University of Central Florida

Department of Computer Science

CDA 5106: Fall 2020

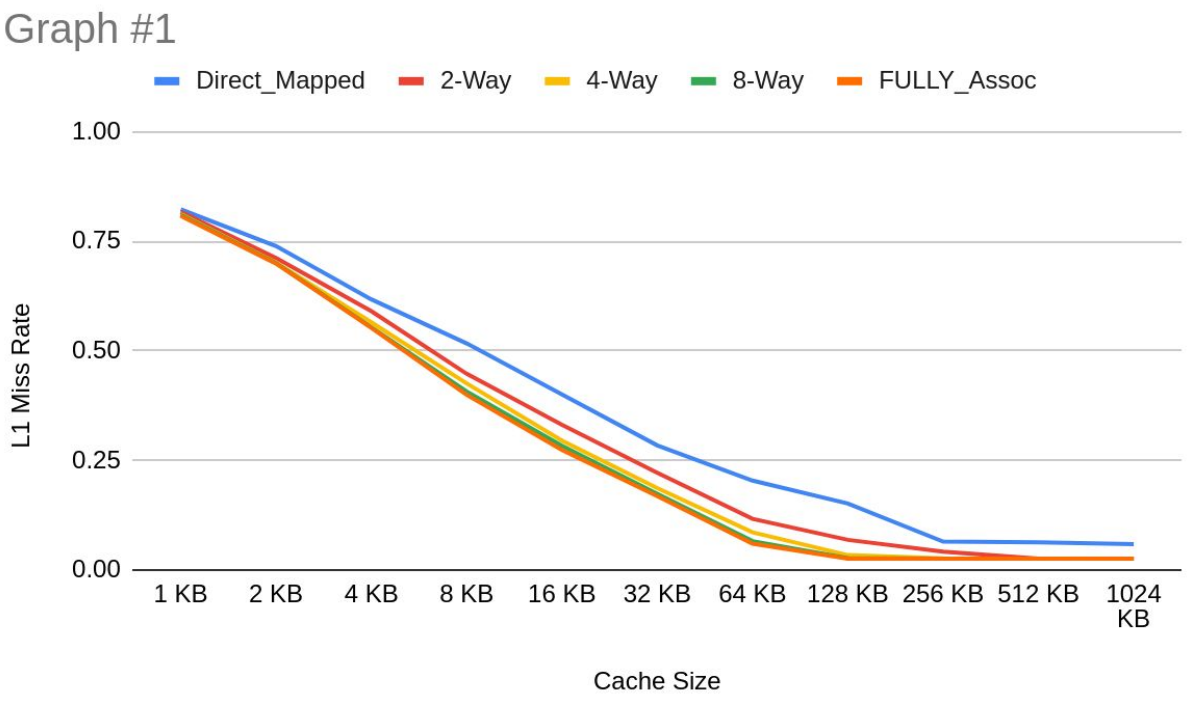
Machine Problem 1: Cache Design, Memory Hierarchy Design

by

Sheikh Nabil Mohammad

Honor Pledge: "I have neither given nor received unauthorized aid on this test or assignment."
Student's electronic signature: SHEIKH NABIL MOHAMMAD
(sign by typing your name)

Graph 1 (Plot L1 miss rate on y-axis vs log2(SIZE) on x-axis)



Size	Direct_Mapped	2-Way	4-Way	8-Way	FULLY_Assoc
1 KB	0.82278	0.81703	0.8118	0.81007	0.80795
2 KB	0.73898	0.7116	0.70135	0.7	0.69903
4 KB	0.61739	0.5905	0.5652	0.55339	0.55247
8 KB	0.51696	0.44773	0.42561	0.40708	0.39963
16 KB	0.40052	0.33129	0.29522	0.28328	0.27411
32 KB	0.28432	0.22179	0.18687	0.17399	0.1686
64 KB	0.20396	0.11673	0.08588	0.06583	0.06018
128 KB	0.15164	0.06871	0.03383	0.02726	0.02582
256 KB	0.06474	0.04193	0.02628	0.02582	0.02582
512 KB	0.06304	0.02582	0.02582	0.02582	0.02582
1024 KB	0.05871	0.02582	0.02582	0.02582	0.02582

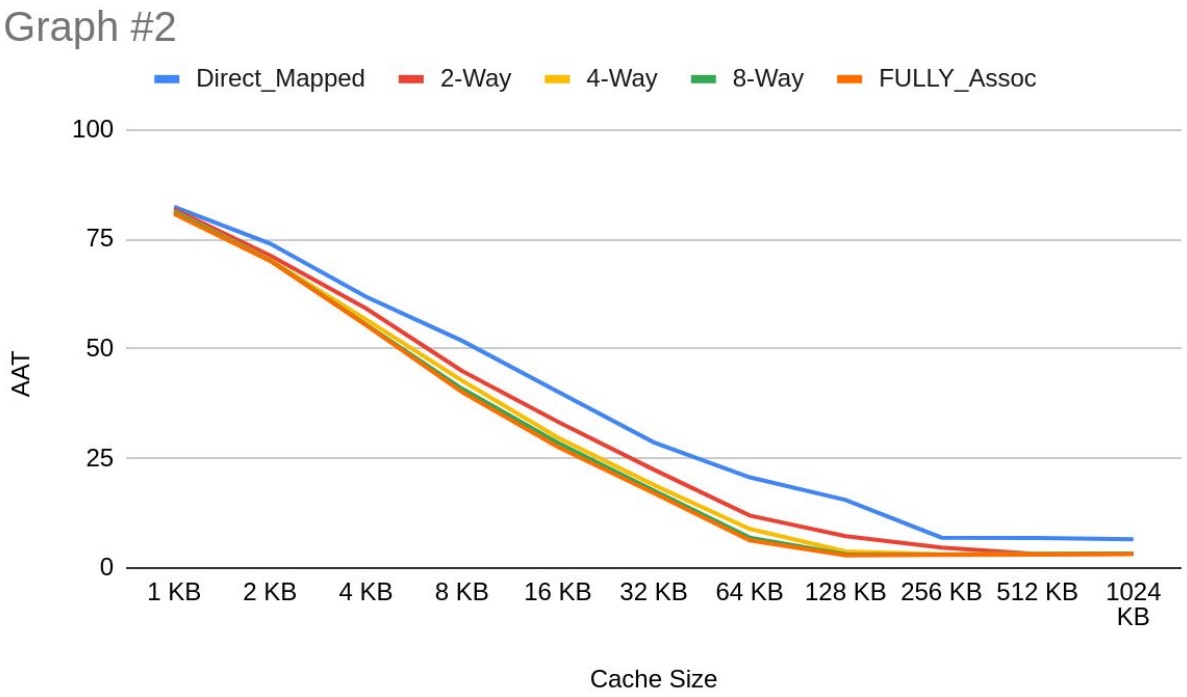
Discussion:

1) Here, from the above graph the miss rates gets lower as the cache size increases. Even though we change the associativity, increasing size always reduces the miss rates. And this trend is same for all the associativity we used in the experiment. For direct direct mapped cache, the L1 miss rate is higher than the other associativity. The most lower miss rates is for fully associative caches, but in the graph the changes is not too significant with the others such as 8-way associativity. But for direct mapped and L2, the change is very significant while increasing the size. Also, even though the points almost same for 1KB size, while we increase the trend differs. And direct mapped miss rate is always higher until the size is 256 KB. Even then its slightly higher then the other ones. Also, for any specific size below the 256 KB, direct mapped has significantly higher miss rate, then comes 2 way associativity. The rest of the associativity the line is pretty much similar.

2) From the above graph, the compulsory miss rate is around 0.81

3) Conflict miss for Direct mapped cache is ~ 0.015 for 1KB size, but it becomes higher when the size increases. So, in 64KB size, the conflict miss for Direct mapped is 0.1438, 2-way mapped is 0.057, 4-way mapped is 0.026 and 8 way is 0.006

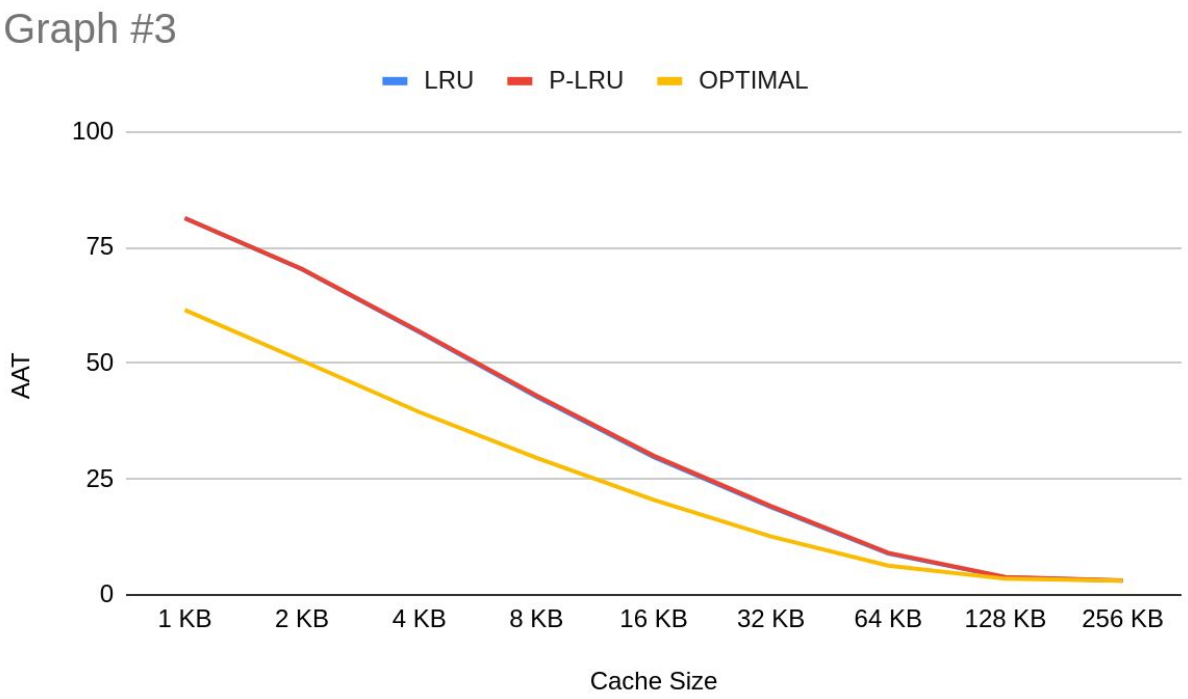
Graph 2 (Same as simulation 1, but with AAT values)



Size	Direct_Mapped	2-Way	4-Way	8-Way	FULLY_Assoc
1 KB	82.392797	81.843329	81.32682	81.162484	80.795
2 KB	74.02709	71.321691	70.289496	70.180686	70.079515
4 KB	61.886005	59.231131	56.705685	55.528065	55.429948
8 KB	51.85983	44.967195	42.772173	40.920911	40.161581
16 KB	40.250417	33.352917	29.755936	28.582354	27.616608
32 KB	28.665353	22.441446	18.95825	17.687511	17.08474
64 KB	20.690627	11.973727	8.907481	6.924213	6.294281
128 KB	15.5308	7.245603	3.76328	3.127236	2.904486
256 KB	6.917812	4.638929	3.085685	3.040925	2.978009
512 KB	6.867451	3.149744	3.146418	3.160177	3.057728
1024 KB	6.57038	3.288046	3.281607	3.287819	3.170474

1. For only L1 cache and block size 32, the Fully Associative cache has the best (lowest) AAT. Direct mapped is the worst, then 2 way and gradually the others.

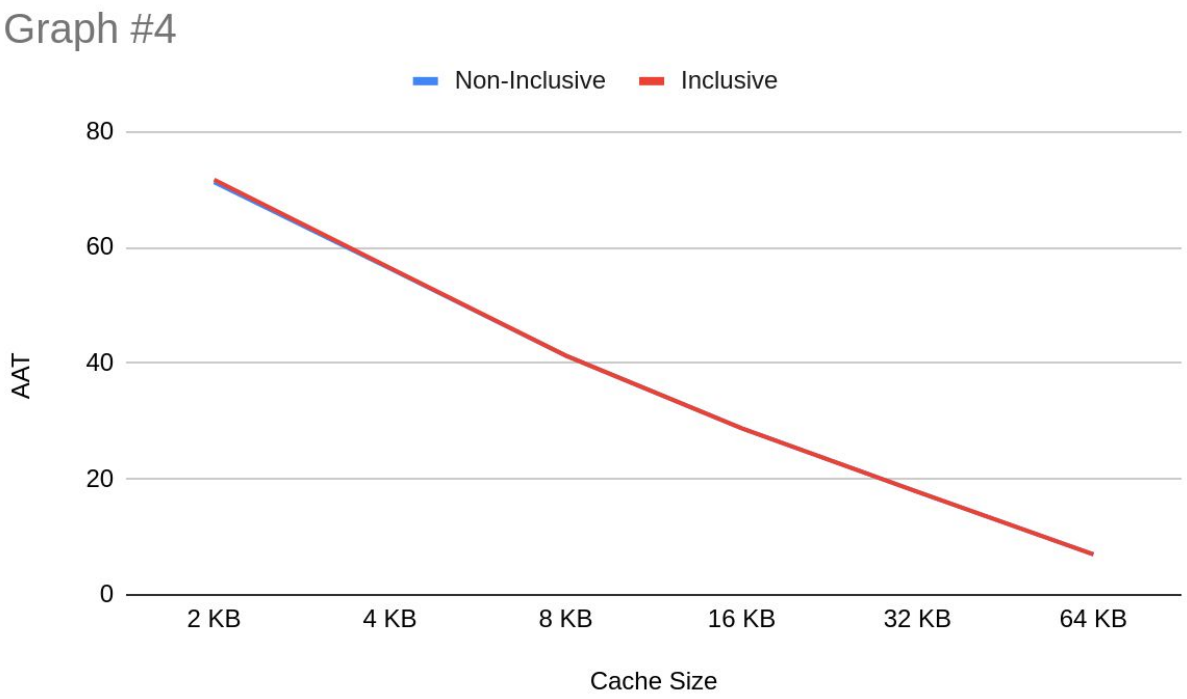
Graph 3 (Replacement policy study)



Size	LRU	P-LRU	OPTIMAL
1 KB	81.32682	81.33882	61.51882
2 KB	70.289496	70.347496	50.552496
4 KB	56.705685	56.877685	39.467685
8 KB	42.772173	43.051173	29.578173
16 KB	29.755936	30.024936	20.487936
32 KB	18.95825	19.14325	12.61525
64 KB	8.907481	9.069481	6.306481
128 KB	3.76328	3.83328	3.47928
256 KB	3.085685	3.087685	3.071685

1. The Optimal replacement policy shows the lowest AAT. The LRU and Pseudo-LRU both show almost same AAT which is higher (worse) than Optimal replacement policy.

Graph 4 (Inclusion policy Study)



Size	Non-Inclusive	Inclusive
2 KB	71.29750671	71.7209157
4 KB	56.34827385	56.48244251
8 KB	41.32367915	41.32670299
16 KB	28.76427508	28.76427508
32 KB	17.79406205	17.79406205
64 KB	7.009787753	7.009787753

1. The Non-Inclusive cache shows better AAT for lower cache size, but when we increase the cache size, the difference is not much significant. And in the graph both policy almost shows similar result, while Non-Inclusive cache has slightly less AAT for lower cache size.