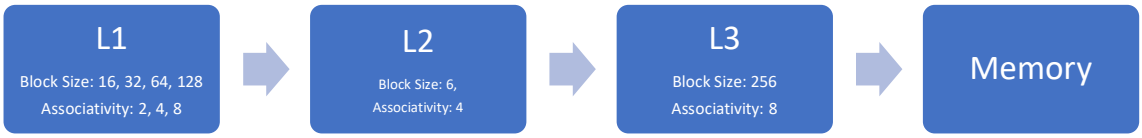
Assignment 1 - Cache System Performance Analysis

by Aakash Wagle – aaw5833

Cache Block Diagram



This report presents a condensed version of 100+ simulation datapoints used for analyzing the cache system performance under various configurations and workloads.

Note: All the simulation outputs are stored in 'Records' folder as .txt files with a serial no. which correspond to the records in the column titled '#' in the .xlsx file, 'Data with Graphs'. Simply search for that record in the txt file using #{sr_num}, for example #33.

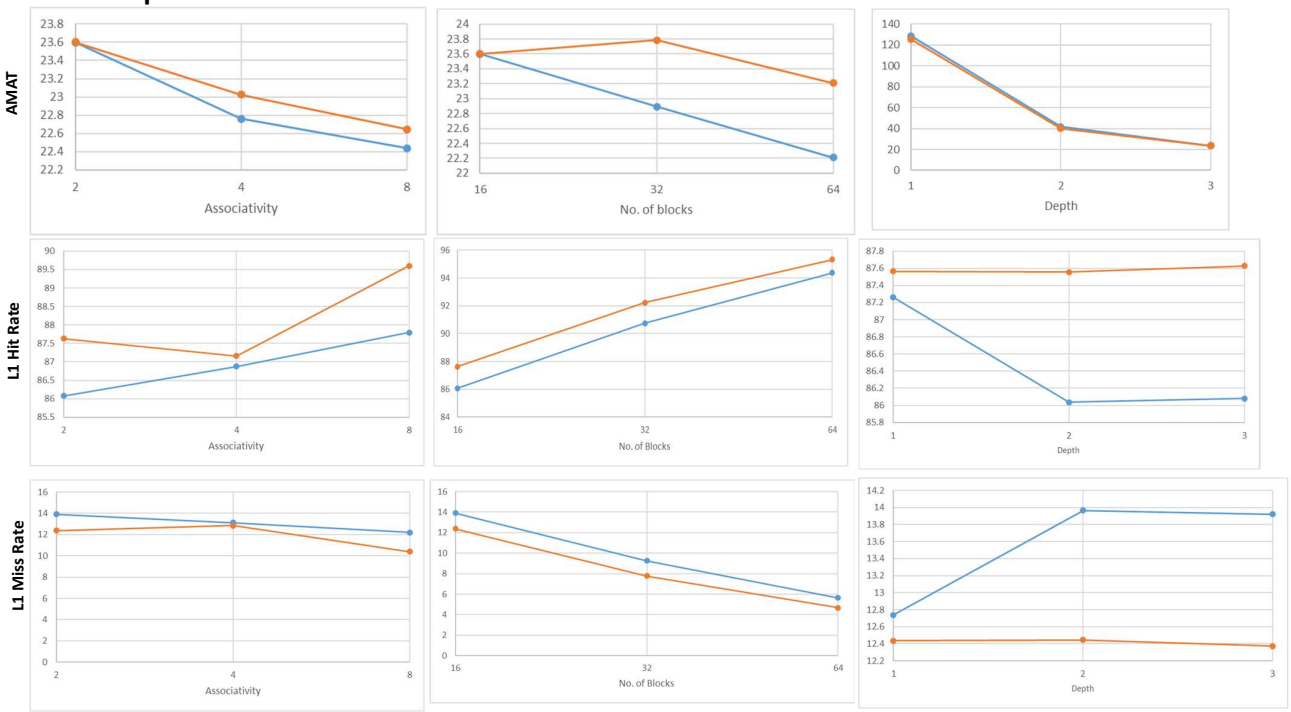
Assumptions made:
For the sake of simplicity and limiting the number variations, only knobs in L1 are varied.
Matrix size: 128x128 and 150x150 for scatter and gather operations; 64x64 and 75x75 for convolution operations
Matrix Sparsity: 50 and 100 for all the operations except when testing the effect of sparsity itself.
Default configuration:

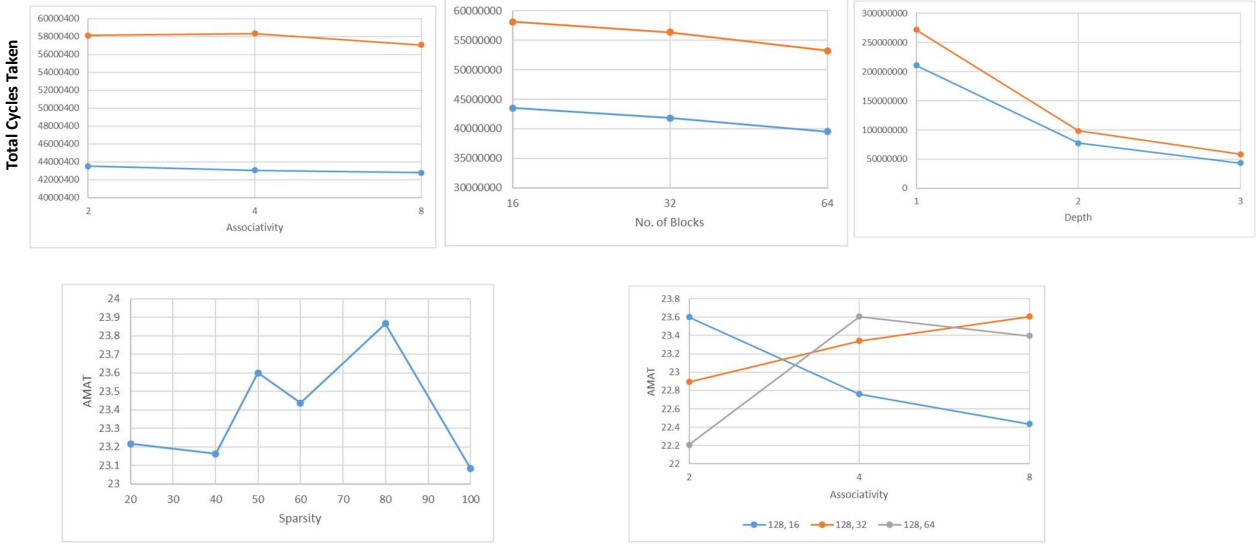
Sparsity	Depth	L1		L2		L3	
		No. of Blocks	Associativity	No. of Blocks	Associativity	No. of Blocks	Associativity
50	3	16	2	64	4	256	8

Legend 128 150 *(for convolution: blue line is 64 and the orange one is 75)*

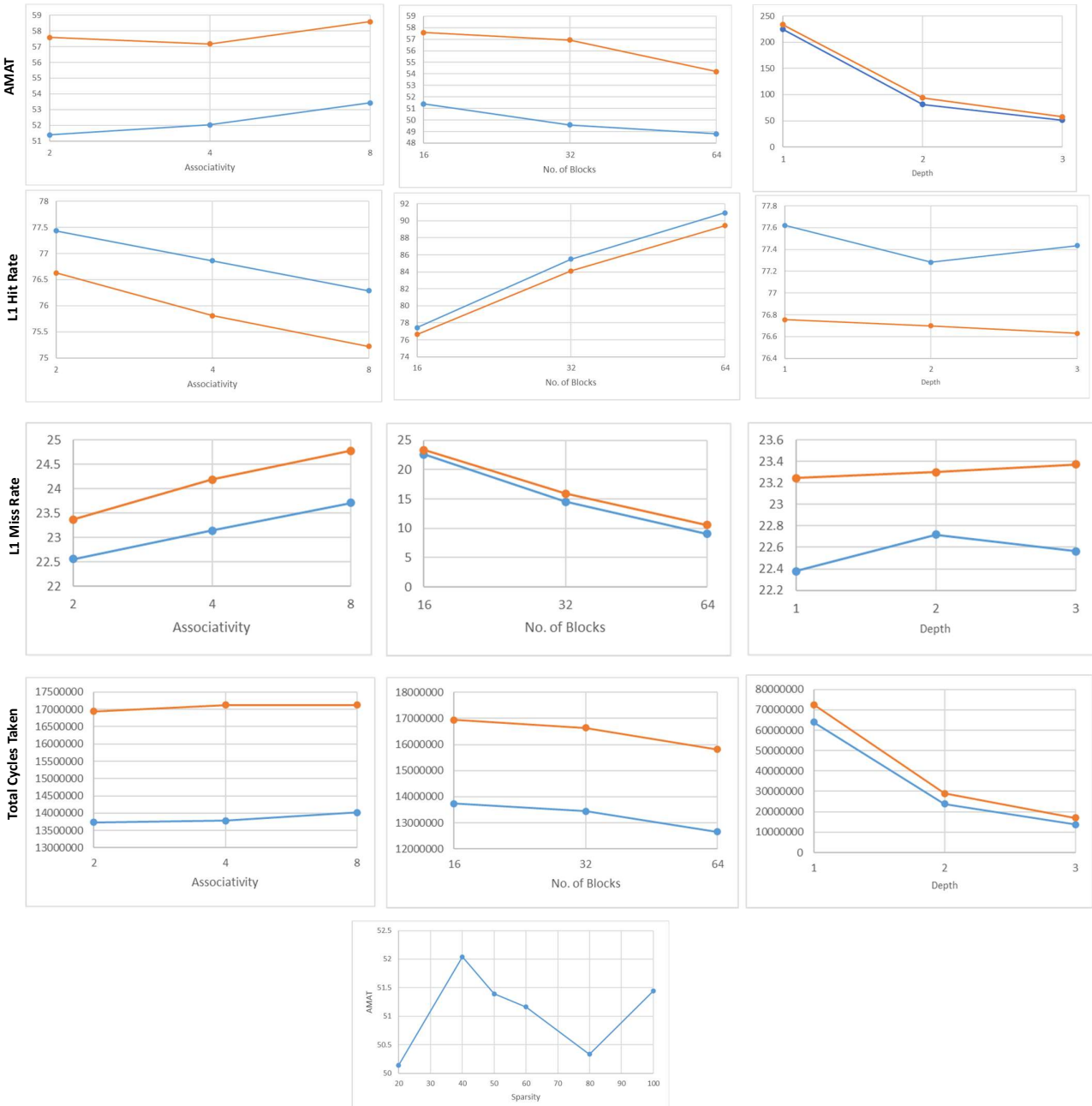
Observations

Scatter Operation

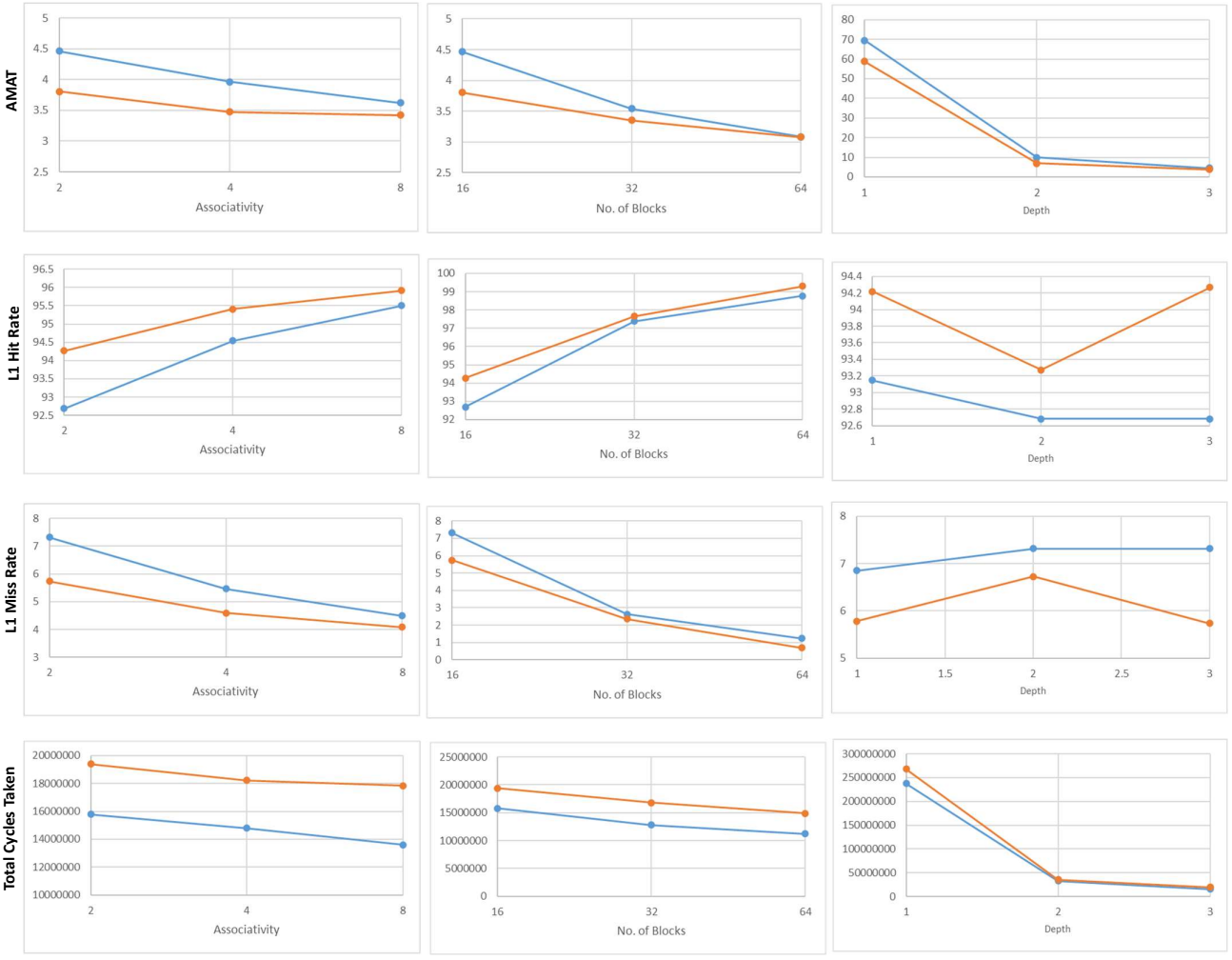


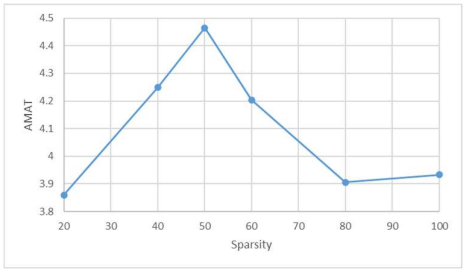


Gather Operation



Convolution Operation





Inference

Generalized effect of the cache parameters:

Number of Blocks:

Effect: Increasing the number of blocks generally decreases AMAT.

Reason: Since more blocks allow the cache to store more data, the frequency of cache misses reduces, improving hit rates.

Depth

Effect: Increasing cache depth generally decreases AMAT.

Reason: Multi-level caches can better exploit different types of locality, ensuring that frequently accessed data is closer to the processor.

Associativity

Effect: Higher associativity usually decreases AMAT.

Reason: Increased associativity increases the options where cache lines can be placed. This flexibility reduces conflict misses.

Scatter Operation:
As the number of blocks, associativity, and depth increase, AMAT and total cycles decrease, L1 hit rates increase and L1 miss rates decrease. This trend is observed for both 128x128 and 150x150 matrix sizes, but the larger matrix shows slightly higher AMAT and miss rates due to its increased data volume and it not being a power of 2, hence not benefitting from the binary math. This suggests that scatter operations on larger datasets may benefit more from optimized cache structures. It may be due to better handling of the increased working set size.

Gather Operation:
The trends in gather operation exhibits some outliers. Unlike scatter, AMAT and total cycles tend to increase with higher associativity, contrary to typical cache behavior. L1 hit rates decrease and miss rates increase with higher associativity, which is unusual and suggests inefficient use of the cache. This could be due to the scattered nature of memory accesses, which may not benefit from increased associativity and could suffer from longer lookup times. The larger working set of the 150x150 matrix amplifies this effect.

Convolution Operation:
Convolution shows more consistent and predictable behavior compared to scatter and gather operations. As cache parameters improve, performance metrics generally improve for both matrix sizes, with the 150x150 matrix showing slightly higher AMAT and miss rates due to its larger size and due to it not being a power of 2. The convolution operation demonstrates smoother improvements in hit rates and more consistent decreases in miss rates compared to scatter and gather. This might be due to its more structured memory access patterns and symmetric nature of the kernels which may increase the chance of hits.

Exploring the effects of sparsity:
The sparsity increases at the beginning, from 20% to 50%. This may be due to reduced spatial locality as more elements become zero and the non-zero elements are more scattered. The peak at 50% might be due to equal amount of non-zero elements and gaps(zeroes), causing frequent misses. Further reduction in sparsity reduces AMAT. This might be due to reduction in non-zero elements leading to lesser conflict misses.

Bonus Question: How many times can you execute an operation for endurance of 10^6?

Operation	Writes	No. of Times
Scatter (128)	581983	1.718263
Scatter (150)	771215	1.296655
Gather (128)	101223	9.879178
Gather (150)	111403	8.976419
Convolution (64)	1005718	0.994315
Convolution (75)	1384680	0.722189