 bytes -
L1 : 2KB direct mapped
L2 : 64KB 8-way Set associative

3. Which memory hierarchy configuration has the smallest total area, that yields an AAT within 5% of the best AAT?

AAT value of 0.7038 is found for following memory hierarchy with 32 Bytes blocksize :
L1: 1KB direct mapped

L2: 64KB 8-way set associative.