University of Central Florida

Department of Computer Science

CDA 5106: Spring 2020

Machine Problem 1: Cache Design, Memory Hierarchy Design

by

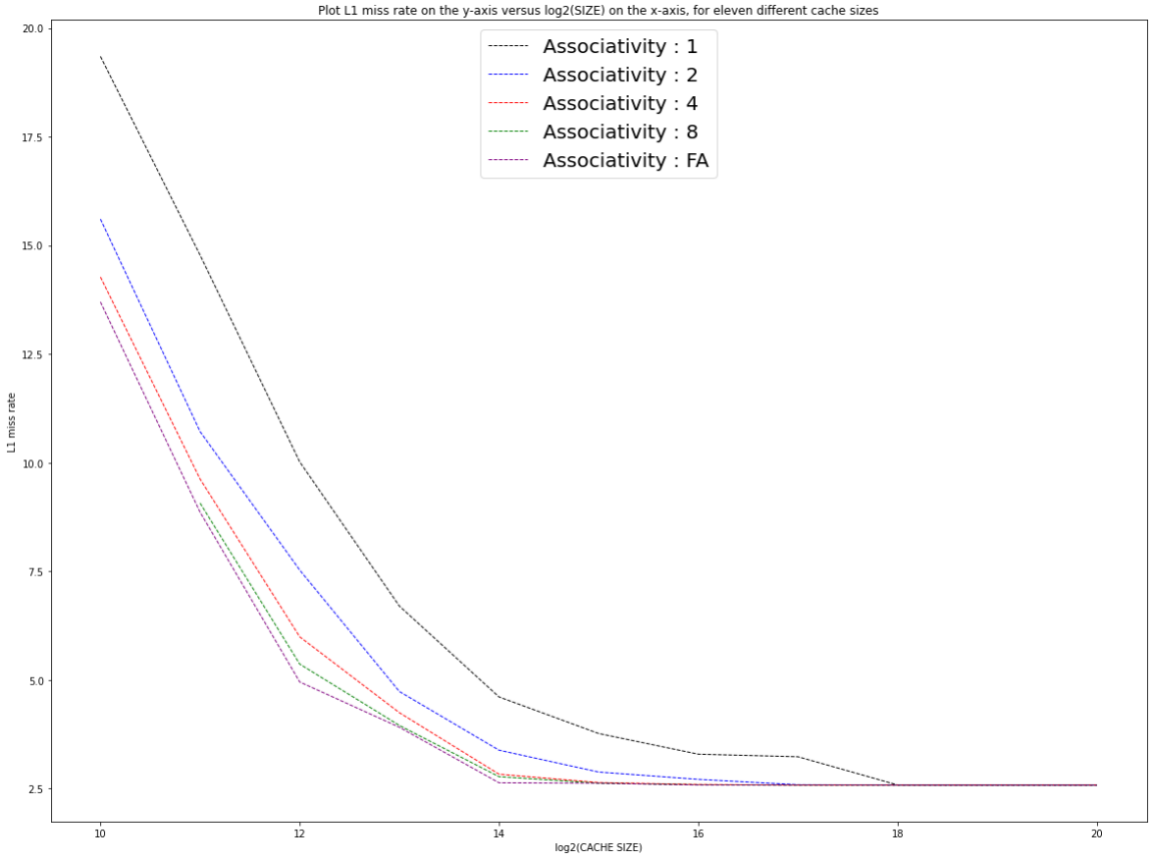
SUMANTH DHULIPALLA

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

Student's electronic signature: Sumanth Dhulipalla
(sign by typing your name)

L1 cache exploration: SIZE and ASSOC

GRAPH #1 (*total number of simulations: 54)*



- Effect of increasing cache size for given associativity on miss rate: For each given set associativity, the miss rate falls as the cache size grows until there are only mandatory misses left.
- The graph shows that there are diminishing returns, meaning that after a certain point, increasing the cache size has no effect on the cache's miss rate.
- Effect of different cache associativity with a fixed cache size on the miss rate: For a given cache size, the miss rate reduces as the associativity grows; a specific cache will have the lowest miss rate when it is completely associative and the highest loss rate when it is directly mapped.

The compulsory miss rate in the above graph is 0.02582.

Each associativity's conflict miss rate is deduced from the graph:
L1 cache size: 8kb

L1 size	Associativity	L1 miss rate
8	Direct mapping	0.067
8	2	0.04734
8	4	0.04247
8	8	0.03954
8	Fully associative	0.03912

Directly mapped : 0.067 - 0.03912 = 0.02788

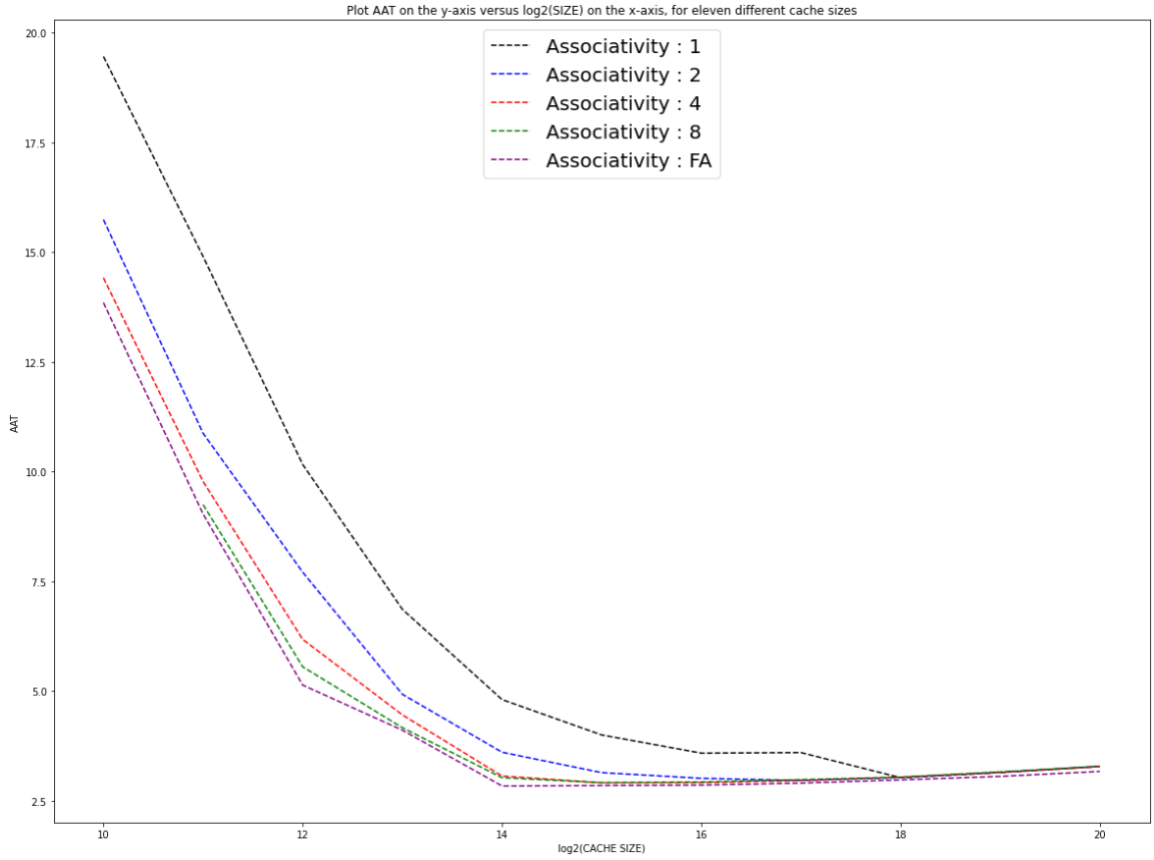
2 way set associative: 0.04734 – 0.03912 = 0.00822

4 way set-associative: 0.04247 – 0.03912 = 0.00335

8 way set-associative: 0.03954 – 0.03912 = 0.00042

Fully associative: 0.03912 – 0.03912 = 0

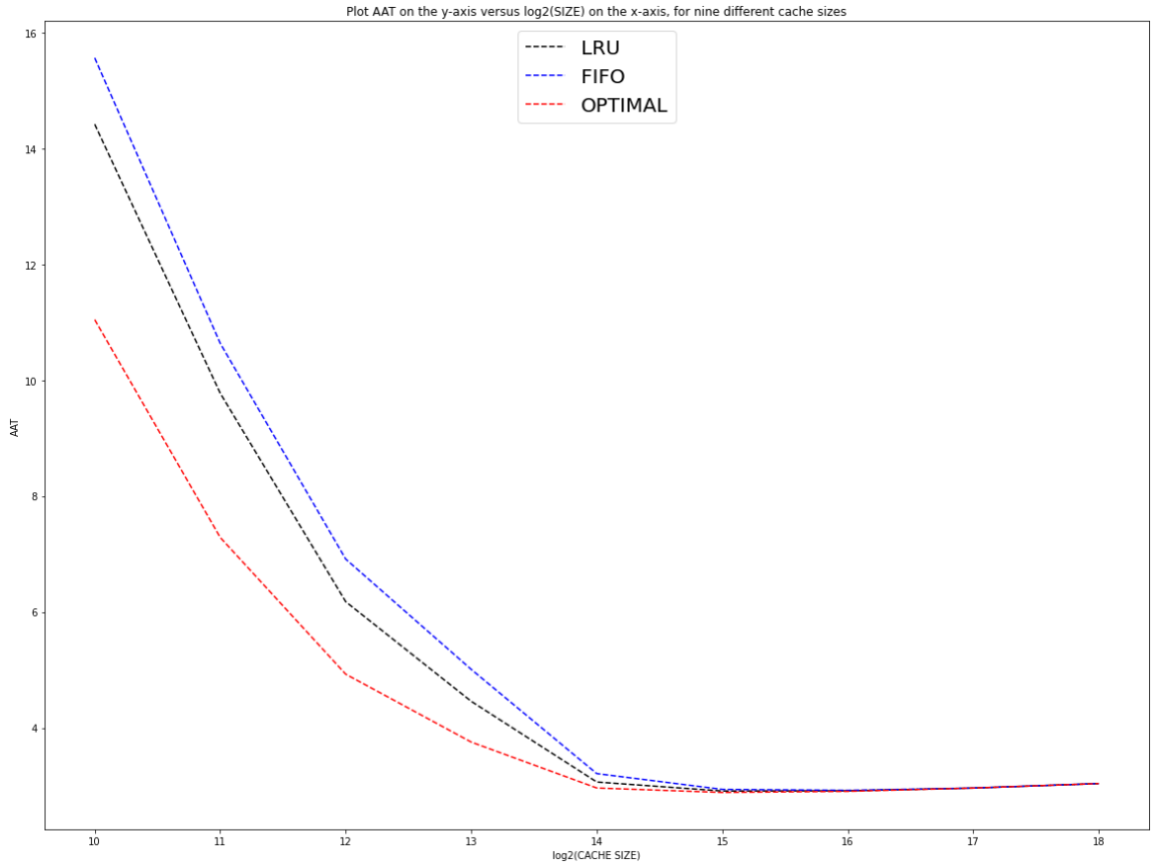
GRAPH #2 *(no additional simulations with respect to GRAPH #1)*



The best (i.e., lowest) AAT 2.83 is produced by a fully associative cache configuration for a memory hierarchy with only a 16kb L1 cache and a block size of 32.

Replacement policy study

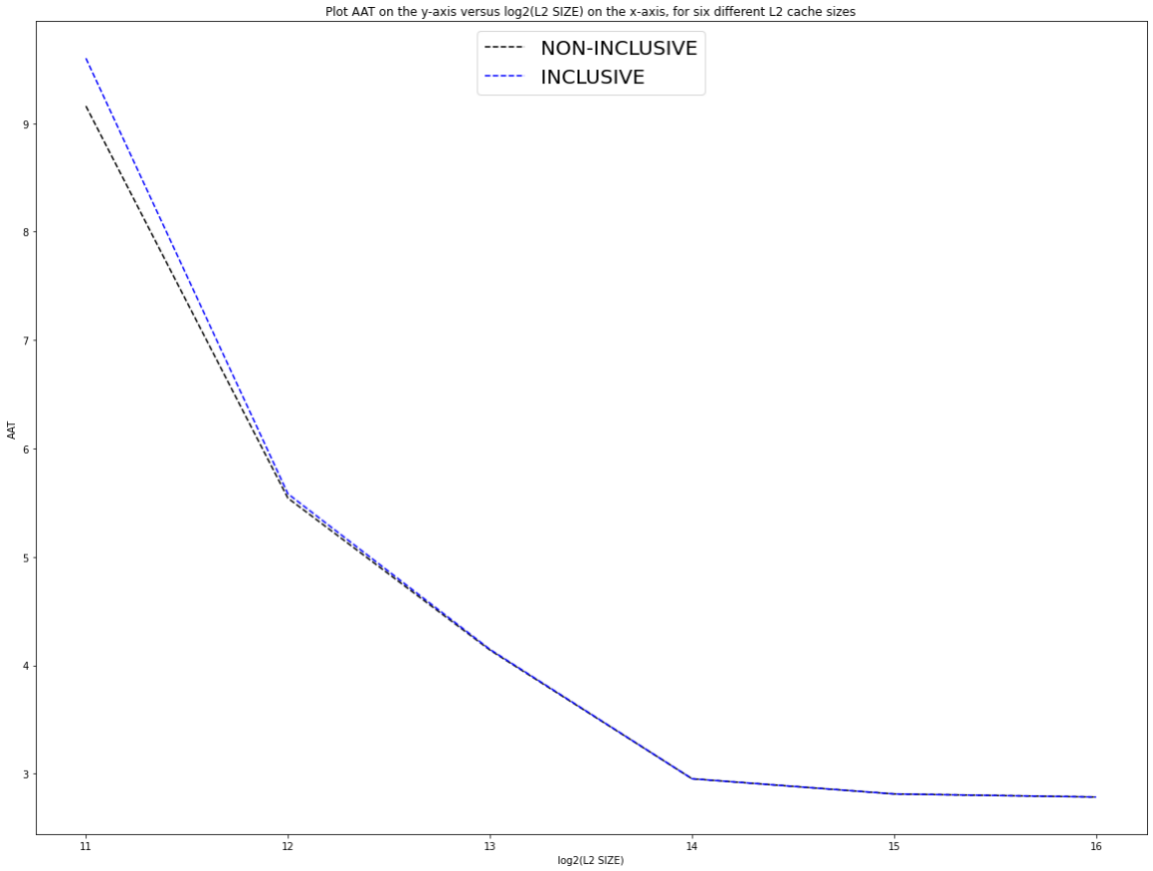
GRAPH #3 (*total number of simulations: 27)*



Cache size is inversely proportional to Average Access Time can be concluded from the above graph. The optimal replacement policy which is colored red in the above graph is showing the least Average Access Time (AAT) but that's only up to 128KB of cache size. After that, all the replacement policies are yielding the same AAT.

Inclusion property study

GRAPH #4 *(total number of simulations: 12)*



The Average Access time is higher for the Inclusive property in the beginning stages which makes it a bad choice when compared to the non-inclusive. Later, after the cache size is more than 16kb, both the inclusive and non-inclusive properties show the same AAT on the graph.