NC State University

Department of Electrical and Computer Engineering

ECE 506: Architecture of Parallel Computers Fall 2021

Project #2: SMP simulator (shared multiprocessor simulator)

by

Zong-Ru Li

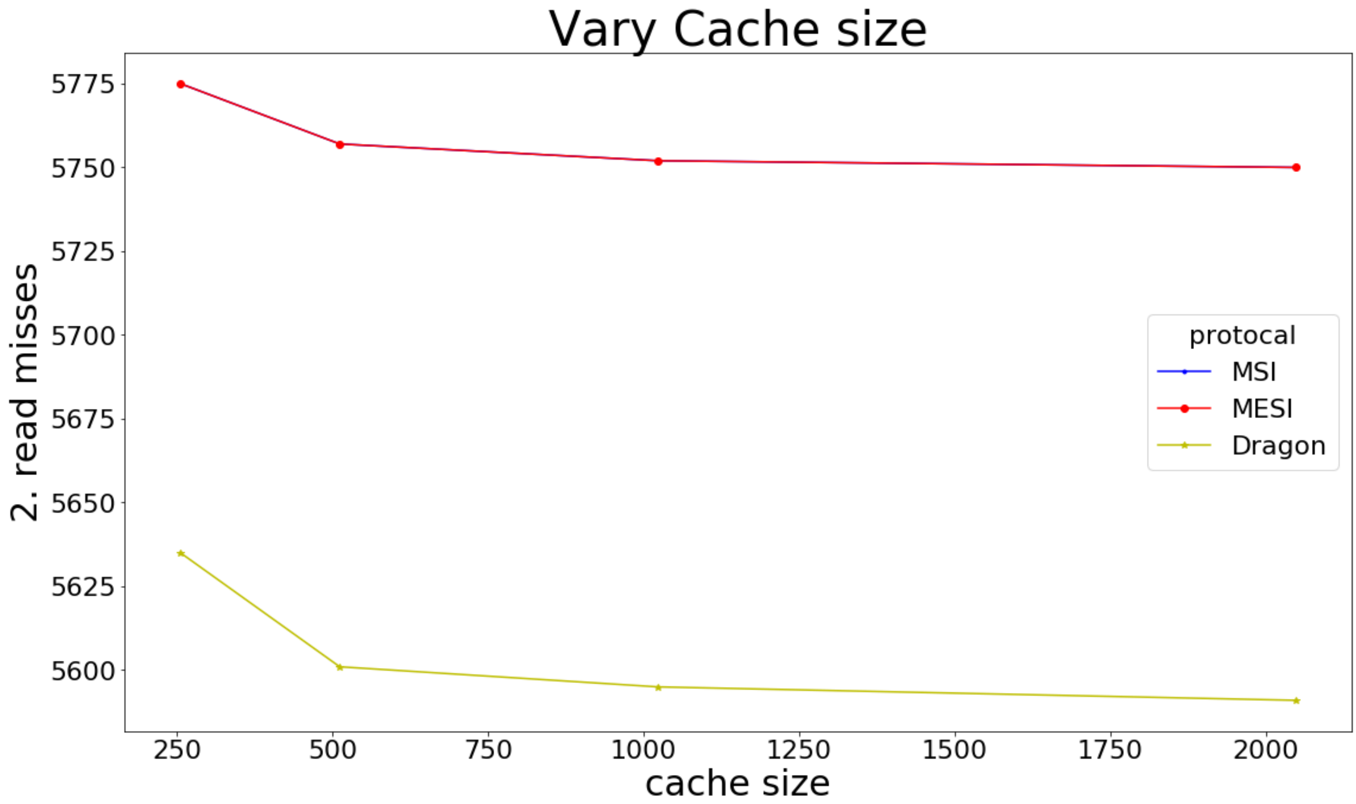
Table of Contents

1. Read Misses, write misses, and Total miss rate

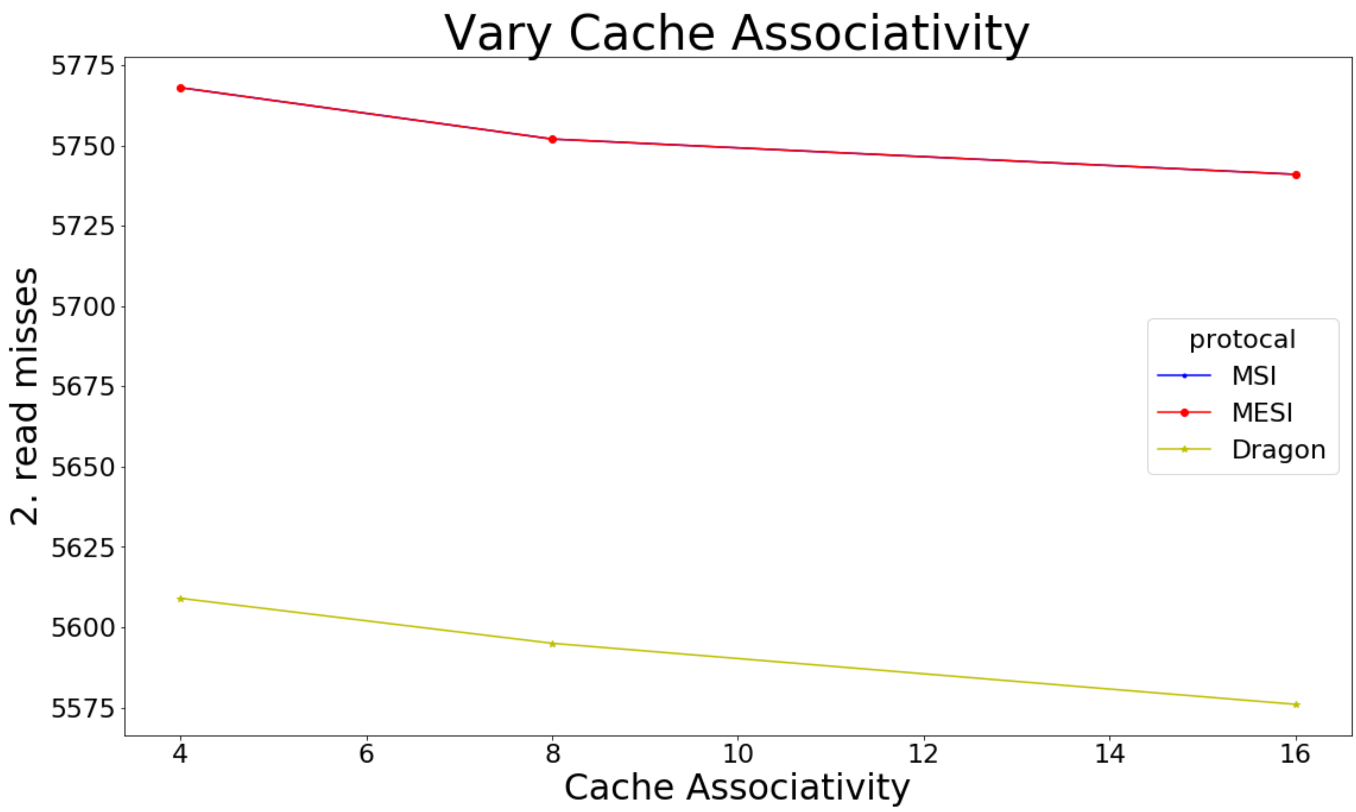
According to results, Dragon protocol has a smaller number of read misses. MSI and MESI protocol has the same number of read misses. Dragon protocol is an update-based protocol. Dragon protocol does not invalidate cache blocks in other processors. Instead, it updates cache blocks in other processors. Therefore, Dragon protocol maintains more valid cache blocks in cache and has a smaller number of read misses.

When we only increase cache size, read misses decrease. Read misses will end up decreasing at compulsory misses. There will be no conflict miss and no size miss. When we only increase cache associativity, read miss will end up decreasing at compulsory misses + size misses. There will be no conflict misses. When only cache block size increases, read miss decreases. The amount of decrease depends on whether there is more access contained in one cache block.

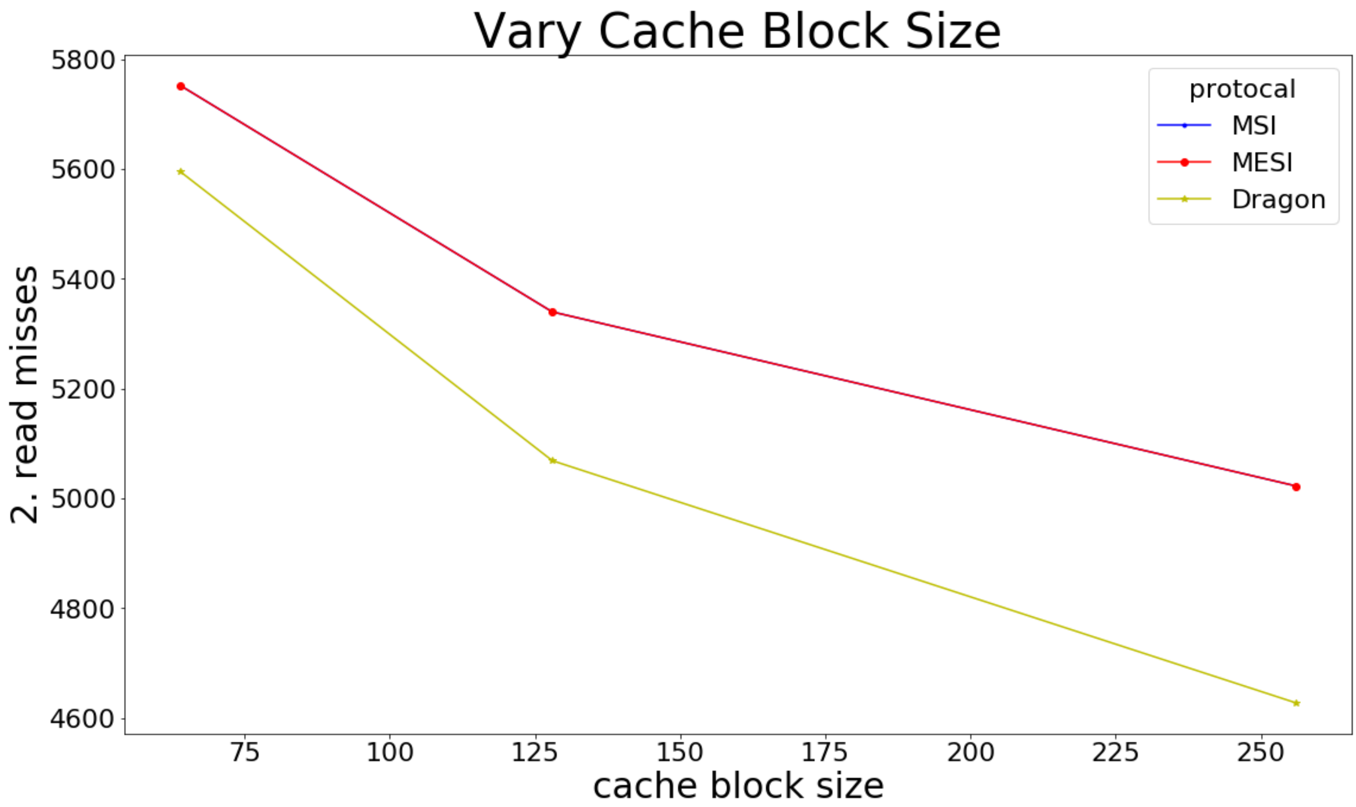
Graph 1.1



Graph 1.2



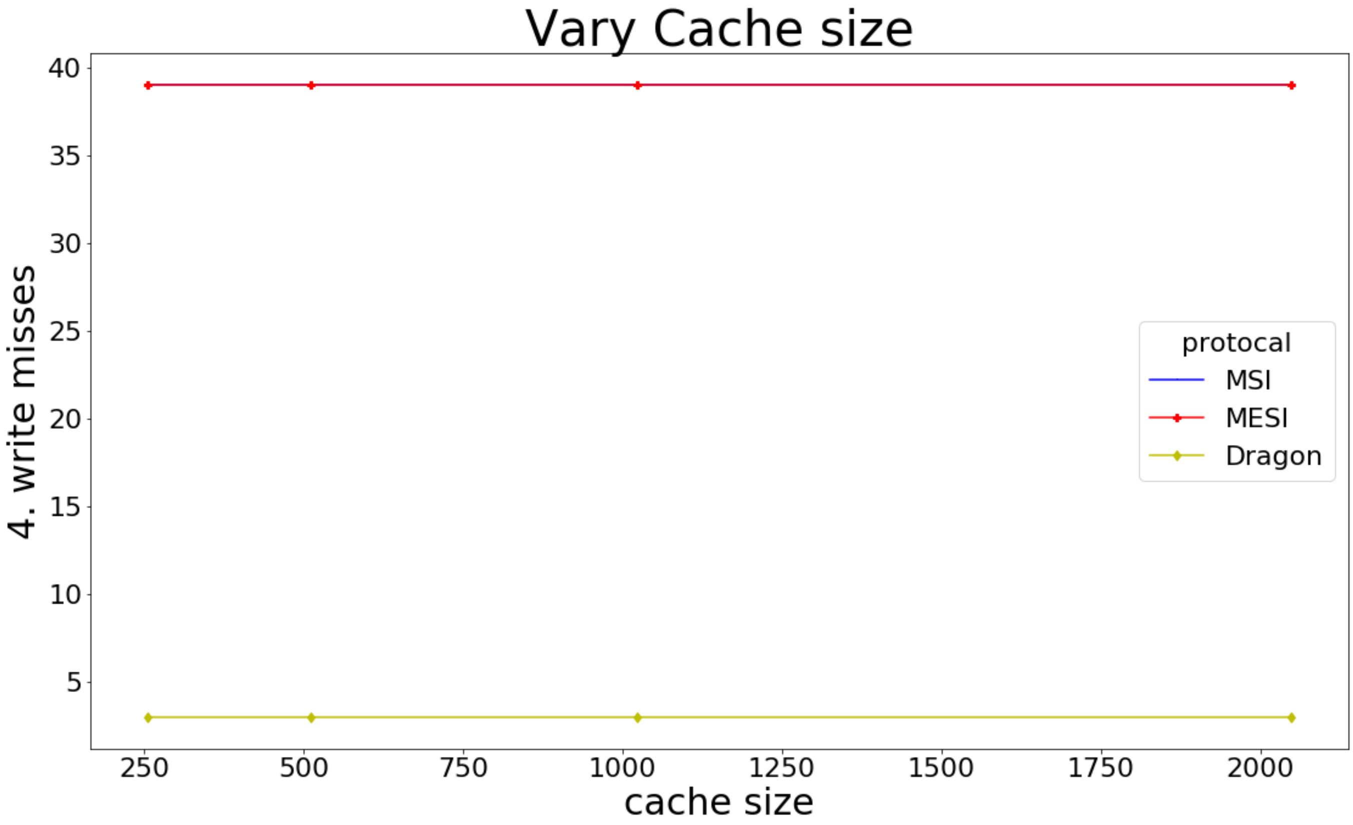
Graph 1.3



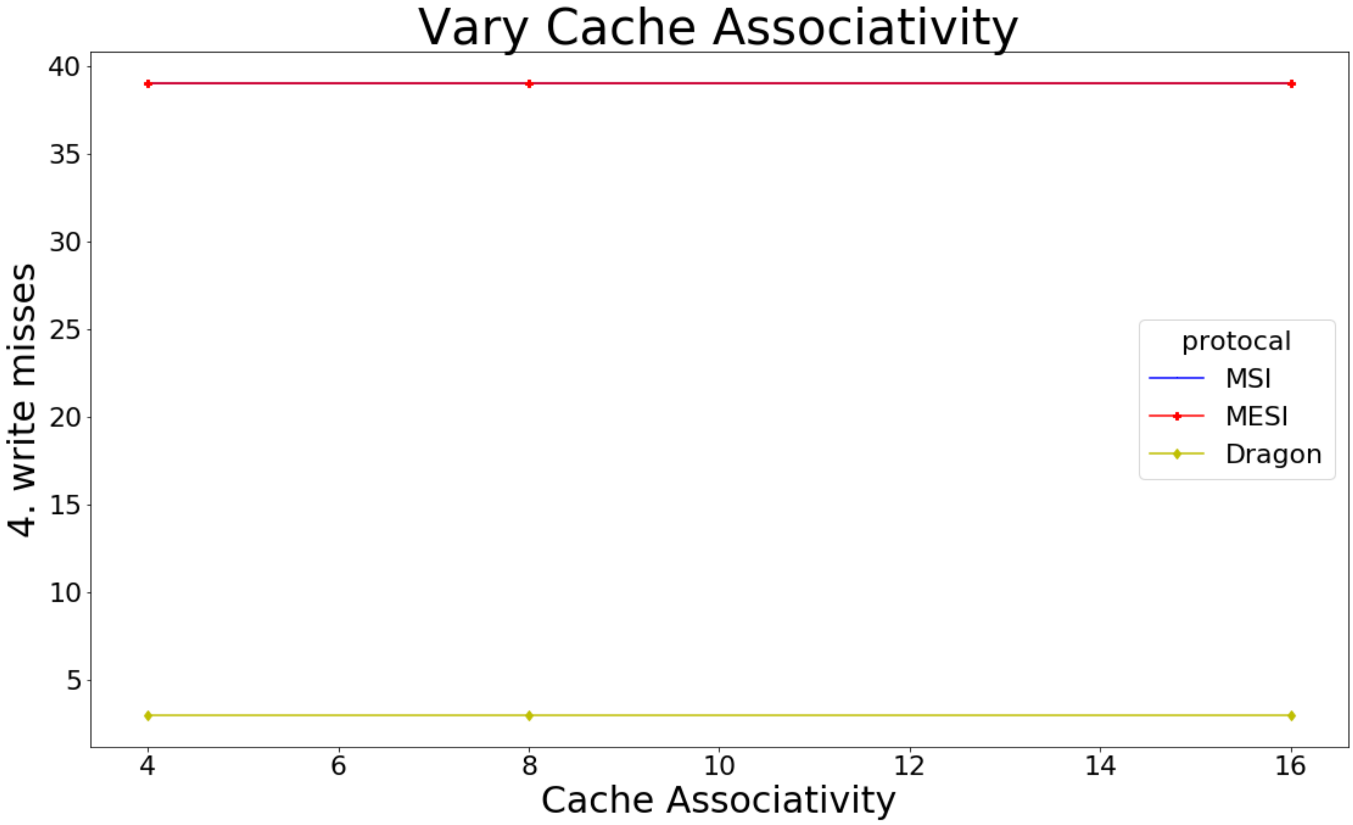
Dragon protocol is an updated-based protocol, and it maintains more valid cache blocks in cache and has a smaller number of write misses than MSI and MESI protocols.

The number of write miss does not change when we only increase cahce size, cache block size or associativity.

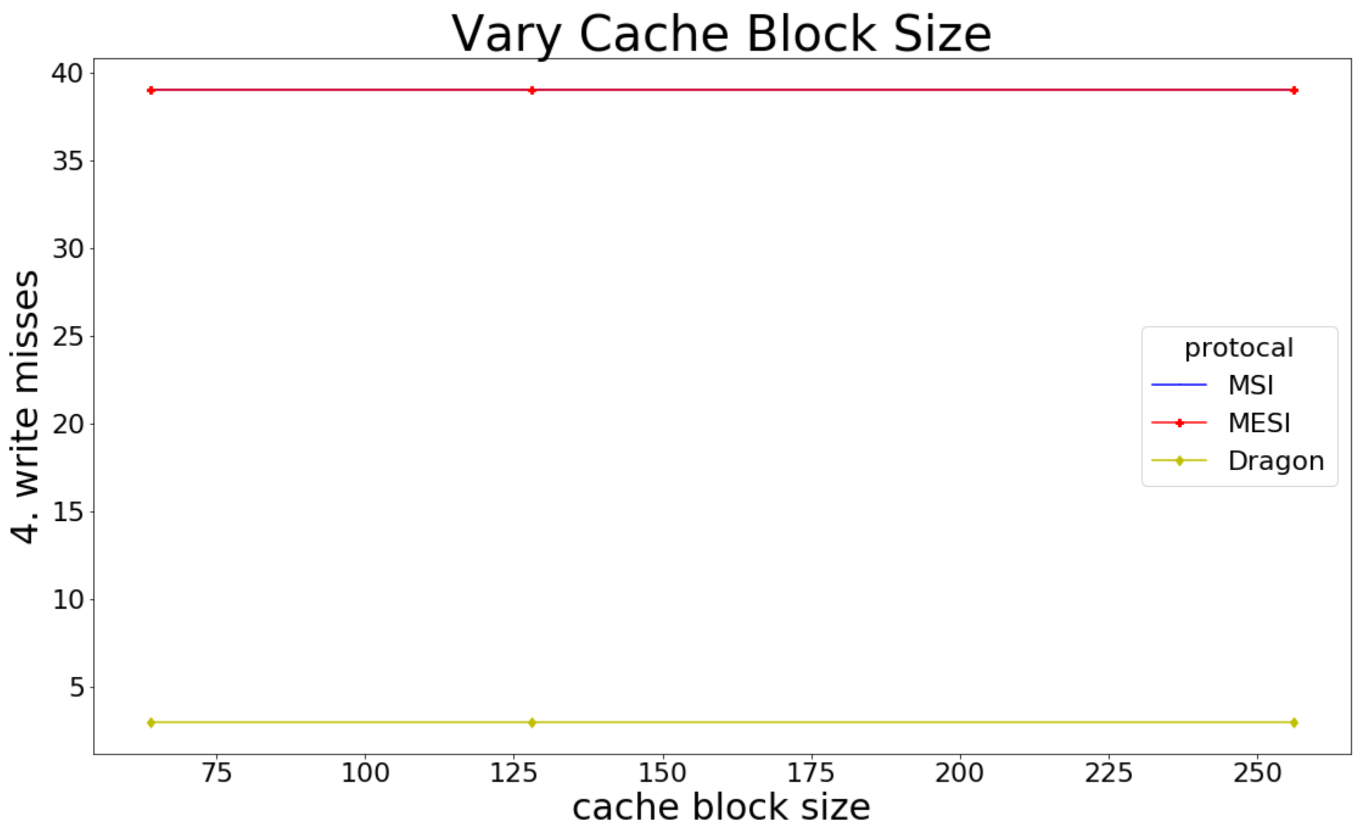
Graph 1.4



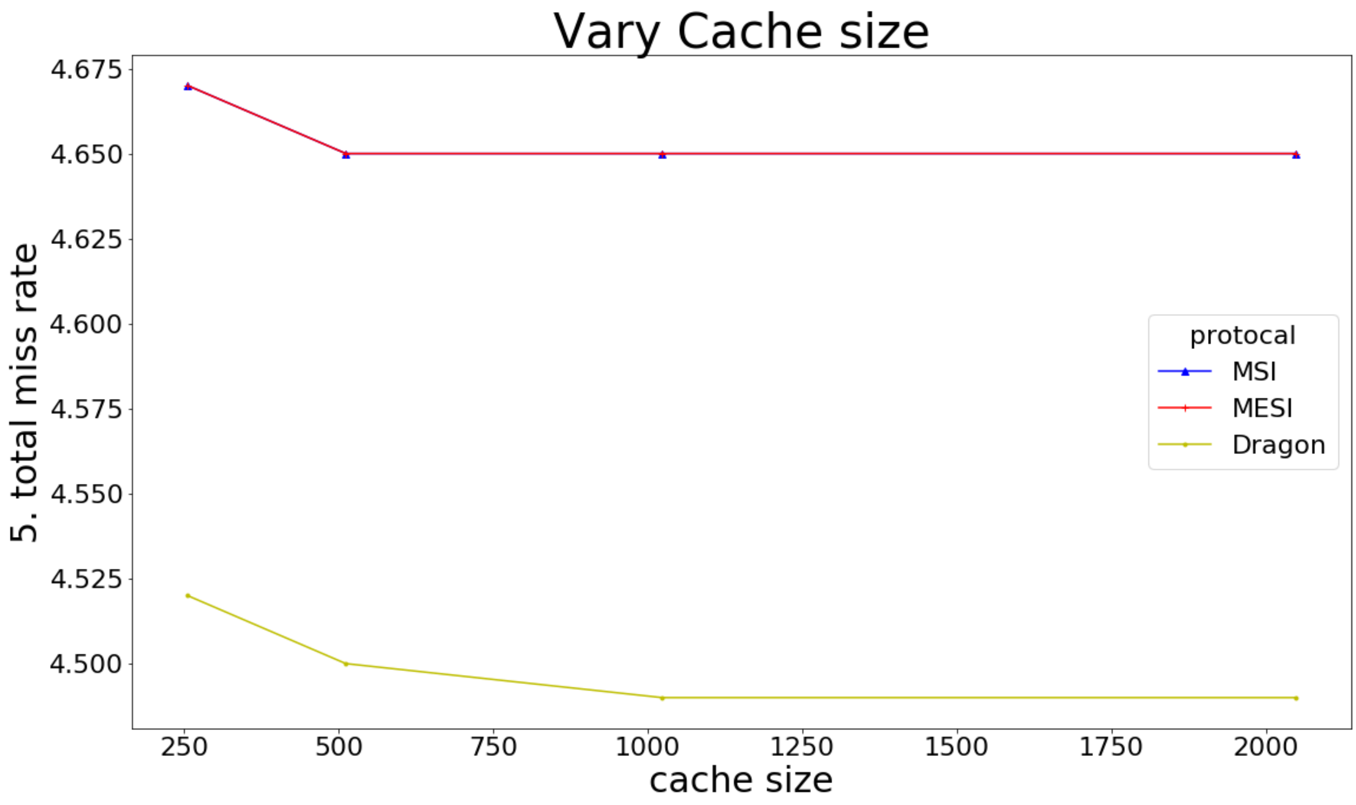
Graph 1.5



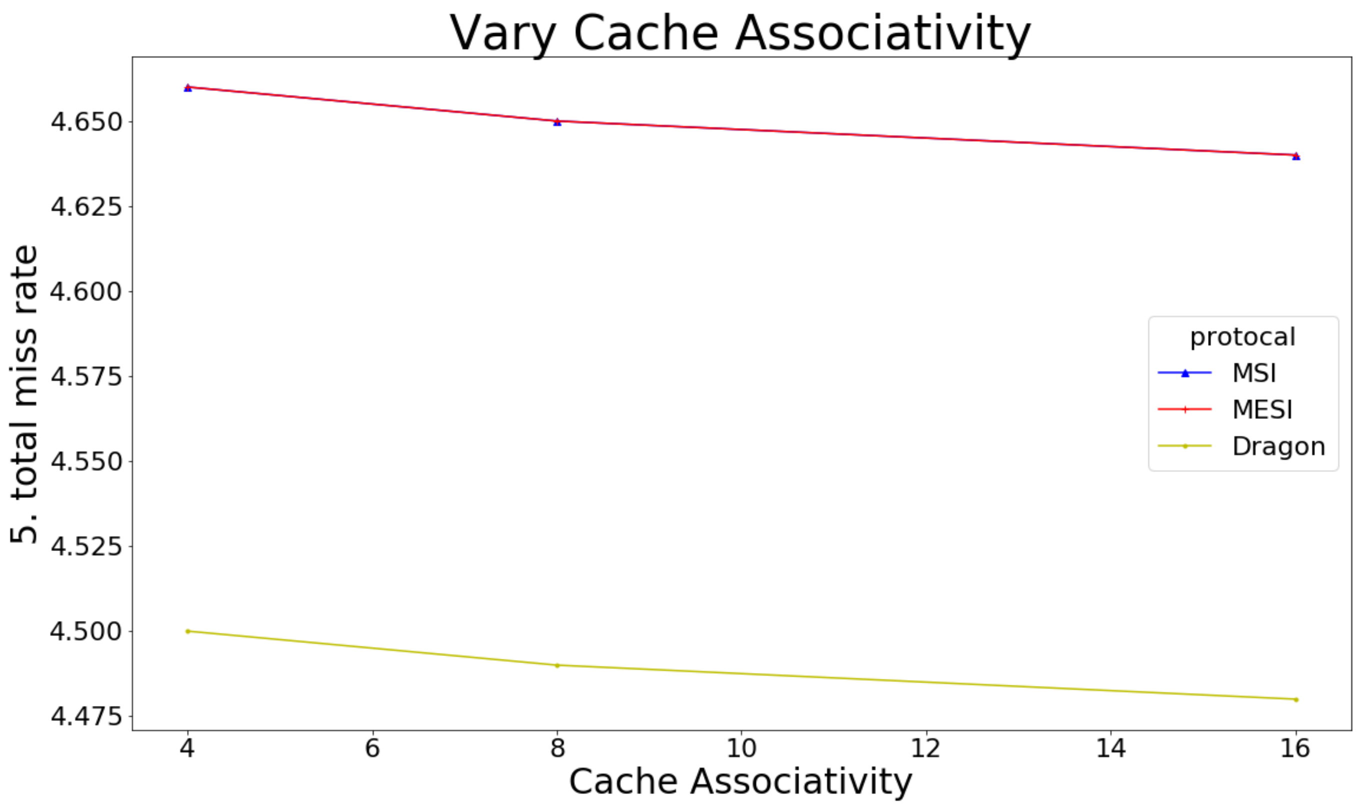
Graph 1.6



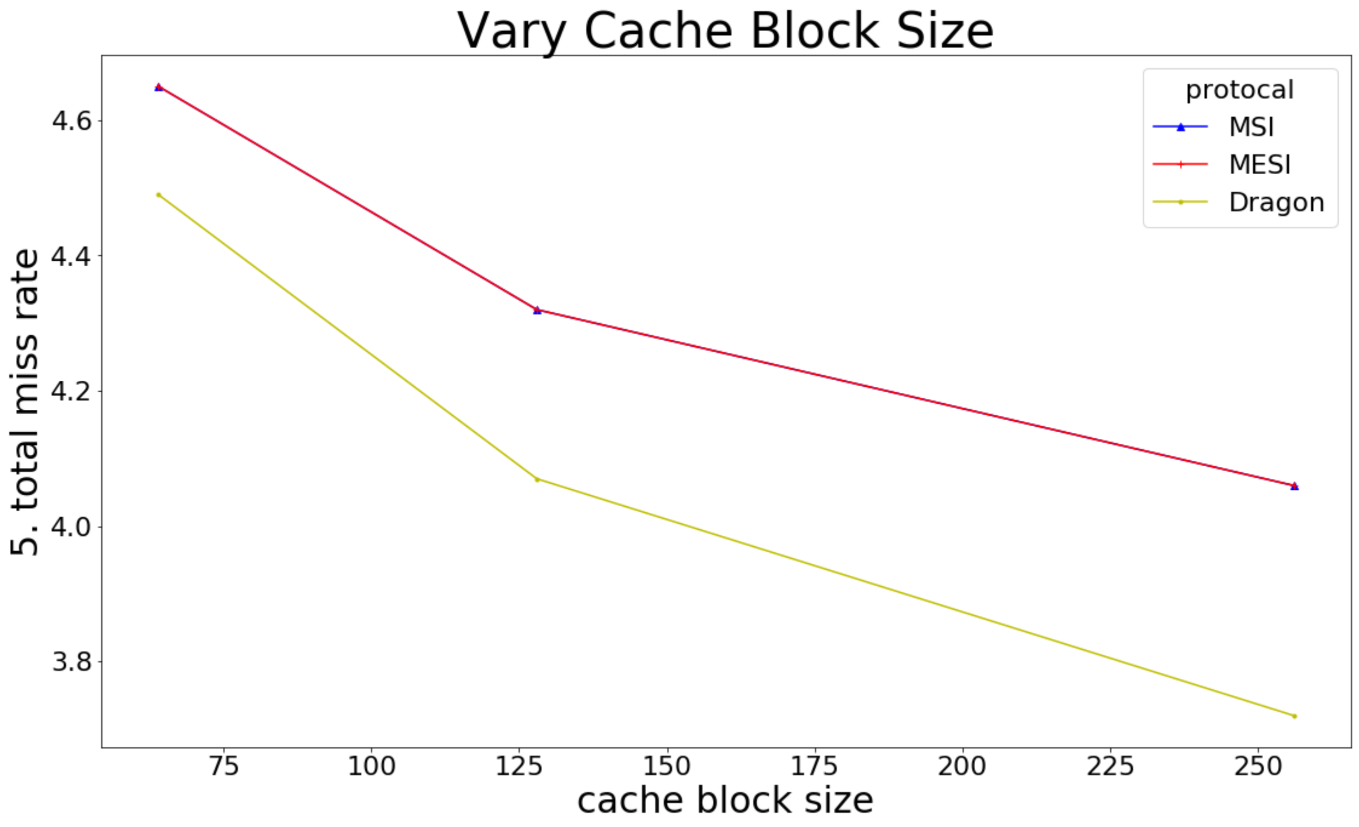
Graph 1.7



Graph 1.8



Graph 1.9

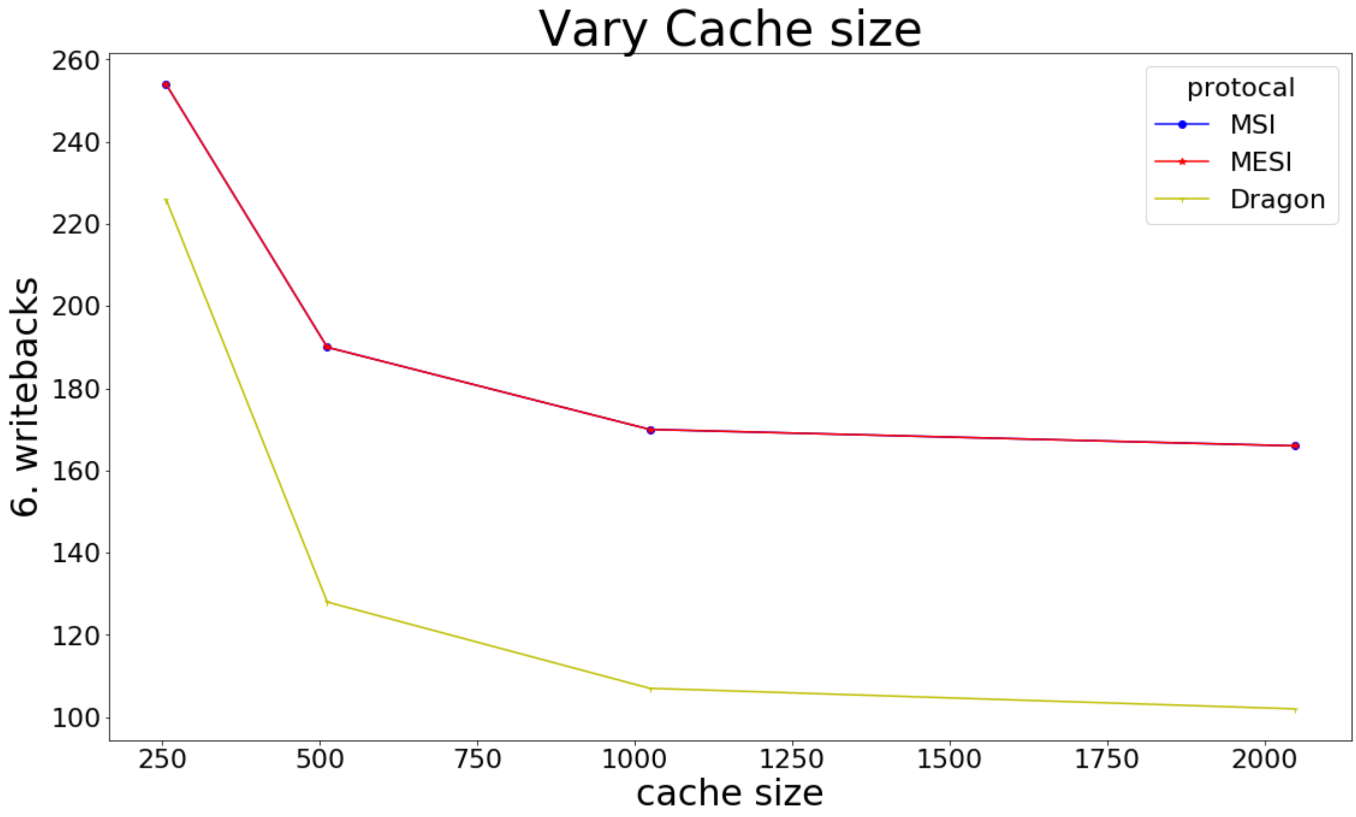


2. written back

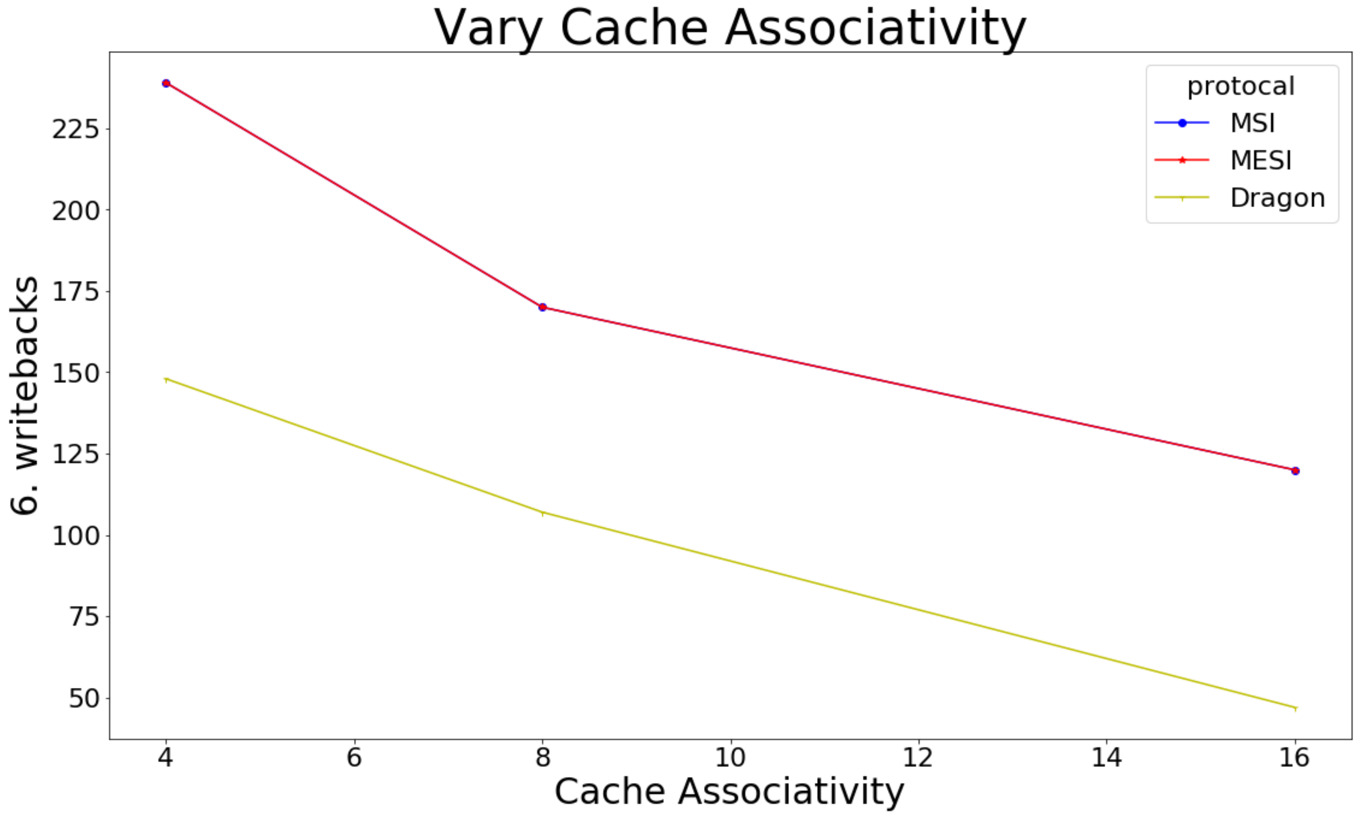
Due to the updated-based protocol's characteristic, Dragon protocol maintains more valid cache blocks in cache and has a smaller number of write back than MSI and MESI protocol.

When only cache size or associativity increase, writebacks decrease. However, when cache block size increases, write back increases. Because the block number in each set decrease, the cache set needs to evict the victim cache more frequently. Therefore, write back increase.

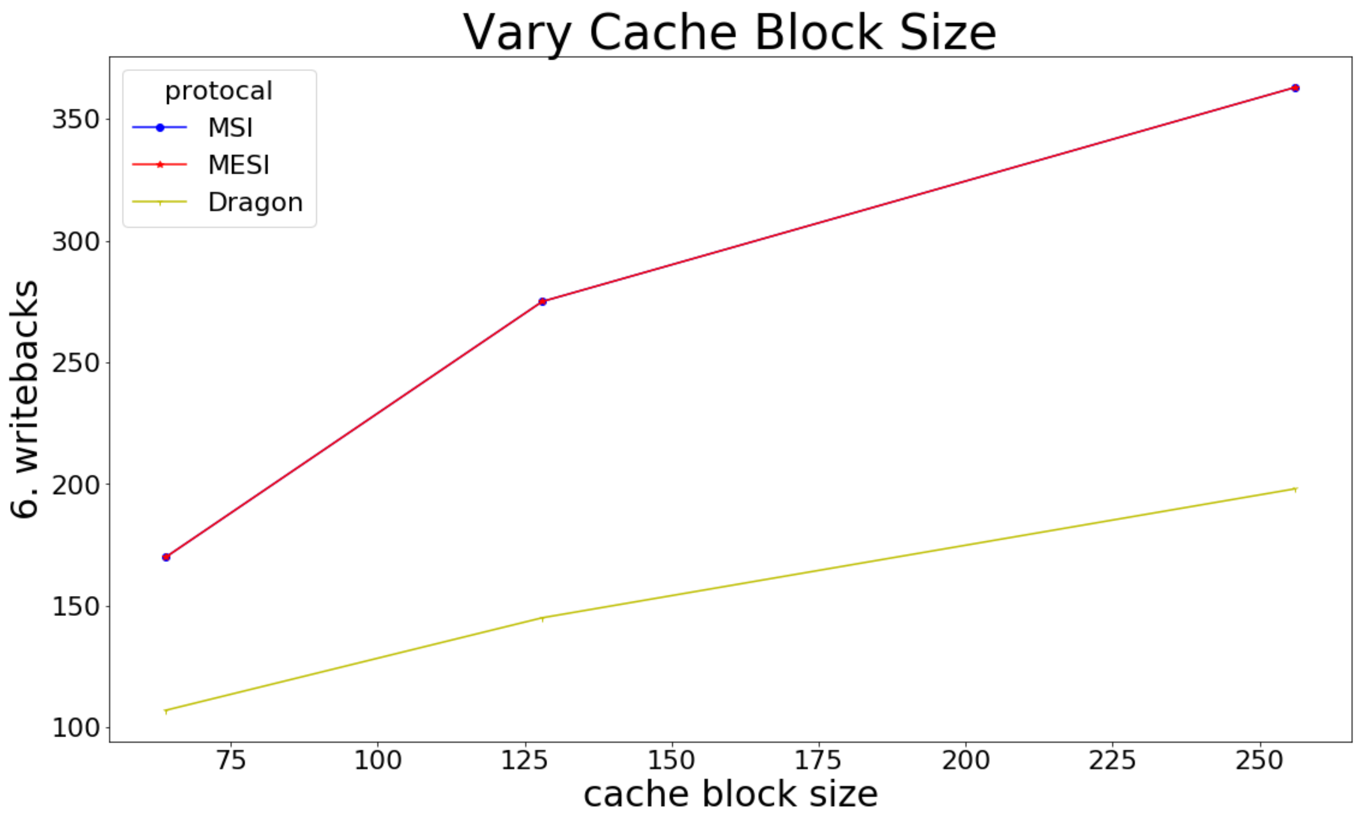
Graph 2.1



Graph 2.2



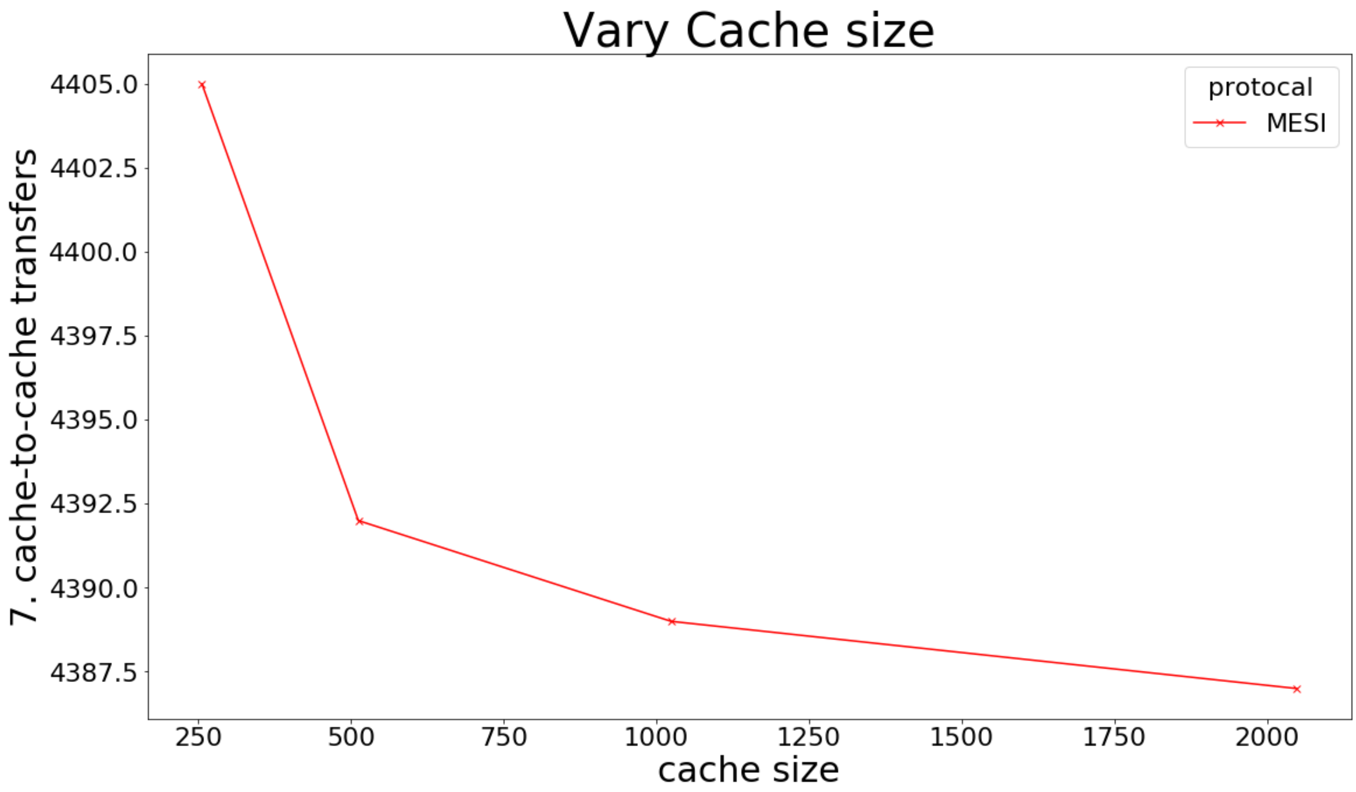
Graph 2.3



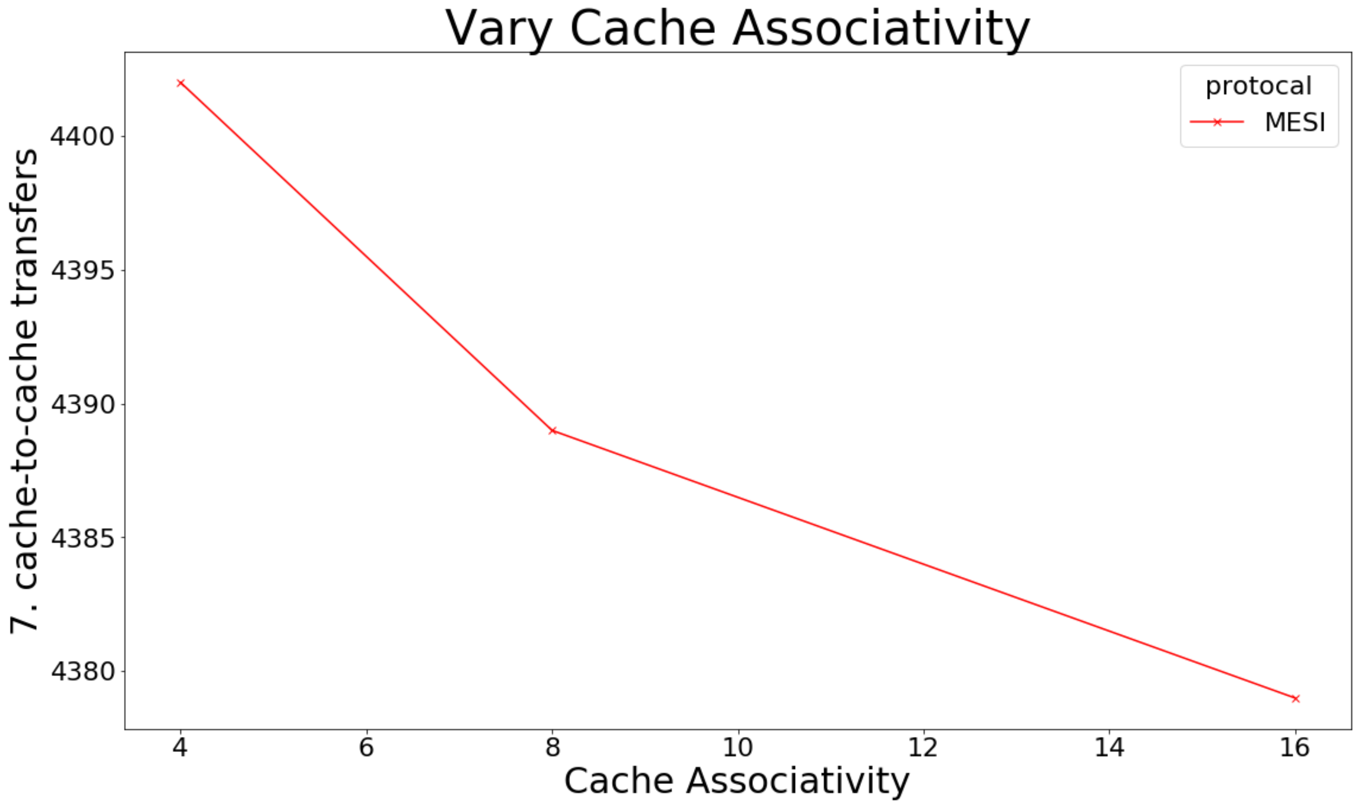
3. cache-to-cache transfers

There is no cache to cache transfer in MSI and Dragon protocol. Referring to the MESI protocol, when only cache block size or cache size, or associativity increases, the number of read/write miss decreases. Therefore, the number of cache to cache transfer decrease.

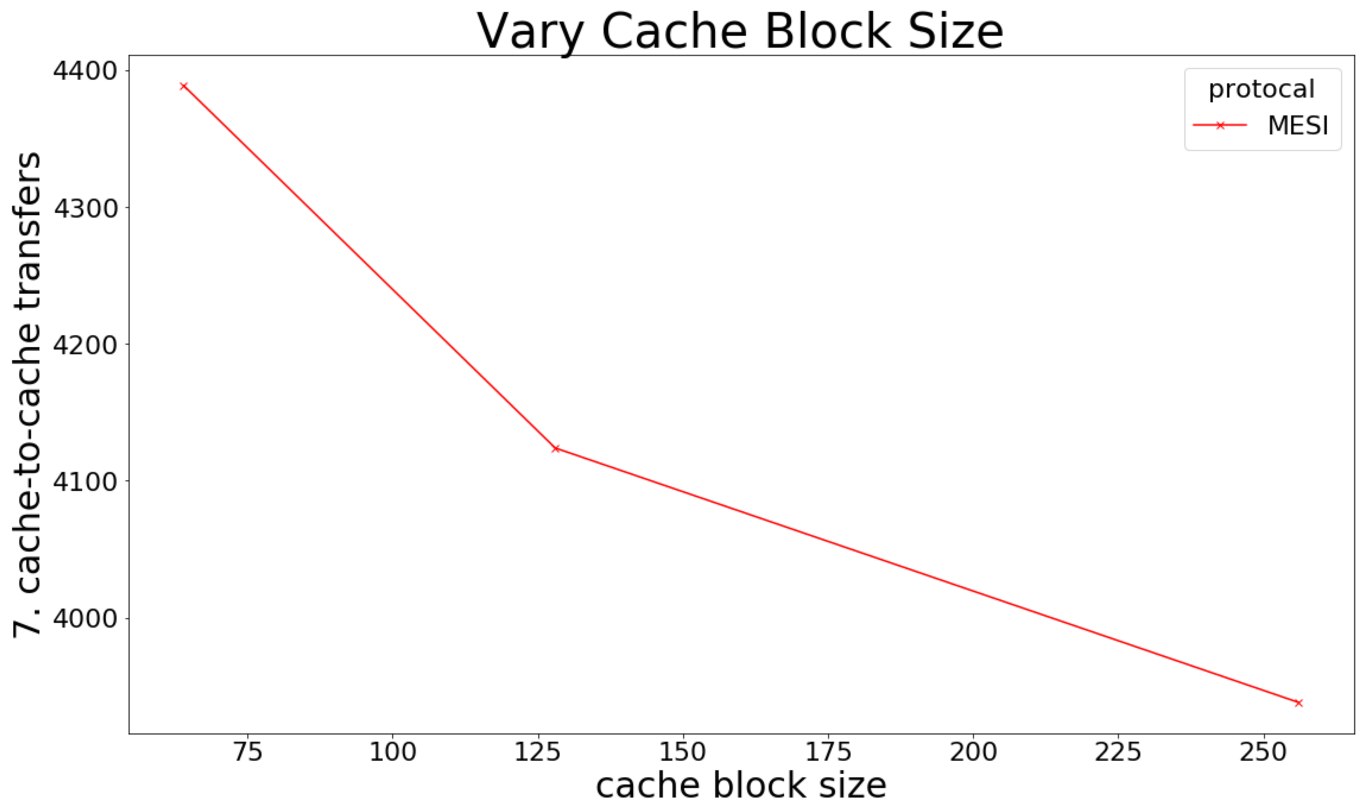
Graph 3.1



Graph 3.2



Graph 3.3

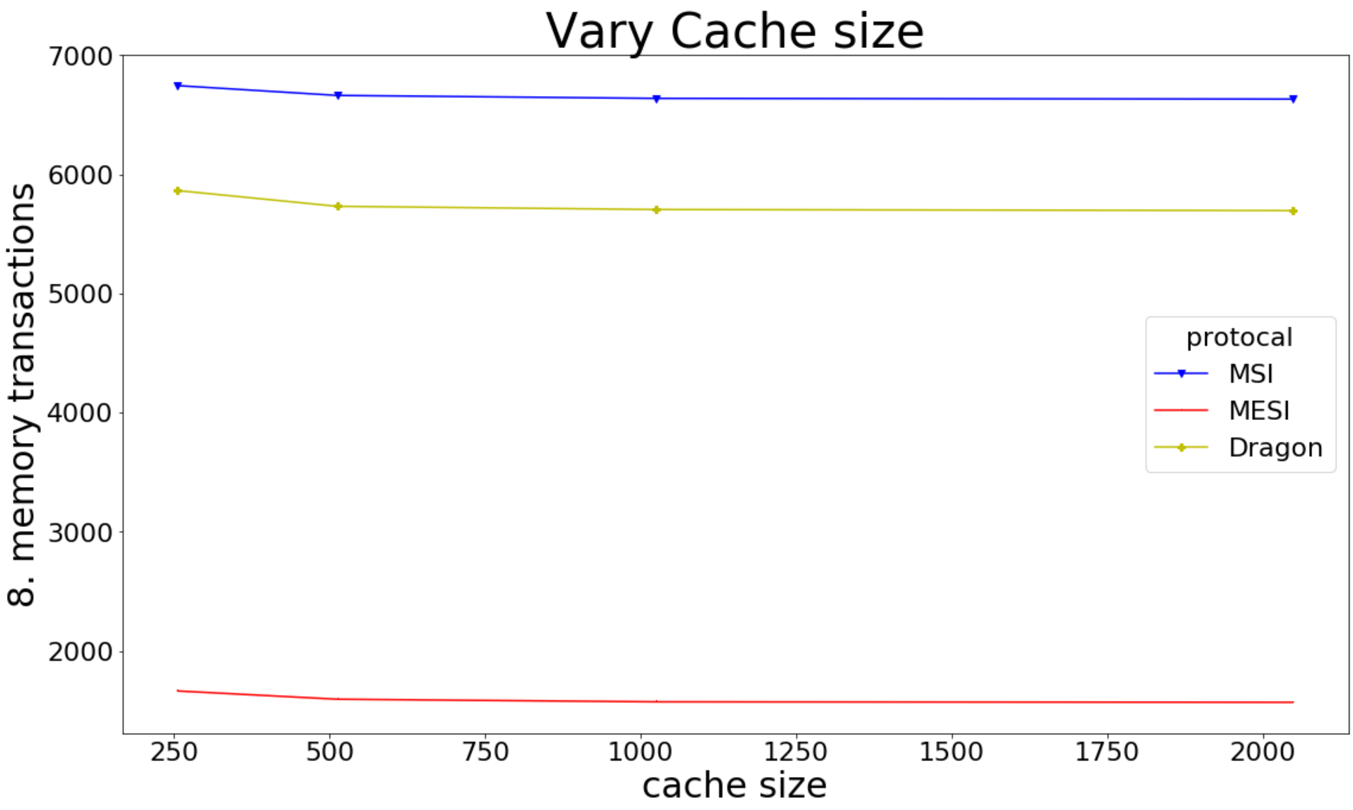


4. memory transactions

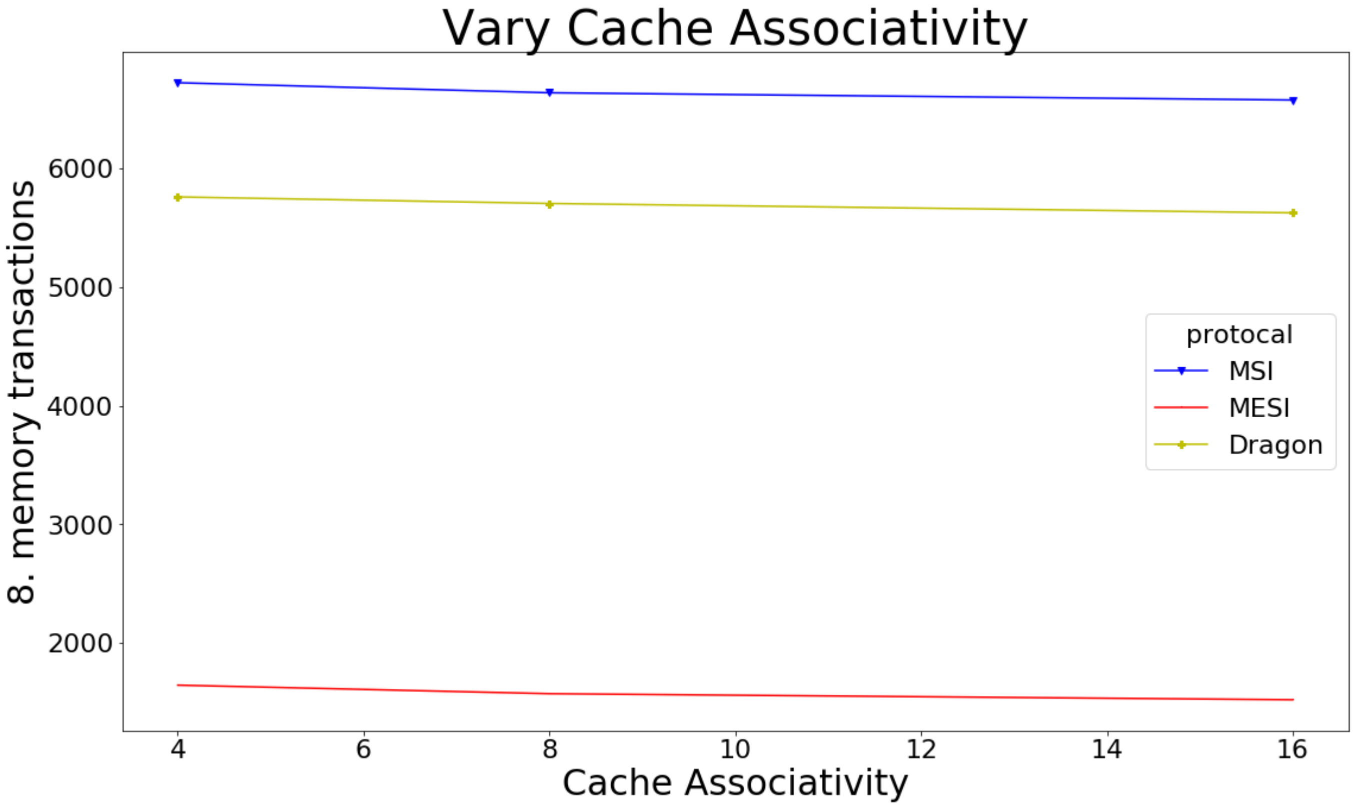
Dragan protocol has fewer memory transactions than MSI protocol. Since Dragon protocol has fewer read/write misses, Dragon protocol does not invalidate cache blocks when snooped.

MESI protocol has fewer memory transactions than Dragon protocol because MESI supports cache to cache transfers and decreases unnecessary memory access.

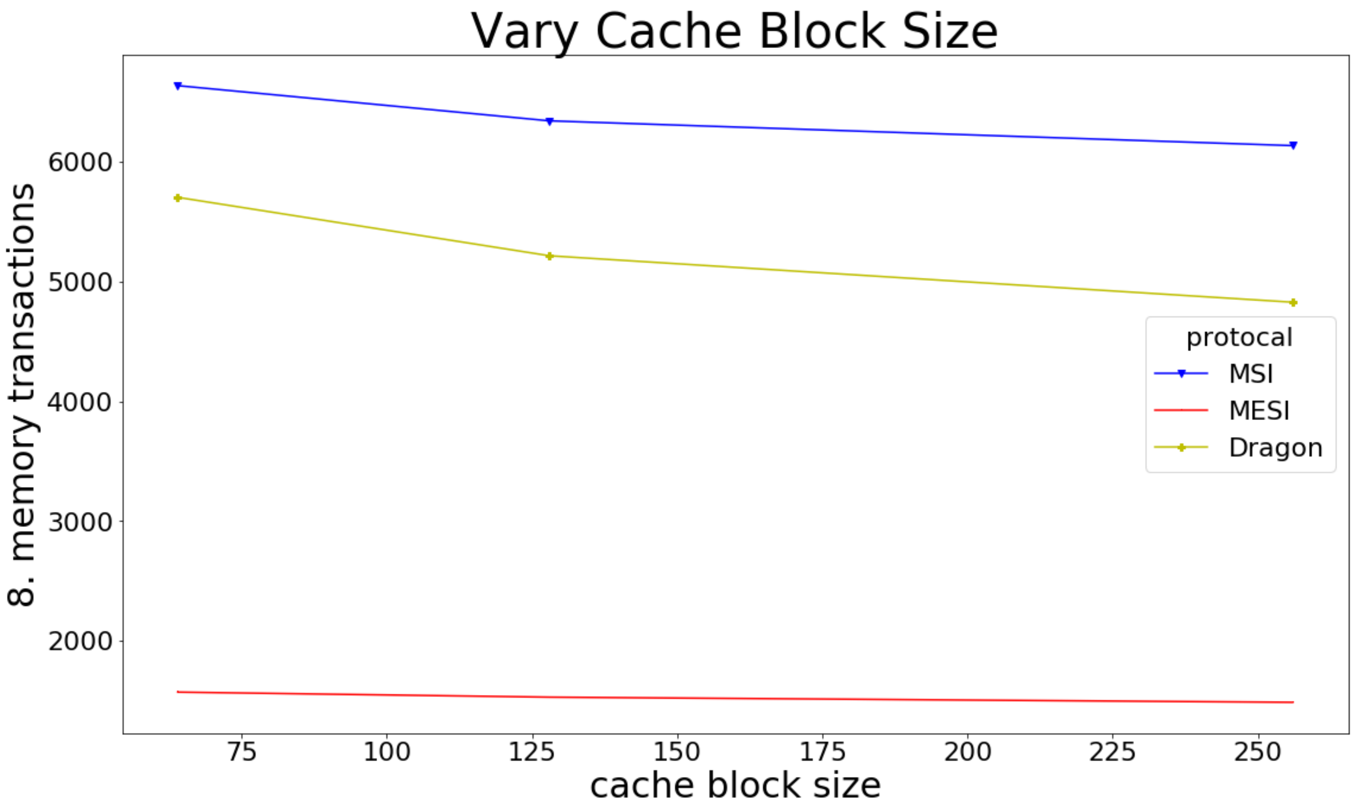
Graph 4.1



Graph 4.2



Graph 4.3



5. interventions

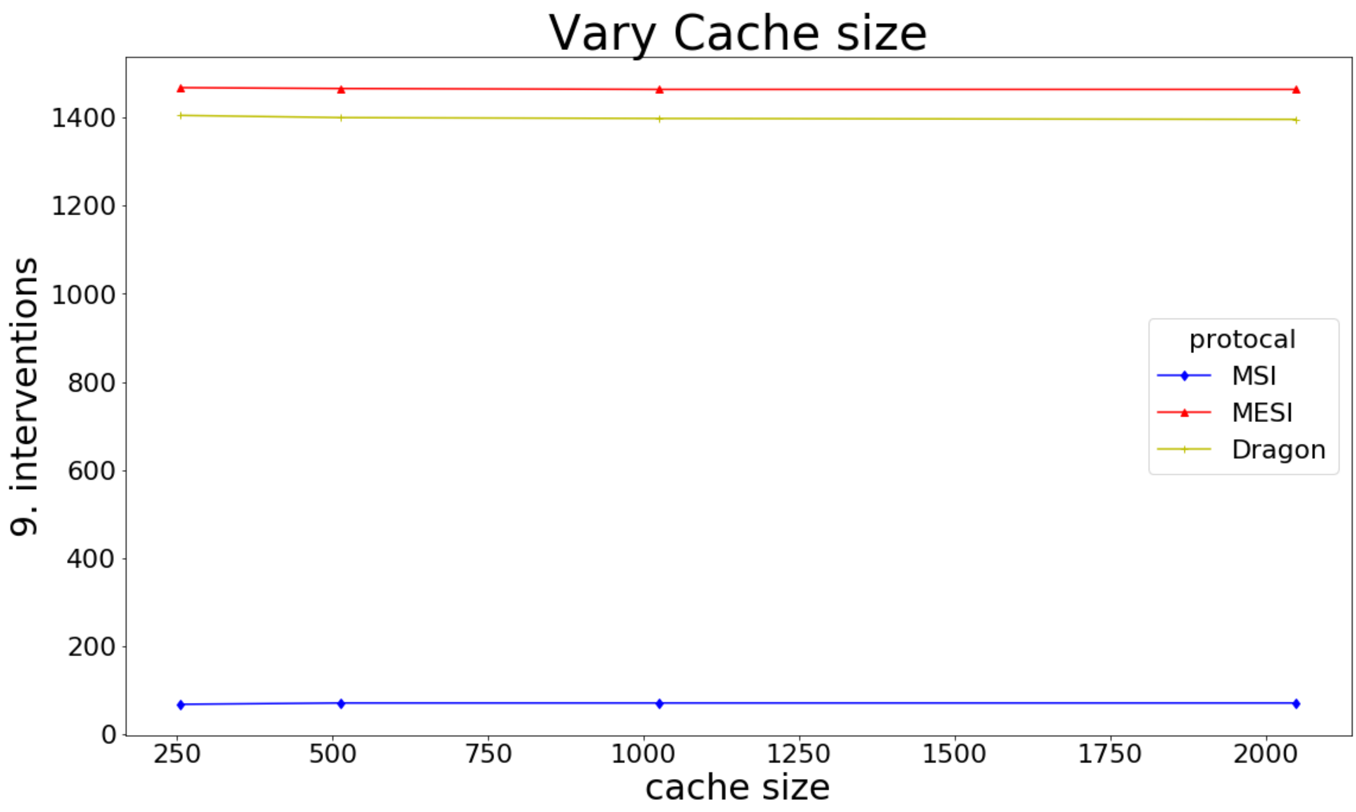
Dragon protocol has fewer interventions than MESI protocol because many state changes occur between shared clean and share modified in Dragon protocol.

MSI has the fewest interventions since interventions only occur when the state change from modified to shared. There is no exclusive state in MSI.

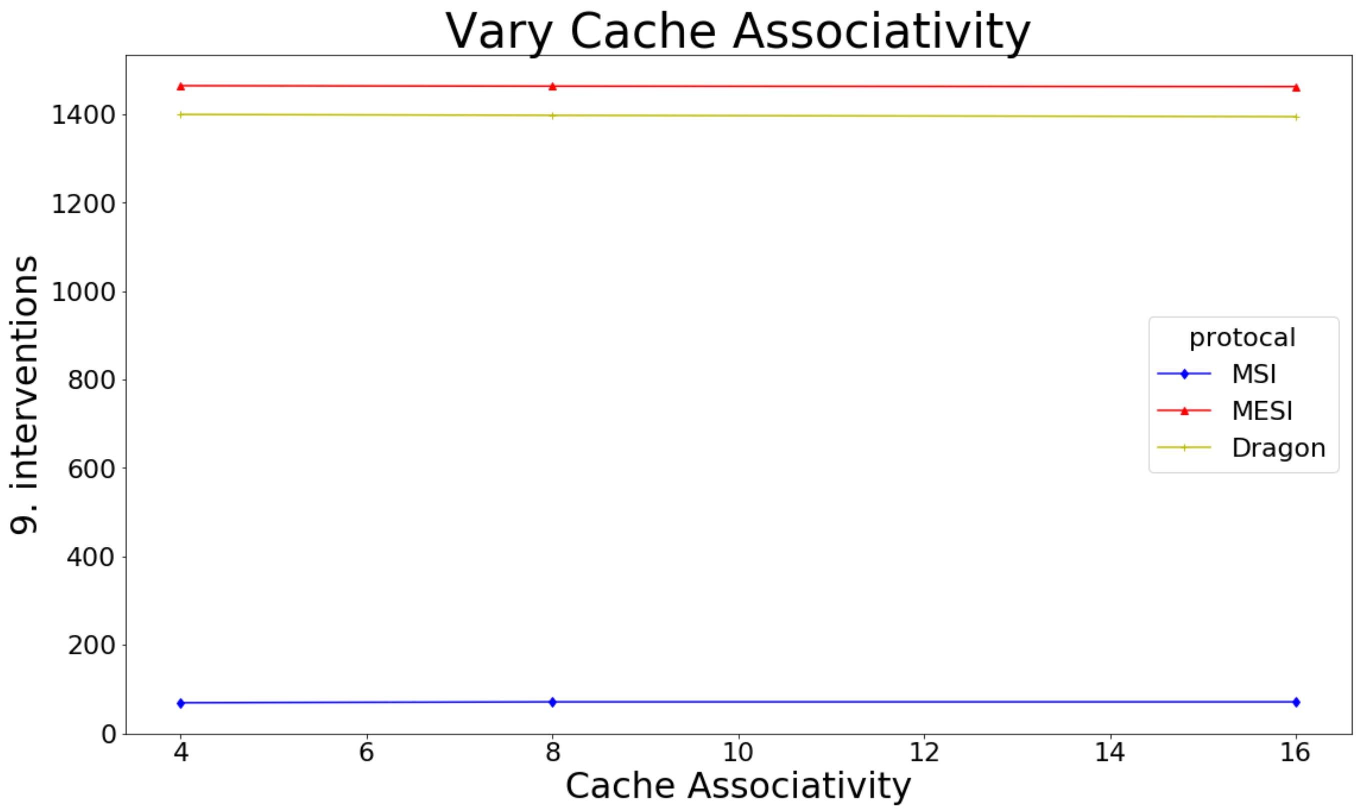
When only cache size or associativity increase, the chance of evicting cache blocks decrease. This factor makes blocks maintain a shared state in cache for a longer time and become harder to be evicted and re-allocated as an exclusive state. Therefore, the change of interventions decreases.

When only cache block size increase, the number of cache blocks in one set decrease. Therefore, more cache access in the same cache blocks. After a cache block has been written and flagged at a modified state, it has a higher chance to be accessed(read request) by processors. Therefore, interventions in MSI protocol increase.

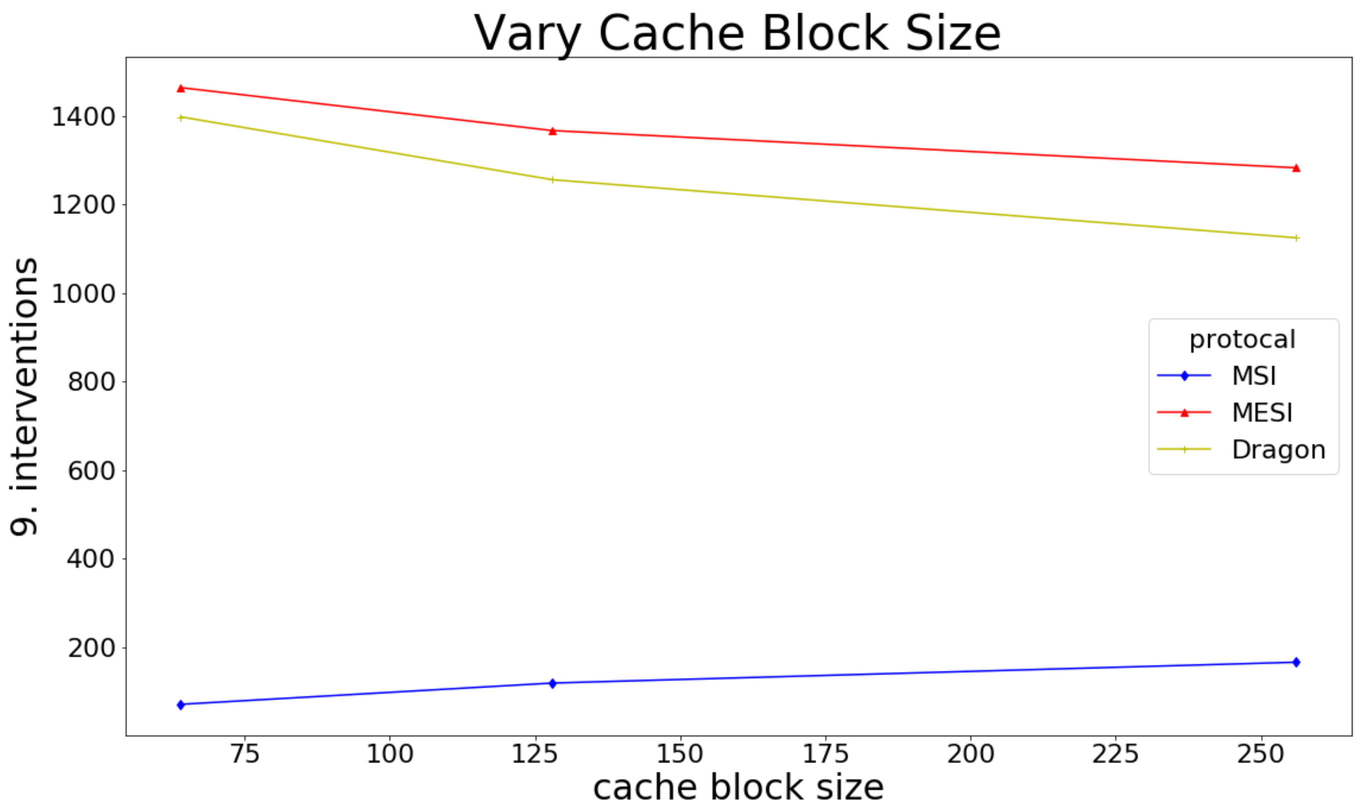
Graph 5.1



Graph 5.2



Graph 5.3



6. invalidations

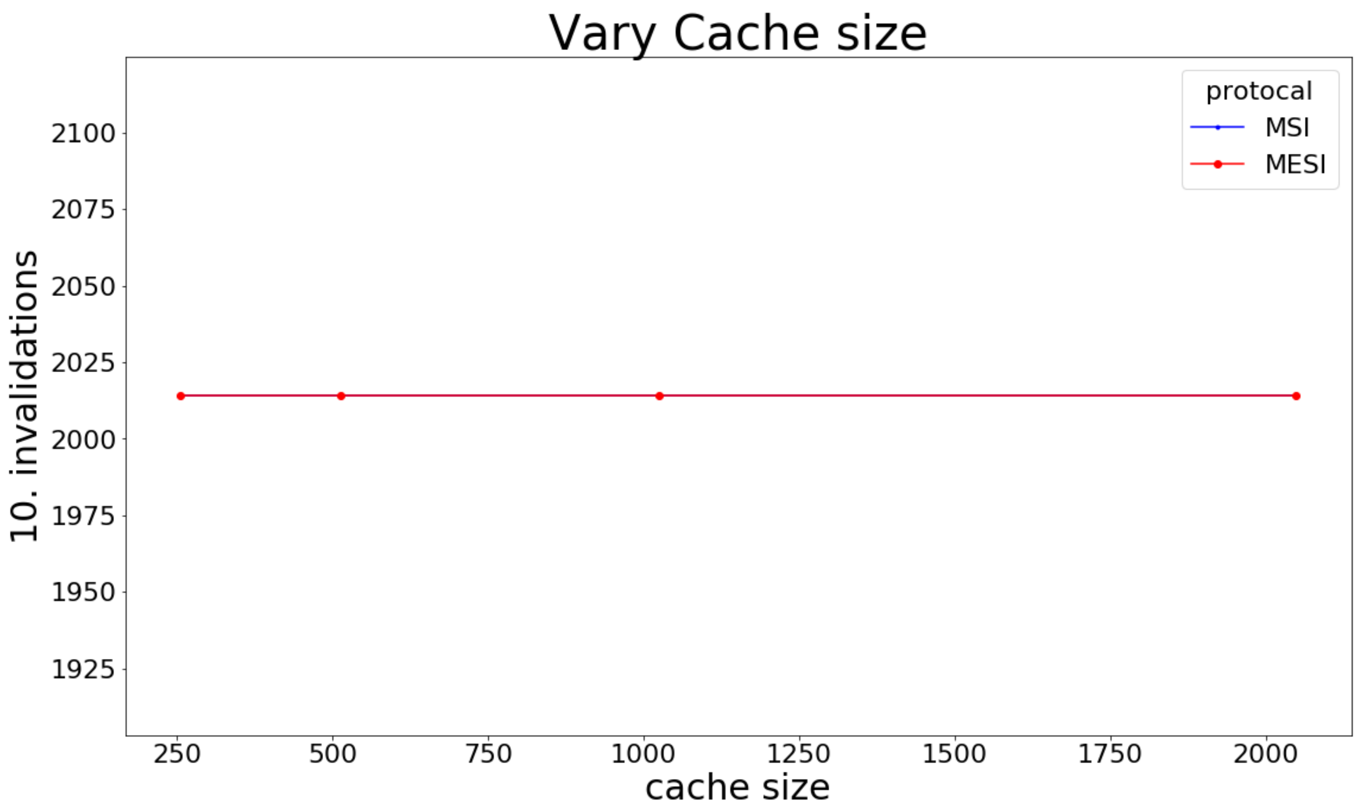
Dragon protocol is an update-based protocol and it does not support invalidations.

Only increasing cache size or associativity does not affect invalidations. However, when we only increase the cache block size, more data will be brought in the same cache block and increase invalidations.

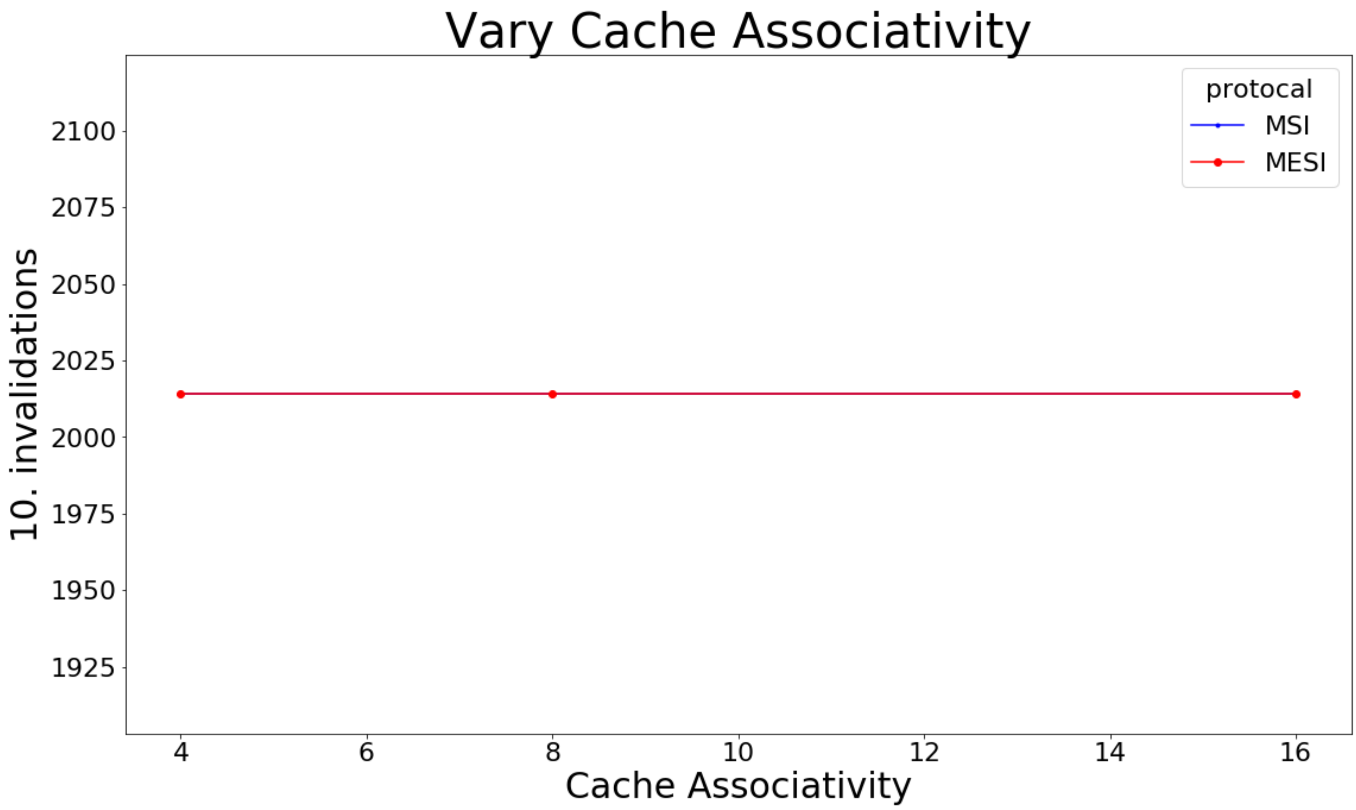
For example, given a scenario where data A and B are located in two different blocks, processor 1 reads data A first, and then processor 2 writes data B second. In this case, there are no invalidations since data B does not exist in any caches.

When we only increase block size four times and read data A, processor 1 might read data B into the same cache block simultaneously. Remember that data B will not be read by processor 1 before we increase the block size. Next, processor 2 write to data B. Because data B is in other caches, invalidation occurs.

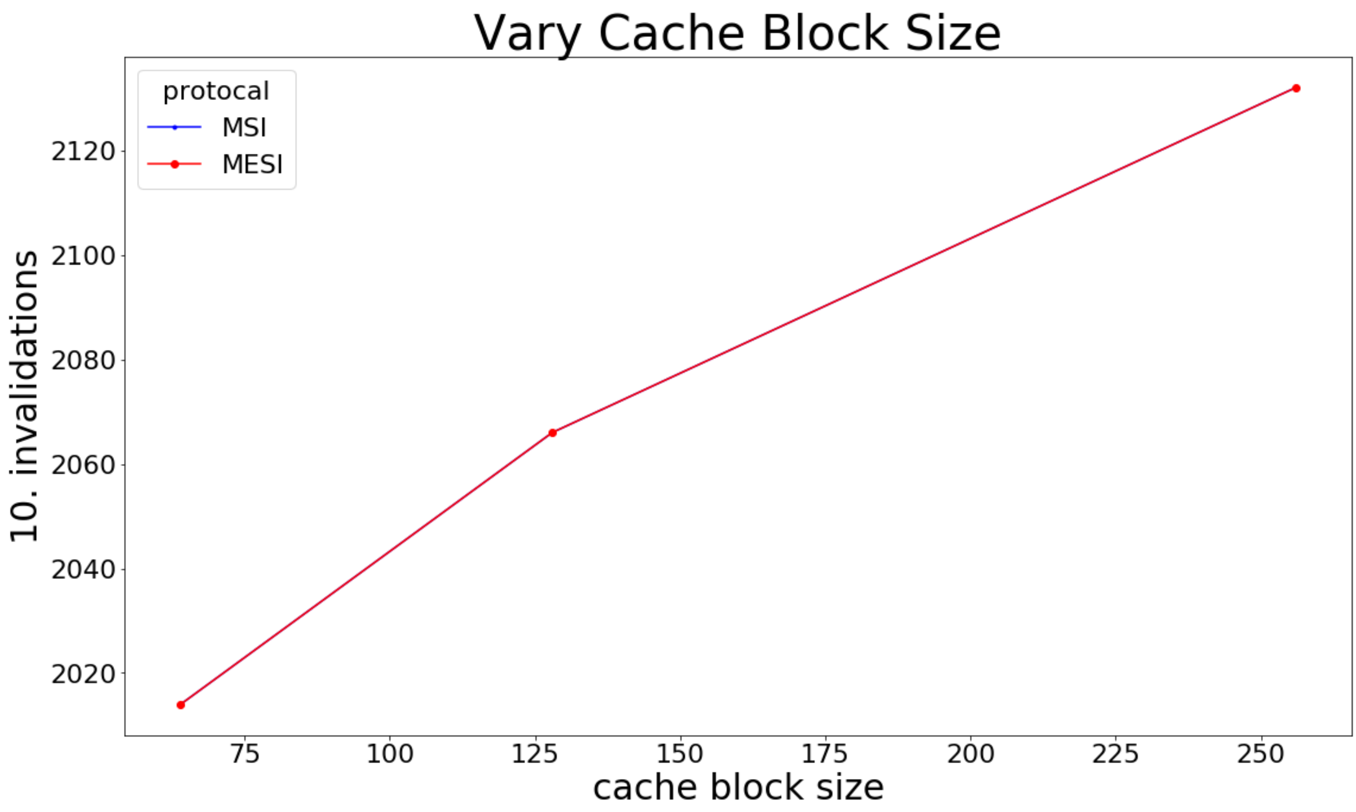
Graph 6.1



Graph 6.2



Graph 6.3

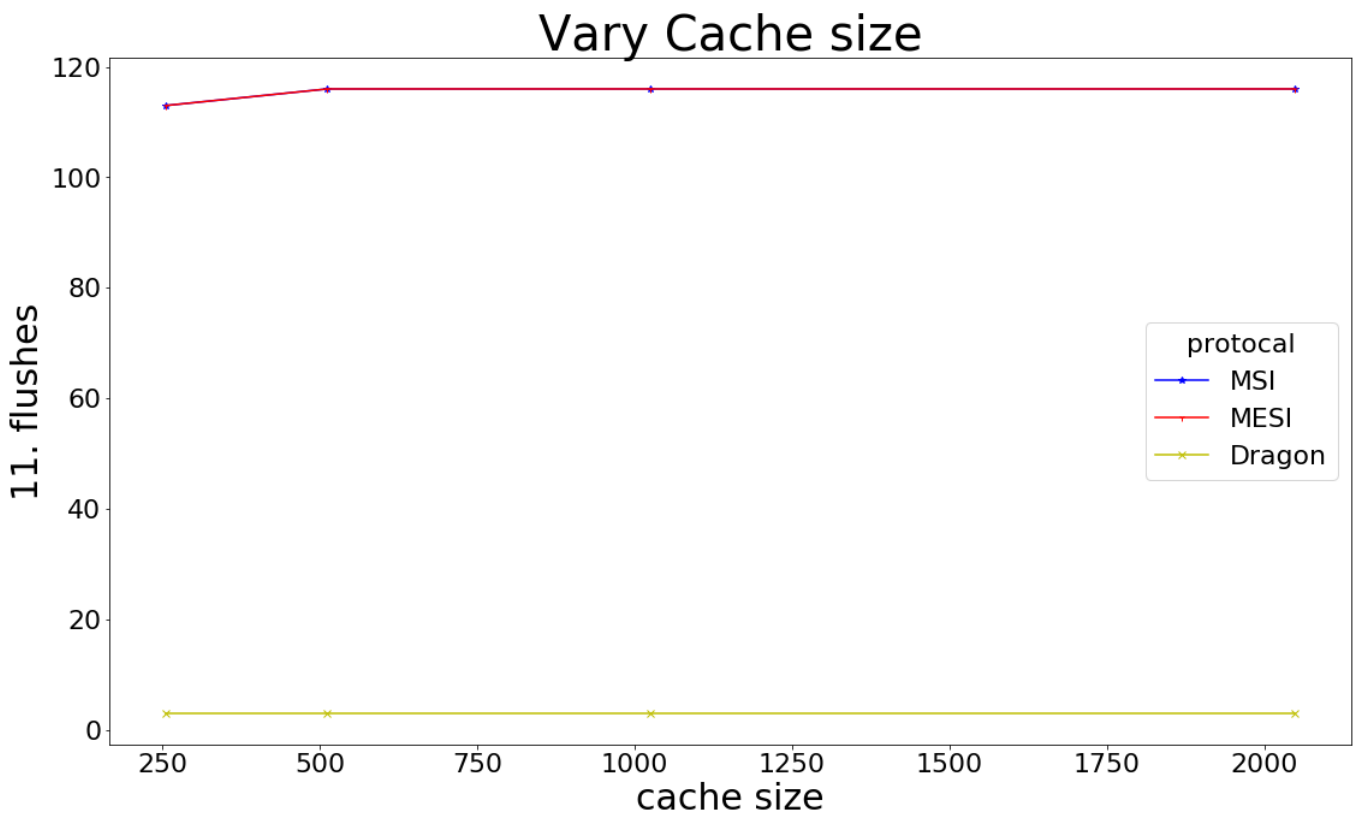


7. flushes

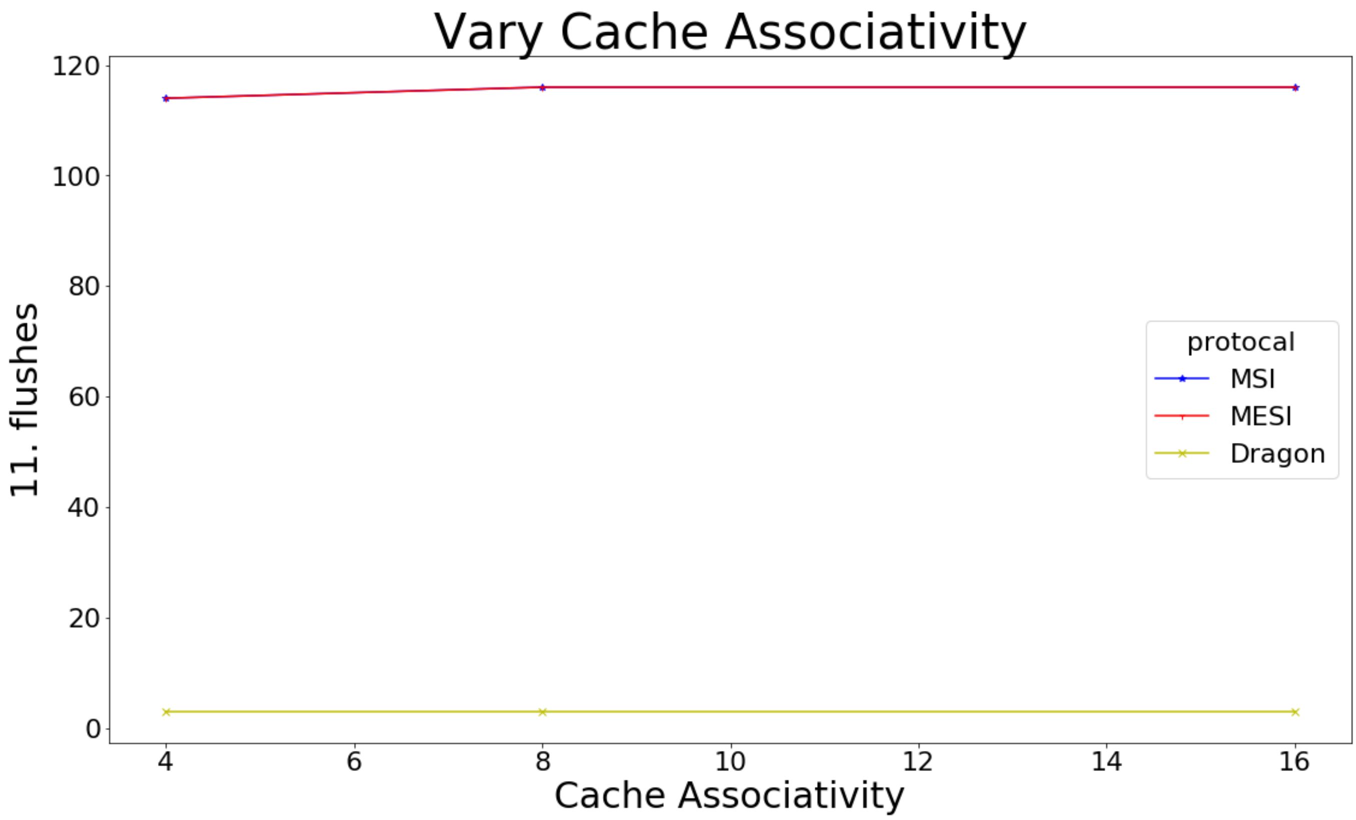
Dragon protocol has fewer flushes than NESI and NSI protocol. Since Dragon protocol supports dirty sharing, there are no flushes when the coherence state change between Sc(share-clean) and Sm(share-modified).

when only block size increase, more data are bring in same cache blocks, and processors access the same cache blocks more times. Therefore, flushes increase.

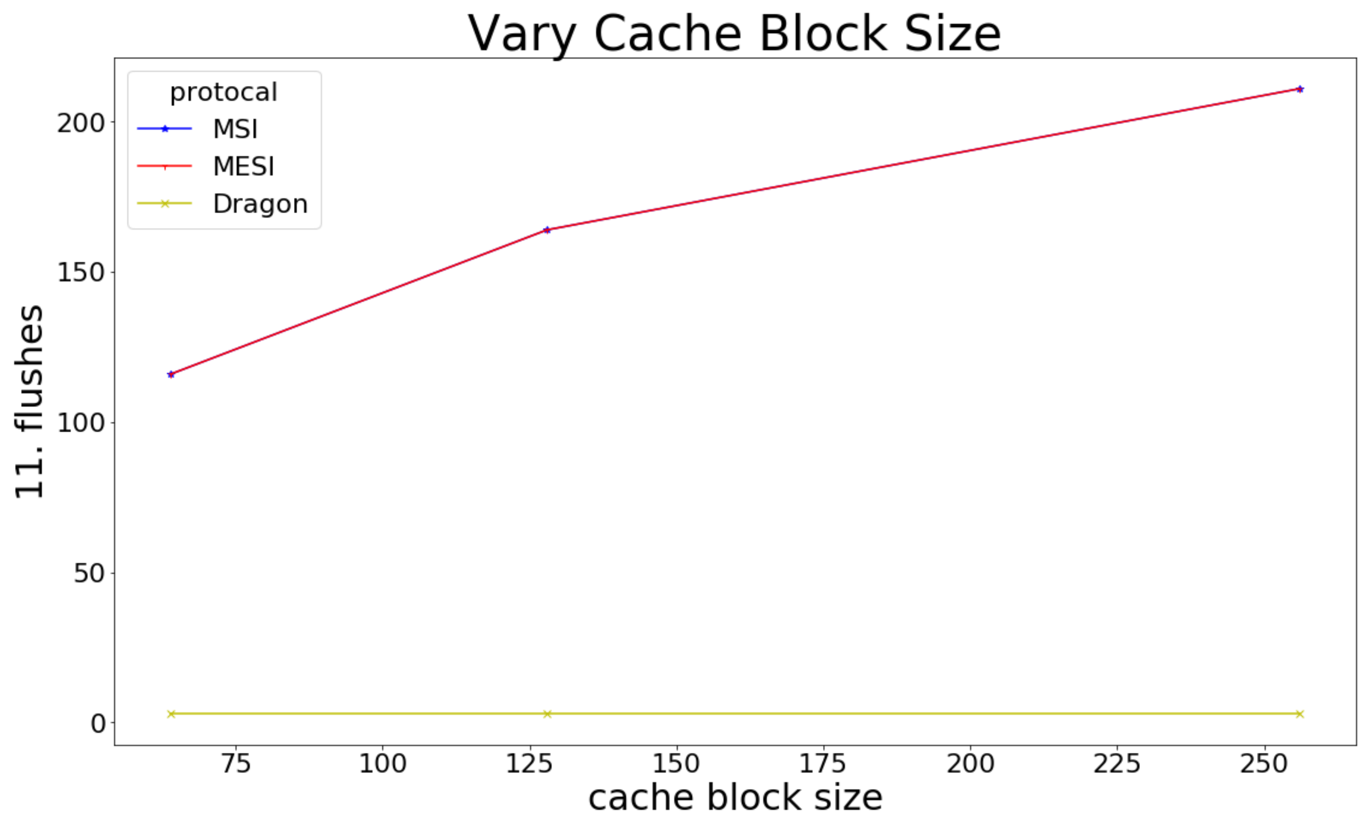
Graph 7.1



Graph 7.2



Graph 7.3

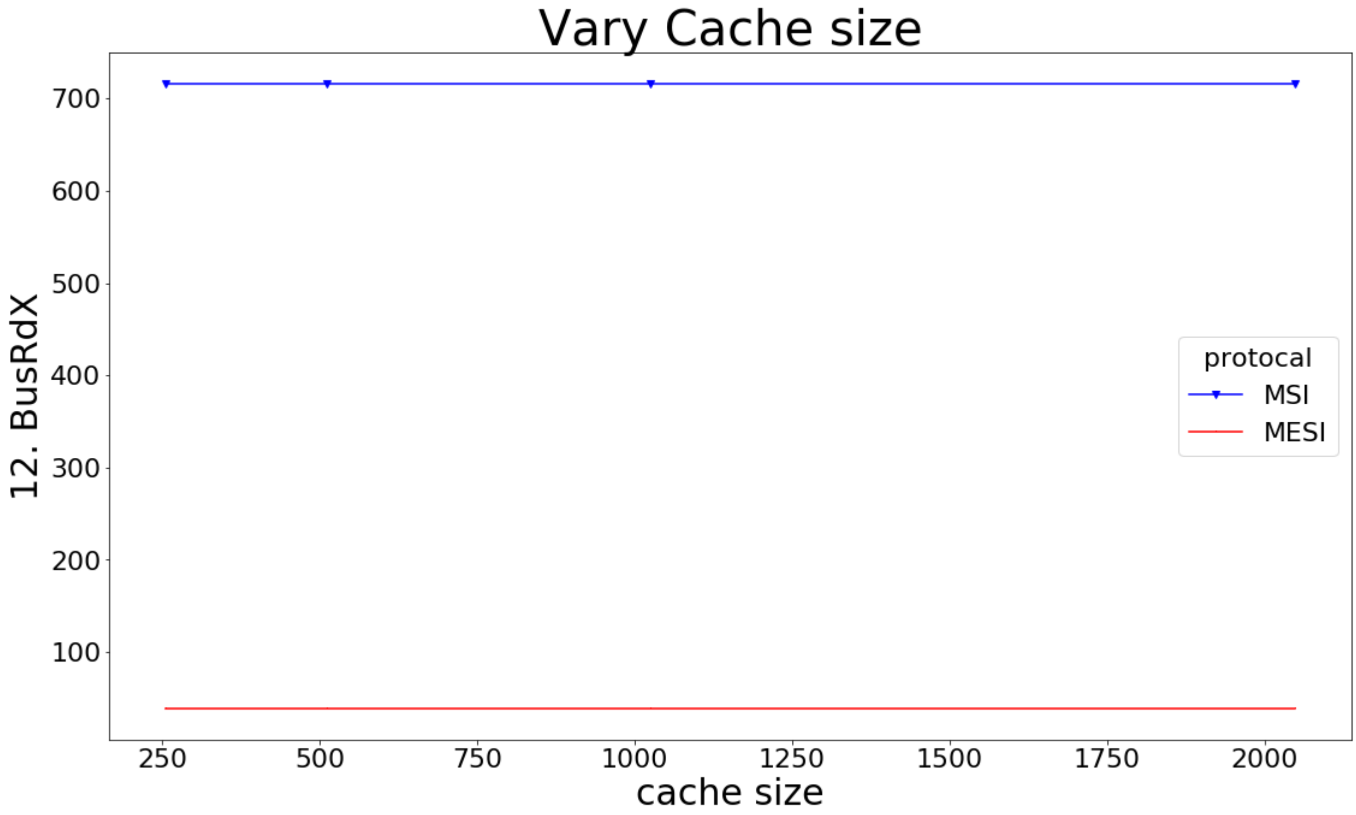


8. BusRdX transactions.

Dragon protocol is an update-based protocol, and it does not support BusRdX.

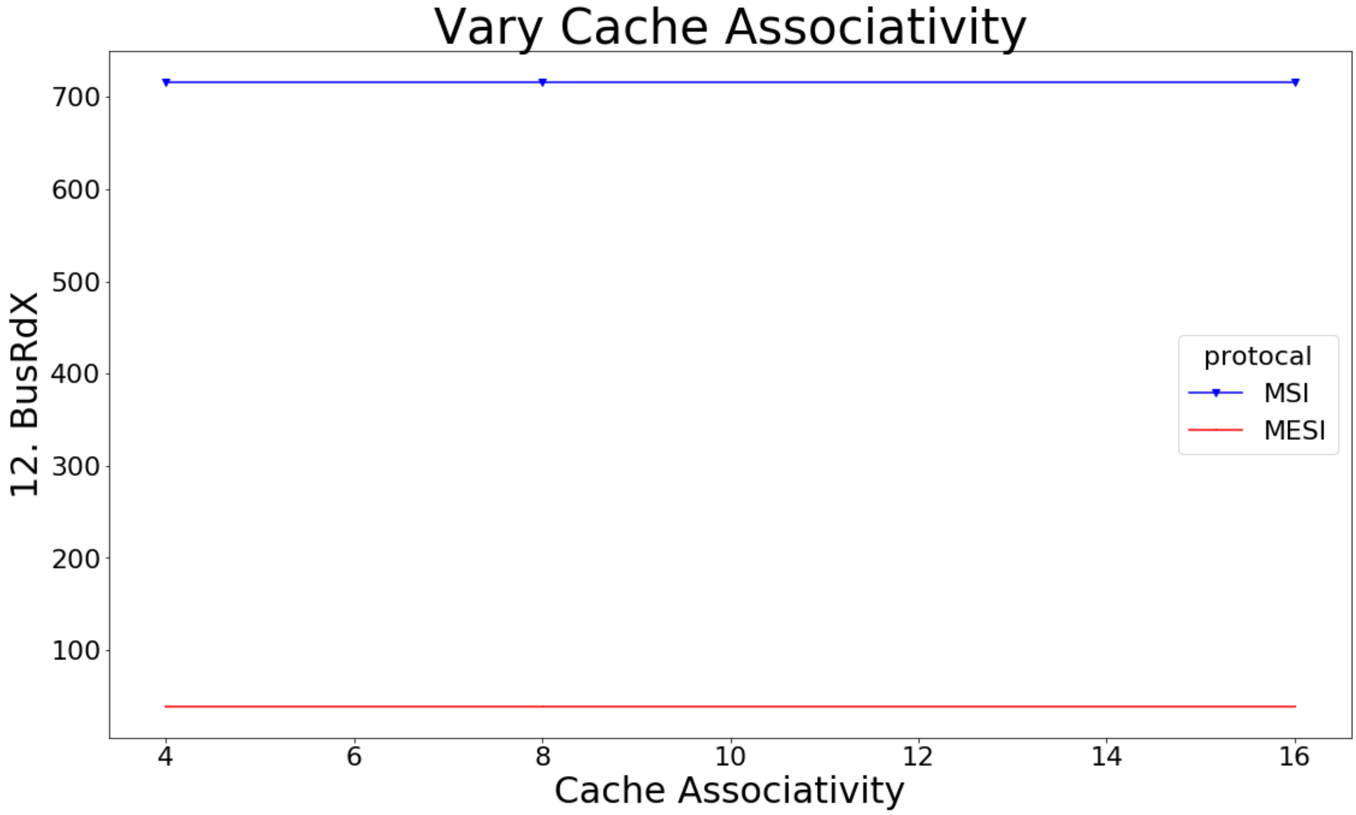
In MSI protocol, when the processor writes after reads. It needs to snoop BusRdX on Bus.
However, In MESI protocol. Processors can do silent writing at an exclusive state and eliminate the BusRdX that occurs when doing write after reading.

Graph 8.1

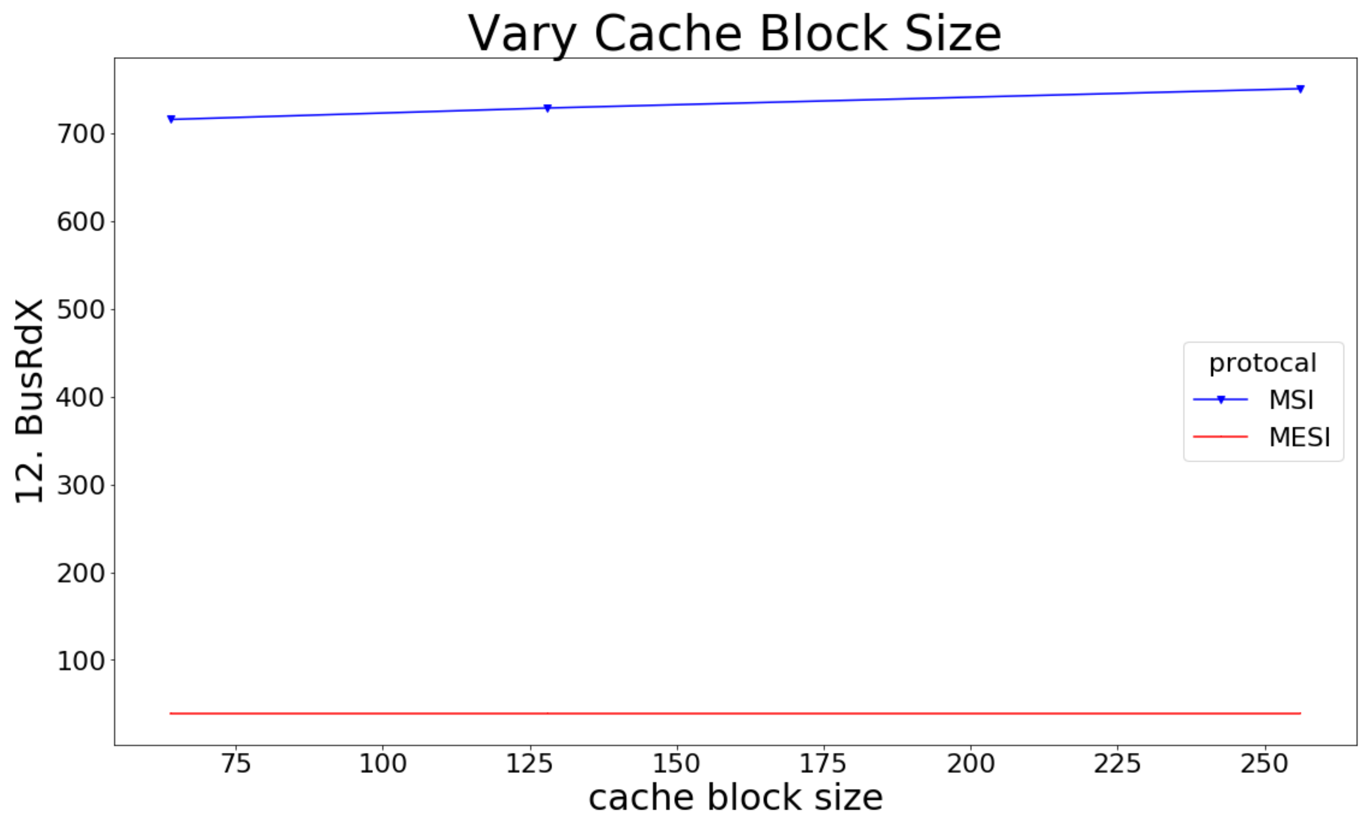


s

Graph 8.2



Graph 8.3


Vary Cache Block Size
12. BusRdX
700
600
500
400
300
200
100
75
100
125
150
175
200
225
250
cache block size
protocal
MSI
MESI