Prefetcher interaction with cache replacement policies

Sumon Nath

December 2021

1 Interaction of prefetcher with cache

Prefetchers learn cache access patterns, predicts future accesses and puts them into the cache. Whereas cache replacement policies decide which block to evict from the cache when it is full to make room for new blocks. So we need a holistic approach where the cache replacement policies and the prefetcher interacts with each other and works together as a single unit. From the perspective of this unit, the cache events that are of interest are:

- cache fill
- cache hit
- cache eviction

We will be looking into cache eviction first.

1.1 Cache Eviction

We will be denoting a prefetched cache block which has not yet been demanded by the core as P and a cache block which has been brought in response to a demand request to the cache as C. The various eviction scenarios possible with respect to P and C are:

- C evicts P
- P evicts P
- P evicts C
- C evicts C

We will be looking into each of these scenarios in detail to characterize the interactions between prefetcher and cache replacement policies. We are not concerned about the scenario where C evicts C as it is a necessary event.

P evicts C

1. P evicts a dead C, where P gets a hit later. This can be characterized as a positive interaction as a useless C block is replaced by a useful P block.
2. P evicts a non-dead C, where P gets a hit later. As C is non-dead it will be refilled into the cache at a later point of time. If C evicts the aforementioned P before it is used, then the whole interaction will be negative. On the other hand if C evicts the aforementioned P after P is used, then the whole interaction is neutral.

3. P evicts a non-dead C, where P does not get a hit later. This can be characterized as a negative interaction as a useful block is replaced by a useless block.

4. P evicts a dead C, where P does not a get a hit later. This interaction is neutral.

C evicts P

1. A dead C evicts P, where P gets a hit later. This can be characterized as a negative interaction as a useless C block replaces a a useful P block.

2. A dead C evicts P, where P does not get a hit later. This interaction is neutral.

3. A non-dead C evicts P, where P gets a hit later. This can be characterized as a neutral interaction.

4. A non-dead C evicts P, where P does not get a hit later. This can be characterized as a positive interaction as a useful C block replaces a a useless P block.

P_1 evicts P_2

We assume P_1 as the prefetch block that is filled into the cache and P_2 as the prefetch block that is evicted from the cache.

1. P_1 gets a hit later and P_2 does not get a hit later. This interaction is positive.

2. P_1 gets a hit later and P_2 also gets a hit later. This interaction is neutral.

3. P_1 does not get a hit later and P_2 also does not get a hit later. This interaction is also neutral.

4. P_1 does not get a hit later and P_2 gets a hit later. This interaction is negative.