Experiment Project 1

Introduction

In this part of the project, we are supposed to find the smallest AAT for two different work load. To find the optimal configuration, we have to use the tricks provided by Dr. Conte in his notes about the hit time reduction and mis rate reduction.

Process

There are certain things in this cache that can be changed so that we can achive the best configuration for our workloads.

- Hit Time : based on the size and the cache type hit time can be lowered
- Miss Rate VC : depending of the size of the VC, and the program, VC miss rate can be changed
- Miss Rate L1: Miss rate depends on Block Size, Cache Size, and associativity.
- Replacement policy: Replacement policy can change misses significantly based on the program.

Based on the table for the Hit time we can see that as we go toward more associativity it increases the Hit time by significant margin. This means we can't just pick A max size fully associative cache to solve this problem. So next best option is to use the 8 way or 4 way. But compare to direct map they also have higher hit rate. Also, replacement policies and VC size are more based on the program specific. Meaning, starting with them will not provide enough evidence in AAT change if we don't change the Main parameters of the cache.

So, what we can do is start with the most insufficient configuration and move forward form it. Most basic configuration is c=9, b=4, s=0 with no victim cache and FIFO rp. We chose this configuration because direct mapped cache has hit time of 1 cycle which is the lowest of all. This is the output of the simulator.

L1 Data Hit Time	1.00000000
L1 Data Miss Penalty	60.00000000
L1 Data Miss Rate	0.11040358
Victim Cache Miss Rate	0.00000000
Average Access Time	7.62421494

Figure 1 mm_basic_192.trace

L1 Data Hit Time	1.00000000
L1 Data Miss Penalty	60.00000000
L1 Data Miss Rate	0.09997610
Victim Cache Miss Rate	0.00000000
Average Access Time	6.99856609

Figure 2 mm_tiled_192_4.trace

Now lets see what happens when we add the victim cache to this and then increase the cache size from this model and try to find some trend to figure out what is best for the given two workloads.

After adding the VC of size 1 AAT reduces for both of the work loadsby about 2.5 – 3 cycles, which is really great so lets try to increase the VCsize and se if it goes even below.

L1 Data Hit Time	1.00000000
L1 Data Miss Penalty	60.00000000
L1 Data Miss Rate	0.11040358
Victim Cache Miss Rate	0.63917219
Average Access Time	5.23401399

L1 Data Hit Time	1.00000000
L1 Data Miss Penalty	60.00000000
L1 Data Miss Rate	0.09997610
Victim Cache Miss Rate	0.65327970
Average Access Time	4.91874149

Figure 2 after VC of size 1 included

As we increase the size of the VC we are able to reduce the AAT upto 3.72 cycyles for mm_basic and 3.031 for the mm_tiled workload. Now lets increase the C and see if its reduces AAT at all, since it affect both the hit time and miss rate penalty.

As we increase the c = 10 we found a small decrease by 0.5-0.3 cycle in bothof the work load but increasing it to 11 make it go higher than before that mean if we increase the cache size it not going to help improving the AAT anymore. So now lets try to increase the `s` and see if it changes anything, since increasing or decreasing the block size can be a hassle.

Increasing Associativity does not help to reduce the AAT by making s = 1 it increases the AAT By 2 cycles which is not a good trade off since we can achieve less AAT by direct mapped cache. Since, associativity doesn't help we can try to change the block size and try to fid the sweet spot.

Increasing the b value does help improving the AAT for the mm_basic it reduce it to almost 3 cycles, but for the mm_tiled version the original config works better with the c=10 and v = 4 its AAT becomes 2.5 cycles.

Results

Here are the result for the configurations discussed above. This configuration suggests that this program has higher temporal locality direct map caches works better when you have temporal locality instead of higher spatial locality. Also Tiled version required 2 less bits because second version is tied by 4, so it can access 4 block at same time when we run it, which allows us toreduce 2 more bits from the offset bits, thus making it much faster than first version.

```
SIMULATION CONFIGURATION
L1 Data Cache:          (C=10, B=4, S=0)
Replacement Policy:     LFU
Victim Cache:           yes; V:4
SETUP COMPLETE - STARTING SIMULATION

SIMULATION OUTPUT
L1 Data Accesses                    256928742
L1 Data Load Accesses               238123396
L1 Data Store Accesses              18805346
L1 Data Misses                      18426158
L1 Data Load Misses                 14336745
L1 Data Store Misses                4089413
L1 Data Evictions                   18426094
Victim Cache Accesses               18426158
Victim Cache Misses                 6250260
Victim Cache Hits                   12175898
Number of writebacks                884783
Bytes transferred from Main Memory  100004160
Bytes transferred to Main Memory    14156528
L1 Data Hit Time                    1.00000000
L1 Data Miss Penalty                60.00000000
L1 Data Miss Rate                   0.07171700
Victim Cache Miss Rate              0.33920582
Average Access Time                 2.45960937
```

```
SIMULATION CONFIGURATION
L1 Data Cache:          (C=10, B=6, S=0)
Replacement Policy:     LFU
Victim Cache:           yes; V:4
SETUP COMPLETE - STARTING SIMULATION

SIMULATION OUTPUT
L1 Data Accesses                    233903814
L1 Data Load Accesses               219673924
L1 Data Store Accesses              14229890
L1 Data Misses                      21880517
L1 Data Load Misses                 17274821
L1 Data Store Misses                4605696
L1 Data Evictions                   21880501
Victim Cache Accesses               21880517
Victim Cache Misses                 7846470
Victim Cache Hits                   14034047
Number of writebacks                4608
Bytes transferred from Main Memory  502174080
Bytes transferred to Main Memory    294912
L1 Data Hit Time                    1.00000000
L1 Data Miss Penalty                60.00000000
L1 Data Miss Rate                   0.09354493
Victim Cache Miss Rate              0.35860533
Average Access Time                 3.01274273
```

Figure 3 mm_basic_192.trace(right) & mm_tiled_192_4(left) final configuration.